** Università degli studi di Salerno**

**Corso di Ingegneria del Software**

****

*Partecipanti:*

|  |  |
| --- | --- |
| **Nome Cognome** | **Matricola** |
| **David Grieco** | **0512105112** |
| **Ciro Perfetto** | **0512105520** |
| **Silvio Esposito** | **0512105100** |
| **Fabrizio Trapani** | **0512105298** |

**Requirement Analysis Document (RAD)**

**1.Introduction**

* **1.1 Purpose of the system**

Il sito GamEvaluate si prefissa di offrire l’accesso ad un ambiente di critica videoludica basato sulle medie delle recensioni e dei voti rilasciati dalla community stessa. Tale obiettivo nasce dalla necessità da parte dei videogiocatori di acquisire informazioni utili riguardo al prodotto che in futuro vorranno acquistare nei negozi, eliminando il rischio di effettuare un acquisto di cui si potrebbero pentire.

* **1.2 Scope of the system**

Il sito GamEvaluate è rivolto alla sempre crescente comunità digitale di videogiocatori. GamEvaluate, infatti, permetterà, dato un Videogioco registrato sul sito, di osservare il parere dell’intera comunità di videogiocatori che hanno in passato provato quel Videogioco in particolare; parere suddiviso in numerose caratteristiche (ad esempio la trama, il gameplay, la grafica, l’ottimizzazione, ecc.) ognuna delle quali rappresentata dalla media ottenuta dalle votazioni di ogni singolo utente in quell’ambito.

* **1.3 Objectives and success criteria of the project**

Risolvendo la problematica riguardante acquisti di giochi non rispecchianti le preferenze dell’utente si può ottenere una comunità di videogiocatori di gran lunga più soddisfatta dei prodotti del settore il che, di conseguenza, porterebbe ad un aumento della partecipazione di tali utenti nel miglioramento e nel prolungamento del ciclo di vita dei prodotti in questione: un videogiocatore contento del Videogioco appena acquistato continuerà con una buona probabilità a far parte della comunità virtuale di tale Videogioco, comprando più facilmente, in futuro, elementi aggiuntivi riguardanti quest’ultimo. Inoltre, aumenta in tal modo la fetta di videogiocatori che acquisteranno le entrate seguenti della serie di cui il Videogioco che hanno gradito faceva parte.

* **1.4 Definitions, acronyms, and abbreviations**
* Videogioco: prodotto digitale che rappresenta un gioco digitale.
* Recensione: commento testuale associato ad un videogioco, effettuato da un utente.
* Voto: votazione che può essere associata ad un campo di valutazione di un videogioco espressa in decimi.
* Utente: account generico registrato al sistema.
* Ban: azione che permette di vietare l’accesso di un utente al sito.
* Unban: azione che permette a un utente bannato di riaccedere al sito.
* Moderatore: account che ha permessi di ban e di modifica/eliminazione dei voti.
* Amministratore: account che possiede permessi a tutte le azioni disponibili.
* Visitatore: persona che non è registrata al sito.
* Nominare: concessione o rimozione di permessi ad un account

**2.Current System**

Allo stato attuale esistono molteplici siti che offrono un sistema di votazione in ambito videoludico. Essi, però, non consentono una votazione dettagliata dei vari aspetti di un videogioco, quali: grafica, gameplay, ecc.; bensì, offrono semplicemente un sistema di votazione basato su un voto generale. GamEvaluate offre un completo ed efficiente sistema di votazione, che abbraccia ogni singolo ambito riguardante un videogioco.

**3.Proposed System**

* **3.1 Overview (da mettere)**
* **3.2 Functional requirements**

|  |  |  |
| --- | --- | --- |
| **Codice** | **Descrizione** | **Priorità** |
| **M\_RF\_1** | **Gestione Account** | **Bassa** |
| **M\_RF\_2** | **Ricerca** | **Alta** |
| **M\_RF\_3** | **Interazione gioco** | **Alta** |
| **M\_RF\_4** | **Amministratore** | **Media** |
| **M\_RF\_5** | **Moderatore** | **Media** |

M\_RF\_1 **Gestione Account**

Questa funzionalità permette di gestire il proprio account e svolgere operazioni a lui consentite:

**M\_RF\_1:1.1 Login:** fornisce agli utenti la funzionalità di accedere alla propria area personale

**M\_RF\_1:1.2 Sign-in:** fornisce agli utenti la funzionalità di registrare un nuovo account

**M\_RF\_1:1.3 Logout:** fornisce agli utenti la funzionalità di vuole effettuare il logout dal sito

**M\_RF\_1:1.4 Modifica password:** fornisce agli utenti la funzionalità di modificare la propria password per accedere al sito

