**Modello dei casi d’uso**

Sommario

Modello dei casid’uso1

1. Requisiti……………………………………………………………………………………………………………………………………………..1
2. Obiettivi e casi d’uso…………………………………………………………………………………………………………………………..2
3. Casi d’uso2

UC1: Gioca partita Gioco dell’Oca2

**1. Requisiti**

Progettare una applicazione (per la simulazione) del famoso Gioco dell'Oca (semplificato).

Un tabellone per il Gioco dell’Oca è costituito da una sequenza di caselle numerate da 1 a N.

Ad ogni casella può essere associata una delle seguenti istruzioni:

* “Avanti di x caselle”;
* “Indietro di y caselle”;

dove x e y sono numeri interi positivi.

Le caselle di partenza (la 1) e di arrivo (la N) non possono avere istruzioni associate.

Scopo del gioco è di arrivare con la propria pedina nella casella di arrivo (o di sorpassarla), a partire dalla casella di partenza, con una successione di mosse ognuna delle quali consiste di:

* lancio di un dado (a 6 facce);
* spostamento in avanti delle pedina di un numero di caselle pari al lancio del dado;
* esecuzione dell’eventuale istruzione associata alla casella di arrivo.

Si suppone che non possano esistere istruzioni che portino la pedina direttamente nella casella di arrivo o oltre, o che portino dietro alla casella di partenza.

Ad una partita del gioco partecipano K giocatori, il primo turno di lancio tocca al giocatore 1, poi il 2, fino al giocatore K e poi nuovamente al giocatore 1, e così via circolarmente. Vince chi per primo raggiunge (o oltrepassa) la casella di arrivo.

Progettare e sviluppare una applicazione in cui:

* un gestore definisca il tabellone di gioco (e quant'altro necessario per l'avvio delle partite)
* un utente possa avviare una (simulazione) di partita definendo quanti giocatori partecipano
* Una volta che la partita è cominciata, ad ogni turno l'applicazione deve far vedere lo stato di tutti i giocatori (cioè dove sono sul tabellone e le mosse che man mano vengono fatte). L’utente deve poter controllare l’avanzamento dei vari passi della (simulazione della) partita

1. **Obiettivi e casi d’uso**

Analizzando i requisiti riportati nel paragrafo precedente, sono stati individuati gli attori principali a cui è destinato il sistema e gli obiettivi che si intende portare a termine; da queste informazioni sono stati ricavati i casi d’uso principali.

|  |  |  |
| --- | --- | --- |
| Attore | Obiettivo | Caso d’uso |
| Utente (osservatore) | Gestire l’avvio di una simulazione della partita del gioco *Gioco dell’Oca* da parte dell’utente | UC1: Gioca partita Gioco dell’Oca |

1. **Casi d’uso**

Viene di seguito fornita una descrizione in formato dettagliato per il seguente caso d’uso:

* Gioca partita Gioco dell’Oca

**UC1: *Gioca partita Gioco dell’Oca***

|  |  |
| --- | --- |
| **Nome del caso d’uso** | **UC1: Gioca partita Gioco dell’Oca** |
| **Portata** | Applicazione Gioco dell’Oca |
| **Livello** | obiettivo utente |
| **Attore primario** | Utente del Sistema |
| **Parti interessate e Interessi** | * **Utente:** vuole osservare l’esecuzione del Gioco dell’Oca, tramite una simulazione, nel modo più fluido possibile. Vuole una visualizzazione chiara dei vari passi della simulazione della partita. |
| **Precondizioni** | L’utente ha avviato la simulazione di una partita del Gioco dell’Oca. |
| **Garanzia di successo (post-condizioni)** | Viene mostrata a schermo la simulazione della partita, con i vari round e i turni eseguiti dai giocatori in ogni round. Per ogni round, inoltre, l’utente ha la possibilità di vedere su quale casella si trova ogni giocatore e il valore del dado lanciato che indica di quante caselle si dovrà muovere. |
| **Scenario principale di**  **successo** | 1. L’Utente richiede l’inizializzazione di una nuova partita e inserisce il numero di giocatori. 2. L’Utente inizia a giocare. 3. Il Sistema visualizza la traccia del gioco per la successiva mossa del giocatore.   Il passo 3 viene ripetuto fino a quando non c’è un vincitore o fino a quando l’Utente non termina di giocare. |
| **Estensioni (o flussi alternativi)** | \*a. In qualsiasi momento, l’applicazione termina in modo anomalo:  per consentire il ripristino, bisogna garantire che il sistema possa essere ripristinato, a partire da qualsiasi passo dello scenario.   1. L’utente riavvia l’applicazione e richiede il ripristino dello stato precedente. 2. L’applicazione ricostruisce lo stato precedente.   2a. L’applicazione rileva delle anomalie che impediscono il ripristino:   1. L’applicazione segnala un errore all’utente, registra l’errore, e passa in uno stato pulito. 2. L’utente inizia una nuova partita. |
| **Requisiti speciali** |  |
| **Elenco delle varianti**  **tecnologiche e dei dati** |  |
| **Frequenza di ripetizione** | potrebbe essere quasi ininterrotta. |
| **Varie** |  |