Università della Calabria

Corso di Laurea Triennale in Informatica

Corso di Virtual Reality – A.A. 2023/2024

Game Design Document

EnchantedVillage

Frandina Ilaria 230569

Sommario

1. Panoramica

1.1 Genere

Il gioco appartiene al genere **strategico** con elementi di costruzione di basi e gestione delle risorse, ispirato al celebre gioco Clash of Clans. I giocatori sono responsabili della costruzione e dello sviluppo di un villaggio, raccogliendo risorse e addestrando truppe per attaccare altri villaggi.

1.2 Meccaniche di gioco in sintesi

Il gioco presenta una **visuale isometrica** che permette al giocatore di avere una panoramica dettagliata del proprio villaggio. Il player può costruire e personalizzare il proprio insediamento acquistando edifici dall'apposito shop, che include una vasta gamma di strutture con diverse funzioni. Gli **edifici** possono essere posizionati liberamente sulla mappa, consentendo una personalizzazione strategica per ottimizzare la raccolta delle risorse e la difesa del villaggio.

Tra gli edifici acquistabili figurano i **collezionatori di elisir** e **collezionatori di oro**, che, una volta piazzati, iniziano a "minare" automaticamente le rispettive risorse. Queste risorse possono essere raccolte dal giocatore una volta che raggiungono una quantità sufficiente, creando un ciclo continuo di gestione delle risorse.

Ogni volta che viene acquistato o potenziato un nuovo edificio, il giocatore guadagna **punti esperienza (EXP)**, che contribuiscono a far salire di livello il proprio account.

Il giocatore ha la possibilità di attaccare villaggi avversari con l'obiettivo di saccheggiare le loro risorse e ottenere vantaggi per la crescita del proprio villaggio.

1.3 Piattaforme di riferimento

"EnchantedVillage" nasce per essere giocato su Mobile ma può essere giocato anche da Pc attraverso il mouse.

2. Descrizione del progetto

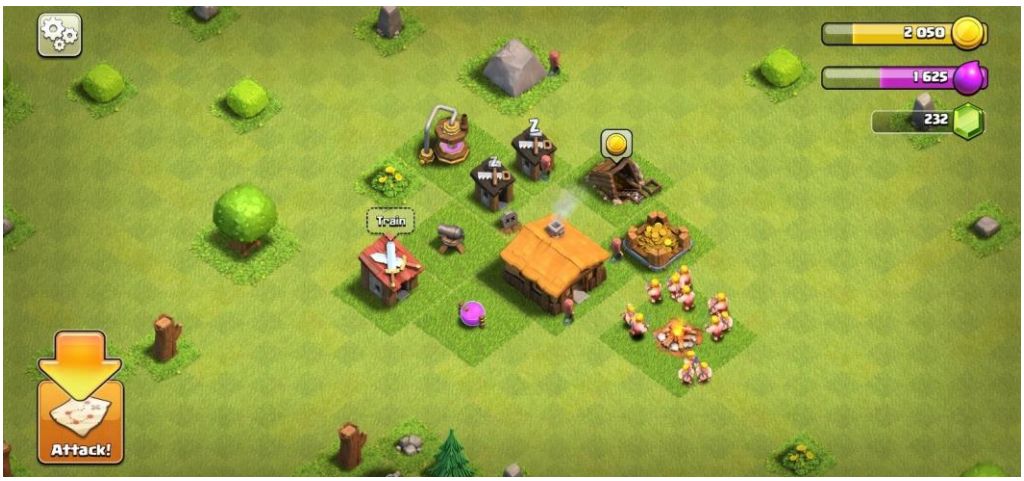
2.1 Ambientazione

Nel vasto continente di **Eldoria**, un mondo antico e ricco di magia, il potere è conteso tra villaggi rivali dopo la caduta di un grande impero. Il giocatore prende il controllo di **Enchanted**, un villaggio magico fondato su terre intrise di **Elisir**, una preziosa risorsa che un tempo alimentava il potere dell'impero.
Con Eldoria divisa tra vari signori della guerra, la missione del giocatore è far crescere **Enchanted** e proteggerlo dagli attacchi nemici. Raccogliendo risorse e addestrando truppe, il giocatore dovrà espandere il proprio regno e saccheggiare i villaggi rivali per ottenere la supremazia. Solo il più forte potrà ristabilire la gloria perduta e riportare **Enchanted** alla grandezza.