**M\_RF\_1:1.5 User info:** fornisce agli utenti la funzionalità di accedere alla propria area personale e vedere le informazioni relative all’account e le proprie recensioni

M\_RF\_2 **Ricerca**

Questa funzionalità permette di cercare un gioco o un utente e svolgere operazioni a lui consentite:

**M\_RF\_2:2.1 Ricerca di un gioco con barra di ricerca:** fornisce agli utenti la funzionalità di ricercare un gioco mediante barra di ricerca selezionando il toggle apposito e inserendo il nome del gioco

**M\_RF\_2:2.2 Ricerca con filtri:** fornisce agli utenti la funzionalità di ricercare un gioco tramite filtri per prezzo o media dei voti

**M\_RF\_2:2.3 Ricerca di un utente con barra di ricerca:** fornisce agli utenti la funzionalità di ricercare un utente registrato al sito mediante barra di ricerca selezionando il toggle apposito e inserendo l’username dell’utente

**M\_RF\_2:2.4 Visualizzare dati di un gioco:** fornisce agli utenti la funzionalità di visualizzare le informazioni relative a un gioco sulla pagina dedicata in base al ruolo (visitatore/utente/amministratore)

**M\_RF\_2:2.5 Visualizzare dati di un altro utente** fornisce agli utenti la funzionalità di visualizzare l’area personale di un altro utente registrato al sito potendo visualizzare username e recensioni da lui effettuate in base al ruolo (visitatore/utente/amministratore)

M\_RF\_3 **Interazione gioco**

Questa funzionalità permette svolgere operazioni a lui consentite su ogni gioco:

**M\_RF\_3:3.1 Aggiungere una recensione:** fornisce agli utenti la funzionalità di aggiungere una recensione, cioè un commento, al gioco

**M\_RF\_3:3.2 Rimuovere una recensione:** fornisce agli utenti la funzionalità di rimuovere una propria recensione

**M\_RF\_3:3.3 Aggiungere voto:** fornisce agli utenti la funzionalità di aggiungere un voto

M\_RF\_4 **Amministratore**

Questa funzionalità permette di modificare, aggiungere o eliminare giochi; rimuovere una recensione; ban e unban utente; gestione utenti.

**M\_RF\_4:4.1 Aggiungi gioco:** fornisce agli utenti la funzionalità di aggiungere un gioco al sito

**M\_RF\_4:4.2 Elimina gioco:** fornisce agli utenti la funzionalità di eliminare un gioco dal sito

**M\_RF\_4:4.3 Modifica gioco:** fornisce agli utenti la funzionalità di modificare un gioco del sito

**M\_RF\_4:4.4 Upgrade utente:** fornisce agli utenti la funzionalità di rendere un rendere un utente moderatore.

**M\_RF\_4:4.5 Downgrade utente:** fornisce agli utenti la funzionalità di retrocedere un moderatore a un utente

**M\_RF\_4:4.6 Elimina utente:** fornisce agli utenti la funzionalità di eliminare un utente registrato

**M\_RF\_4:4.7 Aggiungere un genere:** fornisce agli utenti la funzionalità di aggiungere un genere alla lista dei generi

**M\_RF\_4:4.8 Eliminare un genere:** fornisce agli utenti la funzionalità di eliminare un genere dalla lista dei generi

**M\_RF\_4:4.9 Aggiungere una piattaforma:** fornisce agli utenti la funzionalità di aggiungere una piattaforma alla lista delle piattaforme

**M\_RF\_4:4.10 Eliminare una piattaforma:** fornisce agli utenti la funzionalità di eliminare una piattaforma dalla lista delle piattaforme

M\_RF\_5 **Moderatore e Amministratore**

Questa funzionalità permette di gestire le recensioni e eseguire il ban/unban

**M\_RF\_5:5.1 Rimuovere recensione:** funzionalità che permette di rimuovere una recensione di un altro utente registrato

**M\_RF\_4:5.2 Ban:** funzionalità che permette di negare l’accesso al sito ad un utente

**M\_RF\_4:5.3 Unban:** funzionalità che permette di togliere il ban ad un utente

* **3.3 Nonfunctional requirements**

- Usability. L’utente deve poter utilizzare GamEvaluate con facilità senza conoscenze implicite se non una conoscenza di base del funzionamento di un sito web.

