Arnold E. Kiv
Serhiy O. Semerikov
Vladimir N. Soloviev
Andrii M. Striuk
(Eds.)

Computer Science & Software Engineering

Proceedings of the 2nd Student Workshop, CS&SE@SW 2019

Kryvyi Rih, Ukraine
November 29, 2019

Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (Eds.): Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, online

This volume represents the proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), held in Kryvyi Rih, Ukraine, in November 29, 2019. It comprises 2 invited talks and 17 contributed papers that were carefully peer-reviewed and selected from 30 submissions. The accepted papers present the ideas and early results of master's and PhD projects.

Program Committee

Albert A. Azaryan, Kryvyi Rih National University, Kryvyi Rih, Ukraine
Vitor Basto-Fernandes, University Institute of Lisbon (ISCTE-IUL), Portugal
Helena Fidlerová, Slovak University of Technology, Slovakia
Dmitry I. Ignatov, Higher School of Economics, Moscow, Russia
Arnold E. Kiv, Ben-Gurion University of the Negev, Beer Sheba, Israel
Piotr Lipinski, Lodz University of Technology, Poland
Lukas Pichl, International Christian University, Japan
Antonii Rzheuskyi, Lviv Polytechnic National University, Ukraine
Yaroslav V. Shramko, Kryvyi Rih State Pedagogical University, Kryvyi Rih, Ukraine
Oleksandr V. Spivakovskiy, Kherson State University, Kherson, Ukraine
Aleksandr D. Uchitel, National Metallurgical Academy of Ukraine, Dnipro, Ukraine
Grygoriy Zholtkevych, V. N. Karazin Kharkov National University, Ukraine

Additional Reviewers

Yuksel Celik, Karabuk University, Turkey
Roman Danel, Institute of Technology and Business in České Budějovice, Czechia
Emre Erturk, Eastern Institute of Technology, New Zealand
Pavlo Hryhoruk, Khmelnytskyi National University, Ukraine
Oleksii Ignatenko, Institute of Software Systems, Ukraine
Oleksandr Kolgatin, H. S. Skovoroda Kharkiv National Pedagogical University, Ukraine
Andrey Kupin, Kryvyi Rih National University, Ukraine
Ramkumar Lakshminarayanan, Sur College of Applied Sciences, Oman
Orken Mamyrbayev, Institute of Information and Computational Technologies, Kazakhstan
Sumitra Nuanmeesri, Suan Sunandha Rajabhat University, Thailand
Vasyl Oleksiuk, Ternopil Volodymyr Hnatiuk National Pedagogical University, Ukraine
Viacheslav Osadchyi, Bogdan Khmelnitsky Melitopol State Pedagogical University, Ukraine
Jaderick Pabico, University of the Philippines Los Baños, Philippines
Oleg Pursky, Kyiv National University of Trade and Economics, Ukraine
Zarema Seidametova, Crimean Engineering-Pedagogical University, Ukraine
Serhiy Semerikov, Kryvyi Rih State Pedagogical University, Ukraine
Vladimir Soloviev, Kryvyi Rih State Pedagogical University, Ukraine
Andrii Striuk, Kryvyi Rih National University, Ukraine
Nataliia Veretennikova, Lviv Polytechnic National University, Ukraine
Iryna Zaretska, V. N. Karazin Kharkov National University, Ukraine

Second Student Workshop on Computer Science & Software Engineering

Arnold E. Kiv[1], Serhiy O. Semerikov[2,3,4][0000-0003-0789-0272], Vladimir N. Soloviev[2][0000-0002-4945-202X] and Andrii M. Striuk[3][0000-0001-9240-1976]

[1] Ben-Gurion University of the Negev, P.O.B. 653, Beer Sheva, 8410501, Israel
kiv@bgu.ac.il
[2] Kryvyi Rih State Pedagogical University, 54, Gagarin Ave., Kryvyi Rih, 50086, Ukraine
{semerikov, vnsoloviev2016}@gmail.com
[3] Kryvyi Rih National University, 11, Vitalii Matusevych St., Kryvyi Rih, 50027, Ukraine
andrey.n.stryuk@gmail.com
[4] Institute of Information Technologies and Learning Tools of the NAES of Ukraine,
9, M. Berlynskoho Str., Kyiv, 04060, Ukraine

Abstract. This is an introductory text to a collection of selected papers from the Second Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), which was held in Kryvyi Rih, Ukraine, on the November 29, 2019. It consists of short summaries of selected papers and some observations about the event and its future.

Keywords: computer science, software engineering, student workshop.

1 CS&SE@SW 2019 at a glance

Student Workshop on Computer Science & Software Engineering (CS&SE@SW) is a peer-reviewed international Computer Science workshop focusing on research advances, applications of information technologies.

CS&SE@SW topics of interest since 2018 [5]:

- Computer Science (CS):
  - Theoretical computer science
    - Data structures and algorithms
    - Theory of computation
    - Information and coding theory
    - Programming language theory
    - Formal methods
  - Computer systems
    - Computer architecture and computer engineering
    - Computer performance analysis
    - Concurrent, parallel and distributed systems
    - Computer networks
    - Formal methods

    - Databases
  - Computer applications
    - Computer graphics and visualization
    - Human-computer interaction
    - Scientific computing
    - Artificial intelligence
- Software Engineering (SE):
  - Software requirements
  - Software design
  - Software construction
  - Software testing
  - Software maintenance
  - Software configuration management
  - Software engineering management
  - Software development process
  - Software engineering models and methods
  - Software quality
  - Software engineering professional practice
  - Software engineering economics
  - Computing foundations
  - Mathematical foundations
  - Engineering foundations

This volume represents the proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), held in Kryvyi Rih, Ukraine, on November 29, 2019. It comprises 2 invited talks, 5 CS-related and 10-SE related contributed papers that were carefully peer-reviewed and selected from 30 submissions. Each submission was reviewed by at least 2, and on the average 2.7, program committee members. The accepted papers present the state-of-the-art overview of successful cases and provides guidelines for future research.

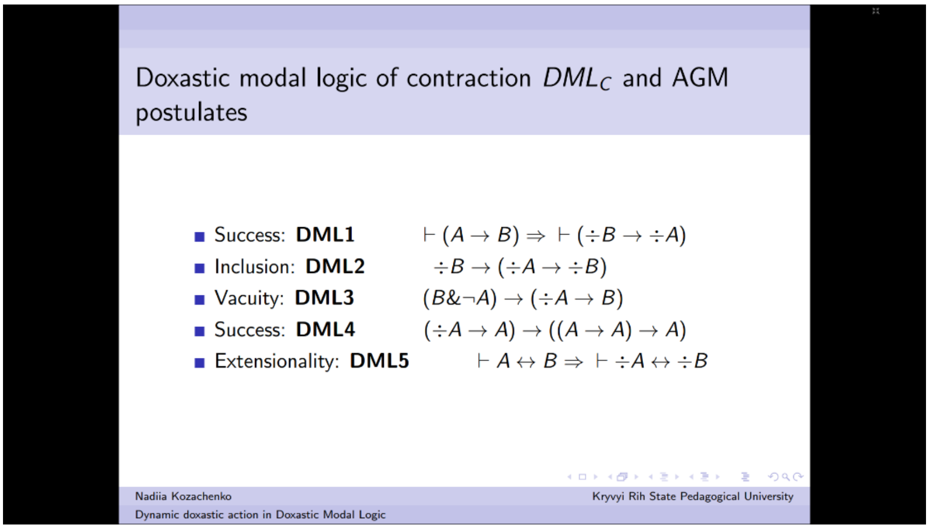
The volume is structured in three parts, each presenting the contributions for a particular workshop track.

2 Invited talks

Professor Nadiia Kozachenko (Fig. 1) is chair of Department of Philosophy at Kryvyi Rih State Pedagogical University. She got PhD in logic from the Institute of Philosophy of the NAS of Ukraine in 2010. The main directions of Dr. Kozachenko' research is logic and philosophy of information society.

In her talk (Fig. 2) Dr. Kozachenko trying to combine two main traditions of belief revision: the so-called AGM-approach and Dynamic Doxatic Logic (DDL) approach [9]. She consider doxastic actions as modal operators partly like DDL-style and compare their features with AGM postulates. Then construct axiomatical systems for the operator of the expansion and the operator of the contraction. Within the system based on this interpretation, we can express and prove the corresponding postulates of expansion and contraction AGM. It demonstrates that these modal operators correspond to the functions of expansion and contraction described in AGM, about that the representational theorem has been formulated.

Fig. 1. Dr. Nadiia Kozachenko



Doxastic modal logic of contraction DML_C and AGM postulates

- Success: **DML1** $\vdash (A \to B) \Rightarrow \vdash (\div B \to \div A)$
- Inclusion: **DML2** $\div B \to (\div A \to \div B)$
- Vacuity: **DML3** $(B \& \neg A) \to (\div A \to B)$
- Success: **DML4** $(\div A \to A) \to ((A \to A) \to A)$
- Extensionality: **DML5** $\vdash A \leftrightarrow B \Rightarrow \vdash \div A \leftrightarrow \div B$

Nadiia Kozachenko Kryvyi Rih State Pedagogical University

Dynamic doxastic action in Doxastic Modal Logic

Fig. 2. Presentation of paper [9]

Professor Andrii Striuk (Fig. 3) is chair of Department of Modelling and Software Engineering at Kryvyi Rih National University. He got PhD in educational technology from the Institute of Information Technologies and Learning Tools of the NAES of Ukraine in 2012. The main directions of Dr. Striuk' research is computer simulaton and software engineering education.

Fig. 3. Dr. Andrii Striuk

Serhiy Semerikov (Fig. 4) is professor of Department of Computer Science and Applied Mathematics at Kryvyi Rih State Pedagogical University. He got both PhD and DSc in education (informatics) from the National Pedagogical Dragomanov University in 2001 and 2009, respectively. The main directions of Dr. Semerikov' research is methods of learning and educational technology.

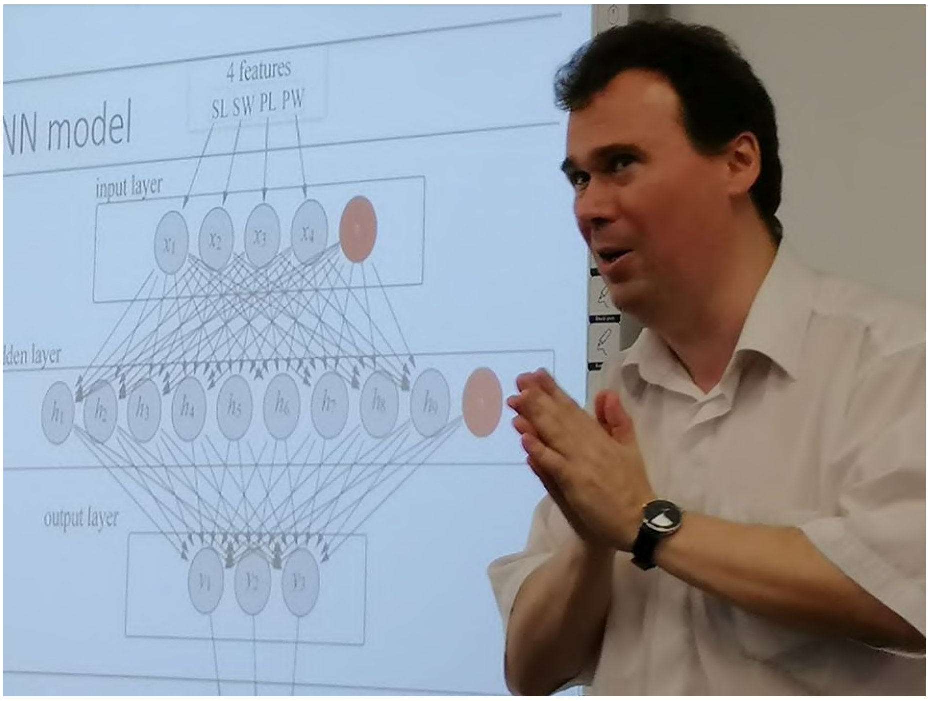


Fig. 4. Dr. Serhiy Semerikov

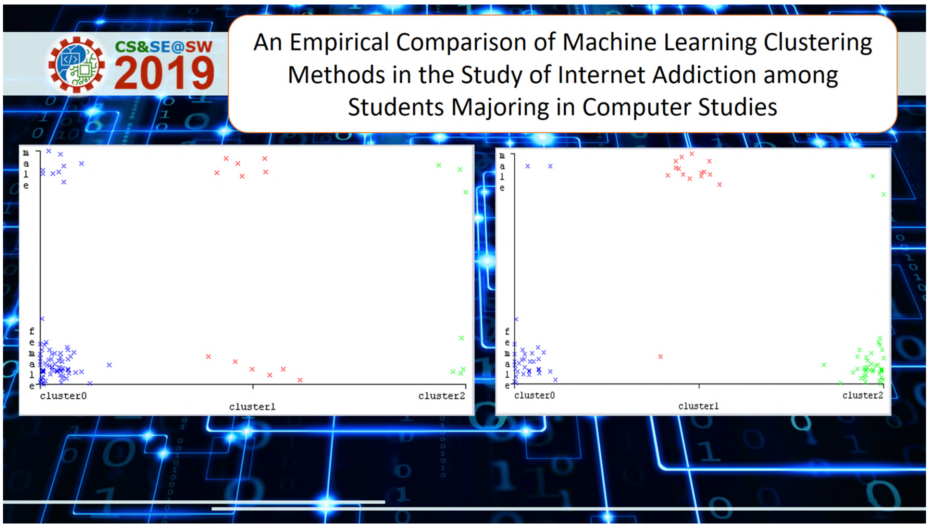
In a joint talk, Andrii Striuk and Serhiy Semerikov presented a retrospective analysis of the first model of training software engineers developed in the early 1970s (Fig. 5) and its compliance with the current state of software engineering development as a field of knowledge and a new the standard of higher education in Ukraine, specialty 121 "Software Engineering" [15]. It is determined that the consistency and scalability inherent in the historically first training program are largely consistent with the ideas of evolutionary software design. An analysis of its content also provided an opportunity to identify the links between the training for software engineers and training for computer science, computer engineering, cybersecurity, information systems and technologies. It has been established that the fundamental core of software engineers' training should ensure that students achieve such leading learning outcomes: to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of language, instrumental and computational tools for software engineering; know and apply the appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development; put into practice the software tools for domain analysis, design, testing, visualization, measurement and documentation of software. It is shown that the formation of the relevant competencies of future software engineers must be carried out in the training of all disciplines of professional and practical training.

CS&SE@SW
2019
2nd Computer Science & Software Engineering Student Workshop
The Dawn of Software Engineering Education
METIHBECT
HETES

Fig. 5. Presentation of paper [15]

3 Computer Science

Oksana Klochko and Vasyl Fedorets present an empirical comparison of machine learning clustering methods in the study of Internet addiction among students majoring in Computer Sciences (Fig. 6). The article [6] describes the technology of empirical comparison of methods of clustering problem solving using WEKA free software for machine learning. Empirical comparison of data clustering methods was based on the results of a survey conducted among students majoring in Computer Sciences and dedicated to detecting signs of Internet addiction as behavioural disorder that occurs due to Internet misuse. Empirical comparison of Expectation Maximization, Farthest First and K-Means clustering algorithms together with the application of the WEKA machine learning system had the following results. It described the peculiarities of application of these methods in feature clustering. The authors developed data instances' clustering models to detect signs of Internet addiction among students majoring in Computer Sciences. The study concludes that these methods may be applicable to development of models detecting respondent groups with signs of Internet addiction related disorders.



CS&SE@SW 2019

An Empirical Comparison of Machine Learning Clustering Methods in the Study of Internet Addiction among Students Majoring in Computer Studies

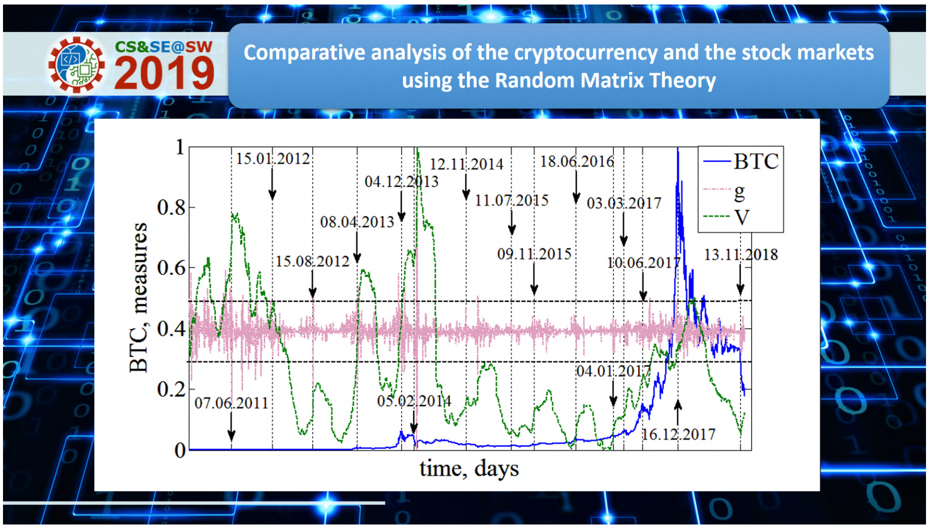
Fig. 6. Presentation of paper [6]

Kateryna Ovchar, Andrii Borodin, Ivan Burlachenko and Yaroslav Krainyk in the paper [11] present an alternative approach to automatic recognition and sorting of the agricultural objects (Fig. 7). The objects are sorted, based on their color. The sorting algorithm works with pictures, captured by a web camera, pre-processes them, determines the mass centers of the recognized objects, and calculates distances to the points which were found. Afterward, the recognized objects are sorted by a hand-manipulator with four degrees of freedom. The manipulator is powered by Arduino Uno Rev 3 and servo motors of two types. The sorting algorithm has been implemented in Python, with the use of the OpenCV library. For image pre-processing, the spatial anti-aliasing, filtering, and morphological opening and closing have been used. Color maps have been created to determine the mass centers of the recognized objects. To calculate the rotation angle of the manipulator's elbow based on the shoulder position, linear regression model is trained. The practical value of the research and development results is the possibility of their use in the harvesting of the agricultural objects with further extend to manufacture with conveyor sorting tapes.

The article [14] of Vladimir N. Soloviev, Symon P. Yevtushenko and Viktor V. Batareyev demonstrates the comparative possibility of constructing indicators of critical and crash phenomena in the volatile market of cryptocurrency and developed stock market (Fig. 8). Then, combining the empirical cross-correlation matrix with the Random Matrix Theory, authors mainly examine the statistical properties of cross-correlation coefficients, the evolution of the distribution of eigenvalues and corresponding eigenvectors in both markets using the daily returns of price time series. The result has indicated that the largest eigenvalue reflects a collective effect of the whole market, and is very sensitive to the crash phenomena. It has been shown that introduced the largest eigenvalue of the matrix of correlations can act like indicators-predictors of falls in both markets.

METIHBECT
The Multi-Agent System of the
of the Agricultural Objects
Kateryna Ovchar, Andrii Borodin
computer engineering

Fig. 7. Presentation of paper [11]



CS&SE@SW 2019

Comparative analysis of the cryptocurrency and the stock markets using the Random Matrix Theory

Fig. 8. Presentation of paper [14]

Andrii O. Tarasenko, Yuriy V. Yakimov and Vladimir N. Soloviev in paper [16] shows the theoretical basis for the creation of convolutional neural networks for image classification and their application in practice (Fig. 9). To achieve the goal, the main types of neural networks were considered, starting from the structure of a simple neuron to the convolutional multilayer network necessary for the solution of this problem. It shows the stages of the structure of training data, the training cycle of the network, as well as calculations of errors in recognition at the stage of training and verification. At the end of the work the results of network training, calculation of recognition error and training accuracy are presented.

METIHBECT
MINISTRY OF EDUCATION AND SCIENCE OF UKRAIN
KRIVYI RIG STATE PEDAGOGICAL UNIVERSITY
DEPARTMENT OF INFORMATICS AND APPLIED MATHEMAT
Convolutional neural netwo
for image classification
Kriviy Rih 2019

Fig. 9. Presentation of paper [16]

Svitlana O. Yaroshchuk, Nonna N. Shapovalova, Andrii M. Striuk, Olena H. Rybalchenko, Iryna O. Dotsenko and Svitlana V. Bilashenko in [18] present the model for scoring assessment of microfinance institution borrowers, which allows to increase the efficiency of work in the field of credit (Fig. 10). The result of the work is a neural network scoring model with high accuracy of calculations, an implemented system of automatic customer lending.

4 Software Engineering

The article of Andrii O. Priadko, Kateryna P. Osadcha, Vladyslav S. Kruhlyk and Volodymyr A. Rakovych [12] escribes the process of developing a chatbot to provide students with information about schedules using the Telegram mobile messenger (Fig. 11). During the research, the following tasks have been performed: the analysis of notification systems for their use in the educational process, identification of problems of notifying students about the schedule (dynamic environment, traditional presentation of information, lack of round-the-clock access), substantiation of the choice of mobile technologies and Telegram messenger, determination of the requirements to the software, generalization of the chatbot functioning features, description of the structure, functionality of the program to get information about the schedule using a chatbot. The following tasks have been programmatically implemented: obtaining data from several pages of a spreadsheet (faculty / institute, red / green week, group number, day of the week, period number, discipline name, information about the teacher); presentation of data in a convenient form for the messenger (XML); implementation of the mechanism of convenient presentation of data in the messenger (chatbot). Using Python and the Telegram API, the software has been designed to increase students; immediacy in getting the information about the schedules, minimizing the time spent, and optimizing of planning of student activities and higher education institution functioning.

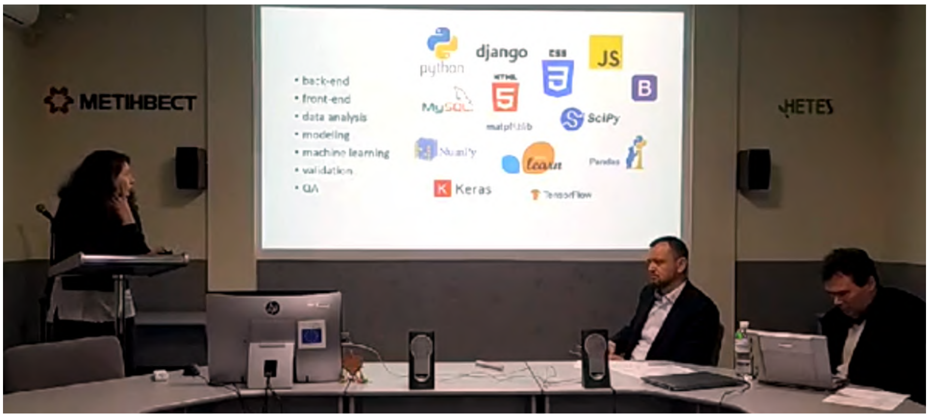


Fig. 10. Presentation of paper [18]

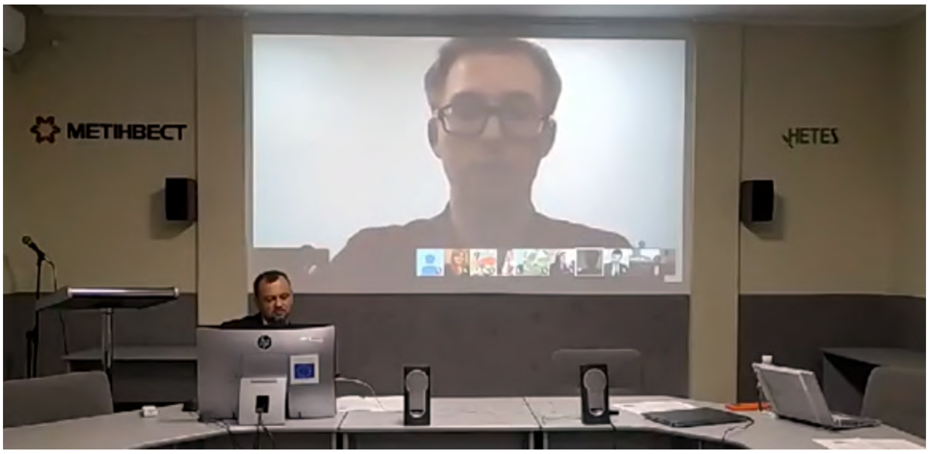


Fig. 11. Presentation of paper [12]

Galina Kirichek, Vladyslav Harkusha, Artur Timenko and Nataliia Kulykovska [4] present the realization method of attacks and anomalies detection with the use of training of ordinary and attacking packages, respectively (Fig. 12). The method that was used to teach an attack on is a combination of an uncontrollable and controlled neural network. In an uncontrolled network, attacks are classified in smaller categories, taking into account their features and using the self-organized map. To manage clusters, a neural network based on back-propagation method used. Authors use PyBrain as the main framework for designing, developing and learning perceptron data. This framework has a sufficient number of solutions and algorithms for training, designing and testing various types of neural networks. Software architecture is presented using a procedural-object approach. Because there is no need to save intermediate result of the program (after learning entire perceptron is stored in the file), all the progress of learning is stored in the normal files on hard disk.

Fig. 12. Presentation of paper [4]

In the paper [3], Ruslan Cherniavskyi, Yaroslav Krainyk and Anzhela Boiko (Fig. 13) establish an investigation on the development of university 3D-model and its possible applications for educational and research fields. Authors assume that 3D-model of university can help in various scenarios and should be used to adopt modern immersing technologies into to university processes. Different means are employed for the development of the model. Bottom-up approach for using these means and their connection with each other are shown in the work. Then, details of the 3D-model design process are provided with peculiarities related to the university building location and corpuses positions. Finally, assembled models of university are shown in 3ds Max and

Unity environments. In the final part of the paper, authors suggest scenarios of model usage for educational and research fields. Universities can gain various benefits from integrating their research efforts to employ new technology and identify new development opportunities for both science and education in university. In case of the developed 3D-model, it is planned to use it in the projects connected with client-server applications, Internet-of-Things, Smart Grid, etc. In the educational process it will be a part of case-studies for learning 3D-modeling, development in Unity environment, training for emergency situations.

Fig. 13. Presentation of paper [3]

The article of Nadiia Balyk, Vasyl Oleksiuk and Anatolii Halas [2] explores the issues of designing and developing a support system for corporate network users (Fig. 14). The authors analyzed existing user support systems, including the concepts of ServiceDesk and HelpDesk. The study describes the process of designing and developing the software “Network user support tool” (NetSupport). This tool stores data in the Google Calendar Cloud Service. The results of the study are: analysis of the functioning of the Google Calendar service, review of the API for working with Google cloud services, development and testing of software to support the work of users. Functionality of Network user support tool allows you to create events on your system administrator or support teams` calendar with a detailed description of the user problem. The article describes the process of developing a network support tool using C# 6.0 programming language and .NET 4.6 technologies. The developed software has been tested in Ternopil Volodymyr Hnatiuk National Pedagogical University.

In this study [13] Oleg Pursky, Anna Selivanova, Tetiana Dubovyk and Tamara Herasymchuk present the software development method for e-trade business process management information system (Fig. 15). As a software platform for implementation of the e-trading business process management system has been selected MSSQL database and software component implemented with the use of ASP.NET MVC Framework. The developed Web-application includes the following functional blocks: user authentication module; administration module; block implementing the functions of domain mathematical model and information processing procedures; database intended to store initial data, trading activity indicators and calculation results, as well as the content of the Web-application interface components; logistics module; server end, which implements the procedures for the database access, the formation of the view modes of the models and functions of the domain mathematical model; interface that provides an interactive user access to the Web-application features. The connection between the Web-application and the database is performed on the basis of the GET http-protocol method and is provided by generating database queries and query results. The developed e-trade business process management information system has a three-level software and hardware structure and is designed to manage the processes of interaction between the online store and on-line consumers.

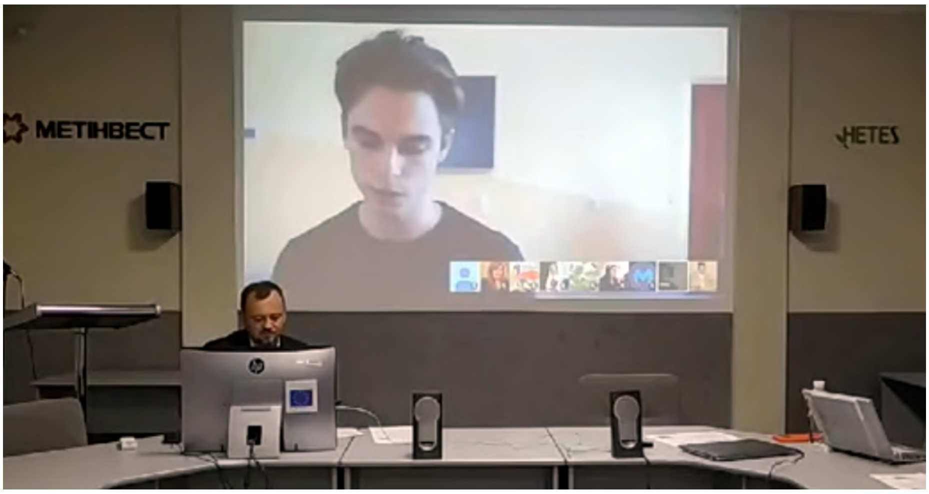
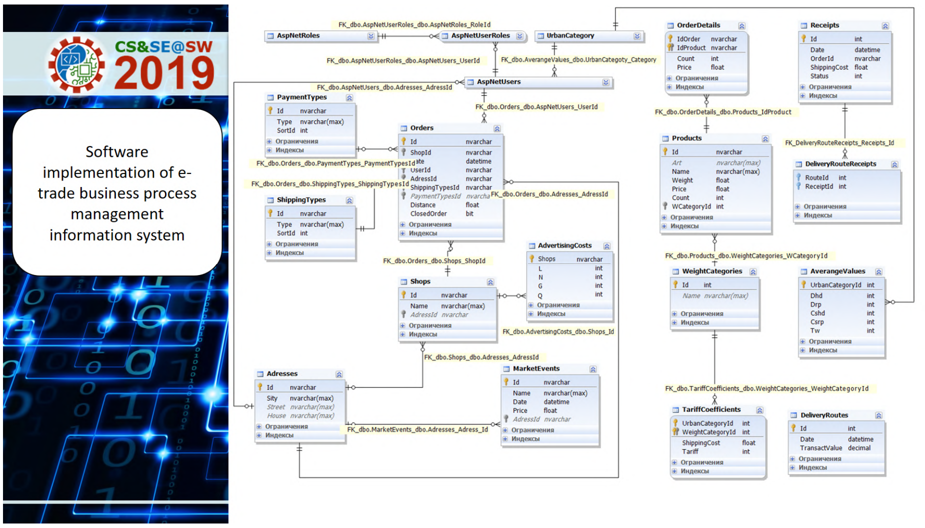


Fig. 14. Presentation of paper [2]

In the article [8] of Olha V. Korotun, Tetiana A. Vakaliuk and Viacheslav A. Oleshko the database was designed and implemented in accordance with the requirements of the relational model, which ensures the storage and collective access to the information of the auto-filling system and CMS WordPress data (Fig. 16). Algorithms of system functioning were developed, the order of interaction of classes during program code execution was determined, as a result of which the application was implemented. Template Method architectural pattern was chosen to implement the web-based automatic content filling system. The following tools and technologies were selected to create the software package HTML markup language for HTML documents; programming language PHP; MySQL database management environment; Apache web server; the OpenServer package. The algorithms of the basic processes of content filling automation were considered and the interaction of the system classes during the processes of parsing, filtering and storing of information were analyzed. The developed system does not require specialized hardware, additional settings and deployment tools other than the standard ones for such plugins. This application is mostly for the site administrator and does not have user interface. That is why the features of the plugin automation system configuration interface; RSS feeds view and management interface, as well as the RSS feed configuration interface are described in detail. In the future, this system can be improved by introducing new functionality and improving the algorithm for reading data.



CS&SE@SW 2019

Software implementation of e-trade business process management information system

Fig. 15. Presentation of paper [13]

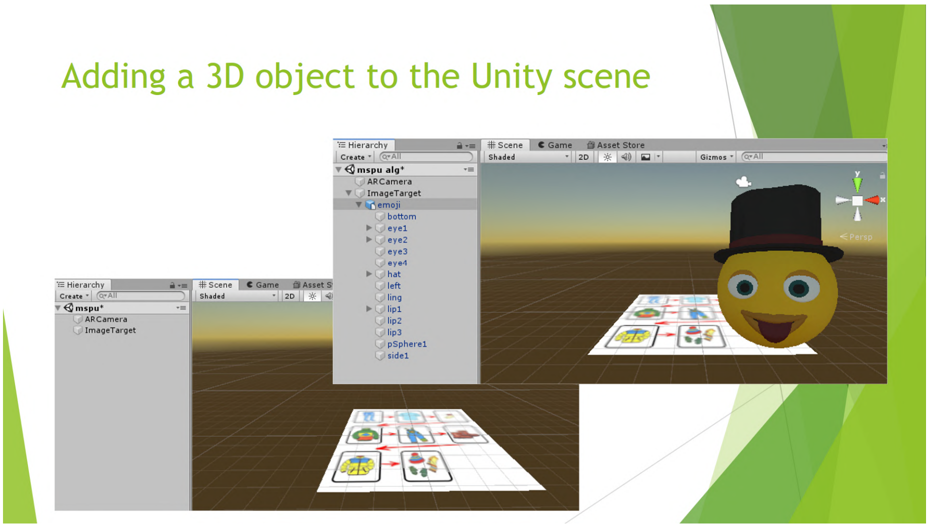
Fig. 16. Presentation of paper [8]

The article [1] of Nadiia Balyk, Svitlana Leshchuk and Dariia Yatsenyak is devoted to designing a smart home educational model (Fig. 17). The authors analyzed the literature in the field of the Internet of Things and identified the basic requirements for the training model. It contains the following levels: command, communication, management. The authors identify the main subsystems of the training model:

communication, signaling, control of lighting, temperature, filling of the garbage container, monitoring of sensor data. The proposed smart home educational model takes into account the economic indicators of resource utilization, which gives the opportunity to save on payment for their consumption. The hardware components for the implementation of the Mini Smart House were selected in the article. It uses a variety of technologies to conveniently manage it and use renewable energy to power it. The model was produced independently by students involved in the STEM project. Research includes sketching, making construction parts, sensor assembly and Arduino boards, programming in the Arduino IDE environment, testing the functioning of the system. Research includes sketching, making some parts, assembly sensor and Arduino boards, programming in the Arduino IDE environment, testing the functioning of the system. Approbation Mini Smart House researches were conducted within activity the STEM-center of Physics and Mathematics Faculty of Ternopil Volodymyr Hnatiuk National Pedagogical University, in particular during the educational process and during numerous trainings and seminars for pupils and teachers of computer science.

Fig. 17. Presentation of paper [1]

The article of Alla Kompaniets, Hanna Chemerys and Iryna Krasheninnik [7] is devoted to the theoretical consideration of the problem and the use of innovative technologies in the educational process in the educational establishment of secondary education in the process of studying the school course of computer science (Fig. 18). The main advantages of using educational simulators in the educational process are considered, based on the new state standard of basic and complete general secondary education. Based on the analysis of scientific and methodological literature and network sources, the features of the development of simulators for educational purposes are described. Innovative tools for simulator development have been investigated, as augmented reality with the use of three-dimensional simulation. The peculiarities of using a simulator with augmented reality when studying the topic of algorithmization in the course of studying a school computer science are considered. The article also describes the implementation of augmented reality simulator for the formation of algorithmic thinking skills by students, presents the results of development and describes the functionality of the software product. In the further prospects of the study, it is planned to conduct an experimental study to determine the effectiveness of the use of software development in the learning process.



Adding a 3D object to the Unity scene

Fig. 18. Presentation of paper [7]

The article of Bohdan Oliinyk and Vasyl Oleksiuk [10] considers the problem of test automation software (Fig. 19). The authors analyze testing tasks that can be automated. They also cite cases where the use of automation is inappropriate. The key factors of using automation are time and cost savings. According to the authors, the advantages of automated tests are: the ability to check the latest changes in the application (regression testing), speed of execution, saving the time of testers, the ability to create self-tests by developers. The disadvantages of automatic tests are: insufficient reliability, need for support, fewer errors detected, a false sense of product quality. The following processes are identified, which can be automated: background processes, file logging, database entry, registration and payment systems, load tests, data entry operations, long-end-to-end scripts, checking complex mathematical calculations, checking correct search. The article provides statistics on the use of programming languages for developing automated tests. A comparative analysis of ready-made software products for automated testing is offered. Based on research analysis and experience, the authors believe that human intelligence is always required to validate the program. So, the authors justify the need to perform a manual and automated test.

The article [17] of Yaroslav Vasylenko, Galina Shmyger and Dmytro Verbovetskyi investigate the principles and technologies of creating such a semantic interconnection system that would be useful and practical for use in areas such as machine translation, search engines and contextual search (Fig. 20). The practical meaning of the results obtained is to create a semantic dictionary of the Ukrainian language that will allow to better analyze Ukrainian texts by searching not only the words themselves, but also words that are in one way or another related to the primary, and that will significantly increase the speed of search and analysis of information. In the created web-application (thesaurus) the basic functions of similar existing systems and the latest methods of information linguistics are implemented.

Fig. 19. Presentation of paper [10]

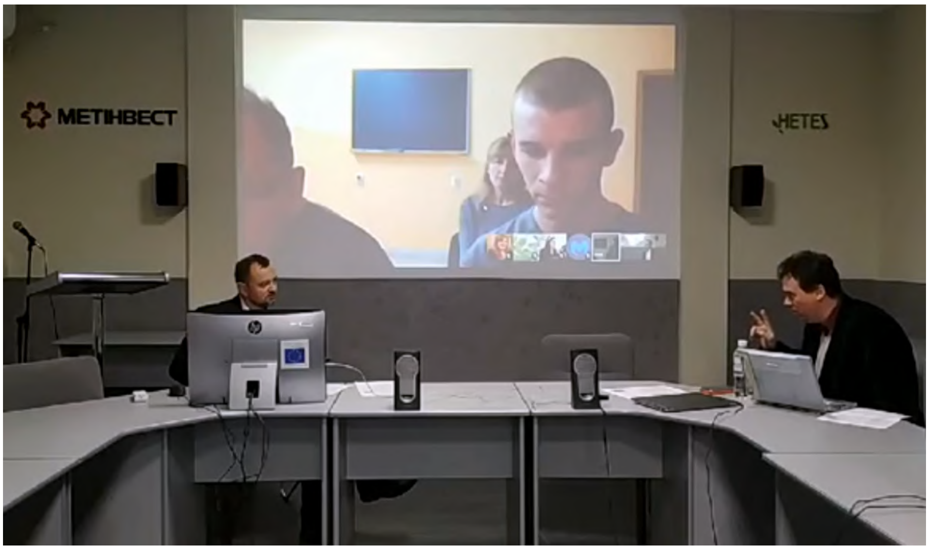


Fig. 20. Presentation of paper [17]

5 Conclusion

The vision of the CS&SE@SW 2019 is provides an expert environment for young researchers, who are at the beginning of their career, such as Master or PhD students, to present and discuss the most recent of ideas and early results of student's projects or other research aiming at receiving a Master or PhD degrees.

Young researchers, who will join us to take part in discussions and/or present their papers, will be offered an opportunity to exchange and discuss their research ideas with their peers, supervisors, and senior scientists working in the fields that are within the scope of CS&SE@SW.

The second instalment of CS&SE@SW was organised by Kryvyi Rih National University, Ukraine (with support of the rector Mykola I. Stupnik) in collaboration with Kryvyi Rih State Pedagogical University, Ukraine (with support of the rector Yaroslav V. Shramko), Institute of Information Technologies and Learning Tools of the NAES of Ukraine (with support of the director Valeriy Yu. Bykov) and Ben-Gurion University of the Negev, Israel (with support of the rector Chaim J. Hames).

We are thankful to all the authors who submitted papers and the delegates for their participation and their interest in CS&SE@SW as a platform to share their ideas and innovation. Also, we are also thankful to all the program committee members for providing continuous guidance and efforts taken by peer reviewers contributed to improve the quality of papers provided constructive critical comments, improvements and corrections to the authors are gratefully appreciated for their contribution to the success of the workshop.

We hope you enjoy this workshop and meet again in more friendly, hilarious, and happiness of further CS&SE@SW 2020 at Kryvyi Rih, Ukraine on November 27, 2020.

References

1. Balyk, N., Leshchuk, S., Yatsenyak, D.: Developing a Mini Smart House model. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 198–212, online (2019, in press)
2. Balyk, N., Oleksiuk, V., Halas, A.: Development a computer network user support tool. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 159–170, online (2019, in press)
3. Cherniavskyi, R., Krainyk, Y., Boiko, A.: Modeling university environment: means and applications for university education. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 149–170, online (2019, in press)
4. Kirichek, G., Harkusha, V., Timenko, A., Kulykovska, N.: System for detecting network anomalies using a hybrid of an uncontrolled and controlled neural network. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 138–148, online (2019, in press)
5. Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M.: First student workshop on computer science & software engineering. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 1st Student Workshop on Computer Science &

Software Engineering (CS&SE@SW 2018), Kryvyi Rih, Ukraine, November 30, 2018. CEUR Workshop Proceedings **2292**, 1–10. http://ceur-ws.org/Vol-2292/paper00.pdf (2018). Accessed 31 Dec 2018

6. Klochko, O., Fedorets, V.: An empirical comparison of machine learning clustering methods in the study of Internet addiction among students majoring in Computer Sciences. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 58–75, online (2019, in press)
7. Kompaniets, A., Chemerys, H., Krasheninnik, I.: Using 3D modelling in design training simulator with augmented reality. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 213–223, online (2019, in press)
8. Korotun, O.V., Vakaliuk, T.A., Oleshko, V.A.: Development of a web-based system of automatic content retrieval database. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 182–197, online (2019, in press)
9. Kozachenko, N.: Dynamic doxastic action in Doxastic Modal Logic. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 21–34, online (2019, in press)
10. Oliinyk, B., Oleksiuk, V.: Automation in software testing, can we automate anything we want? In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 224–234, online (2019, in press)
11. Ovchar, K., Borodin, A., Burlachenko, I., Krainyk, Y.: Automated recognition and sorting of agricultural objects using multi-agent approach. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 76–86, online (2019, in press)
12. Priadko, A.O., Osadcha, K.P., Kruhlyk, V.S., Rakovych, V.A.: Development of a chatbot for informing students of the schedule. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 128–137, online (2019, in press)
13. Pursky, O., Selivanova, A., Dubovyk, T., Herasymchuk, T.: Software implementation of e-trade business process management information system. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 171–181, online (2019, in press)
14. Soloviev, V.N., Yevtushenko, S.P., Batareyev, V.V.: Comparative analysis of the cryptocurrency and the stock markets using the Random Matrix Theory. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 87–100, online (2019, in press)

15. Striuk, A.M., Semerikov, S.O.: The Dawn of Software Engineering Education. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 35–57, online (2019, in press)
16. Tarasenko, A.O., Yakimov, Y.V., Soloviev, V.N.: Convolutional neural networks for image classification. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 101–114, online (2019, in press)
17. Vasylenko, Y., Shmyger, G., Verbovetskyi, D.: Some aspects of designing of the structural semantics visualization system. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 235–248, online (2019, in press)
18. Yaroshchuk, S.O., Shapovalova, N.N., Striuk, A.M., Rybalchenko, O.H., Dotsenko, I.O., Bilashenko, S.V.: Credit scoring model for microfinance organizations. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 2nd Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2019), Kryvyi Rih, Ukraine, November 29, 2019, CEUR-WS.org, pp. 115–127, online (2019, in press)

Dynamic doxastic action in Doxastic Modal Logic

Nadiia Kozachenko[0000−0003−2358−9076]

Kryvyi Rih State Pedagogical University,
54, Gagarin Ave., Kryvyi Rih, Ukraine, 50086
n.p.kozachenk@gmail.com

Abstract. This article is a trying to combine two main traditions of belief revision: the so-called AGM-approach and Dynamic Doxatic Logic approach. We consider doxastic actions as modal operators partly like DDL-style and compare their features with AGM postulates. We construct axiomatical systems for the operator of the expansion and the operator of the contraction. Within the system based on this interpretation, we can express and prove the corresponding postulates of expansion and contraction AGM. It demonstrates that these modal operators correspond to the functions of expansion and contraction described in AGM, about that the representational theorem has been formulated.

Keywords: Belief revision, AGM, DDL, Modal logic, Dynamic modal logic, Doxastic action, Epistemic logic, Doxastic logic, Belief change.

1 Dynamic interpretation of doxastic actions

Belief revision is promising trend in modern epistemic logic deals with changes of our knowledges and beliefs. It is aimed at formal representation of the process of belief change. There are two main traditions of belief revision: the so-called *AGM-approach*, named after its initiators Carlos E. Alchourròn, Peter Gärdenfors and David Makinson (see their most seminal paper [1] and inspired reserches, e.g.: [4]), and *DDL-approach* — Dynamic Doxatic Logic based on ingenious ideas of Krister Segerberg (see, e.g.: [12]). Both approaches are very similar in a general intention and research methodology, furthermore their main purpose is an adequate representation of doxastic actions perfomed in the process of belief change. In both traditions the basic doxastic actions represented by *expansion*, *contraction* and *revision*, but each approach implements them in different ways. This ways depend on the format of main epistemic objects such as epistemic states, admissible epistemic inputs, expected epistemic results etc. Implemetation of doxastic actions also depends on specific notion of changes in belief states — it could be taken as a process (see, e.g.: [2]) or as a result of change (see, e.g.: [3]).

AGM. The main postulates of belief change were described in AGM, so this treatment sometimes is called "postulational approach". Basic properties of expansion, contraction and revision are presented in AGM as a set of formal axioms. It is commonly accepted that any interpretation of doxastic action should satisfy the suitable set of the axioms. However, main drawback of this approach is a static representation of doxastic actions. All doxastic actions in AGM factually are reduced to operations over theories.

AGM expansion: T + A (T — initial knowledge set (theory), + — the sign of expansion and A — new knowledge (or belief) added to the initial set).

AGM contraction: T $\div$ A (T — initial knowledge set (theory), $\div$ — the sign of contraction and A — knowledge (or belief) extracted from the initial set).

AGM considers the status of the theory before and after change, but it doesn't consider the process of change. Thus an underwater part of iceberg of properties of doxastic operations are remained unexplored.

DDL. The idea of constructing an axiomatic system capable to display the dynamic properties of doxastic actions is partially implemented by the dynamic direction in belief revision. The basis of this approach is the idea of representing the dynamic aspect of belief change by specific language of DDL. Doxasic actions are considered as term-formed operators +, $\div$, *.

DDL expansion: $+\phi$ (means it is performing an expansion by ϕ).

DDL contraction: $\div\phi$ (means it is performing a contraction by ϕ).

These terms constitute entire formula with additional operators "[ ]", "<>". Thus, notion $\psi[+\phi]$ means "ψ regularly holds after the agent performs an expansion by ϕ" and $\psi < +\phi >$ means "perhaps ψ holds after the agent performs an expansion by ϕ"

This language has much more resources for a detailed analysis of doxastic actions, but it is too complicated and cumbersome. Whereas the postulational way allows to define doxastic action and describes the basic principles of belief change, it has too simple syntax that makes difficult further detailed researching. Therefore it is reasonable to construct an axiomatic logical system capable to combine advantages of both approaches and overcome their shortcomings. The syntax of this logic should be simple like the syntax of the AGM-system and should be flexible enough to express the process of change.

To solve this task it need to change the aspect of consideration of doxastic actions. In this connection Yaroslav Shramko makes an interesting offer, proposed a more depth consideration of the epistemological grounds of beliefs. He offers to interpret doxastic actions as pure modal operators to relegating the action directly into the center of researching [13]. The proposed idea differs from the modal interpretation of doxastic actions by K. Segerbeg (DDL-approach). If doxastic action (expansion, contraction and etc) is considered as regular modal operator, its application to a Boolean formula creates a dynamic formula. Whereas doxastic operator in DDL forms only term (see: [12], [7]) and thus it is not a purely modal operator. The representation of doxastic actions as formula-formed modal operators considerably simplify syntax and with that allows to analyze the dynamic aspect of the belief revision. It is possible to build specific dynamic modal logic on this notion.

The basic idea of this paper is to represent expansion and contraction actions as modal operators and to show the equivalence of received operators to appropriate operators AGM. We need to consider the AGM postulates for expansion and contraction and to prove that they hold doxastic actions interpreted in the modal way.

Within the AGM belief state of the subject is reduced to a set of beliefs closed under the logical consequence. After the AGM under a consequence operation we mean an operator Cn that takes sets of proposition to sets of proposition, such that three condition are satisfied, for any sets X and Y of proposition: $X \subseteq Cn(X)$, $Cn(X) = Cn(Cn(X))$, and $Cn(X) \subseteq Cn(Y)$ whenever $X \subseteq Y$ [1, p. 511]. Thus we will use the equivalent notation $y \in Cn(X)$ and $X \vdash y$.

Each set of beliefs can represent the belief state, as well as any belief state may be presented as a set of beliefs. Each proposition represents a belief and, accordingly, each belief can be represented in the form of a proposition. That is, the belief state is a set of propositions – beliefs, in which the subject is convinced of the truth. We will examine the change of a theory – consistent sets of beliefs, closed under the logical consequence (Cn).

2 AGM postulates of expansion

Expansion – the simplest doxastic operator in AGM. Its point lies in adding new information to the initial set of beliefs. Expansions is implemented by using $+$, displaying a pair (set of beliefs, statements) on the set of sets of beliefs ($K \times L \to K$). Where K is the initial belief set, the expansion of K by A is denoted by $K + A$. Expansion can be characterized by the following set of postulates [6, p. 48-51].

Closure	$K + A$ – belief set	**E 1**
Success	$A \in (K + A)$	**E 2**
Inclusion	$K \supset (K + A)$	**E 3**
Vacuity	If $A \in K$ then $(K + A) = K$	**E 4**
Monotonicity	If $K \supset H$ then $(K + A) \supset (H + A)$	**E 5**

These postulates describe a family of expansion operators. Factually, with the help of the postulates of expansion E1-E5 can be axiomatically defined the operator of expansion. That is, if the doxastic operator satisfying the postulates of expansion E1-E5, then it is equivalent to the operator expansion of AGM.

3 AGM postulates of contraction

Contraction is applied in those cases where a proposition must be removed from the belief set without giving any new information. Contraction is implemented by using $\div$, displaying a pair (set of beliefs, statements) on the set of sets of beliefs ($K \times L \to K$). Where K is the initial belief set, the contraction of K by

A is denoted by $K \div A$. Contraction can be characterized by the following set of postulates [6, p. 60-64].

Basic postulates of contraction, which factually mean the axiomatic definition contraction, were developed under the traditional approach of AGM [1]. Building a contraction operator is considered to be successfull if it exists a proving that it satisfied the postulates of contraction AGM.

Closure	$K \div A$ – belief set	**C 1**
Success	If $\nvdash A$ then $K \div A \nvdash A$	**C 2**
Inclusion	$K \div A \supset K$	**C 3**
Vacuity	If $A \notin K$ then $K \div A = K$	**C 4**
Extensionality	If $A \Leftrightarrow B$ then $K \div A = K \div B$	**C 5**
Recovery	$K \subseteq (K \div A) + A$	**C 6**

The postulate of recovery is not always achieved because it involves the composition of operators of contraction and expansion, but it leads to inconsistency in the AGM. If the contraction operator satisfies five postulates without recovery, it is named a withdrawal operator. This designation describes irreversible process and defines contraction as a singular operator, independent on the expansion.

4 Principles of Doxastic Modal Logic (DML)

DML language allows to place doxastic actions in the center of consideration. Doxastic formula DML formed by hanging doxastic operator on the Boolean formula. In fact, each doxastic formula DML is the representation of some action of belief revision at the time of conversion (see [10, p. 18]).

Let A be an usual formula of propositional calculus, + – doxastic operator expansion, ÷ – respectively doxastic operator contraction. Consider a fixed theory. Then $+A$ will mean expansion by A. Similarly $\div A$ will mean contraction by A. In DML interpretation doxastic operators can be applied to nondoxastic (Boolean) propositiond. The result of applying these operators will be the doxastic formula. Thus, expansion and contraction are interpreted in DML as modal (single) operators in pure form. And therefore, $+A, \div A$ and similar doxastic formulas do not express a certain state theory but the transformation theory. As a result of doxastic actions a static formula can be obtained, which expresses a certain state theory, or we will need to implement the next steps of doxastic actions.

Doxastic actions of expansion and contraction over a fixed belief set will look this way.

Expansion by A: $+A$

Contraction by A: $\div A$

The general characteristics of dynamic modal logic DML will be common to all axiomatic systems built according to this approach. Basic properties of the axiomatic systems will be presented according to the work of Y. Shramko [13].

As in the language of classical logic the illogical character of a set is an infinite list of propositional variables p, q, r, s. The logical symbols are signs of the truth-functional propositional connectives: $\&, \vee, \supset, \neg$ and the symbols of doxastic operators $+$ and $\div$. The technical characters are left and right parentheses $($, $)$.

DEFINITION 1 Boolean formula
A formula is called boolean if and only if it consists of elementary propositions and propositional connectives.

In building of this logic, we assume that doxastic operators are not applicable to doxastic formulas. Iteration of doxastic operators is possible, but with additional conditions and needs a deeper study. Thus, we have the following definition of an elementary doxastic formula.

DEFINITION 2 Elementary doxastic formula
$+A$ and $\div A$ are elementary doxastic formula iff A is a boolean formula.

DEFINITION 3 Well-formed formula
Well-formed formula (WFF) is called:
(1) any Boolean formula
(2) any elementary doxastic formula
(3) if A and B are WFF, then $A\&B$, $A \vee B$, $A \to B$,$\neg A$ are WFF,
nothing else is WFF.

All doxastic modal logic which have DML syntax will be formulated as a system of axioms with the rules of substitution and inference. First of all, note that all axioms of propositional calculus are axioms of each logic DML. We have the following rule:

PC All axioms of propositional logic are an axiom of DML.

Typically inference rule is Modus Ponens. If the formula A is inferred and formula $A \to B$ is inferred, then formula B is inferred.

MP If $\vdash A$ and $\vdash A \to B$ then $\vdash B$.

To formulate the rules of substitution we will need the notion of correct substitution.

DEFINITION 4 Correct substitution Substituting of formula B instead of occurrence of some variable p in well-formed formula A is called correct if and only if the result of substitution is a WFF.

Now we can formulate a rule of admissible substitutions.

US Let A be a theorem DML, and $p_1, p_2, \ldots, p_n$ are some propositional variables that are included in A. Then formula A, obtained by simultaneous substitution of some WFF $B_1, B_2, \ldots, B_n$ instead of every occurrence of $p_1, p_2, \ldots, p_n$, is a theorem.

Factually, the dynamic modal logic DML is a propositional calculus with the alphabet extended by typing doxastic modal operators:$+$, $\div$. These modal operators are a formal interpretation of cognitive actions: expansion and contraction, respectively.

5 Minimal logic of expansion

Consider the axiomatic construction of doxastic modal logic of expansion based on the interpretation of a cognitive action of expansion as a modal operator [10]. For this reason we need a DML language and series of axioms that define the specific nature of the operator of expansion. Consider the basic properties of the expansion used in the construction of epistemical theories. AGM define it by using the apparatus of set theory [8, p. 4]:

$K + A = Cn(K \cup A)$, where K is the initial belief set and Cn is the operation of closure under the logical consequence.

Thus, if we want to expand the theory by A, we must mechanically add A to the original belief set using the set-theoretic union operation and close the obtained set under the logical consequence.

By completeness of the belief set, each deducible formula of propositional calculus will be added to the resulting theory. Obviously the next property, if A is a theorem of propositional calculus, we must add it to our theory, which is similar to the well-known in modal logic rule of hanging of necessity. In the DML language this rule will take the following form: if A is a theorem of propositional calculus, then $+A$ is a theorem of minimal logic of extension.

N If $\vdash A$, then $\vdash +A$.

In this situation doxastic operator of expansion behaves like a normal modal operator. Remarkable, such behavior should not follow the properties of the operator, but from the definition of cognitive operator, which is represented by this modal operator. It indicates that the chosen method of presentation can actually be used.

The following important property is that the operator extension is closed under Modus Ponens. This characteristic, as the previous one, also follows the definition of expansion adopted in the AGM. By analogy with the principle of bringing of modality for normal modal systems, this principle can be expressed as distributivity operator of expansion with respect to the implication.

K $+(A \to B) \to (+A \to +B)$

Based on the previous note, the operator of expansion can be interpreted as a normal modal operator, which owns the properties of positive modality. Rules as PC, MP, US, N and K are the base of the most weak logic of extension of beliefs.

6 Dynamic modal logic of extension $\boldsymbol{DML_E}$

The expressive power of DML language is sufficient to build a stronger logic of expansion, which will examine in details the properties of AGM expansion. In building this logic we need a criteria of rationality AGM, taken in the concept:

- Requirement of minimal changes in initial beliefs,
- Priority of new information,
- Consistency.

To the list of criteria of rationality also can be added the principle of categorical matching, which is usually introduced implicitly, but is one of the most important when performing any operation. According to the principle of categorical matching (see [8, p. 5]) the result of the operation must be represented in the same form as the original data. Thus, the execution of any doxastic operation on the belief set should guarantee the result as the belief set and nothing else.

The criterion of priority of the new information requires the presence of expression in the belief set after expansion of the theory by this statement. In the AGM postulates for expansion is mentioned a criterion expressed as a postulate of success. Taking a statement to the belief set, the agent elevates it to the rank of belief and commits to adopt it.

T $+A \to A$

If we consider konsequent of T as an expression of truth, that "if A is added to the theory, then A is true", we get a very strong statement. Given the properties of the agent of belief (at least, he may be wrong), we can say that this statement is quite often incorrect. However, if we represent the right side of the formula as an approval A (according to Frege), not requiring a mandatory truth — no factual or logical [13]. Then the formula can be considered as an expression of the sequence of doxastic action which forces an agent to accept all of his beliefs. Factually, if the agent adds a statement to his belief set, then he is obliged to assert it. This expression describes agent's doxastic commitments, that he imposes on himself by adding a statement to the belief set.

Logic defines a set of rules PC, MP, US, N, K and T is *dynamic modal logic of expansion* DML_E.

7 AGM postulates of expansion within DML_E

Expressive capabilities of DML_E allow to formalize the postulates of expansion AGM: closure, success, inclusion, vacuity, monotonicity and minimality. According to the theorem AGM [1, p. 513], the operator which satisfies the given postulates is equivalent to the operator of expansion AGM.

7.1 Closure

The resulting belief set obtained from the expansion of the initial theory by some proposition A, should be closed under the logical consequence. Formally, if K is a belief set, then K + A is a belief set. This postulate expresses the principle of categorical matching whereby the representation of a belief after performing some operations must be the same as the way of presenting the initial state. In DML_E the postulate of closure is expressed in rule K.

K $+(A \to B) \to (+A \to +B)$

7.2 Success

After the expansion of the theory by some proposition A, it must belong to the theory. This postulate expresses the criterion of priority a new information, whereby the input information must be accepted. Formally: $A \in K + A$. In DML_E the success postulate is expressed by axiom T.

T $\quad +A \rightarrow A$

7.3 Inclusion

The initial belief set must be always a subset of the expanded belief set. This is a kind of assertion of ,,purity" of the operator of expansion. According to the postulate of inclusion, making the expansion, we combine the initial belief set and the set, which consists of the added proposition, while no proposition is removed from the initial set. Closure of resulting belief set must include the original belief set. Formally: $K \subset K + A$. In DML_E postulate of inclusion can be expressed the following:

$B\& + A \rightarrow B$

The postulate of inclusion can be obtained from the theorem of the propositional calculus by the substitution rules, so it is deducable in DML_E.

7.4 Vacuity

Expansion of the theory by proposition A which is already presented in it, does not change the theory. Formally: if $A \in K$, then $K + A = K$. In DML_E the postulate of vacuity can be expressed as follows:

$A \rightarrow (+A \rightarrow A)$

The above mentioned postulate AGM expansion may be withdrawn within DML_E using an axiom of propositional calculus and the rules of substitution, so it is a theorem of logic of expansion.

7.5 Monotonicity

Operation of expansion is monotonous under the set-theoretic operation of inclusion. If $K \subseteq N$, then $K + A \subseteq N + A$. The same property retains the operator of expansion with respect to implication in DML_E.

$(B \rightarrow C) \rightarrow ((B\& + A) \rightarrow (C\& + A))$

The postulate of monotonicity of expansion is also inferred in DML_E by the rules of substitution.

7.6 Minimality

For any belief set K and proposition A, $K + A$ is the lowest set of beliefs which satisfies the postulates of closure, success and inclusion. This postulate expresses the criterion of minimality, whereby the modified belief set must be strictly regulated. Expanding the theory by some proposition A we should add only this proposition A and nothing more to the belief set. Closure of the received belief set provides a presence added proposition in the belief set. Thus, if a proposition B does not follow from A, the expanding of the theory by A should not lead expansion of a theory by B.

Theorem. Minimality DML$_E$. $\neg(A \to B) \to \neg(+A \to +B)$

Proof.
1 $\neg(A \to B) \to \neg + (A \to B)$ US, T, contraposition
2 $\neg + (A \to B) \to \neg(+A \to +B)$ 1,K
3 $\neg(A \to B) \to \neg(+A \to +B)$ 1, 2, transitivity

If the implication $(A \to B)$ exists in the initial belief set, then when adding A, proposition B appears in the theory as a result of keeping closure. If B does not follow from A, but was still present in the initial theory, its adding does not change the theory, according to the postulate of vacuity.

Thus, the formalization of operation of expansion allows to express the definition of expansion, which moves in the AGM, the basic properties of this operation and criteria of rationality applicable to the expansion. The postulates AGM, which define the basic properties of operation of expansion, can be formalized and proved in dynamic doxastic logic of expansion. Thus, the logic given by the PC, US, N, K and T is a dynamic modal logic of extension DML_E and represents the function of expansion AGM.

8 Dynamic modal logic of contraction DML_C

Let us consider the operator of contraction. Any logical system based only on the operator of contraction, without expansion, allows us to express the properties of the withdrawal function [11], a specific type of contraction, which does not satisfy the postulate of recovery. Notably, while all the other AGM postulates of contraction are satisfied. It seems inappropriate to build a logical system based on an operator, which does not satisfy all AGM postulates of contraction, however, to express the complex operator, can be based even on withdrawal operator.

To build a logical system based on the operator of contraction, use the alphabet of dynamic modal logic DML with the rules of PC, MP, US. In addition, we introduce some specific rules in a logical system, that determine the properties of contraction, defined by the axiomatic definition of the AGM contraction functions.

First of all, consider the introduction of the operator of contraction. The next rule says that when we contract the theory by proposition B, we should remove all propositions A, which inferred B. Factually, if $A \to B$ is a theorem, then $\div B \to \div A$ is a theorem.

C If $\vdash A \to B$ and $A \to B$ is a boolean formula, then $\vdash \div B \to \div A$.

Note, the same principle, expressed by a formula $(A \to B) \to (\div B \to \div A)$ can not be accepted as an axiom of DML_C, because if $(A \to B)$ is not a tautology, an alternative for removal A (due to contraction by B) could be generally removal formula $(A \to B)$.

The operation of contraction should provide consistency of belief set. If the belief set contains some proposition A, it should not contain a denial of this proposition $\neg A$. Contraposition of these statements is more convenient: if a belief set contains $\neg A$, then in order to preserve consistency of the set, we need to delete A.

This doxastic commitment can be justified in another way, if we believe in a false proposition A (this is possible if A is contrary to our initial belief set and initial set contains $\neg A$), we must reject the false proposition. Indeed, we are trying to get rid of false propositions in the change of belief, no matter for what reason — as a result of obtaining some new information, or, for example, due to review the existing belief set in search of contradictions.

This principle of consistency of a belief set reflects the mechanism of contraction, and therefore should be accepted as an axiom DML_C and can be stated as follows:

U $\neg A \to \div A$

The following axiom expresses an important principle of partial meet contraction AGM, also called ”conjunctive factorization”.

DM1 $\div(A \& B) \to (\div A \vee \div B)$

It is natural to adopt this axiom. If we wish to contract the belief set by a conjunction and there exists some preference between the conjuncts, then this contraction is equivalent to contraction by the non-preferred conjuncts. If the both conjuncts are equal, then we will have to remove both [5, p. 12]. There are three ways to contract initial belief set by conjunction $A\&B$, such us: remove the sentence A but leave the sentence B, remove the sentence B but leave the sentence A, and remove both.

9 Doxastic modal logic of contraction DML_C and AGM postulates

The most convenient kind of contraction is represented in AGM by so-calling ,,partial meet contraction”. The basic postulates of contraction are developed in AGM (six basic and two additional postulates)[1]. All the postulates of contraction can be expressed in the language of DML_C, moreover almost all postulates of contraction can be proved in the doxastic modal logic of contraction.

9.1 Closure

According to the principle of categorial matching, the representation of a belief state after a belief change should have the same form as the representation of the belief state before the change [8, p. 5]. Hence, if the initial theory K is logically closed, then theory $K \div A$ is closed too. Let us contracting the set K by proposition A. Thus, before contraction we need to make sure that the original set K is closed under logical consequense, and after a contraction we must to implement the closure outcome set $K \div A$.

On the expansion postulate of closure acquires the form

$$+(A \to B) \to (+A \to +B)$$

On the contraction this postulate points out that if the contraction the set K by proposition A was done correctly, then closure of the outcome set $K \div A$ must not contain A. The proposal should be deleted explicitly and implicitly. To do this we need to remove all statements that lead A otherwise the closure operation again returns it to the theory. In the language of DML_C it will look as follows.

Theorem 1. ***DML1*** $\quad \vdash (A \to B) \Rightarrow \vdash (\div B \to \div A)$

Proof. If B follows from A, then, if we remove B, then we must remove A. This is a inference rule **C**.

9.2 Inclusion

$K \div A \subseteq K$

A theory obtained by contraction of some theory must be a subset of the initial theory. It is impossible that the outcome belief set $K \div A$ was intersected with the initial belief set or K was a proper subset of $K \div A$.

Theorem 2. ***DML2*** $\quad \div B \to (\div A \to \div B)$

Proof. This principle is a special case of the consequent approval, and therefore, it is the theorem of our system.

A deletion any proposition should not cause an expansion of a system by means of proposition which are not exist in it. Factually, any proposition which does not belong to belief set should not appear in the theory as a result of contraction by anyone proposition.

9.3 Vacuity

$A \notin K \Rightarrow K \div A = K$

If the proposotion does not belong to the original belief set, then its removal should not cause any transformation of the system. In doxastic modal logic of contraction this postulate can be expressed by the following method:

Theorem 3. ***DML3*** $(B\&\neg A) \rightarrow (\div A \rightarrow B)$

In other words, if belief set is compatible with $\neg A$, that it does not contain statement A, then we obtain the same initial set B, when try to remove the A.

Proof.

1 $(B\&\neg A) \rightarrow B$	PC
2 $B \rightarrow (B \vee \neg \div A)$	1, PC
3 $(B \vee \neg \div A) \rightarrow (\div A \rightarrow B)$	2, PC
4 $(B\&\neg A) \rightarrow (\div A \vee B)$	1, 3, transitivity

9.4 Success

$A \notin Cn(\varnothing) \Rightarrow A \notin (K \div A)$

Reformulating this postulate by contraposition, we obtain the next note: if $A \in (K \div A)$, then $A \in Cn(\varnothing)$. That is, if a statement is present in the theory after it was removed, then the statement is a logical tautology. In the language of DML_C this postulate can be written as

Theorem 4. ***DML4*** $(\div A \rightarrow A) \rightarrow ((A \rightarrow A) \rightarrow A)$

Proof.

1 $\neg A \rightarrow \div A$	U
2 $(\neg A \rightarrow \div A) \rightarrow (\div A \vee A)$	1, PC
3 $(\neg A \vee \div A) \rightarrow ((\neg A \vee \div A)\&(A \vee \neg A))$	2, PC
4 $((\neg A \vee \div A)\&(A \vee \neg A)) \rightarrow (\div A\&\neg A) \vee (A\&\neg A) \vee A$	3, PC
5 $((\div A\&\neg A) \vee (A\&\neg A) \vee A) \rightarrow ((\neg \div A \vee A) \rightarrow ((\neg A \vee A) \rightarrow A))$	4, PC
6 $((\neg \div A \vee A) \rightarrow ((\neg A \vee A) \rightarrow A)) \rightarrow ((A \rightarrow A) \rightarrow A)$	5, PC

9.5 Extensionality

If $A \leftrightarrow B \in Cn(\varnothing)$, then $K \div A = K \div B$

Extensionality guarantees that the logic of contraction is extensionak in the sense of allowing logically equivalent sentences to be freely substituted for each other [8, p. 8].

Theorem 5. ***DML5*** $\vdash A \leftrightarrow B \Rightarrow \vdash \div A \leftrightarrow \div B$

Proof. The proof of this rule can be easily obtained using the rule **C**.

Thus, we expressed the five postulates of AGM contraction and built their proof in the system DML_C. Within the logic contraction can not formalize the postulate of recovery, because it requires a explicit use of the operator of expansion.

So, for the modal operator of the contraction imposed by axioms DML_C, performed five basic postulates of contraction AGM: closure, inclusion, vacuity, success, and extensionality. Thus, the modal operator of contraction reflects the properties of the functions of contraction AGM. Now, we can formulate appropriately *representational theorem*, analogous to [1].

Representational Theorem. *Sinse the modal operator of the contraction DML_C satisfies the properties closure, inclusion, vacuity, success, and extensionality, hence the operator of the contraction DML_C is equivalent to withdrawal function AGM.*

Thus, the representation doxastic actions as modal operators allows to construct axiomatical systems for the operator of the expansion and the operator of the contraction. Within the system based on this interpretation, we can express and prove the corresponding postulates of expansion and contraction AGM. It demonstrates that these modal operators correspond to the functions of expansion and contraction described in AGM, about that the representational theorem has been formulated. This method of representation of doxastic actions makes it possible to formulate rigorously known properties of expansion and contraction and, moreover, to trace the new properties of doxastic actions, which manifest themselves by virtue of this representation.

