

Progetto UML: Aggregatore di giochi

Il progetto, da svolgere in gruppi di 4 o 5 studenti, prevede la realizzazione di un "aggregatore di giochi", ovvero un programma che permette a un utente di giocare ad una selezione di semplici videogiochi come Tris, Dama, Scacchi o Forza 4.

Gli utenti si possono registrare alla piattaforma oppure giocare come ospiti. Se si effettua l'accesso, il punteggio ottenuto durante le varie sessioni viene accreditato sull'account dell'utente, che prenderà parte a una classifica organizzata per singolo gioco.

Ogni gioco ha un funzionamento simile: l'utente si collega ad una "sala d'aspetto", in attesa che altri giocatori entrino. Ogni gioco ha un numero fisso di giocatori che possono partecipare a una partita.

Una volta avviata la partita, essa si svolge secondo le regole del gioco in questione.

Alla fine della partita, il punteggio conseguito dagli utenti viene accreditato sul loro profilo personale.

Obiettivi del progetto:

- 1) Effettuare uno studio di fattibilità, costruire lo Use Case diagram dell'aggregatore di giochi e pianificare il suo sviluppo.
- 2) Costruire il Class diagram e il Sequence diagram del gioco Tris (SOLO DI QUESTO). La logica sarà di tipo client-server, entrambi sviluppati secondo il modello MVC.
- 3) Sviluppare un'applicazione desktop (.NET Console, Windows Forms, o qualsiasi altra piattaforma) che permetta di giocare a Tris, secondo le specifiche dettate dall'UML dei punti 1 e 2.
- 4) BONUS: sviluppare una o più delle seguenti funzionalità:
 - Computer come player opzionale per Tris: un "bot" in grado di giocare contro l'utente. È possibile utilizzare qualsiasi strategia di gioco del bot, purché efficace e non copiata, soprattutto se non si conosce come essa opera. La programmazione del bot è limitata al caso di campo 3×3. (2 punti)
 - Sviluppare una view alternativa per il client. Lo scopo è mantenere inalterato tutto il resto del codice, per far risaltare la funzione e la flessibilità del modello MVC. (2 punti)
 - Fare in modo che il server gestisca più partite simultaneamente. Se il modello MVC è stato pensato bene, non sarà necessario modificare estensivamente né i diagrammi né il codice. È possibile aggiungere la funzionalità in un secondo momento oppure iniziare il progetto con tale funzionalità già prevista. (2 punti)

- Gestire il salvataggio dei punteggi degli utenti dal lato del server. Essi possono essere salvati su file o su un database, e devono essere visionabili dall'utente all'interno del gioco. (1 punto)

Si ricorda che la documentazione è fondamentale tanto quanto le funzionalità del progetto.

Sviluppo:

Per la gestione e condivisione del codice e della documentazione è necessario utilizzare GitHub. Create un repository per gruppo, aperto solo ai membri dello stesso, in aggiunta agli utenti riccardo.paleari@issgreppi.it e elia.salerno@issgreppi.it. La documentazione va caricata in una cartella './docs' all'interno del repository.

Il fulcro del progetto non è Git, ma siccome sarà il successivo argomento nel programma, verrà utilizzato per accertare il vostro livello di partenza.

Consegna:

La consegna si suddividerà in due parti:

- Consegna della documentazione UML a una settimana dall'inizio del progetto.
- Consegna eventuale del codice a due settimane dall'inizio del progetto.

Lo scopo è cercare di pianificare il progetto prima di svilupparlo, da cui il motivo della consegna spezzata su due date. Saranno comunque accettati piccoli aggiustamenti ai diagrammi, perché è oggettivamente complicato pensare ad ogni possibile evenienza in anticipo. Non sarà però possibile apportare grosse modifiche, specie alle parti chiave della documentazione.

Valutazione:

Il progetto verrà valutato nel seguente modo:

- Verrà tenuto conto come versione da valutare l'ultima pubblicata sul branch principale (master/main) nel repository di GitHub.
- Per la sufficienza è necessario consegnare la documentazione UML (punti 1 e 2). Da ricordare che la documentazione dev'essere perfetta per raggiungere la sufficienza.
- Il terzo punto porta la valutazione massima ottenibile a 8/10.
- Per ottenere di più, sviluppare una funzionalità aggiuntiva tra quelle del punto bonus.
- Se vengono sviluppate due funzionalità, ne verrà valutata solo una, a scelta del gruppo.
- Dopo la consegna, verranno fatte interrogazioni di accertamento ad ogni gruppo, che hanno come scopo confermare il voto. Queste interrogazioni verteranno sulla teoria e sulla discussione della sua implementazione nel progetto, oltre che come prova che tutti i membri del gruppo abbiano contribuito allo sviluppo del progetto.