

ЕКОНОМІКО-ТЕХНОЛОГІЧНИЙ ІНСТИТУТ ІМЕНІ РОБЕРТА ЕЛЬВОРТІ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

«Допущено до захисту»
Завідувач кафедри
інформаційних технологій
О.П. Бондар
«19» 06 2024 р.

Кваліфікаційна робота
на здобуття ступеня вищої освіти «бакалавр»
зі спеціальності 122 «Комп'ютерні науки»

на тему:
«Розробка застосунку на базі Rep'Py для вивчення історії України»

Гончарік Данило Сергійович

Керівник кваліфікаційної роботи:
Неділько Віталій Миколайович,
кандидат
технічних наук, доцент

Роботу рекомендовано до захисту
на засіданні кафедри інформаційних
технологій
Протокол № 10 від «06.06» 2024 р.
Завідувач кафедри інформаційних
технологій

О.П. Бондар Бондар О.П.

Роботу захищено на засіданні ЕК
з оцінкою
згідно / Д / 70
(за національною шкалою, шкалою
ECTS, бали)
Протокол № 8 від «29» 06 2024 р.
Голова ЕК _____

АНОТАЦІЯ

Гончарік Д.С. Розробка застосунку на базі Ren'Py для вивчення історії України. – Кваліфікаційна робота зі спеціальності 122 «Комп'ютерні науки». – Економіко-технологічний інститут імені Роберта Ельворті, Кропивницький, 2024.

У кваліфікаційній роботі розглядається розробка застосунку на базі Ren'Py для вивчення історії України. Даний застосунок містить інтерактивні розділи, що включають ключові особи, ключові події та тестування знань користувача. Робота складається з трьох розділів загальним обсягом 84 аркуша. У роботі використано одна таблиця, тридцять один рисунок, використано двадцять літературних джерел та два додатки.

Мета кваліфікаційної роботи полягає у створенні інтерактивного освітнього ресурсу, який би сприяв поглибленню знань користувачів про важливі історичні події шляхом інтерактивних елементів та тестування.

Основними результатами кваліфікаційної роботи є розробка дієвого інструменту для вивчення історії, що поєднує в собі навчання та інтерактивний підхід. Отримані результати свідчать про ефективність використання Ren'Py як платформи для створення освітніх застосунків, а також демонструють можливості інтерактивного навчання в покращенні розуміння та запам'ятовування історичних подій.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	9
1.1 Актуальність теми.....	9
1.2 Ren'Py як платформа для розробки.....	10
1.2.1 Переваги Ren'Py.....	11
1.2.2 Недоліки Ren'Py.....	13
1.3 Приклад застосування: Битва під Жовтими Водами.....	15
1.4 Огляд схожих застосунків.....	18
1.5 Порівняння з існуючими аналогами.....	20
1.6 Постановка завдання.....	23
РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ ЗАСТОСУНКУ.....	27
2.1 Опис UML діаграм.....	27
2.2 Діаграма діяльності.....	32
2.3 Розділ «Ключові особи».....	35
2.4 Розділ «Ключові події».....	37
2.5 Розділ «Тестування».....	39
2.6 Інновації у вирішенні завдання.....	41
2.7 Оцінка повноти вирішення поставленої задачі.....	42
РОЗДІЛ 3. ОПИС ІНТЕРФЕЙСУ КОРИСТУВАЧА.....	48
3.1 Реалізація основних розділів.....	48
3.2 Реалізація карток з інформацією.....	53
3.3 Опис інтерфейсу користувача.....	56
ВИСНОВОК.....	70
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	72
ДОДАТОК А.....	74
ДОДАТОК Б.....	81

ВСТУП

У сучасному світі інформаційні технології проникають у всі сфери життя, включаючи освіту. Використання цифрових технологій в освітньому процесі стає не просто популярною тенденцією, а необхідністю. Інтерактивні освітні застосунки дозволяють учням зануритися в навчальний матеріал, роблячи процес вивчення більш цікавим і ефективним. Особливо це актуально для вивчення історії, яка часто сприймається як сухий та нецікавий предмет. Створення інтерактивного освітнього застосунку на базі платформи Ren'Py є важливим кроком у напрямку модернізації освіти, оскільки дозволяє зробити процес навчання більш захопливим і інтерактивним.

Практичне значення даної роботи полягає у створенні інструменту, що сприяє покращенню якості навчання історії. Розроблений інтерактивний застосунок може бути використаний в освітніх закладах як додатковий ресурс для навчання, що допоможе учням краще засвоювати історичні знання. Крім того, застосунок може використовуватися для самостійного навчання, що робить його корисним інструментом для широкого кола користувачів, включаючи школярів, студентів та всіх, хто цікавиться історією.

На сьогоднішній день існує чимало освітніх застосунків, які використовуються для вивчення різних предметів. Проте, багато з них мають обмежені можливості інтерактивності та не забезпечують достатній рівень залученості учнів у навчальний процес. Часто такі застосунки обмежуються простим викладом матеріалу і не включають інтерактивних елементів, що дозволяють учням активно взаємодіяти з контентом. Це призводить до зниження ефективності навчання та втрати інтересу до предмету. Вибір платформи Ren'Py обумовлений її можливостями створювати інтерактивні історії та навчальні програми з високим рівнем залученості користувачів.

Метою даної кваліфікаційної роботи є розробка інтерактивного освітнього застосунку для вивчення історії на базі платформи Ren'Py.

Основними завданнями роботи є:

1. Проведення аналізу предметної області та існуючих освітніх застосунків.
2. Розробка сценарію інтерактивного навчання, що включає історичні події та постаті.
3. Створення користувацького інтерфейсу, який буде зручним та інтуїтивно зрозумілим.
4. Реалізація інтерактивних елементів, що сприяють активному залученню користувачів у навчальний процес.
5. Проведення тестування застосунку та аналіз його ефективності.

Об'єктом дослідження є процес інтерактивного навчання історії за допомогою освітніх застосунків.

Предметом дослідження є розробка інтерактивного освітнього застосунку на платформі Ren'Py, що забезпечує ефективне вивчення історичних подій та персоналій.

Методи дослідження. У роботі використовуються такі методи дослідження, як аналіз літератури та існуючих освітніх застосунків, методи проектування інтерфейсів та розробки програмного забезпечення, а також методи тестування для оцінки ефективності розробленого застосунку.

Таким чином, розробка інтерактивного освітнього застосунку на платформі Ren'Py є актуальним та практично значущим завданням, яке сприяє підвищенню ефективності та якості навчання історії за допомогою сучасних технологій.

1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Актуальність теми

Вивчення історії завжди викликало живий інтерес у суспільстві, оскільки воно дозволяє нам краще зрозуміти минуле та зв'язки між сучасністю та минулими подіями. Для України, яка має багату та важливу історію, підвищення інтересу до вивчення своєї минувшини є критично важливим.

З розвитком технологій та віртуалізації історичних подій, нові методи навчання стають доступними. Вивчення історії через інтерактивні програми та ігри дозволяє залучити молодше покоління та створити для них цікаве та ефективне середовище навчання.

Актуальність теми полягає також у важливості збереження історичної пам'яті та культурної спадщини. Вивчення історії через інтерактивні застосунки дозволяє зберігати та передавати історичні знання новим поколінням у захопливій та привабливій формі.

У контексті сучасних викликів та конфліктів, розуміння історичних подій стає особливо важливим для формування свідомого та відповідального громадянства. Інтерактивні навчальні застосунки, можуть сприяти розвитку критичного мислення та аналітичних навичок серед користувачів.

Крім того, зростання інтересу до історичних тем та підвищення популярності віртуальних середовищ зумовлює потребу у створенні новаторських та високоякісних навчальних ресурсів. Розробка застосунку на базі Ren'Py для вивчення історії України відповідає цим вимогам, дозволяючи використовувати сучасні технології для досягнення навчальних цілей.

Додатково, історичні ігри та навчальні застосунки можуть стати потужним інструментом для привернення уваги молодого покоління до вивчення історії, особливо в умовах швидко змінюючогося світу та загрози втрати історичної свідомості. Відтак, розробка застосунку на базі Ren'Py для вивчення історії України може сприяти збереженню та популяризації

історичних знань серед молоді, що є важливим аспектом культурного розвитку суспільства.

Застосунок, спрямований на вивчення конкретної історичної події, такої як Битва під Жовтими Водами, може стати особливо цікавим для учнів, адже він дозволяє відчувати та пережити ключові моменти минулого через власні дії та вибори, що сприяє глибшому засвоєнню матеріалу та стимулює інтерес до вивчення історії України.

Таким чином, актуальність теми розробки застосунку на базі Ren'Py для вивчення історії України, зокрема події Битви під Жовтими Водами, очевидна з різних поглядів - від збереження історичної пам'яті до популяризації навчання історії у сучасному інформаційному середовищі.

1.2 Ren'Py як платформа для розробки

Ren'Py - це потужна та гнучка платформа для створення візуальних новел та інтерактивних історій, яка базується на мові програмування Python. Вона володіє рядом переваг, які роблять її ідеальним вибором для розробки навчальних застосунків, зокрема програми для вивчення історії України, такої як Битва під Жовтими Водами [1].

Однією з ключових переваг Ren'Py є його простота використання та доступність для початківців у програмуванні. Завдяки дружньому інтерфейсу та інтуїтивно зрозумілим інструментам, розробники можуть швидко створювати ігри та інтерактивні історії без глибоких знань програмування.

Крім того, Ren'Py надає широкі можливості для налаштування та розширення. Розробники можуть легко керувати візуальним оформленням, анімацією, звуком та іншими аспектами гри, щоб створити унікальний та захоплюючий досвід для користувачів. Ще важливо відзначити, що Ren'Py підтримує різні платформи, включаючи Windows, macOS, Linux та Android, що робить його ідеальним вибором для розробки крос-платформених застосунків [2].

Також Ren'Py відомий також своєю активною спільнотою розробників та великою кількістю додаткових ресурсів, які спрощують процес розробки. Відкритий характер платформи дозволяє розробникам ділитися своїми знаннями, досвідом та навичками, що сприяє активному обміну ідеями та сприяє швидкому розвитку проектів.

Ren'Py має вбудовану підтримку мовних локалізацій, що робить його ідеальним вибором для створення мультилінгвальних навчальних застосунків. Це дозволяє розробникам перекладати контент і адаптувати застосунок для різних аудиторій та культурних контекстів.

Загалом, Ren'Py володіє унікальними характеристиками та можливостями, які роблять його ідеальним інструментом для розробки навчальних застосунків. Його простота використання, гнучкість, підтримка різних платформ та активна спільнота розробників роблять Ren'Py оптимальним вибором для створення застосунків для вивчення історії України.

1.2.1 Переваги Ren'Py

Ren'Py - це потужна платформа для розробки візуальних новел та інтерактивних історій, яка відзначається низкою вагомих переваг:

- Простота використання. Ren'Py має дружній та інтуїтивно зрозумілий інтерфейс, що дозволяє навіть початківцям у програмуванні швидко освоїти створення ігор та інтерактивних історій.
- Гнучкість. Платформа надає широкі можливості для налаштування та розширення, що дозволяє розробникам створювати різноманітні ігри з різними геймплейними механіками та стилями візуального оформлення.
- Підтримка мовних локалізацій. Ren'Py має вбудовану підтримку мультилінгвальних проектів, що дозволяє розробникам легко перекладати контент та адаптувати ігри для різних культурних аудиторій.

- Крос-платформенність. Застосунки, створені в Ren'Py, можуть бути легко збілдені для різних платформ, таких як Windows, macOS, Linux та Android, що робить їх доступними для широкого кола користувачів.
- Активна спільнота розробників. Ren'Py має велику та активну спільноту розробників, яка постійно ділиться досвідом та ресурсами, що сприяє швидкому розвитку та підтримці проектів.
- Python-базова. Ren'Py побудований на мові програмування Python, що надає розробникам широкі можливості для створення складних та інноваційних функціональних можливостей. Python відомий своєю простотою та читабельністю коду, що сприяє швидкому розробленню та підтримці проектів [3].
- Велика документація та ресурси. Ren'Py має широкий набір документації, уроків та прикладів, які допомагають розробникам зрозуміти його функціонал та ефективно використовувати його для створення власних проектів [4].
- Багатомовність. Одна з важливих переваг Ren'Py - можливість підтримки декількох мов одночасно в ігровому контенті, що робить його ідеальним інструментом для міжнародного навчання та розширення аудиторії.
- Наявність стандартів та практик. Ren'Py пропонує стандарти та рекомендації щодо структурування та організації проектів, що допомагає розробникам створювати чистий та ефективний код.
- Підтримка графічних та аудіо ресурсів. Ren'Py дозволяє вбудовувати графічні та аудіо елементи, що робить можливим створення візуально та звуково привабливих навчальних застосунків з великим потенціалом для взаємодії користувача.
- Підтримка анімацій. Ren'Py надає можливість додавати анімації до різних елементів гри, що дозволяє розробникам створювати живі та привабливі інтерактивні сцени [5].

- Можливість створення складних гілок сюжету. За допомогою Ren'Py розробники можуть легко створювати складні гілки сюжету, що дозволяє користувачам впливати на розвиток історії через свої вибори та дії.
- Легка інтеграція з іншими системами. Ren'Py підтримує можливість інтеграції з різними сторонніми системами та сервісами, що робить його ідеальним вибором для розробки навчальних застосунків з різноманітними функціональними можливостями.
- Висока продуктивність. Ren'Py оптимізований для швидкої та ефективної роботи, що дозволяє розробникам швидко розробляти та тестувати свої проекти без зайвих затримок.
- Відкритість та безкоштовність. Ren'Py є вільно розповсюджуваною платформою з відкритим вихідним кодом, що дозволяє розробникам вільно користуватися та модифікувати її для своїх потреб без обмежень.

1.2.2 Недоліки Ren'Py

Незважаючи на свої численні переваги, платформа Ren'Py також має деякі недоліки, які варто врахувати при розробці навчальних застосунків:

- Обмеженість графічного рушію. Ren'Py має обмежені можливості у порівнянні з більш потужними ігровими рушіями, що може обмежувати творчість та функціональність деяких аспектів гри.
- Відсутність підтримки 3D-графіки. Платформа Ren'Py фокусується переважно на 2D-графіці, що робить її неідеальним вибором для проектів, які потребують використання тривимірної графіки.
- Обмежена підтримка для складних ігрових механік. Ren'Py підходить для створення базових ігор та інтерактивних історій, але може бути недостатньо потужним для проектів, що потребують складних ігрових механік та великої кількості інтерактивності.

- Залежність від Python. Хоча використання Python дозволяє легко розширювати функціональність Ren'Py, він також може стати обмеженням для розробників, які не володіють цією мовою програмування.
- Обмежена підтримка для великих проєктів. Ren'Py може виявитися неефективним для створення великих та складних ігрових проєктів через обмежену підтримку та оптимізацію.
- Обмежена підтримка для мобільних платформ. Навіть якщо Ren'Py підтримує певні мобільні платформи, такі як Android, розробка під мобільні пристрої може виявитися складнішою через різноманітність розмірів екранів та характеристик пристроїв.
- Відсутність вбудованих інструментів для мультиплеєра. Ren'Py не надає вбудованих засобів для розробки мережевого мультиплеєра, що може бути недоліком для проєктів, які потребують інтерактивності між користувачами.
- Обмежена підтримка для реалізації складного штучного інтелекту. Ren'Py не є ідеальним вибором для розробки ігор з високим рівнем штучного інтелекту, оскільки не має вбудованих інструментів для створення складних алгоритмів та поведінки персонажів.
- Оптимізація для великих обсягів даних. У разі розробки ігор з великою кількістю контенту (таких як ігрові серіали чи епічні історії) можуть виникнути проблеми з продуктивністю та завантаженням даних через обмежену підтримку Ren'Py для оптимізації обсягу даних.
- Відсутність підтримки для інших платформ. Оскільки Ren'Py сконцентрований переважно на десктопних іграх, він може не бути найкращим вибором для розробки застосунків для інших платформ, таких як консолі або веб-сервери.
- Обмежена графічна можливість. Хоча Ren'Py надає можливості для створення візуально привабливих ігор та інтерактивних історій, його можливості в області графічного дизайну можуть бути обмеженими

порівняно з іншими ігровими рушіями, що пропонують більше розширених графічних інструментів.

- Відсутність підтримки для високоякісних графічних ефектів. Оскільки Ren'Py більше спрямований на розробку візуальних новел та інтерактивних історій, він може не мати підтримки для деяких сучасних графічних ефектів, які широко використовуються в інших типах ігор.
- Обмежені можливості розвитку. У порівнянні з іншими ігровими движками, Ren'Py може мати обмежені можливості для розвитку та масштабування проєктів, що може обмежити можливості розробників створити складні та амбіційні ігри.

1.3 Приклад застосування: Битва під Жовтими Водами

Битва під Жовтими Водами є однією з найважливіших та епічних подій в історії України. Ця битва, що відбулася 16 травня 1648-го року, між військом Богдана Хмельницького та польської армії під керівництвом Стефана Потоцького, мала вирішальне значення для подальшої долі українських земель [6].

Створення застосунку на базі Ren'Py, присвяченого події Битви під Жовтими Водами, може відкрити нові можливості для вивчення цієї важливої події та стимулювати інтерес учнів до вивчення історії України.

Ось деякі приклади того, як може бути використаний такий застосунок:

- Інтерактивне вивчення події. Користувачі можуть поглибитися в сюжет та події Битви під Жовтими Водами через інтерактивні сценарії та діалоги. Вони можуть взяти участь у рішеннях та діях князя Данила Галицького та інших історичних постатей під час битви.
- Виконання завдань та випробувань. Через виконання завдань та випробувань, користувачі можуть перевірити свої знання про події, які передували та відбувалися під час Битви під Жовтими Водами. Вони

можуть випробувати свої навички у битвах, вирішувати стратегічні завдання та приймати рішення, які впливають на хід подій.