References

1. Alchourron, C.E., Gaȑdenfors, P., Makinson, D.: On the logic of theory change: Partial meet contraction and revision functions. Journal of Symbolic Logic **50**(2), 510–530 (1985). doi:10.2307/2274239
2. Baltag, A., Fiutek, V., Smets, S.: Beliefs and evidence in justification models. In: Beklemishev L., Demri S., Máté A. (eds.) Advances in Modal Logic, vol. 11, pp. 156–176 (2016)
3. Fermé, E.; Garapa, M., Reis, M.D.L.: On ensconcement and contraction. Journal of Logic and Computation **27**(7), 2011–2042 (2017). doi:10.1093/logcom/exx008
4. Fermé, E., Hansson, S.O.: AGM 25 Years: Twenty-Five Years of Research in Belief Change. Journal of Philosophical Logic **40**(2), 295–331 (2011). doi:10.1007/s10992-011-9171-9
5. Fermé, E., Hansson, S.O.: Belief Change: Introduction and Overview. Springer Briefs in Computer Science Series. Springer, Cham (2018). doi:10.1007/978-3-319-60535-7
6. Gardenfors, P.: Knowledge in Flux: Modeling the Dynamics of Epistemic States. College Publications, London (2008)
7. Girard, P., Rott, H.: Belief revision and dynamic logic. In: Baltag A., Smets S. (eds.) Johan van Benthem on Logic and Information Dynamics. Outstanding Contributions to Logic, vol. 5, pp. 203–233. Springer, Cham (2014). doi:10.1007/978-3-319-06025-5_8
8. Hansson, S.O.: A Textbook of Belief Dynamics: Theory Change and Database Updating. Springer Netherlands, Dordrecht (1999)
9. Hansson, S.O.: Ten Philosophical Problems in Belief Revision. Journal of Logic and Computation **13**(1), 37–49 (2003). doi:10.1093/logcom/13.1.37
10. Kozachenko, N.P.: Doxastic Modal Logic and AGM. In: Markin V.I. (ed.) Smirnov's Readings; 6nd International Conference, Moscow, 17–19 June 2009, pp. 18–19. Sovremennye tetradi, Moscow (2009)
11. Makinson, D.: On the status of the postulate of recovery in the logic of theory change. Journal of Philosophical Logic **16**(4), 383–394 (1987). doi:10.1007/BF00431184

12. Segerberg, K.: Belief Revision from the Point of View of Doxastic Logic. Logic Journal of the IGPL **3**(4), 535–553 (1995). doi:10.1093/jigpal/3.4.535
13. Shramko, Y.: Doxastic actions and doxastic commitments: belief revision as pure modal logic. In: Smirnov's Readings; 2nd International Conference, pp. 90–92. Moscow (1999)

The Dawn of Software Engineering Education

Andrii M. Striuk[1][0000-0001-9240-1976] and Serhiy O. Semerikov[1,2,3][0000-0003-0789-0272]

[1] Kryvyi Rih National University, 11, Vitalii Matusevych Str., Kryvyi Rih, 50027, Ukraine
[2] Kryvyi Rih State Pedagogical University, 54, Gagarina Ave., Kryvyi Rih, 50086, Ukraine
[3] Institute of Information Technologies and Learning Tools of NAES of Ukraine,
9, M. Berlynskoho Str., Kyiv, 04060, Ukraine
andrey.n.stryuk@gmail.com, semerikov@gmail.com

Abstract. Designing a mobile-oriented environment for professional and practical training requires determining the stable (fundamental) and mobile (technological) components of its content and determining the appropriate model for specialist training. In order to determine the ratio of fundamental and technological in the content of software engineers' training, a retrospective analysis of the first model of training software engineers developed in the early 1970s was carried out and its compliance with the current state of software engineering development as a field of knowledge and a new the standard of higher education in Ukraine, specialty 121 "Software Engineering". It is determined that the consistency and scalability inherent in the historically first training program are largely consistent with the ideas of evolutionary software design. An analysis of its content also provided an opportunity to identify the links between the training for software engineers and training for computer science, computer engineering, cybersecurity, information systems and technologies. It has been established that the fundamental core of software engineers' training should ensure that students achieve such leading learning outcomes: to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of language, instrumental and computational tools for software engineering; know and apply the appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development; put into practice the software tools for domain analysis, design, testing, visualization, measurement and documentation of software. It is shown that the formation of the relevant competencies of future software engineers must be carried out in the training of all disciplines of professional and practical training.

Keywords: software engineering, professional training, software, specialist training model, standard of higher education.

1 Introduction

On October 29, 2018, by Order of Ministry of Education and Science of Ukraine № 1166, the higher education standard for specialty 121 "Software Engineering" for the first (bachelor's) level of higher education was approved and implemented

(Standard 2018) [88]. The Standard of higher education contains competencies that determine the bachelor's training specifics in specialty 121 "Software Engineering" and program results of training. The introduction of the standard was preceded by a long work of domestic software engineering (SE) specialist community, the result of which was the formation of future SE specialist' general and special competency-based system.

The rapid development of models, methods and tools of SE and the IT industry as a whole raises the question of whether the proposed competency-based system is sufficient to form a modern SE specialist capable of professional development and self-improvement throughout life. A positive answer to this question will contribute to improving Ukraine's position in the IT international labor market, and a negative one will preserve the current state of technology development and the programmed lag of Ukraine's IT education from other states.

To answer this question, it is necessary to determine the ratio of the stable (fundamental) and mobile (technological) component in the content of vocational training of specialists in SE. As one of the founders of SE Brian Randell (born 1936) pointed out that "Those who cannot remember the past are condemned to repeat it. So I hope that you will ... to pay a little more attention to the past, ... to take some time to read or re-read, for example, the original 1968 NATO Report, ... before you resume work on inventing yet another new language or [programming] technique" [7, p. 8]. That is why an analysis of the main stages of development of software engineering as a branch of knowledge was presented, the fundamental components of the training of future software engineers were identified and development trends for this industry for the next decade were determined in our previous work [9].

The purpose of the article is to analyze the first software engineering specialists training model developed under the guidance of Friedrich Ludwig Bauer, and to establish its compliance with the current state of SE development and the new higher education standard in specialty 121 "Software Engineering".

2 The first software engineering training model (1972)

According to the results of the NATO conference on software engineering 1968 [6] and 1969 [2] an international experts group led by Friedrich Ludwig "Fritz" Bauer (1924-2015) developed "Advanced Course on Software Engineering" [1] in late 1971 – early 1972. The Advanced Course took place February 21 – March 3, 1972, organized by the Mathematical Institute of the Technical University of Munich and the Leibniz Computing Center of the Bavarian Academy of Sciences. In the Preface to the materials of this training program for SE Bauer indicates that the book [1] is the first step towards the concentration and systematization of relevant teaching materials: "Our intention in the planning of this course was to cover as much as we can at the moment of all the aspects of the theme [SE], and to contribute further to the systematization of the field. ... we think it is essential to point out where the ideas of software engineering should influence Computer Science and should penetrate in its curricula ... to your students in an academic environment" [1, p. 2].

The developed training program for SE consisted of four sections.

The first section – "Introduction" – contained two lectures. In the first lecture by Keith William "Bill" Morton (born 1930), using concrete examples, describes the problems that led to the software crisis and identifies 8 main directions for overcoming them when developing application software packages that correspond to learning outcomes of the 2018 standard (Table 1).

Table 1. Directions for overcoming a software crisis

Main directions for overcoming the software crisis according to Morton	Programmatic results for SE professionals according to Standard 2018
project management – training of staff in appropriate programming techniques; setting up standards; sub-dividing work into manageable parts; monitoring progress and quality	PR22 – to know and be able to apply project management methods and tools
product definition – specifying its function; defining user image; effects of host operating system	PR09 – to know and be able to use methods and means of collecting, formulating and analyzing software and PR11 – to select the initial data for design, guided by formal methods for describing requirements and modeling
design and implementation	PR10 – to conduct a pre-project survey of the subject area, a system analysis of the design object, PR12 – to put into practice effective approaches to software design, PR14 – to put into practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software and others
application of different levels problem-oriented languages or abstract machines hierarchies, most suitable for solving applied problems	PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of language, instrumental and computational tools for software engineering and PR15 – motivated to choose programming languages and development technologies for solving problems of creating and maintaining software
documentation – selecting levels, methods and automatic aids; controlling quality; disseminating and updating	PR05 – to know and apply the appropriate mathematical concepts, domain, system and object-oriented analysis and mathematical modeling methods for software development, PR14 – to put into practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software and PR16 – to have the skills of team development, coordination, design and release of all program documentation types
testing – generation of test data; use of test beds	
performance measurement – simulation; measurement tools; monitoring and optimisation	
maintenance and enhancement	

At the end of the lecture, Morton answers the question posed in title – "What the Software Engineer Can Do for the Computer User": improvements to the computer systems that computer user has to use; and tools and techniques that software engineer can make use of in his own work [1, pp. 4–11].

Jack Bonnell Dennis’ (born 1931) the second lecture of the first section “The Design and Construction of Software Systems” examined such basic concepts as “computer systems”, “software systems”, “their hierarchy”, “system software”, and “application software”, methods for describing software systems, their function, correctness, performance and reliability, as well as software projects.

At the beginning of the lecture, the author sets out his vision of SE as the application of principles, skills and art to the design and construction of programs and systems of programs, focusing specifically on the sound principles of SE and the extent of their applicability.

Dennis defined a computer system as a combination of hardware and software components that provides a definite form of service to a group of “users” [1, p. 13]. Therefore, for programming subsystems using the example of the Basic language, the author identifies at least three distinct computer systems and corresponding user groups (consumers of certain services):

1. the computer hardware, whose “users” are operating system implementers;
2. hardware plus operating system, whose “users” are subsystem implementers;
3. hardware, operating system and Basic language subsystem, which services are provided to users of this language.

Based on this unity, the author defined software systems as the software and hardware components that must be added to a specific computer system, called the host system, in order to realize some desired function. Thus, the hardware and the operating system form the base computer system for a software programming system in the Basic language.

The systems hierarchy is formed by expanding them or defining a new way of interacting (linguistic level) with them through translation or interpretation of commands.

System software – a collection of system programs usually forms a hierarchy of software systems having these properties:

1. The collection of programs is implemented under one authority.
2. The hierarchy of software systems defines a single linguistic level which applies to all users of the collection of programs.
3. Inner linguistic levels of the hierarchy are hidden from the user.
4. The outer linguistic level of the hierarchy is “complete” for the goals of the implementing authority.
5. The primary means of defining new linguistic levels is partial interpretation.

Application software – an application program or software system usually has these properties:

1. The programs are expressed in terms of a “complete” linguistic level.
2. The programs define a new linguistic level by extension, translation, interpretation, or by some combination of these techniques.
3. The linguistic level defined by the program or software system is inadequate for defining further linguistic levels.

4. A variety of such programs or software systems are available to clients of an installation, and are often implemented under different authorities.

J. B. Dennis determines the description of the software system through the description of its software component (expressed in a well-defined programming language), hardware component (using a model that reflects the behavior of the hardware component in all normal operating conditions of the software system), the base system and the linguistic level at which a software system should be implemented.

The goals of a software system designing are expressed in its desired properties: functionality (as the correspondence desired of output with input), correctness (using a structural programming style that makes program correctness self-evident, or using proof of correctness of software systems or components), performance (the effectiveness with which resources of the host system are utilized toward meeting the objective of the software system) and reliability (the ability of a software system to perform its function correctly in spite of failures of computer system components).

Dennis' lecture concludes with a description of the typical tasks that arise when developing large software projects.

2018 Standard defines the following programmatic outcomes relevant to this lecture:

PR03 – to know the main processes, phases and iterations of the software life cycle;

PR05 – to know and apply the appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR06 – the ability to choose and use the appropriate methodology for creating software;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR10 – to conduct a pre-design survey of the subject area, a system analysis of the design object;

PR11 – to choose the source data for design, guided by formal methods for describing requirements and modeling;

PR12 – to put into practice efficient approaches to software design;

PR13 – to know and apply methods for developing algorithms, designing software for data structures and knowledge;

PR14 – to apply in practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software;

PR15 – motivated to choose programming languages and development technologies for solving problems of creating and maintaining software;

PR22 – to know and be able to apply methods and means of project management.

The second section – "Descriptional Tools" – contained five lectures. In the first lecture "Hierarchies" Gerhard Goos (born 1937) discusses the methodological foundations of the decomposition of software systems and their application to the development of software and programming languages.

Goos suggests starting the software systems designing with a description of the problem to be solved and the available host system (as interpreted by J. B. Dennis). The task can be formally represented by an abstract machine, which from the input data using the program for this machine produces some source data that solve a certain part of the problem [1, p. 29]. At the first stage (general system design), a set of software components is determined by specifying their external interfaces, each of which solves a part of the original problem, realizing it using some functions of an abstract machine. The internal behavior of each component is determined at the second stage (detailed system design).

The results of general design can be represented as a network of components – a directed graph of arbitrary complexity. The presence of fatal cycles in this reflects the complexity of the design process. Goos calls the principle of structuring systems in the form of a partial ordered set of layers (including a tree-like or linear structure) hierarchical beautification and indicates that its application allows you to divide the system into components in such a way that a picture of their interconnections is obtained and the contribution of each component to solving the general problem of the system is clearly determined. Each layer can be implemented by a specific abstract machine, the sequence of which forms a system, and each next (lower) layer provides all the necessary tools for the implementation of the previous (higher) layer, and the last layer implements an abstract machine, which is identical to the base system. Programming language is defined for each level of abstraction that provides access to all the functions of an abstract machine of a given level.

Goos considers hierarchical design as a means of engineering software development and production. The main assumption of the author is that operations and data structures which are present at the A_{i+1} level, but are absent at the A_i level, cannot be used by programs running at the A_i level. This assumption provides the possibility of using hierarchical ordering as well as testing and debugging tools [1, p. 44-45].

Corresponding lecture program results of training according to 2018 Standard:

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR17 – to be able to apply the methods of component software development.

From an engineering point of view Goos examined how programming languages affect the program development process and its properties in the second lecture “Language Characteristics: Programming Languages as a Tool in Writing System Software”.

Just as natural language affects the thinking of the person who uses it, programming language influences its user at least with respect to the following:

- The conceptual understanding how a problem can be solved by computer
- The range of problems which can be attacked by programming
- The set of basic notions available for programming
- The style of programming (clarity, robustness, readability etc.)
- The meaning of “portability”
- The meaning of “efficiency”

The lecture established a correspondence between the properties of the languages and the properties of the programs described by them, and defined some characteristics of “good” programming languages, the main of which is to encourage the programmer to write correctly (from an engineering point of view) the designed programs. The author paid special attention to the characteristics of high-level system programming languages. Goos primarily refers the means of improving the quality of system software engineering (creating “good” programs) to structural programming tools (in particular, modularity), hierarchical ordering, the use of nesting and visibility, as well as parallel execution.

Corresponding lecture program results of training according to 2018 Standard:

PR06 – the ability to choose and use the appropriate methodology for creating software;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR15 – motivated to choose programming languages and development technologies for solving problems of creating and maintaining software;

The third lecture of the second section “Low Level Languages: Summary of a Discussion Session” is a synthesis of round table materials prepared by M. Griffiths (France), in which F. L. Bauer, G. Goos, M. Griffiths, and other SE experts participated. Answering the question of why creating a new low-level programming language, the panelists indicate that existing machine languages (assemblers) do not provide support for efficient memory allocation and compliance with the programming style, which aims to increase the reliability of software. The discussion participants set the following requirements for new low-level programming languages: the availability of flow control tools (cyclic and conditional statements), support for modularity (at least at the procedure level with parameter passing), data structures (with the possibility of type conversion, indexing, address arithmetic). Moreover, such languages can be either machine-oriented or problem-oriented (such as Forth).

It was proposed to measure the effectiveness of low-level programming languages by three indicators: the complexity of the programmer’s labor, computer time, and memory capacity. The panelists noted regarding the programming style the vital role of training future system programmers using appropriate tools and a good programming style. At the time of the course, the discussion participants considered a successful example of a low-level language that meets these requirements, namely a language that was used to develop the MULTICS – PL / I operating system, more precisely, its early EPL dialect, which significantly influenced the design of the programming language, and today it is a reference implementation of the concept developed during the discussion – C.

Corresponding lecture program results of training according to 2018 Standard:

PR06 – the ability to choose and use the appropriate methodology for creating software;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR15 – motivated to choose programming languages and development technologies for solving problems of creating and maintaining software;

The fourth lecture "Relationship Between Definition and Implementation of a Language" is related to the main area of M. Griffiths research – the development of compilers: "The non-specialist sometimes accuses both computer scientists and software engineers of spending all their time on discussions about languages, to the detriment of all the 'real' problems. Whilst this accusation is net without foundation, it must be clearly understood that language is central to the whole problem of software engineering. If we cannot supply powerful, well-defined, understandable languages with corresponding, economic implementations on existing computing equipment, then the programmer can hardly be expected to express himself in a way which will permit us to use the term which is the theme of this school." [1, p. 77]. The lecture discussed the impact of language definition on the programs written in the language and on the methods used to execute these programs in the computer.

The author identifies three points of view on the definition of language: 1) the user point of view, which focuses on the task, and not on the features of the language description (language description as a guide); 2) the compiler developer point of view, which seeks to implement all the language described features, even those that very little will be used (language description as scripture); 3) the researcher point of view, for whom the language description is a complete, formal, mathematically correct set of statements that reflect traditional mathematical principles of minimizing the interconnected axioms number.

According to the author, the main thing in determining the language is the absence of side effects. He distinguishes two components of semantics for this purpose: static semantics is a part of a programming language semantics that is independent of program execution and determined at compile time (such as the relationship between the use of an identifier and its announcement); dynamic semantics takes into account real object manipulations and their values when program execution.

The leading is the grammatical approach in determining the language syntax, particularly using LL (1)-grammars to implement context-free languages. M. Griffiths gave a range of examples which demonstrates the influence of the method of determining the language on its implementation, showing that the ease, safety, and efficiency of the language implementation directly depends on its definition: "It is this idea which prompts us to suggest that the definition of a language should foresee its implementation, and that the implementation should be strongly directed from the very start. There is here a parallel with the architecture of computer hardware, which is likely improve conceptually each time the hardware engineer works close enough to the software engineer and hence considers the use to which the machine may be put. The use to which a language definition is put is in the first instance the implementation, We consider, therefore, that a language definition should be in terms of an idealised implementation, or at least be accompanied by an "implementers' guide" [1, pp. 99–100].

The following methods for determining the language are discussed in the lecture:

- two-level grammar (a formal grammar that is used to generate another formal grammar), such as the Van Wijngaarden grammar for Algol 68 in conjunction with a stylized natural language description of a program interpretation on a hypothetical computer;
- Vienna Definitions (Vienna Definition Language – VDL), today, it is better known as the Vienna Development Method (VDM) – a set of technologies for modeling systems, analyzing created models and moving to detailed design and programming), which used operational (connotative) semantics that is a way of describing a language using sequences of calculation steps, and it is very closely connected with the implementation of the system in a programming language, since the calculation steps are described in the language of some computer;
- expanding languages that consisting of a relatively simple but self-sufficient base language, which itself contains extension mechanisms that make it possible to define new operators or data types.

Corresponding lecture program results of training according to 2018 Standard:

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR12 – to put into practice efficient approaches to software design;

PR13 – to know and apply methods for developing algorithms, designing software for data structures and knowledge.

The last lecture in the second section “Concurrency in software systems”, written by J. B. Dennis, focuses on large modular programs such as operating systems, compilers, or real-time control programs, many interacting parts or modules. Due to the size of these, programs, it is essential that the parts be represented in such a way that the descriptions of the parts are independent of the pattern in which they are interconnected to form the whole system, and so the behavior of each part is unambiguous and correctly understood regardless of the situation in which it is used. For this to be possible, all interactions between system parts must be through explicit points of communication established by the designer of each part. If two parts of a system are independently designed, then the timing of events within one part can only be constrained with respect to events in the other part as a result of interaction between the two parts. So long as no interaction takes place, events in two parts of a system may proceed concurrently and with no definite time relationship among them. Imposing a time relation on independent actions of separate parts of a system is a common source of overspecification. The result is a system that is more difficult to comprehend, troublesome to alter, and incorporates unnecessary delays that may reduce performance. This reasoning shows that the notions of concurrency and asynchronous operation are fundamental aspects of software systems. According to Dennis, such considerations show that the concept of parallelism and asynchronous operation are fundamental aspects of software systems.

The lecture discusses system models presented as sets of simultaneously acting subsystems interacting with each other through certain communication mechanisms. The author shows that if the interaction between the subsystems is subject to certain environmental conditions, the determinism of the subsystems guarantees the determinism of the entire system (the ability to give the same results for different launches with the same input data). Such system description is performed using Petri nets: the lecture ends with an example of their application to systems with parallel processes that interact by means of semaphores, using the synchronization primitives P and V by Edsger Wybe Dijkstra.

Corresponding lecture program results of training according to 2018 Standard:

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering.

The third section "Techniques" contained four lectures. In the first lecture "Modularity" J. B. Dennis deepens the discussion of modularity concept that begun in the previous lecture, which is closely related to structural programming as a property of computer systems: "A computer system has modularity if the linguistic level defined by the computer system meets these conditions: Associated with the linguistic level is a class of objects that are the units of program representation. These objects are program modules. The linguistic level must provide a means of combining program modules into larger program modules without requiring changes to any of the component modules. Further, the meaning of a program module must be independent of the context in which it is used" [1, p. 130].

Dennis considers modularity both in the context of the previous lecture (competition with communication) and in the procedural context, when one module calls another with a certain input data set, eventually obtaining the result. The lecture considers issues arising in the construction of software modules that require the ability to create, expand and modify structured data. It is concluded that in order to achieve modularity, a computer system must determine a linguistic level that provides an appropriate basic representation for structured data: 1) any data structure may occur as a component of another data structure; 2) any data structure may be passed (by reference) to or from a program module as an actual parameter. 3) a program module may build data structures of arbitrary complexity. At this level, the memory should be addressed not by elements of a predetermined length, but by data structures. The author considers the main memory size of the computer system to be the main limitation on the module size, and virtualization is the way to overcome it.

Significant support for modularity can come from a correctly designed operating system – for example, the author gives unique characteristics of Multics that contribute to the implementation of the concept under discussion:

1. A large virtual address space (approximately 2^{30} elements) is provided for each user.

2. All user information is accessed through his virtual address space. No separate access mechanism is provided for particular sorts of data such as files.
3. Any procedure activation can acquire an amount of working space limited only by the number of free segments in the user's address space.
4. Any procedure may be shared by many processes without the need of making copies.
5. Every procedure written in standard Multics user languages (FORTRAN, PL/I and others) may be activated multiply through recursion or concurrency.
6. A common target representation is used by the compilers of two major source languages – PL/I and FORTRAN.

These achievements are Multics major contributions toward simplifying the design and implementation of large software systems. They were made possible by building the Multics software on a machine expressly organized for the realization of a large virtual memory and shared access to data and procedure segments [1, p. 161].

The last part of the lecture informally presents semantic concepts of the linguistic level (common basic language), which can serve for the general presentation of program modules expressed by various programming languages.

Corresponding lecture program results of training according to 2018 Standard:

PR06 – the ability to choose and use the appropriate methodology for creating software;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR15 – motivated to choose programming languages and development technologies for solving problems of creating and maintaining software;

PR17 – to be able to apply the methods of component software development.

The second lecture "Portability and Adaptabilty" by Peter Cyril Poole (1931 – 2017) and William M. Waite (circa 1936) begins with the definition of portability as measure of the ease with which a program can be transferred from one environment to another, and adaptability as measure of the ease with which a program can be altered to fit differing user images and system constraints. The major distinction between the two concepts is that adaptability is concerned with changes in the structure of the algorithm, whereas portability is concerned with changes in the environment.

The main way to increase program mobility is to use high-level languages, provided that:

- the basic operations and data types required by the problem are available in the chosen language;
- the chosen language has a standard definition, and this standard definition is widely implemented;
- care is taken to avoid constructions which are accepted in the local dialect, but prohibited by the standard [1, p. 187].

Another way is to use abstract machines on which the mobile program will run. In this case, compilation from a high-level language is first performed on the abstract machine language (such as UNCOL – Universal Computer Oriented Language), which must necessarily support at least integers and integer arithmetic, comparison and relational operations, symbolic input and output, as well as real numbers and real arithmetic, strings (connection, substring selection, lexicographic comparison), input and output of memory images, labels and means of control transfer, declarations, arrays and records, conditional and cyclic operators, procedures and blocks. At the end of the lecture, the authors built abstract machines hierarchy that provide mobility and adaptability and level out some of the disadvantages of UNCOL.

Corresponding lecture program results of training according to 2018 Standard:

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR13 – to know and apply methods for developing algorithms, designing software for data structures and knowledge;

PR15 – motivated to choose programming languages and development technologies for solving problems of creating and maintaining software.

At the beginning of the third lecture "Debugging and Testing", P. C. Poole points out that proof of program correctness by formal methods (such as VDM) is too long to find errors, so a good software engineer should plan the testing and debugging phase [1, p. 279] (moreover, testing in no case does not show the absence of errors – only their presence). The standards of software documentation which are discussed in a separate lecture, and the "readability" of program code is played an important role in this, in particular, the presence of meaningful comments is useful for debugging and indispensable for maintaining and improving software. The program code must be accompanied by benchmarks and data that must be changed with the code.

Poole introduced the concept of "guard code" is the use of logical expressions to limit computation options [1, p. 287], which he suggests actively using in conjunction with the "principle of exceptional suspicion", which consists in the fact that the module should not use any data transmitted to it through the interface without first checking its compliance with the interface specification.

The main methods for fixing at that time were analysis of posthumous dumps, debugging output, direct and reverse tracing of the program code part, compilation with the inclusion of data for debugging, code documentation (reformatting the text in order to highlight the structural elements of the code, creating tables containing all procedures, labels and announcements, creating an index of procedure calls, following labels and variable references, visualizing control flows, etc.). The author considers the interactive setup managed by the user as a promising tool (online debugging).

Corresponding lecture program results of training according to 2018 Standard:

PR14 – to apply in practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software;

PR19 – to know and be able to apply software verification and validation methods;

PR20 – to know the approaches for evaluating and ensuring the quality of software.

Dionysios C. Tsichritzis (born 1943) considers the design and construction of reliable software in the last lecture of the section “Reliability”. The author begins the review with the remarks of E. V. Dijkstra “Testing can show the presence of errors and not their absence” and B. Randell “Reliability is not an add-on feature”, emphasizing the main idea of SE is to provide means that will enable the individual medium ability to create quality software [1, pp. 319-320].

Tsichritzis postulates a number of provisions that affect the reliability of software development:

1. different programming languages prompt programmers to various types of errors, which he calls characteristical;
2. the compiler must identify semantic inconsistencies in the programs text;
3. programming style in particular:

   - using appropriate variable names and structuring program code;
   - the use of methods such as cross-checking, checking ranges of variable values, checking the consistency of data changes, observing the uniqueness of names, phased data processing, including error codes in the results;

4. the impact of protection (a managed software environment with clearly defined rules and restrictions) by diagnosing errors as a violation of protection and isolating errors in the modules in which they took place;
5. proof of the program correctness by informal (by introducing checks on the values of input, intermediate, and output variables) and by formal (for example, by calculating first-order predicates) methods;
6. program design aimed at improving reliability:

   - structuring a program to facilitate testing;
   - informal verification of the some program parts logical correctness;
   - application of synchronization and process interaction tools;
   - using a hierarchy of abstract machines;

7. ensuring reliability during program operation by:

   - data integrity using partial or full memory dumps;
   - duplication of key data in different repositories;
   - accounting for hardware failures;
   - availability of program resumption to minimize the failures impact.

Discussing ways to protect the system as a whole, processes and user data, Tsichritzis introduces the concepts of domain (active and potentially aggressive entity) and object (passive and potentially vulnerable entity), giving examples of their implementation through processes and files. At the virtual machine abstraction level, reliability violations are a violation by the user of the operation of their own virtual machine or virtual machines of other users. The author uses a matrix apparatus for allocating resources between parallel processes in conditions of a lack of resources to describe this process. The author offers randomly generated names of synchronization objects (magnum – magic numbers) among the methods used to this day. The lecture discusses in detail the data protection mechanisms in file systems that should be implemented in the kernel of the operating system.

The author interprets the concept of data security (information security, data security or cybersecurity) as controlling access to privileged data stored in large scalable data banks in three categories [1, p. 357]:

1. information privacy involves issues of law, ethics and judgment controlling the access of information by individuals;
2. information confidentiality involves rules of access to data;
3. information security involves the means and mechanisms to ensure that privacy decisions are enforceable.

Tsichritzis distinguishes between protection and data security in the following way: protection concerns only access control to data in the operating system without taking into account the nature of data, and for information security tasks the information system is considered as a whole, and not just a computer-based system. The active elements of the information system are people, and the nature and content of the data are taken into account when providing access to them. That is why the author considers examples of information security tasks not on file systems, but on database management systems.

The author considers the following important tasks of information security:

a. at the login level: user identification, authentication, monitoring of resource use;
b. at the file system level: determining available files, user identification, password-controlled file properties, cryptographic protection;
c. at the data protection level: data separation of various users, data integrity, backup, reliable deletion of protected data;
d. restrictions on use (in particular, forced logout due to non-standard actions).

Corresponding lecture program results of training according to 2018 Standard:

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR06 – the ability to choose and use the appropriate methodology for creating software;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR08 – to be able to develop a human-machine interface;

PR11 – to choose the source data for design, guided by formal methods for describing requirements and modeling;

PR14 – to apply in practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software;

PR18 – to know and be able to apply information technology for processing, storage and data transfer;

PR19 – to know and be able to apply software verification and validation methods;

PR20 – to know the approaches for evaluating and ensuring the quality of software;

PR21 – to know, analyze, choose, and skillfully apply information security (including cybersecurity) means and data integrity in accordance with the tasks being solved and the software systems being created.

The fourth section “Practical aspects” is the largest in the entire course and includes six lectures. D. Tsichritzis formulates the general goal of software project management in the first lecture of “Project management” that is “production of the desired product for specific design goals, specifications and available resources” [1, p. 375]. To achieve this goal the author suggests setting up developer communications (through the use of common documentation, frequent informal communication and regular meetings of small development groups), organizing developers (by identifying a person who has a systematic view of the project, attracting developers to other components of the project and reducing management formalism) and set control points.

Among the software project development tools Tsichritzis pays special attention to information system modeling tools (for example, Petri nets are such a tool for describing parallel processes) and computer-aided design software systems, some of which (such as Project LOGOS) are significantly ahead of their time. The main difference between small and large software projects, in his opinion, is the number of management levels: there should be at least two of them. The issues of reducing the time for implementing software projects and requirements for managers of large projects are discussed at the end of the lecture.

Corresponding lecture program results of training according to 2018 Standard:

PR03 – to know the main processes, phases and iterations of the software life cycle;

PR04 – to know and apply professional standards and other normative legal documents in the field of software engineering;

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR11 – to choose the source data for design, guided by formal methods for describing requirements and modeling;

PR12 – to put into practice efficient approaches to software design;

PR14 – to apply in practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software;

PR16 – to have the skills of team development, coordination, design and release of all program documentation types;

PR18 – to know and be able to apply information technology for processing, storage and data transfer;

PR22 – to know and be able to apply methods and means of project management;

PR23 – to be able to document and present the results of software development.

In the second lecture of the section “Documentation”, G. Goos poses 8 questions that various types of documentation should answer (Table 2) [1, p. 386].

Table 2. Types of documentation (by G. Goos)

Questions	user's guide	conceptual description	design documentation	product documentation
How to use a program?	▨			
What is the state of the project?		▨	▨	▨
What are the overall specifications of the project?		▨	▨	
Which models are used to subdivide the program and to interface different modules?		▨	▨	
Which basic models are used for the modules?		▨	▨	
Flow of control and flow of data through the program			▨	
Detailed description of data.				
What is the meaning of the error-messages?	▨			

Short but comprehensive answers to these questions are one of the conditions for the success of the project – "if programmers cannot get a clear understanding how their work is related to the work of others they must fail because they usually start from wrong assumptions about their environment" [1, p. 387].

Creating a user manual (introductory manual, reference manual and operator's guide) starts first and ends last. The Introductory manual not only describes the standard ways of using the program, serving as a kind of "program cookbook", but also is the basis for its advertising and sale.

The conceptual description is developed during the project as its general description, the project documentation describes the current state of the project at the design phase and is the basis for the design phase, and the product documentation (including with the program text) describes the current state of the project at the design and maintenance phases. Inclusion in the final program text is necessary for cross-referencing interface modules, data, and algorithms.

Keeping documentation up-to-date is an important task for which Goos suggests using distributed text-based documentation systems with timestamp support.

Corresponding lecture program results of training according to 2018 Standard:

PR04 – to know and apply professional standards and other normative legal documents in the field of software engineering;

PR14 – to apply in practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software;

PR16 – to have the skills of team development, coordination, design and release of all program documentation types;

PR23 – to be able to document and present the results of software development;

PR24 – to be able to calculate the cost-effectiveness of software systems.

The third lecture "Performance Prediction" is one of the largest in the section. Its author Robert M. Graham (born 1929) was one of the first experts in the field of cybersecurity. He was responsible for security mechanisms, dynamic linking, and other key components of the operating system kernel in the Multics project.

Using the operating systems design and development as an example, the author offers criteria and methods for evaluating the performance of software systems, and also analyzes the impact of these scopes on software development and implementation processes. In this context, the author considers performance as the efficiency of using system resources when realizing software goals. The author emphasizes the importance of modeling systems to assess their efficiency and productivity. According to the author, the model of a software system reflects the relationship between the variables of this system to varying degrees. The model complexity is directly related to the system complexity and the ways to use it. For simple systems, the model will be so simple that it can exist only in the designer imagination. Due to increase of complexity, there is a need to reflect the system model in some exact and formal way. Graham notes that a complete, detailed description of a system, for example, its program code, is also actually the model. As a rule such a model is not useful, because it contains a large amount of redundant information and does not clearly demonstrate the relationship between the main variables: “A model is an abstraction containing only the significant variables and relations. Hence, it is usually much simpler than the system which it models. How much simpler will depend heavily on the expected use of the model and the precision desired in the results of its use” [1, p. 404].

Conceptually, a model is a function or a set of functions in which the system parameters used to characterize system performance are expressed as functions of the main variables of the system. The author emphasizes that “the performance of a system is not a constant number, rather it is a function, or several functions, whose values depend on the input. In order to characterize the performance of a given system we have to express these functions and this expression is a model of the system” [1, p. 404].

Thus, an important component of the professional training of specialists in software engineering is the mastery of the methods and means of formal description of software system models.

The author identifies two basic types of models: analytical and logical. The analytical model does not reflect the system structure. Actually, it is a set of mathematical equations which express the relations which exist between the basic system variables and the performance parameter. These equations are then solved for the dependent variables, that is, for the performance parameters. After solving these equations the system’s performance is completely known since graphs of the performance parameters can be plotted from the resulting mathematical expressions. On other hand a logical model mirrors closely the structure of the system being modeled. It is important to obtain accurate expressions for evaluating performance from this model. However, a logical model often includes mathematical equations which express some of the relations between variables.

Graham provides a description of processor time planning algorithms in operating systems as an example of the analytical model implementation. The author emphasizes the stochastic nature of such a model, but anticipating certain ranges of variables values in the system and the probability of these values, we can evaluate the expected performance of the system.

According to the author, one of the simplest and most obvious logical system model is its representation in the form of a directed graph. Examples of such graphs are flowcharts. In the general case, such a graph nodes are certain system states (variables values), and arcs are the processes or actions that are necessary for the transition from one state to another. We get the opportunity to evaluate and measure system performance by evaluating the time or other resources needed for such transitions.

Graham pays particular attention to simulation models as the most general and flexible way to evaluate the performance of software systems and demonstrates the use of these models in the operating systems design.

Among the problems that arise in the development of performance models, Graham underlines the model adequacy, the characterization of the tasks or desired system properties, and the simulation results interpretation. The author suggests using specialized modeling languages to describe the models, which should contain the ability to describe classes and their attributes, activities and events, as well as support queues, various probability distributions, and tools for collecting and analyzing data. The fundamental nature of this approach is emphasized by the fact that with the 4 general-purpose modeling languages that the author mentions – GPSS, SIMSCRIPT, SIMULA and CSL – only CSL is currently obsolete. Simula 67 became the basis for the development of C ++ and Java, and the latest versions of aGPSS (1.30) and SIMSCRIPT III (Release 5.0) which is generally date from 2019.

Special-purpose modeling languages are modeling languages of operating systems, both existing and those that have not yet been created. Simulation systems for these languages contain information about the hardware and language constructs for its description. The third lecture of the section ends with examples of such languages use.

Corresponding lecture program results of training according to 2018 Standard:

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering;

PR10 – to conduct a pre-design survey of the subject area, a system analysis of the design object;

PR14 – to apply in practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software;

PR20 – to know the approaches for evaluating and ensuring the quality of software.

The fourth lecture "Performance measurements" by Calvin Carl "Kelly" Gotlieb (1921 – 2016) has been called the "Father of Computing" in Canada – in 1948 he was part of the first team in Canada to build computers and to provide computing services, and in 1950, he created the first university course on computing in Canada. Gradually, his interests shifted towards the socio-economic consequences of ICT use, which led to his authorship and the chapter on the economics of software engineering.

C. C. Gotlieb indicates that performance measurements are needed when:

- installing a new computing system;

- changing the configuration or “tuning’ it to improve throughput;
- comparing systems to determine technological improvements, economies of scale and cost/benefit ratios.

The author refers to measuring performance methods:

1. Establishment of quality indicators (figures of merit) based on the rating (weight) of system components, in particular – empirical Grossch’s law in 1953, formulated by Herbert Reuben John Grossch (1918 – 2010): “giving added economy only as the square root of the increase in speed-that is, to do a calculation ten times as cheaply you must do it one hundred times as fast” [3, p. 310]. Gotlieb points out that value should be the common indicator of productivity, formulating Grosch’s law as

   $$C = K\sqrt{E},$$

   where C is the cost, K is a constant, E is the effectiveness measured in speed, throughput etc. Thus, productivity is proportional to the square of the cost.

   The author proposes for a more accurate assessment to determine the system performance by linking attributes number with each characteristic of the system, taking into account the weight of each attribute determined by the expert assessment method. The overall performance metric is calculated as a weighted sum of characteristics.

2. Launching a set of “kernel”, “benchmark” or synthetic tasks. By “core” (kernel) the author understood a general-purpose benchmark program, for each component of which the necessary measurements (runtime, etc.) were made. Then the performance of two computer systems was compared one relative to the other by launching a “kernel” on each of them. A “benchmark” program is a special program designed to evaluate performance (performance test). Performance measurement was a side effect in the “kernel”, but it is the main purpose for a “benchmark” program. Synthetic programs are designed to comprehensively check the stability of the system in normal and forced modes. Today, all these types of tasks are considered as a component of performance test (benchmarking).
3. Observing and measuring using hardware and software monitors at three levels:

   - the number of program calls, the estimated time to complete the task, the time spent on for completing of the task stages, the selected parameters of the execution time, kernel utilization, reading and writing, printing, rotation time, selected priorities, cost, diagnostics called at the system level is measured at the user tasks level;
   - the resource distribution, the channels and input/output operations activity, the task queues and system queues length, service time, etc. are measured at the system level;
   - the traffic and task flows, service utilization, resource allocation, operator actions and interventions, user requests, requests and complaints, expense and income statistics are measured at the hardware level.

4. Analytical or simulation modeling of the systems.

The last method is perhaps the only tools available at the software design stage when a preliminary performance assessment is performed. The first three methods are more often used to evaluate existing systems and alternative configurations.

Corresponding lecture program results of training according to 2018 Standard:

PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development;

PR19 – to know and be able to apply software verification and validation methods;

PR20 – to know the approaches for evaluating and ensuring the quality of software.

In the fifth lecture of “Pricing mechanisms” section by C. C. Gotlieb shows that pricing serves an important role in allocating service resources and rationalizing planning. Price levels are determined by costs, but also by policy considerations. The different methods of setting levels, along with some of the resulting implications and requirements are examined in the lecture.

The author refers to the main cost components as salaries (specialists in management, operational, applications, development) and fringe benefits (pension, insurance, health plan contributions etc.), equipment (purchase or rental payments, maintenance, communication costs, office equipment), supplies (cards, paper, tapes, documentation), software (purchased, leased, developed in-house), site (space, preparation costs, utilities), overhead (use of purchasing and maintenance services, library), miscellaneous (travel, advertising, user manuals, etc.).

The author analyzes the computer services market, which was formed at that time, and offers examples of various pricing models.

Corresponding lecture program results of training according to 2018 Standard:

PR18 – to know and be able to apply information technology for processing, storage and data transfer;

PR24 – to be able to calculate the cost-effectiveness of software systems.

Hans Jørgen Helms (1931–2010) was one of the co-chairs of the 1968 NATO conference program committee and the author of the last lecture in the section “Evaluation in the Computing Center Environment”, which completes the course. In 1965-1974 he was one of the leading employees of the first Scandinavian supercomputer center – Northern Europe University Computing Center at Technical University of Denmark. Working since 1974 as part of the European Commission, he headed the Joint Research Center of the Science, Research and Development Directorate, which he left in 1995 as Director-General.

Helms points out that this lecture relates not so much to software engineering as to the use of designed and developed programs to provide various services to various user groups. Therefore, the author refers to the computer center environment as a people community using the services of a particular computer system: an airline ticket agent uses a seat reservation system; typist uses a text editing system; bank teller uses an online accounting system; a manager uses a management information system; consulting engineer uses standard engineering programs from the terminal in his office; chemist develops programs to solve own research problems; a student solves tasks from a computer science course; programmer develops programs for the customer, etc. [1, p. 504].

Computing Centre environment is a large, distributed user community that accesses the computing services of remote computers through network terminals. Based on the analysis, the author indicates the directions of the corresponding software optimization for various indicators (runtime, service time, etc.) in order to meet the needs of a growing number of users and avoid potential service "bottle-necks".

The author notes the commonality of university computer centers that provide services without considering their profitability with other providers of utilities, such as postal or transport. As was shown in [5], it was precisely such utility computing services that later transformed into cloud technologies [4].

Corresponding lecture program results of training according to 2018 Standard:

PR08 – to be able to develop a human-machine interface;

PR18 – to know and be able to apply information technology for processing, storage and data transfer;

PR21 – to know, analyze, choose, and skillfully apply information security (including cybersecurity) means and data integrity in accordance with the tasks being solved and the software systems being created.

3 Conclusions

1. The first model of training in software engineering developed under the leadership of F. L. Bauer not only summarized existing experience of the early 1970s of training various software engineering components but also systematized it into the relevant training components. Despite the almost fifty-year age of the course materials, a significant part of them became the fundamental foundation of software engineering, determined the further development of the corresponding theory and empirical generalization.
2. The systematic nature (all sections and chapters are interconnected and cross-linked) and scalability (from a two-week intensive to preparing a bachelor's degree) of the proposed training program largely corresponds to the idea of designing evolutionary software. The analysis of the course content not only showed a number of problems regarding the relationship between the stable (fundamental) and rapidly changing (technological) components of the training content, but also provided an opportunity to identify the links between the training of a specialist in software engineering and training in computer science, computer engineering, cybersecurity, information systems and technology.
3. The analysis of the ratio of the program results of training SE specialists according to F. L. Bauer model (1972) and 2018 Standard provided the opportunity to establish that:
   a. the ability to analyze, purposefully search and select information and reference resources and knowledge necessary for solving professional tasks, taking into account modern achievements of science and technology, should be formed in academic disciplines preceding professional practical disciplines of SE specialist

training and developed in the process of professional training and further professional activity;

b. the acquisition of knowledge of professional ethics codes, understanding of the social significance and cultural aspects of software engineering and their observance in professional activities require separate focused work of the teacher and students both in a separate academic discipline and in the process of professional training;
c. the fundamental core of SE specialist training should ensure that students achieve these leading learning outcomes: PR07 – to know and put into practice the fundamental concepts, paradigms and basic principles of the functioning of the linguistic, instrumental and computational tools of software engineering; PR05 – to know and apply appropriate mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development; PR14 – to apply in practice the software tools for domain analysis, design, testing, visualization, measurement and documenting of software. The development of the relevant competencies of future SE specialists must be carried out in teaching all the disciplines of professional training;
d. F. L. Bauer model (1972) paid insufficient attention to the engineering and socio-economic foundations of the SE specialist professional activities, which are largely reflected in 2018 Standard.

4. Prospects for further research include an analysis of the relationship between the content of the general professional competencies of the bachelor of SE according to 2018 Standard and alternative domestic and foreign standards.

References

1. Bauer, F.L. (ed.). Software Engineering: An Advanced Course. Lecture Notes in Computer Science, vol. 30) (Formerly published 1973 as Lecture Notes in Economics and Mathematical Systems, vol. 81) Springer-Verlag, Berlin, Heidelberg, New York (1975). doi:10.1007/3-540-07168-7
2. Buxton, J.N., Randell, B. (eds.): Software Engineering Techniques: Report on a Conference sponsored by the NATO Science Committee, Rome, Italy, 27th to 31st October 1969. Scientific Affairs Division, NATO, Brussels (1970)
3. Grosch, H.R.J.: High Speed Arithmetic: The Digital Computer as a Research Tool. Journal of the Optical Society of America **43**(4), 306–310 (1953). doi: 10.1364/JOSA.43.000306
4. Markova, O.M., Semerikov, S.O., Striuk, A.M., Shalatska, H.M., Nechypurenko, P.P., Tron, V.V.: Implementation of cloud service models in training of future information technology specialists. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018), Kryvyi Rih, Ukraine, December 21, 2018. CEUR Workshop Proceedings **2433**, 499–515. http://ceur-ws.org/Vol-2433/paper34.pdf (2019). Accessed 10 Sep 2019
5. Markova, O.M., Semerikov, S.O., Striuk, A.M.: The cloud technologies of learning: origin. Information Technologies and Learning Tools **46**(2), 29–44 (2015). doi:10.33407/itlt.v46i2.1234

6. Naur, P., Randell, B. (eds.): Software Engineering: Report on a Conference sponsored by the NATO Science Committee, Garmisch, Germany, 7th to 11th October 1968. Scientific Affairs Division, NATO, Brussels (1968)
7. Randell, B.: Fifty Years of Software Engineering - or - The View from Garmisch. arXiv:1805.02742 [cs.SE]. https://arxiv.org/abs/1805.02742 (2018). Accessed 21 Mar 2019
8. Standart vyshchoi osvity Ukrainy: pershyi (bakalavrskyi) riven, haluz znan 12 – Informatsiini tekhnolohii, spetsialnist 121 – Inzheneriia prohramnoho zabezpechennia (Higher education standard of Ukraine: first (bachelor) level, field of knowledge 12 – Information technologies, specialty 121 – Software engineering). https://mon.gov.ua/storage/app/media/vishcha-osvita/zatverdzeni%20standarty/12/21/121-inzheneriya-programnogo-zabezpechennya-bakalavr.pdf (2018). Accessed 25 Oct 2019
9. Striuk, A.M.: Software engineering: first 50 years of formation and development. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 1st Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2018), Kryvyi Rih, Ukraine, November 30, 2018. CEUR Workshop Proceedings **2292**, 12–36. http://ceur-ws.org/Vol-2292/paper01.pdf (2018). Accessed 21 Mar 2019

An empirical comparison of machine learning clustering methods in the study of Internet addiction among students majoring in Computer Sciences

Oksana Klochko[0000-0002-6505-9455]

Vinnytsia Mykhailo Kotsiubynskyi State Pedagogical University,
32, Ostrozhskogo Str., Vinnytsia, 21100, Ukraine
klochkoob@gmail.com

Vasyl Fedorets[0000-0001-9936-3458]

Institute of Higher Education of the NAES of Ukraine,
9, Bastionna Str., Kyiv, 01014, Ukraine
bruney333@yahoo.com

Abstract. One of the relevant current vectors of study in machine learning is the analysis of the application peculiarities for methods of solving a specific problem. We will study this issue on the example of methods of solving the clustering problem. Currently, we have a considerable number of learning algorithms which can be used for clustering. However, not all methods can be used for solving a specific task. The article describes the technology of empirical comparison of methods of clustering problem solving using WEKA free software for machine learning. Empirical comparison of data clustering methods was based on the results of a survey conducted among students majoring in Computer Sciences and dedicated to detecting signs of Internet addiction (IA) as behavioural disorder that occurs due to Internet misuse. Empirical comparison of Expectation Maximization, Farthest First and K-Means clustering algorithms together with the application of the WEKA machine learning system had the following results. It described the peculiarities of application of these methods in feature clustering. The authors developed data instances' clustering models to detect signs of Internet addiction among students majoring in Computer Sciences. The study concludes that these methods may be applicable to development of models detecting respondent groups with signs of IA related disorders.

Keywords: Empirical Comparison, Machine Learning, Clustering, Internet addiction (IA), IA detection, Internet disorders, Expectation Maximization, Farthest First, K-Means.

1 Introduction

One of research directions in machine learning is the empirical analysis of methods of solving a specific problem. Let us study this issue on the example of methods of solving the clustering problem. Clustering methods are statistic methods of data analysis that enable people to group the given selection of data samples into clusters, classes, taxons depending on the value of their attributes; each of these groups has certain characteristics. The main idea is to use several clustering methods in order to carry out an empirical comparison study and determine which methods ensure the most optimal data grouping while solving a specific problem.

Machine learning classifies clustering problems as problems for unsupervised learning. Currently, there is a considerable number of machine learning algorithms that can be used for clustering, for instance, Expectation Maximization, K-Means, Hierarchical Clustering etc. But not all of them are suitable for solving a specific problem. Data clustering algorithms differ by the cluster model type, the algorithm model type, the nesting hierarchy of clusters, the way of implementation depending on the data set etc. Because of this, there are also certain requirements to the data set parameters.

Popular software products used in machine learning include TensorFlow, WEKA, MATLAB, MXNet, Torch, PyTorch, Microsoft Azure Machine Learning Studio and others.

In this article, we use the WEKA (Waikato Environment for Knowledge Analysis) free machine learning software [19]. The free WEKA machine learning system gives direct access to the library of implemented algorithms written in Java.

Analysis of contemporary studies and publications shows that the issue of analysis and selection of the machine learning method, which would be optimal for processing a concrete data set, is popular in the scientific circles. A considerable number of these studies is dedicated to the application of machine learning methods in the fields of healthcare and life safety.

In their article *A Performance Comparison of Machine Learning Classification Approaches for Robust Activity of Daily Living Recognition* scientists Rida Ghafoor Hussain, Mustansar Ali Ghazanfar, Muhammad Awais Azam, Usman Naeem and Shafiq Ur Rehman studied the application of the machine learning classification methods to find ways to ensure independent daily living of people who have Alzheimer’s disease [7]. The idea of the study is to analyze the data registered by different equipment in order to determine the changes in a person’s behavior that are relevant for the daily life and social interaction. The paper gives a comparison of the efficiency levels of five machine learning classification techniques used for the recognition of a person’s activity (and his/her psychological status). Experimental findings show that compared to traditional methodologies, these approaches give better results in determining the activity of the person and his/her psychological and behavioral peculiarities.

Jonas Krämer, Jonas Schreyögg and Reinhard Busse studied the speed and efficiency of medical aid provision using the databases of the Hospital ER [13]. Applying the Random forest algorithm, the authors developed the model based on the data about the patient’s provisional diagnosis. The use of the controlled machine learning method and model training based on the opinion of a specialized doctor allowed them to achieve high forecasting accuracy (96%) as well as the area under the receiver operating curve (>0.99).

Abdulhamit Subasi, Jasmin Kevric and M. Abdullah Canbaz developed a hybrid model of detecting epileptic fits using the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) to determine the optimal parameters of application of the Support Vector Machine (SVM) algorithm [17]. The hybrid algorithm that they suggested can demonstrate data set classification accuracy of up to 99.38%.

A considerable number of papers appeared, which are dedicated to diagnosing Internet addiction (IA) and studying the mechanisms of this disorder among various social groups. The appearance and use of the Internet has many benefits. However, at the same time, disorders related to pathological use of the Internet are becoming a social as well as a psychological problem. Currently, we face an important psychological, sociocultural and educational issue of detection and prevention of certain pathologies and steady premorbid conditions (state before the disease) caused by inadequate Internet use. Cases of IA were first mentioned in 1995 and attracted considerable attention. Issues related to this one became the research subject of many scientists, including Lyudmyla Yuryeva and Tatyana Bolbot [10], Marharyta Derhach [4] and others. Internet Addiction Disorder (IAD) is also called Pathological Internet Use (PIU). The term “Internet Addiction” was first suggested by Ivan K. Goldberg in 1995. He describes net addiction as a specific pathology characterized by a wide spectrum of behavioral and impulse control disorders (lack of control, absence of voluntary regulation) [1]. In 1996 Goldberg made the first attempt to determine groups of behavioural and psychological signs and symptoms of IA [18], namely: tolerance; abstinence syndrome; difficulties in voluntary regulation of Internet-behaviour; increase of time and financial investments in things related to Internet or computer use; a shift of a person’s interests towards Internet-related activities; extensive Internet use that leads to maladjustment. In 1998 Kimberly S. Young defined IAD as an impulsive-compulsive disorder, which has specific signs or addictions [20; 21]: cyber-sexual addiction, cyber-relationship addiction, net compulsions, information overload and computer addiction. IAD is not officially included into ICD-11 for Mortality and Morbidity Statistics (Version: 04/2019), however, in section 6C51 Gaming disorder the “Gaming disorder” is described as a “pattern of persistent or recurrent gaming behaviour (‘digital gaming’ or ‘video-gaming’), which may be online (i.e., over the Internet)” [8].

Even though the problem of IA is becoming more and more relevant, there are not enough scientific papers dedicated to the study of this issue with the help of machine learning methods. Let us look at some of them. On the basis of the Support Vector Machine algorithm, including the C-SVM and ν-SVM, and applying the Student’s t-test to the data set of the survey conducted among 2,397 Chinese students, scientists Zonglin Di, Xiaoliang Gong, Jingyu Shi, Hosameldin O. A. Ahmed and Asoke K. Nandic proved the utility of using machine learning methods for detecting and forecasting the risk of IA [5]. Wen-Huai Hsieh, Dong-Her Shih, Po-Yuan Shih and Shih-Bin Lin suggested using the EMBAR protected system of web-services based on the ensemble classification methods and case-based reasoning to study the IA of the users and prevent the development of this disorder at the initial stages [6]. Hong-Ming Ji, Liang-Yu Chen and Tzu-Chien Hsiao are currently continuing their research, which aims to create an IA detector that would work in a real-time mode [9]. The authors suggest studying this issue using an adapted system of continuous real-coded variables (XCSR), which determines the level of Internet addiction (high-risk and low-risk) on the basis of the information about the Internet users using the Chen Internet addiction scale (CIAS) or respiratory instantaneous frequency (IF) [9].

Thus, based on the above presented statement of the problem as well as taking into consideration the insufficient amount of research on the application of machine learning methods to IA diagnosing, we determine the aim of our research, which is to conduct an empirical comparison of clustering methods within the WEKA machine learning system in the course of studying the IA disorder among students majoring in Computer Sciences.

2 Selection of methods and diagnostics

Data regarding the spread and severity of IA among students majoring in Computer Sciences were received from an online survey, which used a questionnaire drafted with the help of Google Forms. 263 students majoring in Computer Sciences and coming from different oblasts of Ukraine participated in the experimental study. The data set is presented in the ARFF format and consists of 8 attributes (Fig. 1). The data set contains the fields described in Table 1.

```
@relation answer_IA

@attribute age numeric
@attribute sex {female,male}
@attribute 3 {no,undefined,yes}
@attribute 4 {no,undefined,yes}
@attribute 5 {no,undefined,yes}
@attribute 6 {no,undefined,yes}
@attribute 7 {no,undefined,yes}
@attribute 8 {no,undefined,yes}

@data
18,male,yes,no,no,no,no,yes
28,male,undefined,no,no,no,no,yes
20,female,yes,yes,yes,no,no,no
22,male,yes,no,no,no,no,no
…
```

Fig. 1. Data set on the state of IA among students majoring in Computer Sciences, presented in the ARFF format

Cluster analysis is one of the tasks of database mining. *Cluster analysis* is a set of methods of multidimensional observations or objects classification, based on defining the concept of distance between the objects and their subsequent grouping (into clusters, taxons, classes). The selection of a concrete cluster analysis method depends on the purpose of classification [12]. At the same time, one does not need a priori information about the population distribution. This approach is based on the following presuppositions: objects that have a certain number of similar (different) features group in one segment (cluster). The level of similarity (difference) between the objects that belong to one segment (cluster) must be higher than the level of their similarity with the objects that belong to other segments [12].

Table 1. Data structure on the state of IA among students majoring in Computer Sciences.

Attributes	Contents/Questions	Type	Statistics
age	Age of the student	Numeric	Minimum 16 Maximum 59 Mean 19.756 StdDev 6.806
sex	Student's sex	Nominal	Female 199 Male 63
3	Can't imagine my life without the Internet	Nominal	yes 184 undefined 39 no 39
4	When I cannot use the Internet I fell anxiety, irritation	Nominal	yes 81 undefined 134 no 47
5	I like "surfing" the Net without a clearly defined purpose	Nominal	yes 121 undefined 112 no 29
6	I can abstain from food, sleep, going to classes, if a have a chance to use the Internet for free	Nominal	yes 248 undefined 7 no 7
7	I prefer meeting new people over the Internet rather than in real life	Nominal	yes 185 undefined 37 no 40
8	I often feel that I've spent not enough time playing computer games over the Internet, I constantly wish to play longer	Nominal	yes 178 undefined 61 no 23

Let us look at one of cluster analysis algorithms [12].

Output matrix:

$$\mathrm{X} = \begin{pmatrix} \mathrm{x}_{11} & \cdots & \mathrm{x}_{1n} \\ \vdots & \ddots & \vdots \\ \mathrm{x}_{m1} & \cdots & \mathrm{x}_{mn} \end{pmatrix}.$$

Let us move to the matrix of standardized Z values with elements:

$$\mathrm{z}_{ij} = \frac{\mathrm{x}_{ij} - \bar{\mathrm{x}}_j}{\mathrm{s}_j};$$

where j = 1, 2, …, n – index number, i = 1, 2, ... , m – observation number;

$$\mathrm{\bar{x}_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij}};$$

$$\mathrm{s_j = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(x_{ij}-\bar{x}_j)^2} = \sqrt{\overline{(x_{ij}^2)}-(\overline{x_j})^2}}.$$

There are several ways to define the distance between two observations z_i and z_v:

1. weighted Euclidean distance, which is determined by the formula

$$\mathrm{\rho_{BE}(z_i, z_v) = \sqrt{\sum_{l=1}^{n} w_l(z_{il}-z_{vl})^2}};$$

where w_l is the "weight" of index; $0<w_l\leq1$; if $w_l=1$ for all $l = 1, 2, \ldots, n$, then we get the usual Euclidean distance

$$\mathrm{\rho_{BE}(z_i, z_v) = \sqrt{\sum_{l=1}^{n}(z_{il}-z_{vl})^2}};$$

2. Hamming distance:

$$\mathrm{\rho_{BH}(z_i,z_v)} = \sum_{l=1}^{n} |\mathrm{z}_{\mathrm{i}l}\text{-}\mathrm{z}_{\mathrm{v}l}|;$$

in most cases this way of distance measuring gives the same result as the usual Euclidean distance, but in this case the influence of non-systemic large differences (runouts) decreases;

3. Chebyshev distance:

$$\mathrm{\rho_{BCH}(z_i,z_v)} = \max_{1\leq l\leq n}|\mathrm{z}_{\mathrm{i}l}\text{-}\mathrm{z}_{\mathrm{v}l}|;$$

it is best to apply this distance in order to determine the differences existing between the two objects using only one dimension;

4. Mahalanobis distance:

$$\mathrm{\rho_{BM}(z_i,z_v)} = \sqrt{(z_i-z_v)^T S^{-1}(z_i-z_v)},$$

where S is covariance matrix; this distance measurement gives good results when applied to a concrete data group, but it does not work very well, if the covariance matrix is calculated for the whole data set;

5. Distance between peaks:

$$\mathrm{\rho_{BL}(z_i,z_v)} = \frac{1}{n}\sum_{l=1}^{n}\frac{|\mathrm{z}_{\mathrm{i}l}\text{-}\mathrm{z}_{\mathrm{v}l}|}{\mathrm{z}_{\mathrm{i}l}+\mathrm{z}_{\mathrm{v}l}};$$

presupposes independence of random variables, which indicates the distance in the orthogonal space.

It is best to choose from the above described distance measures after the consideration of the structure and characteristics of the data sample.

Let us present the received measurements in the form of distance matrix:

$$R = \begin{pmatrix} 0 & \rho_{12} & \rho_{13} & \cdots & \rho_{1m} \\ \rho_{21} & 0 & \rho_{23} & \cdots & \rho_{2m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \rho_{i1} & \rho_{i2} & \rho_{i3} & \cdots & \rho_{im} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \rho_{m1} & \rho_{m2} & \rho_{m3} & \cdots & 0 \end{pmatrix}.$$

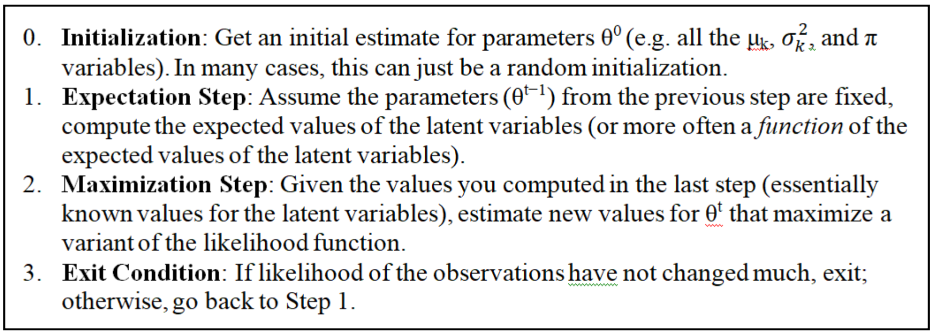
As the R matrix is symmetric, i.e. $\rho_{iv}=\rho_{vi}$, we may confine ourselves to off-diagonal matrix elements.

Using the distance matrix, we can implement the agglomerative hierarchic procedure of cluster analysis. Distances between clusters are determined as the closest or the farthest ones. In the first case, the distance between the clusters is the one between the closest elements of these clusters, in the second case, it is the one between the two farthermost located. The principle of the work of agglomerative hierarchic procedures lies in a consequent grouping of elements, starting from the ones closest to each other and those that are farther and farther apart. During the first step of the algorithm, every observation z_i (i = 1, 2, ..., m) is viewed as a separate cluster. Then, during every next step of the work of the algorithm, two closest located clusters are grouped together and then once again the distance matrix is built, but its dimension decreases by one. The algorithm stops its work when all the observations are grouped into clusters.

Let us look at the algorithms we used while clustering the data set regarding the state of IA disorder among students majoring in Computer Sciences:

1. EM (Expectation Maximization)

Determines the probability distribution for every object, which indicated its belongingness to each cluster. EM methods [11]: Maximum Likelihood Estimation (MLE) or Maximum a Posteriori (MAP). Description of the algorithm is shown in Fig. 2 [11]: at the E-stage (expectation) we calculate the estimated likelihood; at the M-stage (Maximization) we calculate the maximum likelihood estimation, increasing the expected likelihood, calculated at the E-stage; its value is used for the E-stage at the next iteration. The algorithm is repeated until its convergence.

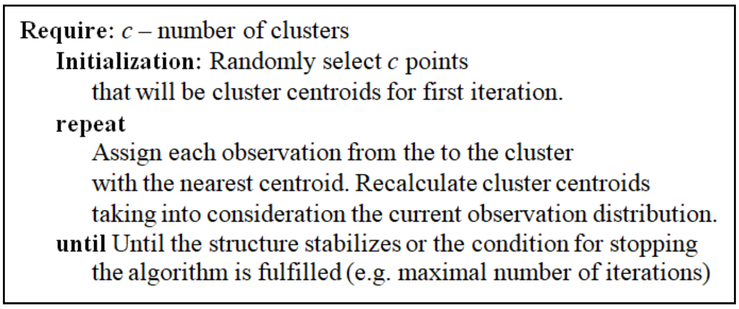


0. **Initialization**: Get an initial estimate for parameters θ^0 (e.g. all the μ_k, σ_k^2, and π variables). In many cases, this can just be a random initialization.
1. **Expectation Step**: Assume the parameters (θ^{t-1}) from the previous step are fixed, compute the expected values of the latent variables (or more often a *function* of the expected values of the latent variables).
2. **Maximization Step**: Given the values you computed in the last step (essentially known values for the latent variables), estimate new values for θ^t that maximize a variant of the likelihood function.
3. **Exit Condition**: If likelihood of the observations have not changed much, exit; otherwise, go back to Step 1.

Fig. 2. Description of how the algorithm EM works from 10,000 feet [11].

2. K-Means algorithm

Aims to partition n observations into k clusters in such a way that each observation belongs to the cluster with the nearest mean value. The shortest distance between the observations and the nearest mean value may be calculated by minimizing the sum of squares of the distances [14] (Fig. 3).

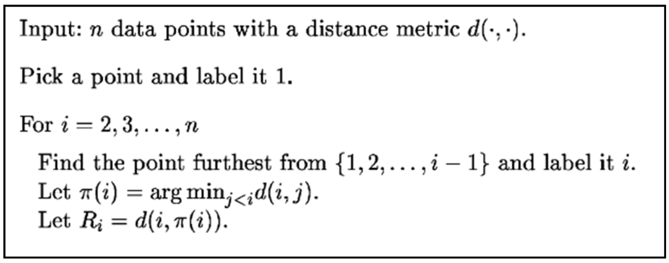


```
Require: $c$ – number of clusters
    Initialization: Randomly select $c$ points
        that will be cluster centroids for first iteration.
    repeat
        Assign each observation from the to the cluster
        with the nearest centroid. Recalculate cluster centroids
        taking into consideration the current observation distribution.
    until Until the structure stabilizes or the condition for stopping
        the algorithm is fulfilled (e.g. maximal number of iterations)
```

Fig. 3. K-Means algorithm [14].

3. Farthest First algorithm

This is a modification of a K-Means algorithm, in which the initial selection of centroids is 2 and higher. Centroids are determined following the remoteness principle, i.e. the point farthest from the rest is selected first. The Farthest First algorithm is described in Fig. 4 [3].



```
Input: $n$ data points with a distance metric $d(\cdot,\cdot)$.

Pick a point and label it 1.

For $i = 2, 3, \ldots, n$
    Find the point furthest from $\{1, 2, \ldots, i-1\}$ and label it $i$.
    Let $\pi(i) = \arg\min_{j<i} d(i, j)$.
    Let $R_i = d(i, \pi(i))$.
```

Fig. 4. Farthest-first traversal of a data set [3]. Take the distance from a point x to a set S to be $d(x, S) = \min_{y \in S} d(x, y)$ [3].

3 Results and Discussion

To cluster data using the WEKA platform, we will use Weka.clusterers.EM, Weka.clusterers.SimpleKMeans and Weka.clusterers.FarthestFirst algorithms [19].

We check the application of clustering algorithms that can be assigned to two classes of clustering algorithms, i.e. distribution based (Expectation Maximization) and centroid-based (K-Means, Farthest First). Such selection is motivated by the fact these algorithms have long been used to cluster different types of data in many fields and are considered to be effective.

Dunn, DB, SD, CDbw and S_Dbw were selected as validity indices for testing [2; 15; 16] (Table 2). In the CDbw index the distance from the point to multitude set in the course of selecting cluster element can be calculated in different ways. In this study, we use the sum of distances of already existing "representatives" of the cluster to each cluster element to calculate this distance. The element, on which the maximum was reached, was selected as the next "representative" of the cluster.

Table 2. Optimal number of clusters, calculated with the help of quality indices.

Index	**Algorithms**		
	Expectation Maximization	**k-Means**	**Farthest First**
Dunn	3	6	6
DB	3	6	4
SD	3	3	3
CDbw	3	3	3
S_Dbw	3	5	4

If the data set has no cluster structure, then such situation is not determined with the help of validity metrics. While using K-Means and Farthest First (Table 2) the numbers of clusters for the two algorithms that were selected as optimal by the majority of indices, can only nominally be defined as cluster structure. As the work of Expectation Maximization algorithm is based on determining the probability of evaluating maximum similarity, the indices calculated for this algorithm are more homogenous. The structure, which is characterized by a small number of clusters that also have to be compact and separable, is determined to be the best one. Judging by the results of evaluation of clustering using the validity indices, we may consider that k-Means and Farthest First algorithms are most likely to give worse clustering results than the Expectation Maximization algorithms.

To cluster the data, we select training/testing using the percentage split option. As a data set for training (model building) we select 66% of data from the set. As a data set for testing we select 34% of data from the set. In addition, we select number of clusters "3" in algorithm settings.

We received the following results:

1. In the course of application of the EM clustering algorithm, according to the built clustering model based on the training data set, three clusters were determined, their characteristics are given in Table 3.

Cluster 0 (63% of respondents): The average age of respondents in this cluster is 17. The group consists predominantly of women. The characteristic feature of the representatives of this group is that they are unable to imagine their life without the Internet. There are variations in the levels of anxiety and irritation, if there is no possibility to use the Internet. There are also varying opinions regarding the aimless use of the Internet. As for other attributes, disorders related to IA may be observed in the insignificant number of respondents, who belong to this cluster. The behavioural model of the representatives of this cluster demonstrated Internet centration in the psychic reality of a personality, which is accordingly reflected in their activity and behavior, other life interests as well as significance of everyday activities lose their importance. The stated tendencies are linked to IA.

Table 3. Model and evaluation on test split by EM algorithm.

Attributes	**Indications**	**Clusters**		
		0 (0,63) 112.1491	**1 (0,13) 24.7781**	**2 (0,24) 44.0727**
age	mean	17.4469	36.2459	19.2906
	std. dev.	1.5994	10.0785	2.243
sex	female	108.8714	16.0638	5.0648
	male	2.2778	7.7143	38.0079
3	no	22.7034	3.1864	10.1102
	undefined	16.0405	6.4026	4.5569
	yes	73.4052	15.1891	29.4057
4	no	54.392	13.8263	27.7817
	undefined	23.6012	5.1903	7.2085
	yes	34.156	5.7615	9.0825
5	no	45.3302	19.3167	26.3531
	undefined	15.1791	2.1415	5.6794
	yes	51.6398	3.32	12.0403
6	no	106.1573	22.7561	41.0866
	undefined	1.0117	1.0098	1.9785
	yes	4.9802	1.0122	1.0076
7	no	81.1224	20.5492	27.3284
	undefined	11.5501	2.168	11.282
	yes	19.4767	2.061	5.4624
8	no	89.4444	19.3333	9.2223
	undefined	7.2533	1.1937	9.553
	yes	15.4514	4.2512	25.2975

Cluster 1 (13% of respondents): For the representatives of this group the average value of the age attribute is 36 and it varies greatly. This is the oldest age group if compared with other clusters. This group has the largest share of women. Representatives of this group, predominantly, cannot imagine their life without the Internet. Thus, according to the centroid values of the attributes, we may diagnose IA related Internet centration in the psychic reality of a personality, which is accordingly reflected in their activity and behavior; other life interests as well as significance of everyday activities lose their importance. There are predominantly no other signs of IA related disorders.

Cluster 2 (24% of respondents): The probabilistic average of the age attribute among the representatives of this group is middle-aged in comparison with other groups and is 19. Male representatives significantly dominate in this group. Regarding the inability to imagine their life without the Internet, opinions differed, however, predominantly respondents believe they have this addiction. Judging by the values of attributes 4, 5, 6 and 7, the vast majority of this group's representatives declare that they do not have other signs of IA. However, the feeling of the lack of time spent playing computer games over the Internet, which was confirmed by the vast majority of respondents, is a warning signal that may signify the existence of IA related disorders. Thus, the characteristic feature of this group is that most of its representatives have IA related disorders such as: Internet centration in the psychic reality of a personality; behavioral impulse control disorders related to online gaming. These people are in the risk group for developing IA related disorders.

2. In the course of application of the Farthest First algorithm, according to the built clustering model based on the training data set, there have also been three clusters formed; their characteristics are given in Table 4.

Table 4. Model and evaluation on test split by Farthest First algorithm.

Attributes	**Clusters**		
	0	**1**	**2**
age	16.0	22.0	20.0
sex	female	male	male
3	yes	undefined	yes
4	undefined	no	yes
5	no	yes	undefined
6	no	no	undefined
7	no	undefined	undefined
8	no	undefined	no

Cluster 0: Contains data instances of the youngest age group, whose age centroid attribute is 16. According to the value of the sex centroid attribute, the group is made up of mostly female data instances. The representatives of this group cannot imagine their life without the Internet, i.e. there is obvious Internet centration in the psychic reality of a personality. Respondents cannot clearly determine whether they feel either anxiety or irritation if they do not have the possibility to use the Internet. Judging by other attributes, data instances of this cluster do not have IA related disorders.

Cluster 1: This cluster contains data instances of an older age group, the age attribute centroid of which is 22. The value of the sex attribute centroid in this cluster is male. A characteristic feature of the cluster is undecidedness regarding the vital need to use the Internet, prevalence of Internet relations over actual real interactions, feeling the lack of time spent playing computer games over the Internet (attributes 3, 7, 8 equal undefined). The value of the yes centroid of attribute 5 shows inclination to use the Internet without a concrete purpose. To give an overall characteristic, this group has signs of IA, i.e. behavior control disorders related to Internet use.

