

Міністерство освіти і науки України
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра економічної кібернетики та інформатики

Буднік Тарас Анатолійович

«Гейміфіковані елементи навчання для посилення мотивації студентів»

спеціальність 124 Системний аналіз

освітньо-професійна (наукова) програма Системний аналіз

випускна кваліфікаційна робота за освітнім ступенем «магістр»

Виконав студент

групи Сам-21

Буднік Т.А.

підпис

Науковий керівник:

д.т.н., професор

Пасічник Р.М.

підпис

Тернопіль – 2024

ЗМІСТ

ВСТУП.....	3
РОЗДІЛ 1. АНАЛІЗ ПРИДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ.....	5
1.1 Теоретичні основи гейміфікації та їх інтеграція в освіту	5
1.2 Аналіз існуючих додатків для навчання.....	9
1.3 Психологічні основи	14
1.4 Переваги та недоліки в освіті.....	15
РОЗДІЛ 2. ПІДХІД В РОЗРОБЦІ ГЕЙМІФІКОВАНОЇ СИСТЕМИ НАВЧАННЯ	18
2.1 Вибір основних підходів та методів гейміфікації.....	18
2.2 Дизайн гейміфікованих завдань	20
2.3. Інструменти та платформи для гейміфікації	22
2.4 Опис використовуваних засобів розробки	27
РОЗДІЛ 3. ТЕХНІЧНА РЕАЛІЗАЦІЯ	32
3.1. Архітектура і структура додатку	32
3.2. Реалізація ключового функціоналу	35
3.3. Тестування додатку.....	38
ВИСНОВКИ.....	42
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	42
ДОДАТКИ.....	43

ВСТУП

Актуальність теми. Елементи ігрового навчання є актуальною темою, оскільки вони можуть значно підвищити мотивацію студентів. Дослідження показали, що інтеграція ігрових елементів у навчальний процес покращує залучення студентів, мотивацію та результати навчання.

Вплив на мотивацію та залучення:

- Гейміфікація може підвищити мотивацію студентів, використовуючи ігрові механіки, такі як виклики, нагороди та самопізнання, щоб стимулювати інтерес та участь у навчанні.

Використання ігрових платформ у навчанні позитивно вплинуло на мотивацію, що вплинуло на задоволеність поведінкою та успішність у навчанні.

Ігрові елементи навчання є ефективним інструментом для підвищення мотивації та залучення студентів і можуть мати позитивний вплив на успішність. Однак для досягнення максимального ефекту необхідно враховувати якість діяльності та адаптувати підхід до мотивації різних типів студентів.

Мета цієї теми - покращити результати навчання. Ігрові предмети можуть допомогти покращити результати навчання, оскільки студенти беруть участь у процесі навчання та готові до досягнення своїх цілей.

Розвиток внутрішньої мотивації: Гейміфікація може сприяти розвитку внутрішньої мотивації, надаючи студентам можливість самостійно вибирати завдання і підвищуючи їх самооцінку.

Надання зворотного зв'язку: ігрові системи часто включають автоматизовані механізми оцінювання, які дозволяють студентам отримувати швидкий і точний зворотний зв'язок. Це важливо для навчання.

Об'єкт дослідження - є процес навчання з використанням гейміфікованих елементів та його вплив на мотивацію студентів.

Предмет дослідження – є вплив гейміфікованих елементів на мотивацію студентів в навчальному процесі.

Методи дослідження базуються на розробці програми, яка в майбутньому дозволить збільшити ефективність навчання студентів.

Наукова новизна дослідження полягає у розробці методів формування навичок об'єктно орієнтованого програмування та формування динамічного контенту веб-сторінок за допомогою елементів гейміфікації у відкритій навчальній системі.

Практична цінність дослідження полягає в тому, що його результати можуть бути використані для:

- Підвищення ефективності навчального процесу;
- Створення мотивуючого навчального середовища.

РОЗДІЛ 1. АНАЛІЗ ПРИДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ

1.1 Теоретичні основи гейміфікації та їх інтеграція в освіту

Гейміфікація в освіті – це застосування елементів гри, таких як бали, призи та завдання, щоб зробити навчання більш цікавим. Вона допомагає краще засвоювати матеріал, бо поєднує навчання із задоволенням. Це дозволяє учасникам відчувати досягнення та прогрес, який вони отримали під час проходження.

Людям, що люблять кидати собі виклик, який буде здаватися їм досяжним, може сподобатись проходження гри, де вони отримають знання в поєднанні з цікавим процесом. В певній мірі це можна назвати основою гейміфікації. Крім того, інтерактивні елементи дозволять учаснику глибше зануритись в матеріал, відчуваючи себе частиною процесу.

Важливо зазначити, що гейміфікація не лише розважає, але й навчає. Вона сприяє розвитку навичок вирішення проблем. Проте існують і труднощі: якщо занадто зосередитися на винагородах, можна втратити справжню суть навчання. Тому гейміфікацію слід використовувати з розумом, щоб вона не замінювала основний навчальний процес.

Розглянемо важливі елементи гейміфікації, такі як:



Рис.1.1.1. Елементи ігрової механіки

Ігрова механіка — це основа гейміфікації, яка дозволяє залучати і мотивувати учасників. Вона складається з різноманітних елементів, кожен з яких виконує свою функцію в побудові цікавої і ефективної освітньої системи. Наприклад, бали, які є однією з найпростіших форм винагороди. Вони мотивують учасників виконувати завдання, досягати цілей і знаходити правильні рішення. Це стимулює бажання докладати більше зусиль, адже кожне досягнення дає відчуття прогресу.

Значки, у свою чергу, виступають символами досягнень і прогресу. Це віртуальні нагороди, які студенти можуть отримати за виконання певних умов чи завдань.

Ще одним важливим елементом є **лідерборди**, які додають елемент змагання. Вони заохочують учасників прагнути досягти більшого, щоб покращити свою позицію в рейтингу.

Механіка рівнів дозволяє поступово підвищувати складність завдань і відкривати нові можливості, що робить процес більш захоплюючим.

Завдання і квести — це ще одна ключова складова ігрової механіки. Завдання можуть бути різної складності й вимагати від учасників креативності та зусиль. Квести, як правило, є серією пов'язаних між собою завдань, об'єднаних загальним сюжетом. Вони дозволяють глибше залучати школярів чи студентів, оскільки вони занурюються в історію, яка розгортається перед ними, і стають її активними учасниками.

Віртуальна валюта додає ще один рівень взаємодії. Учасники можуть "заробляти" її за досягнення та витратити на бонуси або привілеї, також це може бути реалізовано як відсоток від загального балу за проходження. Це створює відчуття контролю над процесом і робить навчання більш персоналізованим. Аналогічно, несподівані подарунки тримають інтерес на високому рівні, адже учасники відчують елемент сюрпризу, що посилює емоційну залученість.

Не менш важливими є **сюжет, персонажі та загальна атмосфера гри**. Ці елементи додають естетичну привабливість і допомагають створити цілісний ігровий світ, в який учасники можуть поринути. Добре продуманий сюжет пов'язує всі механіки.

Принципи ігрового дизайну спрямовані на те, щоб зробити цей досвід позитивним і цікавим. Завдання повинні бути цікавими та відповідати як інтересам учасників, так і навчальній програмі, а зворотний зв'язок — надаватися регулярно, щоб підтримувати мотивацію. Грамотно поєднані конкуренція та співпраця допомагають зберігати баланс між викликами та взаємодією. Усе це разом забезпечує прогрес і створює в учасників відчуття досягнення, яке є ключовим у будь-якій навчальній або ігровій системі.

Історія гейміфікації в освіті.

Ідея використання ігор для навчання має давню історію. Ще в стародавні часи вчителі застосовували ігрові підходи, щоб зробити процес навчання більш цікавим і доступним для дітей. Настільні ігри, загадки та інші форми розваг використовувалися для розвитку мислення, пам'яті та логіки. Однак сучасна концепція гейміфікації, яка включає інтерактивні цифрові інструменти, виникла завдяки стрімкому розвитку технологій.

Ранні приклади гейміфікації в освіті можна побачити в освітніх іграх, симуляторах і рольових іграх, які використовувалися ще до появи самого терміна "гейміфікація". Наприклад, симулятори допомагали студентам медичних і технічних спеціальностей безпечно практикувати складні навички, а рольові ігри сприяли розвитку соціальних і комунікаційних здібностей. Це був перший крок до інтеграції ігрових елементів у навчальний процес.

Справжній прорив у гейміфікації став можливим завдяки розвитку цифрових технологій. Комп'ютери, інтернет, а згодом і мобільні пристрої дали змогу створювати інтерактивне середовище навчання, яке стало доступним для широкої аудиторії. Освітні програми й платформи почали включати такі елементи, як віртуальні нагороди, завдання з інтерактивними сценаріями та адаптивне навчання. Мобільні додатки і освітні ігри стали популярним інструментом для учнів і студентів, дозволяючи їм навчатися будь-де та будь-коли.

Сучасні тенденції у гейміфікації включають використання віртуальної та доповненої реальності. Ці технології дозволяють учасникам глибше зануритися в навчальний процес, створюючи реалістичні симуляції та інтерактивні

середовища. Онлайн-курси та платформи також активно інтегрують гейміфікацію, пропонуючи системи балів, значків, квестів і рейтингових таблиць.

Важливо зазначити, що гейміфікація — це не гра в прямому сенсі цього слова. Вона використовує окремі елементи ігрового процесу, такі як правила, завдання, конкуренція й винагороди, щоб зробити освітній процес більш цікавим і мотивуючим. Однак на відміну від повноцінної гри, гейміфікація завжди має чітку освітню мету. Це інструмент, який доповнює традиційне навчання, допомагаючи здобувачам освіти засвоювати матеріал ефективніше, а викладачам — залучати аудиторію.

Гейміфікація в освіті швидко набирає популярності в усьому світі, і багато країн активно впроваджують її в навчальних закладах. У США, наприклад, платформи як Khan Academy використовують гейміфікацію для підвищення мотивації студентів. Вони надають бали, бейджі та інші нагороди за виконання завдань, що заохочує учнів досягати нових висот. Ще одним прикладом є ClassDojo, яка використовує гейміфікацію для заохочення позитивної поведінки та досягнень учнів, залучаючи також батьків у процес навчання. Duolingo — популярний додаток для вивчення мов, також активно застосовує ігрові механіки, що мотивують користувачів досягати нових рівнів і вивчати мови швидше.

В Європі також активно використовують гейміфікацію. Фінляндія, відома своєю інноваційною системою освіти, експериментує з гейміфікацією в школах, інтегруючи її в навчальний процес для підвищення залученості учнів. У Естонії розроблено національну програму з впровадження цифрових технологій в освіту, яка включає елементи гейміфікації, допомагаючи учням засвоювати матеріал через інтерактивні та гейміфіковані методи. У Нідерландах університети та професійні школи активно експериментують з гейміфікацією, включаючи її в навчальні курси для залучення студентів.

В країнах Азії, таких як Сінгапур та Південна Корея, активно інвестують у розвиток освітніх технологій, включаючи гейміфікацію. Вони використовують гейміфіковані платформи та додатки для підвищення мотивації учнів та

інтерактивного навчання. Японія також включає гейміфікацію в свої навчальні методики, зокрема в школах та університетах, щоб залучити студентів та полегшити процес навчання.

У інших країнах, таких як Австралія та Канада, гейміфікація в освіті також активно впроваджується. Канадські та австралійські школи та університети використовують гейміфіковані підходи для поліпшення освітнього досвіду і залучення студентів.

Важливо зазначити, що гейміфікація в освіті розвивається по-різному в кожній країні. В деяких вона впроваджена як частина окремих уроків, в інших — як частина більш складних навчальних систем. Ефективність цього підходу значною мірою залежить від правильної адаптації гейміфікації до потреб студентів, вибору відповідних ігрових механік і кваліфікації викладачів, що використовують ці методи.

1.2 Аналіз існуючих додатків для навчання

Приклади гейміфікації в освіті:

Навчальні платформи: **Khan Academy**, **Duolingo** та **Coursera** використовують бали, значки та таблиці лідерів для мотивації студентів.

