

Peer-Review 1: UML

Leonardo Marazzi, Alessio Mangiarotti, YanFeng Li

Gruppo 5

Valutazione del diagramma UML delle classi del gruppo 4.

Lati positivi

L'UML del codice del modello è ben strutturato e sintetico; il progetto infatti non contiene classi ripetute o inutilizzate.

La logica del modello rispecchia le dinamiche di gioco interpretate in modo puntale dalla classe Game che divide in modo preciso i turni e le sue fasi.

La gestione dei personaggi tramite Interface riesce a gestire in modo corretto nonostante il pattern strategy avrebbe semplificato ancora di più la logica di implementazione.

In aggiunta l'uso delle classi Enumerazione toglie complessità al codice rimuovendo inutili ripetizioni.

Lati negativi

Collegamenti tra classi incorretti

Nonostante le classi siano ben rappresentative delle dinamiche di gioco i collegamenti tra di esse sono scorretti.

Le isole mancano di connessione con Player che ne gestisce la dinamica e la classe Game, in oltre, dovrebbe essere collegata alla classe Player e non all'interfaccia Character.

Classi eccessivamente cariche di metodi

Riteniamo che molte delle classi presentino un numero elevato di metodi, pensiamo che sarebbe stato opportuno avere un numero maggiore di classi più specifiche per rappresentare meglio i vari aspetti e fasi del gioco.

Sarebbe stato meglio aggiungere ad esempio una classe per gestire il tavolo di gioco e un'altra per gestire le nuvole invece di accorparle nella classe Game.

Tower

Nonostante venga gestita in modo corretto, sarebbe meglio attribuire alla torri un colore in modo da differenziarle, altrimenti incorporare tutto nelle logiche di implementazione può portare ad errori.

Character

Usare il pattern Interface è abbastanza corretto però riteniamo che l'implementazione con strategy sia più corretta.

Confronto tra le architetture

Riteniamo che l'implementazione della classe Game sia più specifica di quella da noi proposta, sebbene riteniamo essere un compito del controller e non del modello.

L'implementazione delle isole è fatta bene anche se un po' complicata rispetto a gestirla con una classe con attributo size, che viene incrementata ogni volta che due isole si uniscano (eliminando l'isola in eccesso).

Il progetto in oltre presenta un grande numero di classi enumerazione che va a rimuovere informazioni ripetute nelle singole istanze delle classi, cosa che invece il nostro modello non considera.