- Візуалізація історичних джерел. Застосунок може включати в себе відтворення або адаптацію історичних джерел, таких як хроніки, літературні твори чи картографічні матеріали про події Битви під Жовтими Водами. Це дозволить користувачам отримати зв'язок між реальними історичними записами та віртуальним відтворенням подій у застосунку.
- Створення інтерактивних уроків. Застосунок може слугувати не лише як засіб розважання, а й як освітній інструмент. Шляхом включення вікторин, завдань, тестів та питань для обговорення, він може стати ефективним інтерактивним уроком для вивчення історії України та розвитку критичного мислення учнів.
- Залучення до аналізу та обговорення. Застосунок може включати в себе елементи, які стимулюють аналіз та обговорення ключових аспектів Битви під Жовтими Водами, таких як стратегії, мотивації та наслідки. Це сприятиме розвитку аналітичних навичок учнів та сприятиме глибшому розумінню історичних подій.
- Впровадження різних режимів гри. Застосунок може включати різноманітні режими гри, такі як навчальний режим для вивчення фактів та історичних подій, режим випробування для перевірки знань, та режим відтворення для дослідження альтернативних сценаріїв та рішень.
- Підтримка використання в навчальних програмах. Застосунок може стати частиною навчальних програм з історії України як додатковий ресурс для вивчення та поглиблення знань про Битву під Жовтими Водами. Його можна використовувати як домашнє завдання або класну діяльність з метою стимулювання інтересу до вивчення історії.
- Використання мультимедійних ресурсів. Застосунок може включати в себе відео-, аудіо- та графічні матеріали, які допоможуть учням краще

зрозуміти та відчувати атмосферу Битви під Жовтими Водами. Візуальні та аудіо елементи можуть бути використані для створення емоційного зв'язку з подією та її учасниками.

- Розширення меж навчання. Застосунок може включати додаткові матеріали, які розширюють межі навчання та дозволяють учням глибше досліджувати тему Битви під Жовтими Водами. Це може включати в себе біографії відомих постатей, аналіз стратегій битви та її вплив на подальшу історію України.
- Залучення до творчості. За допомогою застосунку, учні можуть брати участь у творчих завданнях, таких як написання альтернативних сценаріїв для подій Битви під Жовтими Водами або створення власних історичних артефактів.
- Підвищення мотивації до навчання. Інтерактивний та захопливий характер застосунку може підвищити мотивацію учнів до вивчення історії України. Можливість відчувати себе частиною історичних подій, це може зробити навчання цікавішим та захопливим.
- Збільшення доступності навчального матеріалу. Застосунок буде доступний для завантаження на різних платформах, що робить його легко доступним для використання в школах, бібліотеках або вдома, забезпечуючи широкий доступ до навчального матеріалу про Битву під Жовтими Водами.

Використання розробленого застосунку на основі Ren'Py для вивчення історії України на конкретному прикладі - Битви під Жовтими Водами, свідчить про значний потенціал цієї платформи в освітніх цілях.

Отже, приклад застосування застосунку на основі Битви під Жовтими Водами підкреслює його важливість у покращенні процесу вивчення історії, а також його потенціал у створенні цікавих та ефективних освітніх матеріалів. Такий застосунок не лише надасть користувачам можливість вивчати історію, але й стимулює їхню активність та інтерактивність у навчанні. Він дозволить зануритися у

ключові моменти та пережити емоції, пов'язані з цією важливою подією в історії України.

1.4 Огляд схожих застосунків

«The Expression Amrilato» — це візуальна новела, яка використовується для вивчення есперанто [7].

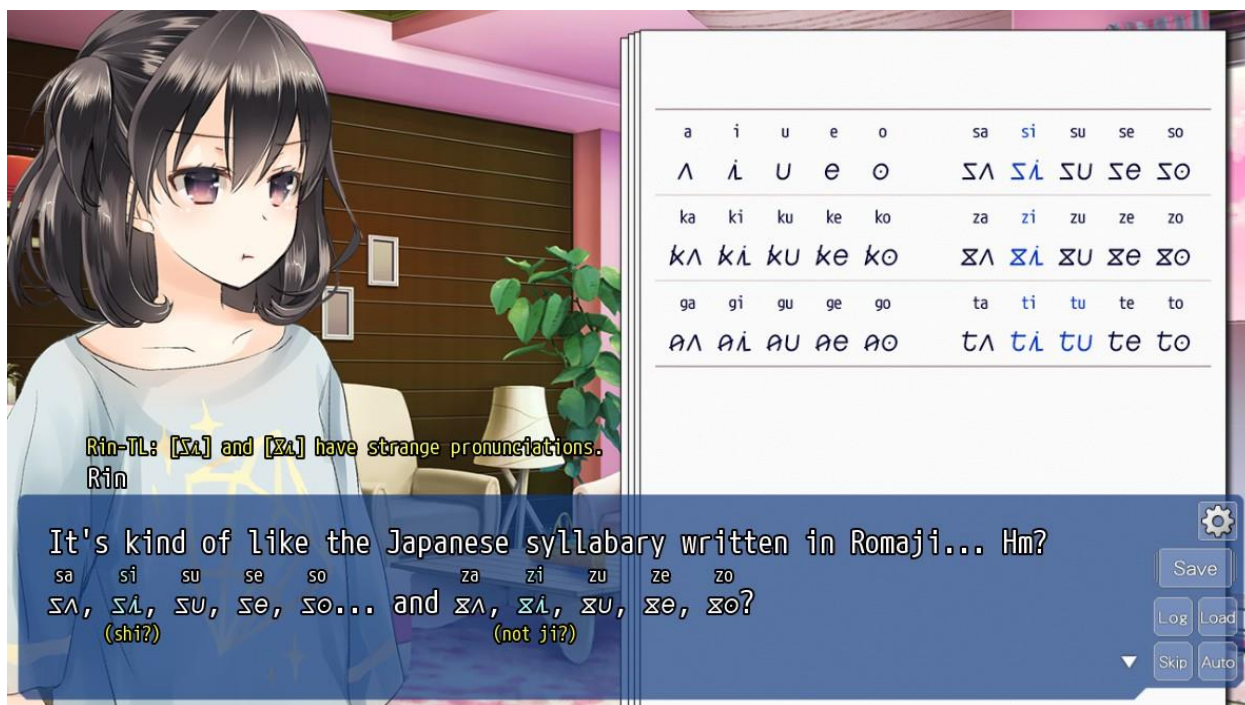


Рис. 1.4.1 - «The Expression Amrilato»

Переваги «The Expression Amrilato»:

- Залучення користувача. Користувач взаємодіє з сюжетом і персонажами, що робить процес навчання більш залучаючим. Це сприяє підвищенню інтересу до мови та мотивації до навчання.
- Практика у реальних ситуаціях. Гра моделює реальні життєві ситуації, що дозволяє Користувачам застосовувати нові знання на практиці.
- Краще засвоєння матеріалу. Навчання в контексті допомагає учням зрозуміти, як використовувати слова та фрази в реальних ситуаціях, що покращує запам'ятовування.

- Природне вивчення граматики. Граматичні конструкції вивчаються через діалоги та текст, а не через окремі правила, що робить процес більш природним.
- Візуальні підказки. Ілюстрації та візуальні ефекти допомагають асоціювати слова з образами, що полегшує їх запам'ятовування.
- Аудіо супровід. Прослуховування правильного вимовляння слів та фраз сприяє розвитку навичок сприйняття мови на слух та правильної вимови.
- Планомірний підхід. Гра поступово ускладнює матеріал, що дозволяє учням адаптуватися до нового рівня знань без перевантаження.
- Підтримка мотивації. Поступове ускладнення допомагає підтримувати інтерес, оскільки Користувачі постійно стикаються з новими викликами.
- Безлімітне повторення. Учні можуть повторювати будь-які розділи гри стільки разів, скільки їм потрібно для повного засвоєння матеріалу.
- Контроль за навчанням. Користувачі самі можуть обирати, які частини гри вони хочуть переглянути ще раз для закріплення знань.
- Гейміфікація. Елементи гри, такі як досягнення та нагороди, стимулюють учнів продовжувати навчання.
- Емоційний зв'язок. Завдяки захоплюючому сюжету та персонажам, учні можуть відчувати емоційний зв'язок з матеріалом, що підвищує мотивацію до навчання.

Недоліки «The Expression Amrilato»:

- Обмежений словниковий запас. Гра може не охоплювати всіх аспектів мови, що необхідні для вільного володіння.
- Базовий рівень. Матеріал гри може бути спрямований переважно на початковий рівень, що не задовольняє потреби просунутих учнів.
- Залежність від сюжету.

- Суб'єктивне сприйняття. Якщо сюжет гри не сподобається учню, це може знизити його мотивацію до навчання.
- Обмеження в різноманітності тем. Сюжет гри може не охоплювати всі теми, які можуть бути цікавими або корисними для учнів.
- Обмежений зворотний зв'язок. Гра не надає можливості отримувати зворотний зв'язок від викладача або носія мови, що є важливим для покращення мовних навичок.
- Відсутність розмовної практики. Гра не може замінити живе спілкування з носіями мови, що є ключовим елементом вивчення будь-якої мови.
- Недостатність для просунутих учнів. Гра може бути більше орієнтована на новачків і не задовольнити потреби учнів, які вже володіють базовими знаннями.
- Відсутність поглибленого навчання. Для досягнення вищого рівня володіння мовою можуть знадобитися додаткові ресурси та методи.
- Можливі технічні проблеми. Користувачі можуть зіткнутися з технічними проблемами, які можуть заважати процесу навчання.

«The Expression Amrilato» пропонує інноваційний підхід до вивчення есперанто, поєднуючи гейміфікацію з освітніми елементами. Це робить гру чудовим інструментом для початківців. Однак, для повного оволодіння мовою необхідно доповнювати гру іншими методами та ресурсами.

1.5 Порівняння з існуючими аналогами

У цьому розділі буде проведено порівняння мого проекту з існуючим аналогом — візуальною новелою «The Expression Amrilato». Це порівняння допоможе визначити унікальні особливості мого проекту, а також виявити можливості для покращення на основі аналізу конкурентних переваг та недоліків аналога.

«The Expression Amrilato» є освітньою візуальною новелою, яка використовує захоплюючий сюжет та інтерактивні елементи для навчання

користувачів мови есперанто. Гра поєднує в собі навчальні матеріали з інтерактивною розповіддю, що робить процес навчання більш приємним і цікавим[7].

Критерії порівняння:

- Функціональність - набір функцій, які надає кожен проект.
- Інтерфейс користувача - зручність та інтуїтивність користування інтерфейсом.
- Інтерактивність - рівень взаємодії користувача з системою.
- Навчальний контент - якість та кількість навчальних матеріалів.
- Адаптивність - здатність системи до адаптації під різні пристрої та платформи.

Функціональність:

- The Expression Amrilato - має інтерактивну сюжетну лінію з елементами навчання есперанто, включаючи словники та вправи. Основний акцент робиться на поєднанні навчання та розваги.
- Мій проект - пропонує широкий набір функцій для інтерактивного навчання, включаючи різноманітні тестові завдання, навчальні матеріали та можливість інтерактивної взаємодії з контентом.

Інтерфейс користувача:

- The Expression Amrilato - інтерфейс розроблений для легкого занурення у сюжет, з простими та зрозумілими елементами управління, але може бути дещо перевантаженим для новачків.
- Мій проект - інтуїтивний і сучасний інтерфейс, адаптований для різних платформ та пристроїв, що робить його доступним для ширшої аудиторії.

Інтерактивність:

- The Expression Amrilato - високий рівень інтерактивності завдяки інтеграції навчальних вправ у сюжетну лінію. Користувачі активно взаємодіють із персонажами та завданнями.

- Мій проект - забезпечує високу інтерактивність через різноманітні навчальні завдання та тести, що стимулюють активне залучення користувачів до процесу навчання.

Навчальний контент:

- The Expression Amrilato - основний акцент на навчання мови есперанто через сюжетну лінію та інтерактивні вправи. Містить словники та граматичні довідники.
- Мій проект - може включати широкий спектр навчальних матеріалів з різних предметів, що дозволяє користувачам отримувати знання в різних областях. Контент адаптований для інтерактивного навчання та перевірки знань.

Адаптивність:

- The Expression Amrilato - гра доступна на кількох платформах, включаючи ПК та мобільні пристрої, але може мати обмеження в адаптивності для деяких пристроїв.
- Мій проект - висока адаптивність для різних пристроїв і платформ, забезпечує оптимальну роботу на мобільних та настільних пристроях, що підвищує доступність для користувачів.

Порівняння з «The Expression Amrilato» показало, що мій проект має кілька ключових переваг, включаючи ширшу функціональність, сучасний інтерфейс та високу адаптивність. Водночас, успішні елементи «The Expression Amrilato», такі як інтеграція навчального контенту в сюжетну лінію та високий рівень інтерактивності, можуть бути використані для подальшого покращення мого проекту. Аналіз конкурентних переваг та недоліків дозволяє нам чітко розуміти, у яких напрямках слід розвивати мій продукт, щоб забезпечити його конкурентоспроможність та привабливість для користувачів.

1.6 Постановка завдання

Метою кваліфікаційної роботи є розробка інтерактивного застосунку на базі Ren'Py з метою покращення процесу вивчення історії України. Дане дослідження буде спрямоване на створення навчального середовища, яке надасть користувачам можливість поглибленого вивчення ключових подій, постатей та періодів історії України на прикладі Битви під Жовтими Водами через інтерактивні та візуальні матеріали.

Завдання кваліфікаційної роботи:

1. Розробка візуальних та аудіо-візуальних матеріалів. Створення відео-, аудіо- та графічних матеріалів, які ілюструватимуть ключові події та періоди в історії України.
 - Створення графічних елементів. Включає в себе розробку зображень, ілюстрацій та графічних сцен для візуалізації історичних подій, постатей та місць.
 - Створення відеоматеріалів. Може включати відео-анімації, документальні фільми, відеоролики та інші візуальні матеріали, які допоможуть користувачам краще зрозуміти історичні події та процеси.
 - Інтеграція вмісту. Візуальні та аудіо-візуальні матеріали повинні бути інтегровані в застосунок з урахуванням його структури та інтерфейсу. Вони можуть використовуватися в різних частинах застосунку, таких як інтерактивні сцени, навчальні модулі, тестування та інше.
2. Інтеграція інтерактивних елементів. Розробка інтерактивних завдань, тестів та вправ, які дозволять користувачам активно залучатися до процесу вивчення та перевірки знань.
 - Відповідальність за вибір. Користувачі можуть мати можливість обирати між різними сценаріями або варіантами відповідей під час виконання завдань або вирішення історичних ситуацій.

- Інтерактивні тестування. Створення тестів або вікторин, де користувачі можуть перевіряти свої знання з історії України та отримувати миттєвий зворотний зв'язок щодо правильності їх відповідей.
- Графічні ілюстрації з анімацією. Використання анімаційних ефектів та анімованих ілюстрацій для покращення візуального враження та залучення уваги користувачів.
- Інтерактивні картки з історичними фактами. Створення віртуальних флеш-карток з історичними подіями, постатями та термінами, які користувачі можуть переглядати та вивчати.
- Графічні інтерфейси для навігації. Розробка інтуїтивно зрозумілих графічних інтерфейсів, які дозволяють користувачам легко переміщатися між різними частинами застосунку та взаємодіяти з ними.

3. Створення зручного інтерфейсу користувача. Розробка інтуїтивно зрозумілого та зручного інтерфейсу, який спростить навігацію та використання застосунку для користувачів будь-якого рівня підготовки.

- Навігаційна структура. Інтерфейс повинен мати чітку та логічну навігаційну структуру, яка дозволяє користувачам легко переміщатися між різними розділами та функціями застосунку.
- Привабливий дизайн. Графічний дизайн повинен бути привабливим та естетично збалансованим, з використанням зручних для очей кольорів та зображень.
- Зручність використання. Користувачі повинні легко зрозуміти, як взаємодіяти з інтерфейсом та як виконувати необхідні дії. Для цього можуть бути використані звичайні UI елементи, такі як кнопки, меню, списки тощо.

- Адаптивність. Інтерфейс повинен бути адаптивним до різних розмірів екрану та пристроїв, щоб забезпечити зручність використання як на комп'ютерах, так і на мобільних пристроях.
- Зручне управління. Управління застосунком повинно бути легким та інтуїтивно зрозумілим, дозволяючи користувачам легко переміщатися між різними розділами та взаємодіяти з контентом.
- Збереження прогресу. Застосунок повинен мати можливість автоматичного збереження прогресу користувача, щоб вони могли повернутися до раніше виконаних завдань або вправ без втрати даних.
- Зворотний зв'язок. Важливо надати користувачам можливість зворотного зв'язку щодо їхнього досвіду використання застосунку, наприклад, через форми зворотного зв'язку або кнопки «Зв'язатися з нами».

4. Тестування та вдосконалення. Проведення тестування застосунку з метою виявлення помилок, аналізу зручності та ефективності, а також внесення необхідних змін та вдосконалень.

- Функціональне тестування. Треба переконатись, що всі функції застосунку працюють правильно та відповідають вимогам. Це включає в себе тестування навігації, інтерактивних елементів, тестів та вправ, а також взаємодії з контентом.
- Тестування на різних пристроях. Впевнитись, що застосунок працює коректно на різних пристроях та в різних роздільних здатностях екрану, включаючи комп'ютери, планшети та смартфони.
- Вдосконалення та виправлення помилок. На основі результатів тестування ввести необхідні виправлення та вдосконалення до застосунку, щоб забезпечити максимальну якість та зручність для користувачів.

Основна мета роботи полягає в створенні інтерактивного та привабливого навчального середовища, яке дозволить користувачам глибше зануритися в історію України та отримати цікаву та ефективну освіту.

Постановка завдання визначила ключові напрямки дослідження та розробки, включаючи, розробку візуальних матеріалів, інтеграцію інтерактивних елементів, розробку зручного інтерфейсу користувача, тестування та вдосконалення застосунку.

В результаті постановки завдання було визначено чіткі цілі та завдання для подальшої роботи, які спрямовані на створення високоякісного та ефективного застосунку для вивчення історії України. Такий підхід дозволить досягти успішних результатів у розробці програмного забезпечення та забезпечити задоволення потреб користувачів у навчанні історії.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ЗАСТОСУНКУ

2.1 Опис UML діаграм

UML (Unified Modeling Language) – це стандартний мова моделювання, яка використовується для візуалізації, специфікації, побудови та документування програмних систем. UML надає розробникам засоби для створення абстрактних моделей програмного забезпечення, що допомагає зрозуміти та спроектувати складні системи. Основні типи діаграм UML включають діаграми класів, діаграми діяльності, діаграми випадків використання, діаграми послідовностей та інші [8].

