

# Relazione

## Colour Square

Corso di Programmazione a oggetti 2014/15

Luca Genghini

Matteo Sertori

27 febbraio 2015

# Indice

## Analisi

1.1 Requisiti.....	3
1.2 Problema.....	3

## Design

2.1 Architettura.....	4
2.2 Design Dettagliato.....	6

## Sviluppo

3.1 Testing.....	7
3.2 Divisione dei compiti e metodologia di lavoro.....	7

## Commenti Finali

4.1 Conclusioni e lavori futuri.....	8
--------------------------------------	---

# Analisi

## 1.1 Requisiti

Il Software che intendiamo realizzare mira alla creazione di un videogioco a scorrimento orizzontale di nome "Colour Square". Il personaggio principale sarà impersonato da un quadrato, il quale dovrà superare i vari ostacoli che troverà sul suo cammino.

Gli ostacoli e il personaggio saranno caratterizzati da 4 specifici colori (Blu, Rosso, Verde e Giallo). L'utente per poter superare gli ostacoli dovrà far assumere al personaggio il medesimo colore dei vari ostacoli che si presenteranno davanti.

Il videogioco sarà suddiviso in 3 livelli con crescente difficoltà; al termine di ciascun livello il personaggio sarà direttamente portato al livello successivo.

## 1.2 Problema

Il problema principale è quello di creare le collisioni del personaggio con i vari elementi della mappa. In particolar modo la tempistica dei confronti deve essere effettuata precisamente nell'istante in cui il personaggio raggiungerà un ostacolo.

Un altro problema è quello della creazione della classe che leggerà la mappa e il modo per farla scorrere e far muovere simultaneamente il personaggio.

Il cambio colore del personaggio sarà una parte fondamentale del lavoro.

Per rendere il gioco più accattivante si dovrà inserire il sonoro. Questo punto non si realizzerà immediatamente ma in futuro con un nuovo aggiornamento.



## MODEL

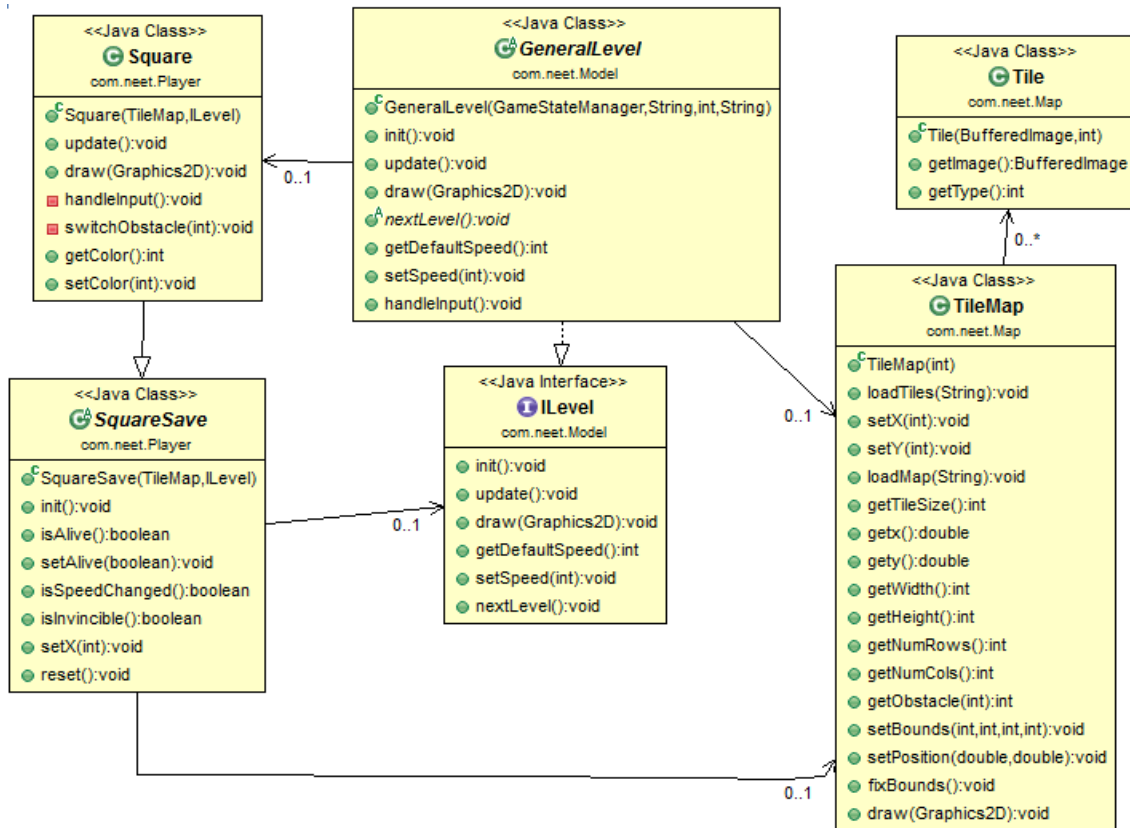


Figura 2.2: Model.

