

Peer-Review 1: UML

Edoardo Giudici, Isabella Guglielmelli, Matteo Huang
Gruppo 50

5 aprile 2022

Valutazione del diagramma UML delle classi del gruppo 49.

1 Lati positivi

L'utilizzo della classe "Deck" per la gestione della mano del player permette di controllare facilmente quali carte sono state utilizzate e quali sono ancora inutilizzate. Inoltre la classe "GameClass" sembra ben strutturata per poter supportare il pattern "Model-View-Controller" rendendo disponibili un numero sufficiente di metodi per l'interazione con il Controller.

2 Lati negativi

Osservando l'UML la prima cosa che si può notare è la scarsa leggibilità delle relazioni fra classi. Inoltre non sono stati utilizzati pattern e risulta poco chiara la gestione della scelta della modalità di gioco. Nella classe "EffectCard" il metodo "AddStudent" sarebbe potenzialmente implementato in ogni istanza di "EffectCard", inutilmente, in quanto non tutte le carte personaggio necessitano del metodo "AddStudent". Il pattern strategy, inoltre, faciliterebbe la scelta implementativa dell'effetto della carta personaggio, in modo da affidare al pattern la scelta della giusta implementazione, evitando il problema di creare un singolo metodo che gestisca i 12 effetti.

3 Confronto tra le architetture

È interessante la scelta implementativa della Sala, pensata come un attributo "tables" di tipo HashMap all'interno della classe Dashboard, in quanto l'utilizzo di tale struttura dati permette di ridurre la complessità del codice. Nel nostro modello abbiamo invece optato per una classe "DiningTable" per rappresentare un tavolo degli studenti, in modo che la Sala sia modellizzata come un array di tavoli.