

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КРИВОРІЗЬКИЙ ДЕРЖАВНИЙ ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ
Фізико-математичний факультет
Кафедра інформатики та прикладної математики

«Допущено до захисту»

В.о. завідувача кафедри

_____ Семеріков С.О.

«_____» _____ 2022 р.

Реєстраційний № _____

«_____» _____ 2022 р.

СТВОРЕННЯ ІНТЕРАКТИВНОЇ НОВЕЛИ В СЕРЕДОВИЩІ INSTEAD

Кваліфікаційна робота студента
групи Ім-17
ступінь вищої освіти «магістр»
спеціальності 014 Середня освіта (Інформатика)
Волошина Дениса Вікторовича

Керівник: доц., к. ф.-м.н.
Мерзликін Павло Володимирович

Оцінка:
Національна шкала _____
Шкала ECTS ___ Кількість балів ____

Голова ЕК _____
Члени ЕК _____

ЗАПЕВНЕННЯ

Я, Волошин Денис Вікторович, розумію і підтримую політику Криворізького державного педагогічного університету з академічної доброчесності. Запевняю, що ця кваліфікаційна робота виконана самостійно, не містить академічного плагіату, фабрикацій, фальсифікації. Я не надавав і не отримував недозволену допомогу під час підготовки цієї роботи. Використання ідей, результатів і текстів інших авторів мають покликання на відповідне джерело.

Із чинним Положенням про запобігання та виявлення академічного плагіату в роботах здобувачів вищої освіти Криворізького державного педагогічного університету ознайомлений. Чітко усвідомлюю, що в разі виявлення у кваліфікаційній роботі порушення академічної доброчесності робота не допускається до захисту або оцінюється незадовільно.

(підпис)

ЗМІСТ

ВСТУП	2
РОЗДІЛ 1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ЖАНРУ ІНТЕРАКТИВНОЇ НОВЕЛИ ТА ЇЇ МІСЦЕ В ІГРОВІЙ ІНДУСТРІЇ	3
1.1 Історія та особливості жанру інтерактивної новели	3
1.2. Порівняння наявних на ринку ігор у жанрі інтерактивної новели	8
1.3. Постановка задачі	12
Висновки до розділу 1	17
РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕРАКТИВНОЇ НОВЕЛИ	18
2.1. Обґрунтування вибору ігрового рушія	18
2.2. Проєктування ігрового сюжету	27
2.3. Особливості програмної реалізації	29
2.4. Тестування	32
Висновки до розділу 2	34
Висновки	36
Джерела	37

ВСТУП

Актуальність теми підтверджується наявністю активності користувачів які грають у візуальні новели, а також розробкою нових ігор у цьому жанрі.

Об'єкт дослідження: сюжетні текстові комп'ютерні ігри.

Предмет дослідження: інтерактивні новели.

Мета: створити інтерактивну новелу в середовищі INSTEAD.

Для досягнення мети слід розв'язати такі **задачі:**

- дослідити історію виникнення жанру інтерактивної новели та його характерні риси;
- порівняти наявні ігри в жанрі інтерактивної новели;
- висунути функціональні вимоги до власної гри;
- обрати найбільш доречний ігровий рушій для реалізації гри;
- спроектувати сюжет гри;
- здійснити програмну реалізацію гри в обраному середовищі;
- здійснити тестування та, за необхідності, усунути виявлені недоліки.

Новизна роботи полягає в тому, що створено першу україномовну гру в жанрі інтерактивної новели для середовища INSTEAD.

Практичне значення роботи полягає в тому, що її результатом є завершена комп'ютерна гра в жанрі інтерактивної новели, в яку можуть грати користувачі Microsoft Windows, GNU/Linux, MacOS, BSD, Android та ін.

РОЗДІЛ 1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ЖАНРУ ІНТЕРАКТИВНОЇ НОВЕЛИ ТА ЇЇ МІСЦЕ В ІГРОВІЙ ІНДУСТРІЇ

1.1 Історія та особливості жанру інтерактивної новели

Візуальна новела — жанр відеоігор, у якому історія подається гравцю через статичні зображення, текстові блоки і звуки. Часто візуальні новели помилково порівнюють із симуляторами побачень. У візуальних новелах акцент робиться на сюжеті, який переважно фіксований і не обов'язково містить романтичні елементи, а в симуляторі побачень — на взаєминах із нечисленними персонажами при достатній свободі дій [1].

Персонажі ігор зазвичай виконані в стилі аніме, який, як і візуальні новели, виник в Японії. Сімдесят відсотків ігор для персональних комп'ютерів, які випускаються в Японії, відносяться до цього жанру [2].

Візуальні новели — це такі інтерактивні ігри, з купою тексту і, зазвичай, статичними спрайтами чи фоном. Впливати на події, що відбуваються, можна виконуючи певні вибори, слідуючи за сюжетом. Зазвичай таких виборів не дуже багато, але їх прийняття може призвести до важливих наслідків.

Виходячи з опису, можна стверджувати, що жанр візуальних новел виник як розвиток Interactive fiction (“інтерактивна художня література”). Поняття дуже широке, але в 70-80-ті роки воно набуло більш зрозумілого значення - **текстові квести**.

В іграх такого жанру гравцю потрібно було тільки читати текст, в якому описувалися події, а потім писати якусь команду, щоб просунути сюжет далі. Наприклад: `watch`, `read`. Таким чином, гравці отримували літературний твір, у якому могли брати безпосередню участь.

Першою відеоігрою такого жанру вважається Advent, пізніше названа Colossal Cave Adventure, яка з'явилася в 1975 році. У грі головний герой, читач, досліджував таємничу печеру в пошуках скарбів, і завдяки введеними

командами переміщувався печерою. За нагоди гравець мав занотовувати мапу, інакше неможливо було запам'ятати всі стіни печер (рис 1.1).



Рис. 1.1 Colossal Cave Adventure

Причина, через яку цей жанр став дуже популярним, досить проста. У той час персональні комп'ютери ще були не в змозі виводити на екран кольорові зображення та й взагалі показували тільки текст. Більше того, монітори були занадто дорогими для більшості користувачів. Успіх Colossal Cave Adventure сприяв розвитку жанру Interactive fiction, і подібні ігри почали добре продаватися.

Також це набуло популярності в США, і ігри почали перекладати англійською мовою. У нас цей жанр не набув такої сильної популярності через різницю в появі технічних можливостей: у нашій країні набагато пізніше з'явилися комп'ютери, вже обладнані моніторами, які здатні були виводити сприйнятливий зображення.

Коли є багато тексту, наголос робиться на сюжеті й можливості вибору, то це теоретично схоже на візуальну новелу. Залишилося додати графіку, бажано кольорову. Звичайно, це не єдині складові візуальної новели, і мало сказати, що це книжки з картинками. Прогрес було не зупинити, персональні комп'ютери ставали все популярнішими, дешевшими, і при цьому більш потужними.

Це і приводить до наступної доби, 80-х. Ігри цього періоду цінують насамперед за сюжет. Одна з цих ігор — **Portopia Renzoku Satsujin Jiken**, у перекладі "Справа про серійного вбивцю з Портопії" (рис. 1.2).



Рис. 1.2 Обкладинка гри Portopia Renzoku Satsujin Jiken

У цій грі, доведеться грати роль слідчого і разом зі своїм помічником розкривати вбивство. З погляду геймплею є купа видів дій, наприклад:

- запитати;
- взяти;
- покликати;
- показати;
- ударити.