## CONTROLLER

Il Controller si occupa di verificare le pressione e il rilascio dei tasti attraverso la classe Keys che viene utilizzato in modo statico.

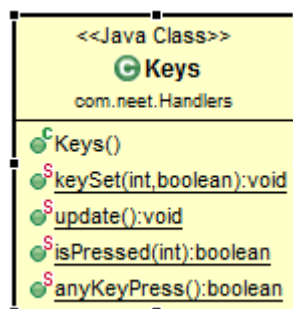


Figura 2.3: Controller.

## 2.2 Design dettagliato

### I Package

- ❖ **Handlers:** Gestione dei tasti premuti e rilasciati.
- ❖ **Main:** Creazione della schermata di gioco e della gestione del thread
- ❖ **Map:** Creazione e gestione della mappa di gioco
- ❖ **Model:** Gestione degli stati del gioco (Menù, Livelli e Fine gioco), e view.
- ❖ **Player:** Gestione del personaggio.

### Gestione del Personaggio

Quando l'utente premerà i tasti (Q, W, E, R), il personaggio dovrà cambiare colore. Si utilizza un metodo che controlla la pressione dei tasti, e una volta spinto il tasto corrispondente a un colore, cambia l'immagine corrente disegnata sullo schermo con quella del colore spinto. Grazie ad un codice che è stato assegnato ad ogni colore, si potrà gestire la collisione con i vari elementi della mappa (muniti anch'essi di un proprio codice).

### Bonus

Ci sono due tipi di bonus: l'invincibilità e il rallentamento. Il primo è rappresentato da uno scudo che può essere rappresentato con uno dei quattro colori caratteristici del gioco; il secondo invece da una freccia anch'essa colorata. Per poterli ottenere ovviamente si dovrà assumere al quadrato lo stesso colore del bonus; se questo accade verranno effettuate modifiche sulla velocità della telecamera di gioco o sul immagine del player.

### Mappa di gioco

La mappa di gioco è costituita da un file di testo contenente una matrice di interi. La classe prenderà in input il path di destinazione della mappa e la leggerà casella per casella. Grazie a un Tileset (un'immagine) la mappa legge i numeri e disegna gli ostacoli o i bonus corrispondenti al codice del tileset.

# Sviluppo

## 3.1 Testing

L'applicazione è stata testata su vari pc:

- ❖ Asus A56C Intel Core i7, Ram 4 GB, SO Windows 8.1
- ❖ Asus F550C Intel Core i7, Ram 4GB, SO Windows 8.1
- ❖ MacBook Intel Core i5 2,6GHz, 16GB RAM, SO Mac OS X Yosemite

Si è inoltre utilizzato il test JUnit per verificare se le collisioni venivano effettuate correttamente.

## 3.2 Divisione dei compiti e metodologia di lavoro

I compiti sono stati divisi in questo modo:

Matteo Sertori:

- ❖ Creazione della mappa e del suo caricamento
- ❖ Creazione del pannello di gioco
- ❖ Gestione degli stati del gioco

Luca Genghini:

- ❖ Gestione del personaggio: Collisioni, potenziamenti e gestione del cambio colore
- ❖ Creazione dei livelli
- ❖ Gestione della pressione dei tasti (classe Keys).

# Commenti finali

## 4.1 Conclusioni e lavori futuri

Il progetto è stato molto impegnativo, soprattutto la prima parte dove ogni componente del gruppo ha fatto ricerche sul web con lo scopo di imparare gli argomenti, necessari per lo sviluppo della applicazione, mai visti in precedenza. Nel complesso ci riteniamo abbastanza soddisfatti del lavoro svolto, reso possibile anche dalle nozioni imparate sia in classe che in laboratorio.

I punti di forza del gioco, pur essendo semplice come concetto, sono: la facilità di inserimento di nuovi livelli (grazie alla classe astratta GeneralLevel), la facilità di gioco e il coinvolgimento all'interno di esso.

I punti deboli sono l'inesperienza da parte nostra di creare un gioco; questa applicazione offre una esperienza di gioco limitata e la mancanza di animazioni sonore e grafiche sono un punto a sfavore di questo gioco.

In futuro provvederemo all'inserimento delle suddette animazioni sonore e grafiche, l'inserimento di nuove mappe e la creazione di una applicazione Android.