2.2 Influenze esterne

Clash of clans

è un popolare gioco mobile sviluppato da Supercell, lanciato nel 2012. Ambientato in un mondo fantastico, il gioco combina elementi di strategia in tempo reale e costruzione di basi. I giocatori costruiscono e personalizzano il proprio villaggio, raccolgono risorse come oro e elisir, e addestrano truppe per difendere il loro insediamento e attaccare quelli avversari.



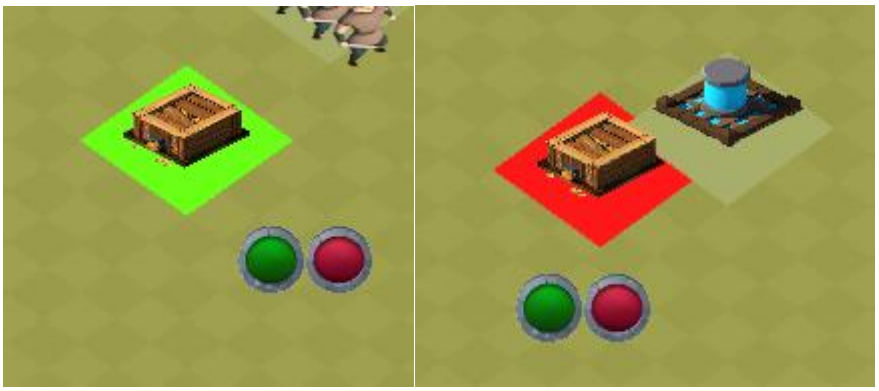
Le meccaniche di gioco di *Clash of Clans* hanno avuto una notevole influenza su *EnchantedVillage*. L'adozione della visuale isometrica, la gestione delle risorse, la costruzione e il sistema di attacchi sono tutti elementi che rispecchiano e si ispirano alle dinamiche del gioco originale. Queste influenze si riflettono chiaramente in ogni aspetto di *EnchantedVillage*, contribuendo a creare un'esperienza di gioco simile e altrettanto coinvolgente.

2.3 Meccaniche di gioco principali

Movimento degli edifici

Nel gioco, tutti gli edifici acquistabili possono essere spostati liberamente all'interno del villaggio. Il giocatore può trascinare gli edifici nella posizione desiderata sulla mappa. Durante il posizionamento, la base dell'edificio cambia colore per indicare la sua idoneità: il verde indica che l'edificio può essere posizionato correttamente, mentre il rosso segnala che l'area non è adatta per il posizionamento.

Il giocatore può confermare la posizione dell'edificio solo quando la base è verde, utilizzando l'apposito pulsante di conferma. Se il giocatore tenta di confermare l'edificio in una zona non valida (indicata in rosso), l'edificio rimarrà "in sospeso", ovvero rimarrà visibile nella mano del giocatore finché non viene posizionato correttamente o annullato tramite il pulsante di annullamento.



Addestramento delle truppe

Attraverso l'apposito pulsante, visibile in alto a destra nella schermata di gioco, è possibile avviare l'addestramento delle truppe. Una volta acquistata una truppa, essa verrà automaticamente assegnata a una delle basi di addestramento nel villaggio che disponga di almeno un posto libero. Questo processo consente al giocatore di gestire e preparare le proprie forze in modo efficiente, ottimizzando le risorse e il tempo a disposizione.

Tra le truppe disponibili per l'addestramento vi sono: vichinghi, arcieri e spadaccini.

Gestione dei livelli

All'interno della schermata principale di gioco, in alto a sinistra, il giocatore può visualizzare il proprio livello.
L'esperienza richiesta, per passare al livello successivo, cresce in modo logaritmico con l'aumentare del livello corrente. Questo significa che l'incremento di esperienza necessario per passare ai livelli successivi aumenta, ma con una crescita decrescente rispetto all'aumento del livello.

```
private int ExperienceForNextLevel(int currentLevel)
{
    const int a = 100;
    const int b = 2;
    const int c = 10;
    return (int)(a * Mathf.Log(b * currentLevel + c));
}
```

Gestione delle risorse

All'interno della schermata principale del gioco, in alto a destra, il giocatore può visualizzare il quantitativo di elisir e oro accumulato. Le risorse possono essere ottenute in due modi principali: combattendo contro altri villaggi e saccheggiando le loro risorse, oppure raccogliendole dagli appositi **collezionatori** di oro ed elisir, che producono continuamente queste risorse.