«Академія Хана» (англ. Khan Academy) — це некомерційна освітня організація, створена в 2006 році педагогом Салманом Ханом для забезпечення «високоякісної освіти для будь-кого і будь-де». Організація створює лекції у формі YouTube-відео. Окрім мікролекцій, на вебсторінці організації розміщені практичні заняття та методичні матеріали для вчителів. Усі ресурси доступні безкоштовно для кожного в усьому світі.



Рис.1.2. Освітня організація «Академія Хана»

Duolingo — електронна платформа вивчення мови і краудсорсингового перекладу тексту, що має платний та безкоштовний види акаунтів. Сервіс розроблений так, що під час уроків Студенти допомагають перекладати вебсайти, статті та інші документи. В цілому, Duolingo - це ефективний та цікавий інструмент для вивчення мов, який може бути корисним для багатьох людей. Однак, важливо пам'ятати, що це лише один з багатьох інструментів, і його слід використовувати в поєднанні з іншими методами навчання.



Рис.1.3. Електронна платформа для вивчення іноземних мов Duolingo

Coursera - це масова відкрита онлайн-платформа (МООС), яка пропонує тисячі курсів від провідних університетів та організацій світу. Заснована в 2012 році професорами Стенфордського університету Ендрю Нгом та Дафною Коллер, Coursera стала однією з найпопулярніших платформ онлайн-освіти.



Рис.1.4. Онлайн-платформа Coursera

CodeCombat - це захоплююча комерційна гра, яка навчає програмуванню в інтерактивному та веселому форматі. Замість сухого вивчення синтаксису, ви пишете справжній код, щоб керувати своїм героєм та проходити рівні.

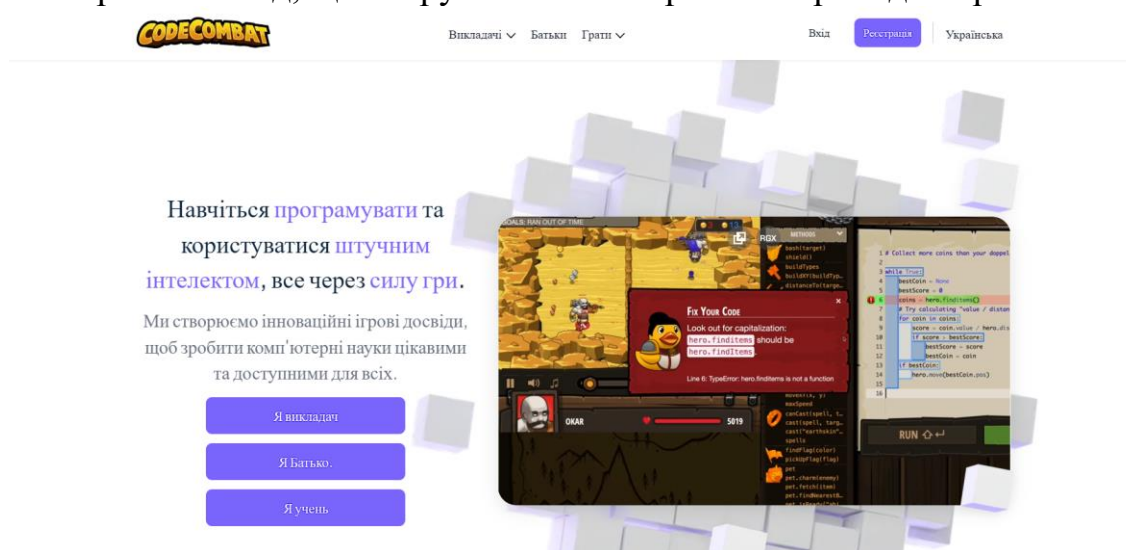


Рис.1.5. Вікно реєстрації

У ній користувачі вивчають різні мови програмування, такі як JavaScript, Python та Lua, виконуючи завдання у фантастичному світі. Навчальний процес у

CodeCombat побудований поступово. На початку гравцям пропонуються прості завдання, які знайомлять їх із базовими поняттями програмування. З часом рівні стають складнішими, що дозволяє опанувати цикли, умовні оператори, функції та роботу з об'єктами. Цей прогресивний підхід гарантує, що учасники не відчуватимуть перевантаження, поступово зміцнюючи свої знання.

Однією з найбільших сильних сторін платформи є інтерактивний геймплей. У CodeCombat ви пишете код у редакторі, а потім одразу бачите, як ваші дії впливають на персонажа та події у грі. Це не лише робить навчання цікавим, а й допомагає краще зрозуміти, як працює код, адже результати видно в реальному часі. Такий формат знижує бар'єр входження для тих, хто лише починає вивчати програмування.

Рівні різноманітні — від боротьби з ворогами до розгадування головоломок, і всі вони потребують застосування програмістських навичок.

CodeCombat також має соціальну складову: гравці можуть співпрацювати, ділитися досягненнями та навчатися разом. Це платформа, яка перетворює навчання програмуванню на захопливу пригоду.



Рис.1.6. Можливості гри Code Combat

CodeCombat - це чудовий спосіб вивчити програмування для людей різного віку та рівня підготовки. Він робить процес навчання цікавим, інтерактивним та мотивуючим.

Ruby Warrior - це гра, спеціально розроблена для вивчення мови програмування Ruby в інтерактивному та захоплюючому форматі. Вона поєднує в собі елементи класичних RPG та головоломок, де ви керуєте воїном, який повинен піднятися на вершину високої вежі, щоб знайти дорогоцінний рубін.

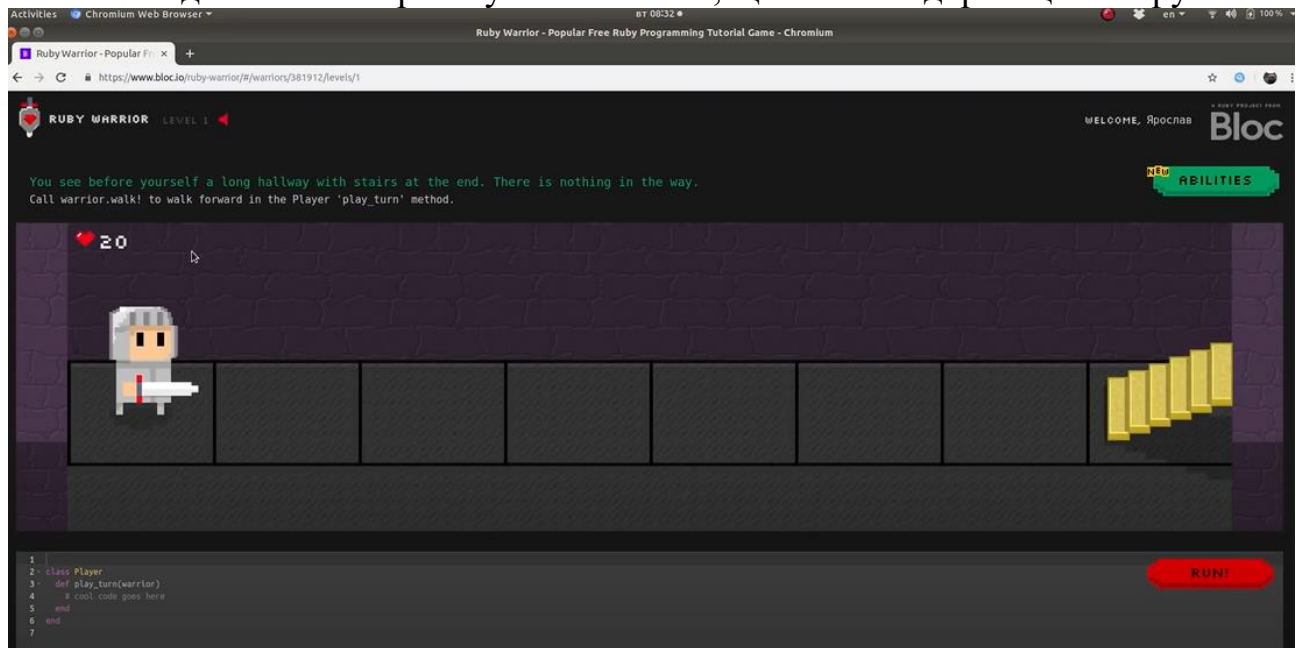


Рис.1.2.6. Перший рівень гри Ruby Warrior

Гравці пишуть код, щоб визначати дії персонажа: переміщення, атака ворогів, захист і взаємодія з оточенням. Наприклад, ви можете спрямовувати воїна вперед, атакувати ворогів мечем, відновлювати здоров'я або збирати предмети. Це не лише розважає, а й наочно демонструє, як працює код.

Навчання в грі побудоване поступово. На перших рівнях ви знайомитеся з базовими командами та синтаксисом Ruby, але з кожним новим етапом складність зростає. Гравець освоює умовні оператори, цикли, методи й об'єкти, застосовуючи їх для подолання різноманітних перешкод. Наприклад, вороги мають унікальні здібності, а пастки вимагають розуміння алгоритмів для їхнього обходу.

Гра пропонує можливість налаштування персонажа: покращення характеристик, придбання нової зброї та обладунків за зароблені очки. Це додає стратегічного елементу, адже від вибору обладнання залежить ефективність у бою. Усі дії виконуються через вбудований редактор коду з підсвіткою синтаксису, що полегшує написання й перевірку команд.

1.3 Психологічні основи

Гейміфікація – це не просто інтеграція ігрових елементів у навчальний процес. Вона базується на психологічних принципах, що враховують природні потреби людей у пізнанні, взаємодії та самореалізації. Завдяки цьому підходу навчання стає більш привабливим, мотивуючим і ефективним.

Внутрішня мотивація та залученість.

Основною перевагою гейміфікації є пробудження внутрішньої мотивації. Уявімо студента, який вирішує складні завдання не заради оцінок, а через схильність до відкриття та відчуття досягнення. Ігрові елементи, такі як квести, випробування та досягнення нових рівнів, створюють почуття пригоди та заохочують до навчання. Це дозволяє студентам отримувати задоволення від процесу та ставити перед собою нові цілі.

Свобода вибору.

Гейміфікація пропонує студентам вибір шляху навчання. Можливість обирати темп, стиль та напрямок навчання дає відчуття контролю. Завдяки цьому учні стають більш відповідальними за свій прогрес і більш відданими процесу. Вони можуть зосереджуватися на аспектах, які їм найбільше цікаві, що підвищує їхню мотивацію та залученість.

Соціальна взаємодія та співпраця.

Людина – соціальна істота, і це враховується у гейміфікації. Таблиці лідерів, командні змагання та спільні завдання дають студенту відчуття розвитку. Ці елементи не лише задовольняють потреби в конкуренції та спілкуванні, а й створюють атмосферу підтримки та співробітництва. В результаті студенти вчаться працювати разом, допомагати одне одному.

Позитивне підкріплення та система винагород.

Ефективність гейміфікації багато в чому забезпечується системою нагород. Кожна значна дія студента, наприклад, виконання завдання чи досягнення нового рівня, супроводжується позитивним підкріпленням: бейджами, віртуальними нагородами чи плюсом до середнього балу за навчання з дисципліни. Це зміцнює впевненість у собі, стимулює активність і допомагає студентам досягати більш високих результатів.

Зворотний зв'язок як інструмент навчання.

У гейміфікації зворотний зв'язок є важливим елементом, який спрямовує студентів до успіху. Він має бути конкретним, своєчасним і адаптованим до індивідуальних потреб. Наприклад, похвала чи підвищення оцінки. Своєчасність також відіграє ключову роль – миттєвий аналіз результатів дозволяє швидко виправляти помилки та вдосконалюватися. Індивідуалізація зворотного зв'язку враховує потреби кожного студента, що робить навчання ефективнішим.

Емоційний компонент гейміфікації.

Гейміфікація не лише навчає, а й дає позитивні емоції. Відчуття правильно виконаних рішень, задоволення від подолання складних завдань створюють сприятливе середовище для навчання. Крім того, ігровий формат допомагає зменшити стрес і тривогу, оскільки помилки сприймаються не як поразка, а як можливість вчитися. Це сприяє комфортному навчанню та розвитку.