Причому за сюжетом кілька разів доведеться бити допитуваних, щоб дізнатися правду.

Ранні візуальні новели відрізняються низьким рівнем інтерактивності, в основному вимагаючи від гравця лише натискання певної кнопки для переходу до наступного уривка тексту або сцени.

Велика частина візуальних новел має розгалужений сюжет і кілька варіантів закінчення, у такому випадку гравцеві в певні моменти пропонується зробити вибір між різними варіантами дій чи відповідей у діалозі, таким чином визначаючи подальші події у грі.

Структура оповідання працює як в інтерактивних книгах, популярних у 1990-ті роки. Однак у візуальних новелах фактична сторона оповідання зазвичай складніша, а розкриття характерів персонажів набагато глибше. Візуальні новели схожі на текстові квест-ігри, але мають різне коріння та переслідують інші цілі. З огляду на слабкий сюжет у більшості сучасних ігор, шанувальники вважають спосіб оповідання візуальних новел сильною стороною цього жанру.

Деякі візуальні новели не обмежуються одним жанром і містять елементи інших. Наприклад, в *Symphonic Rain*, де гравцеві потрібно зіграти на чомусь віддалено схожому на музичний інструмент, для переходу до наступної сцени треба набрати достатньо очок. Часто такі міні-ігри пов'язані зі сюжетними предметами.

Деякі прості ігри не містять моментів вибору, як-от фанатські твори, які досить популярні. Для них існує декілька безкоштовних рушіїв та інструментів для спрощення розробки. Найвідоміші — *NScripter*, *KiriKiri* і *Ren'Py*.

У багатьох візуальних романах персонажів озвучують професійні сейю. Головний герой часто залишається без озвучення, на відміну від інших. Музика зазвичай пишеться спеціально для відеогри, однак можливі й винятки, наприклад, *Crescendo*, у якій використовуються популярні пісні та класична музика.

Найперше, на що звертають увагу, — це візуальне оформлення новел. Глобально виділяють два варіанти: *ADV* і *NVL*. Не рідкість, коли обидва поєднуються, доповнюють один одного в тому самому тайтлі [3].

Як відомо, візуальні новели складаються із трьох частин: текст, звук, графіка. Графіка включає:

- спрайти персонажів;
- двовимірні картинки;
- рухливі картинки;
- бекграунди;

- статичні картинки.

Графічна частина моделює ключові події.

В типі ADV текстове поле займає нижню обмежену частину екрану. Це найбільш популярний метод подачі сюжету та діалогів, знайомий будь-якому шанувальнику RPG. У цьому підвиді текст виводиться через невелике вікно, яке зазвичай знаходиться в нижній частині екрана [4]. Простір, що залишився, відведено під барвисті ілюстрації, фони і спрайти персонажів, зазвичай з широким використанням ефектів:

- зміна палітри;
- наближення/віддалення;
- обертання;
- хвилі;
- ефект тряски екрану.

Для жанру характерні короткі речення, оскільки текстове вікно просто не здатне вмістити великий обсяг тексту. Для імен, значків чи портретів, що ототожнюють персонажів, від чиєї особи вимовляється пропозиція, зазвичай передбачена окрема область текстового вікна, що полегшує глядачеві відстеження діалогу. Коли від глядача вимагається вибір, йому надається яскраве, іноді анімоване, озвучене вікно з варіантами дій. Переважна більшість візуальних романів належать саме до цього підвиду. Приклад - "Grisaya no Kajitsu" (Рис. 1.3).



Рис. 1.3 Ігровий процес Grisaia no Kajitsu

У NVL текст займає весь або майже весь екран поверх спрайтів та бекграундів, тим самим нагадуючи фундаментальний книжковий вигляд.

Найяскравіший представник — “Umineko no Naku Koro Ni” (Рис. 1.4). У цьому підвиді, на відміну від ADV, текст виводиться поверх спрайтів та фонів, на весь екран, як правило, на темному напівпрозорому фоні. У деяких випадках для імен відводиться ліва область екрану, або вони згадуються в самому тексті, в інших випадках на початку репліки персонаж відзначається шляхом зміни емоції на своєму спрайті. Великий розмір текстової області дозволяє вміщати більш довгі репліки, описи, ніж ADV.



Рис. 1.4 Ігровий процес NVL новели Umineko no Naku Koro Ni

1.2. Порівняння наявних на ринку ігор у жанрі інтерактивної новели

Конкуренція на ігровому ринку серед візуальних новел також присутня. Сьогодні інтерактивні новели займають значну долю ринку. Зараз на ринку присутні ігри переважно в стилі аніме та за мотивами аніме-серіалів.

Persona 5 Royal є відгалуженням серії Shin Megami Tensei, що розробляється японською студією Atlus. Як і представники основної серії, Persona є японською рольовою грою.



Рис. 1.5 Обкладинка гри Persona 5 Royal

Сюжет п'ятої частини крутиться довкола психології та іноді філософії. Історія стартує з моменту, коли головний герой за прізвиськом Джокер, переводиться в одну зі шкіл Токіо. Зовсім недавно він отримав умовну судимість за нібито напад на високопоставлену особу. Насправді ж, він просто намагався захистити незнайому дівчину від п'яної людини, що пристає до неї. Маючи судимість, Джокер більше не може бути звичайним підлітком. Тепер усі дивляться на нього косо, сприймаючи лише як бандита. І зараз, потрапивши в нове для себе місце, він знайомиться з новими людьми, які так само не змогли знайти своє місце в суспільстві. Разом вони об'єднуються і набувають незвичайної сили — проникати у підсвідомість людей і змінювати їх світогляд, змушуючи тих визнати свої злочини. І таким чином вони вирішують боротися із лиходіями, намагаючись змінити світ навколо себе [5].

Геймплей будь-якої особи ділиться на дві складові. Перша полягає у дослідженні підземель та битві з ворогами. Тут гравець дослідить велику частину ігрового всесвіту Мегамі Тенсей. Друга частина полягає у симуляторі життя звичайного японського школяра. Тут вам доведеться ходити до школи, займатися, відповідати на уроках, складати іспити, шукати, чим зайняти себе

після уроків і так далі. Сюжет актуальний, глибокий, цікавий і добре опрацьований. Персонажі запам'ятовуються своїм характером, дуже добре описані.

Digimon Story Cyber Sleuth: Complete Edition це рольова відеогра, розроблена Media.Vision і видана Bandai Namco Entertainment, яка була випущена в Японії 12 березня 2015 року. Гра від третьої особи, де гравці керують людським персонажем із можливістю командувати Дігімонами, цифровими істотами зі своїми унікальними здібностями, які борються проти інших Дігімонів (рис. 1.6).