- Supportability. GamEvaluate deve poter essere utilizzato sui principali browser (Chrome, Mozilla, Explorer, Safari).

- Portability. L’utente deve poter accedere a GamEvaluate sia da un PC sia da un dispositivo mobile.

- Implementation. La piattaforma è implementata tramite linguaggio Java e SQL per quanto riguarda il back-end, e linguaggio HTML, Javascript e CSS per quanto riguarda il front-end.

* **3.4 System models (da fare)**

**3.4.1 Scenarios**

* *Scenario 1 (Amministratore):*

Giovanni è l’amministratore del sito GamEvaluate e può gestire i vari utenti e le recensioni del sito. Giovanni si informa attraverso internet riguardo gli ultimi giochi usciti, così da poterli aggiungere al sito e farli valutare dagli utenti. Trova un gioco da poter aggiungere, effettua il log-in compilando il form nella homepage del sito, inserendo come username: admin\_ge e come password: Admin01, e va nella sezione del sito dedicata per poter compilare un form che ha: titolo del gioco, descrizione, genere/i, piattaforma/e di gioco, immagine di copertina e data di uscita. Una completati i campi, clicca sul bottone “aggiungi gioco” e conferma la sua scelta nella finestra di pop-up.

Giovanni, successivamente, nota che Vittorio, un moderatore, non ha effettuato molti dei suoi compiti e quindi decide di renderlo un utente. Quindi accede alla pagina dedicata dove vede tutti i moderatori, clicca sull’elemento della lista che è riferita all’account di Vittorio e si apre un’altra pagina dove vede tutti i dati dell’account. Nella pagina è presente il bottone “Rendi utente”, Giovanni preme quest’ultimo e compare un messaggio che chiede la conferma. Conferma l’operazione e Vittorio diventa un utente normale. Giovanni decide di inserire un nuovo moderatore e quindi essendo a conoscenza della professionalità del suo collega Francesco, va nella pagina di ricerca degli utenti e digita nella barra di ricerca l’e-mail di Francesco. Clicca sull’elemento e accede alla pagina dove visualizza i suoi dati e nella pagina è presente il bottone “rendi moderatore”. Clicca sul pulsante, conferma l’operazione dalla finestra di avviso, e rende Francesco un moderatore.

Giovanni, inoltre, nota che un utente ha un termine offensivo nel suo username e che ha pubblicato varie recensioni con termini altrettanto volgari; decide quindi di cercarlo attraverso la barra di ricerca e clicca sul suo profilo. Viene visualizzato il profilo interessato, quindi Giovanni preme sul pulsante “banna utente”, conferma la sua scelta nella finestra di pop-up e impedisce così all’utente di accedere di nuovo al sito.

Dunque, Giovanni, ordinando i giochi per data di uscita in ordine crescente attraverso i filtri che si trovano vicino alla barra, preme sul pulsante cerca e, dai risultati nota che sulla sua piattaforma è presente un gioco non recensito da molto tempo, il quale non è più disponibile sul mercato. Clicca quindi sul gioco e, nella pagina dedicata, preme sul bottone “Rimuovi gioco”, seleziona il pulsante di conferma del pop-up, ed elimina definitivamente il gioco dalla piattaforma.

* *Scenario 2 (Visitatore/Utente):*

Mario vuole acquistare un nuovo gioco per la sua console, ma è indeciso tra alcuni titoli. Cercando in rete una recensione, si imbatte nel sito GamEvaluate; utilizzando la barra di ricerca, Mario riesce a trovare i titoli in questione e nota che, per ciascuno di questi, nella scheda principale vengono riportate data di uscita, prezzo medio, genere, piattaforma e media generale dei voti in decimi, seguiti da recensioni/commenti e valutazioni numeriche decimali su vari punti, come, ad esempio, trama, gameplay e grafica.

Mario, dopo accurati confronti, sceglie quale videogioco acquistare e sceglie Call of Duty. Dopo averlo giocato per un pò, decide di registrarsi sul sito, in modo tale da poter recensire anche lui il titolo e riportare una valutazione soggettiva riguardo i vari aspetti in aiuto dei futuri visitatori e/o utenti in cerca di informazioni riguardanti il gioco.