Гейміфікація, спираючись на психологічні принципи, перетворює навчання на цікавий і захопливий процес. Вона розвиває внутрішню мотивацію, надає свободу вибору, сприяє співпраці та забезпечує позитивні емоції. Завдяки цьому студенти не лише отримують знання, а й формують важливі навички для успіху в майбутньому.

1.4 Переваги та недоліки гейміфікації в освіті

Гейміфікація в освіті надає унікальні можливості для підвищення мотивації та залучення студентів. Уявіть собі клас, де учні не просто виконують завдання, а змагаються за лідерство у рейтингах, активно взаємодіють із матеріалом та з нетерпінням чекають наступного уроку. Такий підхід трансформує навчальний процес у цікаву гру, стимулюючи внутрішню мотивацію та бажання досліджувати нові горизонти.

Однією з ключових переваг є покращене засвоєння знань. Ігрові елементи сприяють кращому запам'ятовуванню інформації та розвитку критичного мислення. Студенти активно взаємодіють із матеріалом, вирішують завдання, приймають рішення та експериментують, що допомагає перетворити теоретичні знання на практичні навички. Також гейміфікація стимулює розвиток навичок,

таких як командна робота, лідерство, творчість та комунікація. Ці компоненти є надзвичайно важливими в сучасну епоху глобалізації та цифрових технологій.

Гейміфікація створює позитивну атмосферу в класі, сприяючи співпраці та взаємодопомозі. Студенти не бояться помилок, адже вони сприймаються як частина гри та можливість для навчання. Така атмосфера формує комфортне середовище, в якому кожен учень може впевнено розкривати свій потенціал. Крім того, персоналізація навчання дозволяє адаптувати процес до індивідуальних інтересів та потреб студентів, підвищуючи ефективність і задоволення від навчання.

Наукові дослідження підтверджують позитивний вплив гейміфікації. Згідно з роботою Ванга та Хана (2013), використання ігрових елементів сприяє підвищенню мотивації, залучення та академічної успішності студентів. У свою чергу, Сейлор і Хомнер (2013) показали, що гейміфікація допомагає краще запам'ятовувати та застосовувати знання, завдяки таким механікам, як рейтинги, бейджі та система нагород. Після декількох досліджень Хамарі та Койвісто (2013) підтвердили, що гейміфікація є ефективним інструментом для підвищення мотивації та залученості в навчанні.

Проте, як і будь-який інструмент, гейміфікація має свої обмеження. Надмірна увага до ігрових елементів може відволікати студентів від самого процесу навчання. Іноді учні більше концентруються на досягненні нагород, ніж на засвоєнні знань. Також важливо враховувати індивідуальні особливості студентів, оскільки не всі однаково сприймають ігрові механіки. Гейміфікація має бути гнучкою, щоб адаптуватися до різних стилів навчання.

Технічні складнощі та витрати можуть стати ще одним бар'єром. Для впровадження гейміфікації потрібні ресурси, зокрема спеціальне програмне забезпечення, обладнання та час на підготовку. Крім того, деякі ігрові механіки можуть спричинити нездорову конкуренцію або навіть залежність від гри, тому важливо уважно контролювати їх використання.

Ефективність гейміфікації визначається низкою факторів. По-перше, дизайн ігрових елементів повинен відповідати цілям навчання та бути цікавим для студентів. По-друге, якість навчального контенту залишається ключовим –

гейміфікація не замінить змістовні та актуальні матеріали. По-третє, викладачі повинні володіти необхідними знаннями для інтеграції гейміфікації у свої уроки. Зрештою, індивідуальні потреби студентів мають враховуватися, щоб забезпечити максимальну ефективність цього підходу.

Узагальнюючи данні теми можна сказати що гейміфікація відкриває широкі можливості для покращення навчального процесу, надаючи йому новий рівень захоплення та ефективності. Попри певні виклики, її переваги значно переважають ризики, якщо вона впроваджується з урахуванням індивідуальних особливостей студентів, якісного контенту та ретельного планування.

РОЗДІЛ 2. ПІДХІД В РОЗРОБЦІ ГЕЙМІФІКОВАНОЇ СИСТЕМИ НАВЧАННЯ

У цьому розділі пропонується підхід до вирішення проблеми гейміфікації в додатках для навчання студентів. Зокрема, ми розглянемо різні методи гейміфікації та їх вплив на студентів, а також проаналізуємо ефективність використання гейміфікації для навчання.

Завдання полягає в тому, щоб визначити найбільш поширені проблеми, пов'язані з використанням гейміфікації для навчання, і намітити способи їх вирішення. Крім того, мета полягає в тому, щоб запропонувати індивідуальний підхід до використання більш ефективної гейміфікації для навчання. Метою дослідження є вивчення впливу гейміфікації на мотивацію студентів та успішність у навчанні. У ньому буде розглянуто, які елементи гейміфікації можуть бути найбільш цікавими для користувача.

Вони можуть допомогти їм залишатися мотивованими та зацікавленими в процесі навчання. Таким чином, основна мета системи навчання за допомогою гейміфікації - поліпшити якість роботи. Це досягається шляхом створення цікавих завдань, які нагадують ігри. Цільова аудиторія цієї системи – студенти, які хочуть поліпшити свої знання і навички в вибраній області. Для досягнення цієї мети в системі навчання будуть розроблені різні методи гейміфікації, включаючи вибір елементів гейміфікації та інтеграцію в систему.

2.1 Вибір основних підходів та методів гейміфікації.

Для ефективної гейміфікації системи навчання можна використовувати наступні підходи і методи: використання нагород і значків для виконання певних оцінок і завдань. Наприклад, студенти отримують значки за успішне виконання завдань певного рівня складності або за досягнення певного рівня знань з певного предмету, що може підвищити мотивацію та допомогти студентам зацікавити їх у навчанні.

Візуалізація та використання графіки. Цей метод використовує графічні елементи для відображення прогресу, завдань і досягнень користувача. Візуалізація допомагає користувачам бачити свій прогрес і мотивує їх

продовжувати свої зусилля. Доцільно використовувати таблиці лідерів та групову роботу. Студенти можуть брати участь у турнірах, в яких вони змагаються один з одним для досягнення найкращих результатів, а також працювати в групах, в яких вони вирішують проблеми разом.

Існують також різні можливості зворотного зв'язку та оцінки. Студенти повинні отримувати відгуки про свої досягнення, вчитися на своїх помилках і отримувати поради про те, як їх поліпшити. Це допомагає студентам залишатися мотивованими та відточувати свої навички.

Підхід з використанням історій та квестів допомагає студентам зануритися у світ навчання, створюючи цікаві історії та завдання. Студенти можуть виконувати завдання, які розкривають історію та культуру програмування. Або ігровий світ, або завдання з обмеженим часом виконання, або використання "боїв" або елементів змагання для стимулювання конкуренції між студентами. Ці методи допомагають створити цікаве освітнє середовище.

Крім того, ефективний індивідуальний підхід, що враховує особливості та інтереси студентів. Наприклад, створюючи персоналізоване завдання. Важливо зазначити, що додаток повинен бути простим і зручним у використанні, а також мати зрозумілу систему мотивації та винагороди.

Основними причинами, чому це важливо, є: швидкість роботи: нижчі вимоги до легких додатків та апаратних ресурсів. Він працює швидше, що важливо для стабільної роботи та задоволеності студентів. Простий у використанні: легкий додаток простіший у використанні на різних пристроях, включаючи пристрої з обмеженою пам'яттю та обробкою даних.

Для легких додатків потрібно менше ресурсів пристрою, таких як батареї, що дозволяє пристрою працювати довше без підзарядки. Нижчі вимоги до розробки та тестування легких додатків знижують загальну вартість розробки та обслуговування додатків. Вартість ресурсів для підтримки програми може сильно відрізнятися залежно від ряду факторів, таких як розмір програми, кількість студентів, потреба в ємності сервера і необхідність використання додатком зовнішніх API. Крім того, оскільки вартість підтримки додатків залежить від необхідності оновлення програмного забезпечення, виправлення

помилки, підтримки безпеки та інших факторів, ми пропонуємо простий додаток для вивчення програмування, щоб забезпечити ефективну і зручну взаємодію з користувачем і знизити вартість ресурсів пристрою. Важливо економити гроші і скорочувати витрати на розробку.

2.2 Дизайн гейміфікованих завдань

Уявімо роботу, яка не тільки перевіряє знання, але і надихає на випробування, інтриги, роздуми і пробуджує прагнення до знань. Саме таке завдання лежить в основі ефективної гейміфікації.

Для подальшого розвитку потрібно показати студентам, як нові знання можуть допомогти їм у повсякденному житті, майбутніх професіях та вирішенні реальних проблем. Наприклад, замість абстрактних математичних задач, чим більше студентів побачать практичне застосування своїх знань, розрахувавши сімейний бюджет на поїздку, спланувавши ремонт кімнати або проклавши кращий маршрут для їзди на велосипеді, тим більше вони будуть зацікавлені в його отриманні.

Щоб використати сюжет створимо цікаву історію, яка об'єднує всі завдання в єдине ціле. Це може бути детектив, фантастична пригода, історична реконструкція або навіть комікс. Студенти вирушають у віртуальну експедицію на далеку планету, розгадують таємницю стародавніх скарбів, рятують світ від екологічних катастроф або допомагають головному герою знайти вихід з лабіринту. Цей сюжет надає завданням емоційне забарвлення і робить їх більш захоплюючими і такими, що запам'ятовуються.

Для більшої зацікавленості можна додати елемент непередбачуваності до завдання, щоб привернути увагу студентів. Це секретні місії, бонусні бали, несподівані сюжетні лінії, приховані підказки, "пасхальні яйця" (приховані посилання або жарти). Елемент несподіванки створює "дивовижний" ефект, який робить навчання більш динамічним і незабутнім.

Не варто обмежуватись тестами та письмовими завданнями. Потрібно використовувати різні форми: ігри, вікторини, кросворди, презентації, відео, аудіо, творчі проекти, рольові ігри, дискусії, експерименти і т.д., різні форми

дозволяють нам використовувати різні канали сприйняття інформації для підтримки інтересу студентів і розвитку різних навичок.

Система підвищення кваліфікації є важливою складовою гейміфікації, яка допомагає студентам відчувати, що вони прогресують, досягають успіху та вдосконалюються. Ось кілька ідей про те, як створити ефективну систему просування по навчальному матеріалу.

Поділ навчального матеріалу на рівні складності. Студенти можуть почати з простих завдань і поступово переходити до більш складних завдань, відкриваючи нові можливості і отримуючи доступ до нового контенту, інструментів або функцій. Це створює відчуття постійного прогресу і мотивує студентів розвиватися далі.

Беручи активну участь у дискусіях, варто допомагати іншим студентам, винагороджувати їх балами за проходження тестів та виконання завдань. Для цього можна ввести значки, сертифікати, віртуальні нагороди, звання, систему досягнень, яка відображає успішність студента. Це допоможе студентам відстежувати свій прогрес, порівнюючи свої результати.

Візуалізуючи успішність студентів, використовуючи карту, щоб позначити всі рівні, досягнення, нагороди та шляхи, якими студенти ще не пройшли, допоможе їм побачити свої цілі, зрозуміти, чого вони досягли раніше, і мотивувати їх на подальший розвиток.

Дозволяючи студентам вибирати шляхи навчання та використовувати різні стилі навчання, надається можливість зосереджуватися на аспектах, які їх найбільше цікавлять. Це підвищує їх залучення, мотивацію та робить навчання більш ефективним.

Конкурентоспроможність - це потужна мотивація, яка може значно підвищити залучення студентів до навчального процесу. Ось кілька ідей щодо того, як внести конкурентний елемент у ігрову місію:

Можна створити таблицю лідерів, де студенти можуть переглядати результати порівняно з іншими. Це спонукає їх покращувати свої показники, домагатися лідерства і створювати здорове конкурентне середовище.

Додати командні змагання, де студенти діляться на групи та змагаються між собою. Це допомагає розвивати навички співпраці, спілкування та лідерства, командній роботі, досягати спільних цілей та підтримувати один одного.

Даючи студентам виконати складні завдання вчасно або з обмеженими ресурсами, допоможе їм розвинути інтелект, швидку реакцію, здатність працювати під тиском, ефективно використовувати наявні ресурси і навчить знаходити нестандартні рішення.