Cluster 2: By the value of the age attribute centroid, 20, this cluster contains data instances of the middle age group if compared with other clusters. The sex attribute centroid in this cluster is male. The representatives of this cluster cannot imagine their life without the Internet and feel anxiety and irritation when they do not have the possibility to use the Internet. They are characterized by their undecidedness regarding the vital need to use the Internet; giving up other life interests and everyday activities for the sake of free Internet use; prevalence of online relations of real-life interactions (value of attributes 5, 6, 7 is undefined). Thus, the representatives of this cluster have signs of IA, the priority significance of the Internet and behavior control disorders, related to Internet use. Compared to other groups, they are in the risk group for developing IA related disorders.

3. In the course of application of the K-Means algorithm to the clustering model built on the basis of the training data set three clusters have also been formed, their characteristics are presented in Table 5.

Table 5. Model and evaluation on test split by K-Means algorithm.

Attribute	**Cluster**		
	0	**1**	**2**
age	18.4194	21.8605	20.9552
sex	female	male	female
3	undefined	yes	yes
4	no	no	no
5	no	no	no
6	no	no	no
7	no	no	no
8	no	yes	no

Cluster 0: Contains data instances of the youngest age group, whose age attribute centroid is about 18. According to the sex attribute centroid, mostly female data instances are present in the groups. The representatives of this group cannot clearly determine whether they have a vital need to use the Internet. As for other indices, respondents state absence of signs of IA related disorders.

Cluster 1: This cluster contains data instances of the older age group, whose age attribute centroid is about 22. The value of the sex attribute centroid in this cluster is male. Characteristic features of data instances that belong to this cluster include the vital need to use the Internet, feeling the lack of time spent playing online computer games as well as the systemic need to play longer. The overall characteristic of this cluster is the presence of signs of IA, i.e. behavior control issues related to Internet use, namely, gaming Internet addiction. If compared with other cluster, they belong to the risk group that may develop IA related disorders.

Cluster 2: By the value of age attribute centroid, which is about 21 years, compared to other clusters, this cluster contains data instances of medium age group. The sex attribute centroid is female. The representatives of this cluster cannot imagine their life without the Internet. Judging by centroids of other characteristics, respondents of this cluster do not have Internet-related disorders. Thus, the representatives of this cluster have only IA signs associated with the utmost significance of the Internet.

The cluster distribution of test data in the course of application of the three algorithms – the Expectation Maximization, Farthest First and K-Means – using the built training models is presented in Table 6. Thus, as it can be seen from the table, the algorithms have determined three data groups. Clusters were formed, which included 71:12:7, 67:4:19 and 33:15:42 data instances respectively. There is a cluster that has the largest number of data instances; a group, which has the least data instances (exceptions); a group that includes several times more data instances than the smallest group.

Table 6. Clustered Instances determined using Expectation Maximization, K-Means and Farthest First algorithms.

Clusters	Expectation Maximization		Farthest First algorithm		K-Means	
	Instances	%	Instances	%	Instances	%
0	67	74	71	79	33	37
1	4	4	12	13	15	17
2	19	21	7	8	42	47

Fig. 5, Fig. 6 and Fig. 7 present a graphic representation of clusters by age characteristic of data instances, which are built using the training data set and received in the course of implementation of the Expectation Maximization, the Farthest First and the K-Means algorithm respectively. As we can see, the formed clusters differ from each other by the age attribute. For instance, Cluster 0, which contains most data instances, contains instances of respondents of a younger age, if formed through the application of the Expectation Maximization algorithm (Fig. 5). On the other hand, the same cluster received through the implementation of the Farthest First algorithm, contains data instance of various age groups (Fig. 6). Also, a small number of data instances of various age groups is present in Cluster 2, received in the course of implementation of the K-Means algorithm (Fig. 7). Cluster 0 and Cluster 2 formed with the Expectation Maximization algorithm as well as Cluster 1 and Cluster 2 formed with the Farthest First algorithm contain homogeneous age groups, and Cluster 0 та Cluster 1, formed with K-Means algorithm.

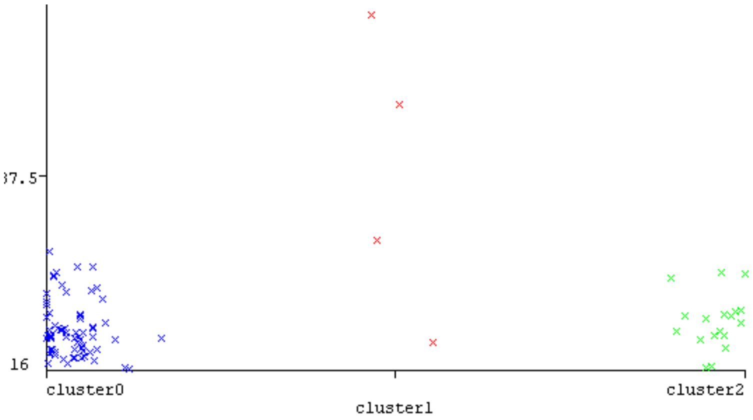


Fig. 5. Plot of cluster distribution applying the Expectation Maximization algorithm depending on the age group attribute

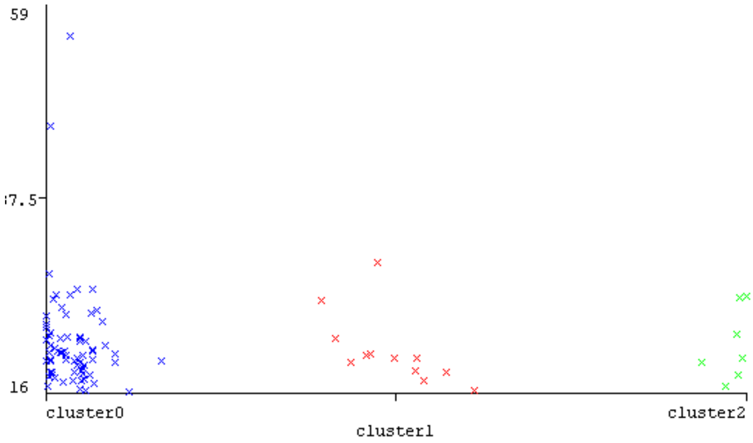


Fig. 6. Plot of cluster distribution applying the Farthest First algorithm depending on the age group attribute

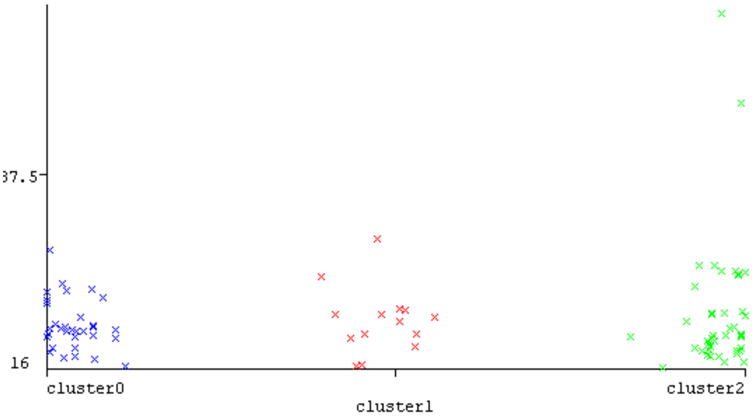


Fig. 7. Plot of cluster distribution applying the K-Means algorithm depending on the age group attribute

Fig. 8, Fig. 9 and Fig. 10 present a graphic representation by sex attribute of clusters formed through the application of the Expectation Maximization, Farthest First and K-Means algorithm respectively. The analysis of Fig. 8, which visualizes clustering through application of the Expectation Maximization algorithm, shows that Cluster 0 contains only female data instances. Clusters 1 and 2 have date instances of both sexes. Female data instances prevail in Cluster 1 and male ones in Cluster 2. Unlike Clusters formed by the Expectation Maximization algorithm, all the clusters formed by the Farthest First algorithm contain data instances of both sex groups (Fig. 9). Female data instances significantly prevail in Cluster 0. All the clusters built using the K-Means algorithm, contain both male and female data instances (Fig. 10).

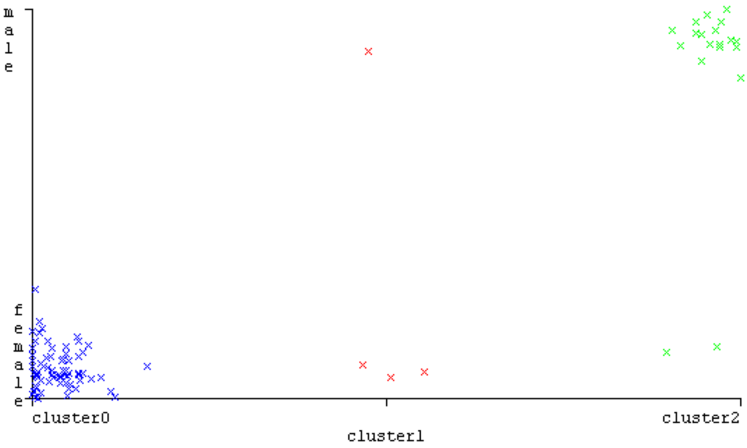


Fig. 8. Plot of cluster distribution applying the Expectation Maximization algorithm depending on the sex attribute

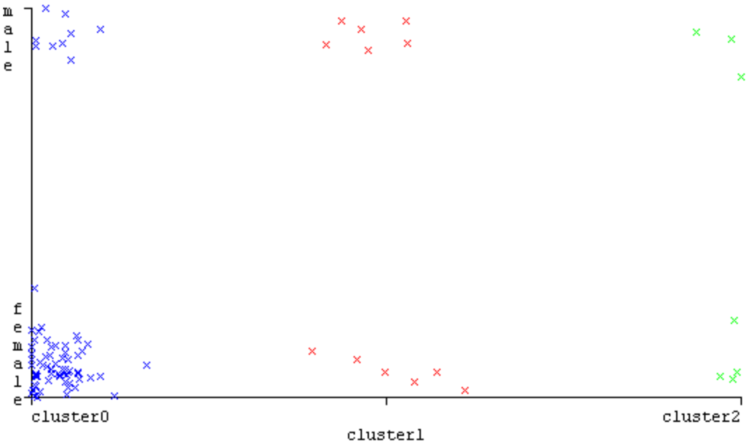


Fig. 9. Plot of cluster distribution applying the Farthest First algorithm depending on the sex attribute

4 Conclusion

In the course of empirical comparison of Expectation Maximization, Farthest First and K-Means algorithms using the WEKA machine learning system to study the signs of IA related disorders among the students majoring in Computer Sciences, the following conclusions have been made:

1. As a result of empirical comparison of Expectation Maximization, Farthest First and K-Means algorithms using the WEKA machine learning system, we developed

models of data instances' clustering to determine the signs of internet addiction disorders among students majoring in Computer Sciences.

2. The implementation of the Expectation Maximization, the K-Means and the Farthest First algorithms each resulted in the formation of 3 clusters. The results of clustering demonstrate that Internet centration in the psychic reality of a personality is a characteristic feature of the respondents that took part in the survey. This also reflects accordingly in their activity and behavior, diminishing other life interests and the significance of everyday activities. In addition, in the course of implementation of the Expectation Maximization algorithm, a cluster was formed, whose representatives have behavior control disorders, related to online gaming. These respondents are in the risk group for developing IA related disorders.
3. Expectation Maximization, Farthest First and K-Means algorithms of data clustering differ by their algorithm model, however, from the point of characteristic features, they produce relatively similar clusters, thus implementing optimized clustering. At the same time, when a data set was grouped into three clusters by implementing these algorithms, the clusters differed by cluster model, namely, by the number of data instances in each cluster, their structure and value of attribute centroids.
4. Judging by the evaluation results of clustering validity using the validity indices, we can state that most likely the K-Means and Farthest First algorithms show worse clustering results than the Expectation Maximization algorithm.
5. Intellectual analysis of the data set regarding the situation with IA among students majoring in Computer Sciences with the application of clustering methods has shown that the methods studied above may be considered suitable for developing models for detecting respondent groups with the signs of IA related disorders.

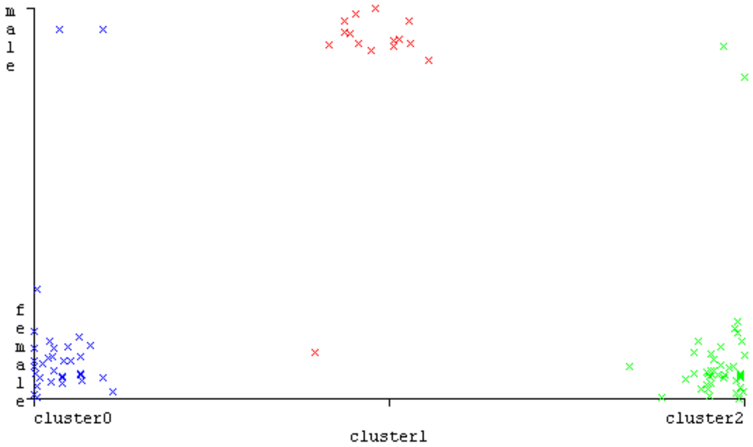


Fig. 10. Plot of cluster distribution applying the K-Means algorithm depending on the sex attribute

Our conclusions may help to determine the signs of IA related disorders among students majoring in Computer Sciences, forecasting the risk of IA and development of services aimed at IA prevention.

References

1. Abbott, D.A., Cramer, S.L., Sherrets, S.D.: Pathological Gambling and the Family: Practice Implications. Families in Society: the Journal of Contemporary Social Services **76**(4), 213–219 (1995). doi:10.1177/104438949507600402
2. da Silva, L.E.B., Melton, N.M., Wunsch II, D.C.: Incremental Cluster Validity Indices for Hard Partitions: Extensions and Comparative Study. arXiv:1902.06711 [cs.LG]. https://arxiv.org/pdf/1902.06711.pdf (2019). Accessed 25 Oct 2019
3. Dasgupta, S., Long, P.M.: Performance guarantees for hierarchical clustering. Journal of Computer and System Sciences **70**(4), 555–569 (2005). doi:10.1016/j.jcss.2004.10.006
4. Derhach, M.: Cyber-Addiction of Students Majoring in Computer Science. Science and Education 7, 92–98 (2016). doi:10.24195/2414-4665-2016-7-16
5. Di, Z., Gong, X., Shi, J., Ahmed, H.O.A., Nandi, A.K.: Internet addiction disorder detection of Chinese college students using several personality questionnaire data and support vector machine. Addictive Behaviors Reports **10**, 100200 (2019). doi:10.1016/j.abrep.2019.100200
6. Hsieh, W.-H., Shih, D.-H., Shih, P.-Y., Lin, S.-B.: An Ensemble Classifier with Case-Based Reasoning System for Identifying Internet Addiction. International Journal of Environmental Research and Public Health **16**(7), 1233 (2019). doi:10.3390/ijerph16071233
7. Hussain, R.G., Ghazanfar, M.A., Azam, M.A., Naeem, U., Rehman, S.U.: A performance comparison of machine learning classification approaches for robust activity of daily living recognition. Artificial Intelligence Review **52**(1), 357–379 (2019). doi:10.1007/s10462-018-9623-5
8. ICD-11 for Mortality and Morbidity Statistics (Version: 04/2019): 6C51 Gaming disorder. https://icd.who.int/browse11/l-m/en#/http://id.who.int/icd/entity/1448597234 (2019). Accessed 29 Aug 2019
9. Ji, H.-M., Chen, L.-Y., Hsiao, T.-C.: Real-time detection of internet addiction using reinforcement learning system. GECCO'19: Proceedings of the Genetic and Evolutionary Computation Conference Companion, pp. 1280–1288. ACM (2019). doi:10.1145/3319619.3326882
10. Jur'eva L.N., Bol'bot T.Ju.: Komp'juternaja zavisimost': formirovanie, diagnostika, korrekcija i profilaktika (Computer addiction: formation, diagnosis, correction and prevention). Porogi, Dnepropetrovsk (2006)
11. Keng, B.: The Expectation-Maximization Algorithm. http://bjlkeng.github.io/posts/the-expectation-maximization-algorithm (2016). Accessed 25 Aug 2019
12. Klochko, O.V.: Matematychne modeliuvannia system i protsesiv v osviti/pedahohitsi (Mathematical modeling of systems and processes in education/pedagogy). Vinnytsia (2019)
13. Krämer, J., Schreyögg, J., Busse, R.: Classification of hospital admissions into emergency and elective care: a machine learning approach. Health care management science **22**(1), 85–105 (2019). doi:10.1007/s10729-017-9423-5
14. Linoff, G.S., Berry, M.J.A.: Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management, 3rd edn. John Wiley & Sons, New York (2011)

15. Moshtaghi, M., Bezdek, J.C., Erfani, S.M., Leckie, C., Bailey, J.: Online Cluster Validity Indices for Streaming Data. arXiv:1801.02937 [stat.ML]. https://arxiv.org/pdf/1801.02937.pdf (2018). Accessed 25 Oct 2019
16. Moshtaghi, M., Bezdek, J.C., Erfani, S.M., Leckie, C., Bailey, J.: Online cluster validity indices for performance monitoring of streaming data clustering. International Journal of Intelligent Systems **34**(4), 541–563 (2019). doi:10.1002/int.22064
17. Subasi, A., Kevric, J., Canbaz, M.A.: Epileptic seizure detection using hybrid machine learning methods. Neural Computing and Applications **31**(1), 317–325 (2019). doi:10.1007/s00521-017-3003-y
18. Wallis, D.: Just Click No: Talk Story about Dr. Ivan K. Goldberg and the Internet Addiction Disorder. New Yorker Magazine. http://www.newyorker.com/magazine/1997/01/13/just-click-no (1997). Accessed 25 Oct 2019
19. Weka 3 - Data Mining with Open Source Machine Learning Software in Java. https://www.cs.waikato.ac.nz/~ml/weka (2019). Accessed 17 Aug 2019
20. Young, K.S.: Caught in the Net: How to Recognize the Signs of Internet Addiction – and a Winning Strategy for Recovery. John Wiley & Sons, New York (1998)
21. Young, K.S.: Internet addiction: The emergence of a new clinical disorder. CyberPsychology & Behavior **1**(3), 237–244 (1998). doi:10.1089/cpb.1998.1.237

Automated recognition and sorting of agricultural objects using multi-agent approach

Kateryna Ovchar, Andrii Borodin, Ivan Burlachenko[0000-0001-5088-6709]
and Yaroslav Krainyk[0000-0002-7924-3878]

Petro Mohyla Black Sea National University, 68, Desantnykiv Str., Mykolaiv, 54000, Ukraine
{kvchar80, electron.t5164, ivan.burlachenko2010}@gmail.com,
yaroslav.krainyk@chmnu.edu.ua

Abstract. In the following paper, we propose an alternative approach to automatic recognition and sorting of the agricultural objects. The objects are sorted, based on their color. The sorting algorithm works with pictures, captured by a web camera, pre-processes them, determines the mass centers of the recognized objects, and calculates distances to the points which were found. Afterward, the recognized objects are sorted by a hand-manipulator with four degrees of freedom. The manipulator is powered by Arduino Uno Rev 3 and servo motors of two types. The sorting algorithm has been implemented in Python, with the use of the OpenCV library. For image pre-processing, the spatial anti-aliasing, filtering, and morphological opening and closing have been used. Color maps have been created to determine the mass centers of the recognized objects. To calculate the rotation angle of the manipulator's elbow based on the shoulder position, linear regression model is trained. The practical value of the research and development results is the possibility of their use in the harvesting of the agricultural objects with further extend to manufacture with conveyor sorting tapes.

Keywords: automatic harvest sorting, robotic arm, OpenCV, Arduino, servo drives, multi-agent system.

1 Introduction

As one of the key branches of the industry of Ukraine, agricultural industry aims to deliver the high-quality goods with lower self-costs, while also reducing the time spent on their production. Replacing the manual labor with the machine labor is an effective way of solving this problem. A human laborer cannot perform repetitive tasks for a long time and stay effective for long.

Since harvesting is one of the primary parts of all agricultural processes, automating this task has been a researchers' priority since the 1960s. However, automated harvesting has several complications to avoid. Accurate recognition of objects and avoiding their damage during the gathering process are the main problems for every researcher aiming to automate this procedure.

One of the first researches devoted to automatic harvest recognition and gathering was performed in 1968 [11]. The authors were comparing the lengths of the waves of light, which were reflected from the fruitage and leaves on the orange trees. When the photographic equipment became more affordable, similar researches started analyzing images of the fruit instead of the objects' surfaces.

Since the problem of the automatic agricultural objects' sorting comprises both object recognition and capturing, it encounters the same complications as the automatic harvesting task does. Nowadays, there are several approaches to the automatic sorting of the agricultural objects – magnetohydrodynamic installations [10], Bayesian classifiers [4], machine vision modules [6]. However, most of these methods don't take into consideration the physical properties of the objects, or are expensive to set up and maintain.

In this work, we propose a comparatively low-complexity and effective approach to the automatic agricultural objects' sorting.

2 Analysis of scientific sources and hardware solutions

In this section, we will consider some of the existing methods of the automatic sorting of the agricultural objects, based on the machine vision.

2.1 Analysis of the existing methods of machine vision-based agricultural objects recognition

Most of the modern researches use images, captured by CCD cameras, for object detection and recognition. Some of them rely on the fuzzy logic algorithms [2]. For object borders detection, the thresholding method was used. Afterwards, the determined objects were sorted by color (97,8% accuracy) and size (88,9% accuracy) by calculating the degree of compliance between the objects.

Bayesian classifiers [9] are also widely used for automatic agricultural objects recognition and sorting. The borders of an object were determined by the method of finding the shortest path on the graph. Then the objects were classified by ripeness, taking into consideration the average values of the R, G and B channels on the image, along with their gradients. For this method, the accuracy of sorting by ripeness is 92,2%.

Clustering methods [7] are another one popular approach to automatic sorting of the agricultural objects. After obtaining the image of an object, its projection on the X axis was made. The points with the lowest color saturation determined the borders of an object. Then the areas of a particular color are determined within each object, and the objects are clustered with a k-means clustering method. The number of clusters is determined by the user.

Neural networks [5] represent the most complicated but the most promising approach to the automatic recognition and sorting of the agricultural objects. Having separated the image dataset in 3 classes, researchers trained a three-layer neural network, which is able to sort objects on images by their brightness (color) and size.

Considering these features, the neural network assigned each object to one of the 3 classes: low-quality fruit, medium-quality fruit, and high-quality fruit. The classification accuracy was up to 80% [5].

All the methods listed above are effective but quite significant amount of time and resource consuming. In this work, we propose an alternative approach to recognition and sorting of the agricultural objects.

2.2 Analysis of the existing hand-manipulator models

There are several criteria for choosing the hand-manipulator prototype:

- a convenient gripper, that would not damage the captured objects;
- minimum 4 degrees of freedom;
- Arduino-compatibility (for reducing the cost of the project).

Wireless Robot Arm [1] is an Arduino-compatible model, but is powered by its own custom secured microcontroller. The manipulator has four degrees of freedom, and its gripper is accustomed to the harvesting tasks. But the parts of the robot arm are made of metal, which is quite an expensive option, if compared to 3D-printing or acryl.

Niryo One from Niryo is an analog of an industrial robot, constructed for learning purposes. It has six degrees of freedom and is made of 3D-printed details. Niryo One is an open-source model, and has its own IDE with a drag-and-drop editor, which makes a model available for all users. The manipulator can be powered by both Arduino and Raspberry PI.

The model also has several grippers which can be mounted into it. But the huge variety of details makes Niryo One complicated to construct and maintain.

Dexter by Haddington Dynamics [3] is a 3D-printed manipulator with 7 degrees of freedom. Powered by the HD Hi-Q CPU, it allows to run the computer vision, neural network training, and signal processing algorithms right on the manipulator. The processing unit also allows to run distributed applications on it. Needless to say, that such a powerful setting is hard to reproduce.

The uArm by UFACTORY is powered by Arduino MEGA 2650 and has four degrees of freedom. Each of the joints can rotate on 180°. The gripper has a round form and is accustomed for capturing round objects. The model is an open-source project, so the manufacturer provides the drawings of the details, which can be carved from acryl to reduce the budget of the project.

Having compared the above-mentioned manipulator models, we decided to perform the reverse engineering of uArm by UFACTORY. This hand-manipulator can lift objects up to 0.5 kg in weight, has a 320 mm working area, and is Arduino-compatible by default.

3 Results

In this work, the hand-manipulator, powered by Arduino Uno Rev3 and two types of TowerPro servomotors, is used. The object recognition algorithm implemented in

Python with the use of OpenCV library, determines the rotation angle for the manipulator's shoulder and elbow. The manipulator rotates on the appropriate angle, captures the object, and puts it either on the left or on the right, based on its color.

3.1 Hand-manipulator construction and controlling

The constructed manipulator has the following characteristics:

- height – 300 mm;
- working area – 140-300 mm;
- carrying capacity – 200 g;
- input current – 6 A;
- board – Arduino Uno Rev 3;
- connection method – MicroUSB.

Fig. 1 shows the constructed hand-manipulator.

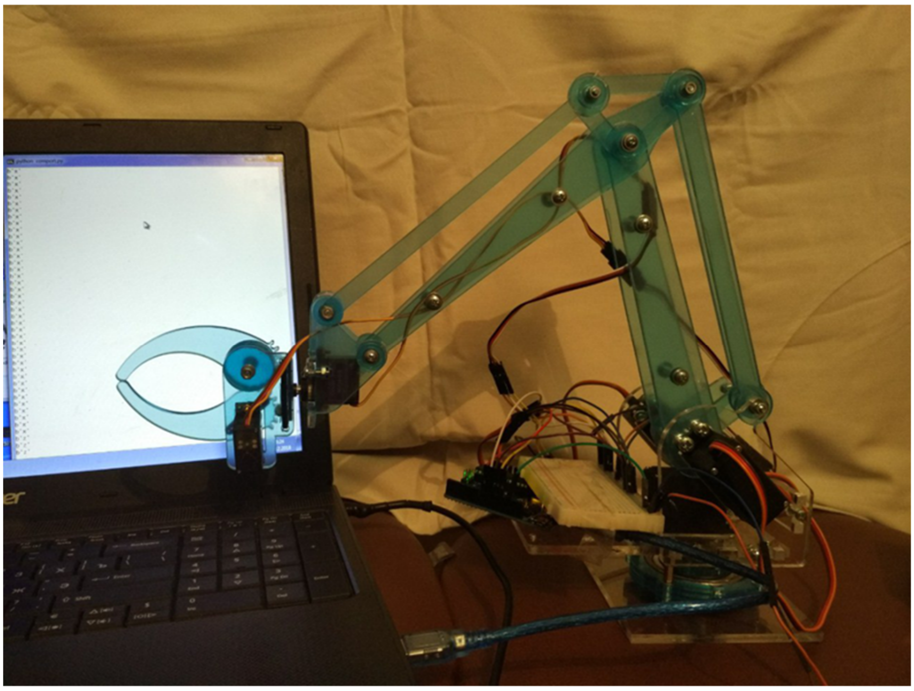


Fig. 1. The constructed hand-manipulator.

All the movable joints of the manipulator have bearings on them, which makes the moves faster and more coordinated. More powerful servomotors (TowerPro MG995) are located in the base, shoulder and elbow of the manipulator, which makes the construction more stable.

The function of Arduino Uno Rev 3 lies in getting the value of the rotation angle for the manipulator's shoulder through the COM port, and sending the respective commands to the servomotors. The Arduino is connected to a computer where the sorting algorithm runs, and is powered by it. For servomotors, the external source of power is used (5 V, 5 A). Fig. 2 shows the connection scheme of the overall complex.

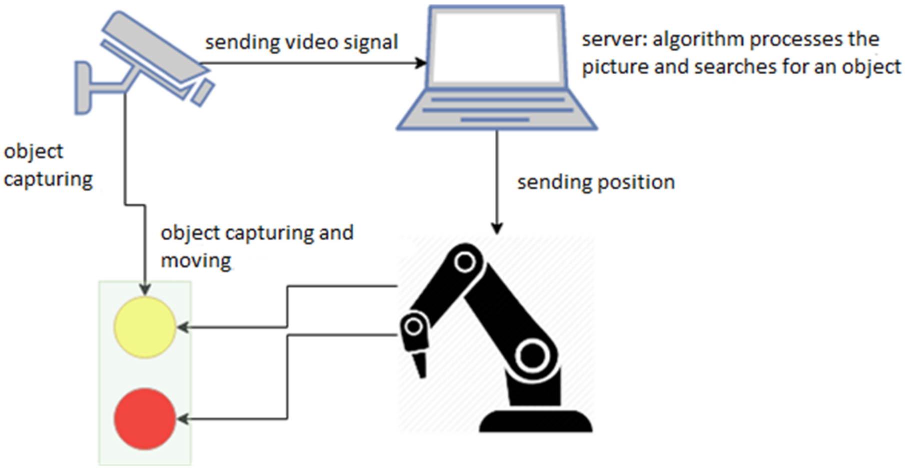

sending video signal
server: algorithm processes the picture and searches for an object
object capturing
sending position
object capturing and moving

Fig. 2. Connection scheme of the sorting complex.

Five servomotors of two types run the hand-manipulator. Three TowerPro MG995 servomotors are placed in the base, elbow, and shoulder of the manipulator, with the torque of 10 kg*cm with the 6 V power supply. The rotation speed with such a power supply is 0,16s/60º. The gripper opening-closing and rotating are performed by the TowerPro MG90S servomotors. Their torque is 2.2 kg*cm with the 6V power supply, and the rotation speed is 0,08s/60º. Both models of servomotors work with any type of the Arduino output (the PWM mode is not obligatory).

The algorithm that controls Arduino can be described by a finite Mealy machine, shown on Fig. 3.

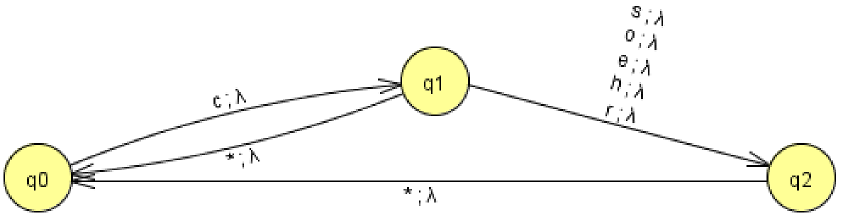


Fig. 3. The finite Mealy machine for controlling Arduino board, that controls the hand-manipulator. The states are: q0 – waiting for the command, q1 – reading the command, q2 – processing the command. The input signals are: c – read command, r – rotate the base, h – open/close the gripper, e – rotate the elbow, o – rotate the gripper, s – rotate the shoulder.

In case the server sends an inappropriate signal to Arduino, it just keeps waiting for the next command to be read.

3.2 Object Detection Algorithm

The object detection algorithm was described on Python with the use of the OpenCV library [8] – an open library for computer vision and machine learning. The algorithm consists of the 3 key parts – image preprocessing, object detection, and object capturing and moving.

The algorithm is meant to sort agricultural objects of two colors – yellow and red. For more convenient recognition of the manipulating hand itself, the green mark was put on the gripper.

First, the algorithm looks for all red objects and moves them to the left. Then, it starts looking for yellow objects and moves them to the right. When all the objects are sorted, the algorithm stops working.

Fig. 4 shows the statechart diagram for the algorithm.

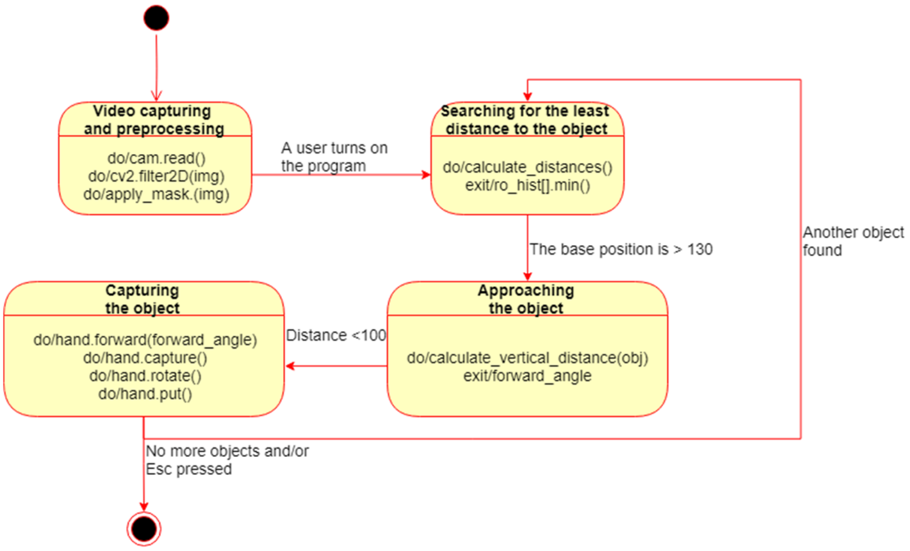
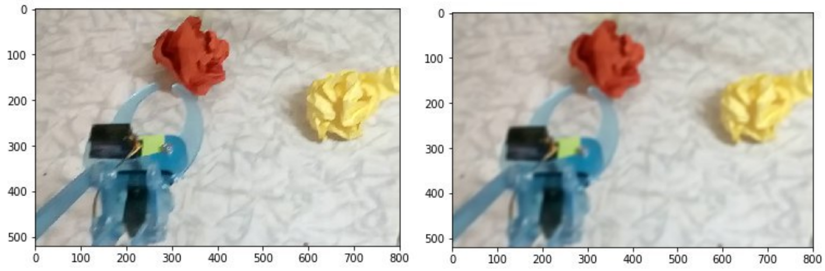


Fig. 4. The statechart diagram for the object recognition and sorting algorithm

To increase the algorithm performance, the image preprocessing starts even before a user starts working with a program. The image preprocessing aims to reduce noise on the image and consists of spatial anti-aliasing, filtering (color maps creation), and morphological opening/closing.

For the spatial anti-aliasing, the mean filtering method is used. The 9x9 filter was used, with the value of each sell being equal to 1/81. Since the input image had the BGR representation, the filter was put on the blue, green and red channels separately. Fig. 5 shows an image before and after the mean filtering.



a) b)

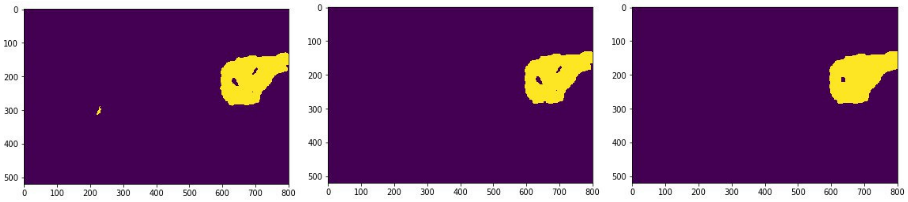
Fig. 5. The image captured by a web camera: a) before mean filtering b) after mean filtering

After applying the mean filter, the image was converted to the HSV-format. The model is a non-linear representation of the RGB-model.

The next stage of an algorithm is creating the color maps for yellow, red, and green colors by highlighting all the pixels of a particular color. The pixel's belonging to the following of HSV-values determined its inclusion on a particular color map:

- green – from (45, 50, 50) to (75, 255, 255);
- yellow – from (20, 100, 100) to (30, 255, 255);
- red – from (0, 70, 50) to (10, 255, 255) and from (170, 70, 50) to (180, 255, 255).

To every color map, morphological opening and closing were applied. Fig. 6 shows the outcome of these operations.



a) b) c)

Fig. 6. a) initial map b) morphological opening c) morphological closing

On the resulting color maps, the mass centers of the objects are determined. For each map, both horizontal and vertical projections of the colored areas are made. Afterward, the means along each axis are calculated. The expected value of the center coordinate is calculated by the formula

$$E = \int_{-\infty}^{\infty} x \cdot f(x) \,\middle|\, \{f(x) > q\} \, dx \,, \tag{1}$$

where q – 90% quantile.

To determine the mass centers of the objects more precisely, the exponential moving average method is used. The method is determined by the formula

$$c = c_{t-1} \cdot (1-\alpha) + x_c \cdot \alpha \,, \tag{2}$$

where c – anti-aliased mass center coordinate,

c_{t-1} – previous value of an anti-aliased mass center coordinate,

x_c – not anti-aliased mass center coordinate,

α – anti-aliasing coefficient.

For the coordinates of the gripper, the moving average formula is used

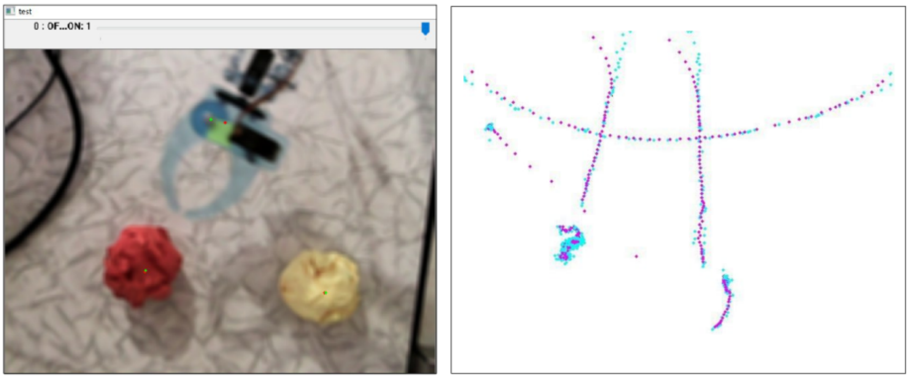
$$v = \frac{1}{n}\left(\sum_{0}^{n-1} v_{t-1}\right), \tag{3}$$

where v — anti-aliased mass center coordinate,

n – anti-aliasing interval,

v_{t-1} – the previous value of the anti-aliased mass center coordinate.

In the program interface, the mass centers before anti-aliasing are shown in red, while the mass centers after anti-aliasing are showed in green (see Fig. 7, a).



a) b)

Fig. 7. a) Mass centers of the objects and b) moving trajectories of the objects during the scanning and capturing process

To determine the least distances to the objects, the manipulator scans all the area, captured by a web camera (the base rotates from 0º to 130º). On each iteration, the rotation angle of the base changes on 2º. The mass centers are re-calculated on each iteration and the Euclidian distance between the mass centers of the gripper and the object are calculated.

When the scanning process is over, the least Euclidian distance from each object is taken from the array, along with the appropriate base rotation angle. To capture the object, the manipulator rotates on the respective angle. Fig. 7, b shows the moving trajectories of the objects during the scanning and capturing process.

During the capturing process, it is important to keep the balance between the manipulator's shoulder and elbow to keep the construction stable. For this purpose, the linear regression model was trained. To capture the object successfully, the gripper should stay approximately 2 cm high from the surface. The dataset for the linear regression model consisted of 20 combinations of the shoulder's and elbow's rotation angles, which guaranteed that the gripper stayed on the needed distance from the surface. The formula for determining the elbow position based on the shoulder rotation angle is

$$p_l = -13 + p_r \cdot 1{,}54 \tag{4}$$

When the manipulator aims to capture the object, the Euclidian distance between the objects and the gripper is calculated on each iteration. If the distance is bigger than 100, the shoulder position changes on 1º. The elbow position is calculated with the formula (4). When the distance becomes less than 100, the object is captured.

4 Testing

The complex was tested during the test session of 30 experiments, with the objects being placed on distances between 20 cm and 30 cm from the gripper. In each experiment, the yellow and red test objects were interchanged.

Testing was conducted in the same circumstances: on the white background, under the same lighting. The starting position of the hand-manipulator was fixed, along with the position of the camera. The accuracy of the object recognition is 93,4%. The accuracy of the object capturing and moving is 78,3%. Fig. 8 shows the testing results for the red objects.

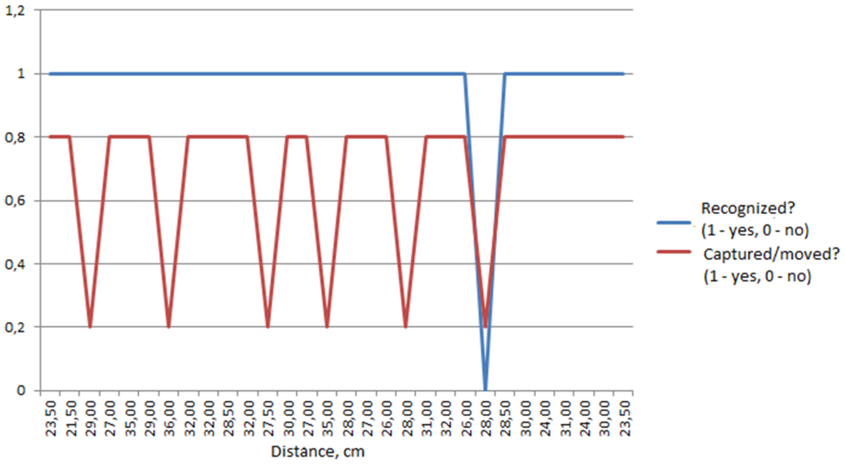


Fig. 8. Testing results for the red objects.

Testing showed that the distance between 26 cm and 30 cm between the manipulator and objects is optimal for successful sorting. The model of investigated multi-agent system (MAS) for efficient sorting based on the autonomous joints rotation is

$$\begin{cases} fp = a \cdot \cos\left(\frac{kb}{n}\right) + \frac{kc}{n}; \\ x_i = \sum_{m=1}^{i=j-1} \frac{1}{n}\cos(fp) + \sum_{m=j}^{i=j+r-1} \frac{1}{n}\cos(fp) + \sum_{m=j+r}^{i} \frac{1}{n}\cos(fp); \\ y_i = \sum_{m=1}^{i=j-1} \frac{1}{n}\sin(fp) + \sum_{m=j}^{i=j+r-1} \frac{1}{n}\sin(fp) + \sum_{m=j+r}^{i} \frac{1}{n}\sin(fp); \\ S_{MAS} = M(x_i, y_i, z = const). \end{cases} \tag{5}$$

where i – the infinitesimal segment along the manipulator, a – the angle of the tip of the manipulator's base with respect to the horizontal axes, b – is the number of periods in a length, and c – the direction of movements for S_{MAS} as handler function.

During the experimental tests an idea about multiple hand-manipulators combined as multi-agent system was released. According to the need for several hand-manipulators to be applied to the sorting process, the problem of intersections between them is become important. Every agent $A_{S_{MAS}}$ should include the minimal behavior's algorithms based on the crossing collisions index r of multiple segments of the hand-manipulators.

5 Conclusions

In the current paper, we present the hardware and software parts of the device for automatic harvest sorting. The hardware part of the complex is the hand-manipulator with four degrees of freedom, controlled by the Arduino Uno Rev 3. The software part of the complex is the algorithm, described on Python with the use of the OpenCV library. The algorithm works with images at 10 FPS and searches for mass centers of the determined objects less than 0,08 s. The accuracy of the object recognition is 93,4%. The accuracy of the object capturing and moving is 78,3%.

The significant advantage of the system is that the hardware part doesn't damage objects during the sorting process. The disadvantages of the tested device are the low speed of the hand-manipulator, and comparatively low accuracy of the objects' capturing and moving. The system can be used on the enterprises with the conveyor tapes, used for harvest sorting.

References

1. Burlachenko I., Zhuravska I., Davydenko Ye., Savinov V.: Vulnerabilities analysis and defense based on MAS method in fast dynamic wireless networks. In: Proceeding of the 4th IEEE International Symposium of Wireless Systems within the IEEE International

Conf. on Intelligent Data Acquisition and Advanced Computing Systems (IEEE IDAACS-SWS 2018), Lviv, 20-21 Sept. 2018, pp. 98–102. IEEE (2018). doi:10.1109/IDAACS-SWS.2018.8525692

2. Chang W.-H., Chen S., Lin, S.-C., Huang P.-Y., Chen Y.-Y.,: Vision based fruit sorting system using measures of fuzziness and degree of matching. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, San Antonio, 2-5 Oct. 1994, vol. 3, pp. 2600–2604. IEEE (1994). doi:10.1109/ICSMC.1994.400263
3. DEXTER – Haddington Dynamics. http://hdrobotic.com/dexter (2019). Accessed 21 Mar 2019
4. Feng G., Qixin C.: Study on color image processing based intelligent fruit sorting system. In: Fifth World Congress on Intelligent Control and Automation, Hangzhou, 15-19 June 2004, vol. 6, pp. 4802–4805. IEEE (2004). doi:10.1109/WCICA.2004.1343622
5. Krainyk Y., Razzhyvin A., Bondarenko O., Simakova I.: Internet-of-Things Device Set Configuration for Connection to Wireless Local Area Network. CEUR Workshop Proceedings **2353**, 885–896 (2019)
6. Kutsevol A., Lega Y., Havrysh O., Bagriy M.: Using machine vision systems for fruit/vegetable sorting automation. Cherkasy State Technological University scientific bulletin 4, 56–62 (2018)
7. Ohali Y.A.: Computer vision based date fruit grading system: Design and implementation. Journal of King Saud University - Computer and Information Sciences **23**(1), 29–36 (2011). doi:10.1016/j.jksuci.2010.03.003
8. OpenCV team: OpenCV. https://opencv.org/ (2019). Accessed 25 Oct 2019
9. Pla F., Sanchiz J.M., Sánchez J.S.: An integral automation of industrial fruit and vegetable sorting by machine vision. In: 8th International Conference on Emerging Technologies and Factory Automation, Antibes-Juan les Pins, 15-18 Oct. 2001, vol. 2, pp. 541–546. IEEE (2001). doi:10.1109/ETFA.2001.997731
10. Renu R., Chidanand D.V.: Internal Quality Classification of agricultural produce using Non-destructive Image Processing Technologies (soft X-ray). International Journal of Latest Trends in Engineering and Technology **2**(4), 535–543. https://www.ijltet.org/journal_details.php?id=895&j_id=2602 (2013). Accessed 25 Oct 2019
11. Schterz C.E., Brown G.K.: Basic Considerations in Mechanizing Citrus Harvest. Transactions of the ASAE **11**(3), 343–346 (1968). doi:10.13031/2013.39405

Comparative analysis of the cryptocurrency and the stock markets using the Random Matrix Theory

Vladimir N. Soloviev[1,2][000-0002-4945-202X], Symon P. Yevtushenko[2]

[1] Kryvyi Rih State Pedagogical University, 54, Gagarina Ave, Kryvyi Rih 50086, Ukraine
[2] Bohdan Khmelnytsky National University of Cherkasy, 81, Shevchenko Blvd., Cherkasy, 18031, Ukraine
{vnsoloviev2016, sivam.evt}@gmail.com

Viktor V. Batareyev

Kryvyi Rih Metallurgical Institute of the National Metallurgical Academy of Ukraine,
5, Stepana Tilhy Str., Kryvyi Rih, 50006, Ukraine
viktor_bat@ukr.net

Abstract. This article demonstrates the comparative possibility of constructing indicators of critical and crash phenomena in the volatile market of cryptocurrency and developed stock market. Then, combining the empirical cross-correlation matrix with the Random Matrix Theory, we mainly examine the statistical properties of cross-correlation coefficients, the evolution of the distribution of eigenvalues and corresponding eigenvectors in both markets using the daily returns of price time series. The result has indicated that the largest eigenvalue reflects a collective effect of the whole market, and is very sensitive to the crash phenomena. It has been shown that introduced the largest eigenvalue of the matrix of correlations can act like indicators-predictors of falls in both markets.

Keywords: stock market, cryptocurrency, Bitcoin, complex system, measures of complexity, crash, Random Matrix Theory, indicator-precursor.

1 Introduction

The instability of global financial systems with regard to normal and natural disturbances of the modern market and the presence of poorly foreseeable financial crashes indicate, first of all, the crisis of the methodology of modeling, forecasting and interpretation of modern socio-economic realities. The modern paradigm of synergetic is a complex paradigm associated with the possibility of direct numerical simulation of the processes of complex systems evolution [1; 11; 20; 19; 28].

Complex systems are systems consisting of a plurality of interacting agents possessing the ability to generate new qualities at the level of macroscopic collective behavior, the manifestation of which is the spontaneous formation of noticeable temporal, spatial, or functional structures. As simulation processes, the application of quantitative methods involves measurement procedures, where importance is given to complexity measures. I. Prigogine notes that the concepts of simplicity and complexity are relativized in the pluralism of the descriptions of languages, which also determines the plurality of approaches to the quantitative description of the complexity phenomenon [21]. Therefore, we will continue to study Prigogine's manifestations of the system complexity, using the current methods of quantitative analysis to determine the appropriate measures of complexity.

The key idea here is the hypothesis that the complexity of the system before the crashes and the actual periods of crashes must change. This should signal the corresponding degree of complexity if they are able to quantify certain patterns of a complex system. Significant advantage of the introduced measures is their dynamism, that is, the ability to monitor the change in time of the chosen measure and compare it with the corresponding dynamics of the output time series. This allowed us to compare the critical changes in the dynamics of the system, which is described by the time series, with the characteristic changes of concrete measures of complexity. It turned out that quantitative measures of complexity respond to critical changes in the dynamics of a complex system, which allows them to be used in the diagnostic process and prediction of future changes.

Cryptocurrency market is a complex, self-organized system, which in most cases can be considered either as a complex network of market agents, or as an integrated output signal of such a network – a time series, for example, prices of individual cryptocurrency. Thus the cryptocurrency prices exhibit such complex volatility characteristics as nonlinearity and uncertainty, which are difficult to forecast and any results obtained are uncertain. Therefore, cryptocurrency price prediction remains a huge challenge.

The stock market is one of the more developed economic segments of the financial market, highly capitalized and globalized with well-studied trends. Therefore, a comparative analysis of fragments of these markets is of obvious scientific and applied interest.

Unfortunately, the existing nowadays classical econometric [5; 8; 34] and modern methods of prediction of crisis phenomena based on machine learning methods [2; 3; 7; 10; 13; 14; 15; 25; 36] do not have sufficient accuracy and reliability of prediction.

Thus, lack of reliable models of prediction of time series for the time being will update the construction of at least indicators which warn against possible critical phenomena or trade changes etc. In our previous works, we constructed some indicators of crisis phenomena using the methods of nonlinear dynamics [276; 27] and the theory of complex networks [31]. Similar approaches, like the Random Matrix Theory, are developed in the framework of interdisciplinary science, called econophysics [17; 24]. This work is dedicated to the construction of such indicators – precursors based on the Random Matrix Theory.

The paper is structured as follows. Section 2 describes previous studies in these fields. Section 3 presents classification of crashes and critical events on the example of a key cryptocurrency Bitcoin during the entire period (16.07.2010 – 10.01.2019) and stock market by the example of the index S&P 500 during the entire period (17.03.1980

– 10.01.2019). In Section 4, new indicators of critical and crash phenomena are introduced using the Random Matrix Theory.

2 Analysis of previous studies

Random Matrix Theory (RMT) developed in this context the energy levels of complex nuclei, which the existing models failed to explain [9; 16; 18; 37]. Deviations from the universal predictions of RMT identify system specific, nonrandom properties of the system under consideration, providing clues about the underlying interactions.

Unlike most physical systems, where one relates correlations between subunits to basic interactions, the underlying "interactions" for the financial systems problem are not known. Here, we analyze cross correlations between financial agents (stocks, cryptocurrencies) by applying concepts and methods of RMT, developed in the context of complex quantum systems. Wherein the precise nature of the interactions between subunits are not known.

RMT has been applied extensively in studying multiple financial time series among which stock markets are central [12; 22; 23; 26; 35]. The first fundamental work in the field of modelling self-organization processes in the US stock market after the S&P 500 index using the RMT method was the study of [23]. Using extensive databases (every minute, hourly, daily), an analysis of their correlation properties is carried out. It is shown that there is a small part of the eigenvalues and eigenvectors containing important information about the structural and dynamic properties of the market. In particular, the authors of [23] found that the largest eigenvalue corresponds to an influence common to all stocks. Analysis of the remaining deviating eigenvectors shows distinct groups, whose identities correspond to conventionally identified business sectors. Finally, the authors discuss applications to the construction of portfolios of stocks that have a stable ratio of risk to return. Further studies, for example, [12; 22; 26; 35] developed the work of [23] and adapted the methodology to other financial objects.

As for the cryptocurrency market, the work here has just begun [3332; 33]. In the work [33], the classic scheme [23] was used for crypto assets with similar conclusions. The authors [32] analyzed the structure of the cryptocurrency market based on the correlation-based agglomerative hierarchical clustering and minimum spanning tree and examined the market structures. As a result, the authors demonstrated the leadership of the Bitcoin and Ethereum in the market, six homogeneous clusters composed of relatively less-traded cryptocurrencies, and transformation of the market structure after the announcement of regulations from various countries.

We will calculate the correlation properties of stock and crypto markets and compare the calculation results.

3 Data

At the moment, there are various research works on what crises and crashes are and how to classify such interruptions in the stock markets and market of cryptocurrencies.

We have created our classification of such leaps and falls, relying on Bitcoin time series during the entire period (16.07.2010 – 10.01.2019) of verifiable fixed daily values of the Bitcoin price (BTC) (https://finance.yahoo.com/cryptocurrencies). Critical US stock market events considered over time period 17.03.1980 – 10.01.2019 (https://finance.yahoo.com/quote /^GSPC?p=^GSPC).

Critical events are those falls that could go on for a long period of time, and at the same time, they were not caused by a bubble. The bubble is an increasing in the price of the cryptocurrencies that could be caused by certain speculative moments. Therefore, according to our classification of the event with number (1, 3–6, 9–11, 14, 15) are the crashes that are preceded by the bubbles, all the rest – critical events. More detailed information about crises, crashes and their classification in accordance with these definitions is given in the Table 1.

Table 1. List of Bitcoin major corrections ≥ 20% since June 2011

No	Name	Days in correction
1	07.06.2011 – 10.06.2011	4
2	15.01.2012 – 16.02.2012	33
3	15.08.2012 –18.08.2012	4
4	08.04.2013 –15.04.2013	8
5	04.12.2013 –18.12.2013	15
6	05.02.2014 – 25.02.2014	21
7	12.11.2014 – 14.01.2015	64
8	11.07.2015 – 23.08.2015	44
9	09.11.2015 – 11.11.2015	3
10	18.06.2016 – 21.06.2016	4
11	04.01.2017 – 11.01.2017	8
12	03.03.2017 – 24.03.2017	22
13	10.06.2017 – 15.07.2017	36
14	16.12.2017 – 22.12.2017	7
15	13.11.2018 – 26.11.2018	14

Accordingly, during this period in the Bitcoin market, many crashes and critical events shook it. Thus, considering them, we emphasize 15 periods on Bitcoin time series, whose falling we predict by our indicators, relying on normalized returns and volatility, where normalized returns are calculated as

$$g(t) = \ln X(t+\Delta t) - \ln X(t) \cong [X(t+\Delta t) - X(t)] / X(t), \qquad (1)$$

and volatility as

$$V_T(t) = \frac{1}{n} \sum_{t'=t}^{t+n-1} \left| g(t') \right| \qquad (2)$$

Besides, considering that $g(t)$ should be more than the ±3σ, where σ is a mean square deviation.

A similar procedure makes it possible to present a classification of crashes, crises and critical events for index S&P 500 with Table 2.

Table 2. List of S&P 500 index historical corrections ≥ 20% since October 1987

No	Name	Days in correction
1	02.10.1987 – 19.10.1987	12
2	17.07.1990 – 23.08.1990	28
3	01.10.1997 – 21.10.1997	15
4	17.08.1998 – 31.08.1998	11
5	14.08.2002 – 01.10.2002	34
6	16.10.2008 – 15.12.2008	42
7	09.08.2011 – 22.09.2011	32
8	18.08.2015 – 25.08.2015	6
9	29.12.2015 – 20.01.2016	16
10	03.12.2018 – 24.12.2018	15

Calculations were carried out within the framework of the algorithm of a moving window. For this purpose, the part of the time series (window), for which there were calculated measures of complexity, was selected, then the window was displaced along the time series in a one-day increment and the procedure repeated until all the studied series had exhausted. Further, comparing the dynamics of the actual time series and the corresponding measures of complexity, we can judge the characteristic changes in the dynamics of the behavior of complexity with changes in the time series. If this or that measure of complexity behaves in a definite way for all periods of crashes, for example, decreases or increases during the pre-crashes period, then it can serve as an indicator or precursor of such a crashes phenomenon.

Calculations of complexity measures were carried out both for the entire time series, and for a fragment of the time series localizing the crash. In the latter case, fragments of time series of the same length with fixed points of the onset of crashes or critical events were selected and the results of calculations of complexity measures were compared to verify the universality of the indicators.

In the Figure 1 output Bitcoin time series, normalized returns $g(t)$, and volatility $V_T(t)$ calculated for the window size 100 are presented.

From Figure 1 we can see that during periods of crashes and critical events normalized profitability g increases considerably in some cases beyond the limits $\pm 3\sigma$. This indicates about deviation from the normal law of distribution, the presence of the "heavy tails" in the distribution g, characteristic of abnormal phenomena in the market. At the same time volatility also grows.

We observe a similar picture for the index S&P 500 (Fig. 2). These characteristics serve as indicators of critical and collapse phenomena as they react only at the moment of the above mentioned phenomena and don't give an opportunity to identify the corresponding abnormal phenomena in advance. In contrast, the indicators described below respond to critical and crash phenomena in advance. It enables them to be used as indicators-precursors of such phenomena and in order to prevent them.

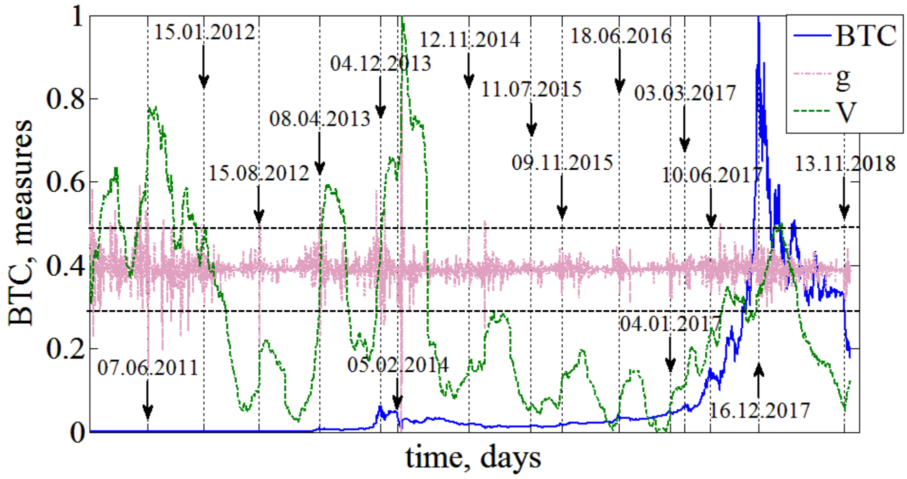


Fig. 1. The standardized dynamics, returns $g(t)$, and volatility $V_T(t)$ of BTC/USD daily values. Horizontal dotted lines indicate the ±3σ borders. The arrows indicate the beginning of one of the crashes or the critical events.

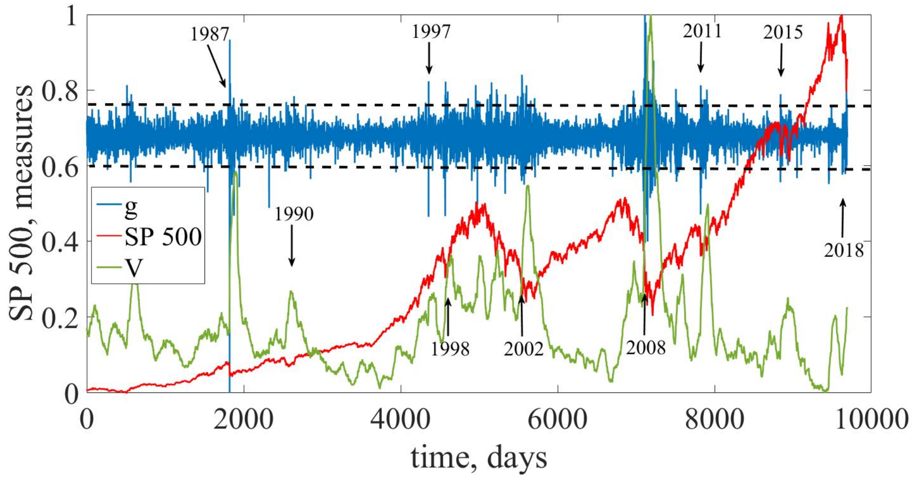
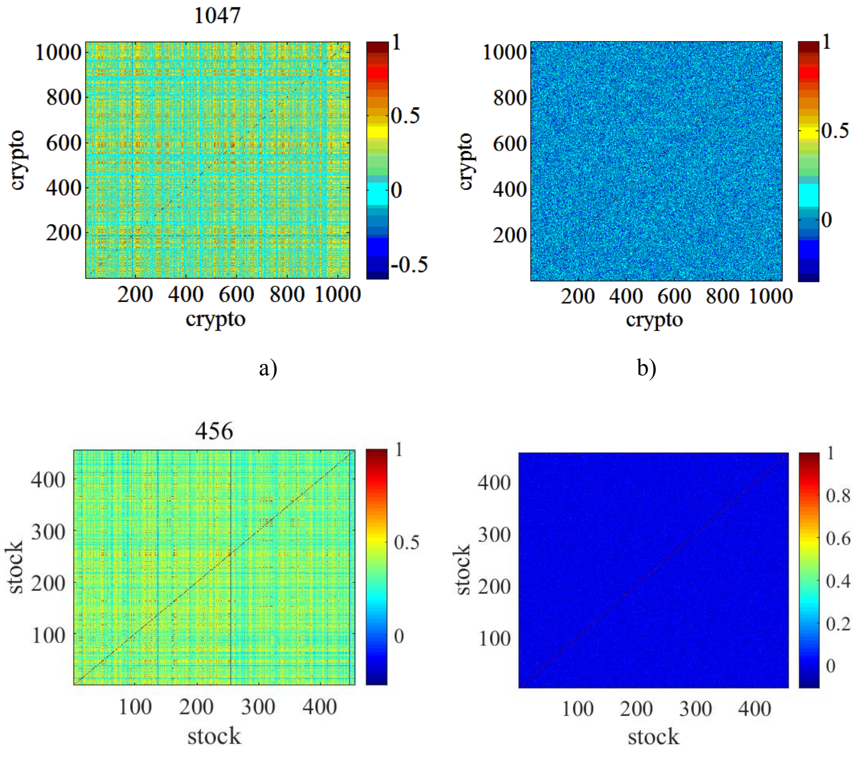


Fig. 2. The standardized dynamics, returns $g(t)$, and volatility $V_T(t)$ of S&P 500 daily values. Horizontal dotted lines indicate the ±3σ borders. The arrows indicate the beginning of one of the crashes or the critical events.

4 Random Matrix Theory

Special databases have been prepared, consisting of cryptocurrency and S&P 500 index components time series for a certain period of time. The largest number of cryptocurrencies 1047 contained a base of 456 days from 31.12.2017 to 10.01.2019, and the smallest (24 cryptocurrencies) contained a base of 1567 days, respectively, from

04.08.2013 to 10.01.2019. For the logarithmic return (1) of the i cryptocurrencies or stock price we calculate the pairwise cross-correlation coefficients between any two returns time series. For the largest databases, a graphical representation of the pair correlation field is shown in the Figure 3a, c. For comparison, a map of correlations of randomly mixed time series of the same length is shown in Figure 3b, d.



a) b)

c) d)

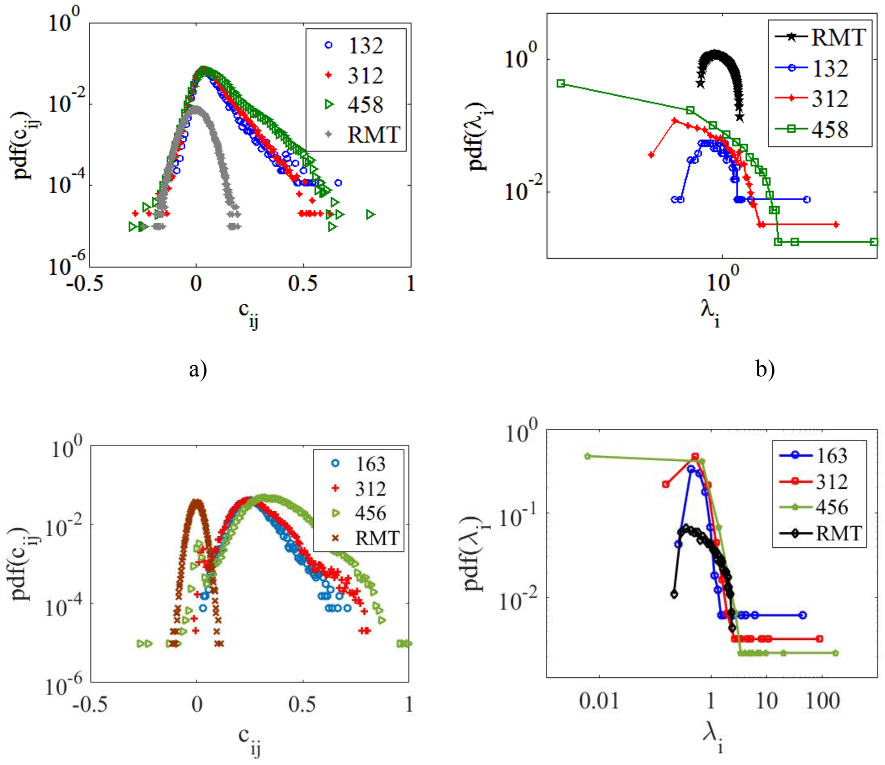
Fig. 3. Visualization of the field of correlations for the initial (a, c) and mixed (b, d) matrix cryptocurrency and S&P 500 index respectively. The largest number S&P 500 index components is 456.

For the correlation matrix C we can calculate its eigenvalues, $C = U\Lambda U^T$, where U denotes the eigenvectors, Λ is the eigenvalues of the correlation matrix, whose density $f_c(\lambda)$ is defined as follows, $f_c(\lambda) = (1/N)dn(\lambda)/d\lambda$. $n(\lambda)$ is the number of eigenvalues of C that are less than λ. In the limit $N \to \infty$, $T \to \infty$ and $Q = T/N \geq 1$ fixed, the probability density function $f_c(\lambda)$ of eigenvalues λ of the random correlation matrix M has a close form [18]:

$$f_c(\lambda) = \frac{Q}{2\pi\sigma^2} \frac{\sqrt{(\lambda_{\max} - \lambda)(\lambda - \lambda_{\min})}}{\lambda} \tag{3}$$

with $\lambda \in [\lambda_{\min}, \lambda_{\max}]$, where $\lambda_{\min}^{\max}$ is given by $\lambda_{\min}^{\max} = \sigma^2 (1 + 1/Q \pm 2\sqrt{1/Q})$ and σ^2 is equal to the variance of the elements of matrix M [18].

We compute the eigenvalues of the correlation matrix C, $\lambda_{\max} = \lambda_1 > \lambda_2 > \cdots > \lambda_{15} = \lambda_{\min}$. The probability density functions (pdf) of paired correlation coefficients c_{ij} and eigenvalues λ_i for matrices of 132, 312, 458 cryptocurrencies and 163, 312, 456 S&P 500 index components are presented in Figure 4.



a)

b)

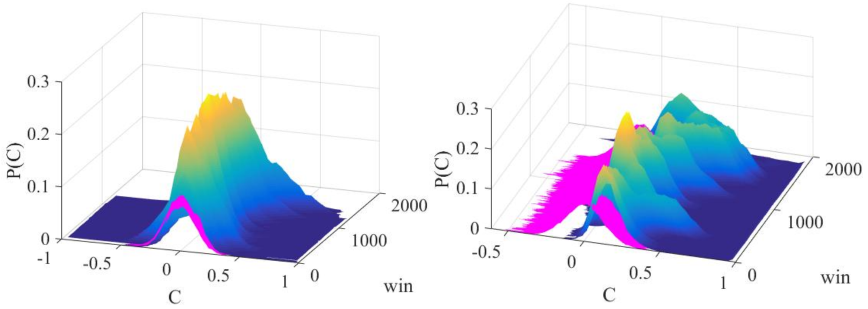
c)

d)

Fig. 4. Comparison of distributions of the pair correlation coefficients (a, c) and eigenvalues of the correlation matrix (b, d) with those for RMT for cryptocurrency market (a, b) and stock market (c, d).

Accordingly, for correlation matrices in the case of S&P 500 index, the dimensions of the matrices are as follows: 163, 312 and 456 (Fig. 4c, d). From Figures 4, it can be seen that the distribution functions for the paired correlation coefficients of the selected matrices differ significantly from the distribution function described by the RMT. It can be seen that the crypto market has a significantly correlated, self-organized system (Fig. 4a) and the difference from the RMT of the case, the correlation coefficients exceed the value of 0.6-0.8 on “thick tails”. The distribution of the eigenvalues of the correlation matrix also differs markedly from the case of RMT. In our case, only one-third of its own values refer to the RMT region. However, the stock market is even more correlated. On it, the difference with RMT data is even more obvious.

The picture of correlations changes with changing market trends. This is clearly demonstrated by Figure 5, which shows the window distribution functions of pair correlation coefficients.



a) b)

Fig. 5. Comparison of the window distributions of the pair correlation coefficients for cryptocurrencies (a) and S&P 500 index components (b).

And in this case, the stock market is more responsive to changes in market dynamics.

Eigenvectors correspond to the participation ratio PR and its inverse participation ratio IPR

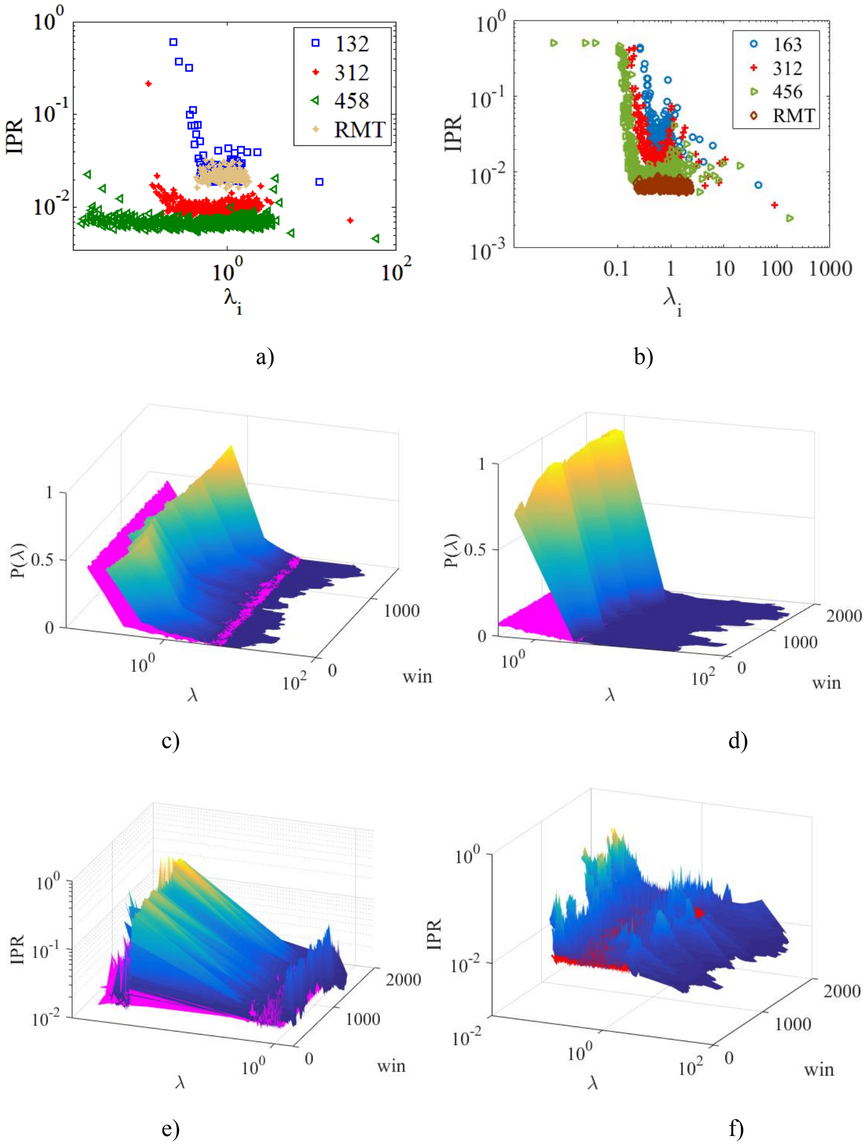
$$I^k = \sum_{l=1}^{N} [u_l^k]^4 , \tag{4}$$

where u_l^k , $l = 1, \ldots, N$ are the components of the eigenvector u^k (Fig. 6a). So PR indicates the number of eigenvector components that contribute significantly to that eigenvector. More specifically, a low IPR indicates that they contribute more equally. In contrast, a large IPR would imply that the factor is driven by the dynamics of a small number of assets. The irregularity of the influence of the eigenvalues of the correlation matrix is determined by the absorption ratio (AR), which is a cumulative risk measure

$$AR_n = \sum_{k=1}^{n} \lambda_k / \sum_{k=1}^{N} \lambda_k , \tag{5}$$

and indicates which part of the overall variation is described from the total number N of eigenvalues.