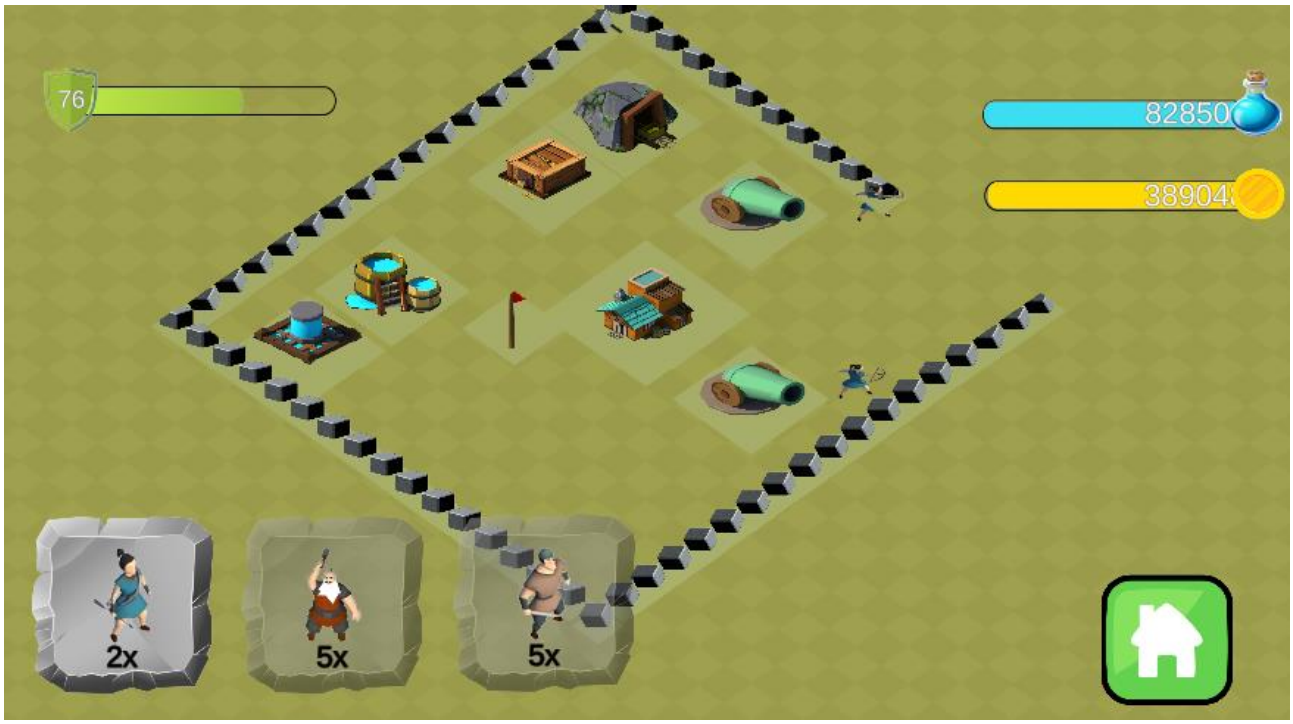
Per raccogliere le risorse dai collezionatori, il giocatore deve semplicemente cliccare su di essi. Quando le risorse sono pronte per essere raccolte, i collezionatori emettono degli **effetti particellari** (particles) visibili, che segnalano visivamente che le risorse sono disponibili. Questo effetto aiuta a indicare al giocatore che è il momento di raccogliere le risorse accumulate.

Gestione delle battaglie

Nella schermata principale del gioco, in basso a destra, si trova un pulsante per avviare una nuova battaglia contro un altro villaggio. Attualmente, è disponibile un solo villaggio da attaccare, e non è ancora possibile posizionare manualmente le truppe.

Quando il giocatore clicca sulla mappa, due arcieri verranno inviati automaticamente per combattere contro il villaggio nemico. Il sistema di riconoscimento degli edifici da attaccare è automatico, e le truppe si orientano verso gli obiettivi principali senza ulteriori interventi da parte del giocatore.