Організація турнірів на різні теми створить атмосферу азартного змагання, допоможе студентам закріпити вивчений матеріал і перевірити свої знання в цікавій ігровій формі.

Розробка ігрових завдань - це творчий процес, що вимагає уяви викладача, знань психології студента і здатності інтегруватися в процес навчання з використанням різних ігрових механік. Але результат того вартий. Цікава робота, ефективна система підвищення кваліфікації та елементи конкуренції допомагають усім студентам створити мотивуюче та ефективне навчальне середовище, де вони можуть розкрити свій потенціал, досягти успіху та насолоджуватися навчанням.

2.3. Інструменти та платформи для гейміфікації

У сучасному світі, де інформаційні технології швидко розвиваються, гейміфікація стає все більш популярним методом навчання. Використовуючи природну людську схильність до ігор, дозволяє зробити процес навчання більш цікавим і ефективним. На сьогодні існує багато інструментів і платформ, які можуть допомогти викладачам легко і ефективно застосовувати гейміфікацію в процесі навчання. Розглянемо приклади цих інструментів, їх плюси і мінуси.

Огляд сучасних інструментів:

Classcraft: ця платформа перетворює клас на справжню рольову гру, в якій учні стають відважними героями - чарівниками, воїнами і цілителями. Вони отримують очки досвіду за виконання місій, оновлення персонажів, вивчення нових навичок і спільну роботу для досягнення спільних цілей. За допомогою

Classcraft ви можете створити в класі прекрасну атмосферу пригод, в якій кожен студент відчуває себе важливою частиною команди.

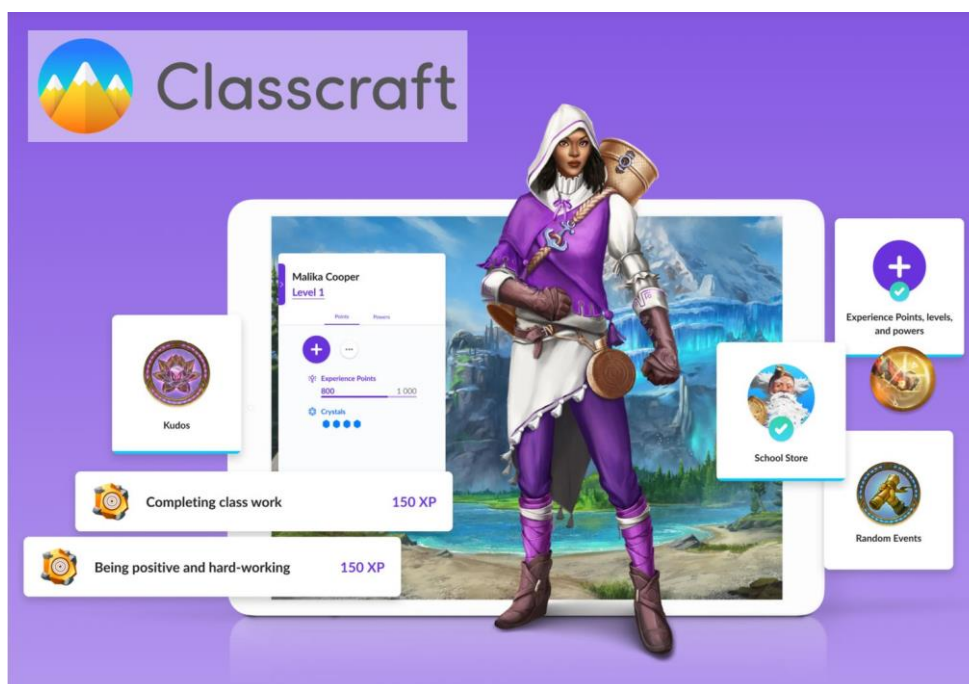


Рис.2.1. Функціонал платформи Classcraft

Переваги: захоплююча історія, широкі можливості персоналізації, інтеграція з іншими платформами, розвиток навичок спільної роботи і спілкування.

Мінуси: платні підписки можуть зайняти більше часу, і вони не підходять для будь-якого віку та предметів.

Kahoot: за допомогою цього інструменту можна створювати інтерактивні вікторини, які перетворюють звичайні уроки в азартні ігри. Студенти відповідають на запитання на своїх смартфонах та планшетах, змагаються за перше місце та надають негайний відгук - це чудовий спосіб перевірити знання студентів, привернути їх увагу та створити атмосферу позитивного збудження в класі.

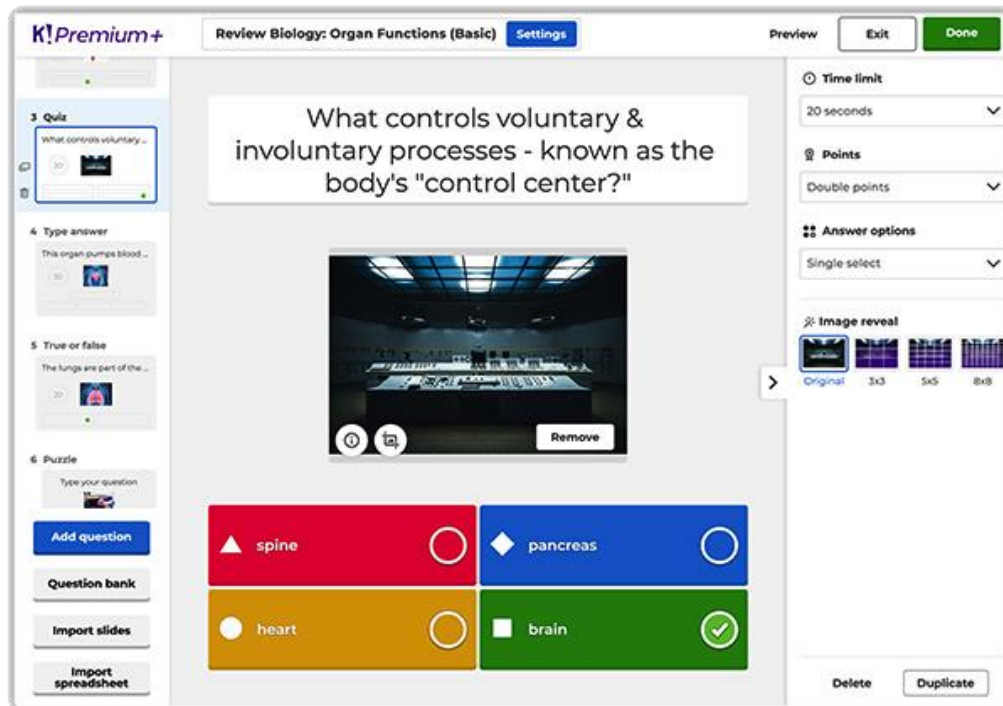


Рис.2 2 Функціонал платформи Kahoot

Переваги: Простота у використанні, безкоштовний доступ, велика бібліотека готових іспитів, можливість створення власного іспиту, мобільність.

Мінуси: обмежені можливості можуть бути недостатньо гнучкими для виконання складних завдань, вони не підходять для всіх видів навчання.

Quizizz: ще одна платформа для створення вікторини, яка пропонує більше можливостей налаштування, ніж Kahoot.

Quizizz дозволяє використовувати різні типи питань (текстові, голосові), регулювати темп гри, додавати меми і GIF-файли, а також інтегрувати вікторини з Google Classroom.



Рис.2.3. Функціонал платформи Quizizz

Переваги: гнучкість, можливість додавати свій власний контент, інтеграція з Google Classroom, різні настройки.

Мінуси: деякі функції доступні лише в платній версії і можуть бути складнішими у використанні, ніж Kahoot.

Blooket: ця платформа являє собою широкий спектр ігрових режимів, в яких учні можуть змагатися за очки, збирати предмети, покращувати свого персонажа, відповідати на питання і вирішувати головоломки, створюючи атмосферу реального ігрового світу в класі.

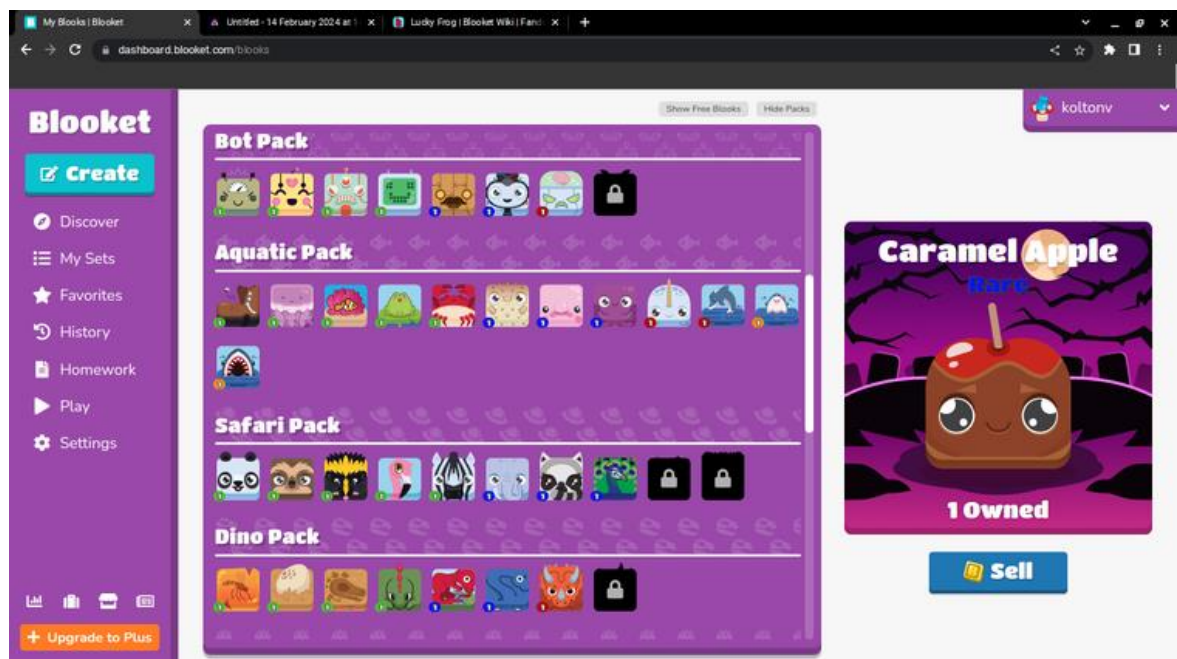


Рис.2.4 Функціонал платформи Blooket

Переваги: цікава ігрова механіка, можливість створення власної гри, вільний доступ, різноманітні режими.

Мінуси: висока складність для початківців і вимоги значних часових затрат, для освоєння функціоналу системи.

Google Classroom: хоча це не спеціальна платформа для гейміфікації, вона надає корисні інструменти, які можна використовувати для створення ігрових елементів у процесі навчання. Наприклад, можна використовувати бали для оцінки завдань, значки для оцінки досягнень та таблиці лідерів для заохочення змагань.

