

Gianmarco Pettenuzzo

Matricola: 1097856

Progetto di Mobile Programming & Multimedia

CUBIK

Anno accademico 2017-18

1) INTRODUZIONE:

Il progetto consiste nello sviluppo di un'applicazione per Android tramite il framework crossplatform Corona SDK. L'applicazione comprende un semplice gioco Endless-Runner. Ho voluto creare un progetto che utilizza gesture anche in contemporanea. La difficoltà del gioco sta infatti nel coordinare, sempre più velocemente, i movimenti da fare su schermo.

2) PANORAMICA DEL GIOCO:

Il gioco consiste nell' evitare degli ostacoli, con un quadrato, in modo da ottenere un punteggio più alto possibile.

Lo scopo del progetto è stato quello di utilizzare gesture al fine di saltare, eliminare o distruggere i vari ostacoli che si presentano.

Ad ogni livello, la difficoltà del gioco aumenta creando nuovi tipi di ostacoli (eliminabili tramite apposite gesture) e aumentando la velocità.

Il gioco, come anticipato, non ha fine ma avrà un termine una volta che il player toccherà un nemico.

Un livello del gioco dura all'incirca 25 secondi (500 score).

3) STRUMENTI UTILIZZATI

La scelta del Framework da utilizzare è caduta su Corona SDK, in quanto, oltre ad essere specializzato nello sviluppo di giochi, offre, una propria libreria per la fisica e un linguaggio particolarmente intuitivo e leggero.

Inoltre, il simulatore fornito dal Framework è senza dubbio la caratteristica principale del software.

Al momento del primo accesso, il software offre un semplice ed esaustivo tutorial, che permette di capire a pieno tutte le sue potenzialità.

Per l'editor delle immagini utilizzate nel gioco ho utilizzato il software gratuito GIMP (GNU image Manipulation Program). Ho usato lo stesso programma per creare lo sheet del pulsante volume.

Gli audio presenti nel gioco sono stati scaricati dal sito <http://www.soundimage.org>.

4) SCENE

4.1) MENU

La prima scena visualizzata all'apertura dell'applicazione è il Menu.

Da qui si può decidere:

- 1) Giocare, premendo sul pulsante "Gioca";
- 2) Visualizzare i migliori punteggi ottenuti, premendo su "Punteggi migliori";
- 3) Imparare le gesture da utilizzare, premendo su "Tutorial";
- 4) Disattivare o attivare il suono, premendo sull'apposita immagine.

All'entrata nel menu, viene riprodotta in loop un audio. Il suddetto audio si blocca soltanto una volta entrato nel gioco, oppure premendo sul pulsante "disattiva suono".

L'audio, quindi resta in riproduzione se si decide di entrare in "Punteggi migliori" o in "Tutorial".

4.2) HIGHSCORES

Si accede a questa scena direttamente dal menu, oppure dal gioco quando il player viene sconfitto.

Questa scena fa visualizzare i punteggi migliori ottenuti dall'utente.

Per salvarle i punteggi viene utilizzata la libreria di json salvando i risultati sul file score.json.

Una volta entrati in Highscore, si può solo andare nella scena Menu.

4.3) TUTORIAL

Si accede a questa scena direttamente dal menu.

Presenta una tabella riassuntiva dei nemici presenti nel gioco e la relativa gesture da utilizzare.

4.4) GAME

Si accede a questa scena direttamente dal menu.

Le gesture presenti nel gioco sono:

- TAP su player: per farlo saltare e schivare i triangoli;
- DOPPIO-TAP su fantasma: con questa gesture si elimina il fantasma (presente solo dal livello 2). C'è da annotare che questa gesture non è implementata a dovere su Corona SDK, in quanto, la maggior parte delle volte, legge prima il primo tap eseguendo un evento "tap", e poi il secondo tap eseguendo un evento "multitouch" (quindi in totale legge 3 tap).
A questo problema, purtroppo, non ho trovato soluzione, ma ho "raggirato" il problema inserendo un campo fantasma.ghostHeart=2, così, ad ogni tap diminuire questo valore, ed eliminare il fantasma solo quando arriva a 0;
- SHAKE: con questa gesture si distrugge il meteorite (presente solo dal livello 3).
- SWIPE-UP su tutto lo schermo: con questa gesture si effettua un doppio-salto, solo se, durante il gioco, il player ha preso un powerup.

Una volta entrato nel gioco viene riprodotto un audio diverso da quello di menu.

5) Grafica

La grafica delle prime 3 scene descritte precedentemente è molto semplice e intuitiva. Per la grafica della scena Game, invece è leggermente più complicata, in quanto ho voluto inserire due immagini background differenti.

Le immagini in background si muovono con velocità differente, in modo da dare un effetto di movimento.

Una volta che il player riesce a raggiungere i 1500 score, lo scenario cambia, e da un effetto chiaro (ma comunque bilanciato) del primo scenario, si passa ad un scenario scuro del secondo scenario.

Nella scena menu, invece, premendo sul quadrato è possibile cambiare provvisoriamente il colore del background.

Ogni scena, per la parte grafica, utilizza la suddivisione in gruppi offerta da Corona SDK. Ogni gruppo viene inserito nel gruppo principale della scena.

6) Commenti su Corona SDK

La maggior parte degli aspetti positivi di Corona SDK riguardano il linguaggio estremamente semplice, leggero ed intuitivo.

Anche la libreria fisica, seppur non abbondantemente utilizzata in questo progetto, è immediata e molto ben fatta.

Grande merito ha anche la community.

Mi ha sorpreso positivamente anche il tutorial offerto per capire le funzionalità del software.

Per quanto riguardano gli aspetti negativi, ho trovato particolare difficoltà nel testare l'audio, in quanto il software (forse non aggiornato all'ultima versione) si arrestava molto frequentemente. Un altro aspetto negativo, è il fatto che, a cambio scena, non tutti gli elementi si eliminavano automaticamente, ma dovevo procedere manualmente con il comando `display.remove()`.

Un altro problema riscontrato riguarda il simultore, in quanto alcuni problemi non vengono riscontrati, ma su device si.

Come espresso precedentemente, non tutti gli eventi/gesture sono implementate correttamente.

Nel complesso, per lo sviluppo di giochi mobile, consiglio l'utilizzo di corona.