

AirLine Traffic Simulator

Relazione di progetto
Programmazione ad Oggetti

Severi Andrea
Foschi Andrea
Rodillosso Daniel

Giugno 2021

Indice

1	Analisi	2
1.1	Requisiti	2
1.2	Analisi e modello del dominio	3
2	Design	5
2.1	Architettura	5
2.2	Design dettagliato	5
3	Sviluppo	6
3.1	Testing automatizzato	6
3.2	Metodologia di lavoro	6
3.3	Note di sviluppo	6
4	Commenti finali	7
4.1	Autovalutazione e lavori futuri	7
4.2	Difficoltà incontrate e commenti per i docenti	7
5	Guida utente	8

Capitolo 1

Analisi

Il gruppo si pone come obiettivo quello di realizzare un videogioco sulla gestione del traffico aereo, ispirandosi al gioco mobile “Flight Control”.

L’obiettivo del giocatore sarà quello di far atterrare il maggior numero di aerei che compariranno progressivamente sulla mappa, evitando di farli collidere tra di loro (causando la fine del gioco).

Il giocatore gestirà personalmente la direzione di ogni velivolo, disegnandone il percorso che dovrà seguire ciascun aereo, con il mouse.

La difficoltà di gioco aumenterà con l’aumentare del numero di aerei che saranno fatti atterrare.

1.1 Requisiti

Funzionalità necessarie:

- Disegnare correttamente con il mouse il percorso che l’utente vuol far seguire al velivolo selezionato.
- Implementare un’intelligenza artificiale, che muoverà gli aerei quando non avranno una destinazione scelta dall’utente (l’aereo una volta entrato nella mappa non potrà uscire, ma dovrà continuare a volare all’interno di essa).
- Realizzare una gestione efficiente degli aerei in entrata sulla mappa e il relativo atterraggio quando raggiungeranno la pista di atterraggio.
- Salvataggio degli score dei vari utenti in un file apposito, che verrà reso disponibile ad ogni avvio.
- Gestione della difficoltà crescente durante la partita.

Funzionalità opzionali:

- Creazione e aggiunta di mappe dinamiche (es: implementazione di oggetti che causeranno la distruzione dell'aereo se sorvolati, animazioni dinamiche nella mappa).
- Implementazione suoni di gioco.
- Aerei speciali, con velocità diverse.
- Gestione del vento: questo causerà una maggiore o minore velocità degli aerei durante la partita.

Challenge principali:

- “Fluidità” delle animazioni (tracciamento della rotta, movimento degli aerei, atterraggio degli aerei, collisione tra gli aerei).
- Corretta implementazione del pattern MVC.
- Gestione della difficoltà (crescente durante la partita).

1.2 Analisi e modello del dominio

AirLine Traffic Simulator dovrà essere in grado di gestire lo spawn (Spawn) degli aerei (Plane) e la loro entrata in partita dopo un quantitativo di tempo prefissato.

È presente una **Mappa di gioco** (Seaside) a tema aeroporto.

All'inizio della **Partita** (Game) verranno creati e appariranno in maniera sequenziale gli **Aerei** sulla mappa, che seguiranno un **percorso casuale** (gli Aerei ancora non possiedono una rotta specifica da seguire ma semplicemente si muovono nella Mappa).

In questo caso quando un aereo non possiede una rotta specifica dovrà muoversi verso una direzione indicata dall'**algoritmo di I.A.**

Compito dell'utente è quello di far atterrare gli Aerei sulle **Piste di Atterraggio** (AirStrip) presenti sulla mappa.

Gli Aerei presenti nel gioco potranno essere di più tipologie (PlaneType), aventi velocità diverse.

Un Aereo potrà atterrare quando sarà in prossimità dell'area di atterraggio, resa disponibile dalla Pista stessa.

Sarà quest'ultima (AirStrip) a convalidare la possibilità o meno di fare atterrare il velivolo in caso di contatto con l'area di atterraggio.

Per ogni Aereo che riuscirà ad atterrare correttamente, il Giocatore (User) incrementerà il proprio **Punteggio** con un valore deciso a priori.

Il **Giocatore** potrà eseguire le seguenti operazioni sull'Aereo che vuole muovere:

- selezionare l'Aereo che vuole controllare.
- disegnare il percorso (Path) che poi l'Aereo dovrà correttamente seguire.

È stato pensato di implementare degli indicatori all'interno dell'area di gioco per segnalare gli aerei in arrivo da fuori dalla Mappa.

Questi segnali scompariranno non appena l'aereo entrerà in partita.

Il giocatore perde se:

- fa collidere gli aerei tra di loro in volo
- urta il bordo con un aereo presente in partita

In entrambi in casi, viene generata un'esplosione a video.

Nel tentativo di evitare ciò, allo scorrere del **Tempo di gioco**, altri aerei che dovranno essere fatti atterrare, compariranno sulla Mappa, con maggiore frequenza in base al punteggio.

È possibile mettere anche in **pausa** la partita tramite pulsante dall'interfaccia di gioco.

Alla fine della Partita, il Giocatore avrà ottenuto un punteggio finale, che sarà salvato su **file**. Se sarà sufficiente per entrare in **Classifica**, verrà inserito in quest'ultima.

Dal **Menù Principale**, oltre che iniziare una nuova partita, si potrà visualizzare tale classifica.

Capitolo 2

Design

2.1 Architettura

2.2 Design dettagliato

Capitolo 3

Sviluppo

3.1 Testing automatizzato

3.2 Metodologia di lavoro

3.3 Note di sviluppo

Capitolo 4

Commenti finali

4.1 Autovalutazione e lavori futuri

4.2 Difficoltà incontrate e commenti per i docenti

Capitolo 5

Guida utente