Використання UML діаграм у процесі розробки програмного забезпечення має кілька переваг:

- Візуалізація. Діаграми дозволяють візуалізувати структуру та поведінку системи, що полегшує розуміння та аналіз.
- Комунікація. UML діаграми сприяють кращій комунікації між членами команди, замовниками та іншими зацікавленими сторонами.
- Документування. UML діаграми служать важливими документами, що відображають архітектуру та дизайн системи.

Діаграма класів (Class Diagram) – це тип діаграми UML, який показує структуру системи шляхом моделювання класів, їх атрибутів, методів, а також взаємозв'язків між класами. Діаграми класів є важливими для об'єктно-орієнтованого проектування, оскільки вони допомагають зрозуміти структуру та взаємодію об'єктів у системі.

Діаграми класів використовуються для візуалізації статичної структури системи, яка включає класи та відносини між ними. Це дозволяє розробникам зрозуміти, як різні частини програми взаємодіють між собою та як дані організовані в системі.

Основні компоненти діаграми класів:

- Клас (Class) - основний будівельний блок діаграми, який представляє сутність з атрибутами (властивостями) та методами (функціями).
- Атрибут (Attribute) - властивість класу, яка зберігає дані.
- Метод (Method) - функція або операція, яку клас може виконувати.
- Асоціація (Association) - відношення між двома класами, яке показує, як класи взаємодіють один з одним.
- Наслідування (Inheritance) - відношення, яке показує, що один клас є підкласом іншого і успадковує його атрибути та методи.
- Агрегація (Aggregation) - відношення між класами, де один клас містить інший клас як частину.
- Композиція (Composition) - більш жорстка форма агрегації, де життєвий цикл включеного класу залежить від життєвого циклу контейнера.

Діаграма послідовності (Sequence Diagram) – це тип діаграми UML, який показує, як об'єкти взаємодіють один з одним у певній послідовності, тобто як вони обмінюються повідомленнями в рамках певного сценарію або процесу. Ці діаграми є корисними для візуалізації динамічних аспектів системи та розуміння порядку виконання дій [9].

Діаграми послідовності використовуються для моделювання часової послідовності взаємодії між об'єктами в рамках окремих сценаріїв. Це дозволяє деталізувати сценарії використання та зрозуміти, як різні частини системи взаємодіють між собою під час виконання конкретних завдань.

Основні компоненти діаграми послідовності:

- Об'єкт (Object) - екземпляр класу, який бере участь у сценарії.
- Життєва лінія (Lifeline) - вертикальна лінія, яка представляє існування об'єкта протягом часу.
- Повідомлення (Message) - стрілки, які показують обмін інформацією між об'єктами.

- Активність (Activation) - прямокутники на життєвих лініях, які показують періоди активності об'єкта.
- Початок (Start) та кінець (End) - позначають початок і кінець сценарію.

Діаграма варіантів використання (Use Case Diagram) – це тип діаграми UML, яка описує взаємодію користувача (актора) з системою для досягнення певної мети. Діаграми варіантів використання фокусуються на функціональності системи з точки зору кінцевого користувача і показують, які дії може виконувати користувач та як ці дії взаємопов'язані [10].

Діаграми варіантів використання використовуються для ідентифікації функціональних вимог системи та визначення всіх можливих способів її використання. Вони допомагають зрозуміти очікувану поведінку системи та забезпечують основу для розробки сценаріїв тестування.

Основні компоненти діаграми варіантів використання:

- Актор (Actor) - зовнішній користувач або система, яка взаємодіє із системою для виконання певної дії.
- Варіант використання (Use Case) - описує певну функціональність або дію, яку може виконувати актор у системі.
- Зв'язок (Association) - лінії, які показують взаємодію між акторами та варіантами використання.
- Система (System) - межа, яка охоплює всі варіанти використання, що відносяться до конкретної системи.

Діаграма станів (State Machine Diagram) є одним з типів UML діаграм, який використовується для моделювання станів об'єктів або системи та переходів між цими станами. Вона дозволяє показати, як об'єкт або система реагує на входні події, змінюючи свій стан та виконуючи відповідні дії [11].

Основні елементи діаграми станів:

- Стан (State) - певний стан, в якому може перебувати об'єкт або система. Наприклад, «ввімкнено», «вимкнено», «очікування», тощо.
- Перехід (Transition) - перехід показує, як система переходить з одного стану в інший після виникнення певної події або умови.

- Подія (Event) - зміна у середовищі, яка викликає перехід між станами.
- Умова (Guard) - умова, яка має бути виконана для того, щоб перехід між станами відбувся.
- Дія (Action) - дія, яка виконується при входженні в певний стан або виходженні з нього.

Діаграма станів може бути корисною для відображення складної поведінки системи, такої як управління станами виконання, керування потоком даних або реакція на події. Вона дозволяє легко розуміти, як система реагує на різні ситуації та події, і які дії вона при цьому виконує.

Діаграма прецедентів (Use Case Diagram) - це один з основних типів UML діаграм, який використовується для моделювання функціональності системи з точки зору її користувачів. Головна мета цієї діаграми - показати функціональність системи, описуючи різні дії або сценарії взаємодії між користувачами та самою системою [12].

Основні елементи діаграми прецедентів:

Прецедент (Use Case) - Це функціональний або діловий вимір системи, який відображає конкретний функціональний або діловий процес, який може бути виконаний користувачем або іншою системою. Кожен прецедент зазвичай має назву, яка описує дію, наприклад, «Створити замовлення» або «Відправити повідомлення».

- Актор (Actor) - це будь-яка зовнішня сутність або роль, яка взаємодіє з системою. Актор може бути реальним користувачем, іншою системою або зовнішнім об'єктом. Наприклад, Клієнт, Адміністратор, Система оплати.
- Відношення (Relationships) - ці відношення вказують на взаємодію між акторами та прецедентами. Основні типи відношень включають:
 - Взаємодія (Association) - зв'язок між актором і прецедентом.
 - Включення (Inclusion) - показує, що один прецедент включає в себе інший.

- Розширення (Extension) - показує, що один прецедент може розширювати інший.

Діаграма прецедентів дозволяє визначити, як користувачі будуть використовувати систему для досягнення своїх цілей, відображаючи функції та можливості, які система надає. Вона є потужним інструментом для спілкування з зацікавленими сторонами та визначення вимог до системи.

Діаграма розгортання (Deployment Diagram) - це один з видів структурних діаграм UML, який використовується для моделювання фізичної архітектури системи. Головна мета цієї діаграми - показати розміщення фізичних компонентів системи, таких як апаратне забезпечення та програмне забезпечення, і їх взаємозв'язки [13].

Основні елементи діаграми розгортання:

- Вузол (Node) - це фізичний об'єкт або обладнання, яке може бути використане для розгортання програмного забезпечення. Це може бути сервер, комп'ютер, мобільний пристрій, тощо.
- Артефакт (Artifact) - це програмне забезпечення або файл, яке може бути розгорнуте на вузлі. Артефакти представляють файли, бібліотеки, виконувані файли, конфігураційні файли тощо.
- Взаємозв'язок (Association) - вказує на зв'язок між вузлами та артефактами. Наприклад, вузол може мати артефакт, який розгортатиметься на цьому вузлі.
- Вміст вузла (Node Contents) - це список артефактів, які розгортані на певному вузлі.
- Умовне зображення (Stereotype) - допоміжні позначки, які дозволяють додатково характеризувати вузли та артефакти (наприклад, веб-сервер, база даних, клієнтське додаток і т. д.).

Діаграма розгортання дозволяє відобразити фізичну архітектуру системи, зокрема розташування апаратного забезпечення та програмного забезпечення. Вона є корисним інструментом для планування та аналізу

архітектури системи, а також для визначення вимог до інфраструктури для успішного розгортання програмного продукту.

2.2 Діаграма діяльності

Я обрав для своєї кваліфікаційної роботи діаграму діяльності. Діаграма діяльності (Activity Diagram) – це тип діаграми UML, який показує динамічні аспекти системи. Вона описує послідовність дій або операцій, які виконуються у процесі виконання певної задачі або сценарію. Діаграми діяльності є корисними для моделювання логіки бізнес-процесів, потоків робіт та алгоритмів.

Діаграма діяльності використовується для візуалізації процесу взаємодії користувача з програмою, включаючи основні дії, умови переходів та послідовність кроків. Це дозволяє визначити всі можливі шляхи виконання задачі та забезпечити правильну реалізацію логіки програми [10].

Основні компоненти діаграми діяльності:

- Дія (Activity) - окремий крок або операція, яку виконує система.
- Перехід (Transition) - стрілки, що показують послідовність виконання дій.
- Умова (Condition) - логічні вирази, які визначають, який шлях вибрати в залежності від виконаних умов.
- Початок (Start) та кінець (End) - спеціальні вузли, які позначають початок і кінець процесу.

Часто виникає питання про різницю між діаграмою діяльності та діаграмою послідовності. Основна мета діаграми послідовності — показати черговість виконання дій. Водночас діаграма діяльності використовується для опису функціонування всієї системи, демонструючи перехід між діями.

Ці дії можуть виконуватися людьми, програмними компонентами або комп'ютерами. Потік керування на діаграмі діяльності відображає послідовний, розгалужений або паралельний перехід від однієї операції до іншої.

Потік керування з'єднує два вузли на діаграмі активності. Вузол розгалуження (вилка) — це керуючий вузол, який розділяє потік на кілька одночасних, створюючи кілька паралельних завдань. Вузол злиття об'єднує ці потоки назад.

Вузол прийняття рішень — це керуючий вузол, який вибирає один істинний потік з декількох можливих. Він аналогічний оператору if у Java або C#. Умови (охоронні умови) записуються в квадратних дужках.






Рис 2.2.1 – Діаграма діяльності застосунку

Таблиця 2.2.1

Позначення діаграми діяльності

Символ	Ім'я	Використання
	Початковий вузол	Відправна точка, або початковий стан
	Дія	Представлення діяльності, завдання для виконання
	Потік керування	Спрямований потік, контрольний потік діяльності
	Кінцевий вузол активності	Кінцевий стан, завершення усіх потоків процесу
	Кінцевий вузол потоку	Кінець одного потоку
	Вузол прийняття рішення	Розгалуження з умовою та кількома варіантами дій. Має один вхід декілька виходів
	Вузол злиття	Об'єднання потоків створених вузлом прийняття рішень. Має кілька входів і один вихід
	Вилка	Розподілення потоку на кілька паралельних без прийняття рішення
	Злиття	Об'єднання декількох паралельних потоків

	Надсилання сигналу	Вказує на те що сигнал надсилається на приймальну діяльність
	Отримання сигналу	Вказує на отримання сигналу
	Коментар	Дозволяє робити коментарі до діаграми. З'єднується пунктирною лінією

2.3 Розділ «Ключові особи»

У цьому розділі використовується UML-діаграма діяльності для моделювання процесу вибору історичної постаті та проходження мінігри.

Пояснення UML-діаграми діяльності:

- Користувач запускає програму та вибирає розділ «Ключові особи».
- Система відображає список доступних історичних постатей для вибору.
- Користувач обирає одну з постатей та переходить до наступного кроку.
- Користувач обирає один з трьох рівнів складності: легкий, нормальний або складний.
- Система ініціює початок мінігри «Memoria» з обраними параметрами.
- Користувач спробує зібрати пари карток.
- Система відображає інформацію про успішність користувача після завершення мінігри.

Нижче представлена UML-діаграма діяльності, яка ілюструє послідовність дій користувача під час використання розділу «Ключові особи».

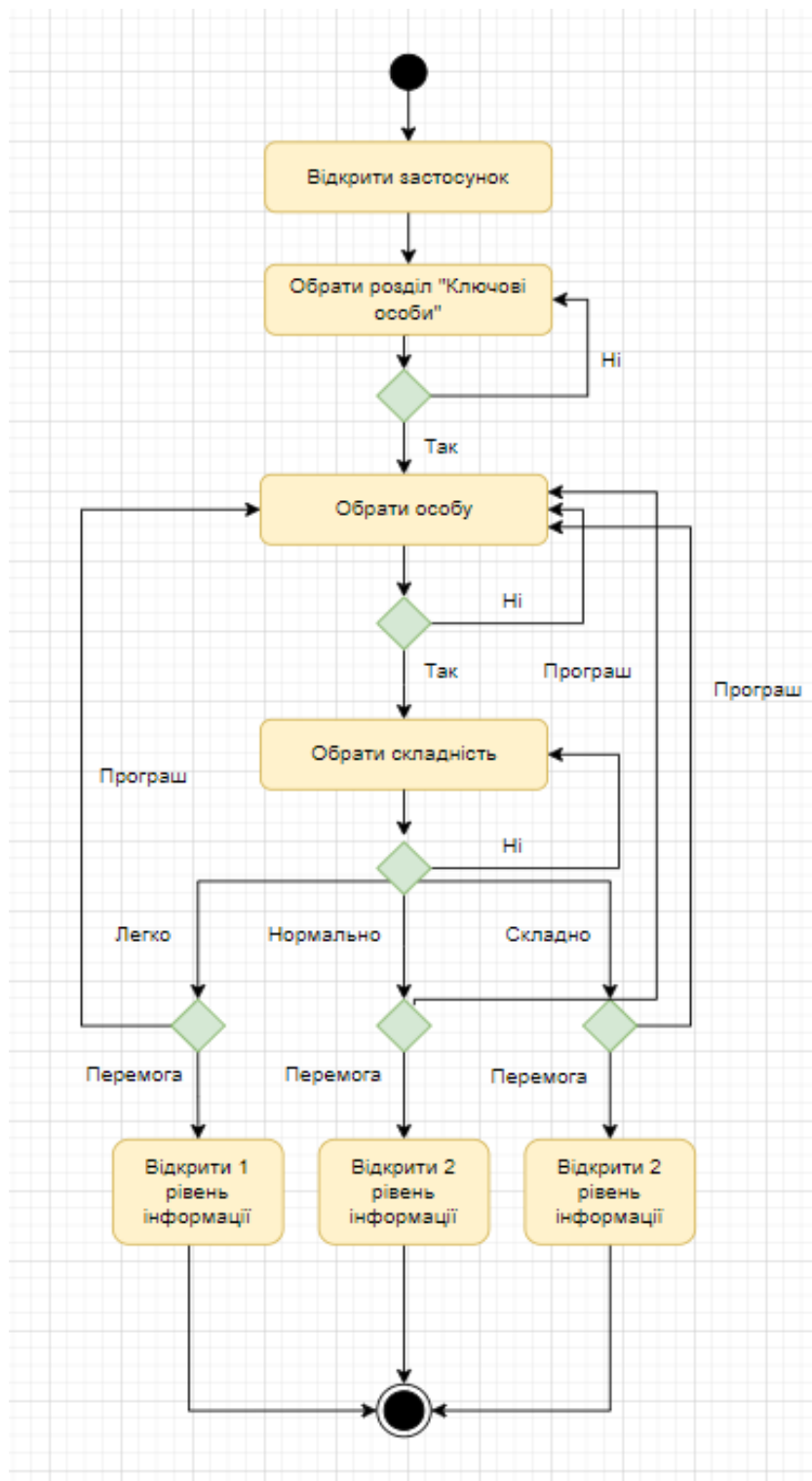


Рис. 2.3.1 - Діаграма діяльності розділу «Ключові особи»

Ця UML-діаграма діяльності наглядно демонструє послідовність дій користувача та системи при використанні розділу «Ключові особи», що дозволяє краще зрозуміти процес вибору історичної постаті та проходження мінігри.

2.4 Розділ «Ключові події»

У цьому розділі використовується UML-діаграма діяльності для моделювання процесу вибору та дослідження ключових історичних подій з можливістю інтерактивної взаємодії з користувачем.

Пояснення UML-діаграми діяльності:

- Користувач запускає програму та обирає розділ «Ключові події» з головного меню.
- Система відображає список ключових історичних подій для вибору.
- Користувач обирає одну з подій, яку хоче дослідити.
- Користувач отримує доступ до інформації про обрану історичну подію, яка включає опис, дати, учасників тощо.
- Під час перегляду інформації про подію користувачеві будуть задаватися питання, це додає елемент інтерактивності у навчання.
- Після перегляду інформації, користувач має можливість повернутися на екран вибору історичних подій для дослідження інших подій або вибрати інший розділ для дослідження.

Нижче наведено UML-діаграму діяльності, що ілюструє послідовність дій користувача під час використання розділу «Ключові події».