Figure 6 shows the results of IPR (a, b) calculations for both sets of matrices, as well as the results in the framework of the algorithm of a moving window, comparative calculations of the distribution function of eigenvalues (c, d) and IPR (e, f).



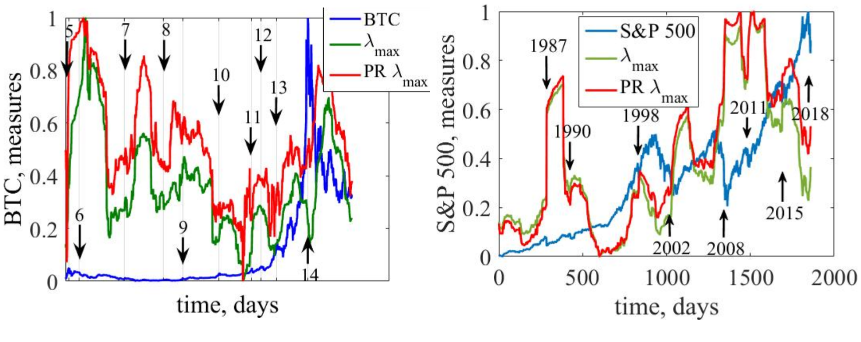
a) b)

c) d)

e) f)

Fig. 6. Inverse participation ratio (a) and moving window dynamics of the eigenvalues distribution (b), IPR for the initial and mixed (or random) matrices (c).

The difference in dynamics is due to the peculiarities of non-random correlations between the time series of individual assets. Under the framework of RMT, if the eigenvalues of the real time series differ from the prediction of RMT, there must exists hidden economic information in those deviating eigenvalues. For cryptocurrencies markets, there are several deviating eigenvalues in which the largest eigenvalue $\lambda_{\max}$ reflects a collective effect of the whole market. As for PR the differences from RMT appear at large and small λ values and are similar to the Anderson quantum effect of localization [4]. Under crashes conditions, the states at the edges of the distributions of eigenvalues are delocalized, thus identifying the beginning of the crash. This is evidenced by the results presented in Figure 7.



a) b)

Fig. 7. Measures of complexity $\lambda_{\max}$ and its participation ratio. The numerics in the figure indicate the numbers of crashes and critical events in accordance with the Tables 1, 2.

We find that both $\lambda_{\max}$ and PR $\lambda_{\max}$ have large values for periods containing the market crashes and critical events. At the same time, their growth begins in the pre-crashes periods. At the same time, the stock market is more responsive to crisis phenomena.

5 Conclusions

Consequently, in this paper, we have shown that monitoring and prediction of possible critical changes on both the stock and cryptocurrency markets is of paramount importance. As it has been shown by us, the theory of complex systems has a powerful toolkit of methods and models for creating effective indicators-precursors of crashes and critical phenomena. In this paper, we have explored the possibility of using the Random Matrix Theory measures of complexity to detect dynamical changes in a complex time series. We have shown that the measures that have been used can indeed be effectively used to detect abnormal phenomena for the used time series data.

As it has been shown by us, the econophysics has a powerful toolkit of methods and models for creating effective indicators-precursors of crisis phenomena. We have shown that the largest eigenvalue $\lambda_{\max}$ may be effectively used to detect crisis phenomena for the cryptocurrencies time series. We have concluded though by emphasizing that the most attractive features of the $\lambda_{\max}$ and PR $\lambda_{\max}$ namely its conceptual simplicity and computational efficiency make it an excellent candidate for a fast, robust, and useful screener and detector of unusual patterns in complex time series.

Thus, the results of this study confirm the main provisions of the concept of early diagnosis of crisis phenomena by calculating various measures of complexity of financial systems [6; 27; 29; 30; 31].

References

1. Albert, R., Barabási, A.-L.: Statistical Mechanics of Complex Networks. Rev. Mod. Phys. **74**, 47–97 (2002). doi:10.1103/RevModPhys.74.47
2. Albuquerque, Y., de Sá, J., Padula, A., Montenegro, M.: The best of two worlds: Forecasting high frequency volatility for cryptocurrencies and traditional currencies with Support Vector Regression. Expert Systems With Applications **97,** 177–192. (2018) https://doi.org/10.1016/j.eswa.2017.12.004
3. Alessandretti, L., ElBahrawy, A., Aiello, L.M., Baronchelli, A.: Machine Learning the Cryptocurrency Market. https://ssrn.com/abstract=3183792 (2018). doi:10.2139/ssrn.3183792. Accessed 15 Sep 2018
4. Anderson, P.W.: Absence of Diffusion in Certain Random Lattices. Phys. Rev. **109**(5), 1492–1505 (1958). doi:10.1103/PhysRev.109.1492
5. Andrews, B., Calder, M., Davis, R.A.: Maximum Likelihood Estimation for α-Stable Autoregressive Processes. The Annals of Statistics **37**(4), 1946–1982 (2009). doi:10.1214/08-AOS632
6. Bielinskyi, A., Soloviev, V., Semerikov, S., Solovieva, V.: Detecting Stock Crashes Using Levy Distribution. In: Kiv, A., Semerikov, S., Soloviev, V., Kibalnyk, L., Danylchuk, H., Matviychuk, A. (eds.) Experimental Economics and Machine Learning for Prediction of Emergent Economy Dynamics, Proceedings of the Selected Papers of the 8th International Conference on Monitoring, Modeling & Management of Emergent Economy (M3E2 2019), Odessa, Ukraine, May 22-24, 2019. CEUR Workshop Proceedings **2422**, 420–433. http://ceur-ws.org/Vol-2422/paper34.pdf (2019). Accessed 1 Aug 2019
7. Chen, T., Guestrin, C.: XGBoost: A Scalable Tree Boosting System. In: KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, August 13-17, 2016, pp. 785-794 (2016). doi:10.1145/2939672.2939785
8. Dassios, A., Li, L.: An Economic Bubble Model and Its First Passage Time. arXiv:1803.08160v1 [q-fin.MF]. https://arxiv.org/pdf/1803.08160.pdf (2018). Accessed 21 Mar 2019
9. Dyson, F.J.: Statistical Theory of the Energy Levels of Complex Systems. Journal of Mathematical Physics **3**, 140–156 (1962). doi:10.1063/1.1703773
10. Guo, T., Antulov-Fantulin, N.: An experimental study of Bitcoin fluctuation using machine learning methods. arXiv:1802.04065v2 [stat.ML]. https://arxiv.org/pdf/1802.04065.pdf (2018). Accessed 25 Oct 2018
11. Halvin, S., Cohen, R.: Complex networks. Structure, robustness and function. Cambridge University Press, New York (2010)
12. Jiang, S., Guo, J., Yang, C., Tian, L.: Random Matrix Analysis of Cross-correlation in Energy Market of Shanxi, China. International Journal of Nonlinear Science **23**(2), 96–101. http://www.internonlinearscience.org/upload/papers/IJNS%20Vol%2023%20%20No%202%20Paper%206-%20Random%20Matrix.pdf (2017). Accessed 25 Oct 2019
13. Kennis, M.A.: Multi-channel discourse as an indicator for Bitcoin price and volume movements. arXiv:1811.03146v1 [q-fin.ST]. https://arxiv.org/pdf/1811.03146.pdf (2018). Accessed 25 Oct 2019
14. Kiv, A., Semerikov, S., Soloviev, V., Kibalnyk, L., Danylchuk, H., Matviychuk, A.: Experimental Economics and Machine Learning for Prediction of Emergent Economy Dynamics. In: Kiv, A., Semerikov, S., Soloviev, V., Kibalnyk, L., Danylchuk, H., Matviychuk, A. (eds.) Experimental Economics and Machine Learning for Prediction of Emergent Economy Dynamics, Proceedings of the Selected Papers of the 8th International

Conference on Monitoring, Modeling & Management of Emergent Economy (M3E2 2019), Odessa, Ukraine, May 22-24, 2019. CEUR Workshop Proceedings **2422**, 1–4. http://ceur-ws.org/Vol-2422/paper00.pdf (2019). Accessed 1 Aug 2019

15. Kodama, O., Pichl, L., Kaizoji, T.: Regime Change and Trend Prediction for Bitcoin Time Series Data. In: Hájek P., Vít O., Bašová P., Krijt M., Paszeková H., Součková O., Mudřík R. (eds.) CBU International Proceedings 2017: Innovations in Science and Education, Prague, 22–24 March 2017, pp. 384–388. Central Bohemia University Prague (2017). doi:10.12955/cbup.v5.954
16. Laloux, L., Cizeau, P., Bouchaud, J.-P., Potters, M.: Noise dressing of financial correlation matrices. Physical Review Letters **83**(7), 1467–1470 (1999). doi:10.1103/PhysRevLett.83.1467
17. Mantegna, R.N., Stanley, H.E.: An Introduction to Econophysics: Correlations and Complexity in Finance. Cambridge University Press, Cambridge (2000)
18. Mehta, L.M.: Random Matrices, 3rd edn. Academic Press, San Diego (2004)
19. Newman, M., Barabási A.-L., Watts D.J. (eds.): The Structure and Dynamics of Networks. Princeton University Press, Princeton and Oxford (2006)
20. Newman, M.E.J.: The Structure and Function of Complex Networks. SIAM Reviews **45**(2), 167–256 (2003). doi:10.1137/S003614450342480
21. Nikolis, G., Prigogine, I.: Exploring Complexity: An Introduction. St. Martin's Press, New York (1989)
22. Pharasi, H.K., Sharma, K., Chakraborti, A., Seligman, T.H.: Complex Market Dynamics in the Light of Random Matrix Theory. In: Abergel F., Chakrabarti B., Chakraborti A., Deo N., Sharma K. (eds.) New Perspectives and Challenges in Econophysics and Sociophysics, pp. 13–34. New Economic Windows. Springer, Cham (2019). doi:10.1007/978-3-030-11364-3_2
23. Plerou, V., Gopikrishnan, P., Rosenow, B., Amaral, L.A.N., Guhr, T., Stanley, H.E.: Random matrix approach to cross correlations in financial data. Phys. Rev. E **65**, 066126 (2002)
24. Saptsin, V., Soloviev, V.: Relativistic quantum econophysics – new paradigms in complex systems modelling. arXiv:0907.1142v1 [physics.soc-ph]. https://arxiv.org/pdf/0907.1142.pdf (2009). Accessed 25 Oct 2019
25. Shah, D., Zhang, K.: Bayesian regression and Bitcoin. In: 2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, 30 Sept. – 3 Oct. 2014. IEEE (2014). doi:10.1109/ALLERTON.2014.7028484
26. Shen, J., Zheng, B.: Cross-correlation in financial dynamics. EPL (Europhysics Letters) **86**(4), 48005 (2009). doi:10.1209/0295-5075/86/48005
27. Soloviev, V., Belinskij, A.: Methods of nonlinear dynamics and the construction of cryptocurrency crisis phenomena precursors. In: Ermolayev, V., Suárez-Figueroa, M.C., Yakovyna, V., Kharchenko, V., Kobets, V., Kravtsov, H., Peschanenko, V., Prytula, Y., Nikitchenko, M., Spivakovsky, A. (eds.) Proceedings of the 14th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume II: Workshops, Kyiv, Ukraine, May 14-17, 2018. CEUR Workshop Proceedings **2014**, 116–127. http://ceur-ws.org/Vol-2104/paper_175.pdf (2018). Accessed 25 Oct 2019
28. Soloviev, V., Moiseienko, N., Tarasova, O.: Modeling of cognitive process using complexity theory methods. In: Ermolayev V., Mallet F., Yakovyna V., Kharchenko V., Kobets V., Korniłowicz A., Kravtsov H., Nikitchenko M., Semerikov S., Spivakovsky A. (eds.) Proceedings of the 15th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume II:

Workshops. Kherson, Ukraine, June 12-15, 2019. CEUR Workshop Proceedings, vol. 2393, pp. 905–918. http://ceur-ws.org/Vol-2393/paper_356.pdf (2019). Accessed 25 Oct 2019

29. Soloviev, V., Serdiuk, O., Semerikov, S., Kohut-Ferens, O.: Recurrence entropy and financial crashes. In: Horal L., Soloviev V., Matviychuk A., Khvostina I. (eds.). Proceedings of the 2019 7th International Conference on Modeling, Development and Strategic Management of Economic System (MDSMES 2019). Advances in Economics, Business and Management Research, vol. 99, pp. 385–388. Atlantis Press, Paris (2019). doi:10.2991/mdsmes-19.2019.73
30. Soloviev, V., Solovieva, V., Tuliakova, A., Ivanova, M.: Construction of crisis precursors in multiplex networks. In: Horal L., Soloviev V., Matviychuk A., Khvostina I. (eds.). Proceedings of the 2019 7th International Conference on Modeling, Development and Strategic Management of Economic System (MDSMES 2019). Advances in Economics, Business and Management Research, vol. 99, pp. 361–366. Atlantis Press, Paris (2019). doi:10.2991/mdsmes-19.2019.68
31. Soloviev, V.N., Belinskiy, A.: Complex Systems Theory and Crashes of Cryptocurrency Market. In: Ermolayev V., Suárez-Figueroa M., Yakovyna V., Mayr H., Nikitchenko M., Spivakovsky A. (eds.) Information and Communication Technologies in Education, Research, and Industrial Applications (14th International Conference, ICTERI 2018, Kyiv, Ukraine, May 14-17, 2018, Revised Selected Papers). Communications in Computer and Information Science, vol. 1007, pp. 276-297. Springer, Cham (2019). doi:10.1007/978-3-030-13929-2_14
32. Song J.Y., Chang W., Song J.W.: Cluster analysis on the structure of the cryptocurrency market via Bitcoin-Ethereum filtering. Physica A: Statistical Mechanics and its Applications **527**, 121339 (2019). doi:10.1016/j.physa.2019.121339
33. Stosic, Darco, Stosic, Dusan, Ludermir, T.B., Stosic, T.: Collective behavior of cryptocurrency price changes. Physica A: Statistical Mechanics and its Applications **507**, 499–509 (2018). doi:10.1016/j.physa.2018.05.050
34. Tarnopolski, M.: Modeling the price of Bitcoin with geometric fractional Brownian motion: a Monte Carlo approach. arXiv:1707.03746v3 [q-fin.CP]. https://arxiv.org/pdf/1707.03746.pdf (2017). Accessed 17 Aug 2019
35. Urama, T.C., Ezepue, P.O., Nnanwa, C.P.: Analysis of Cross-Correlations in Emerging Markets Using Random Matrix Theory. Journal of Mathematical Finance **7**(2), 291–307 (2017). doi:10.4236/jmf.2017.72015
36. Wang, M., Zhao, L., Du, R., Wang, C., Chen, L., Tian, L., Stanley, H.E.: A novel hybrid method of forecasting crude oil prices using complex network science and artificial intelligence algorithms. Applied Energy **220**, 480–495 (2018). doi:10.1016/j.apenergy.2018.03.148
37. Wigner, E.P.: On a class of analytic functions from the quantum theory of collisions. Annals of Mathematics Second Series **53**(1), 36–67 (1951). doi:10.2307/1969342

Convolutional neural networks for image classification

Andrii O. Tarasenko, Yuriy V. Yakimov, Vladimir N. Soloviev[000-0002-4945-202X]

Kryvyi Rih State Pedagogical University, 54, Gagarina Ave, Kryvyi Rih 50086, Ukraine
{vnsoloviev2016, urka226622, andrejtarasenko97}@gmail.com

Abstract. This paper shows the theoretical basis for the creation of convolutional neural networks for image classification and their application in practice. To achieve the goal, the main types of neural networks were considered, starting from the structure of a simple neuron to the convolutional multilayer network necessary for the solution of this problem. It shows the stages of the structure of training data, the training cycle of the network, as well as calculations of errors in recognition at the stage of training and verification. At the end of the work the results of network training, calculation of recognition error and training accuracy are presented.

Keywords: machine learning, deep learning, neural network, recognition, convolutional neural network, artificial intelligence.

1 Introduction

Today, a very old field of research, called artificial intelligence deals with the ability of machines to think like a man. Leading it companies and researchers from universities have made a breakthrough in the research of artificial intelligence and now we have software that can “see” (recognize objects in the image, capable of restoring video), make predictions based on certain data (forecasting stock market indices), make decisions (the ability to play games, sometimes better than a person).

The idea of artificial intelligence originated in the 1940s, when the question first arose whether it was possible to make a computer “think” [10]. Briefly, this sphere can be described as follows: automation of intellectual tasks, which are usually performed by people [1]. After some time, the researchers were able to create models that are capable of learning and perform tasks without a clear statement. Such models are called neural networks. Their peculiarity lies in the ability to learn without programming the networks themselves.

Artificial intelligence has occupied the place of the sphere of research and today includes the paradigm of learning – machine learning [12; 13], which differs from the initial, as it was previously called “symbolic artificial intelligence”, in that in machine learning, the programmer enters into the program data with answers that correspond to these data, and the output receives the rules (the answer), and symbolic artificial intelligence, in turn, performed the rules set by the programmer.

In recent years, much attention has been paid to deep learning, which is successfully used in classification and recognition problems. The key place here is occupied by neural networks, namely convolutional neural networks, which contain the meaning of "depth" [2].

Depth in deep learning does not mean the deeper understanding achieved by this approach, the idea is multi-layered representation. The number of layers into which the data model is divided is called the depth of the model. Other relevant names for this area of machine learning could be: layered learning or hierarchical learning.

Modern deep learning often involves tens or even hundreds of successive layers of representation – all of which are determined automatically by the training data.

2 Analysis of previous studies

Deep neural networks over the past 20 years take a very large part of the world's research and development, but the first mention of deep network algorithms appeared in the mid-1960s in the book of Aleksey G. Ivakhnenko and Valentin G. Lapa [6]. Then, the concept of "deep learning" in the scientific community emerged through the work of Rina Dechter in 1986 [3].

The first models of convolutional neural networks were called "neocognitron" and were discovered in 1980 by Kunihiko Fukushima [4]. Fukushima proposed several algorithms for supervised and unsupervised learning, and the neocognitron itself was a multilayered deep structure.

Taking into account the fact that the recognition of medical images has recently attracted the attention of researchers, there are several difficulties in the study of this area, namely: 1) a small number of training copies, 2) the difference in scale and fuzzy boundaries of images. These disadvantages were taken into account when creating a network model that offers a full-scale convolutional layer extracting patterns of different receptive fields with a common set of convolutional nuclei, so that scale-invariant patterns are captured by this compact set of nuclei [13].

Last decade convolutional networks are gaining a lot of attention of researchers and developers. ImageNet is one of the largest competitions dedicated to artificial intelligence and computer vision. Among the varieties of artificial networks, the prizes were taken using a convolutional network structure, such as AlexNet [7], VGG [14], GoogleNet [15] and ResNet [5]. These networks do an excellent job and have a large percentage of recognition of more than 90%.

New developments in the recognition of biomedical images Shuchao Pang, Anan Du, Mehmet A. Orgun and Zhezhou Yu [9]. This new neural network, which is called "fused" convolutional neural network (FCNN) has a precise and highly efficient classifier, which combines the features of small balls and features of deep layers.

3 Basic information

Neurons transmit electrical impulses. When transmitting a pulse (through an axon), the signal can be amplified or attenuated to be transmitted to a trace neuron (through dendrites). Such basic functions are implemented in the model of artificial neural networks:

1. Data. As signals, the artificial network uses input data (photos, audio files), which are reduced to a certain form, which the network is able to read;
2. Weights are the parameters of each layer of neurons. Act as a force signals by analogy with natural neurons. If the natural neurons the strength of the impulses, the artificial network is a numeric value;
3. Next comes the input function. In this part, the data and numerical values of the weights are processed according to the type of convolution layer;
4. This is followed by the activation function. The function that processes the input data, its value, is the output of the neuron. Then the data are transferred through the scales to the next layer of neurons (Fig. 1).

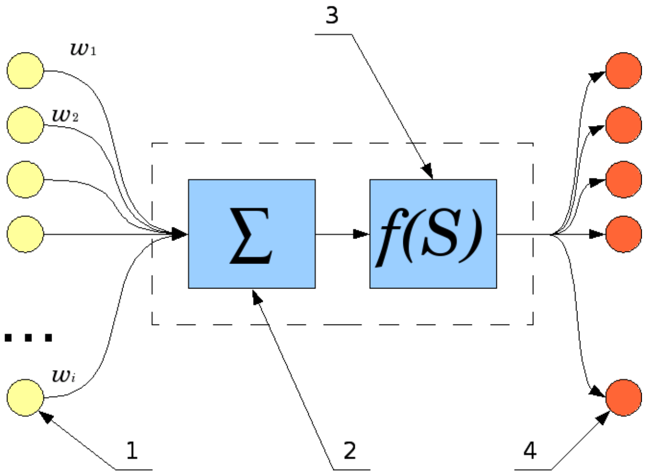


Fig. 1. Representation of artificial neuron [20].

As mentioned earlier, an artificial neural network consists of a large number of artificial neurons, which are combined into layers forming a network of neurons. There are the following types of layers of neurons:

- Input layer. Here the network receives input data that is converted in advance to the desired format (lists, arrays);
- Output layer. A layer that processes the incoming data in the outgoing (arrays of numbers) that satisfy the task;
- Convolutional layers. Models of deep neural networks use convolutional layers, which perform various kinds of "learning" operations of input data transferring to the output layer.

The simplest models of neural networks do not use convolutional layers, they are also called "single-layer" (Fig. 2). Input data described above – (x_1, ..., x_n), relevant learning factors – (w_{nm}), activation function – ($\sum$), and actually the output data – (y_1, ..., y_m).

Deep learning uses convolutional neural networks, which can contain many convolutional layers of neurons. The task of such layers is to process the input data in such a way that the output receives data that satisfies the problem condition. As with a single-layer network, these are lists of numbers.

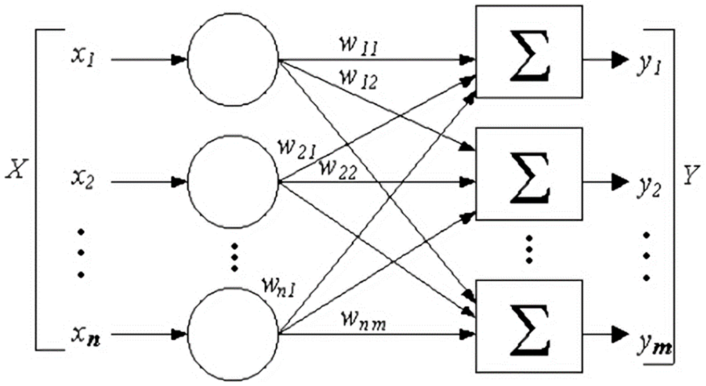


Fig. 2. Model of single-layer neural network.

Deep learning uses the concept of a container tensor for data. In fact, the tensor is a matrix of numbers, or one number is also called a tensor, or else – a scalar.

Tensors are characterized by the following key characteristics:

- Number of axes (rank, dimension). For example, a matrix is a two-dimensional tensor;
- Form. A list of integers describing the number of dimensions on each axis of the tensor;
- Data type. The type of data belonging to the tensor; for example: float32, float64, uint8, etc.

It should also be noted that, although the data is stored in the tensor as a mass, the neural network does not process the entire data set at once, but divides it into so-called "packets" of data. That is, the data package represents several separate samples with their other characteristics.

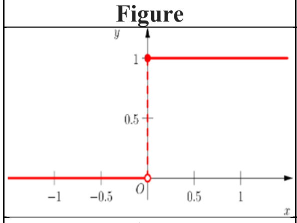
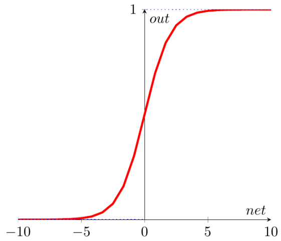
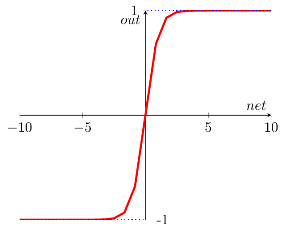
There are the following main categories of data:

- Vector-two-dimensional tensors with shape (samples, features);
- Time series-three-dimensional tensors with shape (samples, time labels, signs);
- Image-four-dimensional tensors with shape (samples, height, width, color);
- Video-five-dimensional tensors with shape (samples, frames, height, width, color).

Also, returning to the structure of the neuron, it is necessary to mention the function of action. It has been said that the function takes some value as a parameter. This value is the data that has gone through operations on the tensors (each convolutional layer performs its own operations), after which the data becomes the source of this convolutional layer.

Depending on the complexity of the task set for the neural network, activation functions can be different (table 1):

Table 1. Some of the graphs and descriptions of some activation functions.

Figure	Function name	Description
	Unit function $f(x) = \begin{cases}1, x \le b \\ 0, x \ge b\end{cases}$, b – adder output.	As you can see in the graph, the function has two states 0 or 1. It is usually used in tasks where it is enough to give the answer “Yes” or “No”
	Logistic function $f(x) = \frac{1}{1+\exp(-a*x)}$, a – the coefficient that characterizes the curvature of the graph	The most commonly used function is due to the fact that among the two States 0 and 1 there are many others, for example 0.234532, or 0.7. This makes it possible to get from the neural network not only one answer, but, for example, 10 or 1000
	The hyperbolic tangent $f(x) = tan\frac{x}{a}$	It is used for a more realistic model of a neural network, which can give the initial values not only positive, but also negative, that is, from -1 to 1

The following are the steps that are performed in the training cycle:

- The neural network receives a data packet with training instances and corresponding data for verification (must be different);
- The network performs data processing (this step is called a direct pass) and receives a packet of predictions;
- An estimate of the discrepancy between the network prediction and the validation data is calculated, that is, there must be a function to estimate this discrepancy;
- The parameters are adjusted to reduce discrepancies on this data packet.

The learning cycle is repeated as many times as the problem condition requires. After completing the training, we will get a network that has a low grade of disagreement.

Correction parameters occurs in the calculation of the gradient differences of network parameters. In this case, an offset parameter is added to the training parameters, which is the opposite of the network variance gradient.

Then, the training cycle will look like this:

1. The neural network receives a data packet with training instances and corresponding data for verification (must be different);
2. The network performs data processing (this step is called a direct pass) and receives a packet of predictions;

3. An estimate of the discrepancy between the network prediction and the validation data is calculated, that is, there must be a function to estimate this discrepancy;
4. The variance gradient for the network parameters (reverse) is calculated);
5. The parameters are adjusted by a small value in the direction opposite to the gradient to reduce discrepancies on this data packet.

To achieve the result, apply the gradient descent method to the gradient of differences on the selected data package. Before the training cycle, the point of calculating the loss gradient on a certain data package is entered, after which the global minimum of this function is calculated by the gradient descent method.

Fig. 3 images of the gradient descent operation on the function are presented:

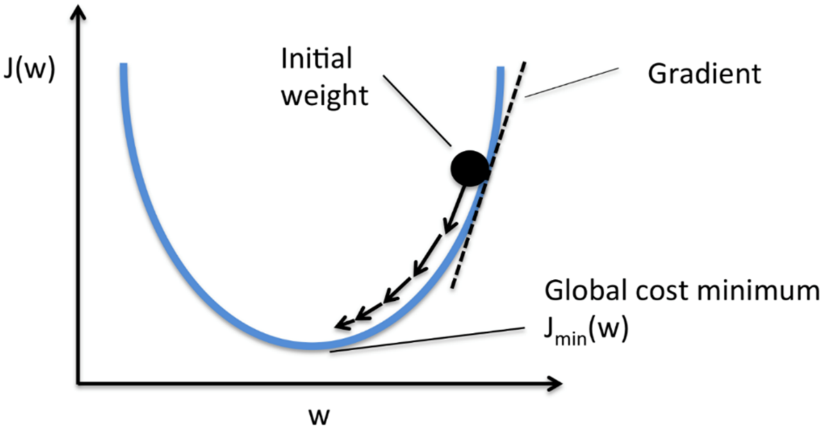


Fig. 3. Gradient descent.

The architecture of neural networks is determined by the tasks for which neural networks are designed.

The task of this work is the recognition of objects in images. Therefore, it is necessary to use the appropriate network architecture, which highlights the features in the images and forms a representation of them about the object in the photo.

We review the principle of convolutional neural networks for image distribution.

First, you need to determine the type of architecture. As described earlier, convolutional neural networks in most cases have a sequential architecture, where the neural network receives certain data at the input (tensors), after which the data is sequentially processed by the source layer-sequential architecture.

Convolutional layers are used for recognition, in which the features of each object are selected. For example, the image falls on the input layer, then the neural network tries to select one common image (usually in simple networks to recognize one object). Selection of an object is performed by means of so-called layer filters. The filter is a small window relative to the image that reads a certain area of the image. Usually the window size is 3x3 pixels, or 5x5 pixels, with different image sizes. In order to determine the image values, such a filter must pass through the image. Usually the filter starts to recognize from the upper left corner of the image and moving 1 pixel to the side, and then to the bottom passes through the entire image. At the initial stages of research, it is advisable to use a 28x28 pixel image (Fig. 4). Because large images require more time passage, and therefore more time to learn.

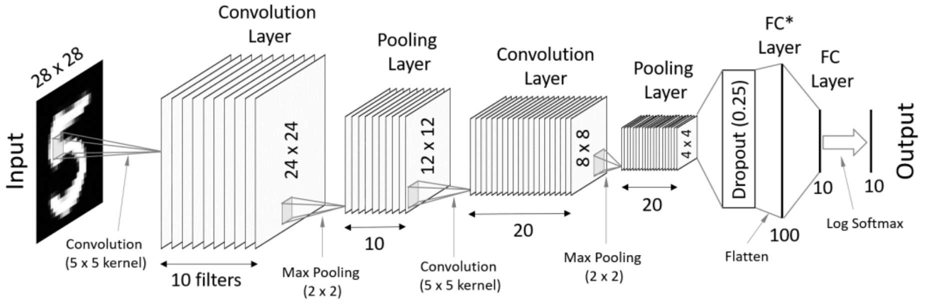


Fig. 4. Convolutional neural network architecture [16].

So, at the first convolutional stage, the neural network uses filters to determine the main object in the image.

The next step in network training is to highlight the spatial hierarchy of features. That is, having identified in the picture, for example, a cat, the network must divide the cat into parts of objects, which also must "remember" (Fig. 5). After training, the network will form a representation of such objects about the object as a whole, that is, about a cat, or other object of recognition.

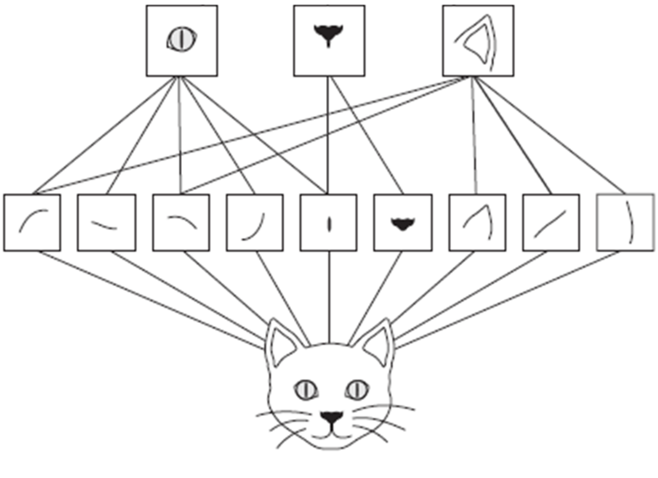


Fig. 5. Define the spatial hierarchy of features.

So, we have determined that using convolutional layers of filters with a certain size, to determine the primary features of the object in the image.

However, with the help of convolutional layers alone, it is impossible to achieve recognition of the spatial hierarchy of objects. To achieve this goal, you should perform certain operations on the image. One of these operations, which is most often used in practice – “selection of the largest of the neighboring” (Max Pooling).

The reason for using Max Pooling is that the neural network must determine the spatial hierarchy of objects, and for this we need to reduce the image by 2 times.

The Max Pooling operation is similar to the convolution operation. Take the window, now 2x2 pixels and perform one simple action with the taken four elements-Tami-choose the largest number. Thus, the initial data is filled only with the largest numbers for each window and the resulting image is reduced by half.

The third step is to define a data set for neural network training.

Since classification tasks are primarily supervised learning (teacher-assisted learning), then we will use a special data package that contains training data and feature class labels that refer to the recognized data.

The sigmoid activation function is suitable for classification problems, but in our case we will use the ReLU activation function (Fig. 6). The advantages of using such a function are the sparsity of the activity, that is, the involvement of only a part of neurons, which will accordingly reduce the load in the calculation.

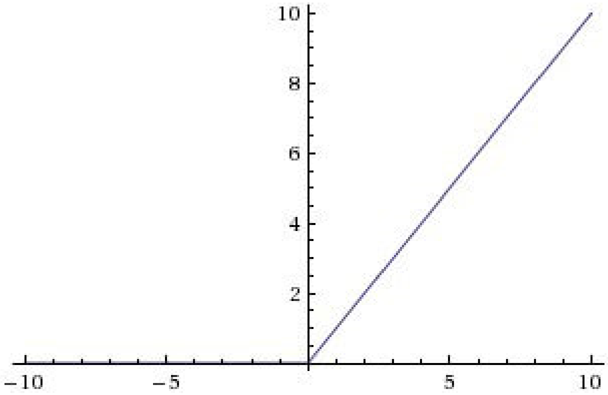


Fig. 6. Graph of ReLU function [1].

Although the use of this activation function makes part of the network passive (some neurons are not activated), it produces good results on testing [2].

$$f(x) = \max(0, x)$$

To train the network, the change of the corresponding weights for each neuron is used, namely, they are changed with the help of the optimizer. The optimizer uses gradient descent in this case.

4 Development tool

An important stage in the development of the program is the choice of development tools, because it affects the complexity and quality of the result. In fact, any programming language has tools for the development of neural networks, the difference is only in the complexity of their implementation in the chosen language. There are even many implementations of neural networks in the web programming language. Today Python is quite a popular language for researchers. After all, it is quite easy to understand and has a large community of developers. Python has many libraries for research and data visualization, so it is advisable to choose Python as the language of implementation of your own software tool.

To control the versions of the libraries and the development environment, it is advisable to choose the Anaconda tool. The choice of Anaconda lies in the convenience of version control of programming environments and libraries. Anaconda has a base of libraries for Python, which are quite convenient to install, and supports several programming environments, one of which is Jupyter Notebook, which was originally developed for the convenience of research, but eventually gained popularity among the community of developers. It will also be used as a programming environment.

To achieve this goal in the beginning, we will use a lot of additional libraries, in particular should be allocated Keras and TensorFlow.

Keras is an open source deep learning library written in the Python programming language. The library was created to improve research and design of neural networks. The library has a lot of implementations of convolutional layers, optimizers, activation functions, work with images and text.

TensorFlow is an open-source software library for machine learning developed by Google to solve the problems of building and training a neural network in order to automatically find and classify objects, achieving the quality of human perception. It is used both for research and for the development of Google’s own products. The main API for working with the library is implemented for Python.

NumPy is an open-source module for python that provides General mathematical and numerical operations in the form of pre-compiled, fast functions. They are combined into high-level packages. They provide functionality that can be compared to that of MatLab. NumPy (Numeric Python) provides the base methods for handling the large arrays and matrices. SciPy (Scientific Python), which we will also use, extends numpy functionality with a huge collection of useful algorithms such as minimization, Fourier transform, regression, and other butt-end mathematical techniques.

Matplotlib is a Python programming language library for visualization of two-dimensional (2D) graphics data (3D graphics is also supported). The resulting images can be used as illustrations in the publication.

OpenCV is a computer vision and machine learning library. We use the OpenCV library to take a picture from a webcam.

PIL (python image library) – a set of tools for working with images in Python.

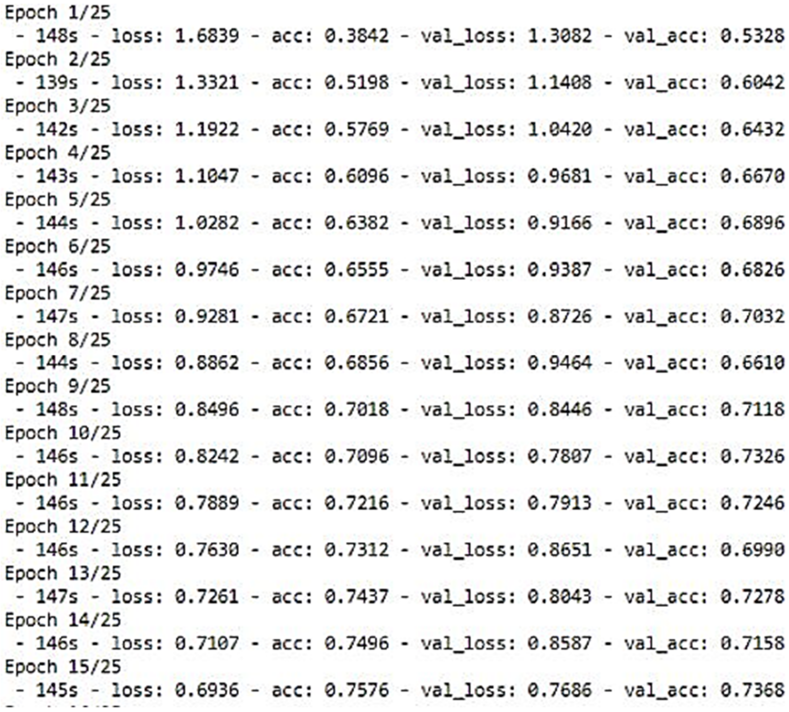
Dlib is an open source machine learning library. Dlib contains in its database a pre-trained neural network that recognizes the descriptors of the person's face from the photo, use in future work.

Tkinter is a cross-platform library for building window interfaces, included in the standard set of Python modules.

We used as a data set for training and testing x-ray images of fractures of human body parts. The images are taken from such datasets:

- MURA;
- MedPix;
- OmniMedicalSearch.

The learning process is shown in Fig. 7. In the picture, each iteration of the learning (epoch) is accompanied by calculations of errors and accuracy on the training data set and on the test data set. The data to be checked contain images that are not present in the training set, so this accuracy should be guided. The number of epochs 25 for testing was randomly selected. However, analyzing the accuracy of recognition with each epoch, it becomes clear that at first the neural network increases the percentage of accuracy, and then there are fluctuations by several percent. Starting from the 8th epoch, the accuracy falls, then increases, then repeats until the end. This is a clear sign of retraining the network.



```
Epoch 1/25
 - 148s - loss: 1.6839 - acc: 0.3842 - val_loss: 1.3082 - val_acc: 0.5328
Epoch 2/25
 - 139s - loss: 1.3321 - acc: 0.5198 - val_loss: 1.1408 - val_acc: 0.6042
Epoch 3/25
 - 142s - loss: 1.1922 - acc: 0.5769 - val_loss: 1.0420 - val_acc: 0.6432
Epoch 4/25
 - 143s - loss: 1.1047 - acc: 0.6096 - val_loss: 0.9681 - val_acc: 0.6670
Epoch 5/25
 - 144s - loss: 1.0282 - acc: 0.6382 - val_loss: 0.9166 - val_acc: 0.6896
Epoch 6/25
 - 146s - loss: 0.9746 - acc: 0.6555 - val_loss: 0.9387 - val_acc: 0.6826
Epoch 7/25
 - 147s - loss: 0.9281 - acc: 0.6721 - val_loss: 0.8726 - val_acc: 0.7032
Epoch 8/25
 - 144s - loss: 0.8862 - acc: 0.6856 - val_loss: 0.9464 - val_acc: 0.6610
Epoch 9/25
 - 148s - loss: 0.8496 - acc: 0.7018 - val_loss: 0.8446 - val_acc: 0.7118
Epoch 10/25
 - 146s - loss: 0.8242 - acc: 0.7096 - val_loss: 0.7807 - val_acc: 0.7326
Epoch 11/25
 - 146s - loss: 0.7889 - acc: 0.7216 - val_loss: 0.7913 - val_acc: 0.7246
Epoch 12/25
 - 146s - loss: 0.7630 - acc: 0.7312 - val_loss: 0.8651 - val_acc: 0.6990
Epoch 13/25
 - 147s - loss: 0.7261 - acc: 0.7437 - val_loss: 0.8043 - val_acc: 0.7278
Epoch 14/25
 - 146s - loss: 0.7107 - acc: 0.7496 - val_loss: 0.8587 - val_acc: 0.7158
Epoch 15/25
 - 145s - loss: 0.6936 - acc: 0.7576 - val_loss: 0.7686 - val_acc: 0.7368
```

Fig. 7. Plotted accuracy on the data for verification.

The plot at fig. 8 shows that the accuracy increases to the tenth epoch, then decreases, then increases. The number of learning epochs should be reduced.

Fig. 9 shows the errors on the same data. The smallest error is recorded on the tenth epoch.

During training, we obtain certain data after the completion of each era of training: loss – error on training data; acc – accuracy on training data; val_loss – error data for testing; val_acc – precision on the data for verification.

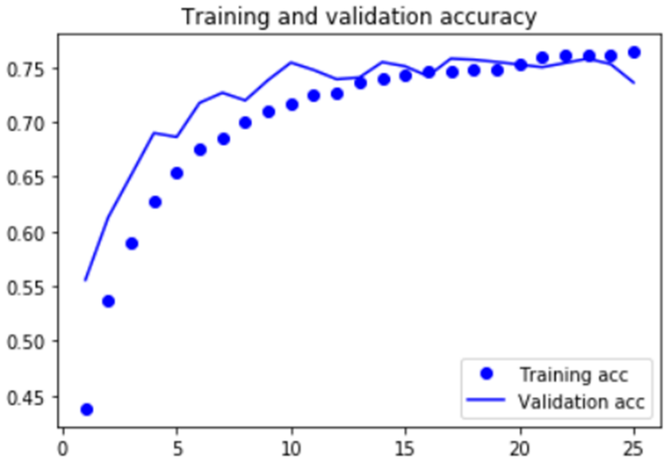


Fig. 8. Accuracy on data for training and verification.

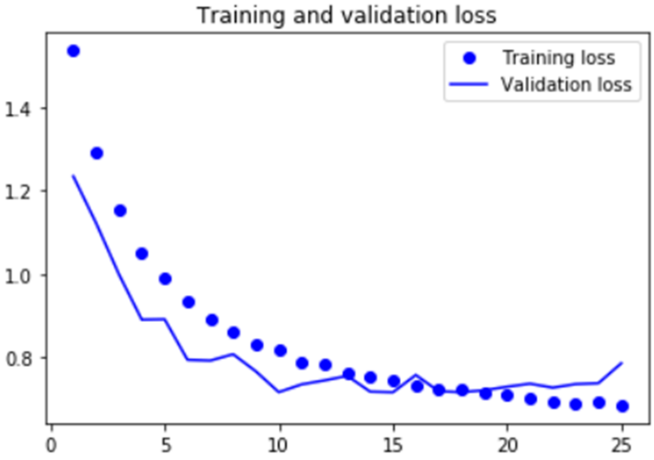


Fig. 9. Chart errors.

As a result of the training, we received an accuracy of 73.13% on the test data.

We use of Convolutional neural network in the recognition of medical images. The neural network is trained on a set of data from broken and whole human bones (Fig. 10). The neural network is able to identify obvious bone fractures on X-rays (Fig. 11).

5 Conclusion

Consequently, in this the important bases of structure and a structure of convolution neural networks and a cycle about the theory of neural networks were shown. As it was shown in the end, the result of training of neural networks allows, without their explicit programming, “teach” the program to recognize objects in the images.

This paper provides examples of databases, both existing and self-assembled, on which the neural network learns to distinguish objects. The assessment of accuracy and errors at the stages of training and verification was also carried out.

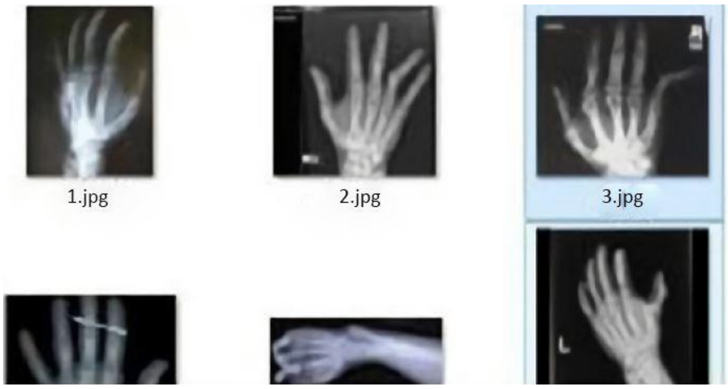
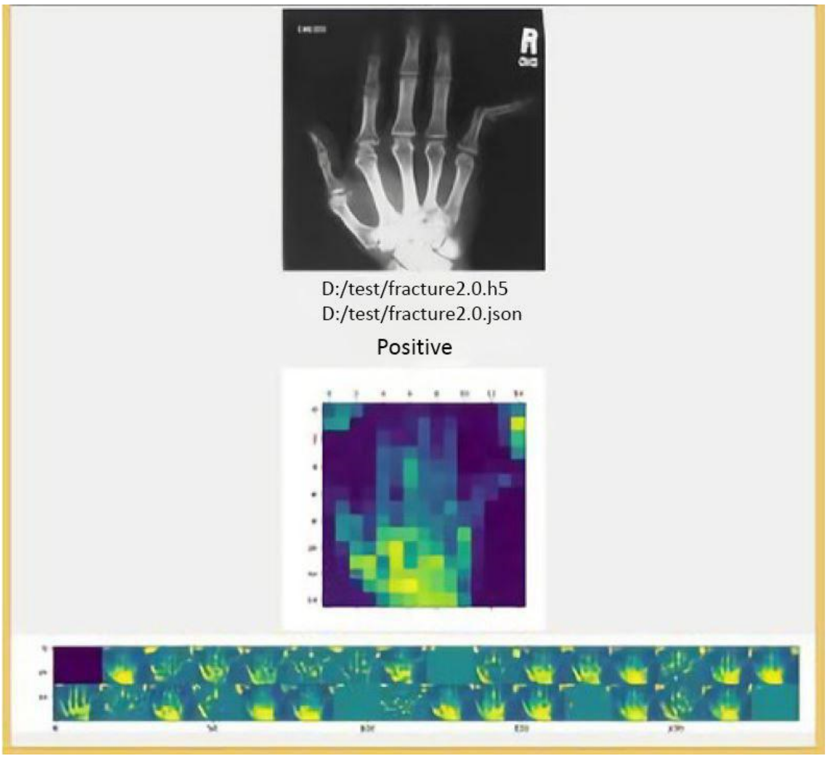


Fig. 10. Selection of objects.



D:/test/fracture2.0.h5
D:/test/fracture2.0.json
Positive

Fig. 11. Result of recognition of a fracture on a finger of the person.

References

1. Brownlee, J.: A Gentle Introduction to the Rectified Linear Unit (ReLU). Machine Learning Mastery. https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/ (2019). Accessed 25 Oct 2019
2. Brownlee, J.: A Tour of Machine Learning Algorithms. Machine Learning Mastery. https://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/ (2019). Accessed 25 Oct 2019
1. Chollet, F.: Deep Learning with Python. Manning, Shelter Island (2017)
2. Courville, A., Goodfellow, I., Bengio, Y.: Deep Learning. MIT Press, Cambridge (2016)
3. Dechter, R.: Learning while searching in constraint-satisfaction-problems. In: AAAI-86 Proceedings The Fifth National Conference on Artificial Intelligence, August 11–15, 1986, in Philadelphia, Pennsylvania., pp. 178–183. https://aaai.org/Papers/AAAI/1986/AAAI86-029.pdf (1986). Accessed 17 Aug 2019
4. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics **36**, 193–202 (1980). doi:10.1007/BF00344251
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 27–30 June 2016, pp. 770–778. IEEE (2015). doi:10.1109/CVPR.2016.90
6. Ivakhnenko, A.G., Lapa, V.G.: Cybernetics and forecasting techniques. American Elsevier Publ. Co., New York (1967)
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Communications of the ACM **60**(6), 84–90 (2017). doi:10.1145/3065386
8. Laves, M.H., Ihler, S., Ortmaier, T.: Deformable Medical Image Registration Using a Randomly-Initialized CNN as Regularization Prior. In: Medical Imaging with Deep Learning 2019. https://openreview.net/pdf?id=S1ehZFQ15E (2019). Accessed 25 Oct 2019
9. Pang, S., Du, A., Orgun, M.A., Yu, Z.: A novel fused convolutional neural network for biomedical image classification. Medical & Biological Engineering & Computing **57**, 107–121 (2019). doi:10.1007/s11517-018-1819-y
10. Semerikov, S.O., Teplytskyi, I.O., Yechkalo, Yu.V., Kiv, A.E.: Computer Simulation of Neural Networks Using Spreadsheets: The Dawn of the Age of Camelot. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 122–147. http://ceur-ws.org/Vol-2257/paper14.pdf (2018). Accessed 30 Nov 2018
11. Semerikov, S.O., Teplytskyi, I.O., Yechkalo, Yu.V., Kiv, A.E.: Computer Simulation of Neural Networks Using Spreadsheets: The Dawn of the Age of Camelot. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 122–147. http://ceur-ws.org/Vol-2257/paper14.pdf (2018). Accessed 30 Nov 2018
12. Semerikov, S.O., Teplytskyi, I.O.: Metodyka uvedennia osnov Machine learning u shkilnomu kursi informatyky (Methods of introducing the basics of Machine learning in the school course of informatics). In: Problems of informatization of the educational process in institutions of general secondary and higher education, Ukrainian scientific and practical conference, Kyiv, October 09, 2018, pp. 18–20. Vyd-vo NPU imeni M. P. Drahomanova, Kyiv (2018)

13. Semerikov, S.O.: Zastosuvannia metodiv mashynnoho navchannia u navchanni modeliuvannia maibutnikh uchyteliv khimii (The use of machine learning methods in teaching modeling future chemistry teachers). In: Starova, T.V. (ed.) Technologies of teaching chemistry at school and university, Ukrainian Scientific and Practical Internet Conference, Kryvyi Rih, November 2018, pp. 10–19. KDPU, Kryvyi Rih (2018)
14. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. In: International Conference on Learning Representations. https://www.robots.ox.ac.uk/~vgg/publications/2015/Simonyan15/simonyan15.pdf (2015). Accessed 25 Oct 2019
15. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going Deeper with Convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, 7–12 July 2015, pp. 1–9. IEEE (2015). doi:10.1109/CVPR.2015.7298594
16. Theart, R.: Getting started with PyTorch for Deep Learning (Part 3: Neural Network basics). Code to Light. https://codetolight.wordpress.com/2017/11/29/getting-started-with-pytorch-for-deep-learning-part-3-neural-network-basics/ (2017). Accessed 25 Oct 2019

Credit scoring model for microfinance organizations

Svitlana O. Yaroshchuk, Nonna N. Shapovalova[0000-0001-9146-1205],
Andrii M. Striuk[0000-0001-9240-1976], Olena H. Rybalchenko[0000-0001-8691-5401],
Iryna O. Dotsenko[0000-0001-7912-2497] and Svitlana V. Bilashenko[0000-0002-4331-7425]

Kryvyi Rih National University, 11, Vitalii Matusevych Str., Kryvyi Rih, 50027, Ukraine
yaroschucksvetlana@gmail.com, shapovalovann09@gmail.com,
andrey.n.stryuk@gmail.com, ellinaryb@gmail.com,
irado441@gmail.com, SvitlanaViktorivnaBilashenko@gmail.com

Abstract. The purpose of the work is the development and application of models for scoring assessment of microfinance institution borrowers. This model allows to increase the efficiency of work in the field of credit. The object of research is lending. The subject of the study is a direct scoring model for improving the quality of lending using machine learning methods. The objective of the study: to determine the criteria for choosing a solvent borrower, to develop a model for an early assessment, to create software based on neural networks to determine the probability of a loan default risk. Used research methods such as analysis of the literature on banking scoring; artificial intelligence methods for scoring; modeling of scoring estimation algorithm using neural networks, empirical method for determining the optimal parameters of the training model; method of object-oriented design and programming. The result of the work is a neural network scoring model with high accuracy of calculations, an implemented system of automatic customer lending.

Keywords: neural network, machine learning, lending, scoring.

1 Introduction

Increasing the profitability of credit operations is directly related to the quality of credit risk assessment [1]. In recent years, there has been a rapid increase in retail lending. Competition is increasing, the range of services is expanding, the process of obtaining a loan is being simplified, and the decision-making time is significantly reduced. The quality and speed with which a credit request is generated, as well as the reliability and simplicity of this process are crucial factors in a complex competitive process.

An important component of the bank's stable development under conditions of volatile financial position is the compliance of the risk management system with modern standards of quality of management, as well as the degree of protection against unpredictable external influences. Thus, banking organizations need to introduce scoring systems that allow to resolve qualitatively emerging issues.

Scoring is a way of quickly evaluating a potential customer of a bank or microfinance organization. The assessment is performed by analyzing the borrower's questionnaire and calculating each customer's score according to the rules set out in the specific financial structure. The scoring system is a special program with a built-in algorithm for deciding on certain parameters. The reliability and quality of the response depends on the quality of the algorithm. Therefore, it is important not only to create a scoring model, but also to minimize the error of results.

2 Research apparatus

The aim of the study is the theoretical justification and development of a scoring model for microfinance borrowers.

Objectives of the study:

1. To analyze the needs of the lending industry in applying scoring assessment, types of scoring and optimal solutions for this sector.
2. To consider methods of constructing a model for scoring assessment, choose the most optimal one.
3. To choose a model architecture, create software for practical demonstration of scoring.
4. To process the input data, evaluate the initial result and achieve maximum accuracy of the system.

The object of research is creating software for scoring of borrowers.

The subject of research is the development of a scoring assessment model for microfinance organizations.

Research methods: analysis of the literature on banking scoring; artificial intelligence methods for scoring; modeling of scoring estimation algorithm using neural networks, empirical method for determining the optimal parameters of the training model; method of object-oriented design and programming.

The practical significance of the obtained results is a software for microfinance organizations that helps to assess the risk of issuing a loan to a specific borrower, thereby improving the efficiency of these institutions.

3 Theoretical foundations of banking scoring

Credit is an important category of a market economy that reflects the real ties and relationships of economic life in society. The loan originated from the practical needs of production development, its adaptation to the conditions of permanent capital shortage – monetary and material resources.

Credit relations operate in the system of economic relations. They are based on the movement of a special kind of capital – loan capital.

In today's context, the approach to credit organization has changed fundamentally: there has been a shift from object to direct lending to entities. This means that the emphasis in the lending mechanism has shifted from the selection of the entity to the entity's valuation. Commercial and partnership relations between the parties to the agreement exclude the creditor's dictate in determining the object of credit. The risky operations that give the highest income to the bank need to study not only the effectiveness of the activities (projects) under which the funds are allocated, but also the creditworthiness of the client.

A borrower's creditworthiness is his ability to fully and timely settle financial obligations.

The borrower's creditworthiness, unlike its solvency, does not record any insolvency for the current period or for any date, but predicts its solvency in the near term.

One way to organize credit relations is to qualitatively assess the creditworthiness of the borrower. Commercial banks are in dire need of information about the creditworthiness of farms. Their profitability and liquidity depend largely on the financial position of the customers, since the reduction of the risk when performing loan operations can be achieved only based on studying the creditworthiness of clients.

One of the most effective tools for such assessment is the scoring system. Credit scoring enables to make a quick and qualitative decision on a loan application. In addition, its reliability and simplicity are crucial factors in the complex competition [10].

In general, credit scoring can be defined as an assessment of the level of credit risk that results from the processing of various credit history data, which directly or indirectly affects the level of payment discipline [1].

Applying credit scoring, that is, a systematic approach to dealing with credit applications as a whole, allows the bank to:

- Increase the loan portfolio by reducing the number of unjustified refusals of loan applications;
- Improve the accuracy of the borrower's valuation;
- Reduce the level of defaults;
- Speed up the borrower's valuation process;
- Create centralized accumulation of borrower data;
- Reduce provisions for possible losses on credit liabilities;
- Quickly and qualitatively evaluate the dynamics of changes in the credit account of the individual borrower and the credit portfolio as a whole.

All of these have many advantages over a conventional customer rating system and establish the bank's performance.

4 Definition of creditworthiness of the client

Determining the borrower's creditworthiness is an important step in approving a loan application. The main task of the lender is to assess all the risks associated with the possibility of non-repayment of the funds provided.

Credit companies consider the following nuances when evaluating creditworthiness [3; 10; 15]:

- The financial position of the potential borrower;

- Debt load – the ratio of existing liabilities and the requested loan to the applicant's principal income;
- Credit reputation of the client;
- The value of the property owned by the borrower;
- The applicant's social status, personality, career advancement and other factors.

Banking organizations analyze the ability of an individual to pay on a loan. In this case, not only the borrower's monthly income and expenses, but also other factors are taken into account. For example, the risk of job loss and other insured events [15].

Microfinance companies use cheap and fast valuation methods. Therefore, it will take a minute to conduct a scoring test. This is very handy for small loans. However, when it comes to large bank loans, credit professionals can apply all of these methods in combination. This approach will make a specific prediction.

5 Data and methods of scoring in microfinance

5.1 Structure of the main modules

Designing a system that solves the problem of credit scoring can be divided into two main modules: data processing, which includes bringing data to a format that is favorable for computer computing, and directly the module of calculations of the loan decision, containing the interpretation of the algorithm selected for solving the problem teaching. The output of the first module is the input for the second. Therefore, the quality of the result depends on the degree of processing of the primary data.

Data processing includes:

- Data preparation: delete duplicate records, non-informative data columns, records with many null values. Assess the significance of each trait included in the training sample by conducting correlation and regression analyzes.
- Data conversion: categorical features are reduced to a vector form and numeric values should be reduced to a single standard, such as the interval [0, 1] or [-1, 1].

The calculation module consists of:

- Choosing the architecture, parameters of the algorithm.
- Model training on the selected algorithm.
- Testing the model on a deferred sample, determining the calculation error with further correction [9].

5.2 Methods of scoring

Credit scoring is a typical machine learning task. It refers to the type of supervised learning (training with the teacher) [4], namely to the problems of classification, because the solution of the task is reduced to the identification of risks of granting credit for two types (classes): "good" and "bad". The good ones will be those customers who are likely to repay the loan, the bad ones – those who will have a delay of more than 3 months. Sometimes, the “bad” risks include those customers who repay loans early or within a specified period, and the bank does not have time to profit from such clients.

The credit scoring problem can be solved by different machine learning classification methods [7]. These include [2]:

- Statistical methods based on discriminant analysis (linear regression, logistic regression);
- Different linear programming options;
- Classification tree or recursion-partition algorithm;
- Neural networks;
- Genetic algorithm;
- Method of nearest neighbors.

Traditional and most common are regression methods, primarily linear multivariate regression [2]. The disadvantage of the model is that on the left side of the equation is a probability that takes values from 0 to 1, and variables on the right can take any values from $-\infty$ to $+\infty$. In addition, this model is unstable to emissions; any sudden value that gets out of the picture can lead to the wrong answer.

Linear programming also leads to a linear scoring model. It is impossible to carry out a completely accurate classification of “bad” and “good” clients, but it is desirable to minimize the error. The task is to find the weights for which the error will be minimal [17].

Classification trees are a method that allows the observation or object to be assigned to a particular class of categorical dependent variable according to the values of one or more predictor variables. Classification trees are tailored to the graphical representation, so they have a more convenient look for human understanding. The disadvantages are instability, small changes in the data can significantly change the built decision tree, the problem of finding the optimal depth of the tree, the complexity of data gaps.

The genetic algorithm is based on an analogy with the biological process of natural selection. In the field of lending, it looks like this: there is a set of classification models that can be “mutated”, “crossed”, and as a result, the “strongest” model is selected, which gives the most accurate classification.

When using the nearest-neighbor method, a unit of measure is selected to determine the distance between clients. All clients in the sample are given a specific spatial position. Each new client is classified based on which clients – good or bad – are more around him [2].

Neural Networks – a common solution to classification problems. Artificial neural network – a mathematical model, which is built on the principle of organization and operation of biological neural networks, is a system of connected and interacting simple processes [8].

In scoring, the use of neural networks has the least use compared to other methods. Nevertheless, the neural network has significant advantages. These advantages include the possibility of automatic learning of the model, the versatility of working with different scales of measurement of dependent and independent variables, the ability to approximate any continuous function of dependence. The mathematical model of neural network scoring makes it possible, based on a set of known characteristics of a research object, to predict a specific characteristic that is unknown to the researcher [6; 18].

The neural network model meets all the above-mentioned needs of the domain, such as speed and precision of calculations, scalability of data, possibility of introduction of new characteristics without significant deterioration of quality of estimation, easy modernization of the system on demand. Therefore, based on the advantages of using neural networks, this method was chosen to solve the credit-scoring problem.

Building a neural network has its own characteristics and steps that you must go through to get a truly high-quality model.

Building a neural network starts with data preparation. At this point, it is necessary to delete duplicate records, non-informative data columns, records that have many null values. At the same time, there should be a sufficient number of examples for neural network training. There is an imperial rule that establishes the recommended ratio between the number of training examples that have input and the correct answers and the number of connections in the neural network: X<10.

For the facts included in the training sample, it is advisable to estimate its significance in advance by conducting correlation and regression analyzes and to consider the ranges of their possible changes. All this is an important component in the methodology of building a scoring model [3].

The second stage is the conversion of the initial data, taking into account the nature and type of problem solved by the neural network, the means of presentation of information are selected. For example, categorical features should be reduced to a vector form and numerical values should be reduced to a single standard, such as the interval [0, 1] or [–1, 1].

The third stage is the choice of network architecture. It is necessary to define such parameters as the number of network layers, the number of neurons of each layer, and the activation functions to be used [18].

The number and type of independent variables in the model determine the number of neurons in the input layer. For categorical variables, it is advisable to use one input neuron for each category, with only one neuron in the group being activated for each observation.

The architecture of the source layer of the neural network is also dictated by the structure of the problem. One output neuron is created for each dependent quantitative variable.

No techniques have been developed to determine the number of hidden layers and the number of neurons in them. In practice, these parameters are experimentally determined by analyzing the quality of approximation provided by networks of different sizes. Thus, a network with an input layer of 58 inputs, one hidden layer with 30 neurons and 2 outputs was selected.

At the input of a neuron, we have a vector of parameters. These are the results of the collection of billing information about a potential customer, presented in numerical form $X^i = \{x^i_{1,} x^i_2, \ldots, x^i_n\}$.

In this case, each client is responsible for the class Y^i. In total, there are two classes of the set Y with the following values: 1 – to give credit, 0 – not to give. The neural network, in fact, must find the optimal separating hypersurface in the vector space, the dimension of which will correspond to the number of features. Learning the neural network in this case is to find such values (coefficients) of the weight matrix, in which the neuron responsible for the class will give values close to one in cases where credit is approved, and values close to zero, if not.

As we can see from formula 1, the result of the neuron $h_w(X)$ is a function of activation f from the sum of the product of the input parameters x_k to the coefficients required $w_k \cdot x_k$ in the learning process:

$$h_w(X) = f(\sum_{k=1}^{|w|} w_k \cdot x_k)h_w(X) = f(\sum_{k=1}^{|w|} w_k x_k) \quad (1)$$

It is desirable to interpret the value derived from a neuron in the range [0, 1] as the probability of belonging to a class. Therefore, such a monotonous smooth function is required, which will display elements of the set of real numbers in the range from zero to one. The activation function of sigmoid (2) is the best way to do this. The function graph is shown in Figure 1.

$$\partial(w \cdot x) = \frac{1}{1+e^{-x}} \quad (2)$$

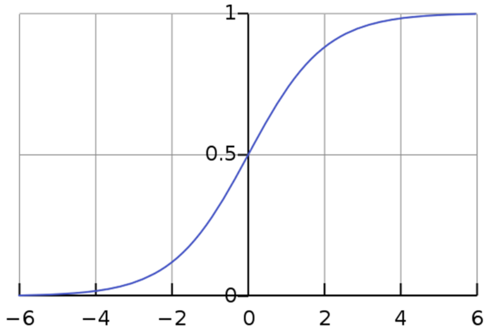


Fig. 1. Plot of sigmoid function

The fourth stage is learning the network. In the selected data set, each sample object is assigned a class to which it belongs. Therefore, we are tasked with the type of supervised learning [16]. Therefore, in the learning process, the network must review the sample many times, each complete passage of the sample is called the learning age. To train the model, you need to split the data into two parts – the actual training and the test.

We apply the standard separation in the ratio of 80% and 20% respectively [11].

Neural network training should be understood as the weighting for each of the traits based on the results obtained from past data views. The backpropagation method is selected for weight correction. See [5] for more on this method.

It has been determined experimentally that 64 epochs are required to select the optimal weights for a given neural network. 221,712 training sample records are processed for each epoch. The results are Table 2.

Table 1. The results of calculating epochs

Epoch number	Calculation time, s	Error	Accuracy
1	22	0.4871	0.6860
2	21	0.3066	0.7271
3	22	0.2472	0.7748
4	21	0.2128	0.7951
5	23	0.1970	0.8155
…	…	…	…
61	22	0.0304	0.9518
62	22	0.0275	0.9529
63	22	0.0255	0.9537
64	22	0.0251	0.9540

The fifth step is to test the received neural network model on a delayed sample. The test sample includes only those records that did not participate in the training network. We have 55,428 records. The accuracy of the calculations on the test sample is 95.4%. In this case, the error in the decision to issue a loan – 0.00469, the error in the loan prohibition decision is 0.00406.

6 Software architecture and operation

6.1 Architecture

The client server architecture of the program was chosen to develop the software for rapid assessment of customer solvency.

Client-server is a software architecture model of two parts, client systems and server systems, both communicating over a computer network or on the same computer. A client-server application is a distributed system made up of both client and server software. The client-server application provides a better way to share the workload. The client process always initiates a connection to the server, while the server process always waits for requests from any client.

The client-server relationship describes the relationship between the client and how it makes a service request to the server, and how the server can accept these requests, process them, and return the requested information to the client [12].

6.2 Description of software operation

To build a functional diagram, the IDEF0 methodology was chosen, which is considered the classic method of the process approach to design. In IDEF0, the system that is being modeled is represented as a set of interrelated works (functions, activities). To develop a functional diagram, 4 main functions of the program were presented, which are presented on blocks A1–A4 (Fig. 2). Each of these functions has input and output data, control information and mechanisms through which the function can be executed.

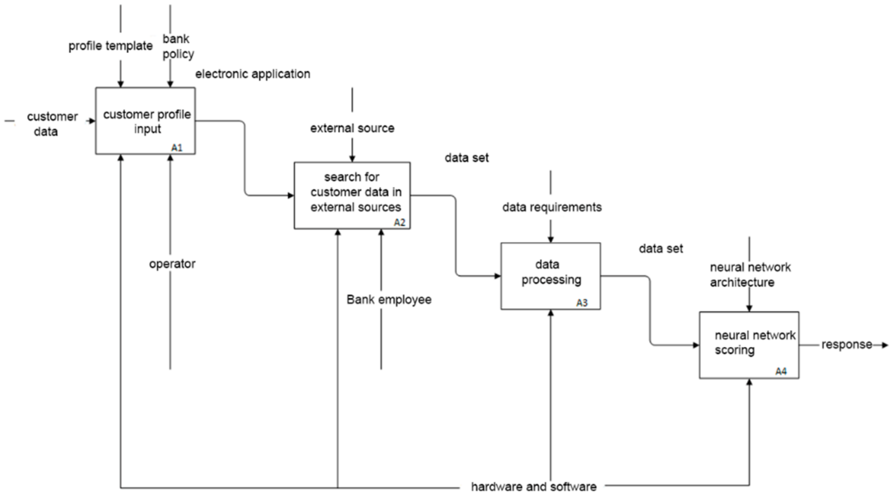


Fig. 2. Functional program scheme

Block A1. Block maintaining the customer profile. At the input, it has data coming from a bank client. This data is written to the database in a format structured in accordance with the bank policy and the accepted questionnaire template. The bank operator maintains the questionnaire. It can both enter data, view it, perform a search, delete it.

Block A2. After the data are entered into the database and the electronic application is generated, the customer data is searched in external sources. As a rule, these are Internet portals or credit bureaus. This approach helps to obtain more detailed information about the client and his solvency.

Block A3. Data processing. It provides functionality for preparing data for further calculations, all non-informative data is deleted, data is brought to a single range. Data processing is based on the analysis of data requirements.

Block A4. The processed network dataset enters the neural network model. The structure and quality of functioning of the neural network are determined by its architecture. The model gives the result of a loan to the borrower.

All blocks as the executing mechanism have software and hardware.

The system is a module for the banking system, so this imposes certain requirements on the program interface.

Firstly, an official and minimalistic style of design should be maintained. Light background colors of the components and dark text color on them.

Secondly, all elements should be located at an optimal distance from each other, to prevent errors that in the banking sector can cost both time and money.

Thirdly, the size of the inscriptions should be sufficient for the readability of the text. The text on the components must match the actual functionality that the component executes.

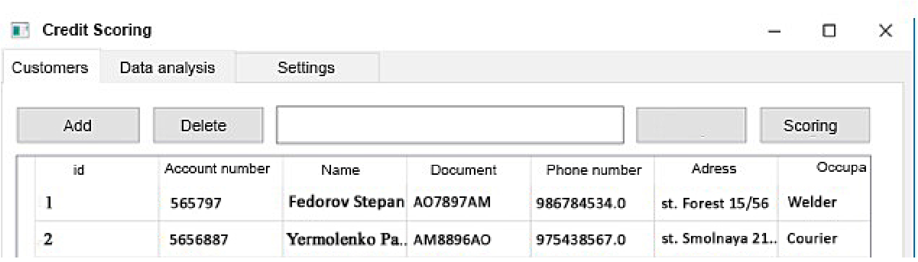
An important part of the interface is the presence of messages about the actions performed in the program, as well as the status of request processing.

The developed software consists of several tabs: “Clients”, “Data Analysis” and “Settings”.

The Customers tab has buttons for managing customer data. It is possible to add new clients, remove them from the database, search the client for the database, and directly score on the selected client.

The developed software consists of several tabs: “Clients”, “Data Analysis” and “Settings”.

The Customers tab (Fig. 3) has buttons for managing customer data. It is possible to add new clients, remove them from the database, search the client for the database, and directly score on the selected client.



id	Account number	Name	Document	Phone number	Adress	Occupa
1	565797	Fedorov Stepan	AO7897AM	986784534.0	st. Forest 15/56	Welder
2	5656887	Yermolenko Pa..	AM8896AO	975438567.0	st. Smolnaya 21..	Courier

Fig. 3. Customers tab

To view the analytics data to monitor the status of credit disbursements and other parameters, go to the Data Analysis tab (Fig. 4). The graph will be automatically generated according to the selected criterion from the drop-down menu.

