

## ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

# Кваліфікаційна робота

Розробка ігрового застосунку в жанрі Strategy на основі патерну MVVM та технології Unity

Виконав:  
ст. гр. КІУКІ-21-1  
А.С.

Серов

Керівник:  
ст. викл. Фомічов О.О.

## Мета роботи

- Створення гри в жанрі стратегії
- Отримання навичок роботи з Unity
- Вивчення патернів програмування

## Актуальність теми

- Unity активно використовують у світі
- MVVM спрощує масштабування проєктів
- Ігри — популярна форма взаємодії



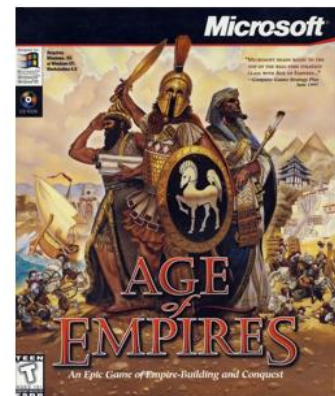
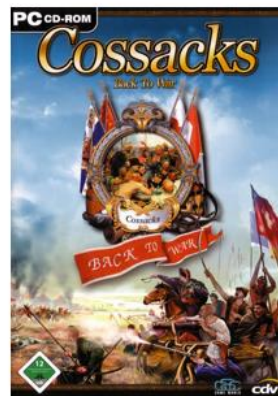
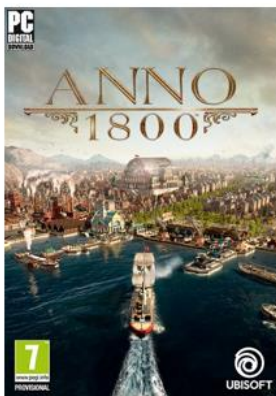
## Вибір жанру

- Стратегії розвивають логіку гравця
- Жанр має сталий попит
- Баланс дій та мислення



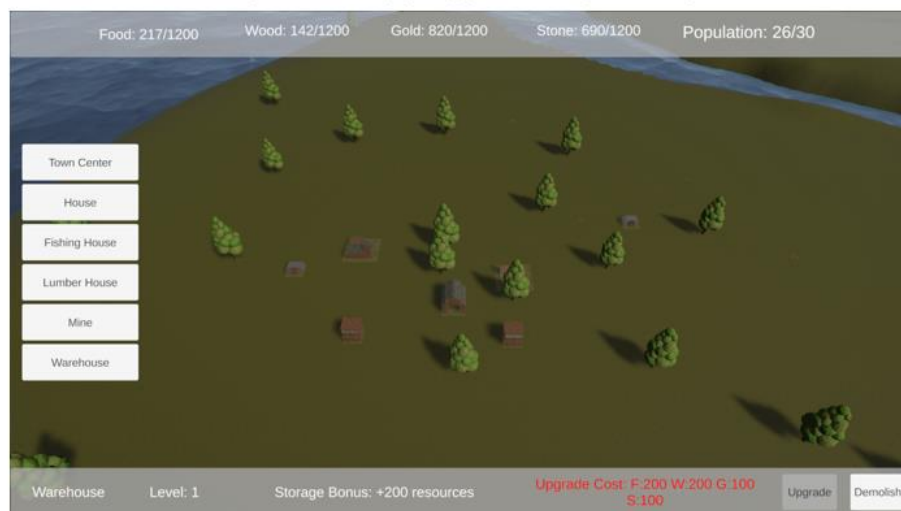
## Приклади схожих ігор

- Anno 1800
- Cossacks
- Age of Empires



## Основні вимоги до гри

- Управління ресурсами та будівлями
- Інтуїтивний, адаптивний інтерфейс UI
- Модульна структура для розширення



## Технічні обмеження

- Лише одна цільова платформа — Windows
- Обмеження продуктивності середнього ПК
- Без використання платних ресурсів



Протестовано



Не протестовано



Mac OS

Не Підтримується

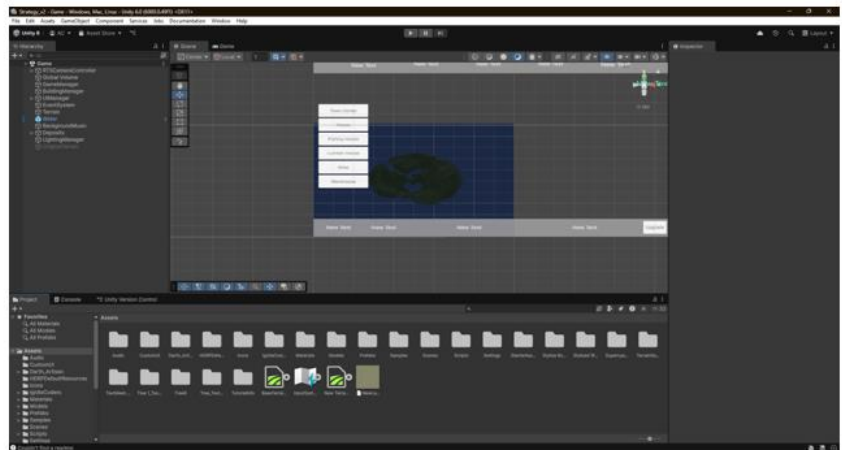
## Часові й ресурсні обмеження

- Обмеження навчальним навантаженням та сесією
- Розробка індивідуальна, без команди
- Ассети — лише безкоштовні ресурси



## Вибір рушія

- Unity — оптимальний для мого рівня
- Підтримує C#, знайомий інструмент
- Мінімальні вимоги до продуктивності системи