In futuro, è previsto l'aggiornamento per introdurre la possibilità di **scegliere le truppe** da schierare e **posizionarle strategicamente** all'interno della mappa nemica. Questo miglioramento offrirà ai giocatori una maggiore flessibilità e strategia durante le battaglie.



Salvataggio dei progressi e degli edifici

Il salvataggio dei progressi del giocatore e di tutte le modifiche apportate al villaggio avviene automaticamente ogni volta che si verifica un cambiamento significativo o quando il giocatore esce dal gioco.
Per gestire la persistenza dei dati, utilizzo **PlayerPrefs**, che memorizza tutte le informazioni necessarie, inclusi gli edifici e le truppe. I dati vengono serializzati in formato **JSON**, permettendo una memorizzazione e un recupero efficienti delle informazioni.

3. Scene di gioco

Il gioco si svolge principalmente nella **scena del villaggio del giocatore**. Da questa schermata centrale, il giocatore può accedere facilmente ai pulsanti per due funzioni principali: **lo Shop**, dove è possibile acquistare e posizionare edifici, e **l'area di battaglia**, per iniziare nuove incursioni contro villaggi nemici.

3.1 Shop

Alla pressione dell'apposito tasto, in basso a sinistra, nella scena di gioco principale è possibile accedere allo shop di "EnchantedVillage".

Suddiviso in diverse categorie, in ordine:

- Produzione
- Difesa
- Decorazioni
- Costruzioni

Le altre scene, come la scena principale e il villaggio nemico, sono state riportate nelle pagine precedenti.

4. Implementazione

Per permettere il corretto funzionamento del gioco ho fatto uso di diversi script, in grado di interagire tra loro e svolgere specifiche funzioni.

4.1 Camera

EnchantedVillage adotta una **visuale isometrica**, con una vista elevata e inclinata a circa 45°. Il gioco utilizza una **camera ortografica** piuttosto che prospettica.
La camera è stata configurata nella scena con una distanza e un angolo di inclinazione ottimali, ed è impostata su modalità **ortografica**. È gestita da uno script che consente di spostare la visuale all'interno della mappa, assicurandosi che non esca dai confini predefiniti.

4.2 Script

Gli script utilizzati sono stati scritti nel linguaggio di programmazione C#, e sono raggruppati in sottocartelle a seconda della loro funzione:

- **Building**: raggruppa tutte le funzioni legate agli edifici, al loro posizionamento e alla gestione delle risorse;
- **Input**: contiene tutti gli script legati alla gestione degli input;
- **Troops**: raggruppa tutti gli script legati alla gestione delle truppe;
- **UI**: contiene tutti gli script legati alla gestione della grafica, come tutti i pulsanti e lo shop;
- **General**: raggruppa tutti gli script generici, come quello per il controllo della camera (CameraController.cs), quello per il controllo del Player (Player.cs) etc.

Di seguito verrà riportata una panoramica generale sugli script più importanti.

CameraController.cs

Lo script *CameraController.cs*, catalogato come *General*, gestisce la telecamera isometrica del gioco, controllando il movimento e lo zoom. Utilizza una **camera ortografica** e permette la navigazione attraverso:

- **Movimento della camera**: Gestito tramite i metodi MovePressed e MoveStopped, che attivano e disattivano il movimento basato sugli input dell'utente.

- **Zoom della camera**: Controllato con ZoomPressed e ZoomStopped, supporta sia il mouse che i tocchi su dispositivi mobili. Il livello di zoom viene regolato e limitato tra valori minimi e massimi.

- **Aggiornamenti e limiti**: La funzione Update gestisce il movimento e lo zoom della telecamera, mentre AdjustBounds assicura che la telecamera non esca dai confini definiti della mappa.

BuildingController.cs

Lo script *BuildingController.cs*, catalogato come *Building*, gestisce il posizionamento e la movimentazione degli edifici nel gioco. Ecco le principali funzionalità:

- **Gestione del Trascinamento**: I metodi OnMouseDown, OnMouseDrag, e OnMouseUp abilitano il trascinamento degli edifici. OnMouseDown inizia il trascinamento e memorizza l'offset, OnMouseDrag aggiorna la posizione dell'edificio mentre viene trascinato, e OnMouseUp finalizza il posizionamento.

