

Peer-Review 1: UML

Lorenzo Giarnese, Mattia Laviola, Alessia Abbondanza
Gruppo 19

1 aprile 2022

Valutazione del diagramma UML delle classi del gruppo 18.

1 Lati positivi

Secondo noi, alcuni lati positivi del progetto UML potrebbero essere:

- Buon utilizzo del pattern Observer;
- Buon utilizzo del pattern State;
- Utilizzo di enum per rappresentare il colore degli studenti e dei professori;
- Salvataggio studenti nel sacchetto per le estrazioni per colore.

2 Lati negativi

Secondo noi, alcuni lati negativi potrebbero essere:

- Classi "pesanti" nel Model: per esempio la classe GameModel potrebbe essere divisa in più classi, tenendo conto della differenza tra la plancia di gioco (insieme di isole) e l'istanza di gioco per sé. Considerando una classe GameBoard per la gestione delle isole (vista come astrazione) e una classe GameModel per l'istanza di gioco. Tenendo delle classi troppo unite si rischia di avere difficoltà mentre si programma, garantendo una minore usabilità delle classi e meno ordine a livello di codice;

- Cattivo incapsulamento: deriva dal fatto di avere classi dense. Infatti è possibile, per esempio, accedere a determinate variabili, anche usando metodi di classi che teoricamente non potrebbero avere accesso;
- Manca classe per gestire la scuola: si può pensare ad una classe che gestisca la scuola, ovvero la plancia assegnata ad ogni giocatore;
- Nella carta personaggio è presente solo il costo, ma siccome ci sono personaggio che hanno lo stesso costo viene difficile distinguere un personaggio da un altro di stesso costo. Si potrebbe pensare ad un ID o un codice;
- La classe Professor invoca il metodo goToSchool(), ma non è presente nessuna variabile che tenga conto della scuola;
- Usando un ArrayList di isole, quando si fa il merge delle isole, è necessario shiftare a mano la posizione delle rimanenti isole. Una soluzione più comoda potrebbe essere una LinkedList;
- HashMap per studentsBag, al posto di EnumMap che è la versione ottimizzata apposta per le enumerazioni.
- Non si tiene conto del numero di monete finito.

3 Confronto tra le architetture

Confronto tra le architetture e alcuni spunti che potremmo prendere per continuare a sviluppare il nostro progetto:

- Interessante è il fatto che si salva il fatto che il player utilizza la GUI o la CLI;
- Package View più sviluppato rispetto al nostro, con utilizzo del pattern Observer;
- Salvataggio studenti nel sacchetto per le estrazioni per colore, noi abbiamo salvato gli studenti come una variabile che decrementiamo, senza discriminare tra il colore, quindi senza riuscire a controllare se effettivamente ci sono ancora studenti di quel colore.