

ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

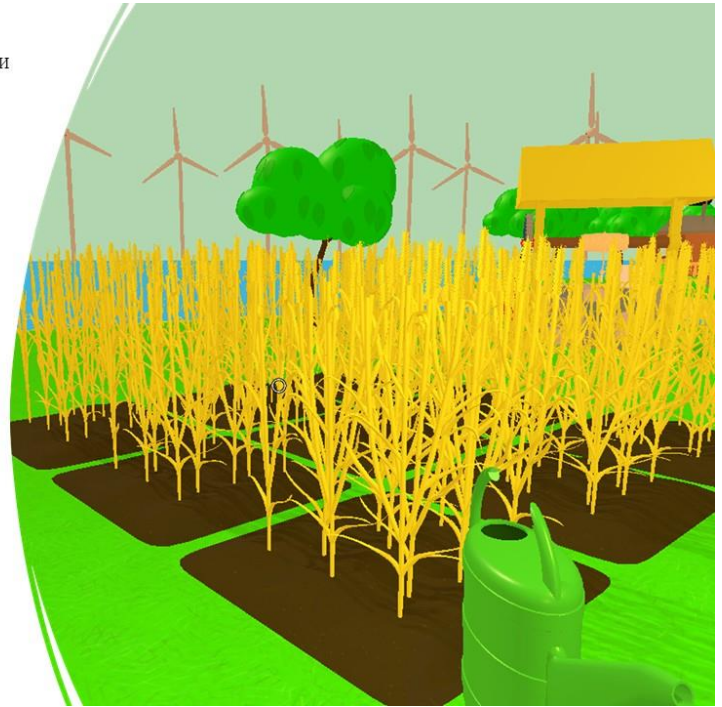
Кваліфікаційна робота

На тему: «Ігровий застосунок у жанрі симулятора з використанням рушія Unity»

Виконав здобувач
групи КІУКУ-22-1
Мухін Є.І.

Керівник
кваліфікаційної роботи
доц. Філімончук Т.В.

Харків 2025



Мета кваліфікаційної роботи

Метою кваліфікаційної роботи є розробка ігрового застосунку в жанрі симулятора фермера. Для досягнення поставленої мети передбачається використання ігрового рушія Unity, мови програмування C# та магазину асетів Asset Store. Проект спрямований на створення функціонального та інтерактивного продукту, що демонструє практичне застосування сучасних інструментів розробки ігор.

Аналіз додатків жанру симулятор фермера

В ході роботи були проаналізовані інші подібні ігри цього жанру:

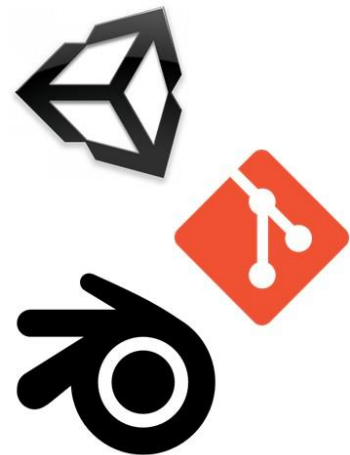
- Slime Rancher – гравець доглядає за кумедними тваринами та продає ресурси що вони виробляють;
- Farming Simulator – дає можливість керувати великою кількістю сільськогосподарської техніки;
- Stardew Valley – гра, з великою кількістю контенту;
- FarmVille – онлайн симулятор фермера, в який можна грати через браузер



3

Технології та інструменти

1. Unity – провідний ігровий рушій, який надає розробникам унікальні можливості для створення інтерактивних застосунків та відеоігор на різноманітних платформах.
2. Git – сучасна система контролю версій, яка надає розробникам широкі можливості для спільної роботи та відстеження еволюції програмного коду.
3. Blender – це багатофункціональна програма для розробки 3D-графіки, яка дає творцям, митцям та проєктувальникам численні інструменти для тривимірного моделювання, створення анімацій, візуалізації та спецефектів.

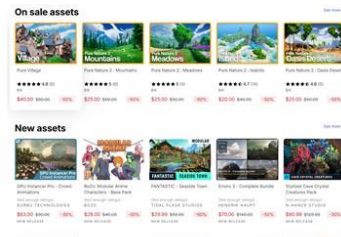
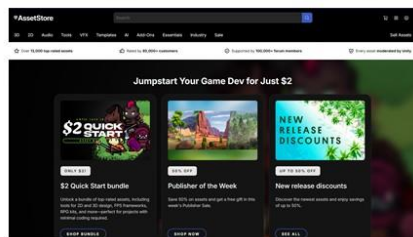


4

Магазин асетів Unity AssetStore

Unity AssetStore – це офіційний інтернет-магазин компонентів для ігрового двигуна Unity, який пропонує творцям ігор доступ до розмаїття готових елементів, що допомагають пришвидшити розробку ігрових та інтерактивних проєктів.

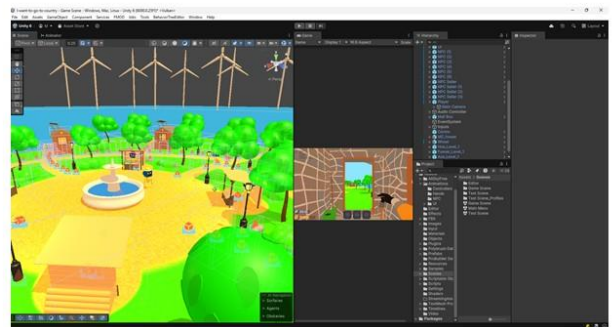
Під час роботи над проєктом, даний магазин використовувався з ціллю прискорення розробки та виділення часу на пріоритетніші напрямки, такі як написання коду чи оптимізація гри.