Рис. 1.6 Діалог з дігімоном Etemon

Гравці можуть вибирати між Палмоном, Тер'ермоном або Хагурумоном як початковим партнером на початку гри, більше дігімонів можна отримати, коли гравці пробираються в нові області. Загалом представлено 249 унікальних дігімонів, у тому числі сім, які були доступні як DLC протягом життя гри, і два, які були ексклюзивними для західного випуску. У назві є режим New Game Plus, у якому гравці зберігають усіх своїх дігімонів, неключові предмети, гроші, пам'ять, звання детектива, відсотки сканування та прогрес Digifarm. Гравці беруть на себе роль Такумі Айба або Амі Айба,

молодої японської студентки, яка живе в Токіо, а їх мати, репортер, працює за кордоном. Отримавши повідомлення від хакера, Айба досліджує кіберпросторову мережу фізичної взаємодії EDEN, де вони зустрічають Нокія Шірамін і Арату Санаду. Хакер дає їм програми «Digimon Capture» і блокує їх в EDEN. У пошуках виходу Айба зустрічає Юуго, лідера хакерської команди «Захон»; Юуго вчить Айбу, як використовувати їх Digimon Capture, і каже їм, що Арата сам є вмілим хакером. Айба зустрічається з Нокіа та Аратою, які відкривають вихід, але потім на трьох нападає таємнича істота, яка хапає Айбу та руйнує процес їхнього виходу.

SWORD ART ONLINE Alicization Lycoris (рис. 1.7). Сюжет Майстри меча онлайн передає сюжет ранобе з 9 по 14 томи, що відповідає третьому сезону аніме.



Рис. 1.7 Майстри меча онлайн

Це велика сюжетна пригода. Воно вимагає занурення та неабиякої допитливості для вивчення деталей світу. Зате згодом ти отримувеш багато задоволення від геймплею і з ще більшим інтересом стежиш за історією. У Alicization Lycoris вийшло саме так: починаючи з другого розділу бойова система та геймплей проекту розкриваються у всій красі. Ця гра приклад ADV новели з цікавими механіками взаємодії користувача з персонажами

гри. Псевдовідкритий світ має в своєму розпорядженні зручну систему фаст-тревела, і дозволяє не тонути в морі грінда, а з задоволенням боротися в тій мірі, в якій ви хочете зануритися в цю пригоду. Для проходження сюжету тут досить просто йти основними квестами без необхідності виконання побічних завдань у дусі ММО-активностей з полюванням на 10 пташок і збиранням 5 стеблів кропу[6].

1.3. Постановка задачі

Для створення новели потрібно виділити пункти, слідуючи яким вдасться створити цікаву інтерактивну новелу.

Вибір жанру. Необхідно визначитись, буде новела в жанрі хорор, симулятор побачень, квест, психологія, повсякденність. Від жанру залежить цільова аудиторія новели:

- вік;
- стать;
- люди, що належать до товариств.

Слід обміркувати, про що розповідає новела. Придумати сеттинг, у якому відбуватимуться дії. Чітко продумати його. Музика, графіка та інші елементи гри підлаштовуються під сеттинг, щоб створити атмосферу.

Коли ідеї сформовані, потрібно перейти до розробки персонажів.

Доцільно почати з новели за мотивами твору, книги, фантастичного роману, аніме, манги.

Персонажі в новелах дуже різноманітні та контрастують між собою. Кожному з них мають бути притаманні унікальні риси. Персонажу необхідно задати риси характеру які проявлятимуть себе під час ігрового процесу. Прописати частину історії яка буде знайомити гравця з ним. Задати модель поведінки під час вибору гравця, обрати з якими предметами може

взаємодіяти персонаж. Чим більше буде в грі різноманітних персонажів, тим цікавіший можна зробити сюжет гри з багатьма поворотами [7].

Далі перейти до розробки сценарію, перш ніж приступати до програмування. Продумати, якою буде кінцівка або кінцівки, з чого все почалося, як героя можна привести від початку до цих кінцівок. Як інші персонажі реагують на вибір протагоніста. Все це дуже важливо для сюжету.

Щоб почати розробку інтерактивної новели потрібно обрати інструмент. Мною проведено розгляд ігрових рушіїв які використовуються для створення візуальних новел.

Rus Quests Maker Dvade 1.0. Конструктор для створення квестових ігор. Дистрибутив займає 4,91 Мб дискового простору, надає дуже багато і зрозумілий на найпростішому рівні розуміння. Програма російськомовна. При першому запуску тяжко було правильно увійти в редактор, автор пише «Піднесіть курсор до нижньої кромки». Виявилось, що до нижньої кромки вікна. Те, що в меню, це меню майбутньої гри. Загалом програма добре і об'ємно виконана. Конструктор передбачає розробку квестів у віконному режимі зі чітко встановленою роздільною здатністю 1009x700. Є можливість самостійно налаштувати:

1. курсори;
2. фони;
3. музику;
4. анімації героїв (спрайтова анімація).

Перехід від сцени до сцени статичний, одна сцена змінює іншу.

Конструктор загалом розрахований на новачків у розробці ігор, дітей, підлітків. Автор програми дає кілька ключових рекомендацій: *«Кроки зі створення квесту. Чим хитріший сюжет, тим простіше заплутатися, та й час заощадите:*

1. Продуманий сюжет, ескізи на папері з усіма розташуваннями проходів, дверей, предметів, персонажів.
2. Створення фонів сцен.
3. Нанесення на всі сцени зон проходів та переходів між сценами.»

INSTEAD — це редактор текстових новел з картинками, які можна подати як квести. Автори називають це «*інтерпретатором простих текстових пригод*». Досить легкий в освоєнні, зрозумілий інтерфейс. Організовується ком'юніті для просування продуктів, створених у INSTEAD на просторах Інтернету. Можна звернути увагу на цей засіб, оскільки він не тільки може сподобатися дітям, але й допомогти тим самим вчителям урізноманітнити уроки комп'ютерними програмами або тестами, які дуже просто створювати дорослій людині. Навичок програмування не потрібно [8].

Action/Indie Game Maker (AGM/IGM) — потужний конструктор для створення комп'ютерних ігор у 2D різних жанрів. Створений в Японії, є англійськомовний інтерфейс та відповідна реалізація на фан-сайтах можна знайти іншомовні патчі. Програмування не потрібне, хоча продукти можна отримати дуже професійні. Слід зазначити, що особливу увагу розробники приділили навчанню, тобто є спеціальні шаблони, у яких прикріплений віртуальний вчитель, який веде від початку розробки до її завершення. Якість графіки має високі професійні характеристики. Можливостей дуже багато — управління камерою, скролінгом карти, спеціальні можливості для різноманітних індикаторів:

- нарахування очок;
- час;
- життя героя.

Окремо варто сказати про експорт готової гри, зробленої в Action/Indie Game Maker. А саме, їй можливо зробити експорт у *.exe-файл для PC, *.swf-файл, якщо бажано відтворювати все засобами Adobe Flash, а також до

стандарту XNA для Xbox 360. (JDK), при цьому варто розуміти, що через технологію відключається або незграбно трансформується ряд ключових можливостей, тому перевагу краще віддавати варіантам EXE або Xbox. Дистрибутив важить 492 Мб, це пробна версія на 30 днів [9].

Unity — професійний конструктор високого класу. Коштує близько \$1200, є можливість завантажити пробну безкоштовну версію, яка діятиме протягом 30 днів, або оформити навчальну підписку, якщо інструмент потрібен для навчання. Unity називають кросплатформним ігровим рушієм, що багато в чому відповідає істині, хоча за простотою використання та реалізації покрокового режиму створення гри це ближче до конструкторів. Разом з тим, необхідно мати навички програмування JavaScript і C#, для яких є багаті бібліотеки та велика кількість документації [10].