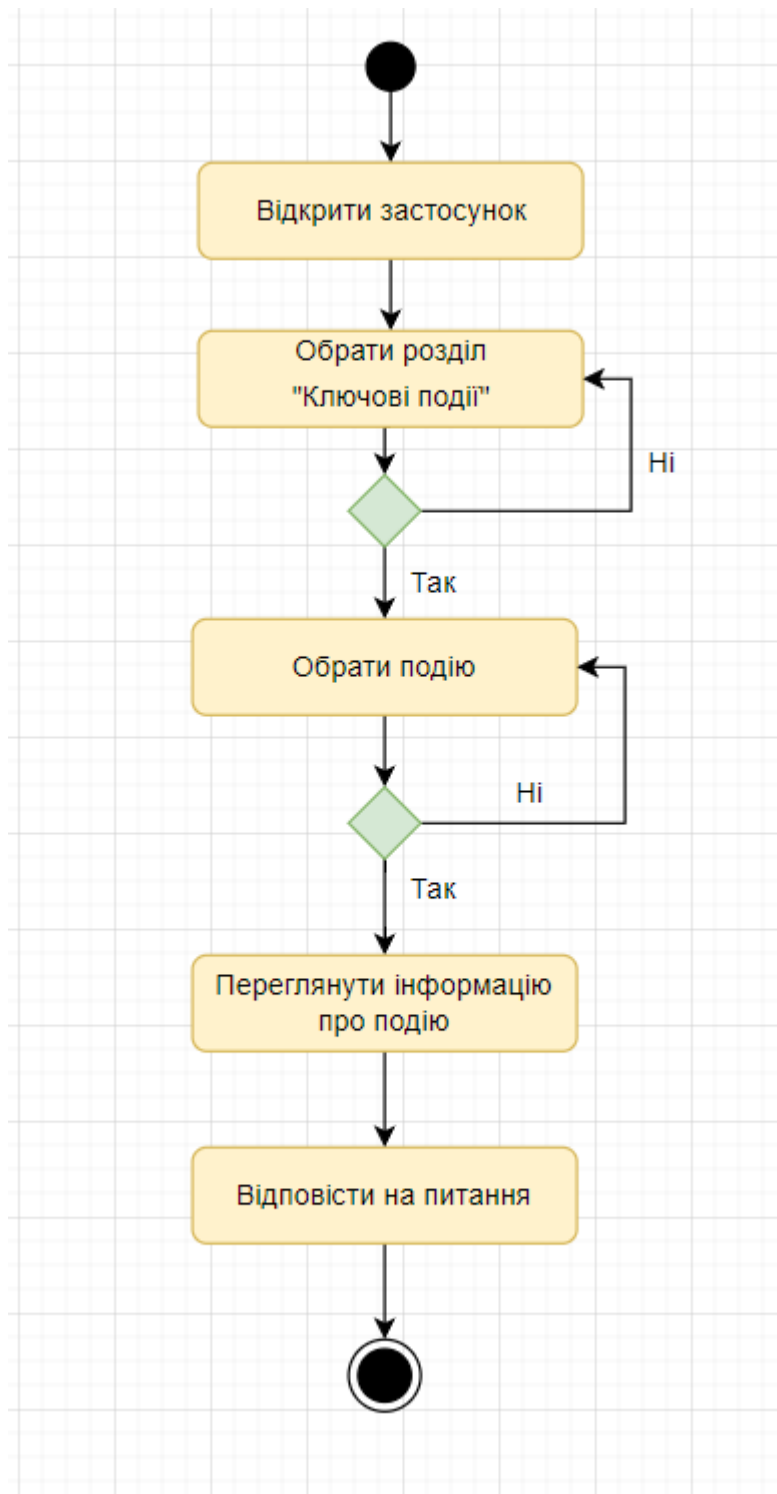


Рис. 2.3.2 - Діаграма діяльності розділу «Ключові події»

Ця UML-діаграма діяльності ілюструє послідовність дій користувача під час використання розділу «Ключові події» і дозволяє краще зрозуміти процес вибору та дослідження ключових історичних подій України.

2.5 Розділ «Тестування»

У цьому розділі використовується UML-діаграма діяльності для моделювання процесу тестування знань користувача щодо історичних подій та осіб

Пояснення UML-діаграми діяльності:

- Користувач запускає програму та обирає розділ «Тестування» з головного меню.
- Система запускає процес тестування знань користувача.
- Система відображає перше питання з тесту.
- Користувач надає свою відповідь на питання.
- Система перевіряє правильність відповіді користувача.
- Якщо є ще питання, система відображає наступне питання.
- Якщо всі питання вичерпані, система завершує процес тестування.
- Система відображає результати тестування, такі як загальна кількість правильних відповідей.
- Після перегляду результатів тестування, користувач може повторно пройти тест або повернутися на головний екран для вибору іншого розділу.

Нижче наведено UML-діаграму діяльності, яка ілюструє цей процес.

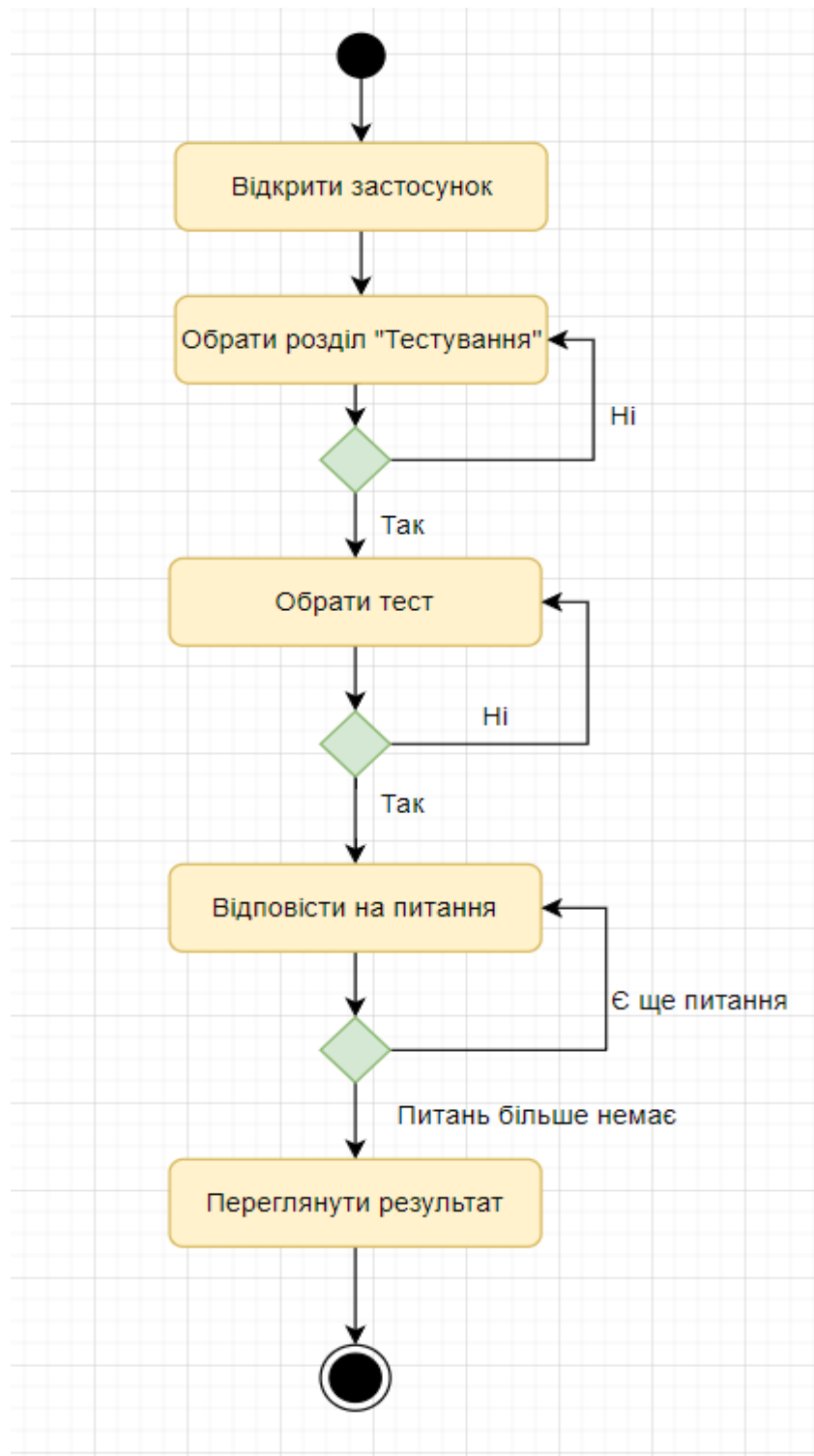


Рис. 2.3.2 - Діаграма діяльності розділу «Тестування»

Ця UML-діаграма діяльності ілюструє послідовність дій користувача під час проходження тесту знань щодо історичних подій та осіб України, що дозволяє користувачам ефективно перевіряти свої знання та вдосконалювати їх.

2.6 Інновації у вирішенні завдання

Ця кваліфікаційна робота вносить кілька нововведень у вирішення завдання з розробки застосунку для вивчення історії України.

- Інтерактивність та залучення користувача. Застосунок використовує різноманітні інтерактивні методи, такі як міні-ігри, тестування, що робить процес навчання більш захоплюючим та ефективним.
- Гнучкість у виборі матеріалу. Користувачі можуть обирати те, що їм цікаво вивчати, і переглядати інформацію в різних форматах, що дозволяє персоналізувати процес навчання та забезпечує оптимальний досвід користувача.
- Підвищення мотивації до навчання. Застосунок стимулює користувачів до активного вивчення матеріалу шляхом можливості отримання результатів тестування, участі в ігрових елементах та отримання додаткової інформації.
- Інноваційний підхід до навчання. Використання інтерактивних методів та ігрових елементів у навчальному процесі дозволяє зробити процес вивчення історії більш захоплюючим та ефективним, привертаючи увагу користувачів та сприяючи кращому засвоєнню матеріалу.
- Мультиплатформеність та доступність. Розроблений застосунок підтримується на різних платформах, що робить його доступним для широкого кола користувачів. Незалежно від того, чи використовує користувач ПК, смартфон чи планшет, він може отримати доступ до цікавого та корисного контенту для вивчення історії України.
- Зручний інтерфейс. Застосунок має інтуїтивно зрозумілий інтерфейс, що спрощує навігацію та робить вивчення історії приємним та безпроблемним процесом для користувачів будь-якого рівня навчальної підготовки.

Ці нововведення дозволяють застосунку виходити за межі стандартного підходу до вивчення історії та надають користувачам унікальний та ефективний спосіб засвоєння матеріалу.

2.7 Оцінка повноти вирішення поставленої задачі

В процесі розробки освітнього інтерактивного застосунку на основі історичної події "Битва під Жовтими Водами" було поставлено низку завдань, які необхідно було вирішити для досягнення цілей проекту. Розробка інтерактивного освітнього застосунку на платформі Ren'Py включала кілька етапів, кожен з яких був спрямований на досягнення високого рівня функціональності, інтерактивності та зручності використання.

Вибір платформи Ren'Py

Переваги Ren'Py:

- Доступність та простота. Ren'Py є безкоштовною платформою з відкритим вихідним кодом, що робить її доступною для розробників з будь-яким рівнем досвіду. Простий синтаксис дозволяє швидко розпочати розробку навіть новачкам.
- Широкі можливості для створення інтерактивних історій. Ren'Py спеціалізується на створенні візуальних новел та інтерактивних історій, що робить її ідеальним вибором для освітніх застосунків, де важливо подати історичний контент у привабливій формі.
- Підтримка мультимедійних елементів. Платформа підтримує інтеграцію різних мультимедійних форматів, таких як зображення, аудіо та відео, що дозволяє створювати багатий та різноманітний контент.

Недоліки Ren'Py:

- Обмежена функціональність для складних додатків. Хоча Ren'Py є потужним інструментом для створення візуальних новел, його можливості можуть бути обмеженими для більш складних інтерактивних застосунків.
- Залежність від сценаріїв. Велика кількість сценаріїв може ускладнити управління проектом та підвищити ризик виникнення помилок.

На початковому етапі було проведено аналіз вимог до застосунку, визначено основні цілі та завдання. Враховуючи специфіку освітнього контенту, було вирішено зосередитися на наступних аспектах:

- Подання історичних фактів у формі інтерактивної історії.
- Включення мультимедійних елементів для підвищення зацікавленості користувачів.
- Створення тестів для перевірки знань.

Було розроблено UML діаграми для візуалізації структури застосунку та взаємодії між його компонентами.

- Розділ «Ключові особи» - містить детальні біографії ключових осіб, які брали участь у битві під Жовтими Водами. Для кожної особи надаються зображення та аудіо матеріали.
- Розділ «Ключові події» - описує основні етапи битви, включаючи підготовку, хід битви та її наслідки. Цей розділ містить інтерактивні елементи, що дозволяють користувачам взаємодіяти з контентом.
- Розділ «Тестування» - система тестів, яка дозволяє користувачам перевірити свої знання після ознайомлення з матеріалом. Після завершення тестування користувач отримує зворотний зв'язок з кількістю правильних відповідей та поясненнями.

Завершальний етап включав тестування застосунку з метою виявлення та виправлення помилок, а також оптимізацію продуктивності:

- Функціональне тестування. Перевірка правильності роботи всіх функцій та взаємодії між ними.
- Тестування інтерфейсу користувача. Оцінка зручності використання та виявлення можливих проблем у навігації.
- Оптимізація. Поліпшення продуктивності та зменшення часу завантаження мультимедійних елементів.

У результаті виконаної роботи був створений інтерактивний освітній застосунок, що відповідає сучасним вимогам до навчальних програм.

Основні досягнення включають:

- Забезпечення високого рівня інтерактивності. Користувачі можуть взаємодіяти з контентом, отримувати миттєвий зворотний зв'язок та проходити тестування.

- Різноманітність навчальних матеріалів Поєднання текстових, візуальних та аудіо матеріалів дозволяє максимально ефективно передавати історичний контент.
- Зручний інтерфейс Інтуїтивно зрозумілий інтерфейс забезпечує легкість використання та доступність для широкого кола користувачів.

Таким чином, розробка інтерактивного освітнього застосунку на платформі Ren'Py дозволила створити ефективний інструмент для вивчення історичних подій, що сприяє підвищенню інтересу до історії та покращенню знань користувачів.

Створення змістовного історичного контенту є ключовим елементом розробки інтерактивного освітнього застосунку. Тут розглянуто основні аспекти та підходи до створення якісного контенту, який не тільки інформує користувачів, але й робить процес навчання цікавим та захоплюючим.

- Дослідження та аналіз історичних джерел. Виконано ретельний аналіз історичних подій, ключових осіб та контексту битви під Жовтими Водами. Всі матеріали базуються на надійних історичних джерелах.
- Наповнення контентом. Застосунок містить докладні описи подій, біографії ключових осіб, ілюстрації та інші матеріали, що допомагають користувачам краще зрозуміти історичний контекст.

Розробка системи тестування знань є важливою складовою інтерактивного освітнього застосунку. Вона дозволяє користувачам перевіряти свої знання, отримувати зворотний зв'язок та оцінювати свій прогрес у навчанні. Нижче описані основні етапи та підходи до розробки такої системи.

- Структура тестів. Впроваджено розділ «Тестування», що містить питання різного рівня складності для перевірки знань користувачів. Після завершення тестування користувач отримує зворотний зв'язок з кількістю правильних відповідей та рекомендаціями.

- Якість питань. Питання тестів розроблені таким чином, щоб охопити ключові аспекти битви під Жовтими Водами, перевірити як фактологічні знання, так і розуміння історичного контексту.

Розробка системи тестування знань в інтерактивному освітньому застосунку на платформі Ren'Py вимагає ретельного підходу до створення запитань, проектування інтерфейсу та забезпечення зворотного зв'язку для користувачів. Впровадження різноманітних типів запитань, адаптація складності та інтерактивні елементи підвищують ефективність тестування, сприяючи покращенню знань та мотивації до навчання. Зберігання та аналіз даних тестувань дозволяють постійно вдосконалювати систему та забезпечувати високу якість освітнього процесу.

Створення змістовного історичного контенту для інтерактивного освітнього застосунку на платформі Ren'Py вимагає комплексного підходу, що включає дослідження, структурування, візуалізацію та адаптацію матеріалів. Використання інтерактивних елементів, дозволяє зробити процес навчання більш захоплюючим та ефективним, забезпечуючи користувачам глибоке розуміння історичних подій. Залучення користувачів до створення контенту сприяє розвитку спільноти та підвищенню якості освітнього продукту.

Проектування та реалізація користувацького інтерфейсу є ключовими аспектами розробки інтерактивного освітнього застосунку. Зручний, інтуїтивно зрозумілий та привабливий інтерфейс значно підвищує ефективність навчання та задоволення користувачів.

- Дизайн інтерфейсу. Створено інтуїтивно зрозумілий та привабливий інтерфейс, що відповідає сучасним стандартам UX/UI дизайну. Інтерфейс забезпечує легкий доступ до всіх розділів застосунку та зручність навігації.
- Тестування інтерфейсу. Проведено тестування з залученням користувачів, що дозволило виявити та виправити можливі недоліки, оптимізувати зручність використання.

Порівняння з існуючими аналогами:

- Аналіз аналогів. Виконано порівняння з існуючими аналогічними освітніми застосунками, такими як The Expression Amrilato. Визначено сильні та слабкі сторони нашого застосунку, що дозволило виділити його переваги та можливості для подальшого вдосконалення.

Розробка користувацького інтерфейсу для інтерактивного освітнього застосунку на платформі Ren'Py дозволила створити зручний, функціональний та привабливий інструмент для вивчення історичних подій. Завдяки ретельному аналізу вимог, створенню прототипів та поетапній реалізації вдалося забезпечити високий рівень якості та задоволення користувачів.

При розробці інтерактивного освітнього застосунку на платформі Ren'Py було впроваджено кілька інноваційних та унікальних рішень, які значно підвищують його ефективність та зручність для користувачів. У цьому розділі розглянемо основні нововведення та оригінальні підходи, застосовані у проекті. Однією з головних інновацій застосунку є інтерактивний контент, який дозволяє користувачам активно взаємодіяти з навчальним матеріалом.

Використання елементів гейміфікації допомагає підвищити мотивацію користувачів та зробити навчання більш захопливим:

- Система досягнень. Користувачі отримують нагороди та досягнення за виконання різних завдань та тестів, що стимулює їх до подальшого вивчення матеріалу.
- Рівні та прогрес. Вбудована система рівнів дозволяє відслідковувати прогрес у навчанні та отримувати візуальні індикатори досягнень.

Застосунок розроблений з урахуванням модульної архітектури, що дозволяє легко додавати новий контент та функціональні можливості:

- Розширювані модулі. Кожен розділ (наприклад, «Ключові особи» або «Ключові події») реалізовано як окремий модуль, що може бути легко оновлений або розширений без змін у основному коді.

Дизайн інтерфейсу поєднує сучасні тенденції та функціональність, забезпечуючи зручність та естетичну привабливість:

- Адаптивний дизайн. Інтерфейс автоматично адаптується до різних розмірів екранів та роздільних здатностей, що забезпечує комфортне використання як на комп'ютерах, так і на мобільних пристроях.
- Інтуїтивна навігація. Логічна структура та зрозумілі навігаційні елементи дозволяють легко орієнтуватися у застосунку та швидко знаходити необхідну інформацію.

