

Peer Review Gruppo PSP16

Lorenzo Motelli, Giovanni Moretti, Manuel Ianieri

3 aprile 2022

Cose positive

- UML ben strutturato, con già la divisione in package.
- Utilizzare la classe Map di java può risultare molto utile per identificare univocamente ogni professore e studente, ma anche complesso essendo una classe che viene utilizzata soprattutto per controlli su liste.
- Le classi sono ben strutturate, con pochi metodi essenziali.
- Buona implementazione generale dei pattern.
- Buono che è stato messo controller.

Cose negative

- Gestione personaggi non particolarmente chiara in questo UML iniziale.
- Ripetizione delle isole, GameTable e MotherNature hanno, presumibilmente, entrambi la stessa coppia di lista di isole presenti nel gioco.
- La DiningRoom ha sempre lo stesso numero di studenti massimi che può contenere, a questo punto dichiarare direttamente che sia una variabile di tipo final.
- A cosa servono i due metodi privati checkProfessorOwnership e getTowerZone?
- Perché solo un'enumerazione è stata collegata? E quelle non collegate sono enumerazioni vuote? A cosa servono? STATUS in particolare.
- Come viene implementato ExpertGame?

Confronto e conclusioni

La struttura generale del model è molto simile, con varie classi che gestiscono in ugual modo determinate parti (ad esempio Table è GameBoard, GeneralGame è Game). Le principali differenze sono sulla scelta di gestione delle diverse componenti del gioco, come gli studenti utilizzando liste o map, oppure madre natura come boolean o classe a sé stante. In conclusione per essere una versione iniziale, l'UML risulta dettagliato e ragionato. Per questo motivo saltano all'occhio problematiche di comprensione e/o di dettaglio. Con un confronto diretto molte di queste incomprensioni si sarebbero potute chiarire immediatamente, ma leggendo soltanto l'UML alcune scelte non sono state comprese. Nel complesso un buon lavoro.