

Peer-Review 1: UML

Mazza, Moneta, Pizzocolo
Gruppo 10

2 aprile 2022

Valutazione del diagramma UML delle classi del gruppo 9.

1 Lati positivi

Un lato positivo dell'UML è sicuramente la divisione del Model in sotto-package. In particolare abbiamo trovato molto valido il pacchetto Board che comprende la scuola, le isole e le nuvole.

Altri aspetti consistenti sono le variabili 'currentHelper' e 'currentPlayer', infatti, troviamo sia utile sapere qual è la carta scelta da ogni giocatore per capire chi debba iniziare il turno dopo.

Abbiamo trovato interessante anche l'approccio alternativo tra Model-View-Controller con la View che non osserva direttamente il Model ma tramite il Controller, poiché ci sembra più facile da implementare, ma (vedi sotto) crediamo ci siano anche dei contro.

Guardando il diagramma, inoltre, abbiamo apprezzato come la classe enumerativa 'Team' venga usata sia per le modalità a 2/3 giocatori che per quella a squadre.

Infine, è stato stimolante vedere l'utilizzo di un factory pattern nella parte delle carte.

2 Lati negativi

Riprendendo dalla sezione sopra, parlando del pattern MVC, crediamo possa essere rischioso questo approccio poiché la View rischia di non riflettere il Model.

Non risulta chiaro il modo in cui sono stati trattati gli studenti, dal diagramma sembrerebbe che siano stati istanziati 130 studenti singolarmente, cosa che non ci sembra molto efficiente.

Parlando delle 'CloudTile' ci siamo accorti che nel metodo 'withdraw()' vengono inizializzati 3 studenti, ma nella versione del gioco a 3 giocatori gli studenti sono 4. Inoltre manca un metodo di controllo per capire se le nuvole non hanno più gli studenti sopra (quindi, ad esempio, qualcuno ha già pescato da quella nuvola).

Un altro lato negativo con dei numeri che non ci tornano, è sulle 'IslandTile', dato che inizialmente vanno da 1...10 e successivamente (nella classe 'CompoundIslandTile') diventano un array da 1...9.

Parlando di efficienza, abbiamo trovato in alcune classi, ad esempio in 'Player' o in 'SchoolDashboard', tanti metodi legati alla modalità esperto. Però, in una partita in modalità base, tutti quei metodi non saranno mai utilizzati, il che non è funzionale.

Un ultimo consiglio, parlando della classe 'CompoundIslandTile', ci sembra sia utile sapere qual è il numero delle torri sulle isole aggregate.

3 Confronto tra le architetture

Confrontando la due architetture abbiamo trovato molto interessante la sezione della board, per l'ordine che crea nell'UML. Infatti molto probabilmente prenderemo come spunto quest'organizzazione, togliendo però la parte della dashboard.

Un'altra cosa che manca al nostro UML che invece ci è sembrato utile è l'utilizzo delle classi enum per i vari colori degli studenti o delle torri.

In conclusione, la divisione tra IslandTile e CompoundIslandTile per indicare le isole singole o gli aggregati di isole, troviamo sia una scelta molto valida che potremmo fare anche sul nostro diagramma.