

# Peer-Review 1: UML

Federico Pizzi, Niccolò Maria Rizzi, Giacomo Savazzi  
Gruppo 53

4 aprile 2022

Valutazione del diagramma UML delle classi del gruppo 63.

## 1 Lati positivi

Il modello presenta tutte le classi necessarie per coprire interamente oggetti e entità del gioco permettendo quindi di svolgere in modo completo e coerente una partita.

Dal punto di vista della progettazione abbiamo apprezzato particolarmente l'aver individuato e raggruppato i personaggi per caratteristiche simili in quanto rende possibile un'efficiente applicazione di pattern: in questo caso ha portato all'applicazione di ben tre pattern all'interno del model.

## 2 Lati negativi

Le associazioni non risultano molto chiare (anche perchè non identificate da una label) e le relative cardinalità, quando presenti, non risultano sempre coerenti (ad esempio stando all'UML ogni plancia dovrebbe possedere un solo tavolo di studenti). Altre perplessità:

- Non è molto chiaro come venga gestita Madre Natura dalla funzione `moveMotherNature` (che non considera le mosse scelte dall'utente) e quale sia l'associazione con Isola se l'unico attributo che possiede è una posizione di un'isola.

- Cloud non viene associata a Game e questo porta ad alcuni dubbi sul comportamento di metodi come ad esempio `DrawStudentFromCloud()` (che non ne riceve in input)
- Viene utilizzata una strategy per la gestione del solo personaggio 1

Abbiamo poi trovato poco vantaggiosa la gestione delle torri come oggetti: l'isola può contenere torri di un solo colore per volta e il numero di torri che contiene è determinato univocamente da `size`. Ragionamento simile si può fare per le plance dei giocatori che possono contenere torri di un solo colore per tutta la partita; gestirle come oggetti richiede numerosi controlli che potrebbero essere evitati.

### 3 Confronto tra le architetture

Le due architetture sono molto simili come struttura generale e come classi principali presenti.

Una differenza sostanziale è nella gestione dei personaggi: in questo modello gli 8 personaggi presenti vengono realizzati come classi distinte, mentre noi ci siamo limitati alle singole classi comuni per personaggi generici e personaggi con la possibilità di contenere studenti.

A questi 8 personaggi vengono applicate tre strategy differenti: una per il calcolo dell'influenza e le altre due per lo spostamento di studenti. Di queste strategy nel nostro modello abbiamo individuato solo quella per la gestione dell'influenza e potrebbe essere vantaggioso implementare anche quella per lo spostamento degli studenti.