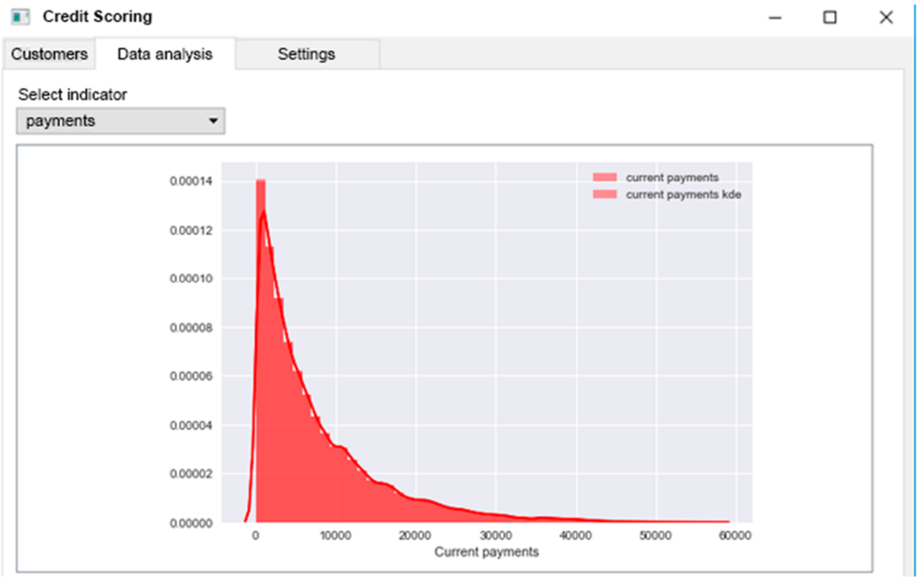


Fig. 4. Data Analysis Tab

The user can also change the neural network settings by going to the Settings tab. The banking system is dynamic and sometimes it is necessary to re-evaluate the weight of the features. In the list of features, you should select the ones that will be included in the new model using the multiple choice in the box. You can select the number of hidden layers and the number of neurons in them.

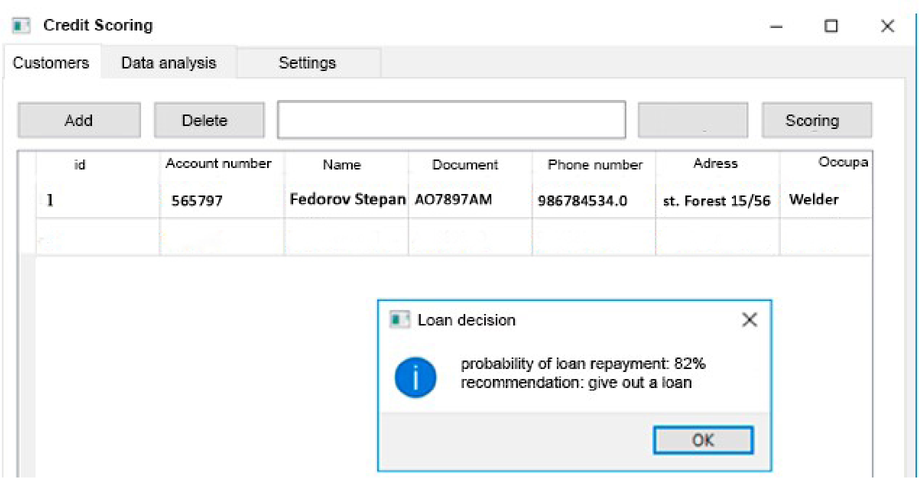
7 Results

As a result, a system of automatic crediting of clients in the sphere of microfinance was created. For this system, the best artificial neural network architecture with all the relevant settings was selected.

Using this decision support system on the artificial neural network platform to make serious decisions such as credit decisions significantly simplify lending operations, reduce the risk of default.

Testing the system gave fairly accurate results, which indicates a high degree of trust in the software.

The result of the program (Fig. 5):



id	Account number	Name	Document	Phone number	Adress	Occupa
1	565797	Fedorov Stepan	AO7897AM	986784534.0	st. Forest 15/56	Welder

probability of loan repayment: 82%
recommendation: give out a loan

Fig. 5. The result of the program

8 Conclusion

The study showed how artificial neural networks could be used to build an automatic customer lending system.

In accordance with this goal, the following results were obtained: the current state of the problem and task of crediting clients are analyzed; the modern decision support systems are analyzed and the choice of artificial neural network systems as the basic technological platform is grounded; the data model necessary for the proper functioning of the system is built; selected the best neural network architecture for a specific task; automatic system of crediting of clients is implemented.

Building a neural-boundary model for the credit-scoring problem has shown that this method is one of the best for finding the most accurate solution for issuing a loan, as evidenced by a very low calculation error.

The introduction of such a system in a microfinance institution helps to improve the performance of the institution, automate the processes associated with credit loans, reduce the risk of errors and fraud.

References

1. Aleshin, V.A, Rudayeva, O.O.: Kreditnyj skoring kak instrument povysheniya kachestva bankovskogo risk-menedzhmenta v sovremennyh usloviyah (Credit scoring as an instrument for improving the quality of banking risk management in current conditions). Terra economicus. **10**(2), 27–30 (2012)
2. Allison, P.D. (ed.): Logistic regression using the SAS system: theory and application. SAS Institute, Stanford (2012)
3. Anderson, R.: The credit scoring toolkit: theory and practice for retail credit risk management and decision automation. Oxford University Press, New York (2007)
4. Coelho, L.P., Richert, W.: Building Machine Learning Systems with Python. Packt Publishing, Birmingham (2013)
5. Flach, P.: Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge University Press, Cambridge (2012)
6. Haykin, S.: Neural Networks and Learning Machines, 3rd edn. Pearson, New Jersey (2008)
7. Kiv, A., Semerikov, S., Soloviev, V., Kibalnyk, L., Danylchuk, H., Matviychuk, A.: Experimental Economics and Machine Learning for Prediction of Emergent Economy Dynamics. In: Kiv, A., Semerikov, S., Soloviev, V., Kibalnyk, L., Danylchuk, H., Matviychuk, A. (eds.) Experimental Economics and Machine Learning for Prediction of Emergent Economy Dynamics, Proceedings of the Selected Papers of the 8th International Conference on Monitoring, Modeling & Management of Emergent Economy (M3E2 2019), Odessa, Ukraine, May 22-24, 2019. CEUR Workshop Proceedings **2422**, 1–4. http://ceur-ws.org/Vol-2422/paper00.pdf (2019). Accessed 17 Aug 2019
8. Lewis, E.M. An introduction to credit scoring. Athena Press, London (1992)
9. Luo, F.L., Unbehauen, R.: Applied Neural Networks for Signal Processing. Cambridge University Press, Cambridge (1997)
10. Mays, E. (ed.): Handbook of credit scoring. Global Professional Publishing, Chicago (2001)
11. Rojas, R.: Neural Networks: A Systematic Introduction. Spring-Verlag, Berlin (1996)
12. Saternos, C.: Client-Server Web Apps with JavaScript and Java. O'Reilly Media, Sebastopol (2014)
13. Semerikov, S.O., Teplytskyi, I.O., Yechkalo, Yu.V., Markova, O.M., Soloviev, V.N., Kiv, A.E.: Computer Simulation of Neural Networks Using Spreadsheets: Dr. Anderson, Welcome Back. In: Ermolayev, V., Mallet, F., Yakovyna, V., Kharchenko, V., Kobets, V., Korniłowicz, A., Kravtsov, H., Nikitchenko, M., Semerikov, S., Spivakovsky, A. (eds.) Proceedings of the 15th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer (ICTERI, 2019), Kherson, Ukraine, June 12-15 2019, vol. II: Workshops. CEUR Workshop

Proceedings **2393**, 833–848. http://ceur-ws.org/Vol-2393/paper_348.pdf (2019). Accessed 30 Jun 2019

14. Semerikov, S.O., Teplytskyi, I.O., Yechkalo, Yu.V., Kiv, A.E.: Computer Simulation of Neural Networks Using Spreadsheets: The Dawn of the Age of Camelot. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 122–147. http://ceur-ws.org/Vol-2257/paper14.pdf (2018). Accessed 30 Nov 2018
15. Siddiqi, N.: Credit risk scorecard: developing and implementing credit scoring. John Wiley and Sons, New Jersey (2006)
16. Sorokin, A.S.: K voprosu validacii modeli logisticheskoj regressii v kreditnom skoringe (On the validation of the logistic regression model in credit scoring). Naukovedenie 2. http://naukovedenie.ru/PDF/173EVN214.pdf (2014). Accessed 10 Nov 2019
17. Sorokin, A.S.: Postroenie skoringovyh kart s ispolzovaniem modeli logisticheskoj regressii (Construction of scoring maps using a logistic regression model). Naukovedenie 2. http://naukovedenie.ru/PDF/180EVN214.pdf (2014). Accessed 10 Nov 2019
18. Sorokin, S.V., Sorokin, A.S.: Ispolzovanie nejrosetevyh modelej v povedencheskom skoringe (Use of neural network models in behavioral scoring). Prikladnaja informatika **10**(2(56)), 92–109 (2015)

Development of a chatbot for informing students of the schedule

Andrii O. Priadko, Kateryna P. Osadcha[0000-0003-0653-6423],
Vladyslav S. Kruhlyk[0000-0002-5196-7241] and Volodymyr A. Rakovych[0000-0002-0402-8073]

Bogdan Khmelnytsky Melitopol State Pedagogical University,
20, Hetmanska Str., Melitopol, 72300, Ukraine
mrstr91@gmail.com, okp@mdpu.org.ua,
krugvs@gmail.com, rackovychv@gmail.com

Abstract. The article describes the process of developing a chatbot to provide students with information about schedules using the Telegram mobile messenger. During the research, the following tasks have been performed: the analysis of notification systems for their use in the educational process, identification of problems of notifying students about the schedule (dynamic environment, traditional presentation of information, lack of round-the-clock access), substantiation of the choice of mobile technologies and Telegram messenger, determination of the requirements to the software, generalization of the chatbot functioning features, description of the structure, functionality of the program to get information about the schedule using a chatbot. The following tasks have been programmatically implemented: obtaining data from several pages of a spreadsheet (faculty / institute, red / green week, group number, day of the week, period number, discipline name, information about the teacher); presentation of data in a convenient form for the messenger (XML); implementation of the mechanism of convenient presentation of data in the messenger (chatbot). Using Python and the Telegram API, the software has been designed to increase students; immediacy in getting the information about the schedules, minimizing the time spent, and optimizing of planning of student activities and higher education institution functioning.

Keywords: chatbot, schedule, mobile applications, API, Telegram, Python, XML.

1 Introduction

Setting up an effective notification for students about the classes schedule is a problem that leads to the development of convenient and effective tools. There were attempts to develop programs and systems for automated scheduling. In particular, with the use of genetic algorithms, Wilhelm Erben and Jürgen Keppler [4] developed a specific chromosome representation and knowledge-augmented genetic operators that “intelligently” avoid building illegal timetables. The prototype timetabling system which is presented has been implemented in C and PROLOG, and includes an interactive graphical user interface and was tested using real university data, which produced promising results. Yulan He, Siu Cheung Hui and Edmund Ming-Kit Lai [6] claim that their approach based on artificial immune system gives better results in creating automated individual schedules than those obtained with a genetic algorithm. For automatic scheduling developers [13] also use a network flow algorithm to match resources and assign them to classes. As a result, the schedule control application automatically generates class schedules, with consideration to resources and limits specified by user. The algorithm used to generate the schedules is based on Ford-Fulkerson algorithm. The application aggregates data and sends to the algorithm as the input via XML. The algorithm generates classes in the schedules in the form of XML. The application allows the user to select one schedule out of three presented schedules, modify and finally export it. It is expected that time spent on the whole schedule publication process is reduced due to the reduction in time spent on the sub-process, and errors are kept low comparing to manual work. Using the features of Google Calendar, Illia V. Oleksiienko and Vasyl M. Franchuk [18] have developed a web-based system for entering and viewing an electronic class schedule for a higher education institution. The Malaysian scientists' solution is interesting and effective [11], they offered a model of university courses' scheduling using an ant colony optimization algorithm as a powerful approach to solving various combinatorial optimization problems.

However, the problem of informing students about the schedule of classes and its changes is not sufficiently explored. Some studies have examined notification systems and their use in the learning process:

- a portable organizer device through which a student may use and maintain a personal calendar [10];
- the learning notification express delivery service which provides students with information about their learning status, follow-up activities, and more [22];
- mobile agent-based event notification system [3];
- an online reference and notification system for college students based on individual user profiles [2];
- students attendance system and notification of college subject schedule based on IBeacon use [7] and more.

2 Problem statement

In higher education institutions, class schedules play a major role in the daily lives of students, faculty, and staff, informing about the time and place of study. Changes can occur when a schedule is formed in a dynamic environment such as school. The question arises as to how to promptly inform students about the stationary schedule and changes in it.

Nowadays, the schedule notification system is traditional: a paper version on the bulletin board for students and at the department for teachers or at an electronic educational institution. It is not always convenient for students to use a paper version that does not have 24-hour access. It is also inconvenient to use a photo taken on a mobile phone: it is hard to navigate, it is easy to lose the photo of a schedule among many photos on a mobile phone, it is impossible to receive notification about changes in the schedule.

It is advisable to use the most commonly available online programs and services for developing a notification system, so that students can receive timetable information promptly. As stated in We Are Social and Hootsuite [8], the proportion of Internet users using mobile phones and the proportion of mobile messenger users is increasing worldwide. According to a study [12] by the Mobile Marketing Agency LEAD9 and the Kiev International Institute of Sociology, the most popular messengers among Ukrainians are Viber, Facebook, Skype and Telegram. Using Facebook and Skype to develop a timetable retrieval system is not appropriate because only few students use these messengers on mobile phones. Therefore, it is advisable to use the capabilities of messengers such as Viber or Telegram.

3 Mobile technologies in the educational process

Mobile technologies in the educational process of higher education institutions are most commonly used for the organization of mobile learning. Use of a mobile device for didactic purposes is possible for: 1) student's work remotely; 2) delivery of educational materials; 3) testing the course knowledge. Remotely, the student may perform certain educational or research work on a teacher's assignment. Sending training materials today depends on the type of mobile phone and the technologies it supports. The most up-to-date mobile devices use Opera Mini, Internet Explorer Mobile, Safari, etc. to browse the web are fairly capable of displaying readable text and graphics [19]. Cloud-based learning environment built on the integrative usage of mobile Internet devices promotes the forming of Bachelor's of Electromechanics professional competencies [15; 16; 17].

Mobile learning is based on the intensive use of modern mobile tools and technologies. It is also linked to learning mobility in the sense that students should be able to participate in educational activities without restrictions in time and space [21]. Therefore, the timely receipt by students of schedule information and schedule changes has a direct relation on the effectiveness of the educational process. Therefore, the development and use of software to enable this are appropriate.

4 Defining software requirements

Given that the schedule is presented in Google Sheets format embedded in the website, the task is:

1. get data from several pages of the spreadsheet (faculty / institute, red / green week, group number, day of the week, pair number, discipline name, teacher information);
2. submit them in a convenient form for the messenger;
3. implement a mechanism for convenient presentation of data in the messenger.

An interesting solution to this problem is the software solution given in the work [9], which presents a process for extracting AIML (artificial intelligence markup language) chatbot knowledge bases from a text corpora using the AGATA (Automatic Generation of AIML from Text Acquisition) system, designed to facilitate and reduce the time needed in that task. The study consisted of converting the contents of two textbooks into AIML and sending the files to a chatbot knowledge base. This technology allows teachers to easily create their own chats according to the topics of the course.

Based on this research and the research discussed above, the chatbot, that retrieves data from electronic documents with schedule information using a high-level program that extracts them into an XML file and it provides the user with scheduling information as requested by the messenger API, is set out to be created.

Given that Viber chatbots are paid [20], we have chosen the Telegram messenger to develop a chatbot for scheduling information.

5 Features of chatbots functioning

Setting up an effective notification of students about the classes schedule is a problem that leads to the development of convenient and effective tools.

Currently, according to an OpenMarket study [14], 77% of US consumers are more positive about text messaging companies, and 64% of Internet users prefer texting rather than voice messaging. These circumstances prompted the development of chatbots. The formation of artificial intelligence as a new research direction has stimulated the development of modern Intelligent Information Systems (IIS) and intelligent information technologies, which have found applications in almost all areas of human activity, providing automation of manufacturing processes, improving the effectiveness of management and training, in the development of ontologies, Knowledge Bases (KB) and in other cases [5].

A chatbot is a program that provides people with the service of communication through artificial intelligence. Chatbots are designed to work with virtual assistants. They themselves provide a platform for the promotions of the Products and Services online [1]. The use of chatbots is gaining popularity in several sectors: medicine, economics, psychology, sociology, etc. However, in the educational field, chatbot is rarely used as an assistant in training, education or organization of the educational process.

The process of using chatbots in messengers is pre-programmed based on appropriate databases of user interaction, sometimes in complex systems using artificial intelligence. Some chatbots use sophisticated natural language processing systems, but many simpler ones scan for keywords in typed text or a voicemail and then retrieve the answer with the most relevant keywords or most similar formulation template from the database. With the help of a chatbot, the user can get answers to questions, information about specific requests, recommendations, psychological support and more. The value of chatbots in dialog systems is their high engagement, responsiveness, and ability to automate basic user interaction processes.

6 Description of the developed application for obtaining schedule information using a chatbot

In accordance with the Python programming language requirements, an application was developed to retrieve the data from the spreadsheet csv-pages and convert them to xml format to further use the Telegram API. The application in its structure has:

1. Python (.py) files – a chatbot application file, a program file to update schedule information, and a basic interaction file for all application elements;
2. folder with xml-files – a list of disciplines, teachers, students and schedule;
3. folder with Python files for retrieving csv files and csv processing results.

Xml files with schedule information and discipline list are updated upon request. Schedule data is downloaded from Google Drive, which stores the schedules of all departments. To make the application work the list of teachers and students with relevant data should be made. For the teachers list: the department where the teacher works, the code, the surname, first name and patronymic of the teacher. For the student list, the group number, surname, first name and patronymic of the student. The xml file with the schedule information contains the following entries: day of the week (number), group number, pair number, subject code (and list of subjects), teacher code (from the list of teachers), office number.

To run the application, the chatbot application must work on a computer connected to the network or on a web host. The Telegram messenger must be installed on the user's mobile phone and the user needs to access the @mr_raspisane_bot chatbot. After joining the chatbot, pressing the START button in the Telegram chat starts the chatbot. The user sends a message to chatbot and it sends it to the main Telegram server, which sends it to the computer or hosting that runs the chatbot. Everything is processed there and the answer is returned in the reverse order to the user (Fig. 1).

The following Python language modules and libraries were used to develop the chatbot application:

1. standard: os – provides functions to work with the operating system; datetime – provides classes for processing time and dates in various ways; xml.etree.ElementTree – API implementation for working with XML files in Python; csv – implements classes for reading and writing tabular data in .csv format;
2. not standard: telebot – a library of Telegram bots with simple route decorators, distributed under the MIT license for free software; wget – allows downloading the contents of files located at a certain url; openpyxl – Python library for reading / writing Excel 2010 files xlsx / xlsm / xltx / xltm;
3. designed: raspisanie – with features for finding a group number, discipline name, teacher information, and classroom.

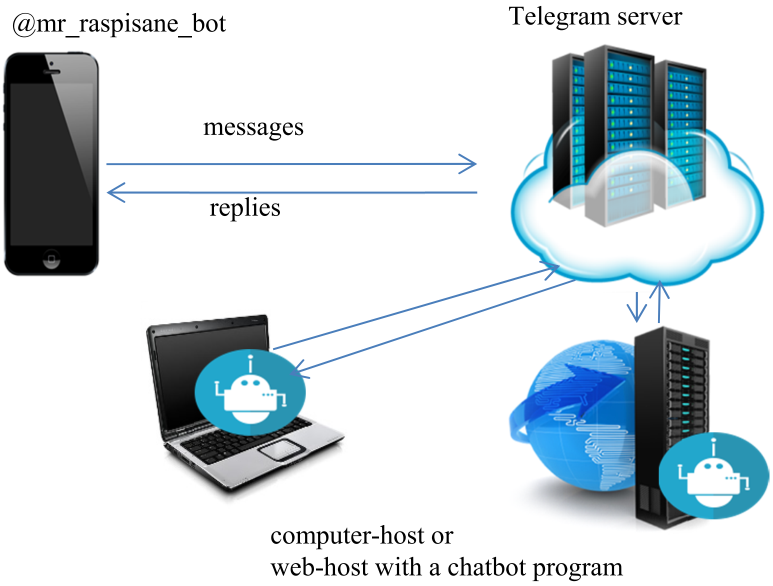


Fig. 1. Functional diagram of the chatbot (image for the scheme is taken at https://www.google.com/imghp with a re-use license)

The following is a snippet of raspisanie.py module code that allows you to find the class by name (parser_uchen function), name of the subject (function parser_predm), teacher information (function fio_prep) and student information (poisk_w_baze function):

```
import xml.etree.ElementTree as ET

import datetime

#finds a class by name
def parser_uchen(wr_fio):
    tree = ET.parse('xml/Uchen.xml')
    root = tree.getroot()
    for s in root:
        for ss in s:
            fio = ""
            for sss in ss:
                fio += str(sss.text) + " "
            if wr_fio in fio:
                qwe = (s.attrib)
                return(qwe['NomerClassa'])
```

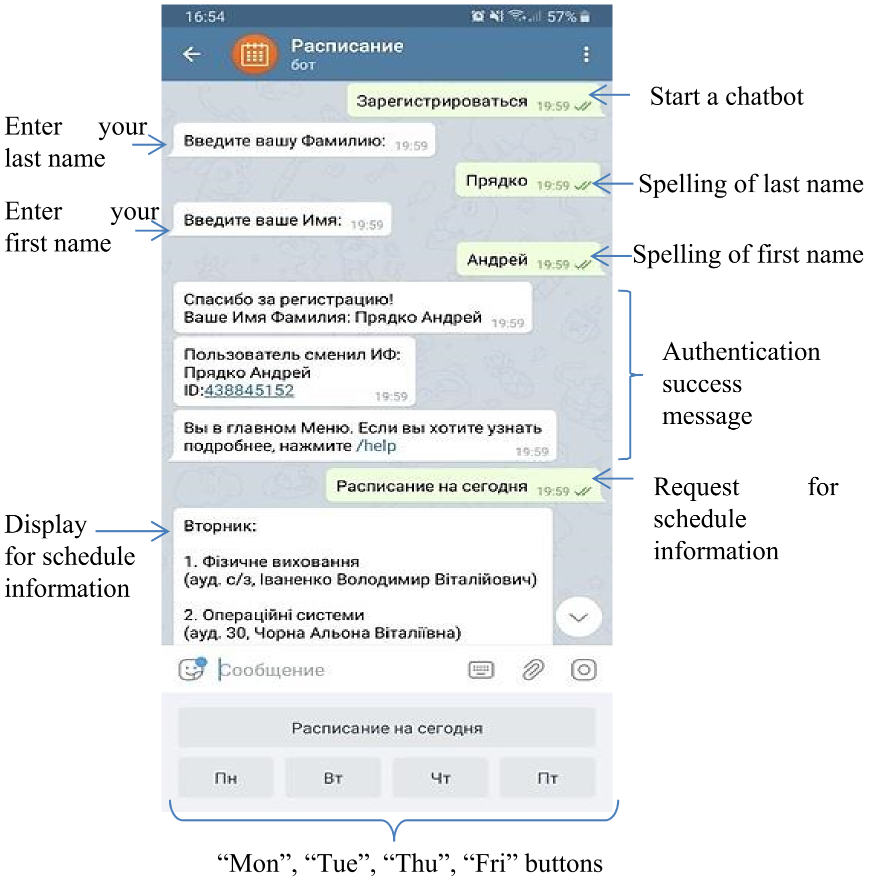
```
#find out what subject
def parser_predm(kod_predm):
  tree = ET.parse('xml/Predmet.xml')
  root = tree.getroot()
  for s in root:
    if s.attrib['kodPredm'] == kod_predm:
      return(s.text)

#find the name of the teacher
def fio_prep(kod_prep):
  fio_prepod = ""
  tree = ET.parse('xml/Prepod.xml')
  root = tree.getroot()
  for s in root:
    if int(s.attrib['KodPrep']) == int(kod_prep):
      for ss in s:
        fio_prepod += str(ss.text) + " "
  fio_prepod = fio_prepod[:-1]
  return(fio_prepod)

#find the student’s name and surname
#from the list in the database
def poisk_w_baze(chat_id):
    d = {}
    with open("baza.txt") as file:
        for line in file:
            d = line.split()
            if str(d[2]) == str(chat_id):
                return(str(d[0] + " " + d[1]))
```

Example structure of a xml-document with schedule information :

```
<?xml version="1.0" encoding="UTF-8"?>
<Raspisanie xsi:noNamespaceSchemaLocation="rasp.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Den DenNed="1">
<Class NumClassa="217-i">
<NumLesson NumLes="1">
<Predmet kodPredm="4">
<Prepod KodPrep="4">
<Kabinet>75</Kabinet>
</Prepod>
</Predmet>
</NumLesson>
<NumLesson NumLes="2">
<Predmet kodPredm="3">
```

```
<Prepod KodPrep="6">
<Kabinet>11</Kabinet>
</Prepod>
</Predmet>
</NumLesson>
```

The result of the chatbot in Telegram is shown in Fig. 2.



16:54 57%

Расписание
бот

Зарегистрироваться 19:59

Введите вашу Фамилию: 19:59

Прядко 19:59

Введите ваше Имя: 19:59

Андрей 19:59

Спасибо за регистрацию!
Ваше Имя Фамилия: Прядко Андрей 19:59

Пользователь сменил ИФ:
Прядко Андрей
ID:438845152 19:59

Вы в главном Меню. Если вы хотите узнать подробнее, нажмите /help 19:59

Расписание на сегодня 19:59

Вторник:

1. Фізичне виховання
(ауд. с/з, Іваненко Володимир Віталійович)

2. Операційні системи
(ауд. 30, Чорна Альона Віталіївна)

Fig. 2. The result of the chatbot in Telegram

During the “communication” with the chatbot, the user has to enter his/her name and surname. By clicking on the “Schedule for Today” button, the user instructs to search the information according to the data and receives a response in the form of text describing the schedule for the current day. The “Mon”, “Tue”, “Thu”, “Fri” buttons can be used to get a schedule for these days.

7 Conclusions

In the research, it was determined that modern chatbot software in popular messengers on mobile phones can increase the efficiency in getting students information about the schedule. Reducing waste of time contributes to the efficiency of the educational process organization and the planning of students' educational activities. The chosen characters development tools for informing the student about the schedule (Python, XML, Telegram API) made it possible to quickly develop and launch the software tool in the student environment.

References

1. Arun, K., Sri Nagesh, A., Ganga, P.: A Multi-Model And Ai-Based Collegebot Management System (Aicms) For Professional Engineering Colleges. International Journal of Innovative Technology and Exploring Engineering **8**(9), 2910–2914 (2019). doi:10.35940/ijitee.I8818.078919
2. Brody, B.M., Brody, B.R.: Directory and notification system for college students based on individual user profiles. US Patent US20110107232A1, 5 May 2011
3. El-Gazzar, R.F., Badawy, O., Kholief, M.: Agent-Based Mobile Event Notification System. International Journal of Interactive Mobile Technologies **4**(4), 25–30 (2010)
4. Erben, W., Keppler, J.: A genetic algorithm solving a weekly course-timetabling problem. In: Burke E., Ross P. (eds.) Practice and Theory of Automated Timetabling. PATAT 1995. Lecture Notes in Computer Science, vol. 1153, pp. 198–211. Springer, Berlin, Heidelberg (1996). doi:10.1007/3-540-61794-9_60
5. Eremeev, V.S., Osadchyi, V.V., Gulynina, E.V., Doneva, O.V.: A mathematical model of an intelligent information system for a comparative analysis of European qualification standards. Global Journal of Pure and Applied Mathematics **12**(3), 2113–2132. https://www.ripublication.com/gjpam16/gjpamv12n3_14.pdf (2016)
6. He, Y., Hui, S.C., Lai, E.MK.: Automatic Timetabling Using Artificial Immune System. In: Megiddo N., Xu Y., Zhu B. (eds) Algorithmic Applications in Management, First International Conference on Algorithmic Applications in Management, Xian, June 2005. Lecture Notes in Computer Science, vol. 3521, pp. 55–65. Springer, Berlin, Heidelberg (2005). doi:10.1007/11496199_8
7. Hidayat, M.A., Simalango, H.M.: Students Attendance System and Notification of College Subject Schedule Based on Classroom Using IBeacon. In: 3rd International Conference on Information Technology, Information System and Electrical Engineering, Yogyakarta, Nov. 2018, pp. 253–258. IEEE (2018). doi:10.1109/ICITISEE.2018.8720948
8. Kemp, S.: Digital in 2018: World's Internet users pass the 4 billion mark. We Are Social. https://wearesocial.com/blog/2018/01/global-digital-report-2018 (2018). Accessed 17 Aug 2019
9. Krassmann, A.L., Flach, J.M., da Silva Grando, A.R.C., Tarouco, L.M.R., Bercht, M.: A process for extracting knowledge base for chatbots from text corpora. In: IEEE Global Engineering Education Conference, Dubai, 8-11 April 2019, pp. 322–329. IEEE (2019). doi:10.1109/EDUCON.2019.8725064
10. Kulack, F.A.: Pervasive educational assistant and study aid for students. US Patent 10,039,301, 12 Apr. 2005
11. Mazlan, M., Makhtar, M., Ahmad Khairi, A.F.K., Mohamed, M.A.: University course

timetabling model using ant colony optimization algorithm approach. Indonesian Journal of Electrical Engineering and Computer Science **13**(1), 72–76 (2019)

12. MediaSapeins: Viber vyiavyvsia naipopuliarnishym mesendzherom sered ukraintsiv - doslidzhennia (Viber was the most popular messenger among Ukrainians – research). https://ms.detector.media/mediaprosvita/research/viber_viyavivsya_naypopulyarnishim_ mesendzherom_sered_ukraintsiv_doslidzhennya (2018). Accessed 14 Aug 2019
13. Mekareeya, W., Satiti-Thanawisit, G., Suansilppongse, N., Siripolsomsuk, C., Praiwattana, P.: Schedule management application: Automatic schedule generation using network flow algorithm. In: 3rd ICT Intl Senior Project Conference, Nakhon Pathom, 26-27 March 2014, pp. 21–24. IEEE (2014). doi:10.1109/ICT-ISPC.2014.6923209
14. Michael: eRetailers: Text Consumers or Risk Irrelevancy in 2017. https://mobilemarketingwatch.com/eretailers-text-consumers-or-risk-irrelevancy-in-2017-69642 (2016). Accessed 17 Aug 2019
15. Modlo, Ye.O., Semerikov, S.O., Nechypurenko, P.P., Bondarevskyi, S.L., Bondarevska, O.M., Tolmachev, S.T.: The use of mobile Internet devices in the formation of ICT component of bachelors in electromechanics competency in modeling of technical objects. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018), Kryvyi Rih, Ukraine, December 21, 2018. CEUR Workshop Proceedings **2433**, 413–428. http://ceur-ws.org/Vol-2433/paper28.pdf (2019). Accessed 10 Sep 2019
16. Modlo, Ye.O., Semerikov, S.O., Shmeltzer, E.O.: Modernization of Professional Training of Electromechanics Bachelors: ICT-based Competence Approach. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 148–172. http://ceur-ws.org/Vol-2257/paper15.pdf (2018). Accessed 21 Mar 2019
17. Modlo, Ye.O., Semerikov, S.O.: Xcos on Web as a promising learning tool for Bachelor's of Electromechanics modeling of technical objects. In: Semerikov, S.O., Shyshkina, M.P. (eds.) Proceedings of the 5th Workshop on Cloud Technologies in Education (CTE 2017), Kryvyi Rih, Ukraine, April 28, 2017. CEUR Workshop Proceedings **2168**, 34–41. http://ceur-ws.org/Vol-2168/paper6.pdf (2018). Accessed 21 Mar 2019
18. Oleksiienko, I.V., Franchuk, V.M.: Web-oriented electronic schedule. CEUR Workshop Proceedings **2292**, 128–131 (2018)
19. Osadchiy, V.V.: Teoriia i praktyka orhanizatsii mobilnoi tekhnolohii navchannia u pedahohichnomu universyteti (Theory and Practice of Mobile Learning Technologies Setup in the Pedagogical University). Pedahohichnyi dyskurs 9, 258–263 (2011)
20. Russell J.: Rakuten's Viber chat app plans to charge to operate chatbots in controversial move. URL: https://techcrunch.com/2019/02/22/viber-chatbot-charge (2019). Accessed 14 Aug 2019
21. Semerikov S.O., Striuk, M.I., Moiseienko, N.V.: Mobilne navchannia: istoryko-tekhnolohichnyi vymir (Mobile learning: historical and technological dimension). In: Konoval, O.A. (ed.) Teoriia i praktyka orhanizatsii samostiinoi roboty studentiv vyshchykh navchalnykh zakladiv, pp. 188–242. Knyzhkove vydavnytstvo Kyreievskoho, Kryvyi Rih (2012)
22. Wu, T.H., Kuo, R., Chang, M., Heh, J.: Developing a Learning Notification Express Delivery Service for Students based on Petri-Net. In: Sixth IEEE International Conference on Advanced Learning Technologies, Kerkrade, 5-7 July 2006, pp. 425–426. IEEE (2006). doi:10.1109/ICALT.2006.1652462

System for detecting network anomalies using a hybrid of an uncontrolled and controlled neural network

Galina Kirichek[0000-0002-0405-7122], Vladyslav Harkusha[0000-0001-5980-4802], Artur Timenko[0000-0002-7871-4543] and Nataliia Kulykovska[0000-0003-4691-5102]

National University “Zaporizhzhya Polytechnic”,
64, Zhukovsky Str., Zaporizhzhia, 69063, Ukraine
{kirgal08, garkusha2580, artureuro, natalya.gontar}@gmail.com

Abstract. In this article realization method of attacks and anomalies detection with the use of training of ordinary and attacking packages, respectively. The method that was used to teach an attack on is a combination of an uncontrollable and controlled neural network. In an uncontrolled network, attacks are classified in smaller categories, taking into account their features and using the self-organized map. To manage clusters, a neural network based on back-propagation method used. We use PyBrain as the main framework for designing, developing and learning perceptron data. This framework has a sufficient number of solutions and algorithms for training, designing and testing various types of neural networks. Software architecture is presented using a procedural-object approach. Because there is no need to save intermediate result of the program (after learning entire perceptron is stored in the file), all the progress of learning is stored in the normal files on hard disk.

Keywords: neural network, learning, intrusion, anomalies detection, SOM.

1 Introduction

The probability of threats in computer networks increases every year and is a rather serious issue, so the use of intrusion detection technologies is an important issue in providing network and computer security. The process of detecting an attack is implemented as a monitoring of events in the system or computer network, and allows you to determine, with the indicated probability, an intrusion or not.

Modern filters of network traffic, detection systems and counteraction interventions become ever less effective when dealing with large volumes of traffic in high-speed networks and also unsuitable for recognizing new types and methods of attacks on computer systems and networks. Inductive methods provide the opportunity to obtain accurate identification or prediction of various complex processes in the case of short or noisy input data. This is relevant for network traffic recognition based on protocol classification because most of the normal network thread meets the RFC standards set by the developers, and the anomalies most often manifest themselves in non-standard behavior and packet status.

The system of intrusion detection verifies the network traffic that is being investigated for suspicious activity and also alerts the system or system administrator of possible attacks. The main purpose of intrusion detection system is to protect the availability, confidentiality and integrity of critical network information systems. Two main approaches to the system of intrusion detection are used: the detection of abuses and abnormalities [5].

Detection of misuse is based on the description of known dangerous actions. This description is often modeled as a set of rules that are referred to as signature attacks. The Anomaly Detection ID looks for a threat and applies the rules or predefined terms: the normal and abnormal activity of the system. In the future, we use it to detect the difference in threats from the normal system behavior, to monitor the report, or to block the threats when they arise. Different methods of artificial intelligence are used in intrusion detection system (IDS) anomalies, such as machine learning [11; 14], intelligent data analysis, image recognition and neural networks [18].

To identify abnormalities, it is more rational to develop an interactive intrusion detection system than regular rules and programs that work under the normal principle of detecting and responding to anomalies in the network. Therefore, it makes sense to integrate the classical approaches of IDS and approaches to data analysis using neural networks, which is considered a more flexible approach to the analysis and data classification [6; 9; 15].

2 Formulation of problem

The aim of this work is to develop an IDS system prototype based on a hybrid neural network to detect anomalies and threats from the network, based on the principle of self-organizing maps and the error backpropagation of neural network (learning with teacher). The object of research is the implementation process of modules for detecting threats and anomalies in network. The subject is to formulate model and implementation methods of system prototype.

Research has made it possible to determine that the software should perform analysis and separate the usual and dangerous data based on the input data, in this case, on the basis of network packets. But after revealing the dangerous data, he still needs to carry out the classification of the threat type.

The approach to using neural networks (perceptrons) is chosen as the basis for fulfilling the tasks. These are neural networks based on self-organizing maps used to analyze data and to detect ordinary packets on the network when filtering traffic. After analyzing the data in the first neural network, potentially dangerous data is transmitted to the next neural network to detect the threat type based on the reverse error propagation. In this case, both perceptrons need to be trained to distinguish between suspicious packages and types of threats, respectively. The data used in the study of perceptrons is a dataset of the Lincoln Laboratory of Massachusetts University of Technology. This set is designed to evaluate DARPA intrusion detection systems and is considered to be a benchmark for IDS research [1; 7; 17].

In conducting the experiment for classifying network traffic models, in accordance with the taxonomy of the five templates, we use a data set consisting of five classes of packages, which include: ordinary packages, packages for sensing and scanning infrastructure, packets that caused denial of service equipment, packages that have increased user privileges to super user and external threat packages.

3 Software tools and solutions

It was decided to use two interceptors for a more convenient scaling and use software with independent modules. This simplifies not only the ability to scale but also reduces the concentration of responsibility on each of the software modules. That is to teach and arrange two smaller perceptrons is much easier than one big one. Hybrid network approach based on a neural network without a teacher (first module) and a neural network with a teacher (second module) is used when developing a network threats analyzer. As a network without a teacher, use self-organizing maps.

One of the main approaches to solving cluster analysis problems is a self-organizing maps (SOM) [10; 12]. They are adapted for using learning without teacher, that is, without the end result.

The method of back propagation is a learning method that is controlled by the training of artificial neural networks [16]. The purpose of back propagation is to prepare the network to achieve a balance between the ability to respond correctly to the input models used for learning (memorization) and the ability to give intelligent input responses, as in the training.

The process of learning without a teacher in SOM can be briefly described in three stages. In the first stage, weights of the connection are assigned small random numbers and the choice of the speed learning parameter is made.

At the second stage, the best matching block is fixed, with determination of neuron with the greatest weight in the layer of neural network, Euclidean square is used to measure the distance between the input vector and the weight vector, and also the unit chosen whose weight vector has the smallest Euclidean distance from the input vector is selected.

At the last stage, weights are updated according to the rule of training of Kohonen network according to formula (1):

$$\omega_{ij}{}^{\text{new}} = \omega_{ij}{}^{\text{old}} + \alpha(x_i - \omega_{ij}{}^{\text{old}}), \tag{1}$$

where x_i is the i-th input vector, ω_{ij} is the j-th column of the weight matrix, and α, the learning rate, decreases as learning proceeds. Updating neuronal weight in the network occurs only for active output neurons. It is allowed to teach a unit whose weighted vector is closest to the input vector [2].

Learning process itself continues until all input vectors are processed. Criterion of convergence in neural networks is an epoch. This is one iteration in the learning process, which includes representation of all examples from training set, as well as verification of quality training in a controlled set. Epoch determines how many times all input vectors must be submitted to the SOM for learning.

This algorithm is also called the gradient descent algorithm because the strategy of selecting such an important parameter as the weight for each neuron of a multi-layered network is based on the gradient method. Continuous target function as a measure of network success in the general case is defined as quadratic amount difference between an actual result and expected output value. Algorithm for the reverse distribution of a learning error uses two extensions of the network - direct and reverse.

At the very beginning of the algorithm there is a direct passage where input data in the form of a vector implement distribution among the layers, from the original to the last. As a result of direct pass, a set of output signals is generated, which determines response of the network to the input data. During a straight pass, all synaptic weights of network are fixed. The second stage of algorithm is return pass where parameters (all synaptic weights) are adjusted according to the error correction rules. The essence of the rule is as follows: expected output values subtract resulting (resulting) value of an actual output and error signal is generated as a result of such an operation. Error signal extends like an echo in opposite synaptic bonds, so the algorithm got this name. And synaptic scales, in turn, are adapted to maximize expected output of network's output signal.

Scales are adjusted to reduce the error by distributing original error back through the network. Training kit is supplied several times to the network, and the weight values are corrected until overall error exceeds the specified one. Developed system uses identification process of abnormal and normal packets in a computer network. Whole process of system development can be divided into 2 stages. The first is the stage of training in which the SOM neural networks and reverse error propagation have been trained for a certain amount of time (epoch), it is shown in Fig. 1.

Next step is to detect threats or testing yourself. Model of method functioning is shown in Fig. 2. Since the usual packet-transfer analysis operations are specified and they display the expected behavior, we can initiate knowledge-based definitions (improper use), whereas the non-typical packet activity (the invasion is likely to indicate the non-typical behavior of the packet) is constantly being developed and can't be regarded as defined an attack, so identifying IDS abnormalities is performed on attacks.

An uncontrolled neural network based on a self-organizing map (SOM) divides classification of threats into smaller categories, taking into account their similar features, and then, clustering of threats is performed based on the error of nonpropagation of the neural network.

The SOM training is implemented on the basis of data from KDD-99 (knowledge discovery in databases), which is a set of data used during the second international competition on open knowledge and data mining.

Connections in KDD-99 are presented in the form of functions, each of which is located in significantly different ranges, in one of the continuous, discrete, and symbolic forms. Functions in this set are protocol type, service type and respectively. The protocol type value can match tcp, udp or icmp; the service type may be one of the different network services, such as http, smtp, etc.; the checkbox corresponds to one of 11 values, such as SF or S2. Other parameters in these connections are the length of connection; number of bytes of data from the beginning to destination and vice versa;

number of connections to the same end node as the current connection in the last two seconds, etc. The full list of attributes set for the connection records is given in corresponding sources of information [3; 7].

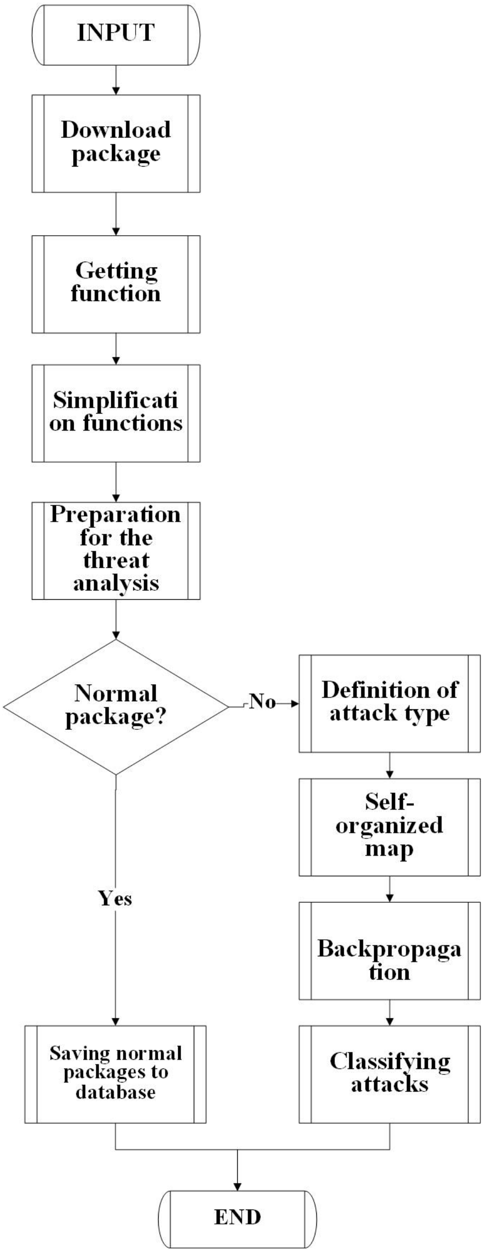

INPUT
Download package
Getting function
Simplificati on functions
Preparation for the threat analysis
Normal package?
No
Definition of attack type
Self-organized map
Backpropaga tion
Classifying attacks
Yes
Saving normal packages to database
END

Fig. 1. Phase training algorithm.

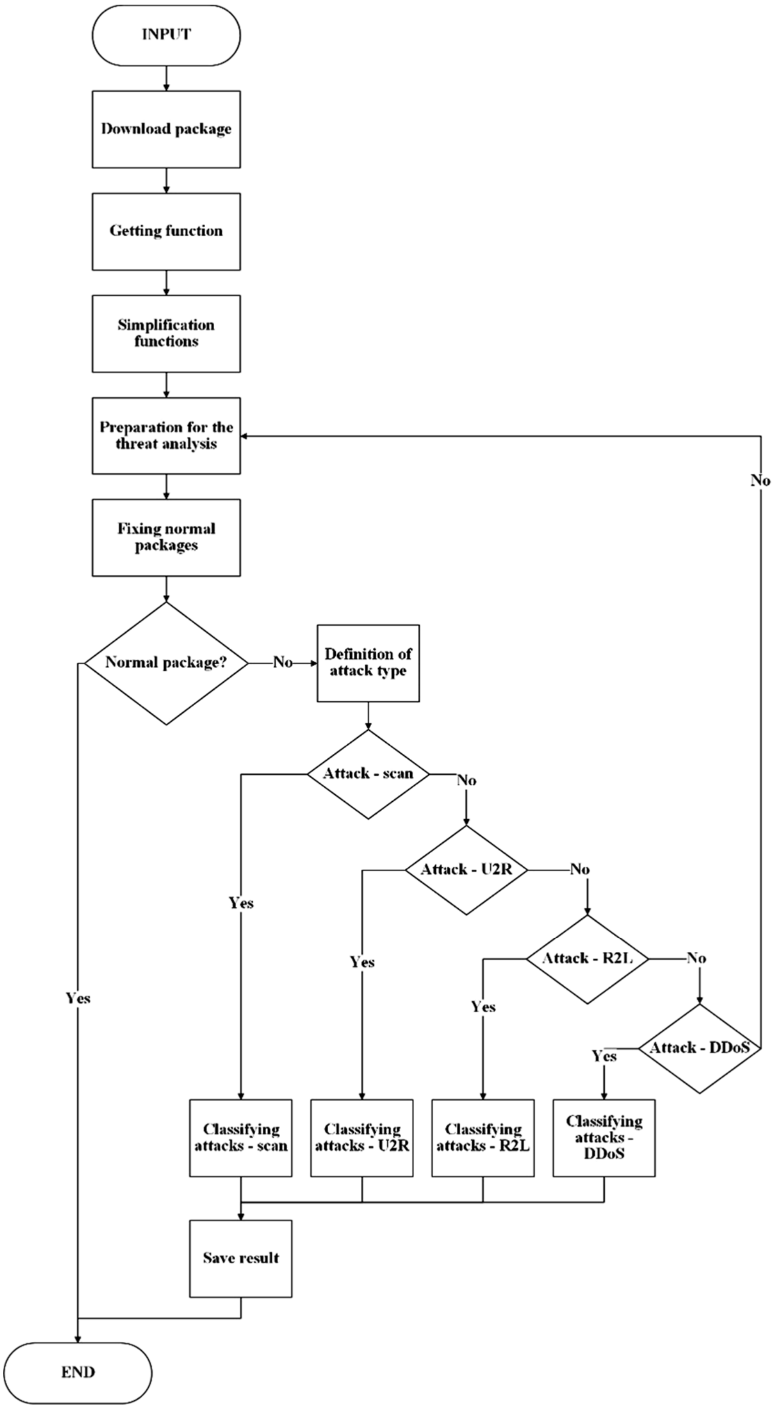


Fig. 2. Detection phase algorithm.

4 System for detecting network anomalies

In the process of designing and system software implementation, authors developed software (utility) that allows you to fulfill the purpose of the work – detection and classification of threats in network traffic (packets).

As a software method, Python has been selected as an interpreted object-oriented programming language that supports module packs and several programming paradigms: object-oriented, procedural, functional, and aspect-oriented [4; 8]. The choice has been influenced by: Python's support of object-oriented approach, simplicity of syntax, and availability of built-in functions and data structures. Also, in this language, a large number of ready-made solutions and documentation in the field of development and training of perceptrons is realized.

We use PyBrain as the main framework for designing, developing and learning perceptron data. This framework has a sufficient number of solutions and algorithms for training, designing and testing various types of neural networks. Software architecture is presented using a procedural-object approach. Because there is no need to save intermediate result of the program (after learning entire perceptron is stored in the file), all the progress of learning is stored in the normal files on hard disk.

We use setuptools as the main software, when creating this utility, to simplify the construction of the main framework of the system.

Program consists of the main function that is called when the program is started and after class initialization [13].

```
from __future__ import __all__
from scipy import random
from scipy.ndimage import minimum_position
from scipy import mgrid, zeros, tile, array, floor, sum
from module import Module

class Kohonen_SOM_Map(Module):
  learn_rate = 0.01
  neighbourd = 0.9999
  outdim, winner, neurons_num, dist_matrix, inputs_num, diff,
neurons, outputFullMap = None, None, None, None, None, None

  def __init__(self, dim, neurons_num, name=None,
output_full_map=False):
    outdim = 2 if output_full_map else neurons_num ** 2
    Module.__init__(self, dim, outdim, name)
    Kohonen_SOM_Map.outputFullMap = output_full_map
    Kohonen_SOM_Map.neurons = random.random((neurons_num,
neurons_num, dim))
    Kohonen_SOM_Map.winner = zeros(2)
    Kohonen_SOM_Map.diff = zeros(self.neurn.shape)
    Kohonen_SOM_Map.inputs_num = dim
    Kohonen_SOM_Map.neurons_num = neurons_num
    Kohonen_SOM_Map.neighbours = neurons_num

    # Init matrix of predicates
    Kohonen_SOM_Map.__dist_matrix_create()

  def _forward_err_implement(self, inbuf, outbuf):
    Kohonen_SOM_Map.diff = Kohonen_SOM_Map.neurons - tile(inbuf,
(Kohonen_SOM_Map.neurons_num, Kohonen_SOM_Map.neurons_num, 1))
    error = sum(Kohonen_SOM_Map.diff ** 2, 2)
    Kohonen_SOM_Map.winner = array(minimum_position(error))
    if not Kohonen_SOM_Map.outputFullMap:
      outbuf[:] = Kohonen_SOM_Map.winner

  @classmethod
  def _backward_err_implement(cls):
    n = floor(cls.neighbours)
    cls.neighbours *= cls.neighbourdecay
    tl = (cls.winner - n)
    br = (cls.winner + n + 1)
    tl[tl < 0] = 0
    br[br > cls.neurons_num + 1] = cls.neurons_num + 1

    # calculate distance matrix
    tempm = 1 - sum(abs(cls.dist_matrix - cls.winner.reshape(1, 1,
2)), 2) / cls.neurons_num
    tempm[tempm < 0] = 0
    distm = zeros((cls.neurons_num, cls.neurons_num, cls.nInput))
    for i in range(cls.nInput):
      distm[:, :, i] = tempm
      distm[:, :, i] = tempm
    cls.neurons[tl[0]:br[0], tl[1]:br[1]] -= cls.learningrate *
cls.diff[tl[0]:br[0], tl[1]:br[1]] * distm[tl[0]:br[0],
tl[1]:br[1]]

  @classmethod
  def __dist_matrix_create(cls):
    if not cls.neurons_num:
      print ("Kohonen_map: not setted neural layers")
    distx, disty = mgrid[0:cls.neurons_num, 0:cls.neurons_num]
    cls.dist_matrix = zeros((cls.neurons_num, cls.neurons_num, 2))
    cls.dist_matrix[:, :, 0] = distx
    cls.dist_matrix[:, :, 1] = disty
```

The KohonenMap _forward_err_implement method assigns one of the neurons to input dates in the input buffer and fixed coordinates of neurons in the output buffer, and also performs calculation of the largest neuron, calculating data with the slightest error using square of difference.

The KohonenMap _backward_err_implement method training the Kohonen map in an uncontrolled mode, moving the closest neuron and neurons adjacent to it closer to the input template [4; 13].

The main function performs initialization of an instance of the class, namely, it creates the KohonenMap object and assigns variable to given object. After that, in the input buffer, training data is asked in order to conduct training of this object. Learning result is stored on the hard disk after training for several cycles.

The obtained results confirm that the quality of the classification of packages depends on the number of standards of separate classes in the educational voter. If the number is small, then the detection rate of the attacks is high and the number of detected intrusions by class is improved. This indicates that the method works in real-time with high performance.

5 Conclusion

The purpose of this work is to develop a hybrid neural network (perceptron) based on 2 other neural networks, namely, the Kohonen neural network and the neural network with back propagation. Data set from the Lincoln Laboratories of Massachusetts Technology University from United Stateswas used as learning data sets. This data set includes type of package, its useful data and metadata. The Python language and PyBrain framework are selected as the software component.

When developing software based on the idea of hybridization of neural networks, the problem was solved to ensure protection of internal network from external threats using packet filtering for threats such as denial of service and unauthorized increase of user privileges. Effectiveness of methods to protect computer networks from harmful traffic has been increased using prior analysis of packets risk. Also, in this neural network, the so called boosting is applied - an increase in the efficiency of the neural network at the expense of another neural network, which delivers already filtered information to the inputs.

In the following, the possibility of using this software on operating system for such routers as OpenWrt is considered. This integration will not only increase an efficiency of this system while protecting the network, but also will increase an accuracy of the perceptron through the adoption of a large number of network traffic with self-study of neural networks.

References

1. Akbar, S., Rao, K.N., Chandulal, J.A.: Intrusion detection system methodologies based on data analysis. International Journal of Computer Applications **5**(2), 10–20 (2010). doi:10.5120/892-1266

2. Bahrololum, M., Salahi, E., Khaleghi, M.: An improved intrusion detection technique based on two strategies using decision tree and neural network. Journal of Convergence Information Technology **4**(4), 96–101 (2009)
3. Chebrolu, S., Abraham, A., Thomas, J.P.: Feature deduction and ensemble design of intrusion detection systems. Computers & Security **24**(4), 295–307 (2005). doi:10.1016/j.cose.2004.09.008
4. Dierbach Ch.: Python as a first programming language. Journal of Computing Sciences in Colleges **29**(6), 153–154 (2014)
5. García-Teodoro, P., Díaz-Verdejoa, J., Maciá-Fernández, G., Vázquez, E.: Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security **28**(1–2), 18–28 (2009). doi:10.1016/j.cose.2008.08.003
6. Imamverdiyev, Y.N., Sukhostat, L.V.: Obnaruzhenie anomalii v setevom trafike na osnove informativnykh priznakov (Network traffic anomalies detection based on informative features). Radio electronics, computer science, control 3, 113–120 (2017) doi:10.15588/1607-3274-2017-3-13
7. KDD Cup 1998 Data. http://kdd.ics.uci.edu//databases/kddcup98/kddcup98.html (1999). Accessed 21 Mar 2019
8. Kirichek, G., Kurai, V.: Implementation quadtree method for comparison of images. In: 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), pp. 129–132. IEEE (2018) doi:10.1109/TCSET.2018.8336171
9. Kirichek, G., Tymoshenko, V., Rudkovskyi, O., Hrushko, S.: Decentralized System for Run Services. CEUR Workshop Proceedings **2353**, 860–872 (2019)
10. Kohonen, T.: Self-Organizing Maps. Springer-Verlag, Berlin, Heidelberg (2001). doi:10.1007/978-3-642-56927-2
11. Mukkamala, S., Janoski, G., Sung, A.: Intrusion detection using neural networks and support vector machines. In: Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02. Honolulu, HI, USA, pp. 1702–1707 (2002). doi:10.1109/IJCNN.2002.1007774
12. Ritter, H., Martinetz, T., Schulten, K., Barsky, D., Tesch, M., Kates, R.: Neural Computation and Self-Organizing Maps: An Introduction. Addison-Wesley, Reading (1992)
13. Rueckstiess T.: Python PyBrain package v0.3, pybrain.structure.modules.kohonen module source code :: PyDoc.net. http://pydoc.net/PyBrain/0.3/pybrain.structure.modules.kohonen (2009). Accessed 17 Aug 2019
14. Sabhnani, M., Serpen, G.: Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset within Misuse Detection Context. In: Proceedings of the International Conference on Machine Learning; Models, Technologies and Applications. MLMTA'03, June 23–26, 2003, Las Vegas, Nevada, USA, pp. 209–215. CSREA Press (2003)
15. Salnik, S.V., Salnyk, V.V., Symonenko, O.A., Sova, O.Ya.: Metod vyiavlennia vtorhnen v mobilni radiomerezhi na osnovi neironnykh merezh (Method of intrusion detection in mobile radio networks on the basis of neurals networks). Science and Technology the Air Force of Ukraine 4(21), 82–90 (2015)
16. Semerikov, S.O., Teplytskyi, I.O., Yechkalo, Yu.V., Kiv, A.E.: Computer Simulation of Neural Networks Using Spreadsheets: The Dawn of the Age of Camelot. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 122–147. http://ceur-ws.org/Vol-2257/paper14.pdf (2018). Accessed 30 Nov 2018

17. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp. 1–6. IEEE (2009). doi:10.1109/CISDA.2009.5356528
18. Zhang, Z., Manikopoulos, C.: Neural networks in statistical anomaly intrusion detection. Neural network world **11**(3), 305–316 (2001)

Modeling university environment: means and applications for university education

Ruslan Cherniavskyi, Yaroslav Krainyk[0000-0002-7924-3878] and Anzhela Boiko[0000-0002-3449-0453]

Petro Mohyla Black Sea National University, 68, Desantnykiv Str., Mykolaiv, 54000, Ukraine
cherniavskyj@gmail.com, yaroslav.krainyk@chmnu.edu.ua, anzhela.boiko@chmnu.edu.ua

Abstract. In the paper, we establish an investigation on the development of university 3D-model and its possible applications for educational and research fields. We assume that 3D-model of university can help in various scenarios and should be used to adopt modern immersing technologies into to university processes. Different means are employed for the development of the model. Bottom-up approach for using these means and their connection with each other are shown in the work. Then, details of the 3D-model design process are provided with peculiarities related to the university building location and corpuses positions. Finally, assembled models of university are shown in 3ds Max and Unity environments. In the final part of the paper, we suggest scenarios of model usage for educational and research fields. Universities can gain various benefits from integrating their research efforts to employ new technology and identify new development opportunities for both science and education in university. In case of the developed 3D-model, it is planned to use it in the projects connected with client-server applications, Internet-of-Things, Smart Grid, etc. In the educational process it will be a part of case-studies for learning 3D-modeling, development in Unity environment, training for emergency situations.

Keywords: 3D-model, university, education, technology, development.

1 Introduction

Usage of modern information-communication technologies (ICT) is a crucial point for a successful university education system. ICT present in literally every field of life and knowledge and effective work with ICT is becoming more and more valuable. Students and enrollee are attracted by possibility to gain new knowledge about new technologies and their practical applications. Nowadays, they are surrounded by new technical means and those means are highly involved in the lifestyle. For instance, smartphone has become indispensable part of communication. However, when it comes to education process, those devices are not employed up to their full potential. We emphasize the following reasons to explain that condition:

1. Velocity of technology improvement process is much higher than capability of adaptation to new challenges in educational environment. By the time when technology is implemented for education process, it might be outdated already.
2. Deficiency of technical means in educational institutions.
3. Lack of systematic approach to this issue. No framework has been developed to join education system with rising technologies as they appear.

Therefore, we can conclude that to cope with the problem educational institutions and specifically universities should not wait for opportunity to integrate new technology. They have to establish general idea of the development in form of framework and integrate immersing technical achievements into educational process fluently.

In this work, we share experience of modeling university environment of Petro Mohyla Black Sea National University in the Computer-Aided Design (CAD) software. The developed 3D-model can be used in different scenarios and in tight cooperation with other modern technologies.

2 Review of scientific sources

In general, digitalization of educational process is one of the main trends in educational process nowadays, especially in higher education [6]. Both students and lecturers are ready to use modern technologies and devices for the classes. Moreover, one of the peculiarities is that students are prepared better for the technologies usage on a basic level than lecturers are. Virtual Reality (VR) and Augmented Reality (AR) can be considered as the most promising technologies for education. They provide advanced learning experience have almost unlimited possibilities for application. For instance, VR can be used for development of virtual labs [3], while AR is an effective tool for enhancing existing learning materials [12]. However, AR is still an adequate choice for development of new courses. These technologies provide gamification of education [11] that increases students' engagement into the process [6; 11; 13].

From the research perspective, convergence and synergy of multiple emerging trends are very prominent [9]. They can be used together to provide benefits to all involved directions. In this work, we consider how 3D-model of university can be used in research topics such as IoT [7], AR [12; 13], Smart Grids, etc. Publications [15] aimed to develop the generalized approach for AR infrastructure for educational purposes. However, this work established general solution while peculiar cases might have significant deviation from the common software architecture. At the same time, subject-specific educational software [8] does not suppose possibility of extension to other fields. To conclude this part of the review, it can be stated that AR-application that exploits 3D-model of university building requires integration with software architecture to execute target functions.

Let us consider in more details main environment for bringing new functions to 3D-model of university. Unity [14] is a state-of-the-art toolkit for the task of 3D-modelling and game development. It provides high-level programming interface to developers and reduces time to release of the final version of the product. In comparison with native interfaces for specific platforms, fewer code instructions are necessary to achieve desired functionality. Unity supports deployment to multiple platforms. Thus, designs based on Unity automatically gain advantage of being able to work on various devices with different operating systems like Android, Windows, and MacOS. The list is not limited to the mentioned platforms. Developers can switch target platform and easily get application for new device without changing software side. However, the main drawback of such universality is high-resource usage of the device. Output file with compiled program code usually exploits much more device's resources, especially disk storage capacity.

Unity includes environment for modeling scene in 3D-view. You can import models into scene editor and combine them according to the needed location, positions of light sources, etc. However, it is not a Computer-Aided Design (CAD) program and it is more convenient to use other software tools for model development. We consider tools that assisted us during development stage in the main part.

The last but certainly not least to mention feature of Unity is cohesion with other modern technologies like VR and AR. The first one supposes that person wears special equipment (glasses or helmet) with integrated means for displaying virtual image and can interact with virtual environment via gestures. VR exploits the idea of creating new reality with clear margin between existing and virtual objects. In contrary, AR is trying to provide new experience by augmenting real world. Objects of AR "appear" in our own environment and interact with it. For now, AR requires markers to be present on the scene and when device recognized markers, it brings new objects into the scene. Regarding connection VR and AR with Unity, both technologies can use models created in Unity. Moreover, Unity supports inclusion of a few AR-libraries so AR-features can be designed directly from Unity infrastructure. As VR does not depend on technology behind the curtain, any modeling tools can be used for scene generation. However, Unity is a mainstream technology and this is the reason why VR-projects include Unity as a basic instrument for modeling.

Many universities develop 3D-models of their building and location or panorama views. Those designs give students the opportunity to be acquainted with university space, find necessary places, etc. However, the idea behind the model of university can have much more practical applications. It may be a complex platform for studying process, research environment, and investigation of contemporary technologies. But university building relates to the complex structures [5] with multiple details and developed model of university should be relevant to employ it in all the related activities.

3 Results

3D-model of university is a complex composition of multiple objects with different properties. Some of them comprise large number of nodes in space (e.g. university buildings) and other might be quite simple for design (e.g. part of interior design). During designing model of the university, we used bottom-up approach, e.i. start from low-complexity parts and then add more details, improve quality of each model, etc. Eventually, this approach leads to the state when design satisfies expected criteria (dimensions consistency, texture selection, number of polygons for object detalization, etc.).

Bottom-up approach supposes that a combination of different tools is harnessed for design process. We can distinguish next steps for retrieving 3D-model:

1. Design of low-polygonal models.
2. Improvement of models created at first step by adding additional polygons in their representation.
3. Texture application and editing.
4. Preparation of light sources.
5. View cameras setup and adjustment.

The following software products have been used during development of the university 3D-model:

- Blender [4];
- SketchUp [1];
- 3ds Max [2].

The selection of the aforementioned software tools is explained by the availability of the extensive instrument set for 3D-model design with employment of different techniques. Moreover, the software tools are available under terms of free license for educational institutions.

SketchUp is a lightweight software for designing of 3D-models. It is available in form of native application and can be used in web-browser. Because SketchUp is suitable for fast design of low-polygonal models, this software has been used for the university model. The basic models were created in SketchUp and then they have been improved in other modeling environments.

Blender is another instrument to work with 3D-models. Its main purpose is creation of 3D-objects animations. The great advantage of Blender is that it has a free license and can be used even in commercial projects.

3ds Max is probably one of the most advanced instruments when it comes to modeling complex 3D-objects. During development of the university model, 3ds Max was employed to enhance detalization level of the models from the previous step. Autodesk provides special free educational license for university for three years.

3ds Max was also involved in the work with texture. It contains material editor that facilitates wide variety of configuration for the work with texture objects. To create as much realistic model as it is possible, textures have to be obtained first. They have been collected as storage of photos made around and inside the university. Only photos of the most important and visually notable objects were taken.

Because models are stored in interchangeable format, output from one software can be used as an input model for other one. Thus, model can pass through different instruments before appearing in the final scene. It may be considered as a software pipeline for 3D-models that aims to improve quality of object with each next step or to bring new functions to available models. Fig. 1 demonstrates interconnections between software tools in terms of model processing.

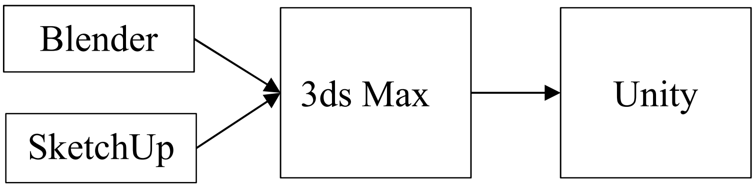


Fig. 1. Structural flow interconnection between modeling tools

The design of university 3D-model started from information available at the photos and blueprints of the university buildings. In general, main university building consists of four university building. At the location where university placed, slope can observed clearly. This fact produces one of the main difficulties for designing complex 3D-model. All components of the model should be in conformity state with each other to ensure that geometrical proportions match in joint points. Another complexity issue is concerned with architectural solutions for the building. Different floors of the university has different geometrical parameters and they vary significantly. Thus, we cannot just use cloning operation to obtain model of all stores in the corpus. Moreover, constructions in the main building have different number of floors and adjacent buildings are not equal by this parameter. Some of the photos that demonstrate outside facade of the university and internal space surrounded by the university are shown in Fig. 2.

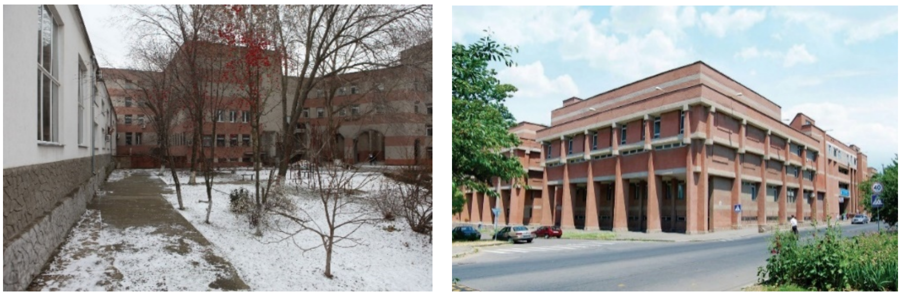


Fig. 2. Outer view and internal space of the university

As blueprints contained almost no information about height parameters and proportions of the inter-corpus connection, they have been reproduced from photo materials and direct measurements obtained using a laser range finder.

Basic low-polygonal models have been developed in SketchUp modeling environment. Different views of university corpuses are illustrated in Fig. 3. This way outer frame of the corpuses has been visualized. From this moment, 3D-model is ready for further detalization and enhancement.

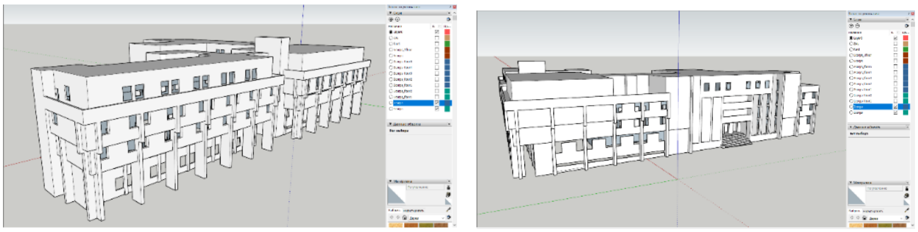


Fig. 3. Modeling university in SketchUp

From the Fig. 3, you also can see that model is described only using straight line and can be composed of as a union of rectangles. With outer frame, we can move on to the next stage that is design of internal parts. We modeled each floor separately. Afterward, all designed floors were embedded into the outer skeleton. The preview of combined floors models is shown in Fig. 4.

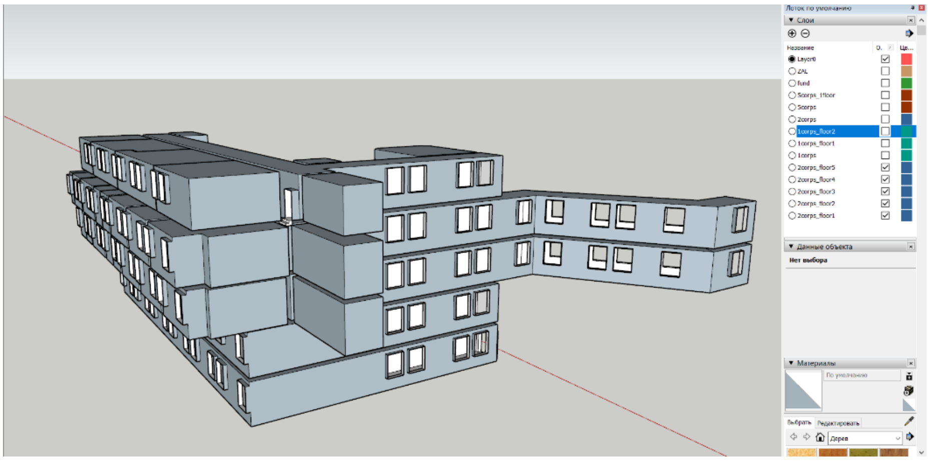


Fig. 4. Models of stories of the university building

It demonstrates floor models for one of the university buildings. You can visually notice that floors have different level of extension to the outer side. Additionally, two of the floors shown in Fig. have connection with other university structures. Stories of each building have been modeled individually because it makes construction modular and easier to change. However, this approach conceals a problem of consistency. Elements of the stories from different buildings have to be connected at the same level and no geometrical forms of connection should infer appearance of artifacts in the model (obstacles in passages, disconnected surfaces, etc.).

At the next stage, prepared 3D-models are exported to the environment with wider instrument set. In this case, we perform export to 3ds Max 2019. With export done, we can perform rendering of the model from a different point of views to retrieve high-quality image of the model. One of the rendering results is demonstrated in Fig. 5.

