

Peer-Review 1: UML

Camilla Andiloro, Emanuele Cimino, Marco Crisafulli

Gruppo AM35

Valutazione del diagramma UML delle classi del gruppo AM08.

Lati positivi

- Metodi protected: il loro utilizzo rende il codice più solido e meno soggetto a errori dato il forte utilizzo degli array.
- Classe PawnPositon: questa classe rende comoda l'implementazione di tutte le azioni relative all'uso degli studenti durante il gioco.
- Corretto uso dei package: la divisione dei package rende l'UML molto ordinato.

Lati negativi

- Poca astrazione: poco utilizzo dei package nativi di Java (per esempio uso delle Collection). Le strutture dati presentate sono a basso livello, svantaggiando un'eventuale modifica futura.
- Poco chiara la divisione tra Match principiante, Match esterno e i Match a più giocatori. Le classi sono poche e molto grandi, ad esempio playerManager gestisce troppi casi, ovvero la gestione delle partite con un diverso numero di giocatori e la variante esperto potrebbe essere scorporati tramite l'utilizzo di un pattern per la gestione di più casi simili tra loro.
- Implementazione pattern strategy non chiara, discrepanza tra le signature del metodo useEffect.
- Scarso utilizzo di Design Pattern, consigliamo l'utilizzo di possibili pattern che potrebbero rendere la struttura del progetto più solida.

Confronto tra le architetture

La divisione dei package e il conseguente utilizzo dei metodi protected rende il codice meno esposto a futuri errori.

È una idea che pensiamo di poter inserire in maniera più consistente all'interno della nostra implementazione.