Великим плюсом є чудова автоматизація процесів. Є можливість прямого імпорту 3D-моделей, текстур, скелетів. Фактично з будь-якого професійного пакета 3D-моделювання та анімації. Такі можливості ігрового рушія рідко де можна зустріти. Коли починається робота з Unity, автор поринає у цілий світ, можливостей для творчості та реалізації навіть найсміливіших задумів дуже багато. Причому, якщо у звичайній ситуації для одержання подібної якості необхідна дорога команда, то тут через автоматизацію й добре поставлене уявлення процесу розробки дуже багато можна робити самостійно. Програма виявиться дуже зручною не тільки для професіоналів, а й для початківців [11].

При цьому є вбудовані редактори з розрахунком ландшафтів, освітлення, тіней, анімації частинок. Загалом, все, що потрібно геймдеву. Окремо варто сказати про підтримувані платформи: Web, PC, Mac, Wii, iPhone, iPod, Xbox 360. У рамках конструктора створюються високоякісні 3D онлайн/оффлайн-ігри.

Майже всі новели мають графіку. Можна власноруч малювати персонажів для гри, всі необхідні спрайти, фони або за допомогою засобів фотошопу, обробки графіки. Для повноцінної гри необхідно представити всі ігрові елементи, елементи меню, налаштувань, самої гри на її протязі. Усіх персонажів з кількома емоціями на кожного. Також багато фонів. Загалом роботи багато, графічне оформлення гри — це один з фундаментальних кістяків інтерактивної новели.

Якщо розробник новели володіє базовими речами будівлі тонічного, субдомінантного і домінантного акорду, або з чого складається найпростіший біт, він може сам створити аудіо супровід для гри. Також є безліч зручних програм для редагування аудіо таких як *caustic*, *fl studio*.

Потрібно підготувати необхідну кількість треків на різні моменти гри. Вони мають бути різними за характером, щоб ними можна було виділити ключові моменти пригоди. Також звуки можна брати із сайтів із безкоштовними звуками.

У візуальних новелах мета розробника не перевірити гострий розум або рефлексии гравця, а розповісти йому захопливу історію, яка надовго запам'ятається.

У візуальній новелі має бути достатньо тексту. Пов'язаного та цікавого. Якщо гравець грає в ігри та пропускає діалоги, то має бути такий текст, який не захочеться пропустити.

Good-enough арт. Багато людей помилково кажуть, що в таких іграх якість візуальної частини не є важливою, але якщо гра буде потворною" то ніхто нею не зацікавиться, навіть якщо в грі захоплива історія [12].

Звук задає половину атмосфери візуальній новелі.

Вибори в грі. Є кінетичні новели — візуальні новели без виборів, але в класичній новелі і інших іграх вибір присутній. А отже, кілька виборів, що впливають на розвиток історії, мають сенс.

Висновки до розділу 1

Візуальні новели містять романтичні елементи, однак зустрічаються й такі жанри як наукова фантастика, фентезі чи жахи. Стиль оповіді у візуальних новелах дещо відрізняється від друкованих книг. Здебільшого розповідь ведеться від першої особи, і спостерігачем усіх подій є протагоніст. Протагоністу необхідно робити вибори, які впливають на розвиток подій, під час розповідання сюжету. Ці вибори приводять гравця до одного чи декількох фіналів.

Порівнявши наявні інтерактивні новели на ігровому ринку, я зробив висновок, що переважна більшість в стилі аніме. Це пов'язано з тим, що інтерактивні новели пішли від візуальних новел, батьківщиною яких є Японія.

Для розробки гри потрібно продумати сюжет, описати персонажів гри, написати сценарій до сюжету. Необхідно додати графічний і музичний супровід до новели — так вона краще сприймається і забезпечує цікавістю та насолодою гравця. Все це необхідно запрограмувати в обраному середовищі.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ІНТЕРАКТИВНОЇ НОВЕЛИ

2.1. Обґрунтування вибору ігрового рушія

Для роботи обрано ігровий рушій INSTEAD 3. Програма-інтерпретатор для простих пригодницьких ігор. Являє собою платформу для ігор, що є сумішню візуальної новели та текстового квесту. Ігри на цьому рушії пишуться в основному англійською та російською мовою.

Для розробки ігор на цьому рушії потрібно мати будь-який текстовий редактор. Зручно використовувати редактори з розпізнаванням синтаксису мови програмування. Notepad++ доцільно використовувати для розробки коду гри, текстовий редактор сам розставляє правильну табуляцію, щоб користувач економив час і не збився з логіки програми.

Особливості рушія:

- усі охочі можуть створити свій квест, для цього бажано хоча б початкове знання простої скриптової мови програмування Lua. Детальна документація до програми російською мовою та курс відеоуроків допоможуть освоїти процес написання ігор навіть тим, хто далекий від програмування;
- крім текстового, також є графічний та музичний супровід;
- за допомогою модуля відтворює URQ-ігри;
- здатний відтворювати квести з ігор Космічні рейнджери та Космічні рейнджери 2: Домінатори, а також будь-які квести, написані за допомогою редактора TGE для KP та KP2;
- можливості для різного оформлення інтерфейсу, а також зміни тем у грі “на льоту”;
- можливість повноекранного відображення;
- гнучке налаштування параметрів;
- навчальний режим;
- керування клавіатурою чи мишею;

- підтримка реагування на клацання мишею по певній області картинці.

У грі «Втеча з туалету» використовуються тривимірні зображення, що ефектно імітують інтерфейс сучасних комп'ютерних квестів. Це перша створена для INSTEAD гра з такою особливістю.

У 2010 році вийшла у світ гра «Карантин», яка стала найбільш яскравим зразком INSTEAD-проекту з нелінійним сюжетом. У грі присутні випадкові події, безліч варіантів закінчень, «негативні» та «позитивні» персонажі щоразу вибираються випадковим чином.

6 вересня 2010 року відбулася важлива з погляду функціональності та перспективи розвитку двигуна історична подія — вийшла збірка пасьянсів «INSTEAD Solitaire». Таким чином, тепер можна розглядати INSTEAD не тільки як програму для запуску текстових квестів, але і як багатфункціональну ігрову платформу.

14 липня 2011 з'явилася перша версія INSTEAD для Adobe Flash. Так, наприклад, вже можна пограти у «Повернення квантового кота» у своєму браузері.

Під час створення проекту доцільно використовувати INSTEAD в режимі debug, який задається в параметрах ярлика програми. Крім того, у режимі debug автоматично підключається відладчик. Можна активувати його за допомогою клавіш ctrl-d або f7. При налагодженні гри зазвичай потрібно часто зберігати гру та завантажувати стан гри. Зручно використовувати стандартний механізм збереження через меню або за клавішами f2/f3, або скористатися швидким збереженням чи завантаженням, клавіші f8/f9.

У режимі debug можна перезапустити гру клавішами alt-r. У комбінації з f8/f9 це дає можливість швидко переглянути зміни у грі після її виправлення.

Сцена – це одиниця гри, в рамках якої гравець може вивчати всі об'єкти сцени та взаємодіяти з ними. Сценою може бути кімната, де знаходиться герой. Або ділянка лісу, доступна для спостереження. У будь-якій грі має бути сцена з ім'ям main:

```
room {
```

```

nam = 'main';
disp = "Головна кімната";
dsc = [[Ви в великій кімнаті.]];
}

```

Запис означає створення об'єкта. Майже всі сутності в INSTEAD — це об'єкти, main типу room (кімната). Атрибут об'єкта nam зберігає ім'я кімнати 'main', яким можна звертатися до кімнати з коду. Кожен об'єкт має унікальне ім'я. Для звернення до об'єкта на ім'я, використовується наступний запис:

```
dprint("Об'єкт: ", _'main')
```

Атрибут dsc – це опис сцени, який виводиться один раз на вході в сцену або при явному огляді сцени. У ньому немає описів об'єктів, які знаходяться у сцені [13].

