CAPITOLO 1

ANALISI

1.1 REQUISITI

Il software "TowerDefenseGame" è un gioco 2D del genere tower defense, sulla falsa

riga di "Age Of War 2".

Lo scopo principale del gioco è quello di bloccare l'avanzata di una schiera di nemici, i

quali non devono arrivare fino alla propria base, tramite lo schieramento di personaggi

alleati.

Requisiti funzionali:

-Il giocatore dovrà essere in grado di generare degli alleati di diverse categorie che

difenderanno la propria base dagli aggressori

-I nemici avanzeranno fino a che non incontreranno un alleato, a quel punto quello dei

due che perde despawnerà, mentre l'altro perderà dei punti salute

-Il gioco progredisce ad ondate sempre più numerose e potenti

-Si guadagnano monete necessarie allo spawn degli alleati sia con il passare del tempo

sia con l'uccisione dei nemici

-Se un nemico riesce ad arrivare fino alla base essa inizierà a perdere vita finchè un

alleato non sconfiggerà il nemico in questione

-Se i punti salute della base scendono a zero la partita finisce

-Sarà possibile attivare una funzionalità speciale molto potente ma con un alto tempo di

ricarica

-Una volta finita la partita all'utente sarà possibile effettuare un salvataggio del proprio

punteggio

-Una classifica dei punteggi migliori può essere consultata dal menù iniziale

CAPITOLO 2

DESIGN

2.1 ARCHITETTURA

Come architettura per il software è stato scelto l'MVC.

La parte della "view" è affidata all'interfaccia Panel, responsabile di mostrare a schermo

tutti gli avvenimenti e di catturare gli input dell'utente.

Il model invece è affidato al'interfaccia GameLogic, la quale si occupa di gestire la

logica di base del programma e le varie entità.

Il gameloop invece è la sezione di programma che si occupa di aggiornare ciclicamente

tutte le altre parti ed è la principale connessione tra view e model.

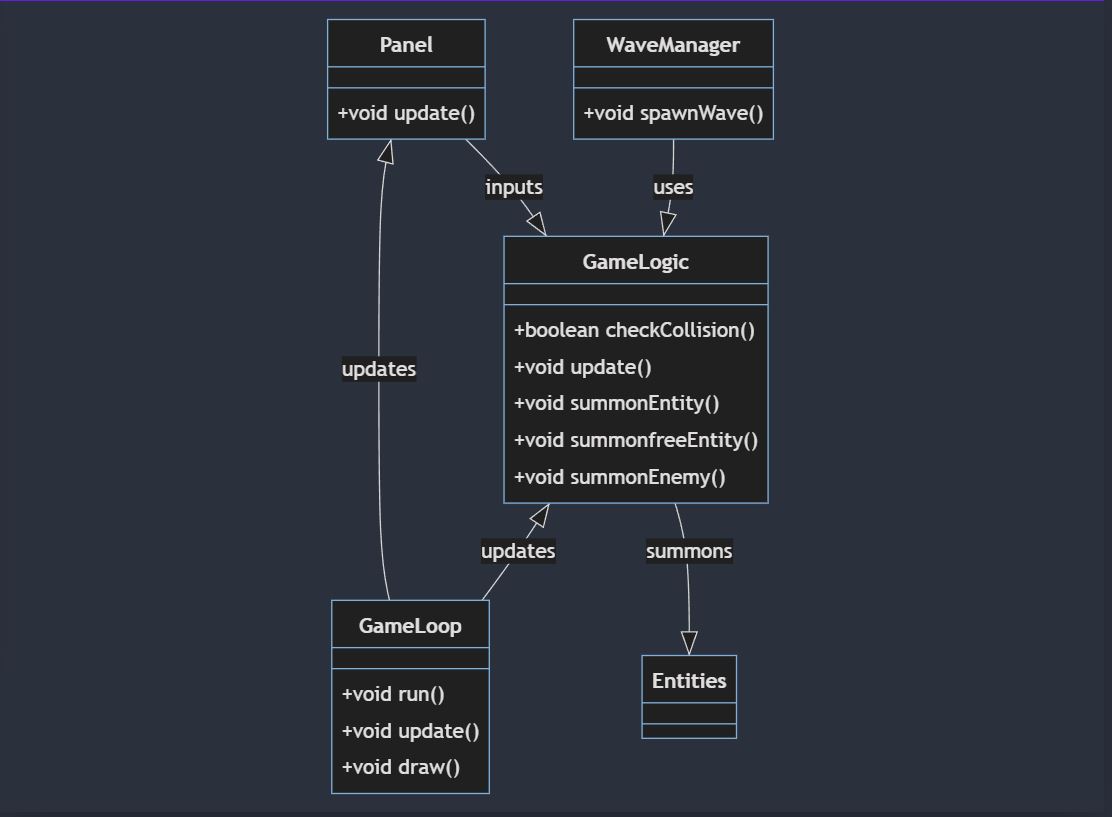
Se si dovesse cambiare in blocco la libreria grafica non ci sarebbero grosse

ripercussioni sul programma, idem per inserire nuovi "potenziamenti".

Per aggiungere delle entità (in particolare per i nemici) invece andrebbe modificato

l'algoritmo dello spawnmanager

Un semplice UML che descrive queste relazioni:



GUIDA UTENTE

All'avvio del gioco sarà possibile scegliere tra 3 menù: "Start Game", "Best Scores" e

"Rules"

Cliccando su "Start Same" il gioco comincerà, cliccando su "Best Scores" si aprirà un

file contenente i punteggi migliori salvati mentre cliccando su "Rules" si aprirà una

finestra con il regolamento del gioco (da cui è possibile tornare indietro con il pulsante

"Back").

Una volta cominciato il gioco inizieranno ad arrivare ondate di nemici, per poterli

sconfiggere bisognerà evocare alleati e/o potenziamenti tramite i pulsanti trovati in cima

alla finestra di gioco, al costo di monete di gioco.

Attualmente le unità/potenziamenti presenti sono:

-Barbaro: unità lenta, con molti punti salute e un discreto attacco

-Cavaliere: unità veloce, con pochi punti salute ma molti danni

-Arcere: unità veloce, con pochissimi punti salute ma molti danni e una gittata media

-Torretta: cannone posto sulla cima della torre, è un'unità monouso con gittata molto

ampia (se dei nemici arrivano alla torre colpiranno prima la torretta fino a

distruggerla, poi la torre)

-Potenziamento: chiamato "Double Allies" permetterà di evocare il doppio degli

alleati al costo di uno solo per un certo lasso di tempo

Una volta evocata un'unità bisognerà aspettare qualche secondo prima di poterne

evocare un'altra

Il potenziamento ha invece un tempo di ricarica molto più lungo

La partita finirà quando i punti salute della torre scendono a zero oppure cliccando il

pulsante "Surrender", a quel punto si verrà reindirizzati al menù di fine partita.

Da qui si potrà salvare il proprio punteggio inserendo il proprio nome e poi cliccando

"Save Score" e/o si potrà chiudere il gioco