- **Visualizzazione e Conferma**: La visualizzazione della posizione valida o non valida dell'edificio è aggiornata tramite UpdatePlacementVisualization. Il metodo Confirm verifica se la posizione è valida e conferma il posizionamento.

- **Verifica del Posizionamento**: CanPlaceBuilding controlla se l'edificio può essere posizionato nella griglia senza collisioni, mentre HasCollisions verifica se l'edificio collide con altri edifici già presenti.

- **Allineamento alla Griglia**: SnapToGrid allinea l'edificio alla griglia di costruzione, posizionandolo al centro della cella della griglia.

- **Collezione Risorse**: Durante il rilascio del mouse, se l'edificio è un collezionista di risorse, OnMouseUp gestisce la raccolta di elisir o oro, aggiornando le risorse del giocatore.

- **Calcolo della Posizione del Mouse**: GetMouseWorldPosition converte la posizione del mouse in coordinate del mondo utilizzando un piano di riferimento.

TroopsPlacer.cs

Lo script *TroopsPlacer.cs*, catalogato come *Troops*, gestisce il posizionamento delle truppe nelle basi di addestramento e le relative operazioni. Le principali funzionalità includono:

- **Posizionamento delle Truppe**: Metodi come PlaceArcher, PlaceSwordMan, e PlaceViking chiamano PlaceTroops, che si occupa di posizionare il tipo specifico di truppe nella prima base di addestramento disponibile. PlaceTroops gestisce anche la verifica della disponibilità di spazi nelle basi e applica un costo per il posizionamento delle truppe.

- **Aggiornamento e Salvataggio**: Dopo aver posizionato una truppa, lo script aggiorna i dati dell'edificio e salva le modifiche nel PlayerPrefs tramite PlayerPrefsController.

- **Recupero delle Basi di Addestramento**: Il metodo GetTrainingBases estrae tutte le basi di addestramento disponibili dal PlayerPrefs, filtrando gli edifici con un indice specifico.

Player.cs

Lo script *Player.cs*, catalogato come *General*, gestisce i dati e le operazioni relative al giocatore, inclusi il livello, l'esperienza, e la gestione degli edifici e delle truppe. Le principali funzionalità includono:

- **Singleton Pattern**: Player utilizza un pattern Singleton per garantire che ci sia solo un'istanza del giocatore. Se non esiste, viene creato un nuovo oggetto Player.

- **Inizializzazione e Avvio**:
  - Awake: Inizializza i contenitori per edifici e truppe all'interno della mappa.
  - Start: Verifica se è un nuovo gioco o se carica i dati salvati. Se è un nuovo gioco, avvia con valori predefiniti; altrimenti, carica i dati del giocatore esistente.

- **Gestione dell'Esperienza e del Livello**:
  - AddExperience: Aggiunge punti esperienza e controlla se è necessario salire di livello.
  - CheckLevelUp: Verifica se il giocatore ha accumulato abbastanza esperienza per salire di livello e aggiorna il livello e i punti esperienza.

- **Calcolo dell'Esperienza Necessaria**: ExperienceForNextLevel calcola la quantità di esperienza necessaria per raggiungere il prossimo livello usando una formula logaritmica.

- **Gestione delle Nuove Partite**:
  - NewGame: Imposta il livello iniziale e i punti esperienza, e inizializza le risorse e gli edifici per una nuova partita.

- **Caricamento dei Dati del Giocatore**:
  - LoadPlayerData: Carica i dati salvati, inclusi edifici e truppe, e li posiziona nella griglia di gioco. Gestisce anche i limiti della griglia e il caricamento delle truppe nei campi di addestramento.

- **Caricamento delle Truppe**: LoadTroopsForTrainingBase gestisce il caricamento e la posizione delle truppe nei campi di addestramento basandosi sui dati salvati.

- **Salvataggio dei Dati**: OnApplicationQuit salva tutti i dati del giocatore, inclusi livello, esperienza, risorse, e edifici, quando l'applicazione viene chiusa.