5

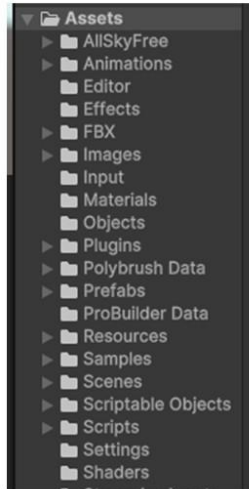
Структура проєкту

Проєкт організовано відповідно до стандартної структури Unity, що передбачає розділення файлів за спеціалізованими каталогами, такими як анімації, моделі, текстури та скрипти. Для архітектури використано підхід Clean Architecture, який забезпечує поділ коду на шари з чіткими відповідальностями. Верхній шар містить контролери, які напряму взаємодіють із об'єктами гри, тоді як нижчі шари обробляють дані. Такий підхід спрощує масштабування проєкту, внесення змін і підтримку коду.



6

Організація ресурсів у проєкті



Ресурси проєкту організовано за каталогами:

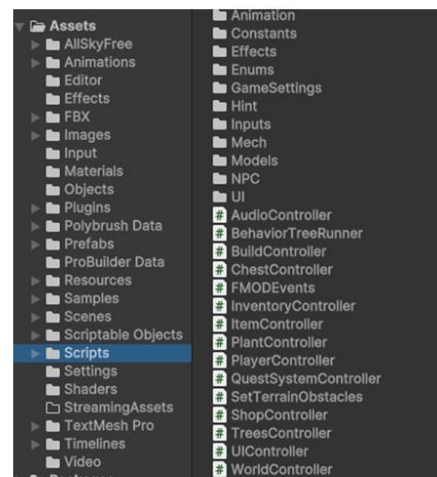
- Animations: анімаційні кліпи та контролери;
- Prefabs: шаблони об'єктів для сцен;
- Scenes: файли рівнів та меню;
- ScriptableObjects: дані для налаштування гри;
- Scripts: скрипти для управління об'єктами та ігровими механіками.

7

Функціонал основних скриптів

Скрипти проєкту реалізують ключові механіки:

- BehaviorTree: реалізує роботу штучного інтелекту в грі;
- PlayerController: контролює персонажа, та виконує взаємодію з іншими об'єктами;
- InventoryController: виконує логіку збереження предметів, які взяв персонаж;
- UIController: контролює та змінює інтерфейс користувача;
- PlantController: опрацьовує зміни пов'язана із рослинами (полив, статус росту, засуха).



8

Неігрові персонажі (NPC)



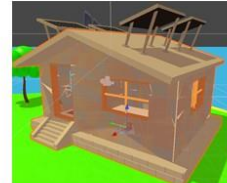
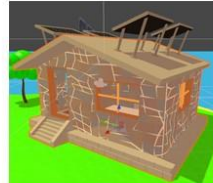
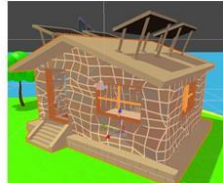
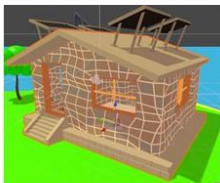
NPC – це ігрові персонажі, які діють автоматично, але з ними можна взаємодіяти. В проєкті для керування поведінкою таких персонажів використовується Behavior Tree – зручна система, яка дозволяє налаштовувати дії NPC прямо в редакторі Unity без складного програмування. Кожен персонаж працює за принципом "дерева поведінки" – структури з різних блоків, що містять умови та команди для виконання. Додатково використовується "загальна пам'ять" – місце, де зберігається інформація, доступна всім частинам системи. Такий підхід робить процес додавання нових персонажів та зміни їхньої поведінки простим і гнучким для майбутнього розвитку гри.

9

Оптимізація

У зв'язку з використанням великої кількості 3D-об'єктів у сценах, виникала загроза зниження продуктивності та появи підлагувань на менш потужних пристроях. Щоб цього уникнути, було впроваджено ряд ефективних методів оптимізації:

- рівні деталізації (LOD) – кожна 3D-модель має кілька версій з різною кількістю полігонів. У залежності від відстані до камери, автоматично підключається відповідний рівень деталізації: чим далі об'єкт, тим менш деталізовану модель бачить гравець;
- Occlusion Culling – механізм дозволяє рендерити лише ті об'єкти, які потрапляють у поле зору камери.



10

Висновки

В рамках кваліфікаційної роботи розроблено комп'ютерну гру-симулятор ферми для операційної системи Windows на базі ігрового рушія Unity.

Робота включала весь процес створення гри – від вивчення ринку та аналізу подібних проєктів до побудови готового ігрового продукту. Реалізовано всі ключові функції гри: рух головного героя, роботу з предметами, вирощування культур, систему завдань та виготовлення речей.

Програмна структура базується на Clean Architecture з використанням принципів SOLID, DRY і KISS, що зробило код організованим, легким для розширення та обслуговування.

Для створення проєкту використовувались сучасні технології: C#, Visual Studio 2022, Git для збереження версій коду та Unity Asset Store для додаткових ресурсів. Підсумком роботи стала надійна ігрова система, яка має потенціал для подальшого вдосконалення або може бути представлена як самостійний комерційний продукт.