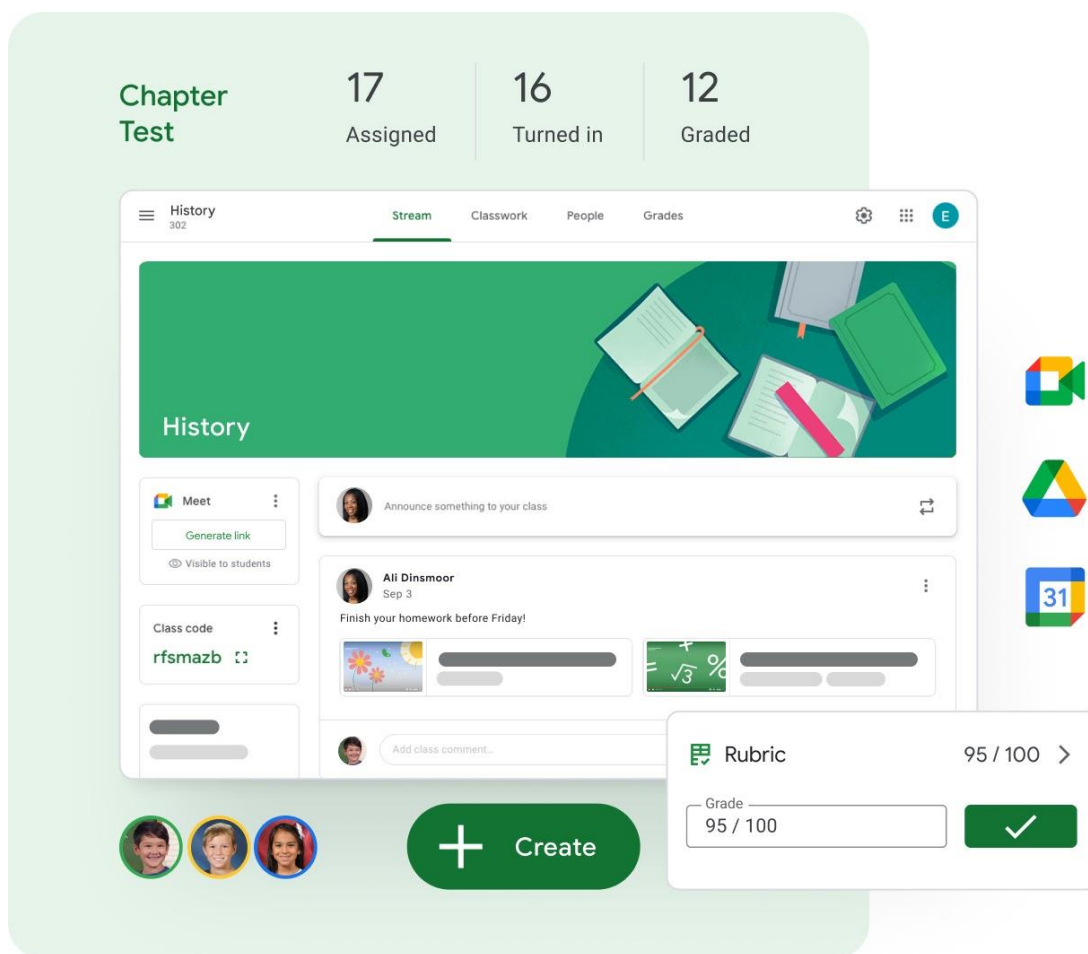


Рис.2. 5 Функціонал платформи Google Classroom

Переваги: інтеграція з іншими сервісами Google, безкоштовний доступ, простота використання.

Недоліки: обмежені можливості гейміфікації, які не підходять для створення складніших ігрових систем.

2.4 Опис використовуваних засобів розробки

Для створення власного додатку, було вибрано платформу Unity, ігрова логіка якої пишеться за допомогою мови програмування C#. Розглянемо, чому саме ця платформа може бути застосована.

Універсальність: Unity не обмежується лише іграми. Її можливості дозволяють створювати різноманітні додатки: від інтерактивних симуляцій та візуалізацій для освіти та науки до програм для архітектури, дизайну та промисловості.

Досить простий та зручний інтерфейс Unity Hub дозволяє нам встановити потрібну чи актуальну версію, додаткові плагіни та платформи за зручної роботи.

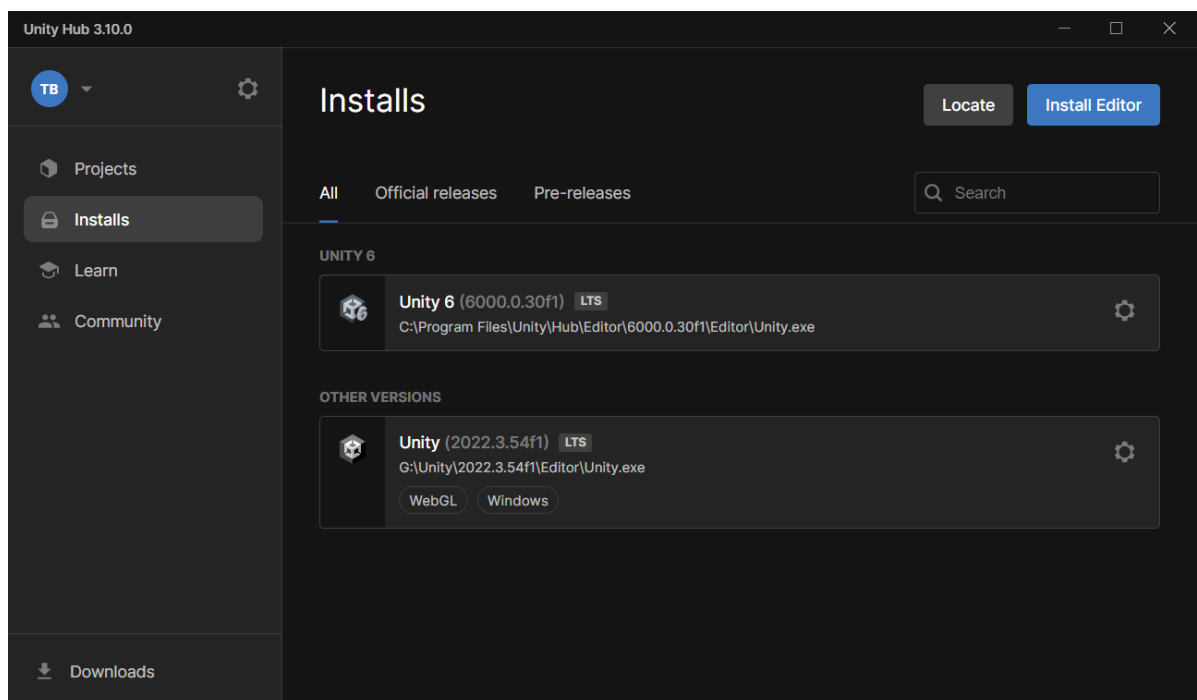


Рис.2.6. Інтерфейс Unity Hub

Платформа є досить простою в засвоєнні, та зручною у використанні, і ось її ключові аспекти:

Ефективність: Завдяки оптимізованому рушію та можливості крос-платформної розробки, Unity дозволяє створювати високоефективні додатки, які працюють плавно на різних пристроях, від мобільних телефонів до потужних комп'ютерів. Програма може бути скомпільована як під додаток, так і під сайт.

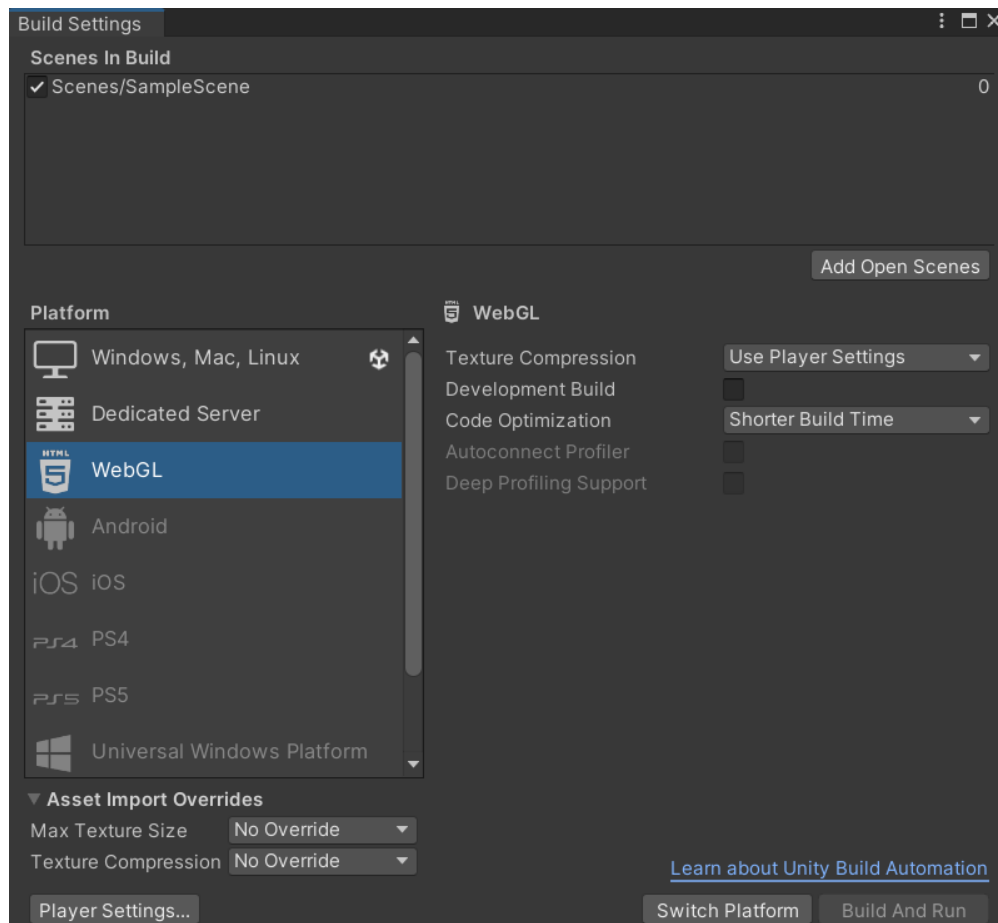


Рис.2.7. Компіляція під HTML сайт

Гнучкість: Unity підтримує широкий спектр форматів файлів та інтегрується з різними інструментами розробки, що надає розробникам свободу вибору та гнучкість у процесі створення додатків.

Розширюваність: Функціональність Unity можна розширити за допомогою Asset Store, величезної бібліотеки готових ресурсів, плагінів та інструментів, створених спільнотою розробників. Це дозволяє швидко додавати нові функції та можливості до проектів.

Різноманітність жанрів: Unity підходить для створення ігор будь-якого жанру, від простих казуальних ігор до складних RPG та шутерів від першої особи.

Інструменти для геймплею: Unity надає потужні інструменти для створення цікавого геймплею, включаючи систему фізики, анімацію, обробку зіткнень, штучний інтелект та систему частинок.

Візуальні ефекти: Вбудований графічний рушій Unity дозволяє створювати цікаву графіку з реалістичним освітленням, шейдерами та ефектами постобробки.

Підтримка VR/AR: Unity спрощує розробку ігор для віртуальної та доповненої реальності, надаючи інструменти для створення цікавих та інтерактивних VR/AR-додатків.

Розробка додатків з Unity:

Інтерактивні симуляції: Unity ідеально підходить для створення інтерактивних симуляцій та тренінгів у різних сферах, таких як медицина, авіація, машинобудування та освіта.

Візуалізація даних: Unity може бути використаний для створення візуалізацій даних, що дозволяє представляти складну інформацію у зрозумілій та привабливій формі.

Прототипування та дизайн: Завдяки можливості швидкого створення прототипів та інтерактивних моделей, Unity є цінним інструментом для дизайнерів та інженерів.

Доповнена реальність (AR): Unity дозволяє створювати AR-додатки, які накладають віртуальний контент на реальний світ, відкриваючи нові можливості для взаємодії з користувачами.

Unity першочергово підтримує C# як основну мову програмування для написання скриптів та логіки гри. C# - це об'єктно-орієнтована мова, розроблена Microsoft, яка відносно легка в освоєнні, але водночас досить потужна для створення складних ігор.

Хоча C# є основною мовою, Unity також пропонує певну підтримку для інших мов та інструментів:

Unity Visual Scripting: Це візуальний інструмент програмування, який дозволяє створювати логіку гри без написання коду. Він ідеально підходить для новачків та дизайнерів, які не мають досвіду програмування.

Bolt: Ще один візуальний інструмент скриптування, який раніше був доступний як окремий актив, але тепер інтегрований в Unity.

JavaScript (UnityScript): Раніше Unity підтримував власну версію JavaScript, але її підтримка була припинена. Існуючі проекти на UnityScript все ще можуть працювати, але нові проекти рекомендується створювати на C#.

Boo: Ще одна мова програмування, схожа на Python, яка раніше підтримувалася Unity, але її підтримка також була припинена.

Чому C# є основною мовою в Unity?

Мова програмування C# дозволяє реалізовувати складну логіку гри та ефективно працювати з ресурсами.

C# підтримує об'єктно-орієнтоване програмування, що дозволяє створювати модульний та легко масштабований код. Також він має велику спільноту розробників, що означає легкодоступність до багатьох ресурсів, навчальних матеріалів та підтримки.