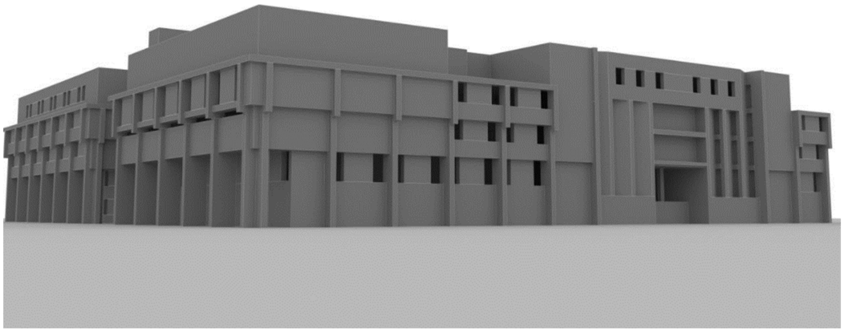
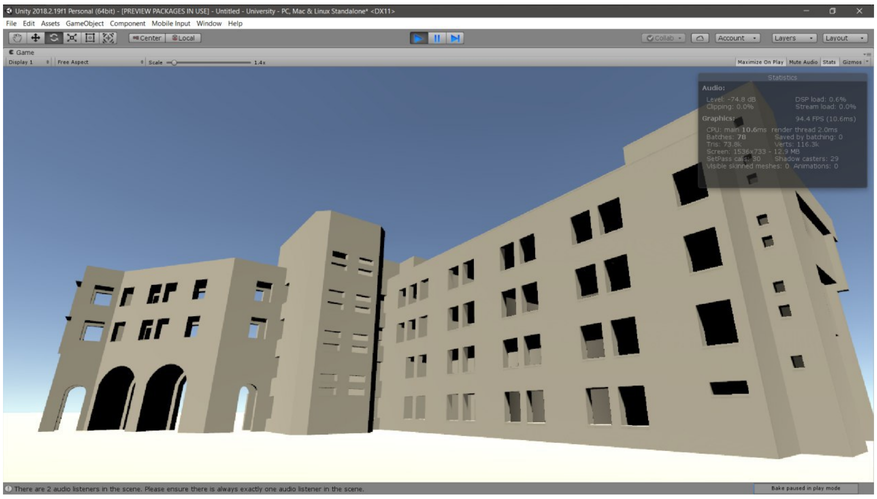


Fig. 5. Rendering results of university model in 3ds Max

At the next stage, we perform texture overlay and set up light sources for rendering of the ready 3D-model. However, to achieve high-quality results and to build final model with texture as maximum close to the original as it is possible, we need to collect those textures and apply them in the modelling environment. The results of this stage will be available in the next publications on the topic.

Although, the model does not contain more details and full set of textures, it is ready for usage in development of the programming projects. We can import assembled model into Unity environment and organize preview using embedded functionality of Unity editor as shown in Fig. 6.



Unity 2018.2.19f1 Personal (64bit) - [PREVIEW PACKAGES IN USE] - Untitled - University - PC, Mac & Linux Standalone* <DX11>

```
Statistics
Audio:
  Level: -74.8 dB              DSP load: 0.6%
  Clipping: 0.0%               Stream load: 0.0%
Graphics:                      94.4 FPS (10.6ms)
  CPU: main 10.6ms  render thread 2.0ms
  Batches: 78          Saved by batching: 0
  Tris: 73.8k          Verts: 116.3k
  Screen: 1536x733 - 12.9 MB
  SetPass calls: 30    Shadow casters: 29
  Visible skinned meshes: 0  Animations: 0
```

There are 2 audio listeners in the scene. Please ensure there is always exactly one audio listener in the scene.

Bake paused in play mode

Fig. 6. Representation of developed university model in the Unity environment

As the model, has been imported into Unity, we can engage it into development of complex systems with combination of different technologies. Unity environment assists a lot in the software products development that includes 3D-model. It conveniently separates visual presentation of the model and model's behavior that is described in controller using programming language (C# or JavaScript). Unity features all conventional instruments (camera view, textures, light sources, model transformations, etc.) for programming 3D-models. However, it provides additional level of abstraction that results in reduced amount of code developer needs to write. It also provide notable advantage for testing because its integrated environment allows running developed solution immediately on the developer's machine.

Let us consider further how model of university can be used with other trending technologies for practical projects and researches. Internet-of-Things (IoT) [10] is very promising conception that supposes very high level of integration for modern devices to the global network. IoT is not limited by any specific application sphere and can be used in almost any activity. In context of 3D-model of university, it can be a complex solution that unites sensor network, network technologies, application server, and client application with support of 3D-graphics. Nodes in the network collect information from sensors and send measurements to the server. Server processes information and notifies connected clients about changes in state of the sensor system. This way client in real time can monitor microclimate in the classroom, identify presence of students, state of the equipment, etc. User can instantly "move" to the necessary place in application and react to the notification. Actions available to user depends on peculiarities of the system.

For instance, we consider Smart Energy Grid (SEG) technologies. First of all, university have to occupy the role of prosumer (producer and consumer of electricity at the same time). Having large areas and huge demands for electricity resources, it makes sense to deploy at least basic photovoltaic (PV) infrastructure. If solar panels are installed on moving platform, operator can adjust their position in real-time using interface of 3D-environment. Consequently, it will result in higher level of generated energy and reduction of payment bill. Another example of electricity management is automatic light control system. If presence of activity is detected in the classroom, the light can automatically turn on. In opposite, if no activity is detected, it goes off. In this case, network node does not need to send all acquired measurements to the server but can just notify server when state of the light is changed.

Another instance of 3D-model use is an interactive 3D-application. The application is responsible for visualization, navigation, and interaction with 3D-model. Thus, it is possible to perform a virtual tour over university. This feature can be employed for solving problem of the positioning in space, finding the route to the target point, and acquiring information about the object. Due to the high level of realism of the developed model, an effect of immersion can be achieved.

Considering applications of the 3D-model in educational process, we can state that they are very promising.

First, the model can be used in the courses connected with development of AR and VR software. Ready-to-use model significantly simplifies development process and allows formulation of the tasks. One thing all applications will share in common is active movement inside the modeling building. Thus, it is suitable for teaching concepts of camera in 3D-environment and its correct adjustment. An example application that employs that functionality is interactive guide for university environment where user moves over the model and gets hints on what is the purpose of each room, who works inside, working hours, etc. Eventually, some part of electronic documentation might be moved into such application. However, this scenario does not limit possible practical benefit of 3D-model for programming purposes. In fact, many programming tasks are becoming more comprehensible and interesting for students when they are connected with programming 3D-models.

Second, 3D-model of university can be effectively used for emergency training. For instance, preliminary training on the actions in case of fire situations can be held in 3D-environment. Various cases are available for such training to guarantee complex approach to the problem. Tested person verifies his/her action in the virtual environment and gets more acquainted with possible critical situations. It helps a lot when it comes to real training or even critical situation because person has visual experience and recalls recommended plan of actions faster.

Third, the final 3D-model and its by-products are actually a good point for learning CAD-tools with emphasis on 3D-tools. They serve as an example of successful design and can be used for further improvement of the model. Tasks related to these disciplines include modeling of interior environment (view of the classrooms), perspective planning for further renovation of the building and evaluation of the proposals for this purpose. They can be visually evaluated and feedback for the project will be given much faster than without visual representation.

Concluding this part, we can allege that proposed use cases do not form a full list of outcomes for research and educational fields of activities. Moreover, new applications should be found as modern technologies appear.

4 Conclusions

In this paper, we provide results of the development of Petro Mohyla Black Sea National University 3D-model. Four main software tools were occupied in the process. Bottom-up approach has been used for 3D-model design where outputs from one application served as inputs for another. It allowed improve detalization level systematically and combine simple models into complex ones. The designed model can be used in a set of various cases for educational and scientific processes. It can be especially effective solutions with rich visual experience and has many perspectives for further extensions.

References

1. 3D Design Software | 3D Modeling on the Web | SketchUp. Trimble Inc. https://www.sketchup.com (2019). Accessed 21 Mar 2019
2. 3ds Max | 3D Modeling, Animation & Rendering Software | Autodesk. Autodesk Inc. https://www.autodesk.com/products/3ds-max/overview (2019). Accessed 21 Mar 2019
3. Aljuhani, K., Sonbul, M., Althabiti, M., Meccawy, M.: Creating a Virtual Science Lab (VSL): the adoption of virtual labs in Saudi schools. Smart Learning Environments **5**, 16 (2018). doi:10.1186/s40561-018-0067-9

4. blender.org - Home of the Blender project - Free and Open 3D Creation Software. https://www.blender.org (2019). Accessed 5 Apr 2019
5. Boiko, A., Bondarenko O., Davydenko Y.: Hull Parametric Modeling of a Small Waterplane Area Twin Hull ships. In: Proceedings of 2019 IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), Polyana, Ukraine. IEEE (2019). doi:10.1109/CADSM.2019.8779352
6. Bond, M., Marín, V.I., Dolch, C. Bedenlier, S., Zawacki-Richter, O.: Digital transformation in German higher education: student and teacher perceptions and usage of digital media. International Journal of Educational Technology in Higher Education **15**, 48 (2018). doi:10.1186/s41239-018-0130-1
7. Duda, O.M., Kunanets, N.E., Matsiuk, O.V., Pasichnyk, V.V.: Information-Communication Technologies of IoT in the “Smart Cities” Projects. CEUR Workshop Proceedings **2105**, 317–330 (2018)
8. Hruntova, T.V., Yechkalo, Yu.V., Striuk, A.M., Pikilnyak, A.V.: Augmented reality tools in physics training at higher technical educational institutions. CEUR Workshop Proceedings **2257**, 33–40 (2018)
9. Kiv, A.E., Soloviev, V.N., Semerikov, S.O.: CTE 2018 – How cloud technologies continues to transform education. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018), Kryvyi Rih, Ukraine, December 21, 2018. CEUR Workshop Proceedings **2433**, 1–19. http://ceur-ws.org/Vol-2433/paper00.pdf (2019). Accessed 10 Sep 2019
10. Krainyk Y., Razzhyvin A., Bondarenko O., Simakova I.: Internet-of-Things Device Set Configuration for Connection to Wireless Local Area Network. CEUR Workshop Proceedings **2353**, 885–896 (2019)
11. Leftheriotis, I., Giannakos, M.N., Jaccheri, L.: Gamifying informal learning activities using interactive displays: an empirical investigation of students’ learning and engagement. Smart Learning Environments **4**, 2 (2017). doi:10.1186/s40561-017-0041-y
12. Lytridis, C., Tsinakos, A.: Evaluation of the ARTutor augmented reality educational platform in tertiary education. Smart Learning Environments **5**, 6 (2018). doi:10.1186/s40561-018-0058-x
13. Merzlykin, O., Topolova, A., Tron, V.: Developing of Key Competencies by Means of Augmented Reality in Science and Language Integrated Learning. CEUR Workshop Proceedings **2105**, 465–468 (2018)
14. Real-time tools for 3D, AR & VR development | Products | UnityUnity. https://unity.com/products (2019). Accessed 21 Mar 2019
15. Syrovatskyi, O.V., Semerikov, S.O., Modlo, Ye.O., Yechkalo, Yu.V., Zelinska, S.O.: Augmented reality software design for educational purposes. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 1st Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2018), Kryvyi Rih, Ukraine, November 30, 2018. CEUR Workshop Proceedings **2292**, 193–225. http://ceur-ws.org/Vol-2292/paper20.pdf (2018). Accessed 17 Aug 2019

Development a computer network user support tool

Nadiia Balyk[0000-0002-3121-7005], Vasyl Oleksiuk[0000-0003-2206-8447]
and Anatolii Halas[0000-0003-0524-4357]

Ternopil Volodymyr Hnatiuk National Pedagogical University,
2, Maxyma Kryvonosa Str., Ternopil, 46027, Ukraine
{nadbal, oleksyuk, galas_av}@fizmat.tnpu.edu.ua

Abstract. The article explores the issues of designing and developing a support system for corporate network users. The authors analyzed existing user support systems, including the concepts of ServiceDesk and HelpDesk. The study describes the process of designing and developing the software "Network user support tool" (NetSupport). This tool stores data in the Google Calendar Cloud Service. The results of the study are: analysis of the functioning of the Google Calendar service, review of the API for working with Google cloud services, development and testing of software to support the work of users. Functionality of Network user support tool allows you to create events on your system administrator or support teams` calendar with a detailed description of the user problem. The article describes the process of developing a network support tool using C# 6.0 programming language and .NET 4.6 technologies. The developed software has been tested in Ternopil Volodymyr Hnatiuk National Pedagogical University.

Keywords: network support, cloud technologies, software development, Google Calendar, HelpDesk, ServiceDesk.

1 Introduction

The development of computer networks has been going on for many years. Today, many computers are integrated into corporate networks. User support on large networks is performed by several system administrators. They form a support teams. Hardware and software issues and staff errors cause a large flow of support requests.

Therefore, there is a need to develop software for scheduling technical support processes. The relevance of developing the program "NetSupport" is to provide operational communications users and system administrators.

The purpose of this article is to develop a software product to support the work of users of the corporate network.

The main tasks of the research are:

- review scientific, technical literature on approaches to support for corporate network users problem;
- analyze available software to support user activity;

- explore advanced cloud technology, including the Google Calendar API (Application Programming Interface);
- design and develop a support system for corporate network users;
- perform testing of the developed tool.

To solve these problems, we used a set of theoretical and empirical methods such as analysis of scientific, technical literature, generalization, modeling of processes of obtaining, storage and processing of data in the investigated systems, observations, analysis of experience of using different systems in the IT infrastructure of the corporate network, modeling method and programming.

2 Analysis of the basic concepts of the study

A corporate network is a computer network that is geographically distributed, ie it brings together different offices, divisions, and the like. The principles underpinning the corporate network are somewhat different from those used to create a separate LAN (Local Area Network) [3]. For the proper operation of the network, a technical support department is required to ensure the correct operation and proper support of the corporate network.

Technical support is a service structure for troubleshooting hardware and software problems, fixing mistakes users. Today to support the work of users use approach from Information Technology Infrastructure Library (ITIL). ITIL is a library for identifying problematic areas of the IT infrastructure, assessing the effectiveness of the IT unit [2], [9].

One of the main processes described by ITIL is incident management. The main purpose of incident management is to eliminate as quickly as possible problems in IT infrastructure – accidents, hardware problems, etc. To implement the process in the organization and create a special department that contacts with employees and coordinates the elimination of problems with the IT unit. This department may be called the Service Desk or the Help Desk [1].

The goal of Service Desk is to register user requests, provide them with the assistance they need, and engage IT staff to resolve problems as quickly as possible. Additionally, this service analyzes the statistics of incidents and the time to resolve them. This is necessary to evaluate and improve the quality of IT services. The Service Desk has the advantage of identifying problems and reducing the cost of sub-services. Such systems can usually collect statistics about PC performance. Help desk is a narrower concept, it is a tool for technical user support. The Helpdesk concept requires a technical support department [8].

Let’s look at some software tools that implement the Service and Helpdesk concept. Article [4] discusses software development to minimize time costs and increase productivity in the area of communication services.

One of the customer support systems is commercial software Zingtree [10]. The system provides customer support. It provides the ability to automate the processing of user requests. With scripts, system administrator can reduce the client’s waiting time for a problem solving. Zingtree further enhances security – large enterprises deploy

Zingtree for added control and security. Zingtree helps solve complex decision-making processes that usually require many steps. Users work with the expert system and populate the decision tree. They answer the questions and the system displays the necessary information in a few steps. To create a decision tree, the user does not need to know programming. It is enough to draw a flow chart or answer questions. As a result, the Zingtree designer system generates the response code itself.

Another HelpDesk system is Dezide. It proposes to create intelligent, interactive maps to solve problems. Skilled workers can create problem maps themselves. In Builder, you can add causes, troubleshoots, and post solutions [6]. The main idea of the program is that if other users encounter such a problem, then they will be able to find a solution in the database.

The system also allows reduce average computer repair time, standardize troubleshooting and maintenance processes, and provide feedback to support engineers. It is possible to connect the system to CRM (Customer Relationship Management), ERP (Enterprise Resource Planning) and CMS (Content Management System).

3 Results

3.1 Designing a computer network user support tool

The concept of the network support utility is based on the provisions such as

1. We will use the Google Calendar cloud service to notify Customer Support about the user's issues.
2. Determine which IT specialist will perform the task based on the user's choice of problem category.
3. The application settings must be secure. Only the system administrator can change the settings.

Students and teachers work in the academic classroom of the university. The university network consists of several hundred stationary computers, which are geographically located in several buildings. Therefore, it is not possible to have a competent system administrator in every classroom. That is why on our personal computers the network user support tool developed by us works.

After the user has filled out and submitted the form, the data is transferred to Google Calendar. Each user hit creates a new calendar event. G Suite has a shared calendar for support team. Accordingly, all devices that have their respective accounts added will be notified of the new event.

Google Calendar is an effective service for organizing activity. For example, the service can be used to schedule classes [5]. Using Google Calendar as a specific database, we can collect and analyze statistics Based on the data obtained, the engineer will be able to anticipate certain problems. The system administrator can pre-order the right parts. Statistics can also identify the types of problems and classrooms they may have.

The applying of network user support tool model will provide the functions such as

- receiving data about a PC problem (IP address, request creation date, error description, feedback email, request priority, classroom selection, and the ability to attach a file to explain the error);
- processing of the entered information and submitting it in a structured format in Google Calendar (the event name contains the classroom in which the request was made, the field "event location" – the IP address of the problem PC, the color of the event determines priority, "attachment" contains the user's file, the date is determined automatically, the event description contains the rest of the user's information);
- entering the processed information into the notification system;
- processing and executing the request by the system administrator.

The calendar is a collection of events. They include additional metadata such as description, time zone, placement, and more. Each calendar has its own ID. It corresponds to the email address of one or more calendar owners.

An event is an object associated with a specific date or time interval. Events are indicated by an identifier that is unique to the calendar. They contain basic information such as date, start time and end time, description, location, status, reminders, attached files. Events can be single or recurring. They can occur between two specific times or span a series of consecutive days.

An event always has one organizer who created the calendar. It contains the main copy of the event. Event can also have multiple participants. The participant is usually sent an invitation from the primary calendar. Each user works with own calendar. It can also view or create events on other calendars it has access to. Users can create events and invite other users (see Fig. 2).

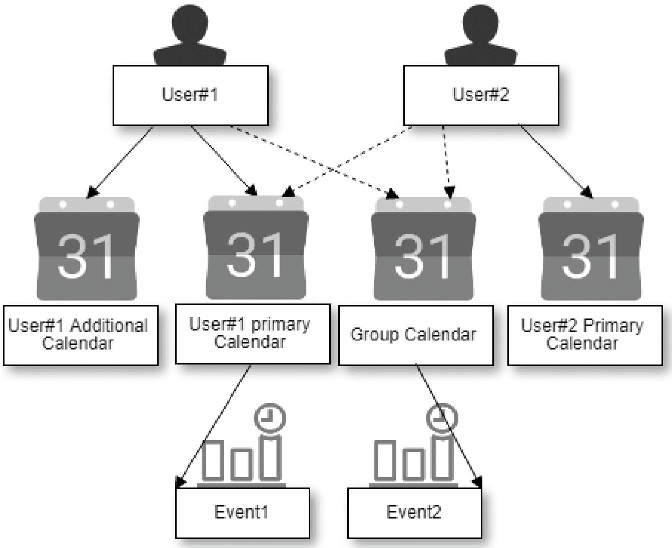


Fig. 1. The concept of user access to calendars and events

Figure 1 shows

- User#1's calendar contains the primary calendar, the group calendar and the additional calendar;
- User#2's calendar contains the main calendar; it also has access to the group calendar and the primary calendar of User#1;
- User#1 created Event1, so he can take any action with it, User#2 Event1 is view-only.

Figure 2 contains the concept of user interaction network user support tool through Google Calendar.

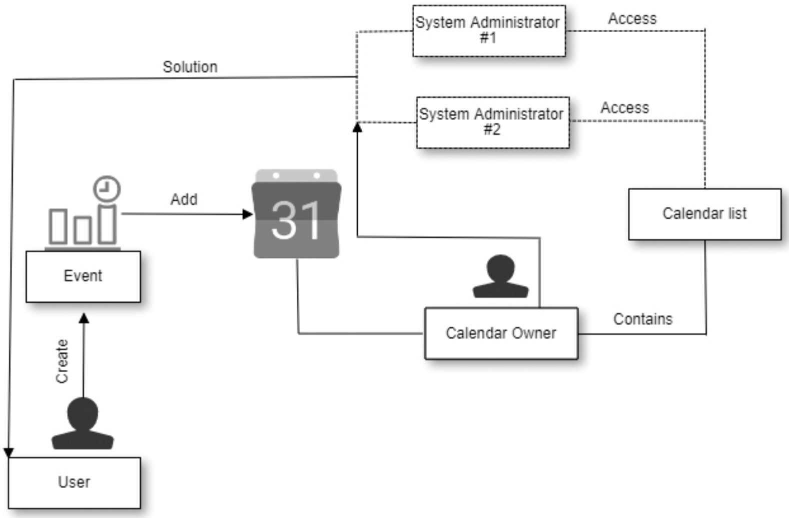


Fig. 2. User's interactions model

As you can see, a user with a calendar can view the entire list of events (calendar list). It is based on calendar entries. The calendar owner can add a participant to each event and specify an access rule by specifying the visibility of the calendar (public, private, or default).

Therefore, the network user support tool model provides for synergies between system administrators and users to solve problems that occur in the latter with subsequent distribution and control of problem solving. The following is intended to develop a server for collecting and processing information.

3.2 Implementation of the network user support tool

To create the network user support tool, we have chosen Microsoft .NET. It is a software technology for creating both desktop applications and web applications. It is largely a development of Java technology ideas and principles. One of the ideas behind

.NET is the compatibility of services written in different languages. .NET is a cross-platform technology, with Microsoft Windows, FreeBSD and Linux platforms being implemented at this time.

.NET is divided into two main parts – the execution environment (virtual machine) and the development toolkit. There are several .NET application development environments such as Visual Studio .NET, SharpDevelop, Borland Developer Studio, Eclipse. In the simplest case, user can develop applications in a text editor and use the console compiler.

To use Google Calendar, we need to use the API of this service. This interface is accessible through Google Code, a software developer service. It is for Google-related products. However, the site also provides third-party development services. One of the main services is the version control system. All projects published on Google Code are open source.

Application Programming Interface (API) is a set of subroutine definitions, communication protocols, and tools for building software. This is usually a method of abstraction between low-level and high-level software. Google offers a wide range of application interfaces. They allow developers to integrate their own applications with Google services, such as AdSense, Google Drive, Calendar, and more.

SOAP (Simple Object Access Protocol) APIs allow developers to integrate native applications with Google services. APIs allow developers to build applications that work with Google's cloud services. These include the Google Apps API, Blogger, Google Base, Google Book Search, Google Calendar, Google Code Search, Google Spreadsheets, Google Notebook, and Picasa. The Calendar API allows you to display, create, and modify calendar events. You can modify and control the event fields.

To use the Calendar API in Visual Studio, we have created a new project on Google APIs. We used the Calendar API for this project. Credentials were created to access the project. We have chosen to access OAuth 2.0 accounts.

The OAuth protocol allows the application to obtain user rights to use some APIs. The rights are denoted by a token whose properties are in no way defined. It may be the same for different users; it may be different for one user at different times. The main thing is that in exchange for the token, the program will be able to perform some actions on some service. Unlike the known OpenID protocol (which provides authentication), OAuth performs authorization [7].

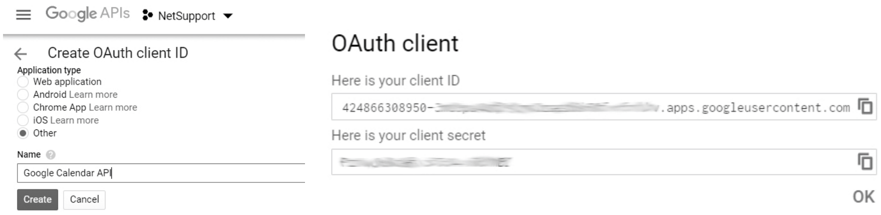
For applications that use the OAuth 2.0 protocol to call Google APIs, we used an OAuth 2.0 client ID to generate an access token. The token contains a unique identifier (Fig. 3). The received client ID and secret can always be accessed from Credentials in APIs & Services.

The JSON (JavaScript Object Notation) file with credentials client_secret.json was then downloaded from Google APIs.

After creating a new project in NuGet Package Manager, we ran the Install-PackageGoogle.Apis.Calendar.v3 command. It installed a library to work with the Calendar API (Fig. 4). A client_secret.json file was also added and set to always copy this file to the project.

For example, a class for retrieving calendar entries contains

- the using directive to allow the use of types in a namespace (Google.Apis.Auth.OAuth2; Google.Apis.Calendar.v3; Google.Apis.Calendar.v3.Data; Google.Apis.Services; Google.Apis.Util.Store);
- methods GoogleWebAuthorizationBroker.AuthorizeAsync and GoogleClientSecrets.Load for user authorize by OAuth2 protocol;
- instance of the class Google Calendar API service;
- parameters of request to service (request.TimeMin, request.SingleEvents, request.MaxResults, request.OrderBy, request.ShowDeleted).



Application type
○ Web application
○ Android Learn more
○ Chrome App Learn more
○ iOS Learn more
● Other

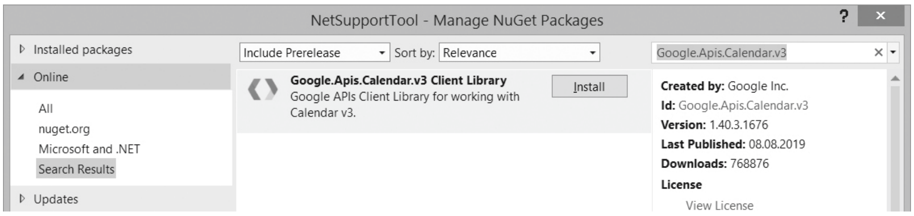
Name: Google Calendar API

OAuth client

Here is your client ID: 424866308950-[redacted].apps.googleusercontent.com

Here is your client secret: [redacted]

Fig. 3. Create a project in Google Developers Console



Include Prerelease | Sort by: Relevance | Google.Apis.Calendar.v3

Google.Apis.Calendar.v3 Client Library
Google APIs Client Library for working with Calendar v3.

Created by: Google Inc.
Id: Google.Apis.Calendar.v3
Version: 1.40.3.1676
Last Published: 08.08.2019
Downloads: 768876
License

Fig. 4. Installing Calendar APIs Packages

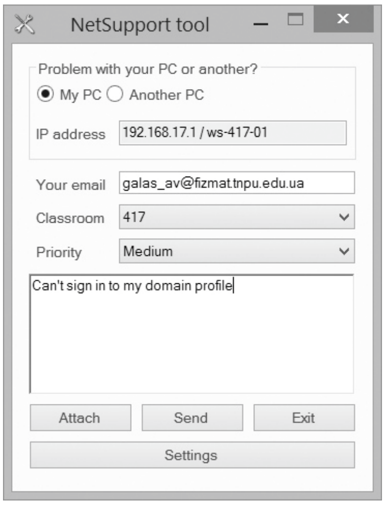
As you know, calendar owners can share the calendar with others. Calendar sharing options are provided in the calendar's ACL (access control list, see Table 1). Each resource in the ACL collection has specific access rights to calendar objects. We used the following Google Calendar ACL APIs (see Table 1).

Table 1. The Google Calendar APIs ACL collections

Property name	Type	Description
Etag	Etag	ETag of the resource.
Id	String	ACL Rule ID
Kind	String	Type of the resource ("calendar#aclRule").
Role	String	The role assigned to the scope. Possible values are: none, freeBusyReader, reader, writer, owner
Scope	Object	The scope of the rule.
scope.type	String	The type of the scope. Possible values are: default (public scope), user, group, domain.
scope.value	String	The email address of a user or group, or the name of a domain, depending on the scope type.

By default, each user has owner access to their master calendar, and this access cannot be changed. Because our university has a Google Suite academic subscription, some Google Calendar settings have been made at the domain level. For example, the support team was forbidden to share calendars outside of the organization. After we shared our calendars, we worked with individual events using the “scope.type” property.

When launching the network support tool, the user will see a fill in form that will form a new event in Google Calendar. The general view of the program is shown in Fig. 5.



Problem with your PC or another?
● My PC ○ Another PC
IP address 192.168.17.1 / ws-417-01
Your email galas_av@fizmat.tnpu.edu.ua
Classroom 417
Priority Medium
Can't sign in to my domain profile

Fig. 5. Interface of the network support tool

The request is formed in the following order. If the problem occurs on a computer where the user is working, then the radio button “My PC” remains in the form. In doing so, the program automatically fills in the IP address field of the computer, where the PC name is additionally written and the field data is not available for change. Instead, if the user selects RadioButton No, the IP address field will be accessible. You can select a classroom from the drop-down list. In the future, the audience number will be the title of the calendar event.

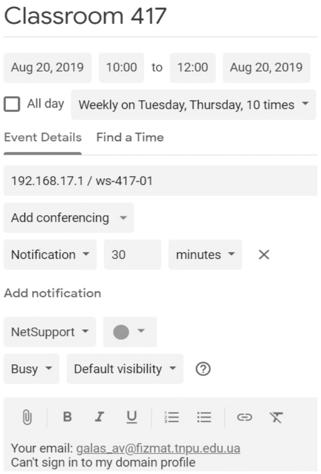
An e-mail box for the system administrator responsible for each classroom is assigned. The network support tool adds an employee as a participant to an event on Google Calendar and notifies them of a problem. The calendar event is marked in color based on data from the Priority list. There are three types of priority: high, medium and low. These priorities are matched by three colors – red, blue, or gray. Thanks to it, the administrator can see which task is of the highest priority. In the text box below, the user enters text describing the problem. You can also insert an image or screenshot here.

Clicking the Submit button prompts you to create an event on Google Calendar. When the request is generated and sent to the server, the program displays a message about the successfully created request with the text of the message. If there is no Internet connection, the user will see an error message about the application. Network support tool will output an error if not all data is entered.

The following code listing shows a method for adding an event to a calendar based on user input.

```
public void Method(string msg, string ip, string summary, string
email) {
    String basicEmail = "netsupport@fizmat.tnpuedu.ua";
    String colorEvent = "";
    String priority = comBPriority.Text;
    switch (priority) {
        case "Hight":           colorEvent = "11"; break; //red
        case "Medium":          colorEvent = "9";  break; //blue
        case "Low":             colorEvent = "8";  break; //grey
        default:                colorEvent = "11"; break;
      }
      UserCredential credential =
GoogleWebAuthorizationBroker.AuthorizeAsync( new ClientSecrets {
          ClientId = "*****.apps.googleusercontent.com",
          ClientSecret = "********",
       },
       new[] { CalendarService.Scope.Calendar },
             "user", CancellationToken.None).Result;
  var service = new CalendarService(new
BaseClientService.Initializer() {
    HttpClientInitializer = credential,
    ApplicationName = "NetSupport tool",
  });
Event myEvent = new Event {
    Summary = summary + "classroom",
    Location = ip,
    Start = new EventDateTime() {
        DateTime = DateTime.Now, TimeZone = "Europe/Kiev"
      },
    End = new EventDateTime() {
        DateTime = DateTimeEnd.AddDays(1),
        TimeZone = "Europe/Kiev"
      },
      Description = "Your email: " + email + "\n" + msg,
      ColorId = colorEvent,
      Recurrence = new String[]  {
        "RRULE:FREQ=WEEKLY;COUNT=10;WKST=SU;BYDAY=TU,TH"
      },
      Attendees = new List<EventAttendee>() {
           new EventAttendee() {
      Email = ws.m_aAudience[comBAudience.SelectedIndex].email},
          new EventAttendee() { Email = basicEmail }
        }
  };
  Event recurringEvent = service.Events.Insert(myEvent,
"primary").Execute();
 }
```

If the administrator goes to his calendar, he/she will see the created event (Fig. 6).



Classroom 417

Aug 20, 2019 10:00 to 12:00 Aug 20, 2019

□ All day Weekly on Tuesday, Thursday, 10 times

192.168.17.1 / ws-417-01

Notification 30 minutes

NetSupport ●

Busy Default visibility

Your email: galas_av@fizmat.tnpu.edu.ua
Can't sign in to my domain profile

Fig. 6. Google Calendar event

The system administrator must enter a login and password in order to enter the application settings. The settings available in the network support tool settings are:

- add a new classroom;
- change administrator's e-mail;
- remove the classroom.

We have created a separate "Workspace" class to process this information to store classroom and email data. This class can write and read data about the contents of an XML (Extensible Markup Language) file. The Workspace class is Singleton. Singleton is a design template, belongs to the class of creation templates. It ensures that a class has only one instance, and provides a global access point for that instance. Thanks to this, we can rest assured that our classroom data and email addresses are securely stored and will not be deleted the next time you launch the application.

4 Conclusions

This article analyzes the basic concepts related to supporting the work of users of computer networks. We have designed the key features that the program should perform, selected the language and technology of the software implementation, described how to create a Google Developer Console project, and steps to gain the rights to use API features. The result of this project is to obtain a JSON file that allows the application to access Google Calendar. It also describes the basic concepts of Google Calendar, its capabilities, and how it works with creating events, sharing them with other users.

When developing the network support tool, we learned to use Google APIs, including the Google Calendar API, to formulate and query calendar events using C# programming language and .NET technology. The main advantage of this software is that it promptly notifies the system administrator of the problem of the user, and allows to distribute and control the process of its execution. The downside is the inability to send a request without an internet connection.

We see prospects for further research, such as

- implementation of the server for information gathering and its analysis;
- creating a module for the system administrator, where he/she can mark the completed tasks, generate requests for events that are entered in the calendar by priority, audiences or performance notes;
- cross-platform development, in particular for OC Linux.

References

1. ChiPer: HelpDesk i ServiceDesk. Chto eto i zachem eto nuzhno vashei kompanii (HelpDesk and ServiceDesk. What is it and why does your company need it). https://habr.com/en/company/deskun/blog/331354 (2017). Accessed 17 Aug 2019
2. Fisher, C.A.: Manage digital assets with ITIL: Improve product configurations and service management. Journal of Digital Asset Management **2**(1), 40–49 (2006). doi:10.1057/palgrave.dam.3640071
3. Kunstová, R.: Process Support of Business and IT Management in Czech Companies. In: Møller C., Chaudhry S. (eds.) Re-conceptualizing Enterprise Information Systems. Lecture Notes in Business Information Processing, vol. 105, pp. 168–181. Springer, Berlin, Heidelberg (2012)
4. Kurhanov, D.A., Azaryan, A.A.: Software development to minimize time costs and increase productivity in the area of communication services. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 1st Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2018), Kryvyi Rih, Ukraine, November 30, 2018. CEUR Workshop Proceedings **2292**, 116–127. http://ceur-ws.org/Vol-2292/paper13.pdf (2018). Accessed 17 Aug 2019
5. Oleksiienko, I.V., Franchuk, V.M.: Web-oriented electronic schedule. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 1st Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2018), Kryvyi

Rih, Ukraine, November 30, 2018. CEUR Workshop Proceedings **2292**, 128–131. http://ceur-ws.org/Vol-2292/paper14.pdf (2018). Accessed 17 Aug 2019

6. Self Service - Deside. https://www.dezide.com/products/self-service (2019). Accessed 17 Aug 2019
7. Using OAuth 2.0 to Access Google APIs. Google Identify Paltform. https://developers.google.com/identity/protocols/OAuth2 (2019). Accessed 17 Aug 2019
8. Warren, M.: IT Help Desks Not Just For Large Enterprises. InformationWeek. https://www.informationweek.com/business-e-business/it-help-desks-not-just-for-large-enterprises/d/d-id/1098348 (2011). Accessed 17 Aug 2019
9. Zhao, H., Wang, Y., Zhang, X., Yang, L.: Campus Network Operation and Maintenance Management and Service Based on ITIL Service Desk. In: Zu, Q., Hu, B. (eds) Joint International Conference on Pervasive Computing and the Networked World ICPCA/SWS 2012: Pervasive Computing and the Networked World. Lecture Notes in Computer Science, vol. 7719, pp. 876–882. Springer, Berlin, Heidelberg (2013). doi:10.1007/978-3-642-37015-1_77
10. Zingtree: Get a Zingtree Demo. https://zingtree.com/demo.php (2019). Accessed 17 Aug 2019

Software implementation of e-trade business process management information system

Oleg Pursky[0000-0002-1230-0305], Anna Selivanova[0000-0001-6559-1508],
Tetiana Dubovyk[0000-0001-9223-4629] and Tamara Herasymchuk[0000-0001-8376-6519]

Kyiv National University of Trade and Economics, 19, Kyoto Str., Kyiv, Ukraine, 02156
pursky@gmail.com

Abstract. In this study we present the software development method for e-trade business process management information system. As a software platform for implementation of the e-trading business process management system has been selected MSSQL database and software component implemented with the use of ASP.NET MVC Framework. The developed Web-application includes the following functional blocks: user authentication module; administration module; block implementing the functions of domain mathematical model and information processing procedures; database intended to store initial data, trading activity indicators and calculation results, as well as the content of the Web-application interface components; logistics module; server end, which implements the procedures for the database access, the formation of the view modes of the models and functions of the domain mathematical model; interface that provides an interactive user access to the Web-application features. The connection between the Web-application and the database is performed on the basis of the GET http-protocol method and is provided by generating database queries and query results. The developed e-trade business process management information system has a three-level software and hardware structure and is designed to manage the processes of interaction between the online store and on-line consumers.

Keywords: e-trade, business process, software development, management system.

1 Introduction

Information technology is a reality of modern economy, and an entrepreneurship is inconceivable without electronic trading methods, without sales promotion on the Internet. The use of the Internet radically transforms the processes of economic cooperation between the companies and their customers, partners, suppliers and even competitors. Methods and means of carrying out commercial transactions in e-trade are somewhat different from those carried out in the real economy [14].

Socioeconomic subject matter of e-trade, its features, and innovative role in the modern economy relate to new, understudied areas of economic science [2]. This fact determines the need for an objective study of organizational and economic foundations for the development of e-trade to solve practical issues of increasing the efficiency and competitiveness of business.

Electronic trading markets operation has caused significant changes in the modern economy. The basic relations and business models have changed: from the manufacturer's domination, and then to the intermediary – to the leading role of the consumer; from mass production – to manufacturing products to order; from the value of material resources – to the value of information. At the same time, electronic trading markets have not changed the basic concepts of market relations, but have identified new mechanisms for their implementation. The electronic trading market is a prevailing electronic trading environment and a set of its participants, products, services and processes of their interaction, characterized by certain regularities in the context of a developed network of information and communication features. E-trade, as compared to traditional business, has substantial advantages. In particular, the use of new electronic communication channels significantly reduces costs related to organization and support business infrastructure, and the possibilities of e-trade allow re-designing business strategy at any moment [13].

An important circumstance that ensures the electronic trading markets operation is the availability of a fully-fledged electronic trading environment. The electronic trading environment is formed in the presence and direct interaction of such factors: modern information and communication tools for e-commerce, a wide range of consumers with full access to information and communication facilities (with Internet access), sellers, using information and communicative means for the sale of goods and services. Information and communicative tools for electronic trading are the modern information technologies and systems that provide effective implementation of procedures for reading, processing, storing and transfer of information resources based on computer networks.

The effectiveness of the electronic trading markets operation depends on the number of potential consumers of products and services presented on the market. An electronic trading environment cannot be formed without a wide range of consumers. Their presence is determined by the level of development and the degree of implementation of information and communication facilities in all spheres of socio-economic activity. Without providing users with Internet access, it's difficult to talk about the availability of an electronic trading environment. The presence of vendors using computer network resources for the sale of goods and services is a proactive factor in the electronic trading environment formation, since sellers are interested in making profits through the use of new, electronic trading mechanisms. At the same time, in order to have an electronic trading environment, sellers have to inform potential buyers in virtual space, while they bear the appropriate promotion expenses and market entry [12]. The cost of electronic trading market entering, it is by definition the financial and material costs to ensure the operation of the Internet store (website creation, delivery of goods organization, electronic payment systems connection, etc.). Promotion expenses in electronic trading are related to the creation of mechanisms for influencing consumers' preferences, by forming the priority consumer goods value. Since there is little or no cost of obtaining information about a particular product or service in an online environment, the role of informative advertising in electronic commerce is significantly restricted, that is, the overwhelming part of advertising is a manipulative advertising, which increases the readiness of potential buyers to buy goods and services [13]. Promotional expenses in electronic trading primarily relate to the arrangement and placement on various electronic resources (own websites and other areas providing advertising) the banners about services and products and various incentive measures. Thus, potential buyers will get information about sellers and can interact with sellers through information and communication tools.

The electronic trading markets operations are associated with the interaction between sellers and buyers. Interaction of the electronic trading market participants is carried out whiles the procedures for the purchase and sale of goods and services implementation. Electronic trading market participants include sellers, buyers and intermediaries, as well as organizations that provide infrastructure and market rules. Regarding the intermediaries, we have not distinguished them separately for the reason that the main purpose of their activity is, as in the direct salesmen, in providing sales of goods and services in order to generate profits. The process of buying and selling goods and services in electronic commerce is much more complicated than in traditional trade, since even the structure and interface of an online store site can affect the specifics of this process. The purchase and sale of goods and services in electronic commerce begins with the search and acquaintance of a potential buyer with the range of products and services he or she needs in the virtual environment. After the final choice, the buyer makes an application for the selected product or service. The Internet seller carries out the order processing specifies the buyer's choice and, if confirmed, forms an account for payment. Then, checking the availability of the desired product or agreeing on the procedure for providing the services selected by the buyer take place. In the case of availability of goods or the buyer’s agreement on the proposed procedure for the provision of services, the buyer is sent an invoice for payment. Payment is made with the mutual consent of the buyer and the seller and can be executed pre-payment (prepayment) or on the fact of delivery of the goods and performance of the service (post payment). After the payment procedure is agreed, the service is performed or the goods are sent to the buyer. Delivery of goods can be carried out both at the address of residence and to the nearest to the buyer pickup point (for example, “FedEx” or “DHL Express”). Payment for goods and services in electronic trading is usually carried out after the delivery of the goods or the performance of the service, this increase the buyers’ confidence.

It should be noted that all the elements of the electronic trading market are present both in the real, physical environment, and indirectly, in a virtual trading environment based on the use of information and telecommunication technologies. This means that, depending on the type of merchandise product on the electronic trading market, it can be presented in the process of purchase and sale as either a real product or as a product presented in digital format (for example, a software product or information service provided using the Internet).

The technological aspects of the electronic trading market are extremely important factors that ensure the effectiveness of its operation. The reliability, safety of technological solutions is the basis of e-commerce, which depends on the participation in procedures of purchase and sale. The current level of technology development allows creating high-performance, protected from external interference information and communication means of ecommerce. There is also a certain list of organizations that support the reliable operation of computer networks, the main means of e-commerce. The organizations providing the World Wide Web are represented by three types of ISPs (Internet Service Provider), IPP (Internet Presence Provider) and PCP (Private Content Publisher). They have rights to different types of servers, including host-computers (full-time operating equipment), and provide the following main service (services): E-mail, FTP (File Transfer Protocol), WWW (World Wide Web – customers hosting).

Automation of large-scale business processes vital for e-trading systems efficiency remains one of the most complicated problems in large enterprises. Known theoretical and practical solutions for small and medium businesses do not comply with large companies' requirements, their functions and development tools, as well as the level of integration of Web-environment and enterprise management systems [15]. The efficient IT support for e-trading is deterred with architecture and capabilities of informational systems not focused on creating an interconnected e-trading network [4]. An urgent need arises for developing new conceptual approach focused on integration of e-trade business processes with informational management systems. The problem of deep integration of informational environment as a basis for building efficient e-trading systems is complicated with technological discontinuity in typical informational systems stipulated by the wide range of software and network solutions during their creation. Therefore, a significant interest of scientists in the management software development in e-commerce, which allows increasing the efficiency of economic activity of enterprises, is quite reasonable. In this research we present the method for software implementation of e-trade business process management information system.

2 Results and discussion

Preliminary studies of the mechanisms of the electronic trading markets development and operation [12; 13] allowed determining the structural components arrangement and the functional use of core information flow processing modules of the e-trade business process management system and provided the opportunity for the transition to its software implementation.

The business processes accompanying the e-trade have been taken into account in the course of the software Web-based application development. The software implementation of the e-trade business process management system has been performed based on the developed transaction and advertising cost models [13]. Important aspects of the developed e-trading business process management system are the automation of all calculation mechanisms and framing procedures for the products delivery routes and cross-browser support for a computer Internet network, providing access to all its resources for geographically spread users, regardless of the location. The developed Web-application includes the following functional blocks:

- user authentication module;
- administration module;
- block implementing the functions of the domain mathematical model and information processing procedures;
- database intended to store initial data, trading activity indicators and calculation results, as well as the content of the Web-application interface components;
- logistics module;
- server end, which implements the procedures for the database access, the formation of the view modes of the models and functions of the domain mathematical model;
- interface that provides an interactive user access to the Web-application features.

User authentication module. This module provides functional differentiation of access modes to the Web-application controls and its resources according to the user’s role and performs verifying the correctness of the registration data entered by the user. To access the application functions, it is required to enter the login and password stored in the system after the registration procedure in the e-trade business process management system. This module verifies the compliance of the user-entered data with the records stored in the database. In the case of the correct data entry, the module provides an access to the Web-application functionality and resources in accordance with the user access level specified by the system administrator.

Administration module. The module is designed to specify the levels of user access to the Web-system resources. The module regulates access modes to the Web-application functions on the basis of the work with three user groups (administrators, managers, and users).

Database. In the course of the domain conceptual model development, its entity and the relationship between them have been determined. Each entity with the database development is converted into a table, each entity attribute corresponds to the field in the database table. Figure 1 show the database model used in the interaction management Web-system. Designing data arrays involves defining their composition, content, structure, and choice of rational way of their provision. The database has been developed with the use of DBMS MSSQL [1]. Its logical structure can be seen in figure 1. The database contains the following tables:

- *Addresses table* – is used to store the information about buyers' addresses;
- *AspNetRoles table* – is used to store the information about user access modes;
- *AspNetUserRoles table* – is used to store the information about the correspondence of access modes and users;
- *AdvertisingCosts table* – contains information on the size of the local e-trade market (number of sellers and potential buyers) and on the level of consumer awareness and the degree of saturation of the market by advertising;
- *AspNetUsers table* – is used to store the information about registered users of the system;
- *AverageValues table* – is an auxiliary table for saving the averaged values of the domain mathematical model;
- *MarketEvents table* – is used to store the information about marketing activities;
- *Orders table* – is used to store information about the order;

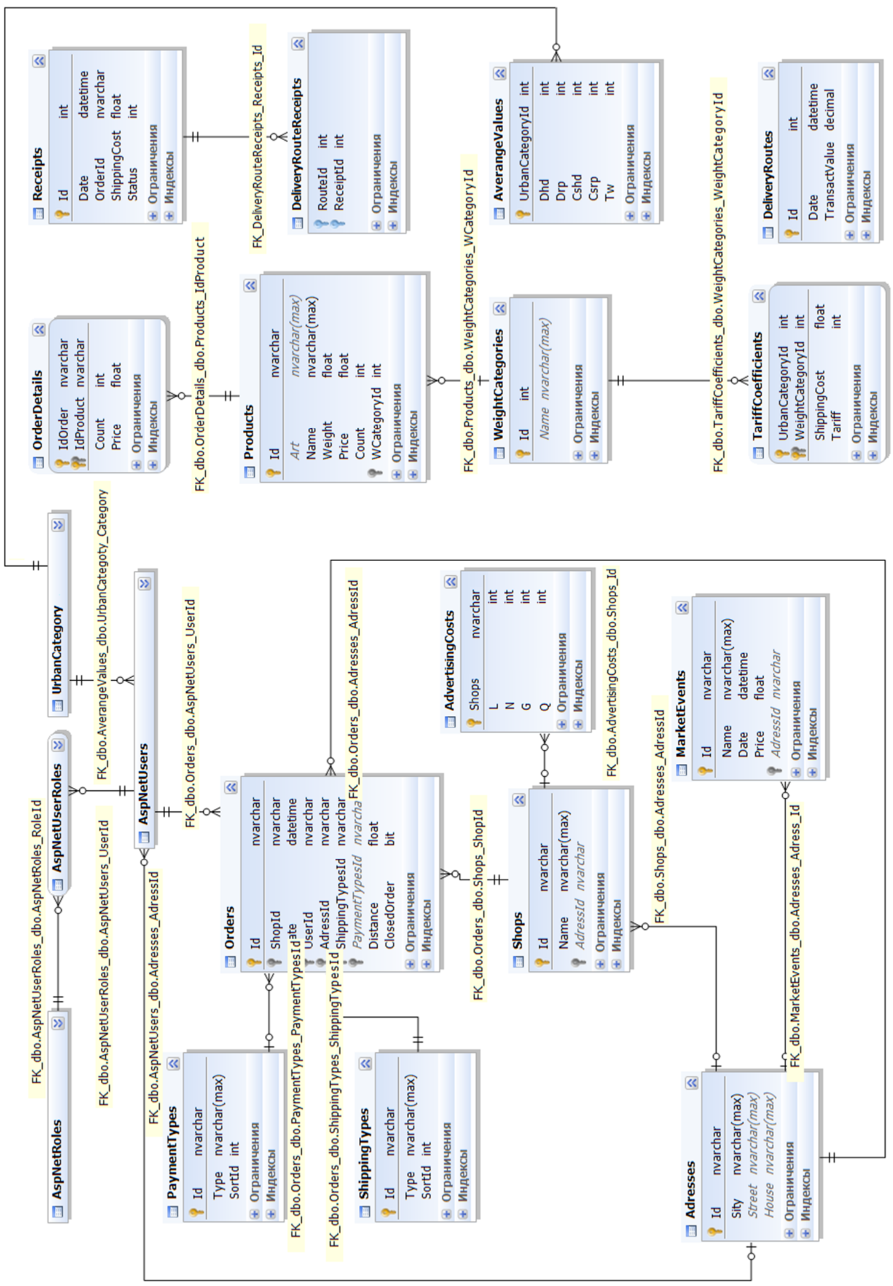


Fig. 1. Database model.

- *PaymentTypes table* – is used to store the information about different forms of payment for products and services;

- *Products table* – is used to store the information on the products;
- *Receipts table* – is used to store the information on payment receipts for products and services;
- *ShippingTypes table* – is used to store the information on delivery methods;
- *Shops table* – is used to store the information about the online store;
- *TariffCoefficients table* – is used to store the information for determining the tariff coefficient;
- *DeliveryRouteReceipts table* – is used to store the information about the correspondence of the route and receipts on orders in the route of goods delivery;
- *DeliveryRoutes table* – is used to store the information about products delivery routes;
- *WeightCategories table* – is used to store the information on weight categories of products;
- *UrbanCategory table* – contains information about urbanization categories.

The connection between the Web-application and the database is performed on the basis of the GET http-protocol method and is provided by generating database queries and query results [8; 11]. The resulting samples are serialized in the *.JSON format for transmission to the Web-application [1]. If it is necessary to save data to the database, the Web-application generates an object in the *.JSON format; the deserialization of the object obtained by the POST http-protocol method, as well as the generating of the corresponding database queries (REPLACE or UPDATE) takes place in the PHP gateway [5].

The block implementing the functions of the domain mathematical model and information processing procedures. The block implements computational algorithms for determining transaction and advertising costs [13] and performs automation the procedures for processing information on sales and delivery of goods.

Logistics module. The module consists of two functional parts - an interactive map and a list of routes with the order structure on the route. The main purpose of the module is to automate the mechanisms of transport logistics for the products delivery to customers and to ensure the visualization of procedures for the delivery routes development with a graphical visualization of the route pattern on the interactive map.

Functionally, the module also provides calculation and visualization (by data transfer to the block of the trade indicators analysis) of transaction costs of consumers with the ordered product individual delivery, when delivering products via a collective route and calculated in accordance with the developed economic-statistical model of transaction costs. There are four main elements in the module structure: *the access block to the functions of the Logistic module; OrdersController control block; Visualization block by applying LogisticViewModel view model; OrderManager providing a block*. The structure of its blocks and interconnections between them are determined, first of all, by the functional requirements for the processes of the sold commodity products delivery to customers in electronic commerce. The operation of the logistics module is carried out as follows:

- *OrdersController* generates a list of *Routes*, based on the data on placed buyers' orders for delivery, which are formed by the *GetReceiptsListAsync* and

GetTariffModel methods, and also transmits a completed list for its visualization using L*ogisticViewModel* tools;

- In the *LogisticViewModel* view model, the calculation of the distances for each route based on the list of products delivery addresses and transferring these data to the *OrdersController* is performed with the use of Yandex.Maps API;
- The *Calqulate method* performs the calculation of the consumer transaction costs based on the delivery route data and the *CalqulateTariff* method that provides the data on the tariff coefficients and returns the calculated data to the *LogisticViewModel* view model;
- After the products delivery route confirmation by the manager and the printout of the corresponding route documents, the *OrdersController* transfers control to the *OrderManager* provider, which changes the order status using the *SetOrderShipping* method and stores the information in the database using the *Save method*.

Server end and Web-application interface. As a software platform for the implementation of the e-trading business process management system the following components has been selected: MSSQL database, the software component implemented with the use of ASP.NET MVC Framework [6] – a framework for creating Web-applications that the Model-View-Controller template generates.

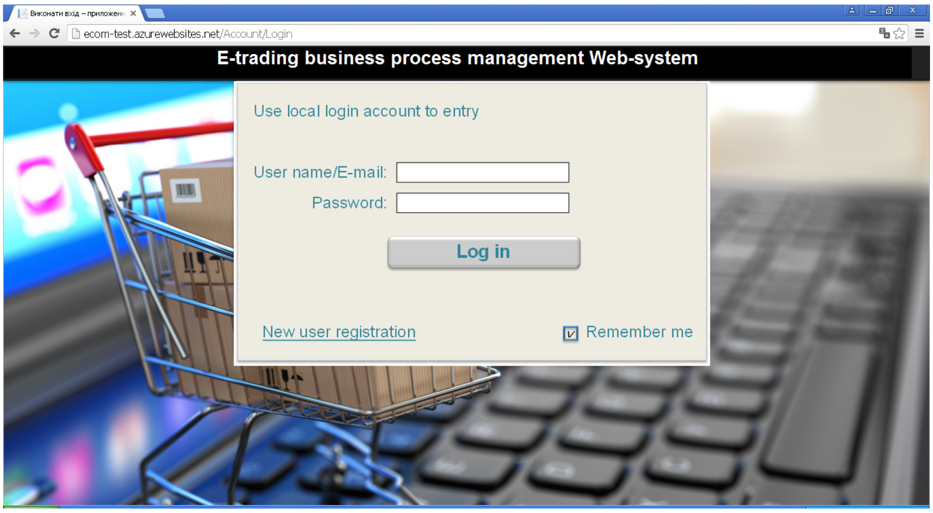
The ASP.NET MVC platform is based on the interaction of three components: controller, model, and view. The controller accepts queries, processes the information entered by a user, interacts with the model and view, and returns the results of the query processing to the user. The query processing generator is used to control the markup and code insertion.

The Razor view generator is used to control the markup and code insertion in the view. The C# programming language [9] is used as the language of the software implementation of the e-trade management Web-application. To provide the interconnection of the interface interactive components of the e-trade business process management system with the Database, the ADO.NET Entity Framework [10; 7] – an object-oriented data access technology is used. The interface of the developed Web-system in e-commerce includes the following parts:

- controls that provide user authentication, verification of the entered data and providing access to the Web-application functions, according to the level of user access;
- controls that provide the Web-application user registration;
- administration module for editing account parameters and assigning user access modes to the Web-application functions;
- menu of the interaction management Web-system in e-commerce, which provides navigation for Web-application users, including the following menu items: homepage, products, documents, marketing, logistics, analysis;
- information block with the viewing of the information on the developed integrated e-trading business process management Web-system and the technology for its use;

- interactive map of the region showing the location of the shoppers of the on-line store with the availability of graphic viewing of the selected route for the products delivery;
- module for viewing the routes for the products delivery and the structure of the orders on the route with the availability of the route documentation printing (delivery route, route task, list of orders on the route, payment receipts and delivery costs);
- block for filing and viewing of the operational and reporting documentation for describing business processes in e-commerce;
- block for viewing the list and characteristics of marketing activities and the results of the advertising costs calculating in e-commerce;
- block for viewing the results of the transaction costs calculations of consumers in electronic commerce;
- block for viewing analytical information on trade activity indicators in the form of diagrams and charts.

Technical requirements. To ensure the Web-application functioning on the server end, the following software is required: MS IIS 7.0 and higher; ASP.NET MVC 5.0 and higher; MSSQL 2008 and higher. A modern system and application software (operating system and browser) should be installed on a client's computer for the reliable functioning of the business process management system in electronic trade. The homepage of e-trade business process management system is given in figure 2.


Виконати вхід – приложени
ecom-test.azurewebsites.net/Account/Login

E-trading business process management Web-system

Use local login account to entry

User name/E-mail:

Password:

Log in
New user registration
☑ Remember me

Fig. 2. E-trade business process management system homepage.

The e-trading business process management system interface is designed to work with three groups of users (administrators, managers, and users) and provides the use of the Web-application functionality and controls, depending on the particular user access level.

Administrators – have access to editing the e-trading business process management system context and can distribute user access rights to Web-application features.

Managers – have access to all Web-application features, with the exception of the above-mentioned administrator rights. Managers can edit tables with the information on the product range and trading procedure characteristics and initiate and regulate all calculation and reporting procedures. In essence, the access mode of the manager provides the full use of all functions of the e-trading business process management system in order to automate the trading activity of the online store.

Users – have access to the selection procedures of the products and the order placement for the purchase of commercial products.

The Web-application has been implemented with the use of the Model-View-Controller pattern (MVC) [3; 6] – a usage pattern of several design patterns through which the application model, user interface, and the interaction between the application and the user are divided into three separate components, so that the modification of one of the components has a minimal impact on the other. The developed e-trade business process management information system has a three-level software and hardware structure and is designed to control the processes of interaction between the online store and on-line consumers. The e-trade system does not require specialized user training and can be used regardless of the trading activity specifics of a certain online store.

3 Conclusion

The developed e-trade business process management information system provides management of the trading activity of an electronic enterprise in the framework of B2C and B2B interaction models. Modern software tools such as MSSQL, ASP.NET MVC Framework and the C# programming language have been selected as the tools for the Web-application software implementation. The Web-application is designed to work with three groups of users (administrators, managers, and users) and provides access to interaction controls depending on the level of a particular user access. For each group of the Web-system users, the measures have been taken to ensure the security and reliability of trading operations procedures, regardless of their territorial distribution. Important aspects of the developed interaction management Web-system in e-trade are the automation of all calculation mechanisms and framing procedures for the products delivery routes and cross-browser support for a computer Internet network. The Web-system provides calculation of advertising and transaction costs and graphic visualization of the transaction costs of consumers with the ordered product individual delivery, with the product delivery on a collective route, which allows an Internet seller to track decrease in the level of transaction costs of consumers as a result of the collective routes introduction for the products delivery.

4 Acknowledgements

This study was supported by the Ukrainian Ministry of Education and Science, Project No. 0117U000507, “Modeling the mechanisms of international e-commerce operation”.

References

1. Assaf, W., West, R., Aelterman, S., Curnutt, M.: SQL Server 2017 Administration Inside Out. 1st edn. Microsoft Press, Redmond (2018)
2. Caporale, G.M., Plastun, A., Oliinyk, V.: Bitcoin fluctuations and the frequency of price overreactions. Financ. Mark. Portf. Manag. **33**(2), 109–131 (2019)
3. Chadwick, J., Snyder, T., Panda, H.: Programming ASP.NET MVC 4. Developing Real-World Web Applications with ASP.NET MVC. O’Reilly Media, Sebastopol (2012)
4. Chaffey, D.: E-business and E-commerce Management. 5th edn. Prentice Hall/Financial Times, New Jersey (2011)
5. Forta, B.: MySQL Crash Course. 1st edn. Sams Publishing, Indianapolis (2004)
6. Frimen, A.: Pro ASP.NET MVC 5. 5th edn. Apress, New York (2013)
7. Jennings, R.: Professional ADO.NET 3.5 with LINQ and the Entity Framework. 1st edn. Wrox, Birmingham (2009)
8. Kromann, F.: Beginning PHP and MySQL: From Novice to Professional. 5th edn. Apress, New York (2018)
9. McDonald, M.: Pro WPF in C# 2010: Windows presentation foundation in NET 4. 3rd edn. Apress, New York (2010)
10. Mueller, J.: Microsoft ADO.NET Entity Framework Step by Step. O’Reilly Media, Sebastopol (2013)
11. Nielsen, P., Delaney, К., Machanic, A., Tripp, K., Randal, P., Low, G.: SQL Server MVP Deep Dives. 1st edn. Manning Publications, New York (2009)
12. Pursky, O., Dubovyk, T., Moroz, I., Buchatska, I., Savchuk, A.: The price competition simulation at the blended trading market. In: Kiv, A., Semerikov, S., Soloviev, V., Kibalnyk, L., Danylchuk, H., Matviychuk, A. (eds.) Proceedings of the Selected Papers of the 8th International Conference on Monitoring, Modeling & Management of Emergent Economy (M3E2-EEMLPEED 2019), Odessa, Ukraine, May 22-24, 2019. CEUR Workshop Proceedings **2422**, 15–26. http://ceur-ws.org/Vol-2422/paper02.pdf (2019). Accessed 14 Sep 2019
13. Pursky, O.I., Grynyuk, B.V., Shestopal, D.A.: Planning of advertising costs and vendor number at e-trade market. Actual Problems of Economics **177**(3), 407-413 (2016)
14. Schneider, G.: Electronic Commerce. 12th edn. Cengage Learning, Boston (2016)
15. Shaw, M.J.: E-business Management: Integration of Web Technologies with Business Models. Springer, New York (2002)

Development of a web-based system of automatic content retrieval database

Olha V. Korotun[0000-0003-2240-7891], Tetiana A. Vakaliuk[0000-0001-6825-4697] and Viacheslav A. Oleshko[0000-0001-6434-250X]

Zhytomyr Polytechnic State University, 103, Chudnivska Str., Zhytomyr, 10005, Ukraine
{olgavl.korotun, tetianavakaliuk, vladolleshko19}@gmail.com

Abstract. In this work, the database was designed and implemented in accordance with the requirements of the relational model, which ensures the storage and collective access to the information of the auto-filling system and CMS WordPress data. Algorithms of system functioning were developed, the order of interaction of classes during program code execution was determined, as a result of which the application was implemented. Template Method architectural pattern was chosen to implement the web-based automatic content filling system. The following tools and technologies were selected to create the software package HTML markup language for HTML documents; programming language PHP; MySQL database management environment; Apache web server; the OpenServer package. The algorithms of the basic processes of content filling automation were considered and the interaction of the system classes during the processes of parsing, filtering and storing of information were analyzed. The developed system does not require specialized hardware, additional settings and deployment tools other than the standard ones for such plugins. This application is mostly for the site administrator and does not have user interface. That is why the features of the plugin automation system configuration interface; RSS feeds view and management interface, as well as the RSS feed configuration interface are described in detail. In the future, this system can be improved by introducing new functionality and improving the algorithm for reading data.

Keywords: system, content, automatic content, development.

1 Introduction

1.1 Formulation of the problem

Professional SEO and website promotion are long processes. In such circumstances, it is difficult to predict the timeframe within which a project will start to return investment and generate profit. To increase the load, you can work for three or hire a copywriter, programmer, and marketer.

An automatic content filling system is a cost-effective alternative that will save you unnecessary costs and reduce the time spent filling the site with content. The secret of auto-filling is extremely simple – the staff is replaced by a special program or plugin, customized for the project's requests. Its functions are to collect, adapt and publish content from competing for RSS feeds on a web resource.

The urgency of the chosen topic is that automation of the automatic filling system will provide information to the web resource without the help of a moderator, which will greatly simplify the maintenance of the web resource with minimal interventions in the process. The functionality of the system for automatic filling of information will allow using it according to the needs of the user.

1.2 Analysis of recent research and publications

The problem of development of the system of automatic filling of the context was investigated in various aspects: application of the information system of content management of a web resource for conducting e-commerce [1]; unified methods of processing information resources in systems of electronic content commerce [8]; peculiarities of formation and analysis of content of Internet newspaper of music news [4]; intellectual content management system for e-business sites [2]; application of content analysis of textual information in e-commerce systems [3], etc.

In particular, in paper [8] is described the formal model of information resource processing in e-commerce systems that simplifies the technology of content formation, management and implementation, and proposes methods for solving e-commerce problems and functional content management services.

That is why the purpose of this article is the design of architecture, the development of algorithms and the implementation of software complex information retrieval by parameters and automation systems for information processing.

2 Methods

Methods of research: theoretical analysis of scientific literature to clarify the state of the problem under study, systematization, generalization. The design method was also used to develop the architecture of the application, the methods of algorithm design and object-oriented programming – to develop algorithms for the operation of individual blocks and the application as a whole.

3 Results

The main purpose of the implementation is to simplify the work of filling the site with information. First, implementation of the system will help the site administrator to automate their functional tasks: fully automate the process of finding the necessary information, automate and organize the storage of data, reduce the time of work with the site, the time of their processing, as well as save money in the promotion of the site.

The result of this task is a comprehensive web-based content automation system, which contains a server structure of data storage, a multi-user client application for the implementation of functionality and means of control and access control [6].

Content Filling Automation automates content collection and publishing on a web-based resource. The modern software market features a wide variety of tools and technologies that help you solve problems related to the automatic search and content parsing processes.

WP RSS Aggregator is the most popular, easy to use and effective plugin for news aggregation. Its main functions are the ability to specify multiple sources, update interval, hide or no source, control the display of material. The plugin is free, but for some add-ons that extend its functionality, you will have to pay. The disadvantages include a small number of content post-processing features.

The FeedWordPress plugin is one of the news aggregators. The news collected by the plugin is copied to the database in the form of notes of a separate type, with the assignment of appropriate tags. If the required tag is not already in the database, the plugin will create one automatically. However, the plugin is very cumbersome and has many settings that will not be clear and useful to the potential user.

WPeMatico is an easy-to-use news aggregator that automatically publishes content from various sources, combining them into so-called "campaigns" according to your chosen topic. It can use keywords, phrases, and regular expressions to filter material, but most of the functionality is paid.

The Push Syndication plug-in has been specifically designed to manage to autocomplete across multiple sites. With one click, you can post to multiple platforms (up to more than 100 sites). The solution can be used to generate API tags used to promote blog content on WordPress, but the plugin does not have content settings.

The Syndicate Out plugin allows blog owners to auto-aggregate or creates content blogs from any number of different sources without relying on RSS feeds. However, there is no media-parsing, configuration, and post processing of the content.

CyberSyn is a powerful, easy and easy-to-use Atom / RSS posting plugin. It allows you to automatically receive and embed videos from YouTube channels. It does not have any problems with the syndication of various types of embedded media content. The disadvantages include storing all links from the source, inability to add multiple RSS feeds at a time.

Therefore, the main features of the new system should be the presence of a web interface, the module for parsing and storage of content and media data, the module for generating articles by parameters, the module for filtering information for parsing by parameters.

The Template Method architectural pattern (Fig. 1, 2) was chosen [7]. The Template Method pattern is widely used in application frameworks. Each framework implements immutable pieces of architecture in the domain and identifies those parts that can or should be customized by the client.

The component designer decides which algorithm steps are unchanged (or standard) and which are variable (or custom). The abstract base class implements standard algorithm steps and can provide (or not) the default implementation for custom steps. Variable steps can (or should) be provided by a component client in specific derived classes.

The component designer defines the required steps of the algorithm, the order in which they are performed, but allows the component clients to extend or replace some of these steps.

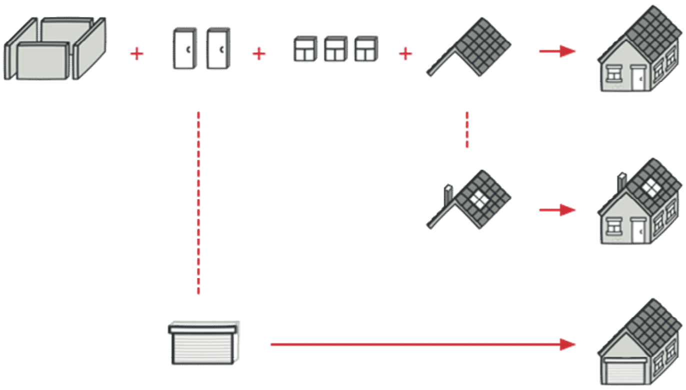
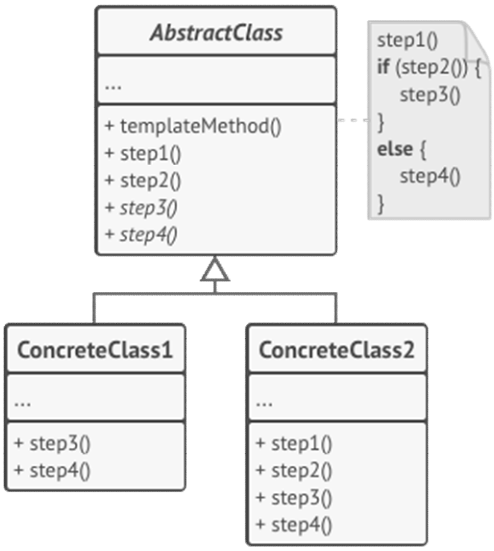


Fig. 1. An analogy to the life pattern of the Template Method



```
step1()
if (step2()) {
    step3()
}
else {
    step4()
}
```

Fig. 2. Template Method Pattern Structure

The abstract class defines the steps of the algorithm and contains a template method consisting of the calls of these steps. The steps can be both abstract and include a default implementation.

The specific class overrides some (or all) steps of the algorithm. Specific classes do not outweigh the template method itself. Concerning the platform on which the system is built, the most popular CMS in the world has been selected today.

In general, Content Management is a web application that allows site owners, editors, authors to manage their sites and publish content without any programming knowledge.

Word Press uses PHP and MySQL, which is supported by virtually all hosting providers [5].

Typically, this CMS is used to create a blog, but a WordPress site can easily be turned into an online store, a portfolio, a periodic site that is indisputably suited to the subject matter of a web-based content filling system.

One of the important features of WordPress is its intuitive and friendly interface.

The important thing is that WordPress is an open-source system and is free for everyone. In addition, it allows millions of people around the world to create modern, high-quality sites that can easily connect to the automatic content filling system and fill your site with content within minutes.

The following tools and technologies were selected to create the software package:

- markup language for hypertext HTML documents;
- PHP programming language;
- MySQL database management environment;
- Apache web server;
- OpenServer package.

The system of automatic filling of content has the main purposes: the project is created to automate the collection and publication of unique content online resource.

User requirements:

External users – User:

1. Two-way communication with the administrator via the email contact form.
2. Getting information about the actual content (on the site).
3. Getting information about current content changes (on the site).
4. The user has the opportunity to post comments about the published content on the site.
5. Provide useful links to related sites.
6. Provide background information on related topics in the form of articles.
7. View the latest news of the site: information about the new features of the information system available to the user.

Internal admin users:

1. Add, remove, and edit content published by the system.
2. Change the content status (delay posting).
3. Database editing.
4. View content that is being processed or published with the participation of this information system.
5. Information exchange with external users via email correspondence.

Characterization of the object of computerization:

The user on the site will be able to view the content that was published by the system of automatic filling of content about current articles, to leave relevant comments on the received content, as well as to receive answers to questions via the contact form.

In turn, the administrator has the opportunity to customize the system to specific content topics, select sources of information, keywords to search, organize a template for the appearance of content design.

Functional requirements:

1. Authorization of users in the system: The system must have the function of authorizing the user and assigning him the appropriate role.
2. Maintenance of the working directory: A set of articles on specific topics, designed by the system. Content management tools should be provided in IP.
3. Ability to store information: The system must store the information and allow the administrator to manage it.
4. Creating conditions for online communication of users: The system should allow users to communicate in the mode of email correspondence.

Non-functional requirements:

1. Perception

- It takes 1 hour for ordinary users to learn application tools and 20 minutes for experienced users.
- The system response time for normal requests should not exceed 1 second and for more requests that are complex 20 seconds.
- The application presentation interface must be intuitive to the user and require no further training.

2. Reliability

- Availability – the time required for system maintenance should not exceed 1% of the total operating time.
- Average continuous working time is 20 working days.
- The maximum rate of errors and defects in the system operation is 1 error per 1000 user requests.