Об'єкти – це одиниці сцени, із якими взаємодіє гравець.

```

obj {
    nam = 'стол';
    dsc = 'В комнаті стоить {стол}.';
    act = 'Гм... Просто стол...';
};

```

Ім'я об'єкта nam використовується під час їх потрапляння у інвентар. Хоча, в цьому випадку стіл навряд чи туди потрапить. Якщо в об'єкті визначено disp, то при потрапленні в інвентар для його відображення використовуватиметься саме цей атрибут.

Декорації. Об'єкти, які можуть бути перенесені з однієї сцени до іншої або потрапляти в інвентар, зазвичай мають ім'я або змінну-посилання. Бо таким чином завжди можна знайти об'єкт будь-де і працювати з ним. Але чималу частину ігрового світу становлять об'єкти, які займають конкретну локацію та служать як декорації.

Таких об'єктів може бути дуже багато, зазвичай, це однотипні об'єкти, такі як дерева і тому подібні об'єкти. Для створення декорацій можна використати різні підходи.

Можна створити один об'єкт, 'дерево' і поміщати їх у різні кімнати:

```
obj {
    nam = 'дерево';
    dsc = [[Тут стоить {дерево}.]];
    act = [[Все дерева виглядають схожими.]];
}
room {
    nam = 'Лес';
    obj = { 'дерево' };
}
room {
    nam = 'Улиця';
    obj = { 'дерево' };
}
```

Графіка. Графічний інтерпретатор INSTEAD аналізує атрибут сцени `pic`, та сприймає його як шлях до картинки:

```
room {
    pic = 'gfx/home.png';
    nam = 'Дома';
    dsc = 'Я у себе дома';
};
```

Підтримуються всі найпоширеніші формати зображень, але рекомендується вам використовувати `png` і `jpg`. Також можна використовувати як картинку анімовані `gif` файли. Є можливість вбудовувати графічні зображення прямо в текст, у тому числі в інвентар, переходи, назви кімнат і `dsc` за допомогою функції `fmt.img`:

```
require "fmt"
obj {
    nam = 'яблоко'
    disp = 'Яблоко' .. fmt.img('img/apple.png');
```

```
}
```

INSTEAD підтримує обтікання картинок текстом. Якщо зображення вставляється за допомогою функції `fmt.imgl` або `fmt.imgr`, воно буде розташоване біля лівого або правого краю [14].

Музика. Для роботи з музикою та звуками знадобиться модуль `snd`.

```
require "snd"
```

Інтерпретатор відтворює в циклі поточну музику, яка задається за допомогою функції: `'snd.music(шлях до музичного файлу)'`. Підтримується більшість музичних форматів, але наполегливо рекомендується використовувати формат `'ogg'`, тому що саме він підтримується найкраще у всіх версіях INSTEAD

```
room {
  pic = 'gfx/street.png';
  enter = function()
    snd.music 'mus/rain.ogg'
  end;
  nam = 'на улице';
  dsc = 'На улице идет дождь.';
};
```

`snd.music()` без параметра повертає поточне ім'я треку. У функцію `snd.music()` можна передавати другий параметр — кількість відтворень. Отримати поточний лічильник можна за допомогою `snd.music()` без параметрів – друге значення, що повертається. 0 – означає вічний цикл. 1..n – кількість програвань. -1 – відтворення поточного треку закінчено [16].

Для того, щоб скасувати відтворення музики, можна використовувати `snd.stop_music()`. Для того, щоб дізнатися, чи грає музика:

```
snd.music_playing()
```

Можна задавати час наростання та згасання музики за допомогою виклику:

```
snd.music_fading(o, [i])
```


Тут про час у мс. для згасання та і час у мс. для наростання музики. Якщо встановлено лише один параметр — обидва часи вважаються однаковими. Після виклику, встановлені параметри впливатимуть на відтворення всіх музичних файлів

Для відтворення звуків використовується `snd.play()`. Рекомендується використовувати формат 'ogg', хоча більшість поширених звукових форматів також працюватиме.

Відмінність між музикою та звуковим файлом полягає в тому, що двигун стежить за процесом програвання музики та зберігає/відновлює поточний трек, що програвється. Вийшовши з гри та завантаживши її знову, гравець почує те саме музичне оформлення, що чув при виході. Звуки зазвичай означають короткочасні ефекти, і двигун не зберігає і відновлює звукові події. Так, якщо гравець не встиг дослухати звук пострілу і вийшов із гри, після завантаження файлу збереження він не почує звук (або його закінчення) знову.

Тим не менш, якщо врахувати те, що `snd.play()` дозволяє запускати зациклені звуки, то різниця між музикою та звуками стає вже не такою однозначною. Отже, визначення функції: `snd.play(файл, [канал], [цикл])`, де:

- файл – шлях та\або ім'я звукового файлу;
- канал – номер каналу [0..7], якщо не вказано, буде обрано перший вільний.
- цикл – кількість відтворень 1..n, 0 – зациклювання.

Для зупинки відтворення звуку можна використовувати '`snd.stop()`'. Для зупинки звуку у певному каналі '`snd.stop(канал)`'.

Загалом — це дуже зручний і зрозумілий для програмістів рушій простих текстових пригод. Що важливо, він не тільки зручний, він ще й легко розширюється, багато ігор пишуться на Lua.

Декілька коротких сценаріїв що описують простоту розробки на INSTEAD.

Сценарій 1: предмет+предмет

Гравець має інвентар, в інвентарі є кілька предметів. Стандартна поведінка INSTEAD така: ви клацаєте на один предмет, потім клацаєте на інший — так ви застосували перший предмет на другий. Щоб зловити цю подію, треба визначити обробник `use` у першому предметі або `used` у другому. Але якщо ви застосуєте другий предмет на перший, то потрібно буде знову перехоплювати вже нову подію. А якщо ви хочете просто комбінувати предмети, без урахування порядку?

```
cobj = function(v)
v.use = function(this,that)
return call(this, 'fuse', that);
end;
v.used = v.use;
return obj(v);
end
```

Ось так. Тепер достатньо оголошувати об'єкти типу `cobj` та давати їм функцію `fuse` [17]:

```
rope = cobj{
nam = 'мотузка',
inv = function()
local response = 'Довга міцна прядив'яна мотузка.';
return response;
end,
fuse = function(this, that)
if (that == lock) then
inv():del(lock);
inv():del(rope);
inv():add(rope_with_lock);
return 'Ви прив'язуєте до кінця мотузки замок.'
end;
```

```
end,
}
```

Сценарій 2: видача випадкової фрази в описі кімнати

Якщо зробити опис кімнати функцією, яка просто повертатиме випадкову фразу, то вона виконається лише один раз і далі видаватиме те саме. Щоб опис таки змінювалося, треба виводити його через `return` функції, а через текстовий буфер — функції `print` [18]:

```
something = function()
  ifsen = ' ';
  response = {'веселого жука.', 'веселого динозавра.',
'веселого людодіда.', 'веселого жука-людодіда.'};
  return ifsen..response[rnd(#response)];
end

dinner = room {
  nam = 'Весела їдальня,
  dsc = function()
  p [[Я бачу!]];
  p (something())
  end
};
```