Запровадження інноваційних та унікальних рішень у розробці інтерактивного освітнього застосунку на платформі Ren'Py дозволило створити ефективний та привабливий інструмент для вивчення історичних подій. Використання інтерактивного контенту, адаптивного тестування, елементів гейміфікації та штучного інтелекту значно підвищує якість навчання та задоволення користувачів. Модульна архітектура та сучасний дизайн інтерфейсу забезпечують зручність та розширюваність застосунку, роблячи його універсальним інструментом для освітніх цілей.

Таким чином, виконані завдання дозволяють стверджувати, що поставлені цілі були досягнуті повністю. Розроблений застосунок відповідає вимогам щодо інтерактивності, змістовності та якості освітнього контенту, а також забезпечує ефективний та привабливий інтерфейс для користувачів.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОЕКТУ

3.1 Реалізація основних розділів

Я реалізував розділ «Ключові особи» у застосунку, використовуючи Ren'Py, для забезпечення користувачам можливості вибору рівня складності для кожного ключового персонажа. Цей розділ включає екран вибору рівня складності, мітки для обробки вибору та відповідні параметри гри [14].

Для кожної ключової особи створено окремий екран вибору рівня складності. Наприклад, для особи Богдана Хмельницького створено екран «bg_hml_dif»:

```
screen bg_hml_dif:

    $ renpy.block_rollback()

    add "gui/cl_os_bg_1.png"

    modal True

    zorder 100

    imagebutton auto "gui/navig/exit_%s.png" action [Hide ("bg_hml_dif"), Show ("cl_os_ch")] focus_mask True xalign 0.9 ypos 5

    hbox:

        xalign .5
        yalign .5

        if bg_hml_em == True or bg_hml_nm == True or bg_hml_hm == True:

            imagebutton auto "gui/button/bg_hml_e_%s.png" action [Hide ("bg_hml_dif"), Jump("bg_hml_easy")] focus_mask True

        else:

            imagebutton auto "gui/button/bg_hml_e_%s.png" action [Hide ("bg_hml_dif"), Jump("bg_hml_easy")] focus_mask True

        if bg_hml_nm == True or bg_hml_hm == True:

            imagebutton auto "gui/button/bg_hml_n_%s.png" action [Hide ("bg_hml_dif"), Jump("bg_hml_normal")] focus_mask True

        else:

            imagebutton auto "gui/button/bg_hml_n_%s.png" action [Hide ("bg_hml_dif"), Jump("bg_hml_normal")] focus_mask True

        if bg_hml_hm == True:

            imagebutton auto "gui/button/bg_hml_h_%s.png" action [Hide ("bg_hml_dif"), Jump("bg_hml_hard")] focus_mask True

        else:

            imagebutton auto "gui/button/bg_hml_h_%s.png" action [Hide ("bg_hml_dif"), Jump("bg_hml_hard")] focus_mask True
```

Рис. 3.1.1 – Вибір рівня складності на екрані Богдана Хмельницького

Цей екран має фон, кнопки для вибору рівня складності (легкий, нормальний, важкий) та кнопку виходу, яка повертає користувача до попереднього меню. Для кожного рівня складності створено мітки, які

налаштовують відповідні параметри гри та запускають основний ігровий процес [15]. Наприклад, для особи Богдана Хмельницького реалізовано мітки «bg_hml_easy», «bg_hml_normal» та «bg_hml_hard»:

```
label bg_hml_easy:

    scene black
    $ bg_hml_difm = 1
    $ mm_kvs_difm = 0
    $ dl_nch_difm = 0
    $ fl_djl_difm = 0
    $ tg_bei_difm = 0
    $ st_pot_difm = 0
    $ st_chr_difm = 0
    $ max_c = 2
    $ max_time = 60
    $ ww, hh = 4, 4
    show screen navig
    call memoria_game from _call_memoria_game
    return

label bg_hml_normal:

    scene black
    $ bg_hml_difm = 2
    $ mm_kvs_difm = 0
    $ dl_nch_difm = 0
    $ fl_djl_difm = 0
    $ tg_bei_difm = 0
    $ st_pot_difm = 0
    $ st_chr_difm = 0
    $ max_c = 2
    $ max_time = 120
    $ ww, hh = 6, 6
    call memoria_game from _call_memoria_game_1
    return

label bg_hml_hard:

    scene black
    $ bg_hml_difm = 3
    $ mm_kvs_difm = 0
    $ dl_nch_difm = 0
    $ fl_djl_difm = 0
    $ tg_bei_difm = 0
    $ st_pot_difm = 0
    $ st_chr_difm = 0
    $ max_c = 3
    $ max_time = 180
    $ ww, hh = 9, 9
    call memoria_game from _call_memoria_game_2
    return
```

Рис. 3.1.2 - Налаштування рівня складності

Кожна мітка встановлює змінні складності для особи, час на виконання завдання та розмір ігрового поля. Після цього викликається основна гра з заданими параметрами.

Реалізація розділу «Ключові особи» у застосунку забезпечує користувачам гнучкість у виборі рівня складності для кожної ключової особи. Це досягається через створення окремих екранів вибору складності та відповідних міток для обробки цього вибору, що налаштовують параметри гри та запускають основний ігровий процес.

Далі я реалізував розділ «Ключові події» у застосунку. Він дозволяє користувачам вибирати підрозділи для вивчення різних історичних подій. Цей екран включає функціональні елементи, такі як кнопки для вибору подій і кнопку виходу, а також забезпечує користувачам можливість переходу до відповідних розділів історії [16]. Екран містить три кнопки, які дозволяють вибирати різні підрозділи для вивчення:

```
screen cl_pod_ch:
    zorder 100
    $ renpy.block_rollback()
    modal True

    add "gui/cl_os_bg.png"

    imagebutton auto "gui/navig/exit_%s.png" action [hide ("cl_pod_ch"), Show ("choise")] focus_mask True xalign 0.99 ypos 5

    hbox spacing 100:
        xalign 0.5
        yalign 0.5

        imagebutton auto "gui/button/cl_pod1_%s.png" action [hide("cl_pod_ch"), Jump("cl_pod1")] focus_mask True

        imagebutton auto "gui/button/cl_pod2_%s.png" action [hide("cl_pod_ch"), Jump("cl_pod2")] focus_mask True

        imagebutton auto "gui/button/cl_pod3_%s.png" action [hide("cl_pod_ch"), Jump("cl_pod3")] focus_mask True
```

Рис. 3.1.3 - Вибір ключової події

Кожен підрозділ реалізований як окрема мітка, що включає діалоги, описи подій та меню для вибору варіантів дій. Наприклад, мітка для підрозділу «cl_pod1»:

```

label c1_pod1:

scene clpd_bg1 with fade

"Підготовка до битви під Жовтими Водами була ретельною і включала кілька ключових етапів, які визначили подальший успіх повстанців."

menu:

    "Як ви думаєте, яким був перший етап?"

    "Дипломатичні зусилля та союз із Кримським ханством":

        "Ні, це не так. Першим етапом був збір козаків та створення армії."

        jump c1_pod1_1

    "Збір козаків та створення армії":

        "Так цілком вірно."

        jump c1_pod1_1

    "Підготовка до оборони та нападу":

        "Ні, це не так. Першим етапом був збір козаків та створення армії."

        jump c1_pod1_1

```

Рис. 3.1.4 – Розділ «Підготовка до битви»

Кожна мітка містить діалоги та меню для вибору дій, що дозволяє користувачам взаємодіяти з історичним матеріалом. Реалізація розділу «Ключові особи» та відповідних міток у застосунку забезпечує інтерактивне навчання історії, дозволяючи користувачам не лише читати про події, але й активно брати участь у них через вибір варіантів.

Реалізація розділу «Тестування». Кожен тест включає низку питань із варіантами відповідей, правильність яких визначає фінальний результат[18]. Кожен тест складається з кількох питань, відповіді на які зберігаються в змінних, що підраховують кількість правильних відповідей [19]. Нижче наведено частковий код реалізації розділу:


```
label test_1_1:

    scene bg_tst with dissolve

    menu:

        "У якому році відбулася битва під Жовтими Водами?"

        "1638":

            $ ce_an_tst1 += 0
            jump test_1_2

        "1648":

            $ ce_an_tst1 += 1
            jump test_1_2

        "1658":

            $ ce_an_tst1 += 0
            jump test_1_2

        "1668":

            $ ce_an_tst1 += 0
            jump test_1_2
```

Рис. 3.1.5 - Приклад запитань

```
label test1_res:

    if ce_an_tst1 <= 4:

        "На жаль, Ви відповіли правильно лише на [ce_an_tst1] питання. Постарайтеся краще наступного разу."

        jump test

    if ce_an_tst1 > 4:

        "Ви відповіли правильно на [ce_an_tst1] питань. Ви молодець :)."

        jump test
```

Рис. 3.1.6 - Результат тестування

Цей код реалізує три тестування, кожне з яких містить сім питань. Відповіді зберігаються у відповідних змінних (ce_an_tst1, ce_an_tst2, ce_an_tst3). Після завершення тесту відображається результат з підрахунком правильних відповідей. Якщо кількість правильних відповідей перевищує 4,

гравець отримує похвалу, в іншому випадку - рекомендацію спробувати знову.

3.2 Реалізація карток з інформацією

Реалізація екрану з картками, дозволяє користувачу взаємодіяти з різними картками, відкриваючи інформацію залежно від рівня розблокування.

```
screen cards:

    tag menu

    add "gui/cards_bg.png"

    textbutton "Повернутися" action Return() xpos 5 yalign 0.95

    hbox:
        ypos 5

        if persistent.bg_hml_card_unl == 1:
            imagebutton auto "gui/button/bg_hml_%s.png" action [Hide("cards"), Show("bg_hml_card")] focus_mask True
        else:
            image "gui/button/bg_hml_lock.png"

        if persistent.mm_kvs_card_unl == 1:
            imagebutton auto "gui/button/mm_kvs_%s.png" action [Hide("cards"), Show("mm_kvs_card")] focus_mask True
        else:
            image "gui/button/mm_kvs_lock.png"

        if persistent.dl_nch_card_unl == 1:
            imagebutton auto "gui/button/dl_nch_%s.png" action [Hide("cards"), Show("dl_nch_card")] focus_mask True
        else:
            image "gui/button/dl_nch_lock.png"

        if persistent.fl_djl_card_unl == 1:
            imagebutton auto "gui/button/fl_djl_%s.png" action [Hide("cards"), Show("fl_djl_card")] focus_mask True
        else:
            image "gui/button/fl_djl_lock.png"
```

Рис. 3.2.1 – Екран карток

На цьому екрані є кнопка для повернення та дві горизонтальні коробки (hbox), кожна з яких містить кілька кнопок-карток. В залежності від значень в persistent, кнопка може бути розблокована (активна) або заблокована (пасивна).

```
screen st_chr_card:
    add "gui/cards_bg.png"

    textbutton "Повернутися" action [hide("st_chr_card"), Show("cards")] xpos 50 yalign 0.95

    hbox:
        ypos -100
        xalign 0.5

        if persistent.st_chr_em == True or persistent.st_chr_nm == True or persistent.st_chr_hm == True:
            image "gui/button/st_chrec_idle.png"

        if persistent.st_chr_nm == True or persistent.st_chr_hm == True:
            image "gui/button/st_chrnc_idle.png"

        if persistent.st_chr_hm == True:
            image "gui/button/st_chrhc_idle.png"

    vbox:
        xalign 0.8
        yalign 0.6
        xsize 1800
        ysize 600

        viewport id"78":
            xsize 1800
            ysize 600
            mousewheel True

            vbox spacing 80:
                if persistent.st_chr_em == True or persistent.st_chr_nm == True or persistent.st_chr_hm == True:
                    text "Стефан Чарнецький, народився 1599, помер 16 лютого 1665 – один із відомих полководців Реческої Республіки"
                else:
                    text "***Для того щоб переглянути інформацію, розблокуйте картку 1 рівня.**" font "Dihjautis-Bo"
                if persistent.st_chr_nm == True or persistent.st_chr_hm == True:
                    text "Під час Хмельниччини, Чарнецький командував польськими військами, які стояли на шляху по"

```

Рис. 3.2.2 – Картка з інформацією

На цьому екрані також є кнопка для повернення. Додається три рівні інформації, які розблоковуються послідовно залежно від значень в persistent. Інформація відображається в viewport з прокруткою. Аналогічні екрани є для кожної ключової особи [20].

Ці екрани мають аналогічну структуру. Кожен екран містить:

- Зображення фону картки.
- Кнопку для повернення до екрану cards.
- Блок зображень, що змінюються в залежності від розблокованих рівнів інформації.
- vbox для відображення тексту з інформацією про відповідну особу. Текст відображається в залежності від рівня розблокованої картки.
- viewport для прокрутки тексту з використанням коліщатка миші.

Основні елементи:

- Кнопка для повернення (textbutton «Повернутися») дозволяє гравцеві повернутися до попереднього екрану.
- Умовні конструкції (if persistent.variable == True) дозволяють динамічно змінювати вміст екрану залежно від стану збережених даних.
- Прокрутка тексту здійснюється через viewport та вертикальну смугу прокрутки (vbar).

Ця реалізація дозволяє створювати інтерактивні екрани з різними рівнями розблокованої інформації, надаючи гравцеві можливість досліджувати нові дані в залежності від їхніх досягнень або прогресу у грі.

3.3 Опис інтерфейсу користувача

Головне меню, представлено на зображенні.



Рис 3.3.1 – Головне меню застосунку

Докладний опис елементів інтерфейсу головного меню.

ПОЧАТИ - Ця кнопка дозволяє користувачу перейти до основного контенту застосунку. Натискання на неї відкриває екран вибору розділів.

ЗАВАНТАЖИТИ - Ця опція дозволяє користувачу завантажити збережену гру. Використовується для продовження навчання з місця, де користувач зупинився.

НАЛАШТУВАННЯ - Відкриває екран налаштувань, де користувач може змінити гучність звуку, швидкість авточитання, вибрати дисплей (на весь екран або у вікні) і налаштувати варіанти пропуску тексту.

ПРО ГРУ - Ця кнопка надає інформацію про гру.

КАРТКИ - Опція для перегляду карток з ключовими історичними особами. Це основний розділ для вивчення історії через інтерактивні картки.

ДОВІДКА - Цей розділ містить інструкції та поради щодо користування застосунком. Він допомагає новим користувачам швидко ознайомитися з основними функціями гри.

ВИЙТИ - Кнопка для виходу з гри. Натискання на неї завершує роботу застосунку.

Головне меню застосунку забезпечує легкий доступ до всіх основних функцій. Використання історичної ілюстрації як фону підкреслює тематику гри і створює привабливу атмосферу для користувачів, які цікавляться історією України. Інтуїтивно зрозумілий інтерфейс та чіткі описи опцій роблять навігацію простою та зручною.



Рис 3.3.2 – Меню вибору розділу

Меню поділено на три розділи.

Перший розділ веде до меню вибору ключових осіб - там можна буде вибрати особу, потім рівень складності і пройти мінігру.

Другий розділ, «Ключові події» - там можна буде вибрати ключову подію та переглянути інформацію про неї.

Третій розділ, тестування - у ньому користувач зможе пройти тести, складені на основі інформації з попередніх розділів.

Екран вибору ключових осіб у застосунку представляє користувачам можливість вибрати одну з семи історичних фігур для подальшого вивчення.



Рис 3.3.3 - Меню вибору ключових осіб

Розділ «Ключові особи» є ключовою частиною програми, яка дозволяє користувачеві поглибитися в історію України через участь у захоплюючій мінігрі «MemoGIA». Ця мінігра спрямована на розвиток пам'яті та підвищення знань про історичних постатей. Користувач обирає одну з семи ключових осіб української історії та грає у мінігру на одному із трьох рівнів складності: легко, нормально або складно. Залежно від пройденого рівня, користувач отримує різний обсяг інформації про обрану особу.

Мінігра «MemoGIA» - це інтерактивна гра, розроблена для підвищення пам'яті та засвоєння історичних фактів. Користувач має на меті зібрати якнайбільше пар однакових карток, які містять різноманітні зображення.

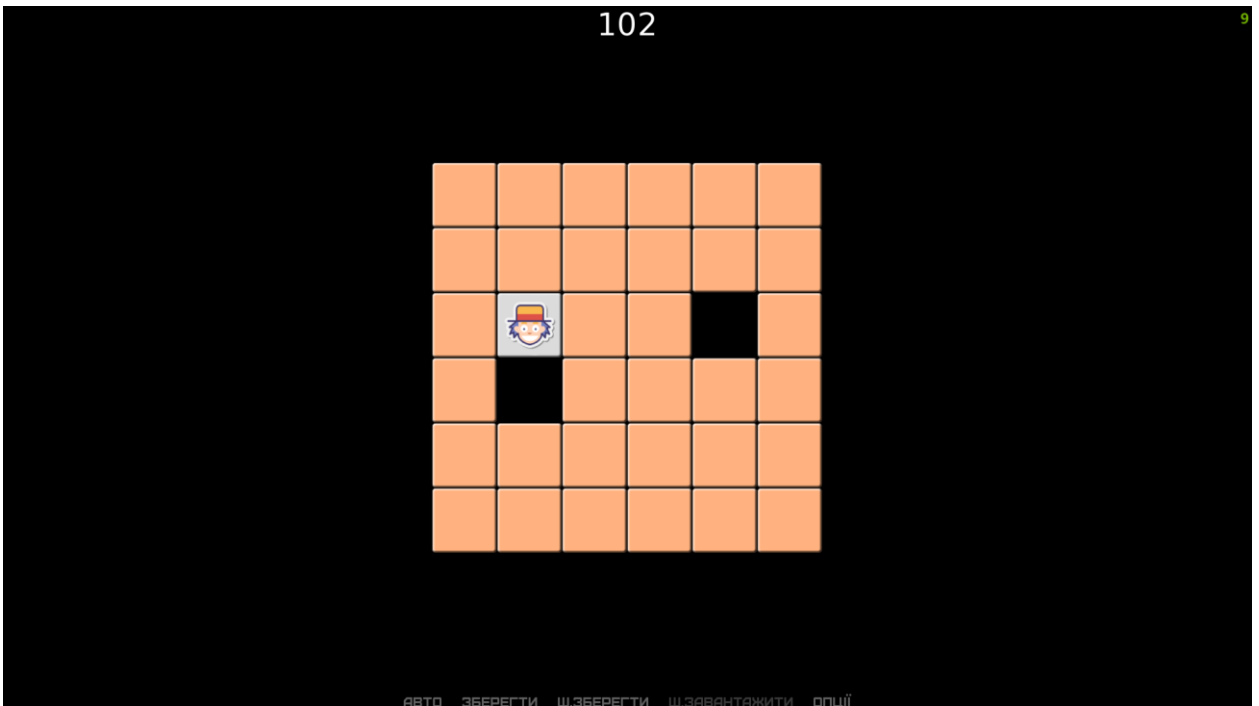


Рис 3.3.4 - Мінігра «Меморія», середня складність

Процес роботи розділу:

- Вибір особи. Користувач обирає одну з семи ключових історичних осіб України для подальшого вивчення.
- Вибір рівня складності. Користувач обирає один з трьох рівнів складності: легко, нормально або складно.
- Проходження міні-гри. Користувач переходить до ігрового поля, на якому розташовані картки з зображеннями. Він обертає дві картки за хід та намагається знайти однакові пари.

