

Peer-Review 1: UML

Mirko Pica, Riccardo Pianalto, Christian Prendin, Aldo Pinelli

Gruppo GC01

Valutazione del diagramma UML delle classi del gruppo GC48.

Lati positivi

È stato fatto un lavoro eccellente a nostro avviso, il design è stato fatto bene e con una buona leggibilità anche grazie ai colori usati. La struttura è chiara anche senza leggere i commenti, dimostrazione di un buon design. Buon utilizzo degli strategy pattern per le condition nelle objective card, e buone anche le idee di usare una waiting room (per gestire la creazione di una partita), una scoreboard (per gestire i punti dei giocatori) e una station (per gestire il campo da gioco).

Lati negativi

Ci è stato difficile trovare lati fortemente negativi o addirittura sbagliati.

Ci sentiamo però di consigliare la modifica di alcuni metodi. All'interno del player e della station i metodi per il piazzamento delle carte prendono in ingresso quattro parametri e questo potrebbe portare a difficoltà nella fase di testing (e non solo), quindi semplificare il metodo sarebbe la scelta migliore. (Ad esempio si potrebbero spostare i metodi playCard e playsAllowed all'interno della classe PlayingCard, il giocatore nella mano ha le sue carte e prima di giocarle potrà fare carta.isPlayingAllowed() e nel caso questo sia positivo fare card.play()).

Cercare quindi di fare in modo che un metodo svolga uno e un solo compito e che non sia troppo complesso.

Confronto tra le architetture

Il punto di forza principale dell'architettura, è che l'UML è diviso in più classi per gestire in maniera efficiente dettagli su cui noi non ci siamo soffermati a sufficienza. Ad esempio, abbiamo notato che la "Waiting Room" è un ottimo modo per gestire la fase pre-partita che permette di attendere i giocatori senza ancora istanziare la vera e propria partita.

Non avevamo le idee molto chiare su come avremmo potuto riconoscere le disposizioni "stair" ed "L" nel campo da gioco per quanto riguarda le Objective Card, sicuramente prenderemo spunto dall'utilizzo della matrice per implementare quella che nell'UML è la classe "Station".