Оскільки C# інтегрований в платформу .NET, це надає доступ до широкого спектру бібліотек та інструментів.

Розробляючи ігри в Unity, рекомендується зосередитися на вивченні C#. Це дасть найбільшу гнучкість, та можливості для створення ігор будь-якої складності.

Отже Unity - це потужна та універсальна платформа, яка надає розробникам широкий спектр інструментів та можливостей для створення ігор та додатків будь-якої складності.

Розглядаючи середовище для написання програми рекомендовано зупинитись на Visual Studio Code.

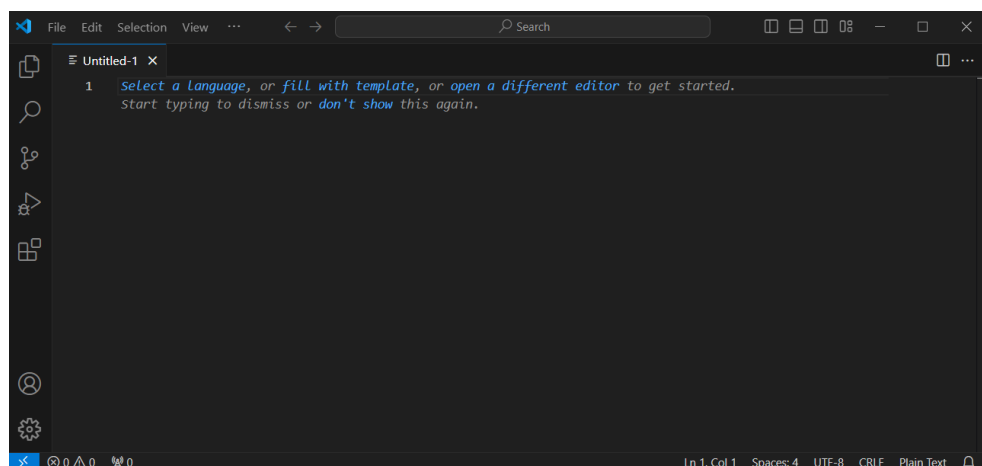


Рис.2.8. Контекстне вікно Visual Studio Code

Visual Studio Code (VS Code) чудово підходить для роботи з Unity. Хоча Unity має власний вбудований редактор коду, багато розробників віддають перевагу VS Code завдяки його гнучкості, широкому функціоналу та зручності.

Основні аспекти при яких VS Code добре працює з Unity:

Легкість та швидкість: VS Code - це легкий редактор, який не споживає багато ресурсів, що особливо важливо при роботі з ресурсоємними проектами Unity.

Відмінна підтримка C#: VS Code має чудову підтримку C#, основної мови програмування Unity, завдяки розширенню C# від Microsoft. Це розширення надає IntelliSense, дебагінг та інші корисні функції для роботи з C#.

Інтеграція з Unity: VS Code можна легко інтегрувати з Unity, встановивши розширення "Debugger for Unity". Це дозволяє вам запускати та дебажити код Unity безпосередньо з VS Code.

Розширення для Unity: Існують інші корисні розширення для VS Code, які покращують роботу з Unity, наприклад:

Unity Code Snippets: Надає швидкий доступ до фрагментів коду Unity.

Unity Tools: Додає корисні функції для роботи з Unity, такі як автодоповнення для компонентів та швидкий доступ до документації.

Visual Studio Code (VS Code) — це безкоштовний і функціональний редактор коду, створений Microsoft. Завдяки своїй універсальності, широкому набору інструментів та підтримці кількох операційних систем (Windows, macOS, Linux), він швидко став фаворитом серед програмістів.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ НАВЧАЛЬНОЇ СИСТЕМИ

3.1. Архітектура і структура додатку

Для реалізації проєкту було створено гру, під назвою JS Coder2d.

Ключові об'єкти які використовуються у грі: Robot (Робот) — це центральний об'єкт гри, яким гравець керує за допомогою команд. Виглядає він як персонаж або спрайт у 2D. Робот об'єднаний кількома ключовими компонентами, які забезпечують його функціональність, такими як Rigidbody2D який відповідає за фізичні властивості робота. Collider2D дозволяє роботу взаємодіяти з іншими об'єктами гри. Усі рухи робота виконуються за допомогою скрипта, який аналізує введені команди гравця та перетворює їх у відповідні дії.

Target (Ціль) — це об'єкт, до якого має дістатися Robot, щоб виконати завдання гри. Зовнішньо він може виглядати як фінішна точка, позначка або інший орієнтир на екрані. Ціль також має компонент Collider2D, який дозволяє перевірити, чи робот досяг її.

Лістинг 3.4

```
if (Vector2.Distance(robot.transform.position, target.position) < 0.5f)
{
    feedbackText.text = "Success! The robot reached the target.";
}
else
{
    feedbackText.text = "Try again. The robot did not reach the target.";
}
```

Game Manager (Головний менеджер гри) є центральним об'єктом, який керує всім ігровим процесом. Він відповідає за обробку введення гравця, координацію взаємодії між роботом та ціллю, а також за управління станом гри. Game Manager постійно стежить за виконанням завдань та забезпечує оновлення інтерфейсу, наприклад, показуючи повідомлення про успішне завершення рівня чи помилки.

Гра використовує компонентно-орієнтовану архітектуру Unity, де кожен об'єкт (робот, ціль, інтерфейс) реалізований як (GameObject) із відповідними компонентами та поведінкою. Для розробки використовувались основні аспекти,

тобто важливі характеристики, які допоможуть зрозуміти з чого складається додаток, наприклад MVC (Model-View-Controller) – модель, подання, контролер. Коротко про те, що виконує кожен з компонентів і його реалізація в вигляді коду:

Модель (Model) - компонент, який зберігає дані про гру, наприклад, де зараз знаходиться робот і куди він повинен потрапити.

Лістинг 3.1

```
public class GameModel
{
    public Vector3 RobotPosition { get; set; }
    public Quaternion RobotRotation { get; set; }
    public Vector3 TargetPosition { get; set; }
    public GameModel(Vector3 robotStart, Quaternion robotRotation, Vector3
targetPosition)
    {
        RobotPosition = robotStart;
        RobotRotation = robotRotation;
        TargetPosition = targetPosition;
    }
}
```

Наступним аспектом є компонент View(Представлення). Він відповідає за те, як все виглядає — графіка, текст на екрані, кнопки (рисунок 3.1). В коді компоненту задані посилання на об'єкт робота, об'єкт цілі та текст для відображення повідомлення гравцю, через модифікатор доступу public:

Лістинг 3.2

```
public class GameView : MonoBehaviour
{
    public GameObject robot;
    public GameObject target;
    public Text feedbackText;
    public void UpdateRobotPosition(Vector3 position, Quaternion rotation)
```

```

{
    robot.transform.position = position;
    robot.transform.rotation = rotation;
}
public void ShowFeedback(string message)
{
    feedbackText.text = message;
}
}

```

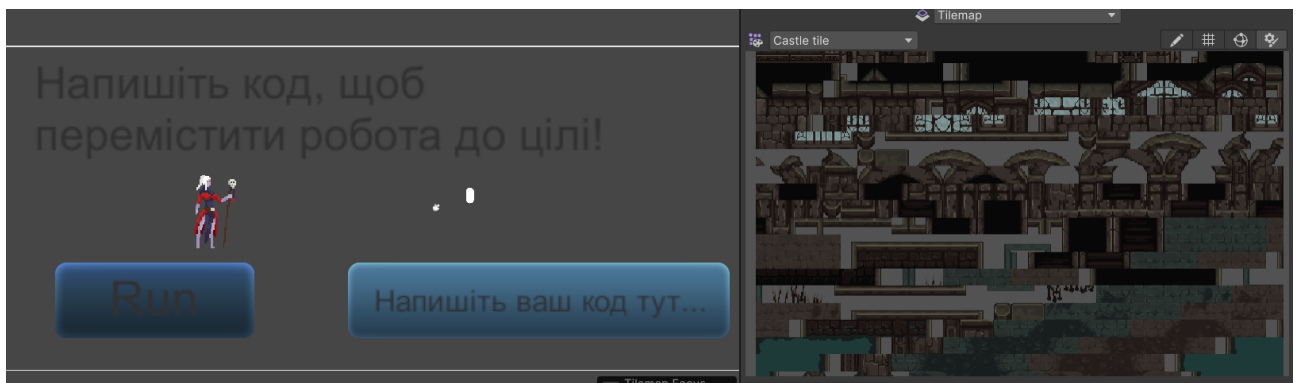


Рис.3.1. Компонент «Представлення», графіка, текст, кнопки

Останнім важливим компонентом є Controller (Контролер). Він обробляє введення гравця та виконує дії, змінюючи стан моделі та вигляд. Контролер у грі виступає як посередник, який приймає введення від користувача через змінну (playerInput), інтерпритує команди та перетворює їх у дію через змінну (parsedCommands), змінює стан гри (gameState), оновлює вигляд гри (gameView), та виводить зворотній зв'язок для гравця (feedbackMessage).

В грі присутня скриптова логіка, де головний скрипт (ProgrammingGame) управляє всією грою, тобто він приймає код, який ввів гравець, і перетворює його на дії, які виконує робот. Скрипт (ProgrammingGame) контролює основну логіку гри. Дії користувача, введені в UI, інтерпретуються через цей скрипт.

Основна структура додатка, або ієрархія об'єктів у грі включає в себе компонент Canvas (UI), це основна частина інтерфейсу гри, яка дозволяє гравцю взаємодіяти з додатком. Вона містить текстове поле (InputField) куди гравець вводить команди, які повинен виконати робот. Кнопка (Button) запускає

виконання коду, коли гравець натискає на неї. Поле тексту (Text) відображає повідомлення, наприклад, результат виконання команд або помилки.

3.2. Реалізація ключового функціоналу

Реалізація ігрового функціоналу створює захоплююче середовище, яке розважає та сприяє розвитку важливих навичок у програмуванні. Гравці вчаться писати логічні команди, аналізувати результати їх виконання і вдосконалювати свої рішення через спроби та помилки. Можливість створювати, тестувати та виконувати код у зрозумілому середовищі є основним засобом реалізації системи.

Основні функціональні можливості та способи їх реалізації які є у додатку наступні: введення та виконання коду, контроль руху робота, взаємодія з ціллю, система перевірки помилок та динамічний інтерфейс користувача.

Введення та виконання коду можна описати так - гравець вводить команди в текстове поле через інтерфейс користувача (UI). Ці команди відповідають базовим функціям програмування, таким як рух робота вперед, повороти чи виконання складніших дій. Код обробляється через спеціальний алгоритм, перевіряє їх на правильність і виконує відповідні дії

Лістинг 3.5

```
private void CheckPuzzle1() // Перевірка відповіді для рівня 1
{
    if (codeInput.text == "console.log('Hello, World!');")
    {
        feedbackText.text = "Правильно! Рух до цілі.";
        target = target1;
        EnableMovement();
    }
    else
        feedbackText.text = "Неправильно, спробуй ще раз.";
}
```

Робот у грі є ключовим елементом. Його рухи визначаються командами гравця. Наприклад, функція MoveForward() змушує робота рухатися вперед на одиницю простору, а TurnLeft() або TurnRight() дозволяють змінювати напрямок, як зображено на рисунку 3.2. Для досягнення цього використовуються

компоненти Unity, такі як Rigidbody2D для фізичних дій і Transform для обертання.



Рис.3.2. – Напрямки руху персонажа

Лістинг 3.6

```
private void ParseCommand(string command)
{
    if (command.StartsWith("robot.MoveForward()"))
    {
        robot.transform.position += new Vector3(1, 0, 0); // Move right
    }
    else if (command.StartsWith("robot.TurnLeft()"))
    {
        robot.transform.position += new Vector3(-1, 0, 0); // rotation=
Quaternion.Euler(0, 0, robot.transform.rotation.eulerAngles.z + 90); // Turn left
    }
    else if (command.StartsWith("robot.TurnRight()"))
    {
        robot.transform.rotation = Quaternion.Euler(0, 0,
robot.transform.rotation.eulerAngles.z - 90); // Turn right
    }
}
```

Взаємодія з ціллю досягається за допомогою перевірки відстані між об'єктом «Robot» і ціллю. Якщо відстань менша за певний поріг, виводиться

повідомлення про успішне виконання завдання(рисунок 3.2.2). Для реалізації використовується функція `Vector2.Distance`, яка обчислює відстань між об'єктами.