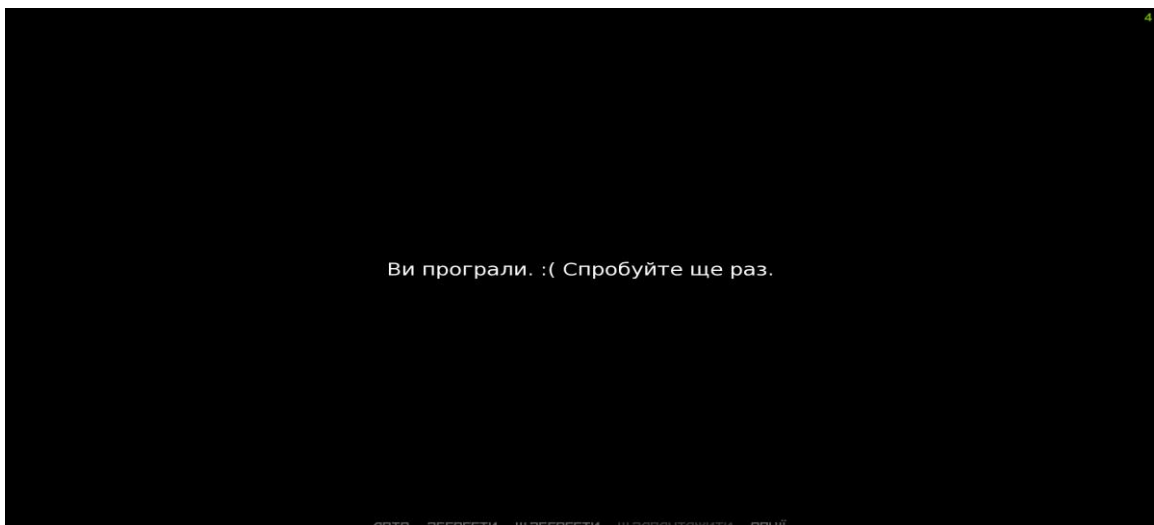


Рис 3.3.5 - Екран поразки

- Отримання інформації. Після завершення гри користувач отримує різну інформацію про обрану особу в залежності від пройденого рівня складності.

Мінігра «Меморія» пропонує три рівні складності, кожен з яких має відмінні характеристики та вимоги до користувача.

- Легкий рівень. Цей рівень призначений для початківців або тих, хто хоче швидко ознайомитися з історичною постаттю. Користувач має менше карток для знаходження пар.
- Нормальний рівень. Рівень для тих, хто вже знайомий з основною інформацією про історичну особу та бажає поглибити свої знання.
- Складний рівень. Рівень для експертів та тих, хто вже має поглиблені знання про історичну постать. Користувач має більшу кількість карток та для складання пари потрібно три картки, а не дві, що вимагає від нього швидкості та точності.

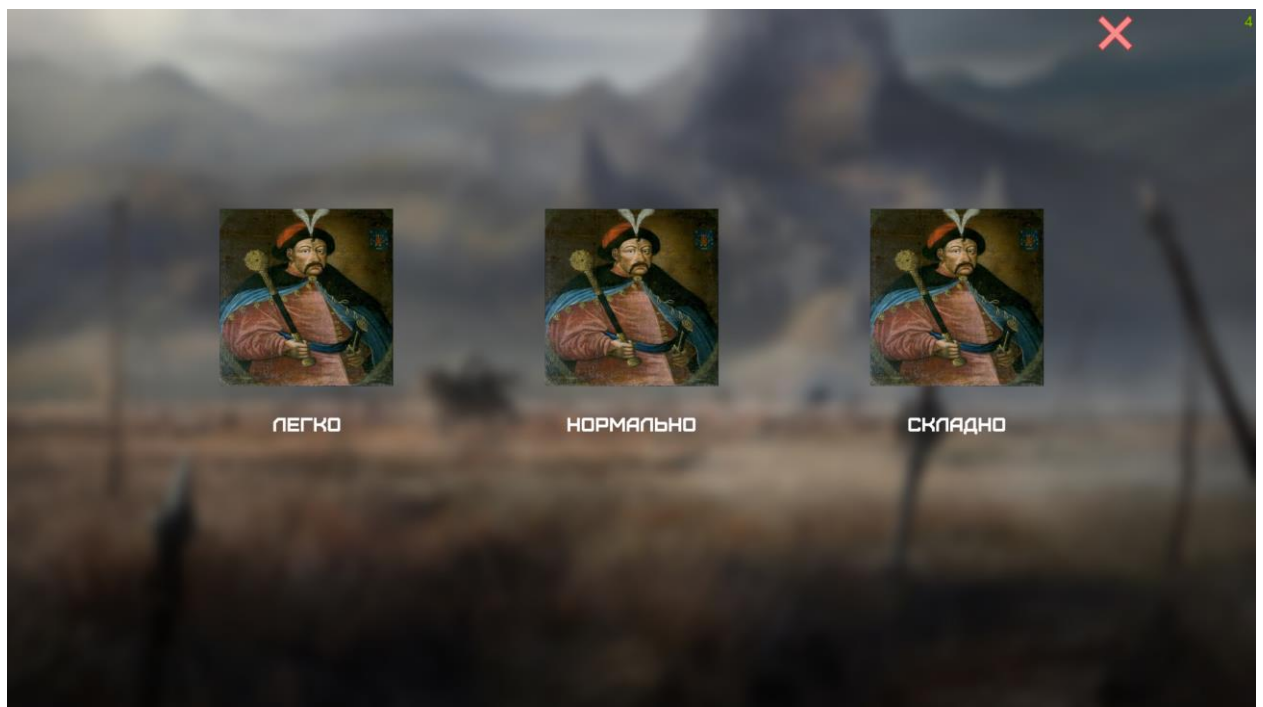


Рис. 3.3.6 - Меню вибору складності

Після завершення мінігри користувач отримує різний рівень інформації про обрану історичну постать, в залежності від пройденого рівня складності. Такий підхід дозволяє користувачеві глибше ознайомитися з обраною постаттю та засвоїти нові знання.

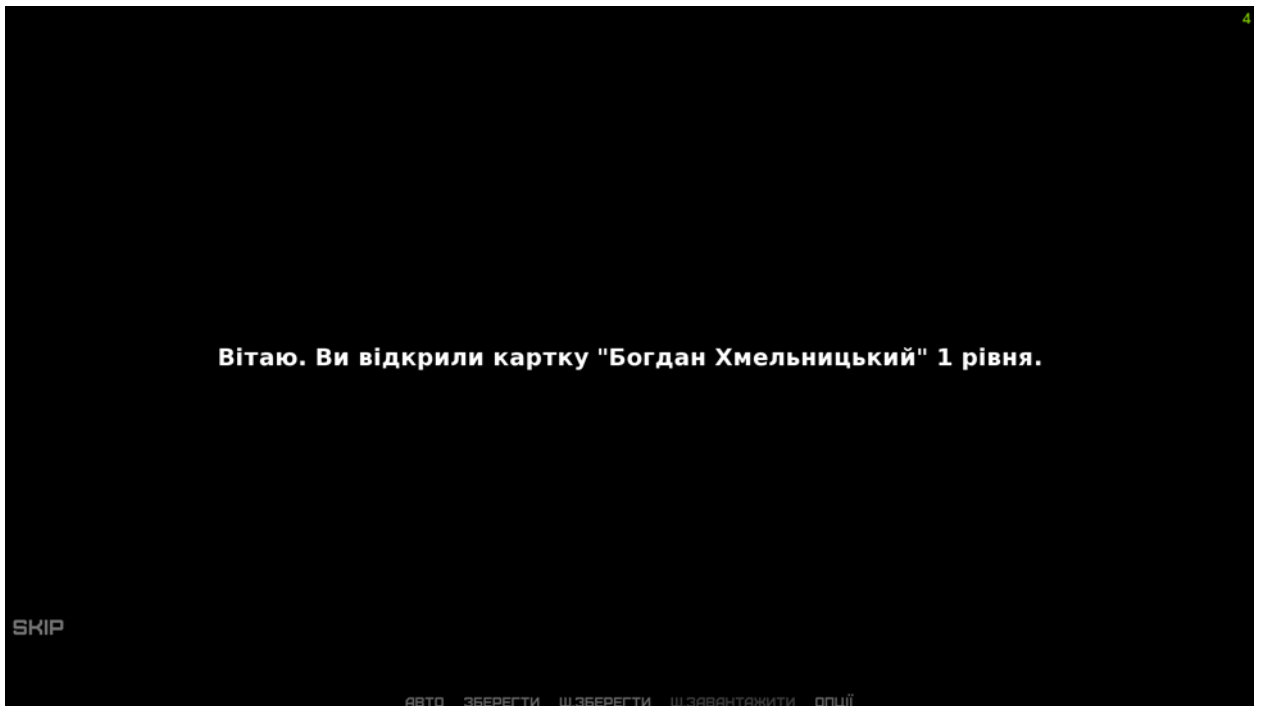


Рис 3.3.7 - Екран розблокування картки з інформацією

Після проходження одного з рівнів складності кнопка змінить зовнішній вигляд (додається трофей). Якщо пройти відразу на максимальній складності, користувач отримає доступ до всієї інформації.

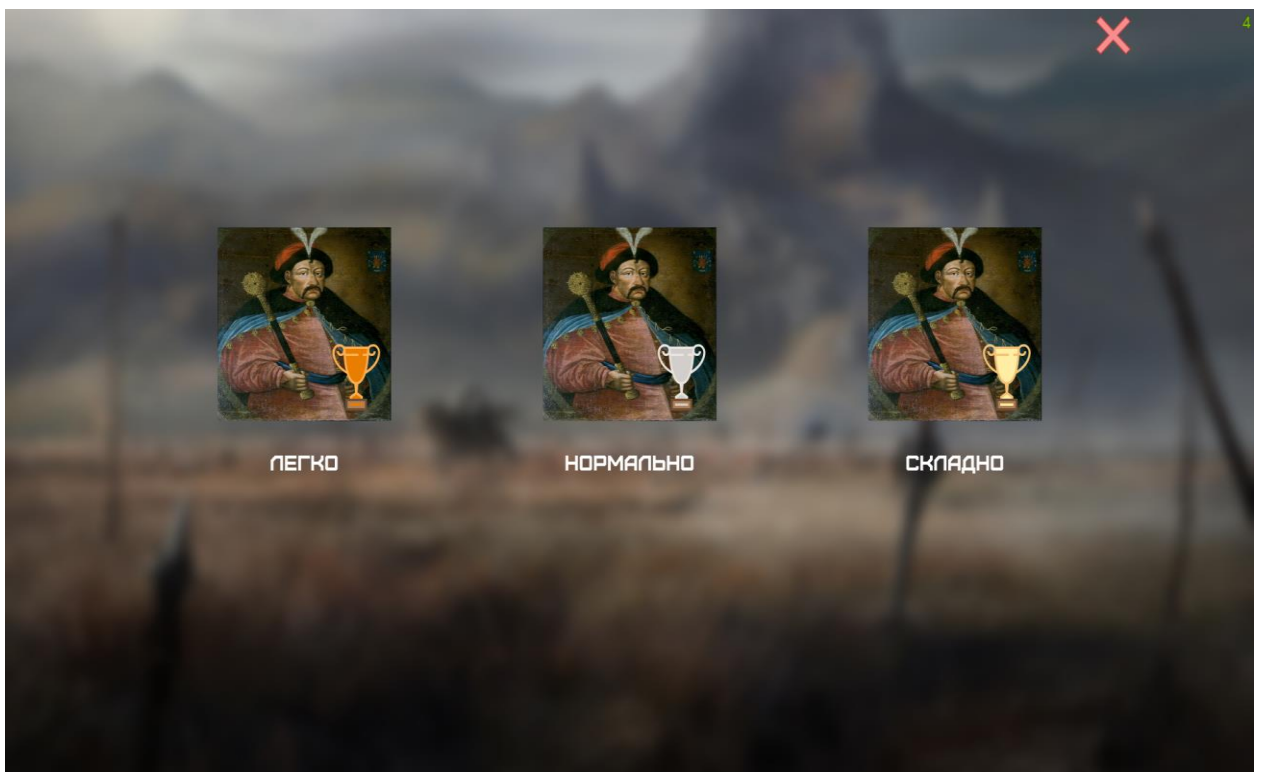


Рис.3.3.8 - Зміна кнопок після проходження міні-ігри



Рис.3.3.9 - Приклад зміни картки з інформацією (складно)

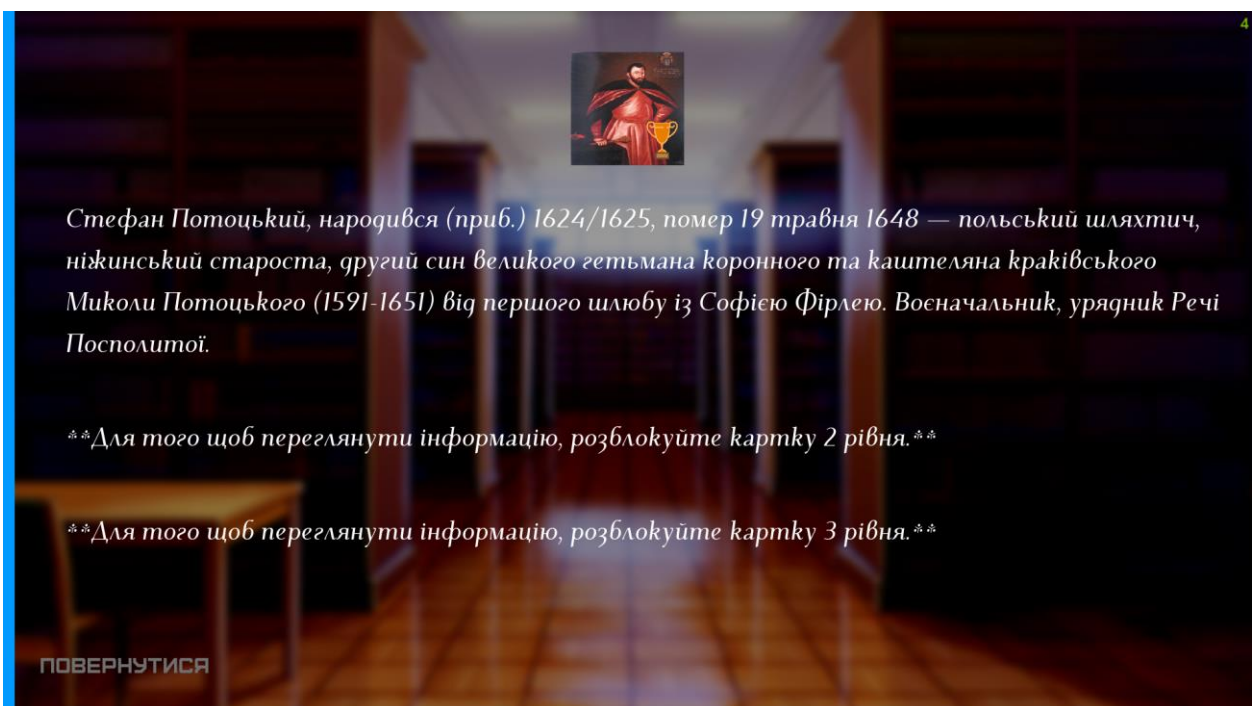


Рис.3.3.10 - Приклад зміни картки з інформацією (легко)

Цей розділ програми сприяє активному вивченню історії України шляхом захоплюючої інтерактивної мінігри. Користувачі мають можливість обирати рівень складності, що відповідає їхнім знанням та навичкам, та отримувати цікаву та корисну інформацію про історичних постатей України.

Розділ «Ключові події» є важливою частиною інтерактивного інтерфейсу користувача, призначеного для залучення користувачів до активної взаємодії з основними етапами гри. У цьому розділі користувачі мають можливість ознайомитися з важливими історичними подіями, які визначають хід гри, а також перевірити свої знання чи інтуїцію щодо цих подій шляхом відповіді на запитання. Структура інтерфейсу розділу включає три основні частини: підготовка до битви, сама битва та наслідки битви.