Mario, quindi, entra nella sezione di registrazione di GamEvaluate tramite apposita icona nella Homepage; inserisce i suoi dati quali username, e-mail e password. Nota subito come lo username, pur non dovendo rispettare alcun criterio (eccetto la lunghezza massima di 12 caratteri), deve risultare disponibile, quindi non in utilizzo da altri utenti precedentemente registrati, così come l’e-mail; la password, d’altro canto, deve essere di un minimo di 8 caratteri e deve contenere almeno una lettera maiuscola ed un numero. L’username inserito è: mario\_123, la mail utilizzata è: [mario\_123@gmail.com](mailto:mario_123@gmail.com), mentre la password scelta è: Mario123

Una volta registrato, Mario effettua il log-in su GamEvaluate con le credenziali appena inserite. Entrato nell’area utente cerca il titolo che intende recensire aiutandosi con la barra di ricerca dove digita “Call of Duty” e, una volta trovato, clicca su di esso per entrare nell’area relativa al gioco. All’interno di tale area nota la presenza di tutti i campi di valutazione relativi ai vari aspetti del gioco misurati in decimi, oltre alla possibilità di assegnare un voto generale e di lasciare una recensione generale di valutazione per il gioco.

Ci sono in totale 9 slider con voti che vanno da 1 a 10, in ordine sono: gameplay, trama, grafica, creatività, innovazione, coinvolgimento, realismo, rigiocabilità, difficoltà.

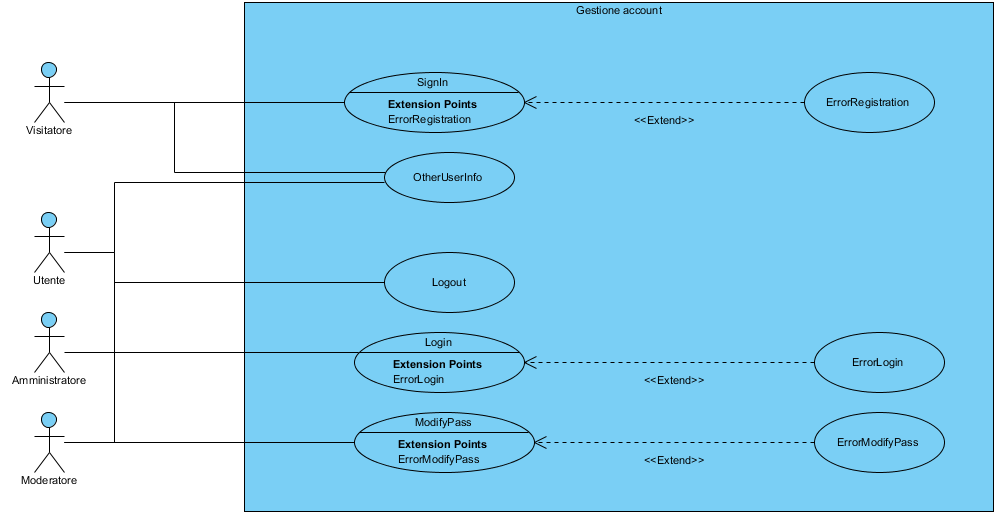
Mario decide quindi di assegnare le votazioni corrette, a suo parere, ed infine lasciare una recensione che descriva la sua esperienza di gioco. Dopo aver scelto tutti i voti preme sul tasto “conferma voti” per aggiungere i suoi voti e dopo aver scritto un commento preme sul pulsante “recensisci” per pubblicare la sua recensione.

* *Scenario 3 (Moderatore):*

Paolo, un moderatore, si occupa di gestire le varie recensioni fatte dagli utenti. Accede sul sito con username: paolo\_mod e password: Pmod01. Paolo decide di controllare le recensioni sugli ultimi giochi appena usciti e nota tra le recensioni un commento con dei termini non consoni al suo interno. Mette da parte l’username dell’utente, “paul\_01”, e attraverso la spunta, visibile solo ai moderatori e agli amministratori, elimina il commento. Poiché vuole essere sicuro che l’utente non abbia effettuato altri commenti negativi, Paolo va nella sezione “ricerca utente” e digita nella barra di ricerca l’username dell’utente, cioè “paul\_01”. Clicca sopra il risultato e arriva nel profilo dell’utente dove può vedere tutti i commenti da lui effettuati nell’apposita sezione “commenti” (questa sezione è visibile anche ad un utente). Dopo aver controllato tutti i suoi commenti nota che non ci sono altri commenti che potrebbero essere eliminati e quindi conclude il suo lavoro.