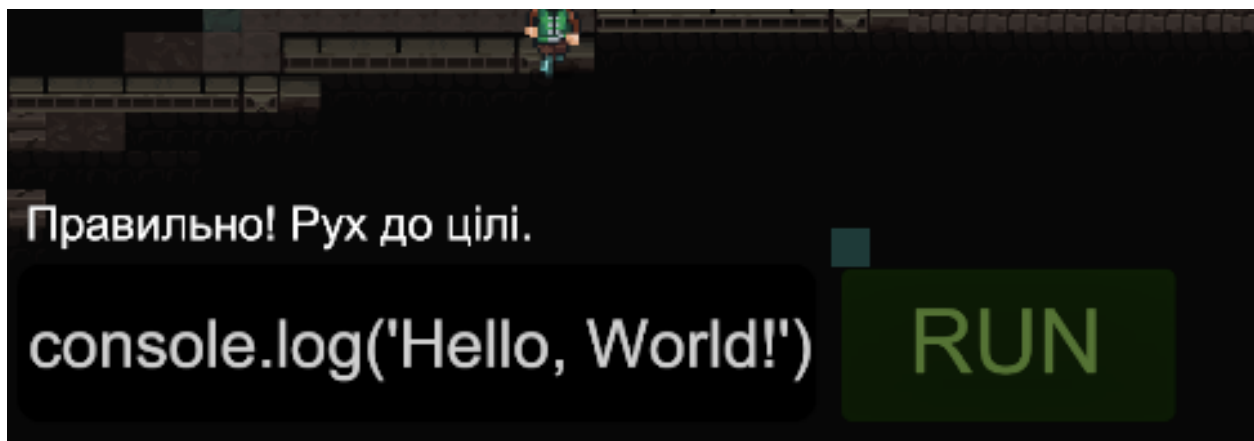


Рис.3.3 - Повідомлення про успішне виконання завдання

Також у грі передбачена перевірка введеного коду на наявність синтаксичних або логічних помилок. Якщо команда введена неправильно або має невірний формат, гравець отримує відповідне повідомлення про помилку в текстовому полі

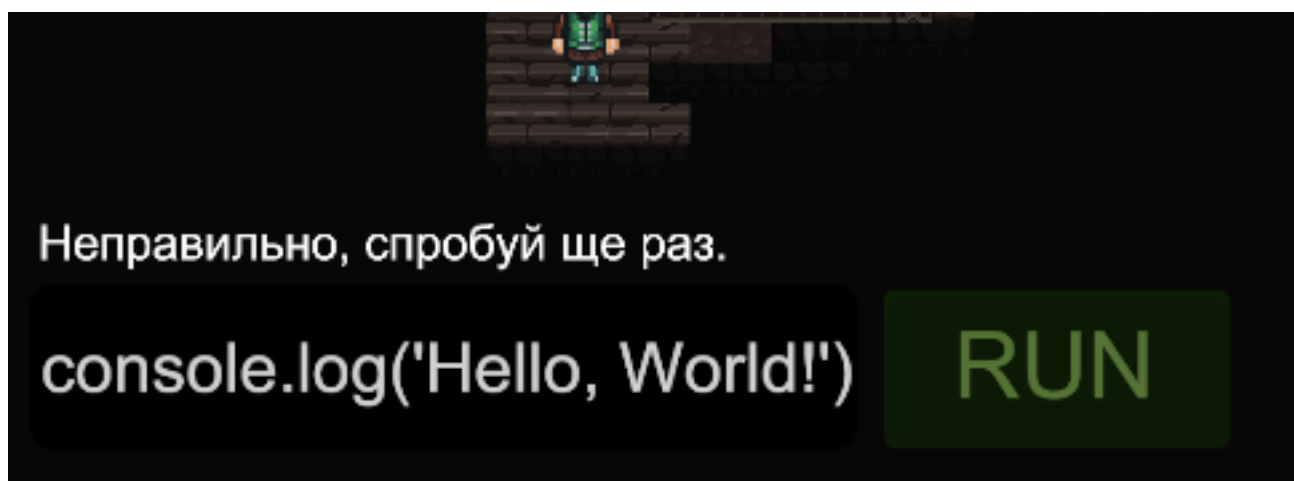


Рис.3.4 – Повідомлення про помилку

Інтерфейс гри дозволяє легко взаємодіяти з усіма елементами. Через Unity Inspector можна налаштувати доступні команди, додаючи їх у список(рисунок 3.5). Гравець може вільно вводити текст і запускати виконання, отримуючи негайний зворотний зв'язок про результат своїх дій.

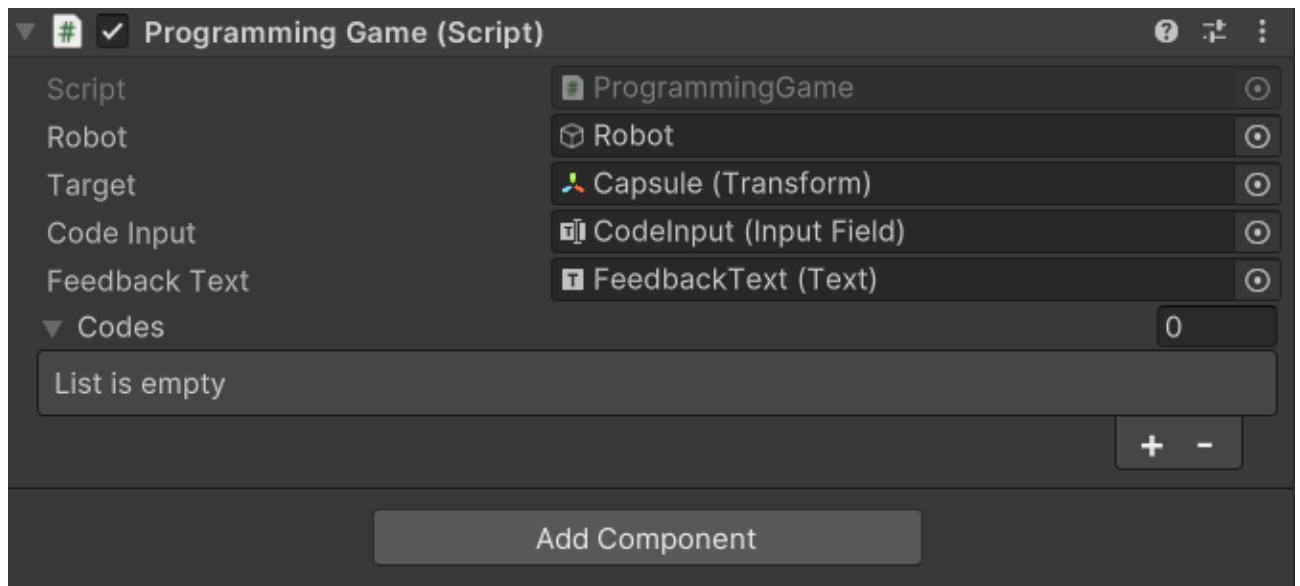


Рис.3.5. Вікно доступних команд та їх додавання

Також в грі реалізована функція “minus”, яка перевіряє кількість успішних виконань коду. Наприклад якщо гравець натисне кнопку “Run”, і дія буде успішною, то вона забирає одну умовну одиницю. Це зроблено для того, щоб зрозуміти скільки гравець зробить помилок при проходженні.

3.3. Тестування додатку

Підводячи підсумок реалізації коду C# на платформі Unity з додаванням плагінів, моделей та текстур отримуємо **досить зручний** інтерфейс гри, зображений на рисунку 3.6, побудовано таким чином, щоб забезпечити просту та інтуїтивну взаємодію користувача з грою.

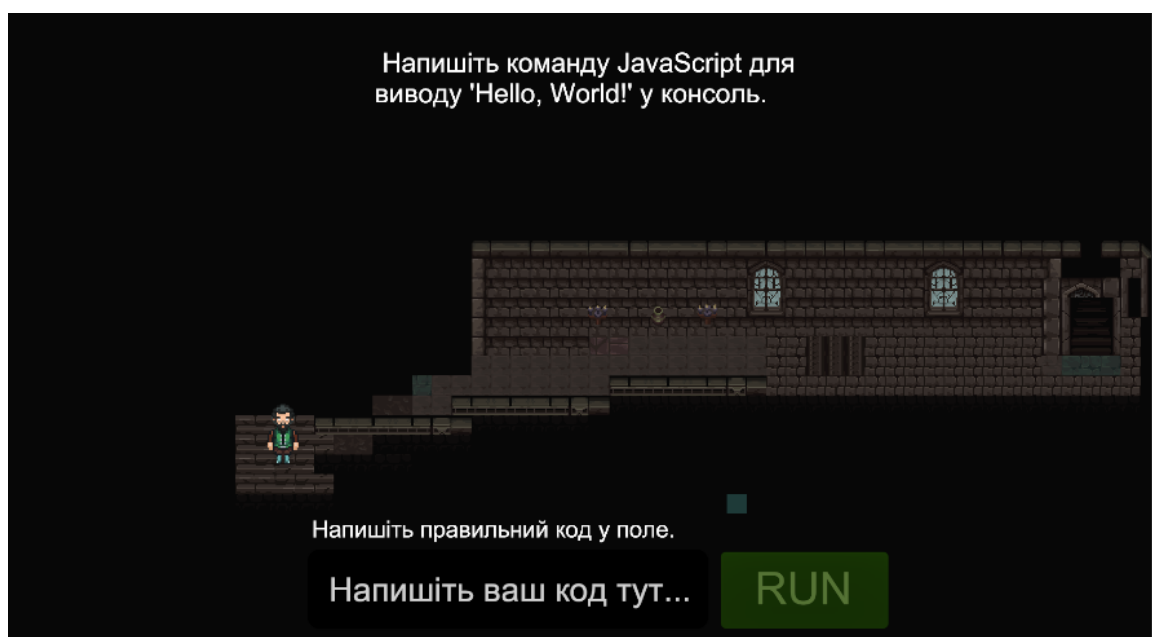


Рис.3.6 – Інтерфейс гри

Основними елементами є текстове поле для введення команд, кнопка запуску команд і поле виведення зворотного зв'язку (рисунок 3.7). Текстове поле дає змогу користувачеві вводити команди у зручному форматі.

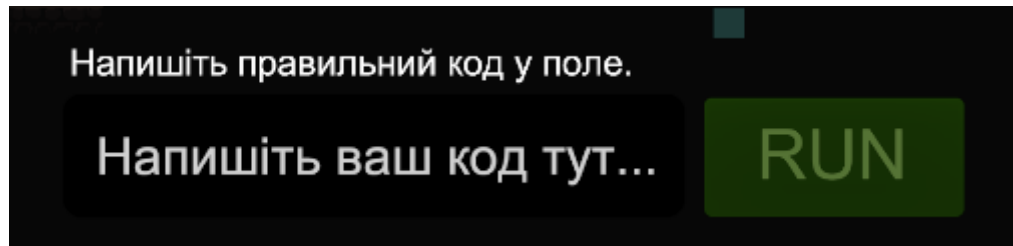


Рис.3.7 – Основні елементи гри

Оскільки гра є інтерактивним додатком для вивчення мови програмування JavaScript, то в ній реалізовано три простих рівні, які покажуть суть проєкту.

В першому рівні (рисунок 3.8) задана одна з найпростіших задач по JS, яка звучить так: «Напишіть команду JavaScript для виводу 'Hello, World!' у консоль». Щоб пройти цей рівень гравцеві потрібно ввести в окреме поле команду, яка є розв'язком данної задачі, в цьому випадку це команда «console.log('Hello, World!');».