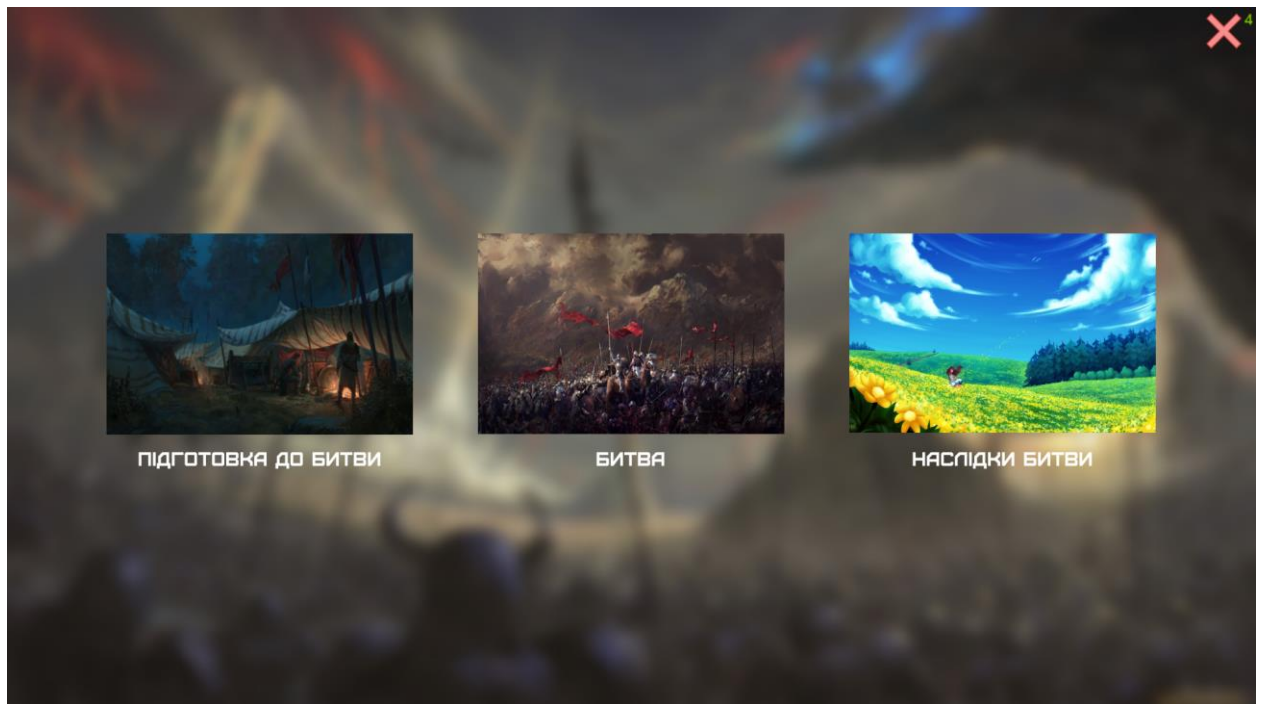


Рис.3.3.11 – Меню розділу «Ключові події»

Підготовка до битви:

- Візуальне представлення цього етапу включає зображення військового табору вночі, де відбувається активна підготовка до майбутньої битви.
- Користувачам пропонується короткий опис підготовчих заходів, таких як збір козаків, створення армії, підготовка до оборони та нападу, або дипломатичні зусилля для союзу із Кримським ханством.
- Після опису задається питання, наприклад: «Як ви думаєте, яким був перший етап підготовки до битви?».

- Користувачі вибирають відповідь із запропонованих варіантів, після чого система надає зворотний зв'язок у вигляді тексту, що підтверджує правильність чи неправильність їхньої відповіді.

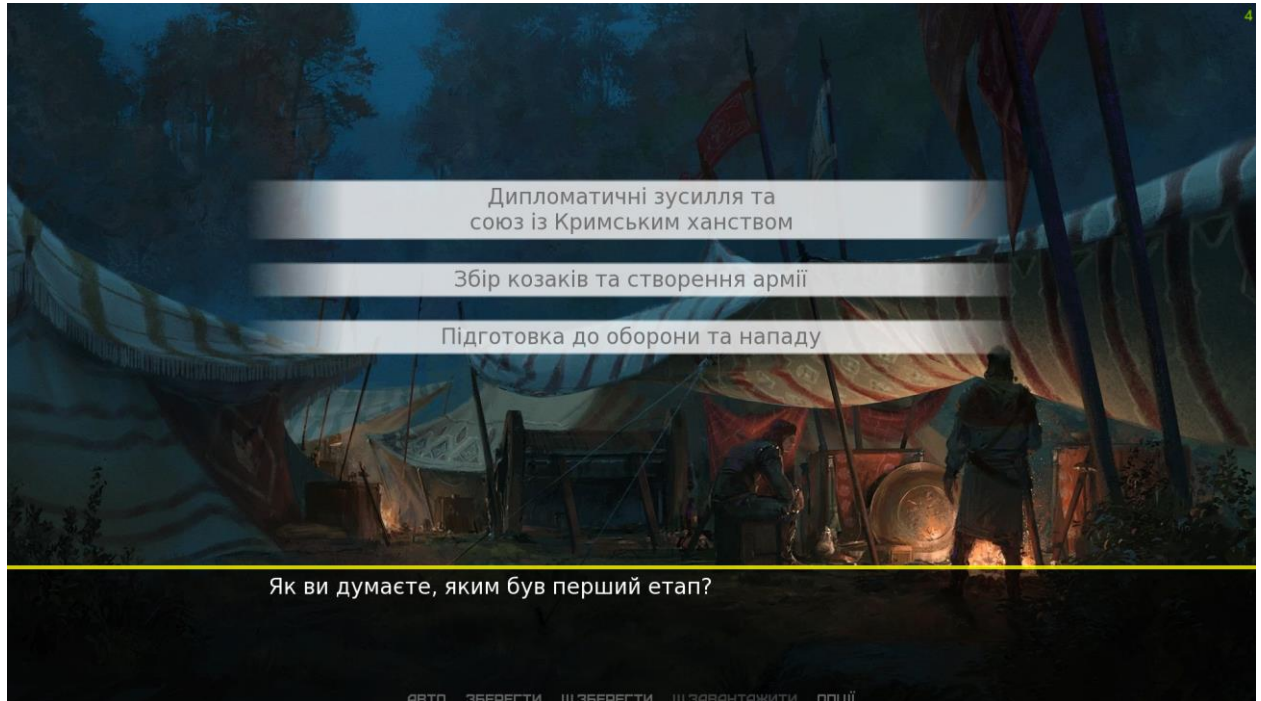


Рис.3.3.12 – Підготовка до битви

Битва:

- Зображення центральної сцени битви відображає інтенсивність та хаос бою, що створює атмосферу напруженості.
- Користувачі читають опис основних тактичних маневрів та важливих моментів битви.
- Потім задається запитання, наприклад: «На вашу думку, яку тактику використовували війська козаків під час війни?».
- Після вибору варіанту відповіді користувач отримує підтвердження чи спростування своєї відповіді.

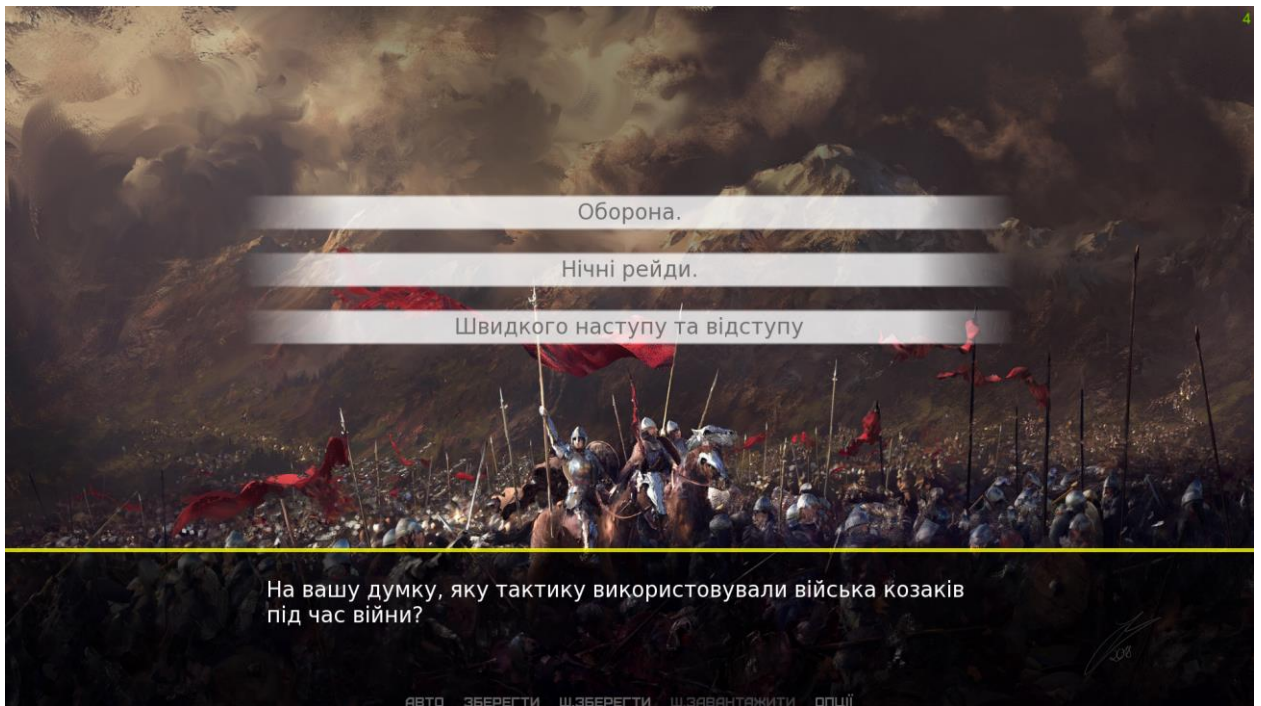


Рис.3.3.13 – Битва



Рис.3.3.14 – Наслідки битви

Наслідки битви:

- Яскраве та спокійне зображення поля після битви символізує завершення конфлікту та перехід до мирного часу.
- Користувачам надається опис наслідків битви, включаючи втрати, здобутки та зміни в політичній ситуації.
- Задається запитання, наприклад: «Як ви думаєте, після перемоги в битві, козаки могли отримати підтримку від інших національних груп?».
- Відповіді користувачів оцінюються системою, і надається відповідний зворотний зв'язок.

Інтерфейс розділу «Ключові події» розроблений для інтуїтивної та зручної взаємодії. Кожен етап представлений візуальною сценою, що допомагає користувачам зануритися в атмосферу гри. Питання, що задаються на кожному етапі, спрямовані на активізацію уваги та залучення користувачів до процесу прийняття рішень.

Опції вибору чітко окреслені та супроводжуються короткими описами, що допомагають зрозуміти можливі наслідки кожного вибору. Після кожного вибору користувачі отримують миттєвий зворотний зв'язок у вигляді текстових повідомлень, які підтверджують або спростовують їхню відповідь. Це створює відчуття безпосередньої участі у подіях та підсилює емоційне залучення до гри.

Інтерфейс розділу «Ключові події» є ключовим елементом, що забезпечує інтерактивність та занурення користувачів у сюжет гри. Завдяки продуманій структурі, візуальному оформленню та інтерактивним елементам, він дозволяє користувачам ефективно управляти подіями, перевіряти свої знання та приймати рішення, що впливають на подальший розвиток гри. Це сприяє глибшому розумінню сюжету та створює більш захоплюючий досвід.

Розділ «Тестування» є частиною інтерфейсу користувача, який дозволяє користувачам перевірити свої знання, отримані під час гри. Цей

розділ складається з меню вибору, де користувачі можуть обрати один з трьох тестів, а також містить кнопку для повернення до головного меню. Після вибору тесту користувачі проходять серію запитань, результати яких демонструють їхній рівень знань і надають зворотний зв'язок у вигляді кількості правильних відповідей.

Головний екран розділу «Тестування» представлений у вигляді меню, що містить три різні тести. Кожен тест охоплює різні аспекти застосунку, наприклад: стратегічні рішення, історичні події чи тактичні маневри.

Для кожного тесту є кнопка вибору, яка дозволяє користувачам обрати бажаний тест для проходження. Також є кнопка повернення до головного меню, яка дозволяє користувачам легко повернутися до попереднього екрану без проходження тесту.

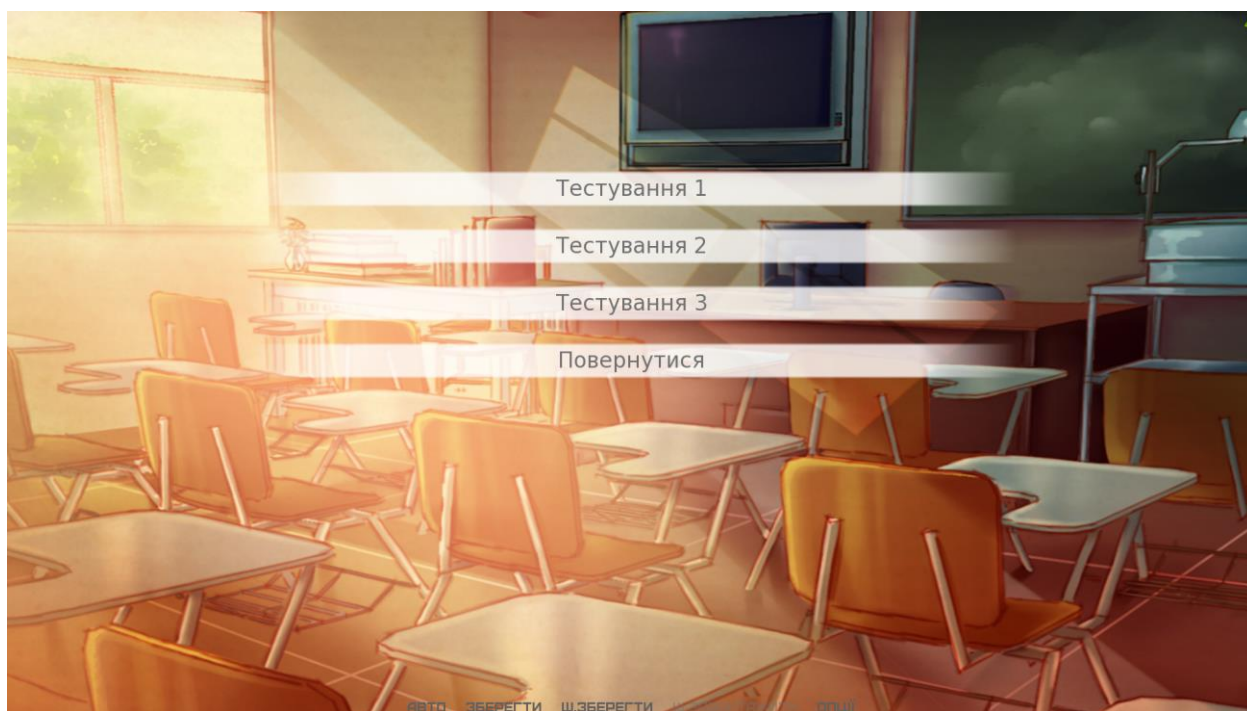


Рис 3.3.15 – Меню «Тестування»

Процес тестування:

- Після вибору тесту користувач потрапляє на екран із серією запитань. Кожне запитання супроводжується кількома варіантами відповідей.
- Користувач вибирає один із варіантів, після чого автоматично переходить до наступного запитання.

- Запитання тесту можуть бути різного типу: множинний вибір, правда/неправда, або відкрита відповідь.

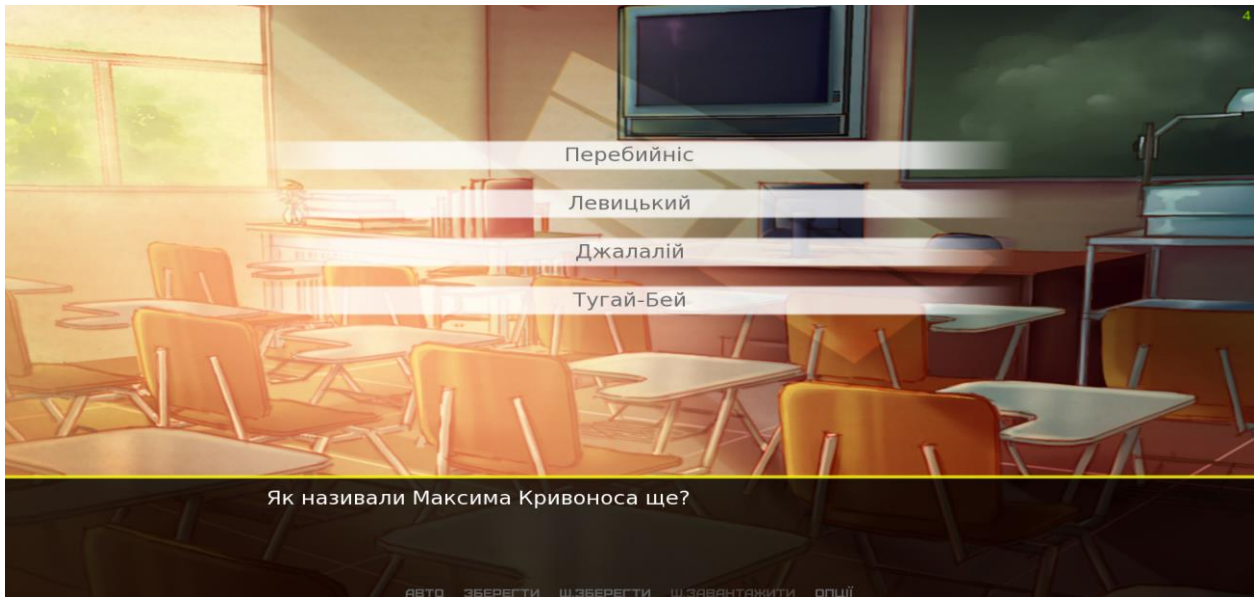


Рис 3.3.16 – Процес тестування

Результати тестування:

- Після завершення всіх запитань користувач отримує екран з результатами тесту.
- На цьому екрані відображається кількість правильних відповідей користувача.
- Залежно від кількості правильних відповідей користувач отримує різний зворотний зв'язок.