3. Productivity

The system must support a minimum of 100 concurrent users associated with a shared database.

4. Ability to operate

- Scaling – the system should be able to increase capacity (productivity), with the increase of users in such a way that it does not negatively affect its performance.

- Version Updates – Updates should be updated automatically depending on the preferences of the users and the expansion of the list of scheduled content.

The CMS WordPress platform and the PHP programming language [9] were chosen to implement the project.

Analysis of functional requirements allowed us to distinguish the following entities that will provide the implementation of the software system. In Fig. 3 presented a diagram of the classes of the system controller level.

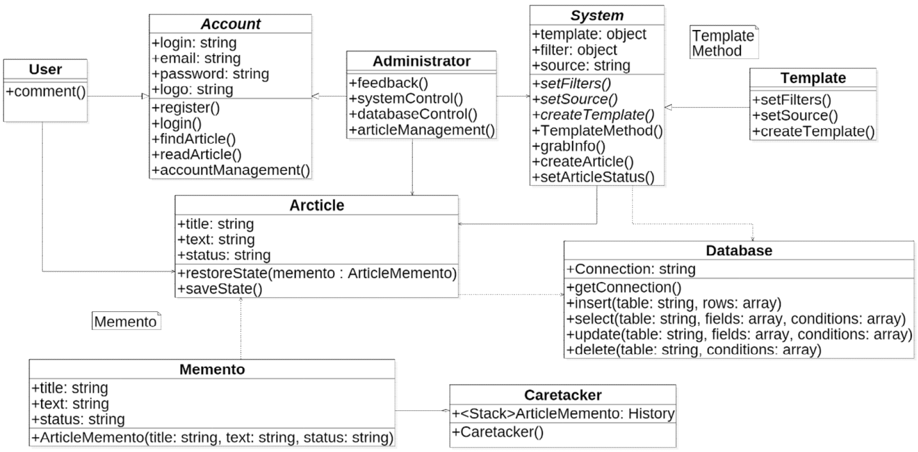


Fig. 3. Class diagram

The following classes can be distinguished in this figure:

- User – user class. The class has the following methods: comment - for commenting on a newly created article.
- Account – a class for saving user data. The class has the following methods: register – for user registration, login – authorization on the news site, findArticle, readArticle, accountManagement – editing user account data.
- Administrator – the site admin class. The class has the following methods: feedback – response to user messages, systemControl – entering and editing the automatic content filling system, databaseControl – receiving and editing database information, articleManagement – viewing and editing articles received by the automatic content filling system.
- System – a class of automatic content filling system. The class has the following methods: setFilters – installing and editing content filters, setSource – installing and editing an RSS feed of a donor site, createTemplate – creating an article template from an RSS feed to create articles, grabInfo – parsing content from a donor site, createArticle, setArticleStatus.
- Article – a class of articles on the automatic content filling system.

Thus, this system implements the functionality of content parsing from the donor site, filtering information, saving data in the form of articles organized on the WordPress platform.

The implemented MySQL database consists of 12 tables that contain all the data for the program. The bulk of these tables were created and maintained automatically by CMS WordPress, so only those that use SANC will be considered. The database is named wp-auto.

The structure of the database is shown in Fig. 4.

Consequently, the database, following the requirements of the relational model, provides the storage and collective access to the information of the autofill system and CMS WordPress data. The database consists of 12 tables. The main ones are wp-posts, wp-postmeta, wp-terms, wp-options.

Design and implementation of algorithms for system operation

The main modules of the system are the Parsing and Storage module in the CMS WordPress database.

User activity:

- when logging in to the site, the user can log in under the rights of the user or administrator, if he has such access;
- the user is logged in as a user, he or she can search for articles in the list available;
- the user opens the article and reads the information;
- the user has the opportunity to leave a comment under the article;
- the user also can manage their account data;
- the user is logged on as an administrator, besides all features of the user under the rights of the user, he/she is additionally able to control the system of automatic filling of content;
- the administrator can edit the information received for the article from the automatic content filling system;
- admin can set the status for the article (delay posting);
- the administrator has the opportunity to customize an article template that copies the auto-fill system;
- the administrator can add, delete and edit sources of information from which the auto-fill system copies the content.

The implementation of the activity of the system provides for the interaction of models Account, Administrator, User, Article, and System (Fig. 5). The main methods used in this process are System – for parsing content from RSS feeds into the system, Template – method of information storage template, Article – a method that saves filtered information in the form of CMS WordPress article.

Thus, the algorithms of the basic processes of content filling automation were considered and the interaction of system classes during the processes of parsing, filtering and information storage was analyzed.

Content Filling Automation is a plugin based on CMS WordPress and created in PHP – it has configuration files that spell out the domain name or path to the files. Before transferring the system to another hosting, you must save the location of all additional libraries to the plugin. The system uses CMS WordPress, so you need to move the plugin to a special plugin folder for the system to work properly wp-content/plugins/.

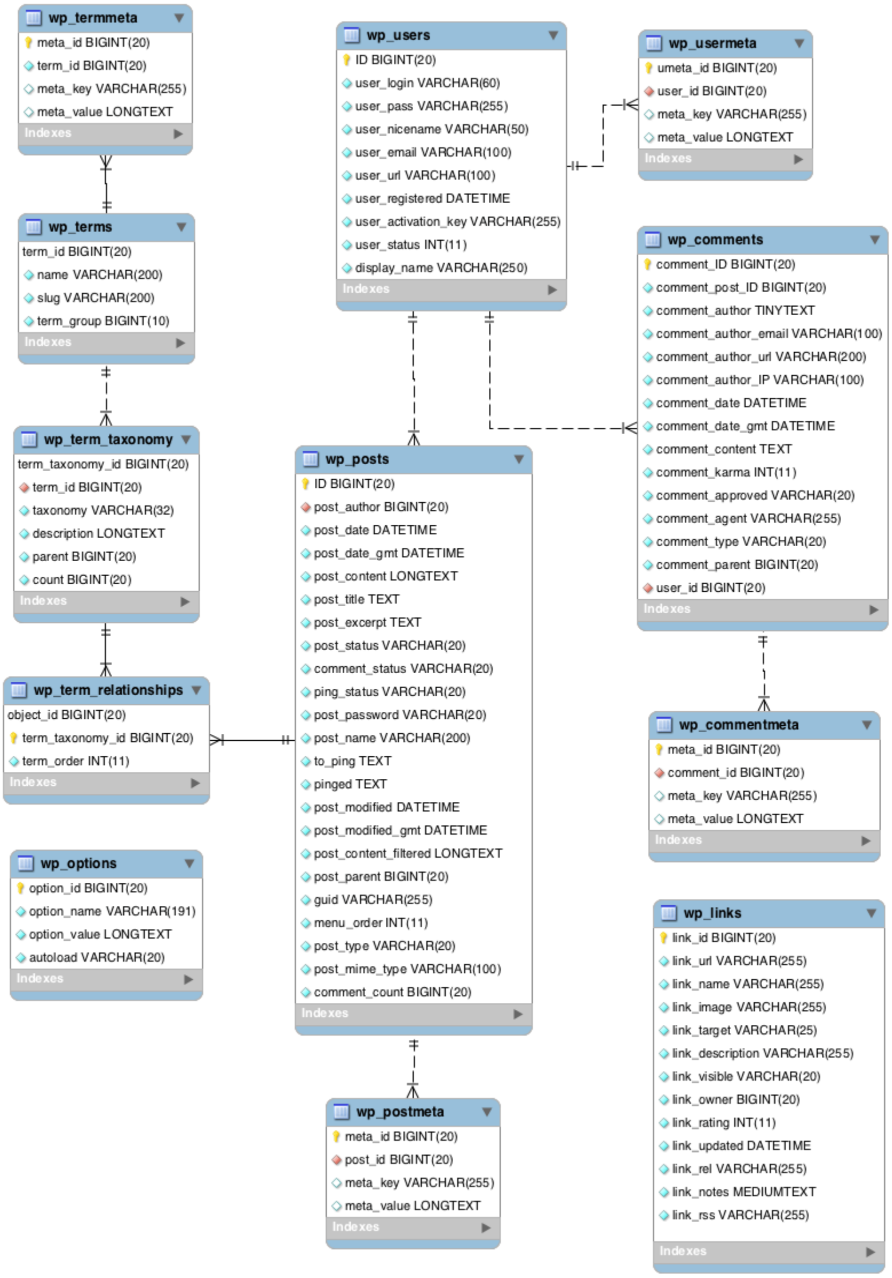


Fig. 4. Structural diagram of the database

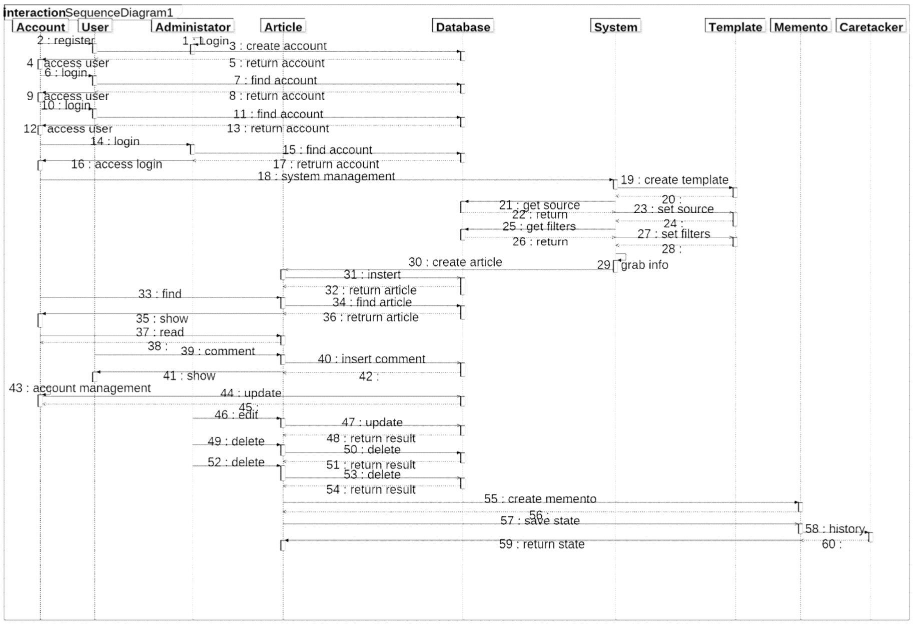


Fig. 5. Sequence diagram for autofill activity

The system does not need to create a copy of the database; instead, it will add the necessary fields and database data to MySQL on WordPress on the new hosting.

Thus, the system developed does not require specialized hardware, additional configuration, and deployment tools beyond the standard for such plug-ins.

In Fig. 6 shown a diagram of system deployment. The diagram shows that this system has three nodes: the program, the interface and the user.

When loading the start page the user has the opportunity to select any of the suggested menu items (Fig. 7).

The main menu of the site displays the main categories of articles that were generated by the system. From the main menu, the user can navigate to a specific category of articles of interest.

In the top menu, the user can change the language of the site and go to the social networks where the site is registered (see Fig. 8).

To get started with the automatic content filling system, you must log in to the site and log in to the WordPress admin panel, and the site administrator must fill in the login information at wp-auto/admin/.

After authorization from the side menu, you need to find the item “Plugins” and go to the page of installed WordPress plugins (Fig. 9), then find the “PRJ-Parser” on the installed plugins page and click the Activate activation link (Fig. 10).

After the plugin is successfully activated, a new item of the automatic content filling system “PRJ-Parser” will appear in the left menu (Fig. 11).

When selecting the PRJ-Parser menu item, the site administrator will be taken to the main page of the automatic content filling system, where he will be able to view the existing list of feeds registered to the systems when they were the last read and when the next content reading is scheduled. The administrator can also add his RSS feed to read it, delete the selected RSS feeds and related articles, delete only the selected RSS feeds, delete the articles related to the selected RSS feeds (Fig. 12).

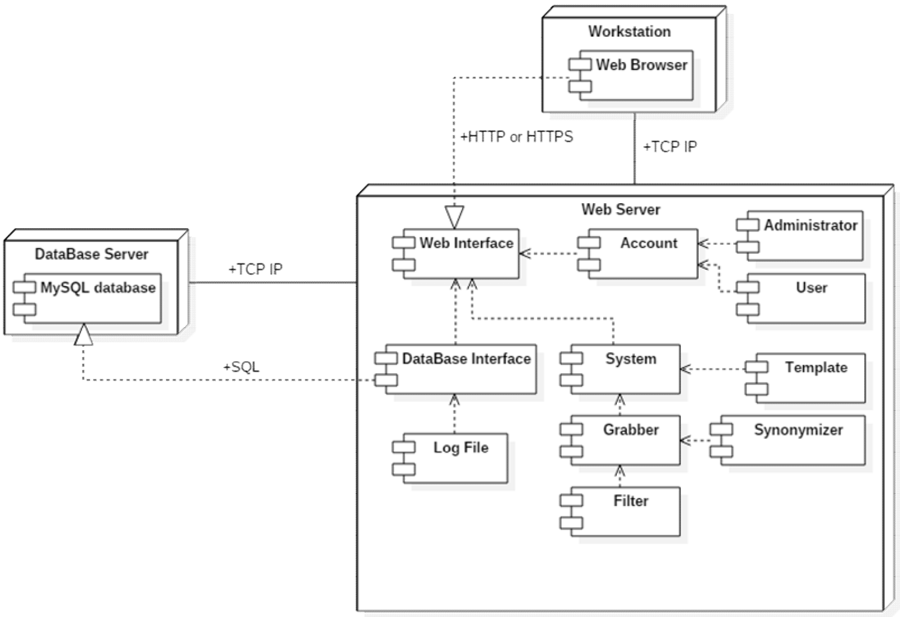


Fig. 6. System Deployment Diagram

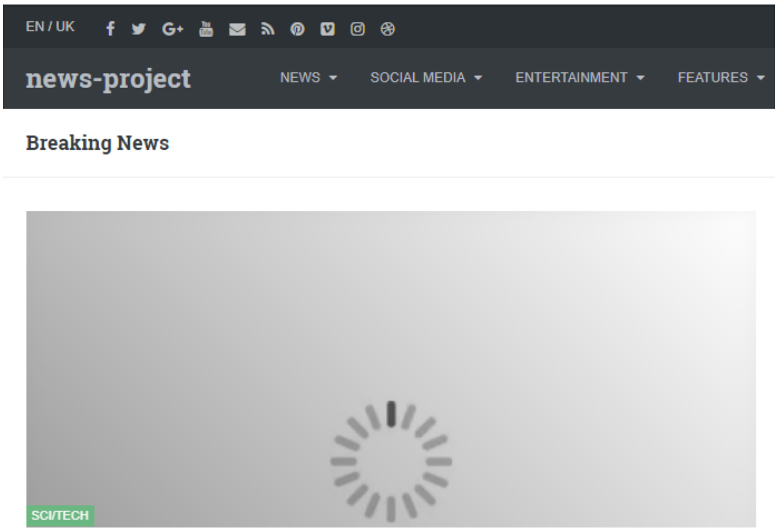


Fig. 7. The main menu of the site

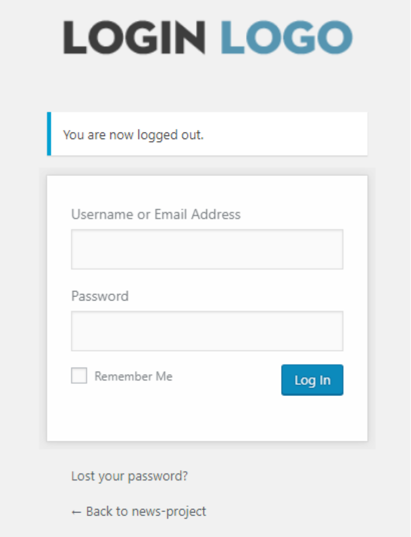


Fig. 8. Authorization to the admin panel.

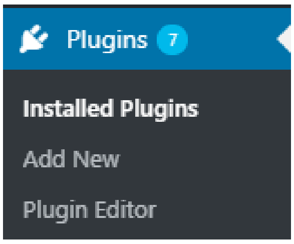


Fig. 9. Plugins menu item

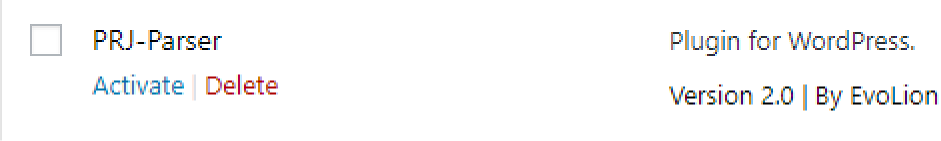


Fig. 10. Activate the plugin in the WordPress administration panel

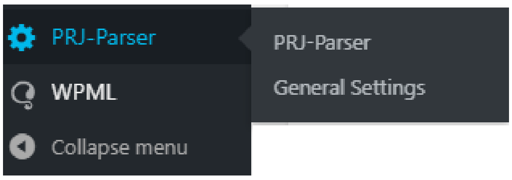
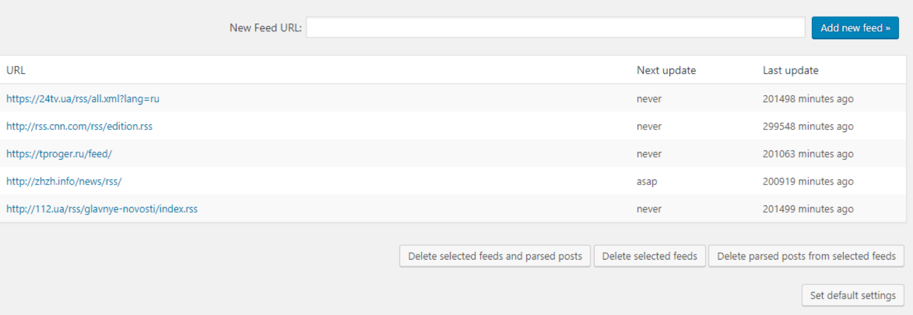


Fig. 11. The plugin is in the administration menu list

After entering the new RSS feed address and clicking the “Add a new feed” button, the administrator will go to the fine-tuning page of the new RSS feed (Fig. 12).


New Feed URL:
Add new feed »

URL	Next update	Last update
https://24tv.ua/rss/all.xml?lang=ru	never	201498 minutes ago
http://rss.cnn.com/rss/edition.rss	never	299548 minutes ago
https://tproger.ru/feed/	never	201063 minutes ago
http://zhzh.info/news/rss/	asap	200919 minutes ago
http://112.ua/rss/glavnye-novosti/index.rss	never	201499 minutes ago

Delete selected feeds and parsed posts
Delete selected feeds
Delete parsed posts from selected feeds
Set default settings

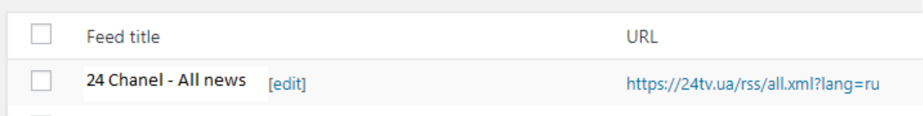
Fig. 12. The homepage of the automatic content filling system

On this page, an administrator can to:

- View and edit the name of the new RSS feed that will be displayed on the system homepage in the feed table;
- View URL of future RSS feed;
- Add categories to which articles from this RSS feed will be published;
- Enable universal reading mode by going to the full article page with additional filters;
- Specify on whose behalf the articles of this RSS feed will be published.
- Enable article tagging;
- Add your tags in the article while reading the current RSS feed;
- Activate the duplicate article check function;
- Activate the automatic reading function for a specified period;
- Specify the number of articles that will be published in a single RSS feed;
- Specify the status of articles to be read. Articles can be published immediately, left for review, saved to draft or hidden;
- Allow or deny users comments;
- Specify what date to use when publishing a read article; The date of publication may be the original date from the donor site or the date of reading by the automatic content filling system;
- Specify where to insert the article attachment. Attachments may be placed at the top of the article, bottom or not at all;
- Activate the thumbnail generation feature of the article. The thumbnail may be generated from the first image in the article, from the last image, read from the donor site or not generated at all;
- Activate the mandatory feature of the article thumbnail. If there is no article thumbnail, then the article will be removed from the site;
- Activate the article coding conversion feature. If RSS feed encoding is different from admin site encoding, then RSS feed will be converted to UTF-8 encoding;

- Enable local image saving from articles. If this feature is not activated, images will be sent to the donor site;
- Determine the range of minutes through which read articles will be randomly published in a single reading;
- Specify the source at the end of the article where the article was read;
- Insert the source at the end of the lifts;
- Activate the function of removing links from words and make links simple;
- Insert video resources into reading articles.

After saving the RSS feed settings, it will appear in the RSS feeds table on the main system page (Fig. 13).

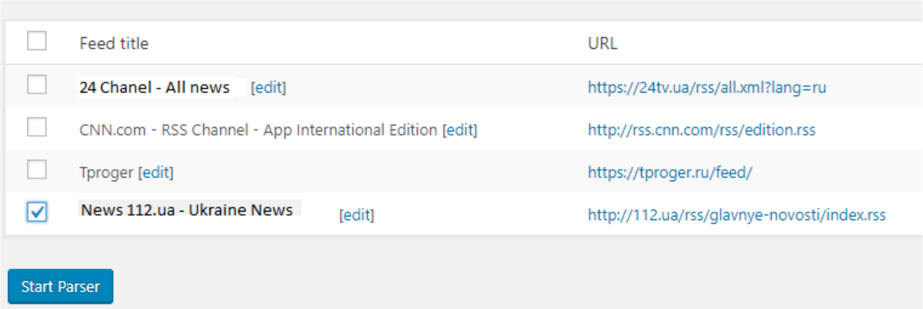


□	Feed title	URL
□	24 Chanel - All news [edit]	https://24tv.ua/rss/all.xml?lang=ru

Fig. 13. New RSS feed

The site administrator can change the selected settings at any time by clicking on the "edit" link (see Figure 13), and then a fine-tuning page with the last saved settings for a particular RSS feed will be opened (see Figure 12).

To read RSS feeds on existing articles, select the required RSS feeds and click the Start Parser button (Fig. 14). Reading will be done according to the settings of each of the RSS feeds marked.

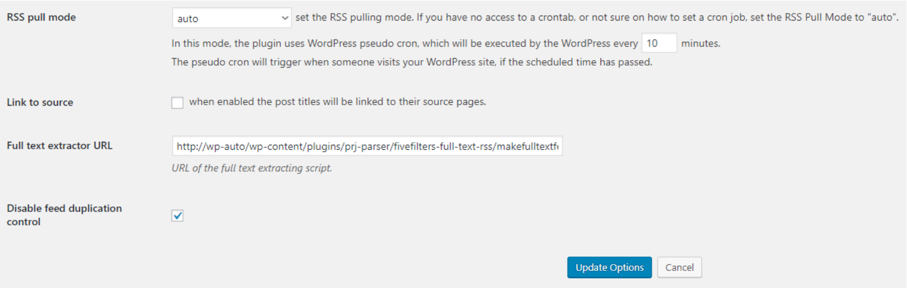


□	Feed title	URL
□	24 Chanel - All news [edit]	https://24tv.ua/rss/all.xml?lang=ru
□	CNN.com - RSS Channel - App International Edition [edit]	http://rss.cnn.com/rss/edition.rss
□	Tproger [edit]	https://tproger.ru/feed/
☑	News 112.ua - Ukraine News [edit]	http://112.ua/rss/glavnye-novosti/index.rss

Fig. 14. Reading RRS feeds

After the action is completed, the system will notify the successful reading and articles will be generated with the corresponding status, which was specified in the RSS feed settings.

Automatic filling systems also have a settings page (Fig. 15), where the site administrator can configure the automatic start of the plugin after a certain period of time, specify mandatory to leave a link to the source at the end of the article, the path to the full function reading libraries and the ability to disable check for duplicate articles.



RSS pull mode: auto — set the RSS pulling mode. If you have no access to a crontab, or not sure on how to set a cron job, set the RSS Pull Mode to "auto".

In this mode, the plugin uses WordPress pseudo cron, which will be executed by the WordPress every 10 minutes.

The pseudo cron will trigger when someone visits your WordPress site, if the scheduled time has passed.

Link to source: □ when enabled the post titles will be linked to their source pages.

Full text extractor URL: http://wp-auto/wp-content/plugins/prj-parser/fivefilters-full-text-rss/makefulltextf

URL of the full text extracting script.

Disable feed duplication control: ☑

Fig. 15. AutoFill Setup Page

4 Conclusions

In this paper, a database was designed and implemented following the requirements of a relational model, which ensures the storage and collective access to information of the autofill system and CMS WordPress data. Algorithms of system functioning have been developed, the order of interaction of classes during program code execution has been determined, and the application has been implemented. Most of the application is intended for the site administrator and has no user interface. The administrator has a plug-in configuration interface for the plug-in automation system, an interface for viewing and managing RSS feeds, as well as an interface for configuring RSS feeds.

In the future, this system can be improved by introducing new functionality and improving the algorithm for reading data.

References

1. Averianov, A.Ye.: Zastosuvannia informatsiinoi systemy upravlinnia kontentom veb-resursu dlia vedennia elektronnoi komertsii (Implementation of web content management information system for e-commerce). Formuvannia rynkovykh vidnosyn v Ukraini 9(160), 171–174 (2014)
2. Berko, A.Yu., Dorosh, V.M., Chirun, L.V.: Intelektualna systema upravlinnia kontentom saitiv elektronnoho biznesu (Intelligent Content Management System for E-Business Websites). Visnyk Natsionalnoho universytetu “Lvivska politekhnika” **715**, 13–23 (2011)
3. Chyrun, L.V.; Vysotska, V.A.: Zastosuvannia kontent-analizu tekstovoi informatsii v systemakh elektronnoi komertsii (Application of content analysis of text information in e-commerce systems). Visnyk Natsionalnoho universytetu “Lvivska politekhnika” **689**, 332–347 (2010)
4. Korobchinsky, M.V., Chirun, L.B., Vysotska, V.A., Kondratiev, E.A.: Osoblyvosti formuvannia ta analizu kontentu internet-hazety muzychnykh novyn (Of content formation and analysis features in online newspaper of music news). Radio Electronics, Computer Science, Control 4, 139–150 (2017)

5. MacDonald, M.: WordPress: The Missing Manual. O'Reilly Media, Sebastopol (2014)
6. Morozov, A.V., Kuzmenko, O.V., Danilchenko, A.A.: Osnovy veb-rozrobky dlia Wordpress ta Yii (Web Development Basics for Wordpress and Yii). ZhDU, Zhytomyr (2018)
7. Sanders, W.: Learning PHP Design Patterns. O'Reilly Media, Sebastopol (2013)
8. Vysotska, V.A., Chirun, L.B., Chirun, L.V.: Unifikovani metody opratsiuvannia informatsiinykh resursiv u systemakh elektronnoi kontent-komertsii (Unified methods of processing information resources in systems of electronic content commerce). Naukovi pratsi [Chornomorskoho derzhavnoho universytetu imeni Petra Mohyly] **213**(201), 13–23 (2013)
9. Williams, B.: WordPress Plugin Development. Wiley Publishing Inc., Indianapolis (2011)

Developing a Mini Smart House model

Nadiia Balyk[0000-0002-3121-7005], Svitlana Leshchuk[0000-0001-6600-7940]
and Dariia Yatsenyak[0000-0002-8427-5532]

Ternopil Volodymyr Hnatiuk National Pedagogical University,
2, Maxyma Kryvonosa Str., Ternopil, 46027, Ukraine
{nadbal, leshchuk_so, yatsenyak_dv}@fizmat.tnpu.edu.ua

Abstract. The work is devoted to designing a smart home educational model. The authors analyzed the literature in the field of the Internet of Things and identified the basic requirements for the training model. It contains the following levels: command, communication, management. The authors identify the main subsystems of the training model: communication, signaling, control of lighting, temperature, filling of the garbage container, monitoring of sensor data. The proposed smart home educational model takes into account the economic indicators of resource utilization, which gives the opportunity to save on payment for their consumption. The hardware components for the implementation of the Mini Smart House were selected in the article. It uses a variety of technologies to conveniently manage it and use renewable energy to power it. The model was produced independently by students involved in the STEM project. Research includes sketching, making construction parts, sensor assembly and Arduino boards, programming in the Arduino IDE environment, testing the functioning of the system. Research includes sketching, making some parts, assembly sensor and Arduino boards, programming in the Arduino IDE environment, testing the functioning of the system. Approbation Mini Smart House researches were conducted within activity the STEM-center of Physics and Mathematics Faculty of Ternopil Volodymyr Hnatiuk National Pedagogical University, in particular during the educational process and during numerous trainings and seminars for pupils and teachers of computer science.

Keywords: Smart home, Mini Smart House, STEM-projects, Smart-technologies, C++, Internet of Things, technology of management.

1 Introduction

The trend of IT development in recent years is Smart-technologies. They are now widely implemented in many industries, in the home and in education. As a result, the modern teacher receives many tools that make the learning process interesting and creative. The development of the Internet of Things (IoT) is an additional opportunity in this aspect, a network concept that contains many devices with built-in transmitters of their physical parameters. The widespread use of these household appliances was predicted at the beginning of the twentieth century, in particular by the eminent physicist Nikola Tesla [4]. In the early 21st century, the number of devices connected to the Internet exceeded the Earth's population. Today, the IoT sphere is one of the major global trends. Almost all devices known in the everyday life become part of the Internet and as a result perform new functions. No wonder this industry is considered the driving force of the 4th Industrial Revolution, which is now underway in the world. Therefore, forming an IoT expert – the person who creates the future – is an important educational task.

Today, the issue of modernization of the educational process is more urgent than ever. In the context of research, they can be filled through the implementation of STEM-projects, the use of Smart-technologies, exploring the possibilities of the Internet of Things [2].

Smart home technology embodies all of the above concepts. By "smart home" can be understood a system that provides security and resource conservation (including comfort) for all users. In the simplest case, it should be able to recognize and respond to specific situations occurring in the home: one of the systems can control the behavior of others using pre-built algorithms. In addition, the automation of several subsystems provides a synergistic effect for the whole complex [14]. With the increasing computing power of gadgets, many smart home technologies and the Internet of Things have been standardized. Also for them the basic rules and recommendations for the construction of the finished product at the level of both the system as a whole and the individual components were defined.

2 Research apparatus

The problem of research is related to the necessity of introducing in the educational process relevant to modern trends the methods and content of training.

The purpose of the project is to develop a smart home model, design and create a Mini Smart House.

Achieving the goal of the research is possible by solving the following tasks:

1. Analysis of the conceptual apparatus in the field of Internet of Things;
2. Designing the smart house training model and choosing the hardware components that will implement the model;
3. Project development through programming of its modules in Arduino IDE environment;
4. Implementation, testing, debugging of some components of the project.

The object of the study is the Internet of Things technologies.

The subject of the study is a smart home educational model and its implementation in the form of a Mini Smart House.

To achieve this goal, we used a set of research methods: theoretical – analysis of scientific and technical sources, generalization, modeling of information processes occurring in the "smart home"; empirical: observations, analysis of the experience of using IoT technologies; practical methods for software development and testing.

The study was tested within the framework of the STEM-Center of the Physics and Mathematics Faculty of the Volodymyr Hnatyuk TNPU, in particular in the educational process and during numerous trainings and seminars for students and teachers of informatics.

3 Results of the study

3.1 Analysis of the basic concepts of the study

Smart home is a system that provides security, resource conservation and comfort for all users [13]. In the simplest case, it should be able to recognize specific situations in the home, and respond to them by the developed algorithms.

The term "smart home" does not have a clear definition, and therefore is used when referring to any system with automated control of the devices, which simplifies the life of a person and increases his level of comfort. The beginning of the story of the "smart home" can be considered 1961, when Joel Solomon and Ruth Rodale Spira invented and patented a special device for smooth regulation of light – dimmer [5].

The first household electronic automation system was called the "home computer of the Echo IV", which in 1966 became the first analogue of a "smart home" [15]. The term "smart home" in 1984 was proposed and introduced by the American Developers Association [6]. It was then that prices for electrical appliances began to decline, which made it possible to build high-functionality offices.

In the future, scientists and engineers went from theory to practice, introducing more and more objects using this technology. An important feature and property of "smart home", which distinguishes it from other ways of organizing living space, is that it is the most progressive concept of human interaction with living space, when a person sets the desired environment with one command, and the automation, in accordance with external and internal conditions, sets and monitors the operating modes of all engineering systems and electrical appliances.

All functional features of such a building can be divided into three categories [16]:

- household functions;
- entertainment;
- protection and technological security.

This system does not require many computers and connections. As experience shows, you can make your home "smart" by your own efforts without making dramatic changes. Thus, it is an ordinary house or apartment (also, it can be an industrial object, a shopping mall, etc.), equipped with a "smart" system that does absolutely any whim, or any desire of its owner and "decides" most household tasks. In such a house it is not only pleasant to live, but comfortable, safe, profitable.

Nowadays, in the age of digitalization, the components of the "smart home", technologies of cloud computing and IoT have considerable pedagogical potential as an object and a learning tool [3; 7; 9; 10; 11; 12; 17].

3.2 Designing a Smart Home Model

Functionality, style, comfort, safety are far from being a complete list of what a smart home can do. IoT technologies are implemented in the concept of the "smart home" training model, which should provide convenient management of basic household appliances and the use of renewable power supplies. In our model of "smart home" we propose the following components:

- a control center (in the form of a tablet or console), which records and interprets data from sensors;
- motion, smoke, flooding, opening windows or doors, light, humidity, temperature;
- automatic water taps;
- temperature regulators for batteries;
- readers of indicators of counters;
- video intercom;
- voice assistants (optional).

All of these components have to independently receive data from the sensors and work according to the developed algorithms. Accordingly, the entire process of their operation will be subject to control and management from mobile devices via the Internet. As can be seen from Fig. 1, our model has three levels: commands, management and communications.

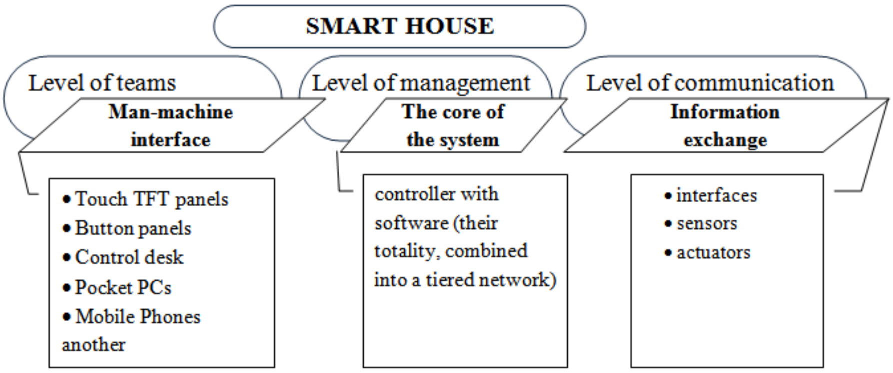


Fig. 1. Educational model of smart home

Mini Smart House embodies a combination of technical, engineering, design techniques, computer and software engineering (computer networks, C programming), and demonstrates practical results. The introduction of such projects into the educational process enables to acquire the skills of modeling the respective processes and to implement similar technologies in real life. Pupils or students will learn a self-created system that ensures the safety, comfort, economy and efficiency of managing their own home and is scaled and customized, ensuring efficiency in managing smart home technology. This integration of academic subjects and the implementation of cross-curricular links will help to prepare the modern specialist.

In our “Smart House” model, we can distinguish the following subsystems:

- communication;
- alarm system;
- lighting control;
- climate control;
- monitoring of sensor data;
- cleanliness control (filling the trash can).

In order to ensure their functioning in practice, appropriate software modules for all components were developed and optimal development tools were selected (see description in subsection 3.3) based on the selected hardware elements for our model (see Table 1).

Table 1. Calculation of the cost of hardware components of the prototype

Name	Model	Quantity	Price UAH
Temperature and humidity sensor	DHT11	1	29
Sound sensor	KY-037	1	29
Hercon sensor for door opening	MC-38	1	49
Arduino pin extension module	I2C на PCF8574T	1	27
LCD 1602 module for Arduino	1602	1	65
Real time clock	DS1302-MOD	1	17
Buzzer	KY-012	1	18
The engine		1	50
Bluetooth module	HC-06	1	100
Ultrasonic distance sensor	HC-SR04	1	37
Conductors		30	1
LED		3	1
Arduino	Uno	2	209-894
Payless Layout Board	MB-102	2	49
Solar battery	RF136X110-3	1	120
Plastic		1	40
Plywood			120
Accessories for doors and windows			10
Laser cutting			100
Other supplies			20

Our Mini Smart House model should take into account the economic indicators of resource use. By analyzing the relevant data, you can realize the savings of utility bills in a smart home.

As the study [8] shows for smart home is characterized by:

- saving resources:
- electricity:
  - in the lighting system – up to 60%;
  - in the climate control system – up to 40%;
- gas and water – up to 40%;
- reduced operating costs:
  - service savings – up to 20%:
  - cost savings on staff;
  - multiple increase of equipment life;
- prevention of emergencies;
- increase of profitability and investment attractiveness:
- more favorable insurance conditions;
- when renting – the basis for raising the rent.

With the hardware components that implement the model, we have selected the components of the open Arduino hardware and software project. Table 1 lists them and provides a cost estimate for creating a Mini Smart House.

3.3 Practical implementation of the model

The practical implementation of the model involves the creation of a layout smart house. The model is a sketch project of the house, which gives an idea of the artistic and stylistic decisions of the building, features of its planning.

First of all, we drew a sketch of the house on paper. Then, according to the sketch, the drawing was done in CorelDraw.

After completing a detailed drawing of the layout, you need to select the material for its construction. Many different materials are suitable for making a model of home, but it is most appropriate to use foam, wood or solid cardboard. We have decided that wood is the most appropriate material for the Mini Smart House prototype. It is easily machined, reliable and durable. So, we chose the plywood. The prepared drawings for the laser cutting machine made it possible to cut out the details.

According to the developed drawings, all structural elements are manufactured. Careful quality work made it easy to connect all the details with each other.

After preparation of all the details, a prototype of the house was assembled, fixed with PVA glue and with hot gluing. Moving parts are attached to the curtains that are screwed onto the screws. In general, the prototype looks like this (see Fig. 2).

The defining function of any SMART-system is to respond to the environment, the parameters of which are measured using sensors, signals, communications and other integrated elements. The received data is processed through the implementation of program code. C++ programming language is used for programming in the Arduino IDE.

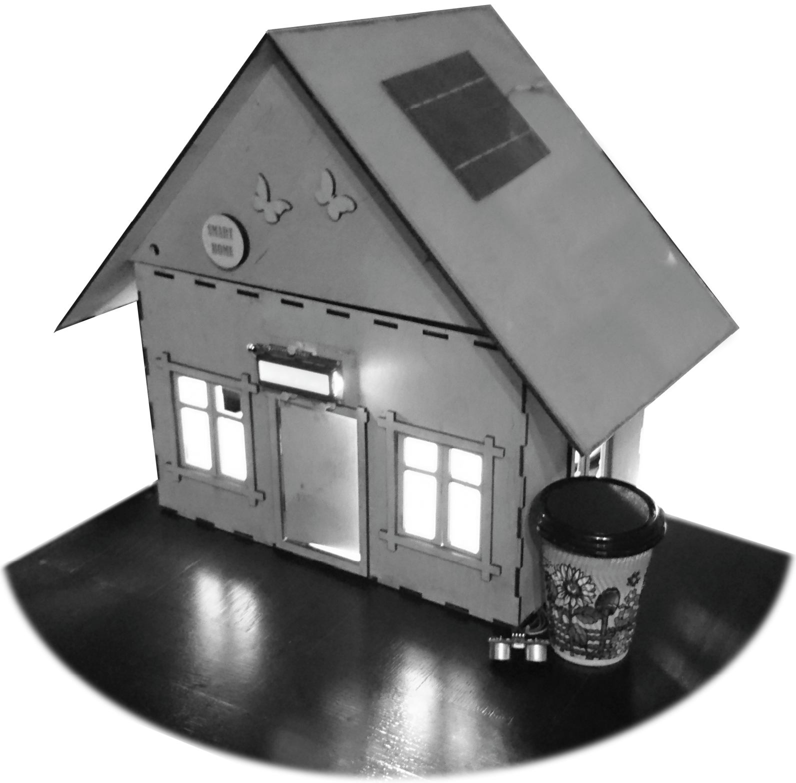


Fig. 2. View of the “Mini Smart House” prototype

Created code from the Arduino IDE environment we upload to the Arduino Uno board. So we program microcontrollers. Arduino Uno is a widely used open source microcontroller board based on the ATmega328P microcontroller. It includes everything you need to conveniently work with your microcontroller: 14 digital inputs / outputs (6 of which can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz resonator, a USB connector, a power connector, connector for programming within the scheme (ICSP) and the reset button [1].

Three communication protocols can be used to communicate with the Arduino Uno board: ZigBee, Wi-Fi and Bluetooth. Given the instructional model purpose of the Mini Smart House, we have chosen the Bluetooth protocol that will allow you to connect to the system using your smartphone or tablet. The disadvantages of Bluetooth certainly include a small distance of signal propagation. However, in our model, the connection within 10 meters is stable and allows to save low power consumption, compact size and relatively low cost of components. Yes, a low-power transmitter consumes only 0.3 mA in standby mode and averages 30 mA during data exchange. In addition, Bluetooth provides encryption of data transmitted using an 8- to 128-bit effective key and one-way or two-way authentication.

Fig. 3 shows a diagram of how we connected a Bluetooth module to an Arduino UNO board.

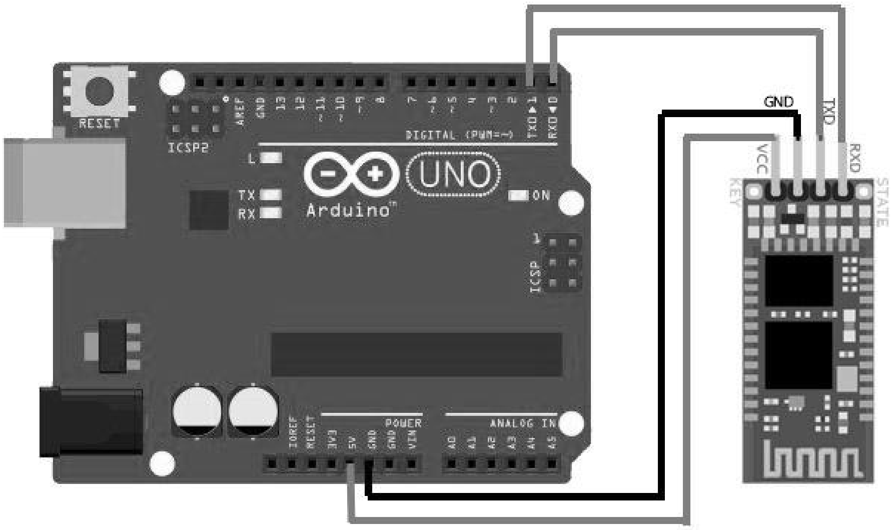


Fig. 3. Bluetooth HC-06 connection scheme

The software to connect the Bluetooth module to the smartphone using the following code:

```
#include "alarm.h"
#include "claplight.h"
#include <SoftwareSerial.h>
int ledpin=12;
int BluetoothData;
void bluetooth_setup() {
  Serial.begin(9600);
  Serial.println("Bluetooth On please press 2-5");
  pinMode(ledpin,OUTPUT);
}
void bluetooth_loop() {
  if (Serial.available())  {
    BluetoothData=Serial.read();
    if (BluetoothData == 50) { clap_bl = true;
      last_bl = false;
      Serial.println("Clap Light  On! "); }
    if (BluetoothData == 51) { clap_bl = false;
      Serial.println("Clap Light  Off! ");}
    if (BluetoothData == 52) { alarm_bl = true;
      Serial.println("Alarm On! ");  }
    if (BluetoothData == 53) { alarm_bl = false;
      Serial.println("Alarm Off! ");}
  }
}
```

So, via Bluetooth, you can control the use of household appliances such as lighting, ventilation, alarms and get sensor data: temperature, humidity, landfill.

An example of the implementation of security control Mini Smart House is the development of alarm systems. To do this, we use a Hercon sensor for door opening and a piezo speaker. When the door is opened, sensor sends an electrical signal to the Arduino, which includes a loud alarm sound. You can turn it off or on using Bluetooth connectivity. The corresponding connection diagram of the mentioned components is shown in Fig. 4.

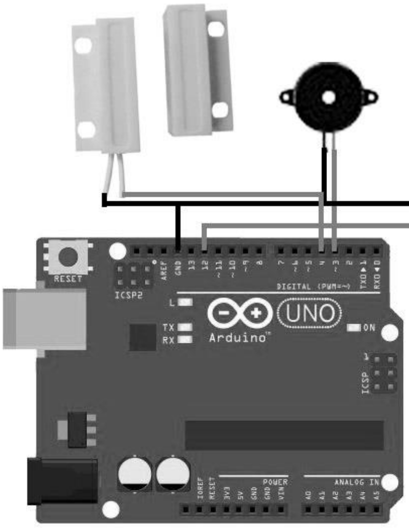


Fig. 4. Scheme of the alarm system

When the alarm system is on, sensor sends an electrical signal when the door is opened on the Arduino board, from which the sound is output to the speaker. The event processing module is as follows:

```
const int buzzer = 3;
const int sensor = 4;
int state; // 0 close - 1 open switch
int alarm_delay = 500;
int alarm_timer = 0;
bool alarm_bl = true;

void alarm_setup(){   pinMode(sensor, INPUT_PULLUP); }
void alarm_loop() {
  if(!alarm_bl)   {    noTone(buzzer);    return;  }
  if(alarm_timer < alarm_delay)
  {    alarm_timer  += 1;    return;  }
  else
  alarm_timer = 0;
  state = digitalRead(sensor);
  if(state==HIGH) {tone(buzzer, 1000); alarm_timer=0;}
  else {    alarm_timer += 1;        noTone(buzzer);    }
}
```

The lighting control subsystem provides convenient switching on and off of the light without coming into the switch. To do this, the user should just clapped his hands. The hardware components of this subsystem are a sound sensor and an LED. An alternative way is to control the lighting from your smartphone. The hardware connection diagram is shown in Fig. 5.

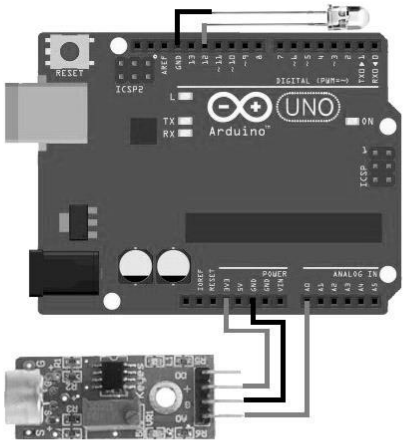


Fig. 5. Scheme of the lighting control subsystem

The processing of subsystem events is carried out by program code.

```
int ledPin = 12;
int threshoid= 20;
int volume;
int v = 0;
int knowckSensor = A0;
int sensorreading = 0;
int ledState = LOW;
bool clap_bl = true;
bool last_bl = false;
int claplight_delay = 50;
int claplight_timer = 0;
void claplight_setup(){
  Serial.begin(9600);
  pinMode(ledPin, OUTPUT);
}
void claplight_loop(){
  if(clap_bl == false) {
    last_bl = clap_bl;
    digitalWrite(ledPin, LOW);
    return;
  }
  else if(last_bl == false && clap_bl == true)
    digitalWrite(ledPin, HIGH);
  last_bl = clap_bl;
  v = analogRead(knowckSensor);
    if(v >= threshoid)     {
    if ((sensorreading == 0))    {
      digitalWrite(ledPin, HIGH);
      Serial.println("Knock!");
      sensorreading = 1;
    }
    else    {
      digitalWrite(ledPin, LOW);
      Serial.println("No!");
      sensorreading = 0;
    }
    //v = 0;
    delay(50);
  }
}
```

The climate control subsystem uses a temperature sensor and a fan. When the temperature rises to a certain point, the Arduino controller supplies current to the fan, and when the temperature drops the fan shuts off. The schematic diagram of the climate control subsystem is presented in Fig. 6.

In our model of smart home implemented the output of sensor data and date and time on the screen. The corresponding subsystem contains a screen, a clock module, sensors. The scheme of their connection is shown in Fig. 7.

Additionally, sensor data is sent in addition to the smartphone.

The Mini Smart House controls the filling of the waste container with the help of an Arduino board and a proximity sensor. It uses acoustic ultrasonic radiation to determining the distance to the object. This contactless proximity sensor provides high accuracy and stability of measurements. Measurement results are virtually unaffected by solar radiation and electromagnetic noises. Sensor captures the fill level of the waste container and sends a signal to the Arduino. If the fill rate is 80% or more, the message “Trash filled” will be sent to the phone.

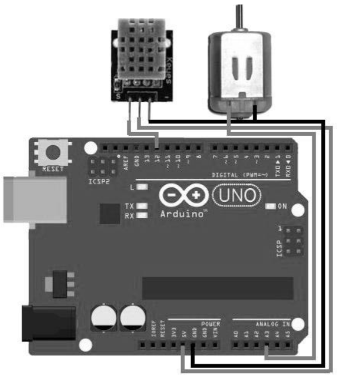


Fig. 6. The scheme of the climate control subsystem

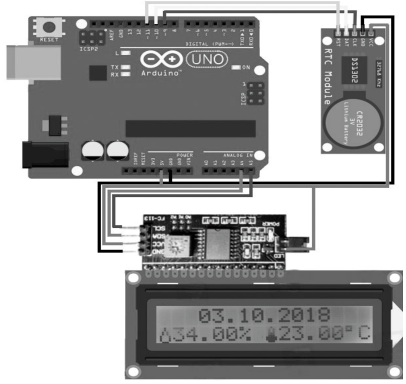


Fig. 7. Schematic diagram of the monitoring subsystem

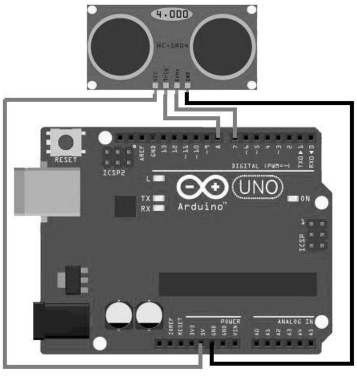


Fig. 8. Scheme of the purity control subsystem

Software implementation of the module of the fill level of the waste container subsystem is:

```
#define trigPin 8
#define echoPin 7
#define led 6
int dumpDelay = 0;
void dump_setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(led, OUTPUT);
}

void dump_loop() {
  dumpDelay += 100;
  if(dumpDelay < 2000)
    return;
  dumpDelay = 0;
  long duration, distance;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(3000);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  Serial.println(distance);
  if (duration < 300 || (duration >= 500 || duration <=
0))     digitalWrite(led, HIGH);
  else    digitalWrite(led, LOW);
}
```

Nowadays, energy-saving technologies are becoming widespread. In the “Mini Smart House” prototype, we use a RF136X110-3 on 5V solar battery connected to a Power Bank battery that nourish our system.

4 Conclusions

The analysis of the possibilities of modern technologies in education (STEM, Smart, Internet of Things) has given theoretical and methodological basis for the design of the smart home educational model and development on its basis Mini Smart House.

We have created a finished product that combines technical, engineering, design methods, computer and software engineering (computer networks, C++ programming) and demonstrates practical result. Using the approaches described above, we can model relevant processes and implement similar technologies in real life. As a result, students can learn a self-created system that provides security, comfort, economy and efficiency in managing their own home, scalable and customizable, ensuring efficiency in managing smart home technologies.

The Mini Smart House project is of practical importance to students and teachers. It can be used in the teaching of various topics in school courses in computer science and physics.

References

1. Arduino Uno Rev3. Arduion Official Store. https://store.arduino.cc/arduino-uno-rev3 (2019). Accessed 15 Sep 2019
2. Balyk, N., Barna, O., Shmyger, G., Oleksiuk V.: Model of Professional Retraining of Teachers Based on the Development of STEM Competencies. CEUR Workshop Proceedings **2104**, 318–331 (2018)
3. Bremer, A.: Diffusion of the “Internet of Things” on the world of skilled work and resulting consequences for the man–machine interaction. Empirical Research in Vocational Education and Training **7**, 8 (2015). doi:10.1186/s40461-015-0021-9
4. daCosta, F., Henderson, B.: Rethinking the Internet of Things: A Scalable Approach to Connecting Everything. Apress, Berkeley (2013)
5. Dimmer. https://en.wikipedia.org/w/index.php?title=Dimmer&oldid=916345104 (2018). Accessed 19 Sep 2019
6. Dingli, A., Seychell, D.: Smart Homes. In: The New Digital Natives, pp. 85-101, Springer, Berlin, Heidelberg (2015). doi:10.1007/978-3-662-46590-5_7
7. Hu, Y., Tilke, D., Adams, T., Crandall, A.S., Cook, D.J., Schmitter-Edgecombe, M.: Smart home in a box: usability study for a large scale self-installation of smart home technologies. Journal of Reliable Intelligent Environments **2**(2), 93–106 (2016). doi:10.1007/s40860-016-0021-y
8. Innovatsiinyi proekt “Rozumnyi budynok” Smart (House Innovative Project). https://otherreferats.allbest.ru/management/00006537_0.html (2008). Accessed 15 Sep 2019
9. Kiv, A.E., Soloviev, V.N., Semerikov, S.O.: CTE 2018 – How cloud technologies continues to transform education. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018), Kryvyi Rih, Ukraine, December 21, 2018. CEUR Workshop Proceedings **2433**, 1–19. http://ceur-ws.org/Vol-2433/paper00.pdf (2019). Accessed 10 Sep 2019
10. Koper, R.: Conditions for effective smart learning environments. Smart Learning Environments **1**, 5 (2014). doi:10.1186/s40561-014-0005-4
11. Markova, O.M., Semerikov, S.O., Striuk, A.M., Shalatska, H.M., Nechypurenko, P.P., Tron, V.V.: Implementation of cloud service models in training of future information technology specialists. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018), Kryvyi Rih, Ukraine, December 21, 2018. CEUR Workshop Proceedings **2433**, 499–515. http://ceur-ws.org/Vol-2433/paper34.pdf (2019). Accessed 10 Sep 2019
12. Modlo, Ye.O., Semerikov, S.O., Nechypurenko, P.P., Bondarevskyi, S.L., Bondarevska, O.M., Tolmachev, S.T.: The use of mobile Internet devices in the formation of ICT component of bachelors in electromechanics competency in modeling of technical objects. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018), Kryvyi Rih, Ukraine, December 21, 2018. CEUR

Workshop Proceedings **2433**, 413–428. http://ceur-ws.org/Vol-2433/paper28.pdf (2019). Accessed 10 Sep 2019

13. Oliveira, L., Mitchell, V., May, A.: Smart home technology – comparing householder expectations at the point of installation with experiences 1 year later. Personal and Ubiquitous Computing 23, 1-14 (2019). doi:10.1007/s00779-019-01302-4
14. Oteafy, S.M.A., Hassanein, H.S.: Resource Re-use in Wireless Sensor Networks: Realizing a Synergetic Internet of Things. Journal of Communications **7**(7), 484–493 (2012). doi:10.4304/jcm.7.7.484-493
15. Prysiazhniuk, A.: Rozumnyi budynok – z choho vin skladaietsia ta chy potriben vam (A Smart Home – What It is Made of and Whether You Need It). Na chasi. https://nachasi.com/2018/06/25/smart-house-faq (2018). Accessed 15 Sep 2019
16. Saushkyn, O.O.: Proekt rozumnyi budynok (The Smart Home Project). http://economstroy.com.ua/mvoprosyiotvety/2303-proektrozumnuybudunok.html (2012). Accessed 15 Sep 2019
17. Spirin, O., Oleksiuk, V., Oleksiuk O., Sydorenko, S.: The Group Methodology of Using Cloud Technologies in the Training of Future Computer Science Teachers. CEUR Workshop Proceedings **2104**, 294–304 (2018)

Using 3D modelling in design training simulator with augmented reality

Alla Kompaniets[0000-0001-8781-9577], Hanna Chemerys[0000-0003-3417-9910]
and Iryna Krasheninnik[0000-0001-6689-3209]

Bogdan Khmelnytsky Melitopol State Pedagogical University,
20, Hetmanska Str., Melitopol, 72300, Ukraine
chemeris@mdpu.org.ua

Abstract. The article is devoted to the theoretical consideration of the problem and the use of innovative technologies in the educational process in the educational establishment of secondary education in the process of studying the school course of computer science. The main advantages of using educational simulators in the educational process are considered, based on the new state standard of basic and complete general secondary education. Based on the analysis of scientific and methodological literature and network sources, the features of the development of simulators for educational purposes are described. Innovative tools for simulator development have been investigated, as augmented reality with the use of three-dimensional simulation. The peculiarities of using a simulator with augmented reality when studying the topic of algorithmization in the course of studying a school computer science are considered. The article also describes the implementation of augmented reality simulator for the formation of algorithmic thinking skills by students, presents the results of development and describes the functionality of the software product. In the further prospects of the study, it is planned to conduct an experimental study to determine the effectiveness of the use of software development in the learning process.

Keywords: elementary school, educational process, computer science, algorithms, training simulator, augmented reality, three-dimensional modelling.

1 The problem statement

The modern world is filled with a variety of different digital technologies that are already an integral part of society. One of the most promising is the augmented and virtual reality technologies that have recently brought to the fore. There are many options for using these technologies in training – from elementary virtual tours of famous museums to the opportunity to see and take part in a historical event of any period with your own eyes. It is the augmented reality technology that will serve as a new stage in the development of modern education, allowing us to reach a new level of visualization. The main feature of this technology is the attachment to the real world, which is so lacking in the already implemented virtual reality [32]. A great number of scientists are focusing on the special possibilities of using augmented reality technology in the study of computer science [3; 22; 31]. This is of particular socio-pedagogical relevance in the current context of reforming and improving the efficiency and quality of education against the backdrop of the introduction of modern progressive information technology. As a result of intensive reform of education in Ukraine, the concept of which is set out as the "New Ukrainian School", the content and methodology of teaching computer science is substantially updated, which in turn requires improvement of the process of its teaching. It should be noted that the basics of algorithmization in the school course of computer science are studied starting from the 2nd class, and given the complexity of learning the basics of the algorithmization of this age group, it is necessary to create effective learning tools and put them into practice. In this regard, there is an urgent need to develop educational software for the study of algorithms, taking into account the age characteristics of these students, which are augmented reality simulators.

2 The aim of article

The aim of article is to research and analyze information about the problem of using three-dimensional simulation technologies for the development of augmented reality simulator to study the topic of algorithmization in the school computer science course, as well as to describe the functionality of the software product.

3 The results of research

3.1 Training simulator as an effective training tool

Classical tutorials and textbooks do not always fully meet the requirements of "digital learners" who seek to catch up everywhere and immediately. The need to reform the current education system is now being talked about by everyone, even those whose field of activity is far from education. Many teachers pay attention to the fact that without the use of modern technologies it is impossible to make a significant breakthrough in solving the problems of reforming education and improving the qualitative parameters of training of future specialists. In this case, informatization is one of the key conditions that determine the further development of the economy, science and culture in general [6]. Undoubtedly, the correct and optimal use of information and communication technologies allows to obtain a number of advantages in improving the efficiency of the educational process, namely: providing visibility of the educational material and combining different ways of perceiving educational information; increasing motivation due to novelty and variety of training using computer equipment; increasing the student's time to work with the study material individually; the presence of objective control and self-control in the correct preparation of test tasks [7, p. 128].

World experience has shown that in recent decades, many workspaces (simulators) have been widely used in many fields of human activity, especially in the learning process [10; 15; 16; 28; 36]. For example, a number of foreign publications are devoted to the use of augmented reality applications for smartphones and tablets in the study of computer science [34], biology [17; 25], chemistry [24], astronomy [8], physics [11], mathematics [12] and other subject areas [20; 21]. Simulators allow to expand resources and training opportunities. These resources include additional visibility and accessibility of training material.

One of the important conditions for enhancing the development of theory and practice is the creation of a system of tasks, their consistency and diversity [19, p. 178]. A computer simulator is a software-hardware training and control tool for training and developing practical skills. A training simulator for educational purposes is a training device that simulates circumstances, actions, creates a situation close to real [23, p. 94]. In a narrower sense, it is a training program for developing the skills and competences of a particular activity, as well as developing related skills.

The training simulators are based on the use of a specific training task. Its essence is that in a short period of time, using different methods of working with educational material, you can quickly teach students to remember it.

3.2 Using an augmented reality in educational software

One way to improve education technologies is to use virtual and augmented reality systems, 3D e-learning systems [4]. Virtual and augmented reality are three-dimensional human-computer interaction technologies implemented using modern ICT tools. Augmented reality technology allows real-time image capture devices to recognize special tags (markers), as well as their position in space, with the subsequent introduction of imaginary objects into real space. Such devices may be used for smartphones, tablets or special augmented reality goggles [30].

The prospect and pace of implementation of virtual and augmented reality technologies indicate that the learning tools developed on their basis will become an integral part of education at all levels of education, and their role will increase significantly both within traditional and e-education [14; 35]. The use of augmented and virtual reality training tools will significantly shorten the training time, improve the quality of training and enhance the practical orientation of the learning process. The main idea of using virtual and augmented reality is to increase the possibilities of human interaction with the environment [18]. The use of augmented and virtual reality training tools will significantly shorten the training time, improve the quality of training and enhance the practical orientation of the learning process. The main idea of using virtual and augmented reality is to increase the possibilities of human interaction with the environment [1, pp. 1–2]. For the education system, virtual and augmented reality are promising in terms of using these technologies as innovative learning tools that will retain the student's interest. Today, these are mostly simulators and simulators. Such training tools allow you to gain knowledge and skills by working with them virtually. The use of augmented reality tools not only visualize the training material, but also combine virtual reality with the physical environment, to face the 3D world, which means to make it brighter, more memorable, even more memorable to even closer modern students. Tutorials, booklets, and other educational material may contain embedded markers that, when scanned with augmented reality devices, provide the student with additional information presented in a multimedia format. Instead of remaining passive recipients, students become active, able to interact with their learning environment [5].

3.3 Features of three-dimensional modelling for training simulator with augmented reality

On the prospect of using 3D modelling technologies as a promising direction for use in virtual reality and augmented reality, in the process of developing gameplay, and for modelling objects for educational purposes, the number of subjects emphasized [2, pp. 17–18].

Modern computer technologies allow create the simulators that include multimedia components – computer animation, audio and video effects, as well as virtual and augmented reality. The use of these tools enhances the sense of reality when working with the simulator and opens up new opportunities in the learning process. The augmented reality simulator's information model is best depicted in terms of the process of sequential and cyclical accumulation of knowledge, highlighting the dynamics of virtual 3D space development as a constructive value of the simulator. The efficiency of using the simulator will depend, first of all, on the saturation of perception of knowledge throughout the action. The saturation of perception can be increased by creating a more capacious and realistic virtual 3D image of augmented reality.

Augmented reality when using such simulators imposes virtual objects, characters, filters or other effects on the surrounding world [27, pp. 4–5]. This is done by playing objects for the smartphone. To create such objects, three-dimensional modelling is often used to ensure the best integration of the object into the environment.

3.4 Design and visualization of training simulator

Nowadays, in connection with the relevance and effectiveness of the use of technologies of virtual and augmented reality for pedagogical purposes, the direction of pedagogical design of teaching aids based on virtual and augmented reality technologies has been determined. Despite the fact that augmented reality technology is really relevant and has several advantages over traditional learning, there are a number of problems in its implementation in education. The main problems with the introduction of augmented reality technology in education are the lack of ready-made tools [13]. In particular, this also applies to the school computer science course, in which augmented reality technology is more commonly used in the process of studying the section of computer science course "Computer Architecture" as the most illustrative [9; 29]. To develop a simulator with augmented reality, we chose the topic "Algorithms" of the school course in computer science. Consider a developed simulator to study the topic of algorithmization when studying a school computer science course. We define the developed simulator (by analogy with [6]) as an imitation learning tool, which is a specialized didactic complex of technical and software tools that implements the interaction of the student and the tools in the learning process. The simulator is designed to develop and improve students' skills and abilities of logical and algorithmic thinking.

Augmented reality simulator to study the topic of algorithmization in the study of computer science course was developed in the Unity 3D development environment. Unity3D is a platform for computer games and applications development. Unity3D allows the creation of applications running on more than twenty different operating systems, including personal computers, game consoles, mobile devices, Internet applications and others [33]. The Unity3D program supports the Vuforia SDK. The Vuforia SDK is a software suite that includes augmented reality platforms and augmented reality software developer (SDK) tools to use AR technologies on mobile devices: tablets, smartphones and AR glasses for iOS, Android and UWP (Windows). The Vuforia SDK is integrated with the Unity3D game engine, which greatly facilitates the development of AR applications. Because AR reality technology is integrated by developers into most mobile operating systems, the mobile AR app supports Android 4.1.x and above, iOS 9 and above, and Windows 10 UWP. By default, Vuforia works with Image Recognition. Image recognition, also called Tracker Recognition or Image Tracking, is the process by which a software recognizes a predetermined image and renders certain content on top of it.

To model an assistant character based on the analysis and classification of 3D modelling tools [26, pp. 77–79] we used a graphical editor for three-dimensional Autodesk Maya modelling.

To perform the multi-level tasks, cards were prepared from which the student must assemble a flowchart that would fit the task. Cards are labels or tracker images that a student operates in the real world. Examples of cards are shown in the Fig. 1.

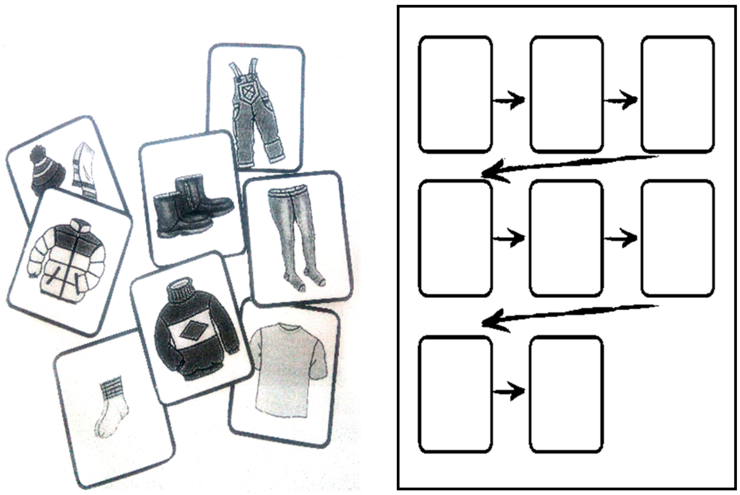


Fig. 1. Examples of tag cards for drawing flowcharts and a playing field

The next step in creating the program is to add the Image Target tag and ARcamera. ARCamera is a virtual camera that simulates the smartphone camera. There is also a prefab Camera managing game stage Unity. ImageTarget is an image tag. The result of adding a label to the project scene is shown in the

Fig. 2.

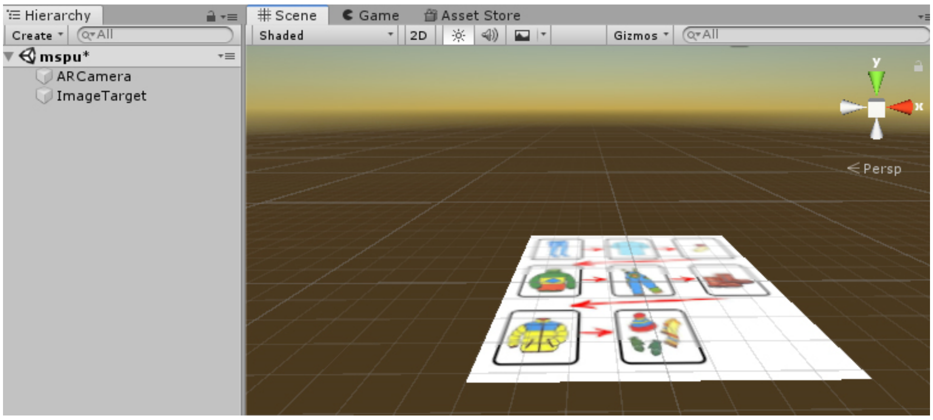


Fig. 2. Placing a label on the Unity stage

Next, a 3D object of the helper character with emotion is added to the scene, which will correspond to the faithful almost wrong decision of the task. The result of adding an object to the scene is shown in the Fig. 3. After all the labels and objects were placed on the Unity stage, the project was saved in apk-format for further testing on an Android smartphone.

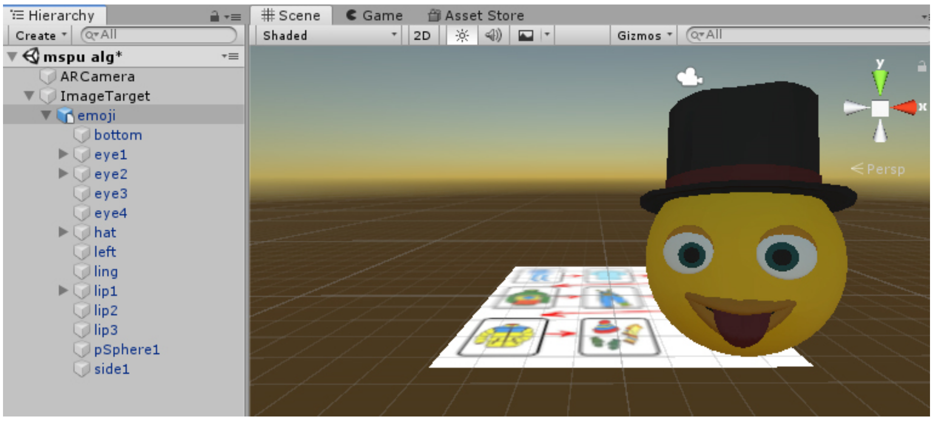


Fig. 3. Adding a 3D object to the Unity scene

The result is a mobile augmented reality application that is used as a simulator for studying the computer science course on “Algorithms and Performers”. To use the bottom app, a student must have a smartphone or tablet on the Android platform. The algorithm of the application is shown in Fig. 4.

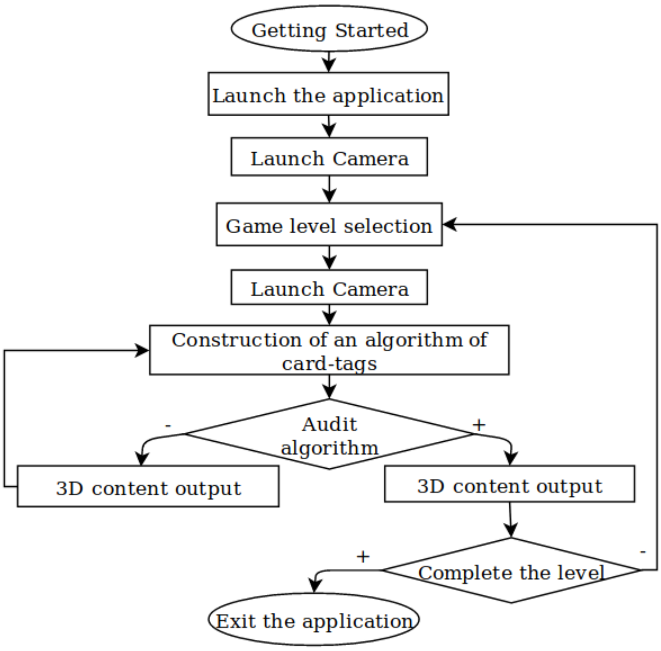


Fig. 4. The algorithm of the application

After starting the application is initialized mobile camera, after which the user selects an educational level. After selecting the required educational level, the user goes to the game levels directory. Then the level at which the user builds a chain of card tags starts. If the chain is constructed correctly, a positive 3D character appears on the screen, if not – the 3D character reports that the algorithm is incorrectly constructed and the level needs to be redone. After completing the level, the user can continue the game to the next level, or shut down the application.