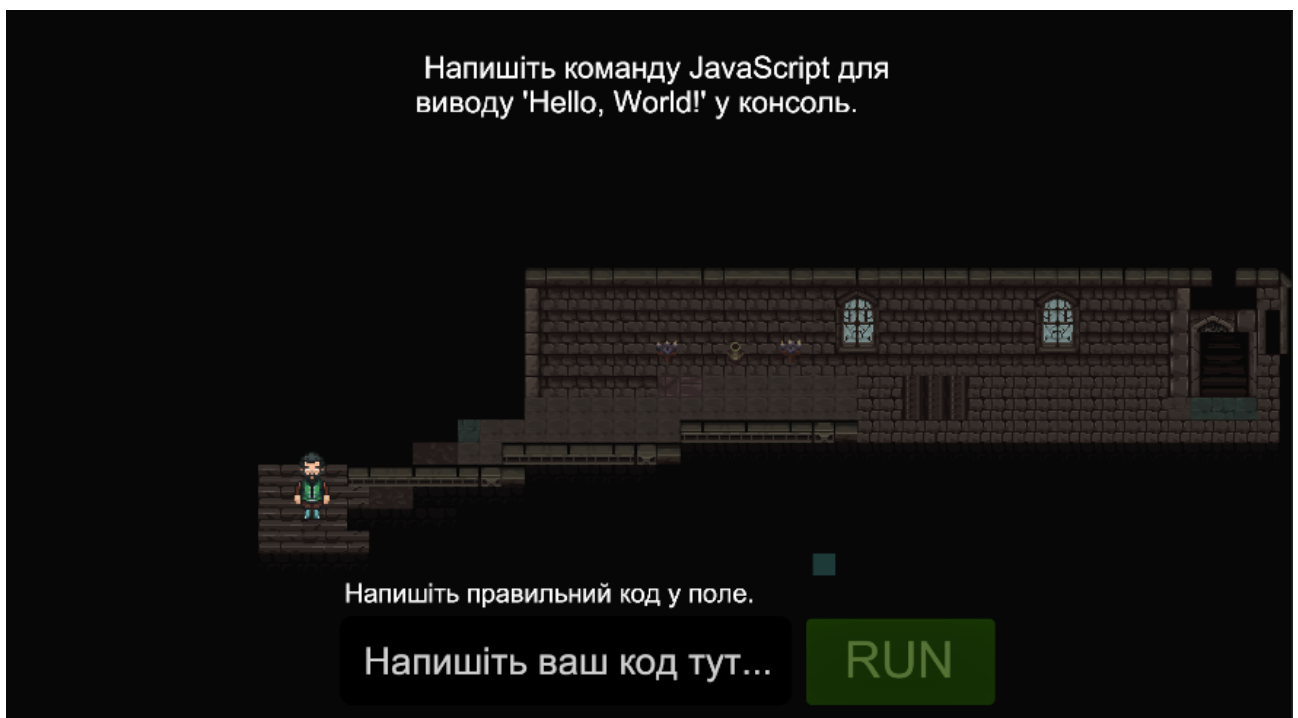


Рис.3.8. – Перший рівень пропонованої гри

В другому рівні (рисунок 3.9) задано дві задачі. Після розв'язку першої персонаж впирається в стіну, і щоб пройти далі потрібно розв'язати другу задачу.

Задачі для розв'язку виглядають так: перший етап – «Напишіть цикл для виконання дії 10 разів.», другий етап – «Напишіть умову для порівняння змінних x та y ».

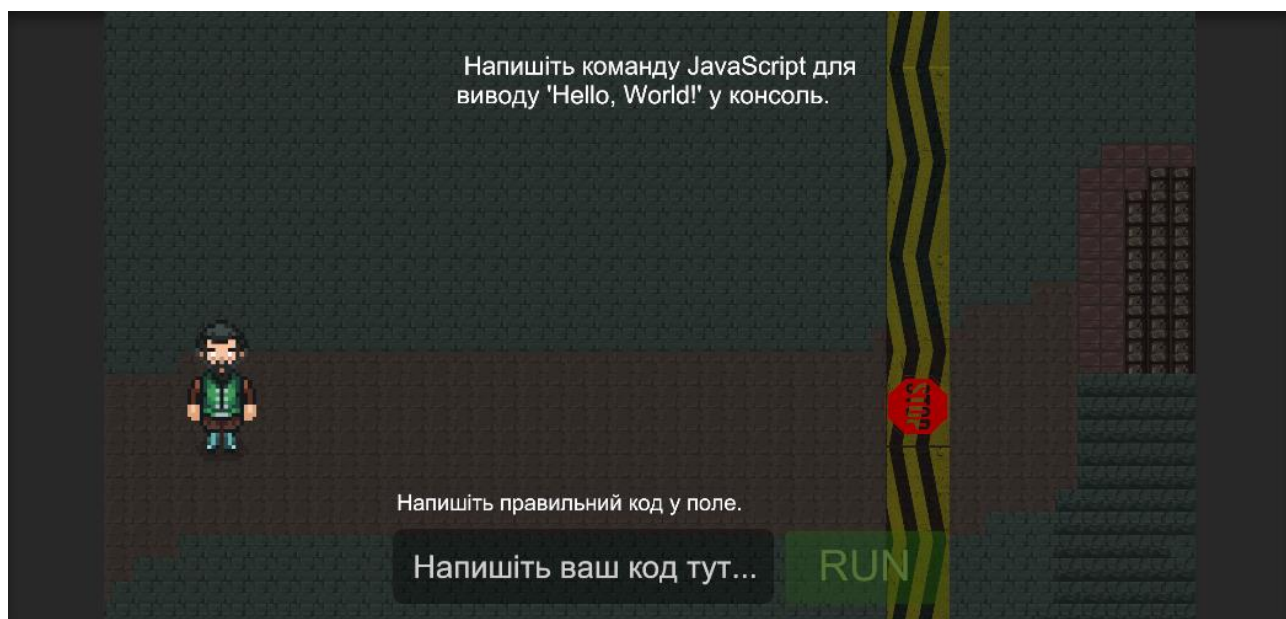


Рис.3.5. – Другий рівень пропонованої гри

Третій рівень(рисунок 3.6) поділений на три етапи, де для кожного задана одна задача. В першому етапі при введенні правильної відповіді персонаж відбиває очки здоров'я в противника. Другий етап реалізовано так само. В третьому етапі персонаж завершує поєдинок та рухається до закінчення рівня.

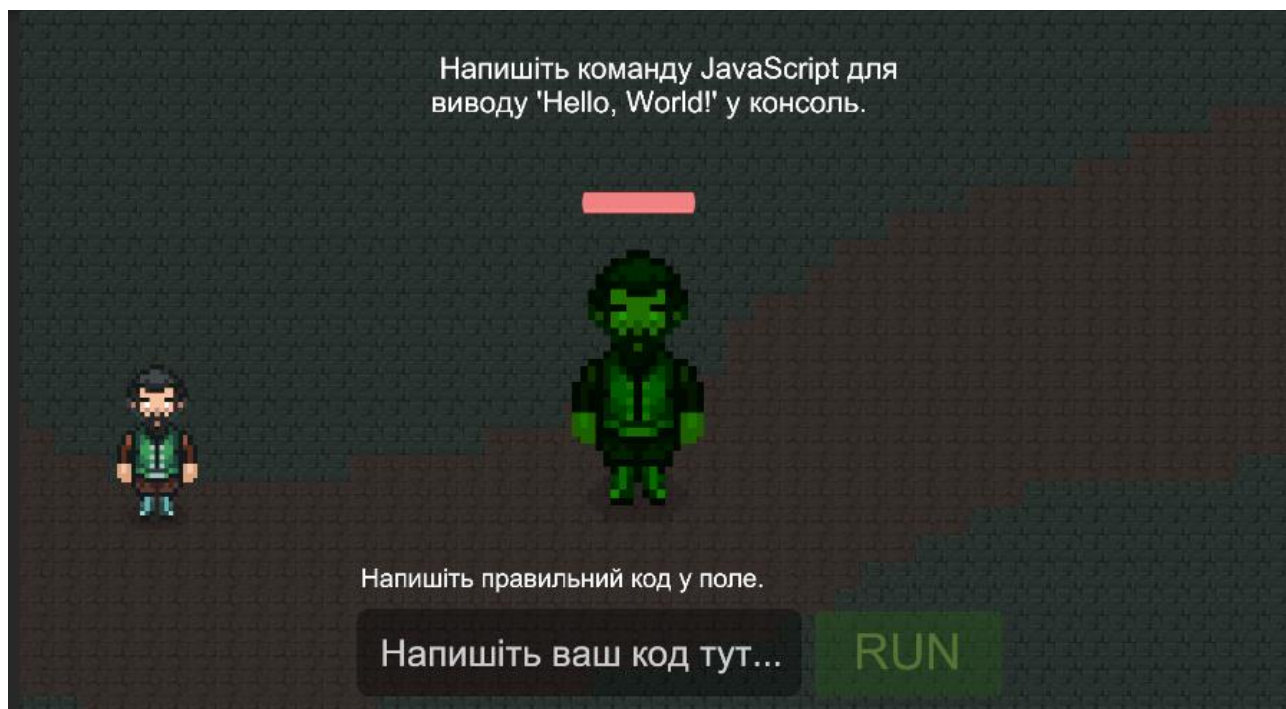


Рис.3.3.5. – Третій рівень пропонованої гри

Задачі для третього рівня: перший етап – «Напишіть ініціалізацію змінної *a*, яка дорівнює 10.», другий етап – «Тепер напишіть умову для перевірки парності числа.», третій етап – «Тепер напишіть цикл для перевірки кількості повторів.».

Додаток співпадає з потребами проєкту, хоч і реалізований на етапі прототипу. Щоб дійсно інтегрувати подібний проєкт в навчання потрібна реалізація наступних етапів, такий як: створення досвідченої команди яка повинна складатись з розробників, дизайнерів, педагогів та методистів, які будуть співпрацювати з викладачами для інтеграції навчальної програми в гру. Такий проєкт потребує значних ресурсів для розробки, тестування та впровадження, тому важливо залучити фінансову підтримку. Також важливо створити інструкції, навчальні посібники та методичні рекомендації для викладачів, які допоможуть їм ефективно використовувати гру в освітньому процесі. Ще одним важливим етапом є оцінювання ефективності, в який входить розробка системи моніторингу та оцінювання успішності. Це дозволить оцінювати вплив проєкту на якість освіти та його потенційне масштабування в інші дисципліни.

Інтеграція такого проєкту стане важливим кроком до модернізації освіти, стимулюючи інтерес до навчання та сприяючи розвитку їхніх програмних навичок у цікавій ігровій формі.

ВИСНОВКИ

Досліджуючи дану тему було проаналізовано як саме гейміфікація може бути корисною в навчанні. В залежності від рівня складності реалізації проекту гейміфікація може бути корисною для кожного, охоплюючи як шкільну програму, так і для студентів коледжу чи університету. Все це буде залежати від інтеграції освітньої програми безпосередньо в проект, а також і від складності завдань.

Гейміфікація вже активно використовується в таких країнах, як Фінляндія, Естонія, Нідерланди, Південна Корея, Японія та деяких інших, хоч і на експериментальному рівні. Для нас ця тема досі знаходиться на рівні інновацій, оскільки використовується на досить низькому рівні, і далеко не у всіх навчальних закладах.

Аналізуючи додатки і платформи, які пропонують нам вивчати матеріал цікавим способом, тобто пов'язуючи традиційне навчання з іграми можна зрозуміти що це досить простий, зручний та цікавий спосіб вивчення та засвоєння матеріалу.

Використовуючи платформу Unity, редактор коду Visual Studio Code та мову програмування C# було розроблено додаток, який хоч і частково, але все ж допоможе познайомитися з цією темою ближче. Додаток показує, як можна інтегрувати елементи навчання в гру.

Гейміфікація навчального процесу є прогресивним напрямком, який суттєво посилює мотивацію студентів і підвищує ефективність освіти. Вона перетворює навчання на цікавий, інтерактивний процес, спрямований на досягнення конкретних цілей. Використання гейміфікованих елементів сприяє розвитку різноманітних аспектів мотивації, зокрема внутрішньої, прагнення до успіху та соціальної взаємодії. Завдяки впровадженню таких підходів студенти активніше залучаються до навчання, краще вправляються у засвоєнні матеріалу та вдосконалюють важливі навички. Гейміфікація може з успіхом застосовуватися у багатьох навчальних дисциплінах і освітніх середовищах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

Додаток А

Код розробки додатку