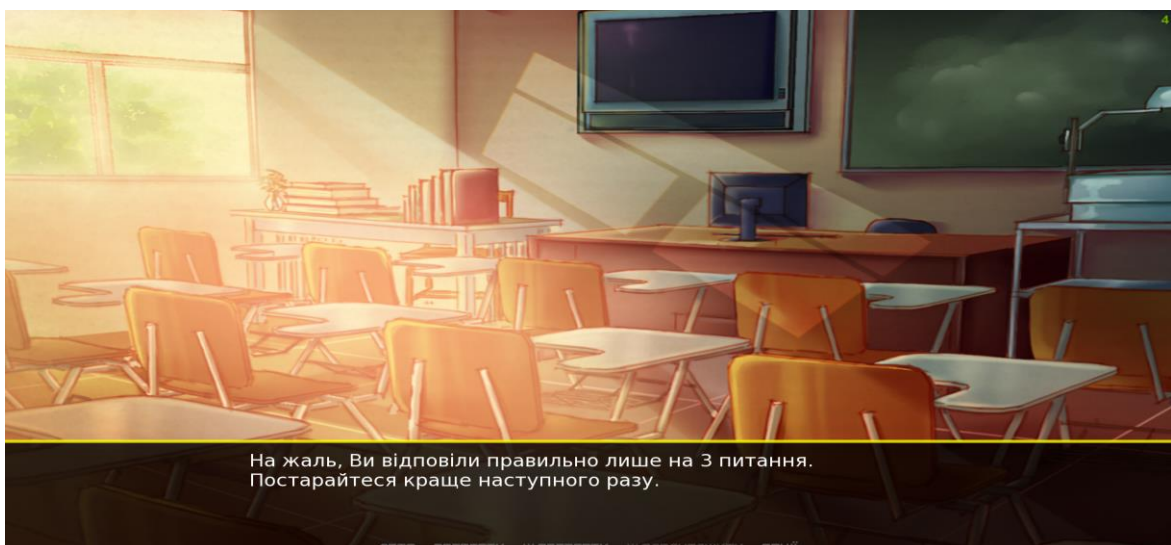


Рис 3.3.17 – Екран результатів (менше 4 правильних відповідей)



Рис 3.3.18 – Екран результатів (більше 4 правильних відповідей)

Інтерфейс розділу «Тестування» розроблений для забезпечення простої та зрозумілої взаємодії. Меню вибору тестів дозволяє користувачам швидко орієнтуватися та вибирати бажаний тест. Процес тестування побудований таким чином, щоб користувач міг легко відповідати на запитання та отримувати миттєвий зворотний зв'язок.

Під час тестування користувачі отримують чіткі інструкції та мають можливість зосередитися на кожному запитанні без зайвих відволікань. Результати тестування надаються у зрозумілому форматі, що дозволяє користувачам швидко оцінити свої знання та отримати мотивацію для подальшого вдосконалення.

Інтерфейс розділу «Тестування» є елементом інтерактивного середовища, який дозволяє користувачам перевірити та закріпити свої знання. Завдяки зручному меню вибору тестів, простому процесу тестування та зрозумілому відображенню результатів, цей розділ сприяє підвищенню інтересу користувачів до гри та їхньому залученню у навчальний процес.

ВИСНОВКИ

У цій кваліфікаційній роботі було розглянуто процес створення освітньої візуальної новели на платформі Ren'Py з акцентом на історичну подію — Битву під Жовтими Водами. У результаті виконано детальний аналіз предметної області, спроектовано та реалізовано застосунок, описано інтерфейси користувача, а також проведено порівняння з існуючими аналогами.

У першому розділі обґрунтовано актуальність створення освітніх застосунків для вивчення історичних подій, підкреслено важливість інтерактивних методів навчання. Вивчено можливості та обмеження платформи Ren'Py. Зокрема, розглянуто її переваги, такі як легкість використання та потужні інструменти для створення інтерактивних історій, і недоліки, включаючи певні технічні обмеження та складності з масштабованістю.

Описано вибір історичної події як основи для навчального контенту, що допомагає користувачам краще розуміти історичний контекст та події. Проведено аналіз аналогічних освітніх застосунків, виявлено їхні сильні та слабкі сторони. Визначено основні цілі та завдання проекту, які включають створення інтерактивного освітнього продукту з фокусом на історичну освіту.

Використано UML діаграми для детального проектування системи, що допомогло структуровано підходити до розробки та уникнути помилок у процесі реалізації. Описано процеси та взаємодії в системі, що дозволило краще зрозуміти логіку роботи застосунку. Реалізовано розділ, який описує основних історичних персоналій, залучених у битву, з можливістю інтерактивного вивчення їхніх біографій та внесків.

Описано та реалізовано ключові події, включаючи інтерактивні запитання для користувачів, що дозволяє краще запам'ятовувати історичні факти. Створено функціонал для тестування знань користувачів, що дозволяє оцінити їхнє розуміння матеріалу. Використано інноваційні підходи, зокрема

інтерактивні методи навчання, що підвищує залученість користувачів і ефективність навчання.

Розроблено інтуїтивний головний екран, що забезпечує легкий доступ до основних розділів застосунку. Створено зрозумілий та інформативний інтерфейс для розділу, присвяченого ключовим історичним особам. Реалізовано інтерфейс для вивчення та взаємодії з ключовими подіями, включаючи інтерактивні елементи. Розроблено зручний інтерфейс для проходження тестів, що дозволяє користувачам легко перевіряти свої знання.

Проведено детальне порівняння з відомою візуальною новелою «The Expression Amrilato». Виявлено переваги нашого проекту, такі як більш сучасний інтерфейс, ширший спектр навчальних матеріалів та вища адаптивність.

У ході виконання кваліфікаційної роботи було створено освітній застосунок, що поєднує в собі інтерактивні методи навчання з вивченням історичних подій. Використання платформи Ren'Py дозволило реалізувати проект з урахуванням всіх вимог та завдань. Застосунок має зручний інтерфейс, широкі можливості для інтерактивної взаємодії та є конкурентоспроможним серед існуючих аналогів. Проект успішно досяг поставлених цілей і може бути використаний як ефективний інструмент для навчання історії.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. What is Ren'Py? – [Електронний ресурс] – URL: <https://www.renpy.org>
дата звернення: (10.05.2024)
2. Why Ren'Py? [Електронний ресурс] – URL: <https://www.renpy.org/why.html> дата звернення: (10.04.2024)
3. Statement Equivalents [Електронний ресурс] – URL: https://www.renpy.org/doc/html/statement_equivalents.html дата звернення: (11.05.2024)
4. Ren'Py's documentation. [Електронний ресурс] – URL: <https://www.renpy.org/doc/html/> дата звернення: (11.04.2024)
5. Movie [Електронний ресурс] – URL: <https://www.renpy.org/doc/html/movie.html> дата звернення: (11.04.2024)
6. Битва під Жовтими Водами: перша перемога козаків у Визвольній війні [Електронний ресурс] – URL: <https://armyinform.com.ua/2021/05/16/bytva-pid-zhovtymy-vodamy-persha-peremoga-kozakiv-u-vyzvolnij-vijni/> дата звернення: (13.04.2024)
7. The Expression Amrilato [Електронний ресурс] – URL: https://store.steampowered.com/app/1044490/The_Expression_Amrilato/ дата звернення: (15.04.2024)
8. Для чого потрібні UML діаграми? [Електронний ресурс] – URL: <https://foxminded.ua/uml-diagramy/> дата звернення: (21.04.2024)
9. Діаграма послідовності (Sequence Diagrams) [Електронний ресурс] – URL: <https://www.maxzosim.com/sequence-diagrams/> дата звернення: (22.04.2024)
10. Як будувати UML-діаграми [Електронний ресурс] – URL: <https://dou.ua/forums/topic/40575/> дата звернення: (23.04.2024)
11. Моделювання станів (State Modelling) [Електронний ресурс] – URL: <https://www.maxzosim.com/state-modelling/> дата звернення: (24.04.2024)
12. Діаграми Прецедентів (Use Case UML Diagram) [Електронний ресурс] – URL: <https://lvivqaclub.blogspot.com/2008/10/use-case-uml-diagram.html> дата звернення: (24.04.2024)
13. Діаграма розгортання [Електронний ресурс] – URL: <https://studfile.net/preview/5010027/page:6/> дата звернення: (25.04.2024)
14. Screens and Screen Language [Електронний ресурс] – URL: <https://www.renpy.org/doc/html/screens.html> дата звернення: (01.05.2024)
15. Configuration Variables [Електронний ресурс] – URL: <https://www.renpy.org/doc/html/config.html> дата звернення: (03.05.2024)

16. Imagebutton [Электронный ресурс] – URL:
<https://www.renpy.org/doc/html/screens.html#imagebutton> дата
звернення:
(10.05.2024)
17. Displaying Images [Электронный ресурс] – URL:
https://www.renpy.org/doc/html/displaying_images.html дата
звернення:
(05.05.2024)
18. Text [Электронный ресурс] – URL:
<https://www.renpy.org/doc/html/text.html> дата звернення: (03.05.2024)
19. Dialogue and Narration [Электронный ресурс] – URL:
<https://www.renpy.org/doc/html/dialogue.html> дата звернення: (07.05.2024)
20. Viewport [Электронный ресурс] – URL:
<https://www.renpy.org/doc/html/screens.html#viewport> дата
звернення:
(13.05.2024)

ДОДАТОК А

Додаток А Код міні-гри «Memoria»

```
init python:
    all_cards = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I',
                'J']
    ww = 6
    hh = 6
    max_c = 3
    card_size = 48
    max_time = 25
    wait = 0.5
    img_mode = True
    values_list = []
    temp = []
    for fn in renpy.list_files():
        if fn.startswith("images/card_") and
fn.endswith(".png"):
            name = fn[12:-4]
            renpy.image("card " + name, fn)
            if name != "empty" and name != "back":
                temp.append(str(name))
    if len(temp) > 1:
        all_cards = temp
    else:
        img_mode = False
def cards_init():
    global values_list
    values_list = []
    while len(values_list) + max_c <= ww * hh:
        current_card = renpy.random.choice(all_cards)
        for i in range(0, max_c):
            values_list.append(current_card)
    renpy.random.shuffle(values_list)
    while len(values_list) < ww * hh:
```



```

        values_list.append('empty')

screen memo_scr:

    timer 1.0 action If (memo_timer > 1,
SetVariable("memo_timer", memo_timer - 1),
Jump("memo_game_lose") ) repeat True

    grid ww hh:

        align (.5, .5)

        for card in cards_list:

            button:

                left_padding 0
                right_padding 0
                top_padding 0
                bottom_padding 0
                background None

                if card["c_value"] == 'empty':

                    if img_mode:

                        add "card empty"

                    else:

                        text " " size card_size

                else:

                    if card["c_chosen"]:

                        if img_mode:

                            add "card " + card["c_value"]

                        else:

                            text card["c_value"] size card_size

                    else:

                        if img_mode:

                            add "card back"

                        else:

                            text "#" size card_size

                action If ( (card["c_chosen"] or not can_click),
None, [SetDict(cards_list[card["c_number"]], "c_chosen", True),
Return(card["c_number"]) ] )

```

```

text str(memo_timer) xalign .5 yalign 0.0 size card_size
label memoria_game:
    $ cards_init()
    $ cards_list = []
    python:
        for i in range (0, len(values_list) ):
            if values_list[i] == 'empty':
                cards_list.append ( {"c_number":i, "c_value":
values_list[i], "c_chosen":True} )
            else:
                cards_list.append ( {"c_number":i, "c_value":
values_list[i], "c_chosen":False} )
    $ memo_timer = max_time
    show screen memo_scr
    label memo_game_loop:
        $ can_click = True
        $ turned_cards_numbers = []
        $ turned_cards_values = []
        $ turns_left = max_c
        label turns_loop:
            if turns_left > 0:
                $ result = ui.interact()
                $ memo_timer = memo_timer
                $ turned_cards_numbers.append
(cards_list[result]["c_number"])
                $ turned_cards_values.append
(cards_list[result]["c_value"])
                $ turns_left -= 1
                jump turns_loop
        # предотвращаем открытие лишних карточек
        $ can_click = False
        if turned_cards_values.count(turned_cards_values[0]) !=
len(turned_cards_values):
            $ renpy.pause (wait, hard = True)

```

```

python:
    for i in range (0, len(turned_cards_numbers) ):
cards_list[turned_cards_numbers[i]]["c_chosen"] =
False
else:
    $ renpy.pause (wait, hard = True)
python:
    for i in range (0, len(turned_cards_numbers) ):
cards_list[turned_cards_numbers[i]]["c_value"] = 'empty'
    for j in cards_list:
        if j["c_chosen"] == False:
            renpy.jump ("memo_game_loop")
        renpy.jump ("memo_game_win")
    jump memo_game_loop
label memo_game_lose:
    hide screen memo_scr
    $ renpy.pause (0.1, hard = True)
    centered "{size=36}Ви програли. :( Спробуйте ще раз.{/size}"
    show screen cl_os_ch
    ""
label memo_game_win:
    if bg_hml_difm == 1:
        $ bg_hml_em = True
        $ bg_hml_emu += 1
        $ persistent.bg_hml_em = True
        $ persistent.bg_hml_card_unl = 1
        hide screen memo_scr
        $ renpy.pause (0.1, hard = True)
        if bg_hml_emu >= 2 or bg_hml_nmu >= 1 or bg_hml_hmu >=
1:
            centered "{size=36}{b}Вітаю. Але картку ви вже
відкрили, так що ви просто розумничка :3{/b}{/size}"

```

```

else:
    centered "{size=36}{b}Вітаю. Ви відкрили картку
\"Богдан Хмельницький\" 1 рівня.{/b}{/size}"
    show screen bg_hml_dif
    ""

if bg_hml_difm == 2:
    $ bg_hml_nm = True
    $ bg_hml_nmu += 1
    $ bg_hml_emu += 1
    $ persistent.bg_hml_nm = True
    $ persistent.bg_hml_card_unl = 1
    hide screen memo_scr
    $ renpy.pause (0.1, hard = True)
    if bg_hml_nmu >= 2 or bg_hml_hmu >= 1:
        centered "{size=36}{b}Вітаю. Але картку ви вже
відкрили, так що ви просто розумничка :3{/b}{/size}"
    else:
        centered "{size=36}{b}Вітаю. Ви відкрили картку
\"Богдан Хмельницький\" 2 рівня.{/b}{/size}"
        show screen bg_hml_dif
        ""

if bg_hml_difm == 3:
    $ bg_hml_hm = True
    $ bg_hml_hmu += 1
    $ bg_hml_nmu += 1
    $ bg_hml_emu += 1
    $ persistent.bg_hml_hm = True
    $ persistent.bg_hml_card_unl = 1
    hide screen memo_scr
    $ renpy.pause (0.1, hard = True)
    if bg_hml_hmu >= 2:
        centered "{size=36}{b}Вітаю. Але картку ви вже
відкрили, так що ви просто розумничка :3{/b}{/size}"

```

```
else:
    centered "{size=36}{b}Вітаю. Ви відкрили картку
\"Богдан Хмельницький\" 3 рівня.{/b}{/size}"
    show screen bg_hml_dif
    ""
```

ДОДАТОК Б

Додаток Б. Код екрану карток та можливості відкриття інформації

screen cards:

```
tag menu
add "gui/cards_bg.png"
textbutton "Повернутися" action Return() xpos 5 yalign 0.95
hbox:
    ypos 5
    if persistent.bg_hml_card_unl == 1:
        imagebutton auto "gui/button/bg_hml_%s.png" action
[Hide("cards"), Show("bg_hml_card")] focus_mask True
    else:
        image "gui/button/bg_hml_lock.png"
    if persistent.mm_kvs_card_unl == 1:
        imagebutton auto "gui/button/mm_kvs_%s.png" action
[Hide("cards"), Show("mm_kvs_card")] focus_mask True
    else:
        image "gui/button/mm_kvs_lock.png"
    if persistent.dl_nch_card_unl == 1:
        imagebutton auto "gui/button/dl_nch_%s.png" action
[Hide("cards"), Show("dl_nch_card")] focus_mask True
    else:
        image "gui/button/dl_nch_lock.png"
    if persistent.fl_djl_card_unl == 1:
        imagebutton auto "gui/button/fl_djl_%s.png" action
[Hide("cards"), Show("fl_djl_card")] focus_mask True
    else:
        image "gui/button/fl_djl_lock.png"
hbox:
    yalign 0.9
    if persistent.tg_bei_card_unl == 1:
        imagebutton auto "gui/button/tg_bei_%s.png" action
[Hide("cards"), Show("tg_bei_card")] focus_mask True
    else:
        image "gui/button/tg_bei_lock.png"
```

```

        if persistent.st_pot_card_unl == 1:
            imagebutton auto "gui/button/st_pot_%s.png" action
[Hide("cards"), Show("st_pot_card")] focus_mask True
        else:
            image "gui/button/st_pot_lock.png"
            if persistent.st_chr_card_unl == 1:
                imagebutton auto "gui/button/st_chr_%s.png" action
[Hide("cards"), Show("st_chr_card")] focus_mask True
            else:
                image "gui/button/st_chr_lock.png"
screen bg_hml_card:
    add "gui/cards_bg.png"
    textbutton "Повернуться" action [Hide("bg_hml_card"),
Show("cards")] xpos 50 yalign 0.95
    hbox:
        ypos -100
        xalign 0.5
        if persistent.bg_hml_em == True or persistent.bg_hml_nm
== True or persistent.bg_hml_hm == True:
            image "gui/button/bg_hmlec_idle.png"
        if persistent.bg_hml_nm == True or persistent.bg_hml_hm
== True:
            image"gui/button/bg_hmlnc_idle.png"
        if persistent.bg_hml_hm == True:
            image"gui/button/bg_hmlhc_idle.png"
    vbox:
        xalign 0.8
        yalign 0.6
        xsize 1800
        ysize 600
        viewport id"78":
            xsize 1800

```



```
ysize 600
mousewheel True
vbox spacing 80:
    if persistent.bg_hml_em == True or
persistent.bg_hml_nm == True or persistent.bg_hml_hm == True:
        text "" font "DihjautiS-BoldItalic.otf" size
40
    else:
        text "***Для того щоб переглянути інформацію,
розблокуйте картку 1 рівня.**" font "DihjautiS-BoldItalic.otf"
size 40
        if persistent.bg_hml_nm == True or
persistent.bg_hml_hm == True:
            text "" font "DihjautiS-BoldItalic.otf" size
40
        else:
            text "***Для того щоб переглянути інформацію,
розблокуйте картку 2 рівня.**" font "DihjautiS-BoldItalic.otf"
size 40
            if persistent.bg_hml_hm == True:
                text "" font "DihjautiS-BoldItalic.otf" size
40
            else:
                text "***Для того щоб переглянути інформацію,
розблокуйте картку 3 рівня.**" font "DihjautiS-BoldItalic.otf"
size 40
vbar value YScrollValue ("78")
```