Thus, by pointing the camera of the smartphone to the solution of the problem on the topic of algorithmization, the student sees the reaction of the assistant character, which illustrates the correctness of the solution of the problem: the emotion of sadness corresponds to the wrong solution of the problem, while the joy illustrates the correct solution.

The survey, conducted among parents of pupils who participated in the project activity using an augmented reality simulator, showed the following results. Parents appreciated such educational experience. Parents expressed a positive attitude towards using mobile devices for learning purposes. The thought of using mobile devices has changed from negative to positive or at least neutral in the process of participating in a project. All parents noticed increased interest in children, increased motivation to learn new material. Many parents have used links to the right programs and engaged with their child outside the institution. The overall result of the project activity was satisfied with both teachers, children, and parents, who expressed their willingness to participate in such activities in the future.

4 Conclusions and prospects for further research

We have analysed scientific and methodological sources for determining the effectiveness of using educational simulators. The analysis of modern approaches, covered in foreign and domestic sources on the experience of using augmented reality in educational simulators, is conducted. Described is a developed educational simulator with augmented reality, aimed at studying the topic of algorithmization in the school course of computer science for children of primary school age. The augmented reality simulator will help teachers, tutors, and even parents to increase the effectiveness of learning a computer science course on the topic of algorithmization. Further research prospects are aimed at expanding the educational tasks of the simulator and the development of playing cards, as well as the experimental verification of the effectiveness of using the developed software product in the conditions of basic and complete general secondary education.

References

1. Billinghurst, M.: Augmented Reality in Education. New Horizons for Learning http://www.solomonalexis.com/downloads/ar_edu.pdf (2002). Accessed 17 Aug 2019
2. Chemerys, H., Osadcha, K., Osadchyi, V., Kruhlyk, V.: Increasing the Level of Graphic Competence Future Bachelor in Computer Sciences in the Process of Studying 3D Modeling. CEUR Workshop Proceedings **2393**, 17–28 (2019)
3. Chen, H., Feng, K., Mo, C., Cheng, S., Guo, Z., Huang, Y.: Application of augmented reality in engineering graphics education. In: 2011 IEEE International Symposium on IT in Medicine and Education, pp. 362–365. IEEE Press, New York (2011). doi:10.1109/ITiME.2011.6132125
4. Cheng, K.-H., Tsai, C.-C.: Affordances of Augmented Reality in Science Learning: Suggestions for Future Research. Journal of Science Education and Technology **22**(4), 449–462 (2013). doi:10.1007/s10956-012-9405-9
5. Digital Union | Office of Distance Education and eLearning. The Ohio State University. https://odee.osu.edu/digital-union (2019). Accessed 11 Sep 2019
6. Donskoy, A.N.: Trenazheryi na baze EVM dlya operativnogo personala TETs (Computer-based simulators for operational personnel of TPP). Energetik 5, 26–30 (1995)
7. Fedotova, M.A.: Formirovanie samostoyatelnoy deyatelnosti studentov v didakticheskoy kompyuternoy srede (Formation of students' independent activity in didactic computer environment). Informatics and Education **10**, 126–128 (2006)
8. Fleck, S., Simon, G.: An Augmented Reality Environment for Astronomy Learning in Elementary Grades: An Exploratory Study. In: Proceedings of the 25th Conference on l'Interaction Homme-Machine (IHM '13), pp. 14–22. ACM, New York (2013). doi:10.1145/2534903.2534907
9. Freitas, M.R., Ruschel, R.C.: What is happening to virtual and augmented reality applied to architecture? In: Stouffs, R., Janssen, P., Roudavski, S., Tunçer, B. (eds.) Open Systems: Proceedings of the 18th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA 2013), pp. 407–416. The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), Hong Kong, and Center for Advanced Studies in Architecture (CASA), Department of Architecture-NUS, Singapore (2013)

10. Freitas, R., Campos, P.: SMART: a SysteM of Augmented Reality for Teaching 2nd Grade Students. In: Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction, vol. 2, pp. 27–30. BCS Learning & Development Ltd., Swindon (2008)
11. Hruntova, T.V., Yechkalo, Yu.V., Striuk, A.M., Pikilnyak, A.V.: Augmented Reality Tools in Physics Training at Higher Technical Educational Institutions. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 33–40. http://ceur-ws.org/Vol-2257/paper04.pdf (2018). Accessed 30 Sep 2019
12. Kaufmann, H.: Dynamic differential geometry in education. Journal of Geometry and Graphics **13**(2), 131–144 (2009)
13. Kirillov, D.Y., Ilyina, L.A.: Sozdanie 3D-panoram (Creating of 3D-Panoramas). In: Informatics and Computer Technics: a collection of scientific works, pp. 96–97. I.N. Ulyanov Chuvash State University, Cheboksary (2016)
14. Lamanauskas, V., Vilkonis, R.: A New Teaching / Learning Platform Based on Augmented Reality Technology: Didactical Scenario 1. In: Information and Communication Technology in Natural Science Education – 2007 (Proceedings of International Scientific Practical Conference), pp. 99–114. Publishing House of Šiauliai University, Šiauliai (2007)
15. Lee, K.: Augmented reality in education and training. TechTrends **56**(2), 13–21 (2012). doi:10.1007/s11528-012-0559-3
16. Li, Y.: Augmented Reality for remote education. In: 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), vol. 3, pp. 187–191 (2010). doi:10.1109/ICACTE.2010.5579661
17. Marzouk, D., Attia, G., Abdelbaki, N.: Biology Learning Using Augmented Reality and Gaming Techniques. In: Proc. of World Cong. on Multimedia and Computer Science, pp.79–86 (2013)
18. Maslennikova, A.E.: Podgotovka studentov universiteta k mezhlichnostnomu poznaniyu sredstvami avtodidaktiki (Preparation of University Students for Interpersonal Knowledge by Autodidactics). Dissertation, Magnitogorsk State University (2005)
19. Melnyk, Yu.: Zadacha yak zasib formuvannya doslidnitskih umin uchniv na urokah informatiki (Problem as a means of forming students' research skills in computer science lessons). Zbirnyk naukovykh prats Umanskoho derzhavnoho pedahohichnoho universytetu 3, 177–182 (2011)
20. Merzlykin, O.V., Topolova, I.Yu., Tron, V.V.: Developing of Key Competencies by Means of Augmented Reality at CLIL Lessons. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 41–52. http://ceur-ws.org/Vol-2257/paper05.pdf (2018). Accessed 30 Sep 2019
21. Mintii, I.S., Soloviev, V.N.: Augmented Reality: Ukrainian Present Business and Future Education. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 227–231. http://ceur-ws.org/Vol-2257/paper22.pdf (2018). Accessed 30 Sep 2019
22. Modlo, Ye.O., Semerikov, S.O., Nechypurenko, P.P., Bondarevskyi, S.L., Bondarevska, O.M., Tolmachev, S.T.: The use of mobile Internet devices in the formation of ICT component of bachelors in electromechanics competency in modeling of technical objects. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018), Kryvyi Rih, Ukraine, December 21, 2018. CEUR

Workshop Proceedings **2433**, 413–428. http://ceur-ws.org/Vol-2433/paper28.pdf (2019). Accessed 10 Sep 2019

23. Morze, N.V., Barna, O.V., Vember, V.P., Kuzminska, O.H., Sarazhynska, N.A.: Informatika: pidruch. dlya 5 kl. zagalnoosvit. navch. zakladiv (Informatics: Textbook for the 5th grade). Osvita, Kyiv (2013)
24. Nechypurenko, P.P., Starova, T.V., Selivanova, T.V., Tomilina, A.O., Uchitel, A.D.: Use of Augmented Reality in Chemistry Education. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 15–23. http://ceur-ws.org/Vol-2257/paper02.pdf (2018). Accessed 30 Nov 2018
25. Ng'ambi, D., Lombe, A., Johnston, K., Kabanda, S.: Podcasting for mobile learners: using ubiquitous technologies to enhance learning in large classes. In: Montebello, M., Camilleri, V., Dingi, A. (eds.) Proceedings of mlearn 2010: 10th world conference on mobile and contextual learning, pp. 256–262. University of Malta, Valetta (2010)
26. Osadcha, K.P., Chemerys, H.Yu.: Three-dimensional modeling tools in the process of formation of graphic competence of the future bachelor of computer science. Information Technologies and Learning Tools **62**(6), 70–85 (2017). doi:10.33407/itlt.v62i6.1713
27. Pasaréti, O., Hajdú, H., Matuszka, T., Jámbori, A., Molnár. I., Turcsányi-Szabó, M.: Augmented Reality in Education. In: INFODIDACT Informatics Methodology Conference. https://people.inf.elte.hu/tomintt/infodidact_2011.pdf (2011). Accessed 10 Sep 2019
28. Phon, D.N.E, Ali, M.B., Halim, N.D.A.: Collaborative augmented reality in education: A review. In: 2014 International Conference on Teaching and Learning in Computing and Engineering, pp. 78–83. IEEE Press, New York (2014). doi:10.1109/LaTiCE.2014.23
29. Redondo, E., Fonseca, D., Sánchez, A., Navarro, I.: New strategies using handheld augmented reality and mobile learning-teaching methodologies in architecture and building engineering degrees. Procedia Computer Science **25**, 52–61 (2013). doi:10.1016/j.procs.2013.11.007
30. Shapovalov, Ye.B., Bilyk, Zh.I., Atamas, A.I., Shapovalov, V.B., Uchitel, A.D.: The Potential of Using Google Expeditions and Google Lens Tools under STEM-education in Ukraine. In: Kiv, A.E., Soloviev, V.N. (eds.) Proceedings of the 1st International Workshop on Augmented Reality in Education (AREdu 2018), Kryvyi Rih, Ukraine, October 2, 2018. CEUR Workshop Proceedings **2257**, 66–74. http://ceur-ws.org/Vol-2257/paper08.pdf (2018). Accessed 30 Nov 2018
31. Şimşek, M., Toklu, S., Özsaraç, H., Zavrak, S., Başer, E., Takgil, B., Kanbur, Z.: An Augmented Reality Application for Computer Engineering Curriculum. Bilişim Teknolojileri Dergisi **10**(1), 47–51 (2017)
32. Syrovatskyi, O.V., Semerikov, S.O., Modlo, Ye.O., Yechkalo, Yu.V., Zelinska, S.O.: Augmented reality software design for educational purposes. In: Kiv, A.E., Semerikov, S.O., Soloviev, V.N., Striuk, A.M. (eds.) Proceedings of the 1st Student Workshop on Computer Science & Software Engineering (CS&SE@SW 2018), Kryvyi Rih, Ukraine, November 30, 2018. CEUR Workshop Proceedings **2292**, 193–225. http://ceur-ws.org/Vol-2292/paper20.pdf (2018). Accessed 17 Aug 2019
33. Unity - Manual: Unity User Manual (2019.2). https://docs.unity3d.com/Manual/index.html (2019). Accessed 10 Sep 2019
34. Utku K., Durmus K., Suleyman A.Y.: An Augmented Reality Based Mobile Software to Support Learning Experiences in Computer Science Courses. Procedia Computer Science **25**, 370–374 (2013). doi:10.1016/j.procs.2013.11.045
35. Vilkonienė, M., Lamanauskas, V., Vilkonis, R.: Pedagogical Evaluation of the Teaching/Learning Platform Based on Augmented Reality Technology: the Opinion of

Science Teachers. In: Information and Communication Technology in Natural Science Education – 2007 (Proceedings of International Scientific Practical Conference), pp. 181–210. Publishing House of Šiauliai University, Šiaulia (2007)

36. Zagoranski, S., Divjak, S.: Use of augmented reality in education. In: The IEEE Region 8 EUROCON 2003. Computer as a Tool, vol. 2, pp. 339–342. IEEE Press, New York (2003). doi:10.1109/EURCON.2003.1248213

Automation in software testing, can we automate anything we want?

Bohdan Oliinyk[0000-0003-3670-2605] and Vasyl Oleksiuk[0000-0003-2206-8447]

Ternopil Volodymyr Hnatiuk National Pedagogical University,
2, Maxyma Kryvonosa Str., Ternopil, 46027, Ukraine
{olijnyk_bm, oleksyuk}@fizmat.tnpu.edu.ua

Abstract. The article considers the problem of test automation software. The authors analyze testing tasks that can be automated. They also cite cases where the use of automation is inappropriate. The key factors of using automation are time and cost savings. According to the authors, the advantages of automated tests are: the ability to check the latest changes in the application (regression testing), speed of execution, saving the time of testers, the ability to create self-tests by developers. The disadvantages of automatic tests are: insufficient reliability, need for support, fewer errors detected, a false sense of product quality. The following processes are identified, which can be automated: background processes, file logging, database entry, registration and payment systems, load tests, data entry operations, long-end-to-end scripts, checking complex mathematical calculations, checking correct search. The article provides statistics on the use of programming languages for developing automated tests. A comparative analysis of ready-made software products for automated testing is offered. Based on research analysis and experience, the authors believe that human intelligence is always required to validate the program. So, the authors justify the need to perform a manual and automated test.

Keywords: Software Testing, Quality Assurance, Manual Software Testing, Automation of Software Testing.

1 Introduction

Testing is a very important stage in software development life cycle (SDLC). This process ensures that most errors are found. However, one of the principles of testing is that all errors cannot be found – exhaustive testing is impossible [8]. Therefore, IT experts believe that the end product (software or hardware) works as it should, if it is as close as possible to certain functionality, reliability, performance.

As the research and practical experience shows, the software development lifecycle takes place during the following stages – analysis, design, development, implementation, testing, deployment, and support (Fig. 1) [5].

As the speed of software development is increasing today, there is a need for quality and timely testing. But some of the testing tasks are too time-consuming to perform their manually. In addition, more and more companies are moving to work according to agile methodology. An agile process begins with initiating the project; in the activities that follow, the software will be developed and deployed into the user environment through multiple iterations. In most agile methodologies, maintenance does not appear as a separate phase, but is rather performed through further iterations of the main development phases [2]. That's why automation is becoming more important than ever [14].

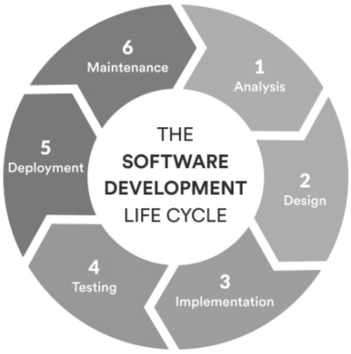


Fig. 1. The Software Development Life Cycle

Even 5 years ago, testing automation in Ukraine was economically unattractive.

In article [2] Nikolay Alimenkov writes: "… look who supports conferences for automated testers? And almost none. In large companies, this does not fit into the "strategic development plan". Small is simply not interesting. As a result, automated testing in Ukraine lives on thanks to the enthusiasm of the automators themselves" [2].

Today the situation has changed. Almost every company wants to have a automated tester, no matter if it's a big company or a small one. Today, automation is not only relevant in startups. As a result, writing self-tests is usually not time consuming. In general, startups may not even have test cases. The growing numbers of communities, forums, channels where automated testers discuss their problems, can also testify to the great development of automation.

2 Relevance of the study

Automated testing is now evolving rapidly. This is due to the fact that automated, debugged processes will require less money, which is one of the key factors in IT. As experience in the early days of the introduction of automation shows, it requires more material costs, while fewer tests are performed instead. However, over time, it is possible to significantly increase the number of test runs without significant investment. The popularity of automated testing is also explained by the fact that more and more "manual" testers want to develop in programming. Automation is a next stage of their development. "The job was pretty boring. We were just supposed to look at flagged differences and decide if they were a big deal or not. I wasn't happy with that. I wanted to know what caused the differences, so I started digging a little deeper", – John Sonmez talks about it in the article "Going from QA (or Another Technical Role) to Software

Developer" [12]. In the paper reports an empirical study on the relationship between code visibility and testability. The authors claim for manual testing, code visibility does not necessarily affect test code coverage and fault detection rate. However, for automated testing using testing tools, low code visibility often leads to low code coverage and poor fault detection rate [10].

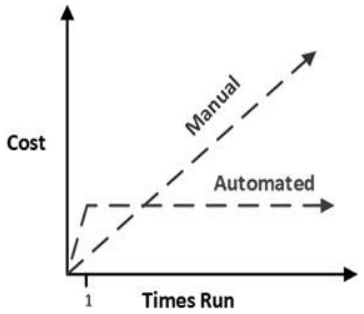


Fig. 2. The dependence of the cost of the test on the number of executions in manual and automated testing

Based on our own experience, we can say that the process depends largely on the project. One can often hear among testers that manual testing is not a tedious process, but an approach is important.

Systematic and automated approaches have shown capable of reducing the overwhelming cost of engineering automated tests. Industrial success cases have been openly reported and academic interest continues to grow as observed by the increasing number of researchers in the field. While there exist various trends on evolving the automation in software testing, the provision of sound empirical evidence is still needed on such area [6].

The authors of [1] describes a strategy to develop automated testing suites to assess the correctness of consent and revocation. This strategy is based on a formal language in order to provide rigorous and unambiguous consent and revocation specifications, and comprises of two novel procedures that facilitate the process of eliciting testing requirements for privacy properties and creating automated privacy-testing suites.

The purpose of the article. The dynamic pace of development of automated testing requires the answer to the question "can we all automate?" What is the role of manual testers in modern software development processes?

3 Automated testing, its advantages and disadvantages

3.1 Types of testing

Software testing can be performed for different purposes. According to the purpose there are such types of testing [11]:

- Functional Testing is a type of software testing whereby the system is tested against the functional requirements/specifications.
- Usability Testing is a type of software testing done from an end-user’s perspective to determine if the system is easily usable.
- Security Testing is a type of software testing that intends to uncover vulnerabilities of the system and determine that its data and resources are protected from possible intruders.
- Performance Testing is a type of software testing that intends to determine how a system performs in terms of responsiveness and stability under a certain load.
- Regression testing is a type of software testing that intends to ensure that changes (enhancements or defect fixes) to the software have not adversely affected it.
- Compliance Testing is a type of testing to determine the compliance of a system with internal or external standards.

All these types of tests can be performed both manual and automated.

Researchers identify such phases of the software testing process [9]:

1. Preliminary Testing phase is conducted especially for testers to clarify the specification requirements of the customer. Preliminary testing is performed during the following steps: review requirements specification, prepare test plan, prepare software tool, prepare test environment, prepare test case, prepare test automation tool, determine acceptance test tool.
2. The testing phase is a separate phase which is conducted by a different test team after the implementation is completed.
3. User acceptance testing phase which provides for checks integration testing, test strategy document, integration testing signoff, repair and coordinate release.

Note that manual and automated testing can be used together at different stages of software quality verification.

Automated testing or testing automation is a method of testing software. The method involves the use of special software tools to control the performance of tests. Then the actual test results are compared with the predicted ones. Most operations are performed automatically, with little or no intervention by the test engineer. Automated testers write scripts (so-called automated test cases) that have a set of actions and checks. Properly written automated tests can have many benefits and can be very useful for the project and organization. However, there are some disadvantages of automated tests that you should also be aware of.

3.2 Features and benefits of automated testing

Regression testing. Automation testing is the most common for this type of testing. Regression testing is a type of testing aimed at checking changes made to an application or environment (debugging, code merging, migration to another operating system, database, web server), to confirm that existing functionality is still working. Regression can be both functional and non-functional tests. Typically, regression testing uses test cases written in the early stages of development and testing. That is, regression automated tests are performed at a predetermined time interval. They are usually downloaded after every successful compilation (in small projects) or every night or every week. This guarantees that the changes to the new version of the program do not damage the already existing functionality. In the paper [7], authors discuss the advantages and drawbacks of using UML diagrams for regression testing and analyze that UML model helps in identifying changes for regression test selection effectively.

Speed of execution. Automated verification scripts may take some time. However, it takes less time to complete them than the person who would perform these checks manually. Therefore, self-tests help to provide quick feedback to the development team.

Time saving for testers. Test automation frees up testers' time. Therefore, they may be more focused on exploring new features. Automated checks can be started automatically with minimal supervision, or without any supervision, or manually. Usually when you do not want to use automated tests there is a cyclical situation of lack of time (Fig. 3).

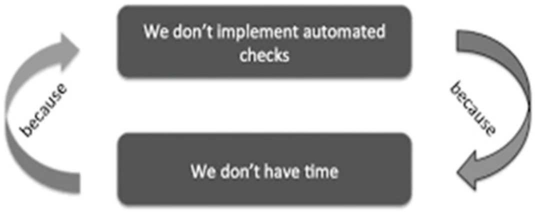


Fig. 3. Causal link lack of time and unwillingness to test automatically

Ability to create automated tests by developers. Automated tests are usually written in the same language as the product being tested. As a result, the responsibility for writing, conducting and performing tests becomes a shared responsibility. Everyone in the development team, not just testers, can contribute to the quality of the software.

3.3 Disadvantages of automated testing

False sense of product quality. For this reason, it is worth paying special attention to successfully passed automated tests. This is especially important for user-level (UI) or system-level functionality testing. The automated test only checks what is programmed to be tested.

All automated tests in the test suite may pass, but some mistakes may be not identified. The reason for this is that this test was not designed to detect these specific failures.

Insufficient reliability. Automated checks may not be successful due to many factors. For example, automatic checks can be broken by changing the user interface, shutting down a service, or having a network problem. These problems do not directly affect the program being tested, but may affect the outcome of automated tests.

Need for support. It should be understood that automated tests require maintenance. Automated checks are short-lived. Failure to update them will cause crashes. It is also possible that some checks are no longer relevant or that they do not correspond to new implementations of the software. These failures can affect the test results.

Writing an automated test case is not a one-time effort. To get the most out of automated tests, they need to be updated and up-to-date. This usually takes time, effort and resources.

Fewer errors detected than manual testing. Most errors are detected “accidentally” or during exploratory testing. This type of testing involves the simultaneous study of a software product, the design of tests and their execution. Its specificity is that at each exploratory testing session there is an opportunity to test the application in different ways.

On the other hand, automated checks always follow the set path, sometimes with the same test dataset. One can mention here one of the principles of testing – the “paradox of the pesticide”. It is that by performing the same tests over and over again, we face the fact that they find fewer and fewer errors. This is due to the development of the system, as a result of many found defects are corrected and old test cases no longer work. This in turn reduces the likelihood of finding new defects in the product.

“Test automation is not always testing”. Here, it is understood that testing is a research activity. Testing requires specific knowledge, a purposeful mind, and a willingness to learn the application. Unfortunately, many people are wrong about the importance of test automation. They get a test automation tool and want to get rid of all the “manual testers”. But this is not possible because testing is not simply about performing a set of predefined testing steps and comparing actual results with expected results. The most recent tasks are automated checks.

Ability to group failures into clusters. In the paper [10] present an approach to automatically detect passing and failing executions using cluster-based anomaly detection on dynamic execution data. The key hypothesis underlying the approach is that failures will group into small clusters whereas passing executions will group into larger ones.

A person’s intelligence is always required to validate the program. Here’s how Tommy Wyher says it: “Every day, we see hundreds of new apps and products out in the market. A lot of testing takes place before they are released to the public. Automated testing can speed up the process and is often seen as a replacement for manual testing. However, manual testing still has a critical role in the QA process. By dedicating your QA resources to only one of these approaches, you’ll miss many opportunities to improve quality. Using manual and automated testing together will lead to a higher quality, more stable product” [15].

3.4 Tests that can be automated

Let’s look at software components and processes that can be automated.

1. Hard to reach places in the system: background processes, file logging, database entry.

2. Frequently used functionality where there is a high risk of errors: payment systems, registrations, etc. Automation of critical functionality checks ensures fast errors, since the test takes an average of several minutes.
3. Load tests that test the functionality of a system with a large number of requests.
4. Template operations, including data searches, input of forms with many fields, checking their preservation.
5. Validation messages: fill in the fields with incorrect data and validation check.
6. Long end-to-end scenarios. For example, an online store scenario that involves: user registration, product detail page (PDP), shopping cart, shopping cart, product purchase, and confirmation of purchase.
7. Verification of data requiring accurate mathematical calculations, eg accounting or analytical processing.
8. Checking the correctness of the data search.

Although newer automated testing tools are emerging, it is still difficult to test the functionality of the user interface. It is also not possible to automate the testing of a new feature without at least one manual test run. Therefore, to say that you can automate everything, for now, will be an exaggeration.

Consider which tools are most often used to write automated tests.

Automated tests are written in various programming languages, the most common being Java. Figure 4 shows the distribution of automated programming languages in the world [16]:

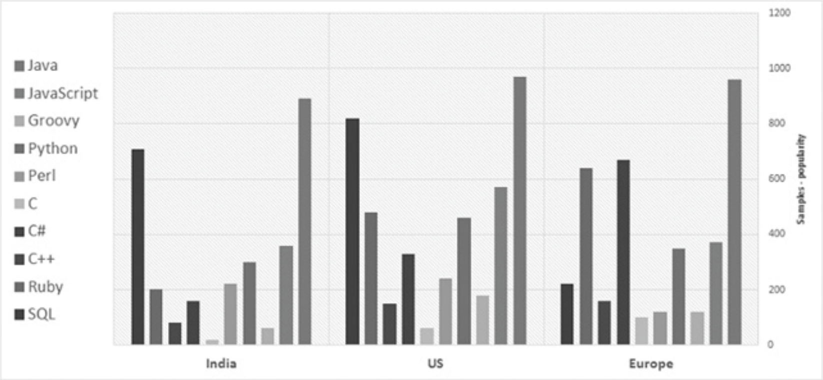


Fig. 4. Use programming languages to create automated tests

3.5 Some testing automation tools

In addition to programming languages, there are other automation tools available in software testing. Their comparative analysis is given in the Table 1 [3].

Table 1. Comparative of testing automation tools

Automation tools	**Selenium**	**Katalon Studio**	**Unified Functional Testing**	**Testcomplete**	**Watir**
Available since	2004	2015	1998	1999	2008
Application Under Test	Web applications	Web, Mobile applications	Web, Desktop and mobile applications	Web apps, Desktop, Mobile applications	Web applications
Pricing	Free	Free	~10000 USD	~100 USD	Free
Supported Platforms	Windows, Linux	Windows, Linux	Windows	Windows	Windows, Linux
Scripting Languages	Java, C#, Perl, Python, JavaScript, Ruby PHP	Java, Groovy	VBScript	VBScript, Python, JavaScript, Ruby PHP, C#, C++	Ruby
Programming Skills	Advanced skills needed to integrate various tools	No required. Recommended for advanced test script	No required. Recommended for advanced test script	No required. Recommended for advanced test script	Advanced skills needed to integrate various tools
Ease of installation and Use	Require advanced skills to install and use	Easy setup and use	Complex in installation. Need training to properly user the tool	Easy setup and use. Need training to properly user the tool	Require advanced skills to install and use

Each of these tools has its own peculiarities and scope. The most common tool for automation is Selenium. This is because it is free and scripts can be written in many programming languages, unlike other tools. Developed over 15 years ago, it has evolved over the next decade. It is now a web browser automation tool. In most cases, it is used to test Web applications, but this does not limit its scope.

JMeter is used for load testing. Initially, this application was intended to test the operation of the Apache Tomcat servlet container, which is essentially a web server. With the development of Jmeter, the user interface has been improved and additional features have been added that have made it an effective tool for performance testing and load testing of web applications.

It is impossible to say exactly what kind of tools will be popular in the future, as they are in rapid development. In our opinion, this will greatly simplify testing automation.

The authors of the article [4] recommend to automate the user interface according to the Mike Cohn scheme (Fig. 5). It shows the recommended proportion of tests to be implemented on each test level. The pyramid contains 3 levels:

1. Low-level tests are much faster by nature. Faster tests give you faster feedback. Faster feedback from tests execution allows to catch issues early on, saving huge amounts of costs.

2. Low-level tests are executed much earlier in the QA automation pipeline. Usually, unit tests are run before each commit test. If this is true, then testers prevent bugs and do not let them get into the project repository.
3. Low-level tests are much more stable than high-level ones.

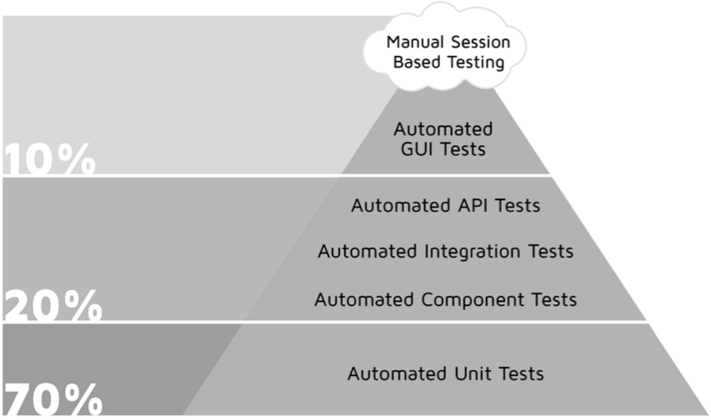

Manual Session Based Testing
10%
Automated GUI Tests
Automated API Tests
Automated Integration Tests
20%
Automated Component Tests
70%
Automated Unit Tests

Fig. 5. Agile test pyramid initially designed

It's need to remember that high-level tests should be only the third defense shield, for catching all the remaining issues that were not caught on the first two levels.

4 Conclusions

In the process of the research, we found out the importance of testing in the software development process. We found out what automation is and what role in testing. We analyzed tools for writing automated scripts, provided statistics on the popularity of programming languages to test automation across regions.

Test automation is evolving very quickly, new tools and technologies are emerging almost every week, but they are not yet fully replaceable, and everything cannot be automated. Without human intervention, it is impossible to construct the testing process, since testing requires intellectual intervention. In our opinion in the near future, there will be a trend of increasing number of automated testers and a gradual decrease in manual ones. It will become an obvious educational goal to become an expert in automation. After training in the practical use the testing tools, manual testers will be able to perform automated testing. Therefore, every tester who wants to be more in demand in the IT should learn at least one programming language. So learning programming and testing are processes that depend on each other.

It cannot be said that automated testing will completely replace manual testing. As a result, the conclusion is one: in the world of software testing, there will always be room for both manual and automatic testing. The choice of testing methods is left for every worker in this area.

References

1. Agrafiotis, I., Creese, S., Goldsmith, M.: Developing a Strategy for Automated Privacy Testing Suites. In: Camenisch J., Crispo B., Fischer-Hübner S., Leenes R., Russello G. (eds.) Privacy and Identity Management for Life. Privacy and Identity 2011. IFIP Advances in Information and Communication Technology, vol. 375, pp. 32–44. Springer, Berlin, Heidelberg (2012). doi:10.1007/978-3-642-31668-5_3
2. Alimenkov, N.: Avtomatizatciia testirovaniia v Ukraine nikomu ne nuzhna (Nobody needs testing automation in Ukraine). XP Injection. https://xpinjection.com/articles/test-automation-not-needed-in-ukraine (2013). Accessed 20 Sep 2019
3. Anderson, B.: Best Automation Testing Tools for 2020 (Top 10 reviews). https://medium.com/@briananderson2209/best-automation-testing-tools-for-2018-top-10-reviews-8a4a19f664d2 (2017). Accessed 21 Sep 2019
4. Bushnev, Y.: Top 15 UI Test Automation Best Practices You Should Follow 2019. BlazeMeter. https://www.blazemeter.com/blog/top-15-ui-test-automation-best-practices-you-should-follow (2019). Accessed 27 Sep 2019
5. Casaglia, G., Pisani, F.: The integration and distribution phase in the software life cycle. In: Ehrig H., Floyd C., Nivat M., Thatcher J. (eds.) Formal Methods and Software Development. TAPSOFT 1985. Lecture Notes in Computer Science, vol. 186, pp. 371–384. Springer, Berlin, Heidelberg (1985). doi:10.1007/3-540-15199-0_24
6. Endo, A.T., Bertolino, A., Maldonado, J.C., Delamaro, M.E.: Guest editorial foreword for the special issue on automated software testing: trends and evidence. Journal of Software Engineering Research and Development **6**(2) (2018). doi:10.1186/s40411-018-0047-3
7. Fahad, M., Nadeem, A.: A Survey of UML Based Regression Testing. In: Shi Z., Mercier-Laurent E., Leake D. (eds.) Intelligent Information Processing IV. IIP 2008. IFIP – The International Federation for Information Processing, vol. 288, pp. 200–210. Springer, Boston (2008). doi:10.1007/978-0-387-87685-6_25
8. Ghahrai, A.: Seven Principles of Software Testing. https://www.testingexcellence.com/seven-principles-of-software-testing (2018). Accessed 21 Sep 2019
9. Lawanna, A.: The Theory of Software Testing. AU Journal of Technology **16**(1), 35–40. http://www.journal.au.edu/au_techno/2012/jul2012/journal161_article05.pdf (2012)
10. Mariani, L., Hao, D., Subramanyan, R., Zhu, H.: The central role of test automation in software quality assurance. Software Quality Journal **25**(3), 797–802 (2017). doi:10.1007/s11219-017-9383-5
11. Software Testing Types. Software Testing Fundamentals. STF. http://softwaretestingfundamentals.com/software-testing-types. Accessed 21 Sep 2019
12. Sonmez, J.: Going from QA (or Another Technical Role) to Software Developer. Simple Programmer. https://simpleprogrammer.com/going-from-qa-to-software-developer (2016). Accessed 20 Sep 2019
13. Tasharofi, S., Ramsin, R.: Process Patterns for Agile Methodologies. In: Ralyté J., Brinkkemper S., Henderson-Sellers B. (eds.) Situational Method Engineering: Fundamentals and Experiences. ME 2007. IFIP – The International Federation for Information Processing, vol. 244, pp. 222–237. Springer, Boston (2007). doi:10.1007/978-0-387-73947-2_18
14. What Is The Benefit of Test Automation and Why Should We Do It? SmartBear Software. https://smartbear.com/solutions/automated-testing (2019). Accessed 20 Sep 2019

15. Wyher, T.: 5 Reasons Why Manual Testing Is Still Very Important. DZone DevOps. https://dzone.com/articles/5-reasons-why-manual-testing-is-still-very-importa (2016). Accessed 21 Sep 2019
16. Yehezkel, S.: World's Most Desirable Test Automation Skills!, TestProject. https://blog.testproject.io/2015/12/03/worlds-most-desirable-automation-skills (2015). Accessed 21 Sep 2019

Some aspects of designing of the structural semantics visualization system

Yaroslav Vasylenko[0000-0002-2954-9692], Galina Shmyger[0000-0003-1578-0700]
and Dmytro Verbovetskyi[0000-0002-4716-9968]

Ternopil Volodymyr Hnatiuk National Pedagogical University,
2, Maxyma Kryvonosa Str., Ternopil, 46027, Ukraine
{yava, shmyger, verbovetskyj_dv}@fizmat.tnpu.edu.ua

Abstract. The *purpose* of this article is to investigate the principles and technologies of creating such a semantic interconnection system that would be useful and practical for use in areas such as machine translation, search engines and contextual search. According to the purpose of the research the main *tasks* are defined: 1) to study and analyze the basic principles of construction of semantic dictionary of English language WordNet; 2) to create a lexical-semantic web dictionary of IT-terms of the Ukrainian language. The *novelty* of the work is to adapt all the principles of WordNet to the Ukrainian language. The *practical meaning* of the results obtained is to create a semantic dictionary of the Ukrainian language that will allow to better analyze Ukrainian texts by searching not only the words themselves, but also words that are in one way or another related to the primary, and that will significantly increase the speed of search and analysis of information. In the created web-application (thesaurus) the basic functions of similar existing systems and the latest methods of information linguistics are implemented.

Keywords: structural semantics, lexical-semantic web-dictionary, structural semantics sort systems, thesaurus.

1 Introduction

Providing automation of the efficient work with data presented in the form of natural language texts is one of the actual tasks of computational linguistics. It is caused both by an increase in the e-information stream, and by the need for critical analysis of texts for the subject of authenticity, similarity, probability, etc. A correct understanding of a language is possible provided there is a knowledge of how words and concepts are related to one another, which is meant by one or another utterance, what the purpose has speaker saying a one or another phrase; what is said and what needs to be found in context or perceived based on previously learned information. To solve the problems of analyzing the relationship between words and concepts, to identify all the features of a language, so-called lexical and lexical-semantic databases were developed. Such systems include Princeton WordNet [7; 8; 9; 10], MindNet (Microsoft Research Project software), FrameNet, VerbNet, HowNet, ConceptNet and more. However, for Ukrainian-language content, such developments are at an early stage [5].

The **purpose** of this article is to describe the structural and logical scheme of building a web application for lexico-semantic analysis of user query in the subject area “Informatics”.

The applied meaning of structural semantics closely intersects with the problem of natural language analysis, which is that structural semantics serves as the key to defining the lexical contextual meaning of words, which is the main task of natural language analysis [1]. At this stage of human development, we have not yet fully learned by machine means to interpret natural language and to create a system that at human level is able to perceive natural language and interpret the results, to continue the dialogue.

That is why structural semantics is today a topical direction in the development of both philological and informational disciplines. The structural semantics is at the intersection of two different approaches to exploring the world, it absorbs the best of both, creating a scientific symbiosis that is the building block of the science of the future.

2 Analysis of the basic concepts of the study

2.1 Natural language thesaurus

Thesaurus is a complex component of the dictionary type, in which all the meanings of the dictionary are interconnected by semantic relations, reflecting the basic relations of concepts in the described subject area of knowledge [6]. In the past, thesaurus was mostly referred to by dictionaries, which with the utmost completeness represented the vocabulary of the language with examples of its use in texts.

The thesaurus consists of tokens relating to four parts of the language: adjective, noun, verb and adverb. The descriptions corresponding to each part of the language have a different structure.

The main relationships in thesauruses are:

Synonymy – a link between words in one language, different in sound and spelling, but having the same or very similar lexical meaning, for example, daring – brave.

Antonymy – the relation between the words of one part of the language, different in sound, has the exact opposite meaning: true – false, good – evil.

Hyperonym – a word with a broad meaning that expresses a general, generic concept, the name of a class (set) of objects (properties, features).

Hyponym – a word with a narrower meaning that names an object (property, feature) as an element of a class (set). These relationships are transitive and nonsymmetrical. A hyponym inherits all the properties of hyperonyms. It is a central relation for the description of nouns.

Meronymia / Partonymy – the relation “part – entire”. Within this relationship stand out the relationship of “being an element”.

In addition to these relationships, they also introduce thematic relationships that connect the concepts of one subject area.

An example of a thesaurus:

The hut is a wooden peasant's house.

[Hyperonym]: a residential building
[Meronym]: a rural settlement
[Synonym]: a house

All relationships create a complex hierarchical network of concepts. The properties of relations in the description of different parts of the language are different. In different systems, a thesaurus can perform different functions:

- a source of specialized knowledge in a narrow or wide subject area, a way to describe and ordering the terminology of the subject area;
- search engine in information retrieval systems;
- manual document indexing tool in information retrieval systems (so-called control dictionary);
- automatic text indexing tool.

The main documents governing the thesaurus format are ISO 2788-1986 standards for describing monolingual thesauruses, and ISO 5964-1985 for multilingual ones.

ISO 2788-1986 defines a thesaurus as a set of terms that relate to each other.

The American standard ANSI / NISO Z39.19-1993 extends and refines the ISO 2788-1986 standard for monolingual thesauruses, and imposes a number of additional restrictions on the thesaurus structure.

Thesauruses remain to current date the most accepted form of description of subject domain knowledge, suitable for human perception. Examples of modern foreign thesauruses are WordNet and EuroWordNet.

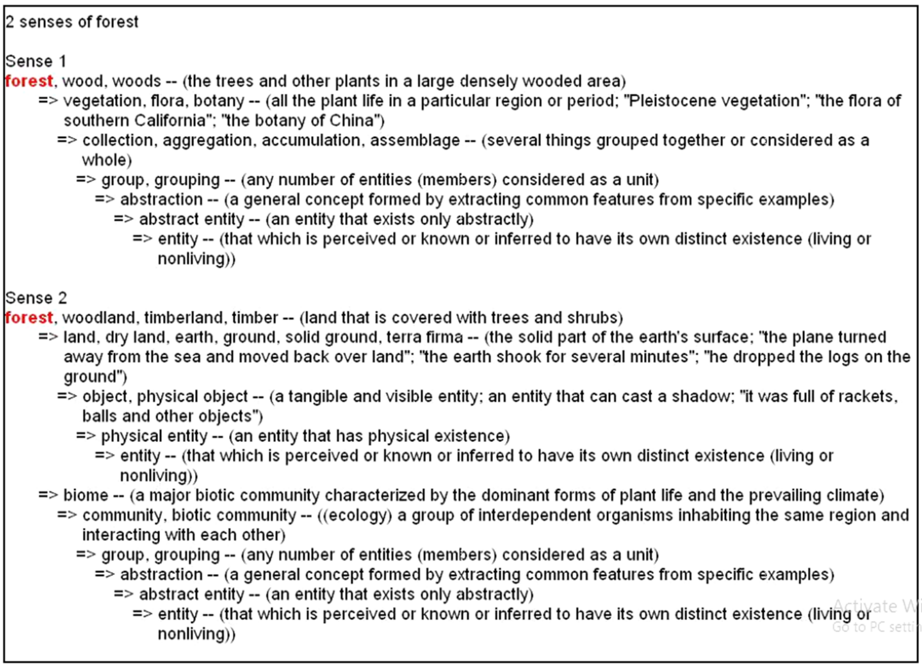
The WordNet English thesaurus emerged in 1990 and began to actively attracted in various areas of automatic word processing. WordNet covers about 100,000 different units (nearly half of which are phrases) organized in 70,000 concepts.

The development of the thesaurus was started in 1984 at Princeton University of the United States under the leadership of the famous psycholinguist George A. Miller [7; 8]. In 1995, WordNet appeared on the Internet freely and caused a surge of research on its use in various computer applications of automatic word processing. The results of using WordNet in automatic word processing turned out to be not unambiguously positive, but WordNet ushered in a new era of developing extra-large structured linguistic resources and caused the emergence of a large number of followers in different countries who create such "natives" for their languages [5]. This thesaurus has also become the basis for multifaceted discussions and research, on the basis of which principles should be built large linguistic resources, suitable for various applications in computational linguistics [4].

The main relation in WordNet is the attitude of synonymy. Synonym sets – synsets – are the basic structural elements of WordNet.

The concept of synonymy used by WordNet developers is based on the criterion that two expressions are synonymous, if replacing one of them with another in the sentence does not change the meaning of the truth of the expression.

The relations between the synsets form a hierarchical structure (Fig. 1). When constructing hierarchical systems on the basis of genitive relations, it is usually assumed that the properties of the parent concepts are inherited by the child – the so-called property of inheritance. Thus, nouns are displayed as a hierarchical system with inheritance. In this case, a systematic effort should be made to find for each synset its generic concept, its hyperonym.



```
2 senses of forest

Sense 1
forest, wood, woods -- (the trees and other plants in a large densely wooded area)
       => vegetation, flora, botany -- (all the plant life in a particular region or period; "Pleistocene vegetation"; "the flora of
          southern California"; "the botany of China")
           => collection, aggregation, accumulation, assemblage -- (several things grouped together or considered as a
              whole)
               => group, grouping -- (any number of entities (members) considered as a unit)
                   => abstraction -- (a general concept formed by extracting common features from specific examples)
                       => abstract entity -- (an entity that exists only abstractly)
                           => entity -- (that which is perceived or known or inferred to have its own distinct existence (living or
                              nonliving))

Sense 2
forest, woodland, timberland, timber -- (land that is covered with trees and shrubs)
       => land, dry land, earth, ground, solid ground, terra firma -- (the solid part of the earth's surface; "the plane turned
          away from the sea and moved back over land"; "the earth shook for several minutes"; "he dropped the logs on the
          ground")
           => object, physical object -- (a tangible and visible entity; an entity that can cast a shadow; "it was full of rackets,
              balls and other objects")
               => physical entity -- (an entity that has physical existence)
                   => entity -- (that which is perceived or known or inferred to have its own distinct existence (living or
                      nonliving))
       => biome -- (a major biotic community characterized by the dominant forms of plant life and the prevailing climate)
           => community, biotic community -- ((ecology) a group of interdependent organisms inhabiting the same region and
              interacting with each other)
               => group, grouping -- (any number of entities (members) considered as a unit)
                   => abstraction -- (a general concept formed by extracting common features from specific examples)
                       => abstract entity -- (an entity that exists only abstractly)
                           => entity -- (that which is perceived or known or inferred to have its own distinct existence (living or
                              nonliving))
```

Fig. 1. Hyperonyms for two values of the forest noun:
forest as a collection of trees and forest as an area where trees grow.

EuroWordNet multilingual thesaurus is currently being developed. Initially, in four languages (Danish, Italian, Spanish, and American English), a network of word meanings is developed that is linked to semantic relationships and allows you to find words that are similar in meaning to different languages. Unlike WordNet, which was designed to describe the lexical and conceptual system of the English language, EuroWordNet is primarily designed to solve the practical tasks of automatically processing large text arrays. The most important tasks that are supposed to be solved with this thesaurus are the following:

- providing multilingual information retrieval;
- increasing the completeness of information search;
- request formulation in natural language;
- semantic indexing of documents, etc.

Domestic scientific institutions have created more than a hundred industry thesauruses that satisfy a certain state standard for dictionaries of this type. They are called – IRT – information retrieval thesauruses.

Standard IRTs are intended primarily for manual indexing of documents, as well as for the formulation and variation of search queries. There are non-standard thesauruses that make the task rather of selective systematizing of terminology in a particular field of knowledge more relevant – this is especially relevant for new subject areas.

2.2 Thesaurus interfaces in information systems

In an information system, a thesaurus is not only an independent information resource, but also a tool for classifying or indexing resources. Thus, the user of the information system should be able to:

- view the thesaurus;
- search for resources by associated terms or concepts (resource search can be accomplished in two ways: keyword search or using a thesaurus);
- do navigation on thesaurus, that is, searching for the desired concept first in the thesaurus, and then querying resources corresponding to this concept.

When searching for keyword resources, the search engine can, by using a thesaurus, extend the search results by giving the user not only the resources that match the keywords entered, but also the resources of related terms or terms, which also denote narrower terms for the original term.

Thesaurus view interface must:

- to show all attributes of a given term or concept;
- to show what terms and concepts are associated with that term or concept;
- to show for the user visually the place of the term or concept in the thesaurus concept hierarchy.

The first 2 points will be fulfilled if show for the user for each thesaurus concept on a separate screen (page) all its attributes, all related terms (in all or in a specific language), and all related concepts. The interface must, at the same time, provide a transition to the viewing page of any of the concepts listed on this page. If the thesaurus data schema allows the term to be bound to more than one concept, then on the same page for each term the terms to which the term is still bound must be listed. If the concept has terms in other languages, fully equivalent to, but attached in the structure of the thesaurus to other concepts, links to pages of those concepts should be provided on the page.

If the thesaurus has a strictly tree-like structure, then the tree is usually presented in the following ways:

- visualization the path of the tree from the root to the current element;
- visualization the path of the tree from the root to the current element, as well as the neighbors of each ancestor of the current element;
- visualization of the whole tree completely. Usually in such cases, the user can open and close the reflection on screen the descendants of any nodes.

To provide an efficient sampling (by one request) of the necessary incisions of hierarchical structures that are fed recursive links between the nodes of these structures, the database tables are expanded with auxiliary columns and integrity conditions.

3 The Presentation of Main Results

3.1 Technological tools for implementing the structural semantics sorting system

Frontend. The Bootstrap framework was chosen as the creation tool of the frontend part of the project, which in the modern world in one way or another serves as the basis for most Internet projects. Bootstrap is the most popular HTML, CSS and JS framework for designing the look and interactivity of web pages.

Designed for anyone and any device, Bootstrap helps you make web pages look faster and easier. It is suitable for people of all levels of experience, for devices of all sizes, and for projects of any size.

Bootstrap comes with pure CSS, but its core code uses the two most popular CSS preprocessors Less and Sass. You can quickly get started with CSS ready, or prefer to building of the styles from core.

Bootstrap was chosen for this project because of the ease, speed and capability of more extensive and easier customization than CMS systems. Also, the choice was made with the expectation that the code of the system, written according to Bootstrap standards can be easily transferred to any other system without significant difficulties, which gives undoubted advantages in the perspective of the project development.

Backend. Since the software part of the project is its core, it was decided not to implement it in the languages of web programming (PHP, JS), but in the full-fledged OOP language C#, using ASP.NET technology [2; 3].

ASP.NET is a technology for creating web applications and web services from Microsoft. It is part of the Microsoft.NET platform.

Since the thesaurus project is essentially a large-scale work on the database, editing it, adding new values and relationships between them, considerable attention was paid to the choice of the database management system (DBMS). The choice was made on MySQL.

Today, MySQL is one of the most well-known, reliable and fastest of the whole existing DBMS family. The principle of operation of MySQL is similar to the principle of operation of any DBMS that uses SQL (Structured Query Language) as a command language to create / delete databases, tables, to replenish tables by data, to perform data sampling.

MySQL, like any other DBMS, is a server program that resides in the computer's memory and maintains a TCP port. The client connects to the DBMS from this port and sends the SQL queries. In turn, the server interprets them by performing the necessary actions and sends the results of the request back to the client. This is how the database server communicates with the client programs.

Because the project is implemented on C# and on ASP.NET technology, choosing a programming environment was not a problem. Because C# is a programming language created by Microsoft and is a product of its own, it was decided to opt for another Microsoft product – Visual Studio (which is perhaps the only full-fledged C# development tool).

3.2 Basic structural elements of the program

Web application Word Topology (WT) consists of such structural elements as database (dictionary, synsets, relationship between synsets), server part (backend), web interface (Frontend).

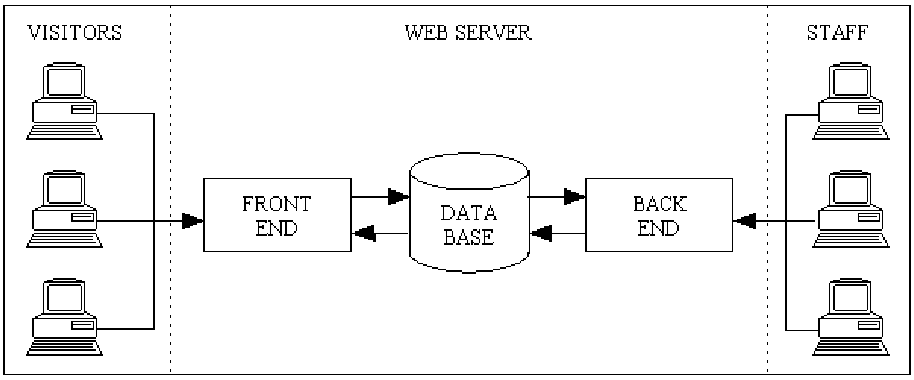


Fig. 2. Scheme of web application work with database, server part and user interface

Database is the place where all the data used by the web application are stored and systemized. As a thesaurus is, in essence, a giant database, as much as possible attention was paid to the DB architecture. Of course, WT works with much smaller amounts of information than, for example, WordNet or other common thesauruses, but the simplicity and ergonomics of the database architecture play an important role even in such projects. Thanks to a well-designed database, it is possible to reduce the server's response time and make the web-application not only a training platform, but also a completely practical system that can be used by users from any corner of the globe.

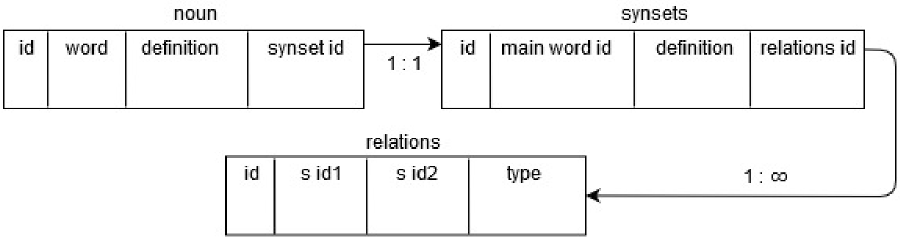
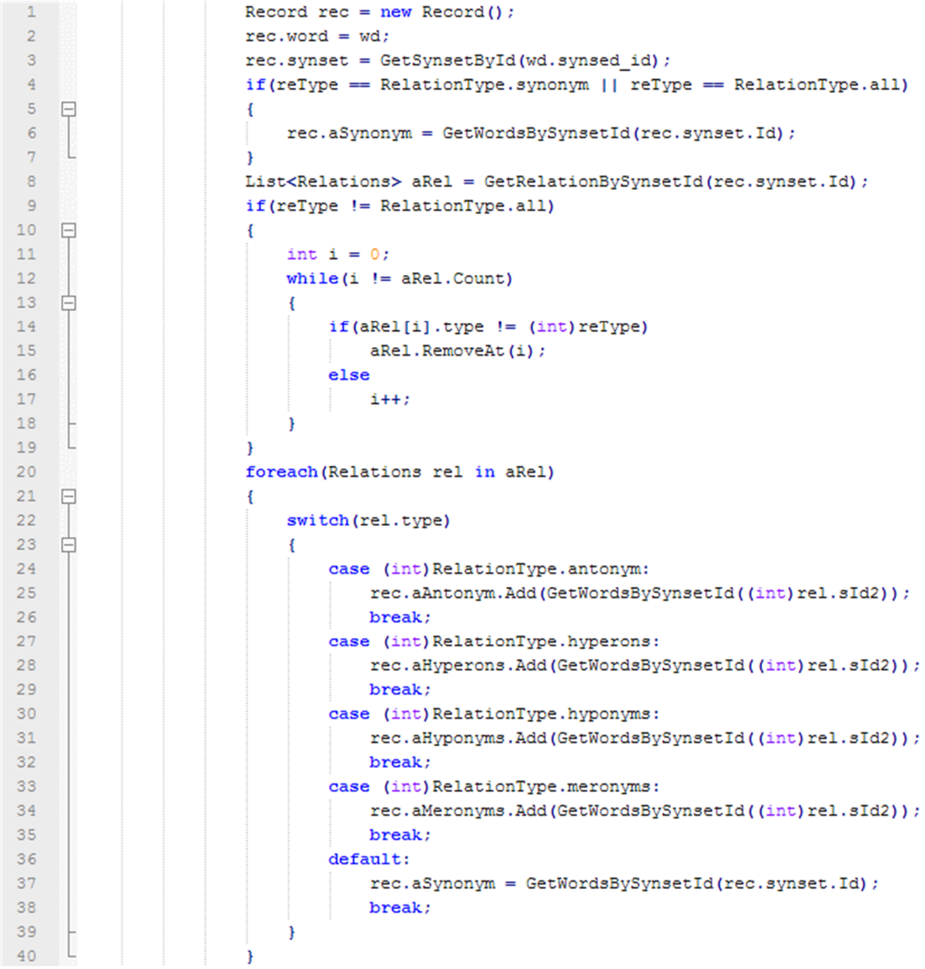


Fig. 3. Organization of the database for the WT web application

Backend is the part of a web application that is responsible for encapsulated user actions and information processing processes. In the WT project, backend is a set of frontend interaction functions, database access, query result formatting, and return of those results back to the request source. Functionality of the backend part is implemented in C# programming language. Here is an example of code from the server side (Fig. 4).

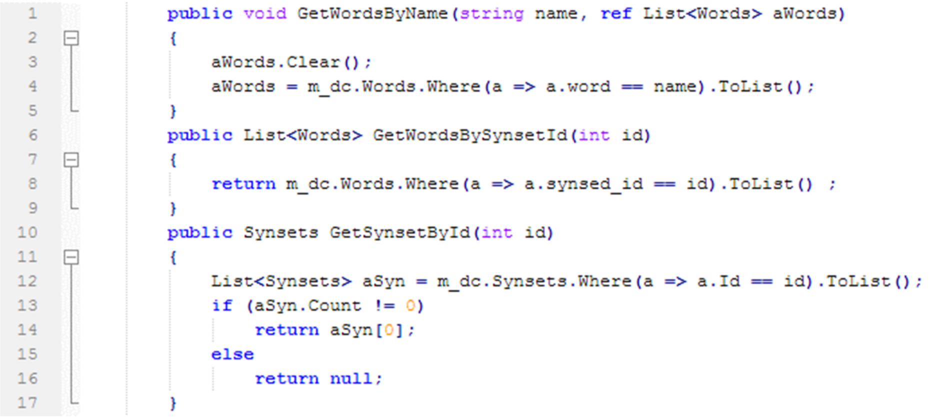


```
Record rec = new Record();
rec.word = wd;
rec.synset = GetSynsetById(wd.synsed_id);
if(reType == RelationType.synonym || reType == RelationType.all)
{
    rec.aSynonym = GetWordsBySynsetId(rec.synset.Id);
}
List<Relations> aRel = GetRelationBySynsetId(rec.synset.Id);
if(reType != RelationType.all)
{
    int i = 0;
    while(i != aRel.Count)
    {
        if(aRel[i].type != (int)reType)
            aRel.RemoveAt(i);
        else
            i++;
    }
}
foreach(Relations rel in aRel)
{
    switch(rel.type)
    {
        case (int)RelationType.antonym:
            rec.aAntonym.Add(GetWordsBySynsetId((int)rel.sId2));
            break;
        case (int)RelationType.hyperons:
            rec.aHyperons.Add(GetWordsBySynsetId((int)rel.sId2));
            break;
        case (int)RelationType.hyponyms:
            rec.aHyponyms.Add(GetWordsBySynsetId((int)rel.sId2));
            break;
        case (int)RelationType.meronyms:
            rec.aMeronyms.Add(GetWordsBySynsetId((int)rel.sId2));
            break;
        default:
            rec.aSynonym = GetWordsBySynsetId(rec.synset.Id);
            break;
    }
}
```

Fig. 4. Creation and populating an instance of the Record class – an intermediate link between the backend and the frontend

Creating an instance of the Record class is the output product of the backend system. The instance attributes store all the information about the result of the database query. From this fragment it is easy to see that such information contains: the word sought, its definition, synsets in which the word resides, the relation of the synsets data to others (hyponymy, hyperonymia, meronymia, antonymy, etc.).

Also noteworthy is the implementation of search methods for records in the database by the entered name and the search for words on the basis of the synset ID (Fig. 5). Implementing two approaches for word search is a necessary step, since the main task of creating a WT thesaurus was to create such a graph-oriented interaction system (which is a thesaurus + relation) so that the user can move freely between the nodes of the graph without any artificial restrictions and with maximum convenience.



```
public void GetWordsByName(string name, ref List<Words> aWords)
{
    aWords.Clear();
    aWords = m_dc.Words.Where(a => a.word == name).ToList();
}
public List<Words> GetWordsBySynsetId(int id)
{
    return m_dc.Words.Where(a => a.synsed_id == id).ToList() ;
}
public Synsets GetSynsetById(int id)
{
    List<Synsets> aSyn = m_dc.Synsets.Where(a => a.Id == id).ToList();
    if (aSyn.Count != 0)
        return aSyn[0];
    else
        return null;
}
```

Fig. 5. The code snippet that is responsible for 2 different methods of finding the desired word in the database

The frontend is written in HTML hypertext markup language and CSS cascading style sheets. The whole system is designed with bootstrap framework and ASP.NET technology, which allows to connect functionally frontend and backend.

The whole interface of the program is implemented (for ease of use) by a structure called accordion. The essence of structure is the submission of information in the form of collapsing lists. The implementation of this element in the software part of the web application is shown in Fig 6.

3.3 Functionality of the program

The main purpose of the program is to create a natural language thesaurus that, taking into account the mistakes of previous similar developments, could serve as a more efficient and accessible capacitive system of human vocabulary, which can be easily used in such areas as automatic translation of texts, systems of parser scanning of documents, systems contextual autocomplete / contextual search in search engines.

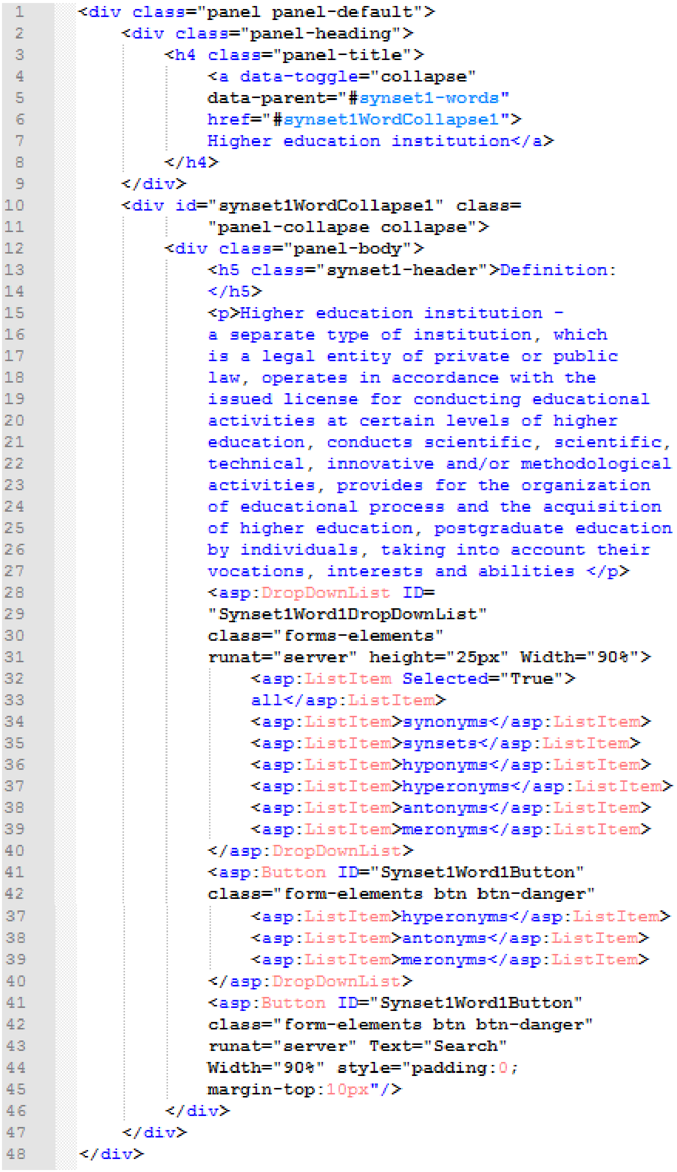
The main purpose of WT has defined the entire functionality of the application.

At this stage of development, the project does not contain all the planned functions, their development requires a deeper analysis of the context of the topic and increased knowledge in parallel with the increase of the project development team.

At this stage the following functions are implemented:

- search word from database;

- search for a word from a database relative to a word synset;
- a convenient way to move between words within the synset and in close (on graph) territories;
- search for all relationships of the synset on all levels of the hierarchy;
- output full information regarding the synset.



```
<div class="panel panel-default">
    <div class="panel-heading">
        <h4 class="panel-title">
            <a data-toggle="collapse"
            data-parent="#synset1-words"
            href="#synset1WordCollapse1">
            Higher education institution</a>
        </h4>
    </div>
    <div id="synset1WordCollapse1" class=
            "panel-collapse collapse">
        <div class="panel-body">
            <h5 class="synset1-header">Definition:
            </h5>
            <p>Higher education institution -
            a separate type of institution, which
            is a legal entity of private or public
            law, operates in accordance with the
            issued license for conducting educational
            activities at certain levels of higher
            education, conducts scientific, scientific,
            technical, innovative and/or methodological
            activities, provides for the organization
            of educational process and the acquisition
            of higher education, postgraduate education
            by individuals, taking into account their
            vocations, interests and abilities </p>
            <asp:DropDownList ID=
            "Synset1Word1DropDownList"
            class="forms-elements"
            runat="server" height="25px" Width="90%">
                <asp:ListItem Selected="True">
                all</asp:ListItem>
                <asp:ListItem>synonyms</asp:ListItem>
                <asp:ListItem>synsets</asp:ListItem>
                <asp:ListItem>hyponyms</asp:ListItem>
                <asp:ListItem>hyperonyms</asp:ListItem>
                <asp:ListItem>antonyms</asp:ListItem>
                <asp:ListItem>meronyms</asp:ListItem>
            </asp:DropDownList>
            <asp:Button ID="Synset1Word1Button"
            class="form-elements btn btn-danger"
                <asp:ListItem>hyperonyms</asp:ListItem>
                <asp:ListItem>antonyms</asp:ListItem>
                <asp:ListItem>meronyms</asp:ListItem>
            </asp:DropDownList>
            <asp:Button ID="Synset1Word1Button"
            class="form-elements btn btn-danger"
            runat="server" Text="Search"
            Width="90%" style="padding:0;
            margin-top:10px"/>
        </div>
    </div>
</div>
```

Fig. 6. The code snippet responsible for displaying the word in synset in accordion form

3.4 Program interface

Much effort has been put into the interface of the program, as in the modern Internet space, users pay attention not only to the functionality and usefulness of the resources used, but also to their appearance. The UI / UX rules of clarity, simplicity, convenience and aesthetic appearance were taken into account when designing the web application interface.

Structurally, the web application interface is divided into two parts - a greeting page and a thesaurus page, which in turn consists of a navigation menu of search, a output field for information about synsets and an output area for information about interconnections between synsets.

3.5 Development of a business layer of structural semantics sorting system

The business layer of this system is implemented using the classes and interfaces listed in Table 1.

Table 1. Classes and interfaces implemented in the system

Name	Attributes	Fields	Methods
class Default	–	o _ws o form o relationDropDown o synsetHolder o txtWord	o Buton1Click() o DisplaySynset() o Page_Load() o RenderControlToHtml()
class Main_dbEntities	o Relations o RelationType o Words	–	o Main_dbEntities() o OnModelCreation()
class WorkSpace	-	o m_ds	o WorkSpace() o AddWord() o DeleteWord() o GetAllWords() o GetSynset() o Init() o SearchWords() o UpdateWord()
class Synset	o ID o Words	o _id o _words	o Synset()
interface Word	o description o Id o name o synset_id	–	–
interface RelationType	o description o Id o name	–	–
interface Relation	o Id o relationType_id o word1_id o word2_id	–	–

Figure 7 presents the classes and interfaces of the developed system.

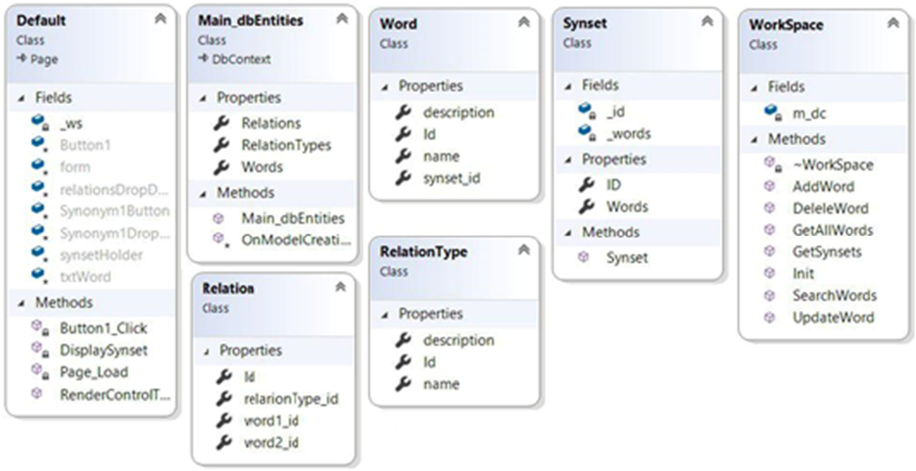


Fig. 7. System classes and interfaces

The search is performed by the word the meaning of which you want to output and by the type of connection that combines the words.

There are 6 types of connections, namely:

- USE
- Used For
- Broader Term
- Broader Term Generic
- Broader Term Partitive
- Related Term

To implement them, a RelationType class was created in the application code.

The program also encounters an All link, which means only that you need to search for words across all links.

The server processes the request and returns a list of elements of the WordSearchResult class. The objects of this class will be created for each meaning of the word searched and will include:

- the word itself
- a synset that includes the specific meaning of the word
- a words list of all types of connections that include the search word, along with a words list for each such connection.

The main logic for working with the database is in the WorkSpace class:

```
public class WorkSpace
{
    public void Init();
    public List<Word> GetAllWords()
    public List<Words> GetWordsBySynsetId(int id);
    public Synsets GetSynsetById(int id);
    public List<Synset> GetSynsets(string sWord)
    public WordSearchRezult SearchWords(string sWord, int? nRelationTypeID = 0)
    public List<string> ParseRelations(string relations);
}
```

4 Conclusions

In this study, most representatives of large thesauruses were analyzed, their source code and algorithms were investigated. Based on the collected data, we created a system of structural-semantic interrelations of words of the Ukrainian language. During the development of the WT web application we took into account the negative aspects of most similar systems and created a combination of the most successful solutions in this field.

During the completion of practical part of the task, an optimal database architecture of dictionaries and other structural units was created, most of the most common thesaurus functions were written, and a user-friendly and intuitive UI was designed that allows to use of thesaurus functionality by ordinary users, not just specialists.

The practical meaning of the developed vocabulary is to improve the search quality in Ukrainian texts. This is directly related to the fact that the search will be conducted not only by a specific word, but also by synonyms, or words that are in one way or another related to the original one.

Also, the initial function of thesauruses of this type cannot be underestimated - finding information in thematic dictionaries is many times more effective than simply browsing the Internet, due to the output of an extremely large number of thematically related information.

The scientific meaning of the dictionary, like most thesaurus dictionaries, provides for the possibility of comparing various aspects of natural languages with one another.

In the the study, the goals and tasks were fulfilled, namely:

- the main principles of WordNet construction and the main types of connections between the synsets were analyzed;
- methods of synsets construction were implemented and optimal database was developed on their basis;
- C# language features were used, namely LINQ (Language Integrated Query) queries, to work with the database efficiently;
- WordTopology web application was developed.

References

1. Anisimov, A., Marchenko, O., Nykonenko A.: Alhorytmichna model asotsiatyvno-semantychnoho kontekstnoho analizu pryrodnomovnykh tekstiv (Algorithmic model of associative-semantic context analysis of natural language texts). Problemy Prohramuvannia, 2-3, 379–384 (2008)
2. Chadwick, J., Snyder, T., Panda, H.: Programming ASP.NET MVC 4: Developing Real-World Web Applications With ASP.NET MVC. O'Reilly Media, Sebastopol (2012)
3. Freeman, A.: Pro ASP.NET MVC 5. Apress, New York (2013). doi:10.1007/978-1-4302-6530-6
4. Kedrova, H., Potemkin, S.: Struktura tezaurusa WordNet i semanticheskaya metrika na lingvisticheskoy baze dannyih (WordNet Thesaurus Structure and Linguistic Metric on Semantic Database). Interface.ru. http://www.interface.ru/home.asp?artId=36137 (2008). Accessed 6 September 2019
5. Kulchytskyi, I., Romaniuk, A., Khariv, Kh.: Rozroblennia WORDNET-podibnoho slovnyka ukrainskoi movy (Development of a WORDNET-like dictionary of the Ukrainian language). Visnyk Natsionalnoho universytetu "Lvivska politekhnika" 673, 306–318. http://ena.lp.edu.ua:8080/bitstream/ntb/6774/1/34.pdf (2010). Accessed 6 September 2019
6. Lukashevych, N.: Tezaurusyi v zadachah informatsionnogo poiska (Thesauri in information retrieval problems). MSU, Moscow. https://istina.msu.ru/publications/book/1283494/ (2010). Accessed 6 September 2019
7. Miller, G.A., Beckwith, R., Fellbaum, C., Gross, G., Miller, K.: Introduction to WordNet: An On-line Lexical Database. http://wordnetcode.princeton.edu/5papers.pdf (1993). Accessed 6 September 2019
8. Miller, G.A., Hristea, F.: WordNet Nouns: Classes and Instances. Computational linguistics **32**(1), 1–3 (2006). doi;10.1162/coli.2006.32.1.1
9. Miller, G.A.: Nouns in WordNet. In: Fellbaum, C. (ed.) WordNet – An Electronic Lexical Database, pp.23-47. The MIT Press, Cambridge (1998)
10. WordNet | A Lexical Database for English official site. The Trustees of Princeton University. https://wordnet.princeton.edu (2019). Accessed 6 September 2019

Author Index