**Peer Review – UML gruppo 67**

Questa è la valutazione peer to peer dell’UML del gruppo 67 (scaglione Sanpietro). In seguito vengono riportate alcune considerazioni che abbiamo fatto confrontando il lavoro dei nostri colleghi col nostro e, successivamente, esprimendo dei pareri sul lavoro a sé stante cercando di capire quali possano essere suoi eventuali punti di forza e sotto invece quali aspetti il progetto può mostrare delle debolezze strutturali e progettuali.

Si vuole precisare il fatto che, sotto anche indicazione dei componenti del gruppo 67 stesso, abbiamo tenuto conto che tale UML non è da considerarsi come definitivo e che alcuni aspetti sono tutt’ora oggetto di modifiche poiché, portando avanti il lavoro anche da un punto di vista di scrittura del codice, appare necessario apportare degli accorgimenti allo schema, che varia quindi di pari passo con l’avanzamento del progetto nel suo complesso.

**UML a confronto**

**Similarità:**

L’UML del gruppo 67 presenta alcune caratteristiche che lo avvicinano al nostro progetto.

Sicuramente ciò che accomuna di più i due lavori è la parte riguardante il Model.

Sia a loro che a noi, per esempio, è parso opportuno implementare con delle enumerazioni pedine e torri e anche molti aspetti del tavolo di gioco si avvicinano

a ciò che avevamo in mente noi.

Simile a ciò che abbiamo pensato noi è anche l’implementazione del factory method per l’inizializzazione della partita, composto principalmente da due macrosettori: uno riguardante il numero di giocatori e l’altro la modalità di gioco (facile o esperto).

**Differenze:**

Una cosa su cui sicuramente abbiamo idee differenti dai nostri colleghi è l’implementazione del Controller, dove il gruppo 67 ha deciso di utilizzare il design pattren decorator per gestire i turni e le diverse azioni che i giocatori possono compiere.

**Aspetti positivi e negativi**

**Pro:**

L’UML del gruppo 67 ci è apparso piuttosto chiaro e di semplice interpretazione.

Ci sembra che abbia tutti quanti gli elementi necessari per poter consentire la corretta implementazione del gioco; infatti mostra tutti i componenti specificati nelle regole e i metodi all’interno delle varie classi che li rappresentano all’apparenza soddisfano i requisiti per poter permettere di giocare una partita ad *Eriantys* senza riscontrare eventuali intoppi dovuti ad un’erronea realizzazione pratica di quanto visto nello schema.

Sebbene non sia la versione definitiva, il modello di MVC pensato dai nostri colleghi appare piuttosto bilanciato per quanto riguarda la spartizione dei vari compiti che devono svolgere i diversi moduli strutturali.

**Contro:**

Il lavoro mostra diverse classi etichettate come *singleton*, una scelta che, pensando ad un’eventuale istanziazione di più partite contemporaneamente, non ci sembra particolarmente previdente. Però, chiedendo chiarimenti ai componenti stessi del gruppo, ci è stato da loro riferito che la singolarità della classe per loro vale solo per la singola partita e non globalmente.

Un altro aspetto che non ci è apparso particolarmente chiaro è stato quello riguardante la scelta di un decorator per il Controller, che applicato in questo modo al *round* (inteso qui come il complessivo giro, ossia il compimento di un turno da parte di tutti i giocatori) ci è sembrato macchinoso.

**Conclusioni**

Per concludere, l’UML del gruppo 67 è un lavoro piuttosto ben organizzato e, nonostante sia ancora in fase di lavorazione, appare come i nostri colleghi abbiano già delle idee piuttosto chiare su come il gioco debba essere implementato.

Ci è parso che in generale si possa dire che abbiamo delle idee simili sotto molti punti di vista, soprattutto per quanto riguarda la struttura del Model e l’inizializzazione della partita.

Un po’ meno semplice da comprendere è invece la visione che hanno loro di come debba essere portata avanti la realizzazione del Controller, ma ci rendiamo conto noi per primi che in questo momento, in cui comunque non si è ancora scritto molto codice e quindi non si può essere completamente sicuri che quanto pensato possa davvero essere di facile implementazione, è ancora molto presto per dichiarare definitivo lo schema generale da seguire. Inoltre, c’è da precisare come la versione di UML da noi visionata è per il momento sprovvista della View, poiché non ancora stata totalmente pianificata, quindi non si può ancora valutare nella sua pienezza il progetto.