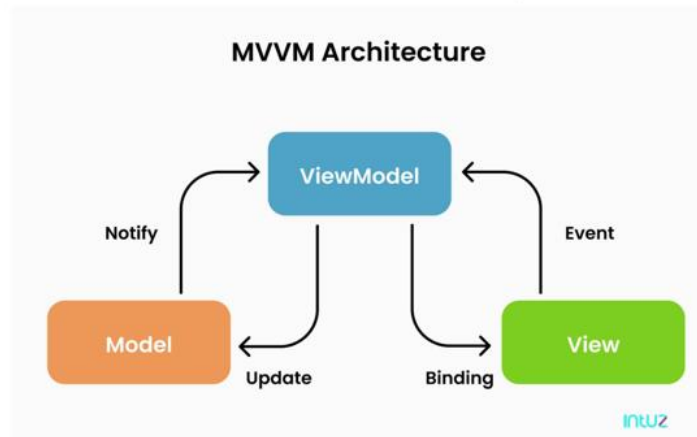
## Альтернативи

- Unreal Engine - неймовірні потужність та вимоги
- Godot - гнучкість, відкритість і мінімалізм



## Патерн MVVM

- Розділення логіки та інтерфейсу
- Зручний для модульного коду
- Підходить для UI-застосунків



## MVVM в Unity

- Потрібна стороння реалізація binding
- Працює з подіями й властивостями
- Полегшує масштабування інтерфейсу гри

```

// Events
public event Action OnResourcesChanged;

// Properties with events
public int Food
{
    get => _food;
    set
    {
        _food = value;
        OnResourcesChanged?.Invoke();
    }
}
  
```

```

private void Start()
{
    // Subscribe to events
    GameManager.Instance.OnResourcesChanged +=
    UpdateResourceDisplay;
}
  
```

```

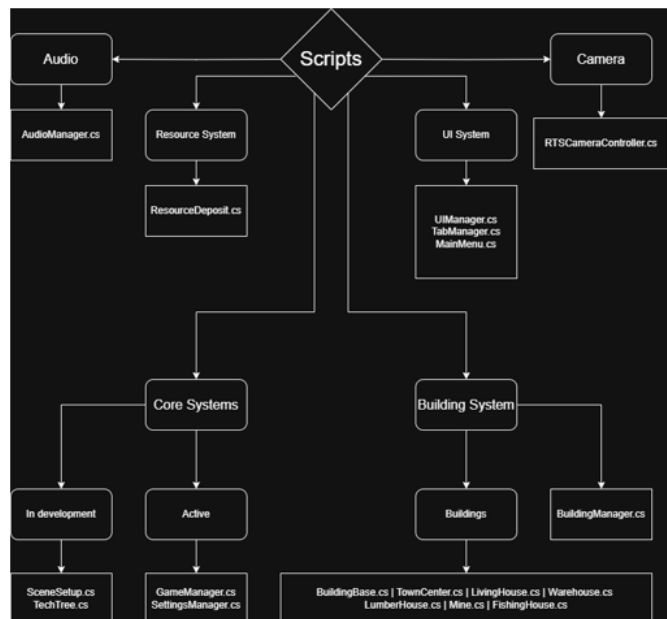
Resource Display
Food Text      [T] Food (Text Mesh Pro UGUI) [O]
Wood Text     [T] Wood (Text Mesh Pro UGUI) [O]
Gold Text     [T] Gold (Text Mesh Pro UGUI) [O]
Stone Text    [T] Stone (Text Mesh Pro UGUI) [O]
Population Text [T] Population (Text Mesh Pro UGUI) [O]
  
```

# Архітектура проєкту

Дві частини: логіка і ресурси.  
Структуровані модулі.

## Основні скрипти

- GameManager
- UIManager
- RTSCamera
- ResourceDeposit
- BuildingBase.



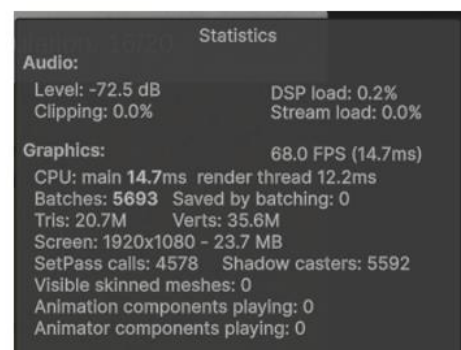
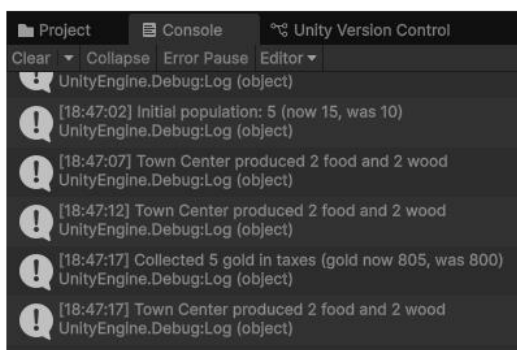
## Ігрова механіка

- Побудова будівель із ресурсами
- Автоматичне зростання населення міста
- Обмеження через дефіцит ресурсів



## Тестування

- Перевірка логіки роботи менеджерів
- Виявлення UI-помилки у грі
- Аналіз продуктивності на середньому ПК



## Можливості розвитку

- Додавання нових типів будівель
- Підтримка багатокористувацького режиму гри
- Порт на інші платформи



## Системні вимоги

- Windows 11 — основна платформа
- Середній ПК для розробника-девелопера
- Низькі вимоги до запуску гри

## Висновки

- Реалізовано базову стратегію на Unity
- Використано MVVM для чистоти коду
- Гра легко масштабована та підтримувана