Сценарій 3: новий тип об'єктів

Припустимо, потрібен новий тип об'єктів у кімнатах. Скажімо, ви пишете RPG (можна хоч MMORPG — аби гравець мав бібліотеки Lua для роботи з мережею), і вам потрібен список квестів [19]. Для цього нема чого перезбирати двигун, досить трохи покопатися в його частині Lua:

```
function quest(v) --constructor
  if v.nam == nil then error ("Не вказано ім'я квеста.", 2) end
  if v.short_dsc == nil then v.short_dsc = "" end
  if v.scene == nil then v.scene = room_scene end
```

```

if v.completed == nil then v.completed = false end
if v.look == nil then v.look = room_look end
if v.save == nil then v.save = room_save end
v.location_type = true;
if v.way == nil then v.way = { } end
v.isQuest = true;
v.way = list(v.way);
v = obj(v);
return v;
end

```

```

function room_look(self)
local i,n,v,ph
for i,o in opairs(self.obj) do
if isObject(ref(o)) and not o.isQuest then
o = ref(o);
if v == nil then v = stead.par(' ',v, o:look());
else v = v .. stead.par(' ',v, o:look());
end
end
end
end

```

```

function room(v)
v.location_type = true;
if v.look == nil then v.look = room_look end
if v.scene == nil then v.scene = room_scene end
if v.quests == nil then v.quests = list {} end
if v.obj == nil then v.obj = v.quests
else for k,m in pairs(v.quests) do v.obj[k] = m end
end

```

```

if v.way == nil then v.way = {} end
v = room(v);
return v;
end

```

Спочатку ми визначаємо новий тип об'єктів — квести. Це практично копіює з `stead.lua` об'єкта `obj`, тільки додані поле `isQuest`, щоб легше розрізнити квести та об'єкти, і додано поле `completed`, яке відповідає за виконання квесту.

`room_look` — це функція, що викликається, коли гравець оглядає кімнату. Зазвичай вона виводить опис всіх об'єктів кімнати (з масиву `obj`). Тут додано лише заглушку: не виводити описи всіх квестів у кімнаті.

Також перевизначено саму кімнату — тепер у ній є масив `quests`. Масив `obj` заповнюється об'єктами з масиву `quests` - прирівнюється, якщо він порожній, і з'єднується, якщо об'єкти в ньому є [20].

2.2. Проектування ігрового сюжету

Сюжет гри тісно переплетений з реальністю, але події відбуваються в іншій реальності. За сюжетом, я як головний герой новели, прокинувся вранці в день захисту магістерських робіт.

Початкова локація — моя квартира де починаються події сюжету. З'явиться текст: «Нарешті цей день настав! Сьогодні я стану магістром!! Що ж доведеться вставати з ліжка адже без цього мені не стати магістром» Після певних дій гравця відбувається перехід до наступної локації, де я, як герой книги буду проходити через труднощі що стоять на моєму шляху до диплому.

На другій локації мене чекатиме діалог з Володимиром Миколайовичем. Перед зустріччю пролог: «На підході до університету я зустрів Володимира Миколайовича. Він похмурий, наче поринув у роздуми і одразу мене не помітив, доки я не підійшов щоб привітатися». На початку діалогу гравець першим заговорить до персонажа:

– Володимир Миколайович, доброго дня!

- Привіт, Денисе, запізнюєшся.
- В мене ще є 15 хвилин, тож встигну.
- Ну дивись.
- Володимир Миколайович, ви про щось задумались?
- Ах ти помітив? Скажи мені... ти готовий до захисту магістерської?

Тут розгалуження відповідей від вибору яких буде залежати відповідь персонажа. В подальшому розвитку сюжету від вибору гравця залежатиме настрої цього ігрового персонажу.

Третя локація ототожнення мого робочого кабінету. Там чекатиме колега гравця, з яким може виникнути діалог. Він запитає в героя чи підготувався він до захисту:

- якщо підготувався:
 - він побажає успіху, діалог завершиться, відкриється 5 локація яка буде актом захисту;
- якщо не готовий:
 - відкриється 4 локація яка приведе героя до його наставника.

Четверта локація зустріч з майстром.

- Доброго дня, Денисе.
- Доброго дня, Павло Володимирович.
- Я сподіваюся що ви підготувалися до захисту і ваша програма вас не підведе.
- Сподіваюсь – мовив Денис намагаючись приховати міміку обличчя яка видавала його напруженість в даній ситуації

Учитель запевнить героя що він готовий стати магістром, надасть впевненості.

П'ята локація фінальна битва дискусій. В аудиторії на героя чекали його колеги, а також бос локації.

- Доброго дня всім!
- Привіт!

Тільки один Сергій Олексійович бачив істину в сьогоднішніх подіях
відповів:

– гадаєте сьогодні добрий день? Що ж, побачимо!

Я одразу побліднів.

В цій сцені буде реалізовано певну кількість діалогів з персонажами гри які вже зустрічалися герою та декількома ключовими босами що вводяться в дану сцену. Системою виборів буде змінюватись загальна атмосфера локації та спрайти персонажів. Від кінцевої атмосфери буде залежати фінал гри.

2.3. Особливості програмної реалізації

Новели на INSTEAD пишуть на мові програмування Lua. Це скриптова мова програмування, розроблена в підрозділі Tecgraf (Computer Graphics Technology Group) Католицького університету Ріо-де-Жанейро (Бразилія). Інтерпретатор мови вільно розповсюджується, з відкритими вихідними текстами мовою Сі [15].

Основні файли гри:

- o game.lua – містить код гри з описом ігрових механік;
- o intro.lua – основний файл в якому описано сюжет гри, діалоги, функції та можливості в грі;
- o main.lua – файл для перевірки сумісності і переходу між версіями ігрового рушія;
- o theme.ini. – конфігураційний файл який задає параметри вікна, значків, фону гри.

Папки з ресурсами:

- o gfx – сховище зображень ;
- o saves – містить збереження гри;
- o snd – бібліотека музики яка використовується в грі.;
- o theme – звідси конфігураційний файл бере значки та фони для відображення вікна гри.