5. Assets utilizzati

Tutti gli assets relativi alla UI sono stati disegnati da me, altri sono stati scaricati da Internet.
Gli assets relativi agli edifici sono stati presi dallo Unity Assets Store.

6. Sviluppi futuri

Livelli degli Edifici e Aggiornamenti

In una futura implementazione, gli edifici del gioco potranno essere aggiornati attraverso un sistema di livelli. Ogni edificio avrà un livello iniziale e potrà essere potenziato per migliorare le sue caratteristiche e abilità.

- **Livelli degli Edifici:** Ogni edificio avrà un livello che può essere incrementato tramite aggiornamenti. I livelli superiori offriranno vantaggi come una maggiore capacità, miglioramenti nella difesa, o abilità aggiuntive.

- **Sistema di Aggiornamento:** Gli aggiornamenti richiederanno risorse specifiche come Oro ed Elisir. L'utente potrà scegliere di aggiornare gli edifici tramite un'interfaccia dedicata che mostrerà il costo e i benefici dell'aggiornamento.

- **Visualizzazione:** L'aspetto visivo dell'edificio cambierà con ogni livello, mostrando miglioramenti visibili come nuove strutture o modifiche estetiche.

- **Gestione Dati:** I dati relativi ai livelli degli edifici e ai progressi di aggiornamento saranno salvati nel sistema di salvataggio del gioco (ad es. PlayerPrefsController) per garantire la persistenza tra le sessioni di gioco.

Miglioramento del sistema di battaglia

Il sistema di battaglia verrà migliorato per offrire una maggiore strategia e controllo all'utente. Gli utenti potranno ora decidere quante e quali truppe portare in battaglia e scegliere le loro posizioni.

- **Selezione delle Truppe:** Gli utenti avranno la possibilità di selezionare il tipo e la quantità di truppe da inviare in battaglia. Sarà implementato un sistema di selezione visiva e un'interfaccia utente per facilitare questa scelta.

- **Posizionamento delle Truppe:** Prima dell'inizio della battaglia, gli utenti potranno posizionare le loro truppe in punti strategici sul campo di battaglia. Questo permetterà una pianificazione tattica e influenzerà l'esito della battaglia.

Modalità Multiplayer

Il gioco introdurrà una modalità multiplayer che permette agli utenti di sfidare i villaggi di amici in battaglie competitive.

- **Modalità Multiplayer:** Gli utenti potranno accedere a una modalità multiplayer in cui potranno sfidare altri giocatori in tempo reale o a turni. Questa modalità includerà opzioni per creare o unirsi a leghe e tornei.

- **Scontri tra Villaggi di Amici:** Gli utenti potranno inviare richieste di battaglia agli amici e sfidare i loro villaggi. Sarà possibile vedere e confrontare le statistiche del proprio villaggio con quello degli amici.

- **Sistema di Inviti:** Sarà implementato un sistema di inviti per permettere agli utenti di invitare amici a battaglie o alle leghe. Gli inviti possono essere inviati tramite una lista amici integrata.

- **Gestione Dati Multiplayer:** Saranno gestiti e salvati i dati delle battaglie multiplayer, delle sfide tra amici e delle classifiche per garantire un'esperienza fluida e coerente.

7. Documentazione Doxygen

Doxygen è uno strumento di documentazione per il codice sorgente che genera documentazione automatica a partire dai commenti del codice.

Per accedere alla documentazione generata da Doxygen per il progetto **"Enchanted Village"**, seguire questi passaggi:

1. **Avviare il Servizio Doxygen:** Assicurati di avere Docker installato sul tuo sistema. Nella cartella principale del progetto, esegui il comando seguente per avviare il servizio Doxygen in un container Docker:

   docker-compose up -d

2. **Accedere alla Documentazione:** Una volta avviato il servizio, apri un browser web e naviga all'indirizzo:

   http://localhost:7000/enchanted-village-client/docs/

Questo indirizzo ti porterà alla documentazione generata da Doxygen, dove potrai esplorare la struttura del codice, le classi, i metodi e le loro descrizioni.

Nota: Se è la prima volta che esegui il container, potrebbe essere necessario attendere alcuni minuti affinché la documentazione venga completamente generata e resa disponibile.

8. Repository Github

Il repository dell'intero progetto è disponibile al seguente url:

https://github.com/ila13-code/enchanted-village-client