Перш за все описую функцію переходу між локаціями:

```
function gnext(v)
    return vway ('дали', '{Далі}.', v );
end
```

Додаю фонову музику до гри:

```
sound_player = obj {
    nam = 'sound player',
    ini = function(s)
        s:life();
    end,
    life = function(s)
        if isHouse(here()) or here()._from_house or
not game.lifes:look(s) then
            set_sound(nil, 1);
        else
            if not is_sound(1) then
                set_sound('snd/theme.ogg', 1, 0);
            end
        end
    end
end
}
```

Починаю оголошення ключових об'єктів:

```
main = room {
    pic = 'gfx/images/',
    nam = 'Ранок',
    dsc = [[
        Нарешті цей день настав! Сьогодні я стану
магістром!!^
        Щож доведеться вставати з ліжка адже без цього
мені не стати магістром
    ]]
};
boss = obj {
    nam = 'Володимир Миколайович',
```



```

dsc = function(s)
    return 'Перед собою я бачу міркуючого
{Володимира Миколайовича}';
end,
act = function(s)
    return walk 'boss_dlg';
end
};
ruban = obj {
    nam = 'Віталій',
    dsc = function(s)
        return 'На мене направив свій погляд
{Віталій}';
    end
};

```

Перша локація з діалогом:

```

il = room {
    nam = 'Біля університету',
    pic = 'gfx/images/soloviev.png',
    dsc = [[
        На підході до університету я зустрів
Володимира Миколайовича.
        Він похмурий, наче поринув у роздуми і одразу
мене не помітив, доки я не підійшов щоб привітатися.
    ]],
    obj = { 'boss' },
    way = { 'toroses' },
};
boss_dlg = dlg {
    pic = 'gfx/images/soloviev.jpg',
    nam = 'Розмова з Володимиром Миколайовичем',
    dsc = 'Він не звартає на мене уваги - тож я
почну...',
    obj = {
        [1] = phr('Володимир Миколайович, доброго дня!',
            '- Привіт, Денисе, запізнюєшся.', [[ ron(2)
        ]]),
        [2] = _phr('В мене ще є 15 хвилин, тож встигну.',
            '- Ну дивись.', [[ ron(3) ]]),
    }
};

```

```

[3] = _phr('Володимир Миколайович, ви про щось
задумались?',
    '- Ах ти помітив? Скажи мені... ти готовий до
захисту магістерської?', [[ pon(4, 5, 6) ]]),
[4] = _phr('Не знаю... Напевно.',
    '- А я гадаю, що ніхто з вашої групи не
готовий до захисту', [[pon(7); poff(5,6);
toroses:enable() ]]),
[5] = _phr('Так! Я закінчив виконання роботи і
готовий її захищати!',
    'Володимир Миколайович скептично подивився на
мене, відвернувся і пішов до університету...',
[[pon(7); poff(4, 6); toroses:enable() ]]),
[6] = _phr('Не хочу про це думати',
    '- Тоді ходімо готуватися - Володимир
Миколайович мовив це і пішов розмовляти по
телефону...', [[pon(7); poff(4,5); toroses:enable()
]]),

[7] = _phr('Закінчити розмову', nil, [[ pon();
back(); return true ]]),
},
exit = function(s)
    s:pon(7);
end
};

```

2.4. Тестування

Тестування програми проводилось в режимі подвійної роботи. Завдяки збірці INSTEAD можна не закривати вікно програми, а одразу змінювати код і перезавантажувати гру в меню рушія (Рис. 2.1, 2.2,2.3).

Марістр

— □ ×

Ранок



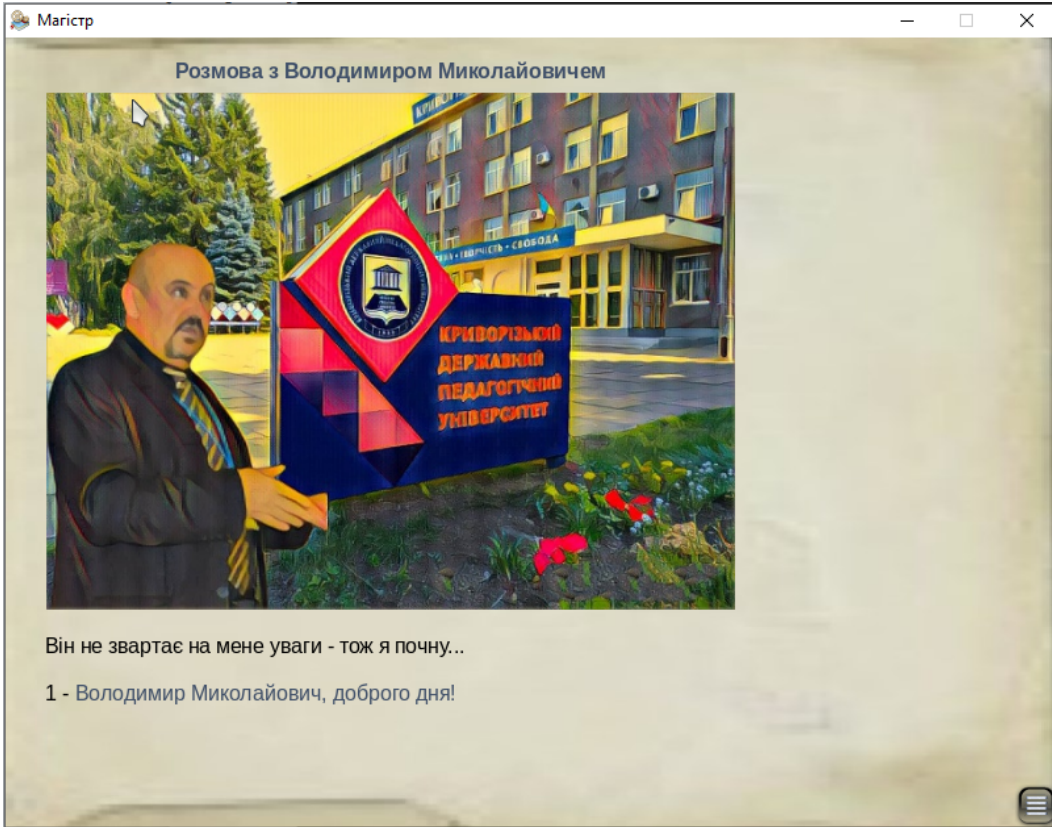
Нарешті цей день настав! Сьогодні я стану магістром!!
Щож доведеться вставати з ліжка адже без цього мені не стати
магістром

Далі.

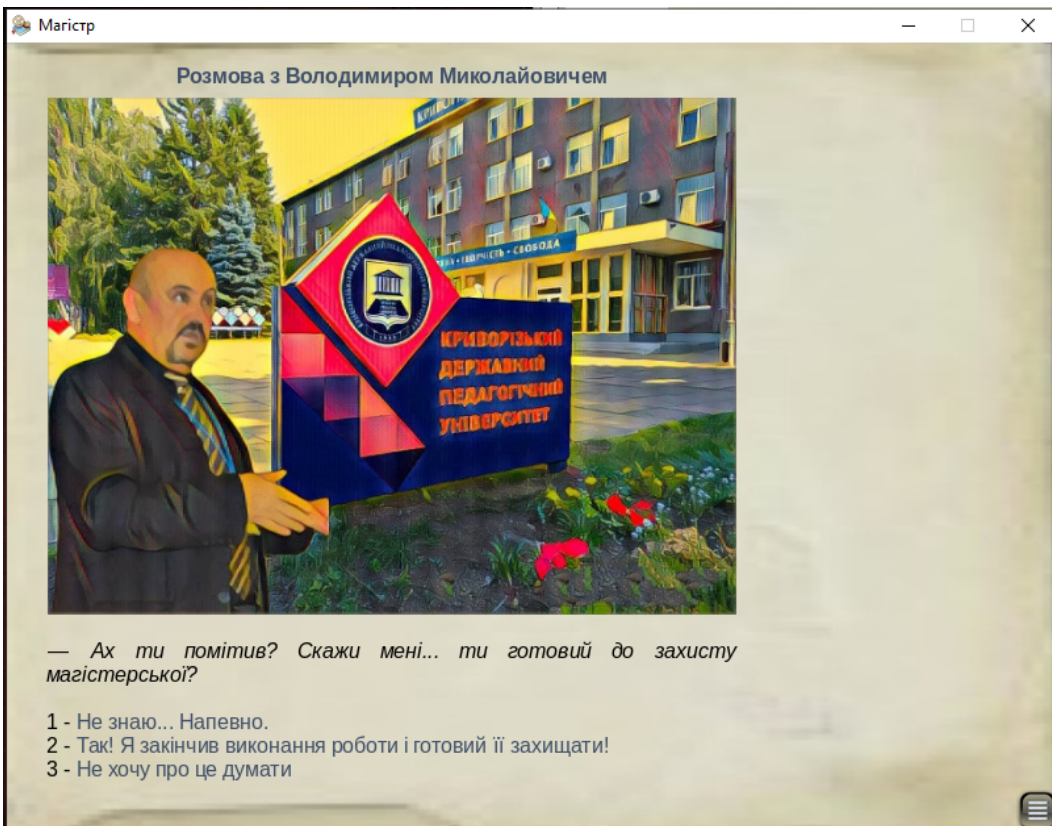
Рис. 2.1 Демонстрація роботи програми



Рис. 2.2 перевірка роботи відображення тексту і функції початку діалогу



а)



б)

Рис. 2.3 робота системи діалогів

Висновки до розділу 2

Я зрозумів, як має виглядати класична інтерактивна новела. Головне вікно гри містить описову частину, інформацію про статичну та динамічну частину сцени, активні події та картинку сцени (у графічному інтерпретаторі) з можливими переходами до інших сцен.

Описова частина сцени відображається лише один раз, при показі сцени, або при явному огляді сцени (у графічному інтерпретаторі – статична частина сцени містить інформацію про статичні об'єкти сцени (звичайно, це декорації) і відображається завжди. Ця частина написана автором гри.

Динамічна частина сцени складена з описів об'єктів сцени, які присутні в даний момент на сцені. Ця частина генерується автоматично і відображається завжди. Як правило, в ній представлені об'єкти, які можуть змінювати своє місцезнаходження.

Ігровий рушій INSTEAD один з багатьох рушіїв для розробки інтерактивних новел, але особливість яка його виділяє це – зручна робота з кодом та його гнучкість в розробці. Його можна підлаштувати під багато задач.

ВИСНОВКИ

У текстографічних ігор величезний потенціал. Вони не такі інтерактивні, вони не відбирають наше життя замість вічно незадоволеного бажання, не змушують цілодобово просиджувати за монітором. Вони можуть взяти найкраще зі світу літератури та комп'ютерних ігор. І те, що жанр здебільшого некомерційний, – навіть плюс.

Порівнявши ринок інтерактивних новел, можна зробити висновок, що переважна більшість у стилі аніме. Це пов'язано з тим, що інтерактивні новели пішли від візуальних новел, батьківщиною яких є Японія.

Виходячи з огляду аналогів, було сформульовано вимоги до майбутньої гри. Зокрема, гра буде україномовною через брак таких ігор на ринку а також кросплатформною для можливості використання на різних пристроях.

Враховуючи сказане, було обрано ігровий рушій INSTEAD, який має весь необхідний функціонал, підтримує більшість сучасних операційних систем, надає широкі можливості для створення ігрової атмосфери.

Для розробки гри було продумано сюжет, описи персонажів гри, написано сценарій до сюжету. Підібрано графічний і музичний супровід до новели — так вона краще сприйматиметься і забезпечить інтерес і насолоду гравця.

Все це було запрограмовано в обраному середовищі мовою програмування Lua.

У результаті тестування суттєвих недоліків не виявлено, створена гра працює належним чином.

Таким чином, усі поставлені задачі виконані в повному обсязі. У майбутньому гра може вдосконалюватися шляхом ускладнення сюжету або може бути взята до уваги при розробці інших ігор у середовищі INSTEAD.

Джерела

1. Гребенюк А. Інтерактивна література як новий феномен української культури / А. Гребенюк // Науковий вісник Східноєвропейського національного університету імені Лесі Українки. Філософські науки. - 2014. - № 18. - С. 55-59. - Режим доступу: http://nbuv.gov.ua/UJRN/Nvvnufn_2014_18_11
2. Как Эроге захватили рынок Японии. – StopGame https://stopgame.ru/blogs/topic/105087/kak_eroge_zahvatili_rynok_yaponii
3. The Visual Novel Database. — Крупнейшая англоязычная база данных по визуальным романам, существующая с 1.10.2007 // <https://vndb.org>
4. Виды визуальных новелл // <https://gamer.livejournal.com/1432596.html>
5. Огляд Persona 5 Royal. Идеал жанра JRPG? <https://www.ixbt.com/live/games/persona-5-royal-ideal-zhanra-jrpg-obzor.html>
6. SWORD ART ONLINE Alicization Lycoris https://store.steampowered.com/app/1009290/SWORD_ART_ONLINE_Alicization_Lycoris/
7. Василий Воронков. Интервью: Петр Косых // IFPrint.org. — Март 5, 2013. — Вып. 1. — ISSN 2307-535X. Архивировано 26 февраля 2015 года.
8. Пётр Косых. INSTEAD: Как все начиналось // IFPrint.org. — Март 18, 2013. — Вып. 1. — ISSN 2307-535X. Архивировано 26 февраля 2015 года.
9. Kerr, Chris. "GameMaker Studio 2 gets new low-cost 'Creators Edition'" <https://www.gamedeveloper.com/business/gamemaker-studio-2-gets-new-low-cost-creators-edition->
10. Хокинг, Джозеф. Unity — в действии. Мультиплатформенная разработка на C# <https://ru.pdfdrive.com/unity-в-действии-Мультиплатформенная-разработка-на-c-e187707045.html>
11. Escape The Toilet Pack: A Triple Flush <https://instead-games.ru/game.php?ID=45>

12. STEAD3 Петр Косых 31 августа 2021 г.
<https://github.com/instead-hub/instead/releases/download/3.4.0/stead3-ru.pdf>
13. Kurt Jung, Aaron Brown. Beginning Lua Programming. — John Wiley & Sons, 2011. — 675 с. — (Programmer to Programmer). —
https://books.google.nl/books?id=Br8l4Z_XUgoC&printsec=frontcover&hl=ru#v=onepage&q&f=false
14. Miller D. Love, Money, Rock'n'Roll Brings A Russian Outlook to Visual Novels/ D. Miller // Cliqist: Indie Gaming – URL: <http://bit.ly/2CwK97r>
15. Sevakis J. Why Haven't Light Novels And Visual Novels Caught On In America? /J. Sevakis. – URL: <http://bit.ly/2qfdMoP>
16. Visual Novel / Everlasting Summer – URL: <http://bit.ly/2lMZA0H>
17. Visual Novels: A History – URL: <https://bit.ly/2vGfUJc>
18. Agnos Jr. T. C. Game Development Of Ibong Adarna Visual Novel / T. C. Agos Jr.,M. L. C. Bal-Ut, L. G. Calam and others // International Journal Of Scientific & Technology Research. 2013. Vol. 2. № 9. – URL: <http://bit.ly/2CyUnEw>
19. Crimmins B. A Brief History of Visual Novels / B. Crimmins – URL: <http://bit.ly/2gIlr85>
20. Vinciguerra, David; Howell, Andrew (16 October 2015).
https://books.google.com.ua/books?id=dM-9CgAAQBAJ&pg=PA3&redir_esc=y#v=onepage&q&f=false