**3.4.2 Use case**

**Gestione Account**



|  |  |
| --- | --- |
| ID | UC\_1.1 |
| Use case name | **Log-in** |
| Participating actor | Iniziato da *Utente/Amministratore/Moderatore* |
| Flow of events | 1. L’*Attore* visualizza la pagina di login 2. Il *Sistema* mostra un form che richiede l’inserimento di diverse informazioni:   ●Email  ●Password   1. L’*Attore* inserisce le informazioni richieste e clicca il tasto per sottomettere il form 2. Il *Sistema* verifica che:   ●I campi richiesti siano specificati  ●Le credenziali inserite corrispondono ad un profilo registrato nel sistema   1. Il *Sistema* mostra la pagina successiva in base al ruolo dell’utente |
| Entry condition | * L’*Utente/Amministratore/Moderatore* clicca sull’icona nella barra di navigazione che ti permette di effettuare l’accesso   AND   * L’*Utente* non ha effettuato l’accesso |
| Exit condition | * L’*Utente/Amministratore/Moderatore* accede alla propria area personale |
| Extension Point | * Al passo 4 se l’*Utente/Amministratore/Moderatore* inserisce delle credenziali che non corrispondono ad un account registrato o se non ha compilato qualche campo e clicca il tasto per sottomettere il form oppure se l’*Utente* è bannato, fare riferimento a UC\_1.5 |

|  |  |
| --- | --- |
| ID | UC\_1.2 |
| Use case name | **Sign-in** |
| Participating actor | Iniziato dal Visitatore |
| Flow of events | 1. L’*Attore* accede all’area registrazione 2. Il *Sistema* presenta all’utente il form con la possibilità di inserire username, e-mail, password 3. L’*Attore* dopo aver completato tutti i campi del form conferma i dati premendo il bottone 4. Il *Sistema* controlla che:  * Non sia già presente un *Utente* registrato con email o username uguali a quelli inseriti * I campi richiesti siano specificati * Il campo contenente un email sia effettivamente un email * La password rispetti i criteri richiesti * L’Username rispetti i criteri richiesti  1. Il *Sistema* informa l’*Attore* con un pop- up dell’avvenuta registrazione dopo essere stato reindirizzato alla pagina di login |
| Entry condition | * Il*Visitatore* si trova nella pagina di registrazione   AND   * L’*Utente* visualizza il form con i campi da riempire e un bottone “Sign in” |
| Exit condition | * La registrazione va a buon fine |
| Extension Point | 1. Al passo 4 se il *Visitatore* inserisce delle credenziali errate, fare riferimento a UC\_1.6 |

|  |  |
| --- | --- |
| ID | UC\_1.3 |
| Use case name | **Log-out** |
| Participating actor | Iniziato da *Utente/Amministratore/Moderatore* |
| Flow of events | 1. L’*Attore* effettua il *log-out* tramite l’apposito pulsante, posto nella barra di navigazione 2. Il *Sistema* esegue il logout dell’*Utente* e lo reindirizza alla home |
| Entry condition | * L’*Utente/Amministratore/Moderatore* è già loggato GamEvaluate. |
| Exit condition | * L’*Utente/Amministratore/Moderatore* esce dal proprio account |
| Quality requirements | NA |

|  |  |
| --- | --- |
| ID | UC\_1.4 |
| Use case name | **ModifyPassword** |
| Participating actor | Iniziato da *Utente/Amministratore/Moderatore* |
| Flow of events | 1. L’*Attore* accede alla propria area di info e clicca sul pulsante per modificare la password. 2. Il *Sistema* mostra un form che richiede l’inserimento di:  * Vecchia password * Nuova password * Conferma password  1. L’*Attore* inserisce le informazioni richieste 2. Il *Sistema* verifica che:   ● I campi richiesti siano specificati  ● La password rispetti i criteri richiesti  ● La vecchia password corrisponde alla password dell’account  ● La nuova password non sia uguale alla password precedente alla vecchia password   1. Il *Sistema* mostra un pop-up che informa l’*Utente* che l’operazione è stata completata |
| Entry condition | * L’ *Utente/Amministratore/Moderatore* è registrato a GamEvaluate.   AND   * L’ *Utente/Amministratore/Moderatore* ha già effettuato l’accesso al sito e si trova nella pagina per visualizzare le informazioni dell’account. |
| Exit condition | * La password è stata cambiata con successo |
| Quality requirements | * Nel caso di modifica con successo, l’*Utente/Amministratore/Moderatore* viene informato tramite *pop-up.* |
| Extension Point | * Al passo 4 se l’*Utente/Amministratore/Moderatore* non ha compilato qualche campo o se la vecchia password non corrisponde con la password attualmente registrata nel sistema e clicca il pulsante per sottomettere il form fai riferimento allo UC\_1.7 |

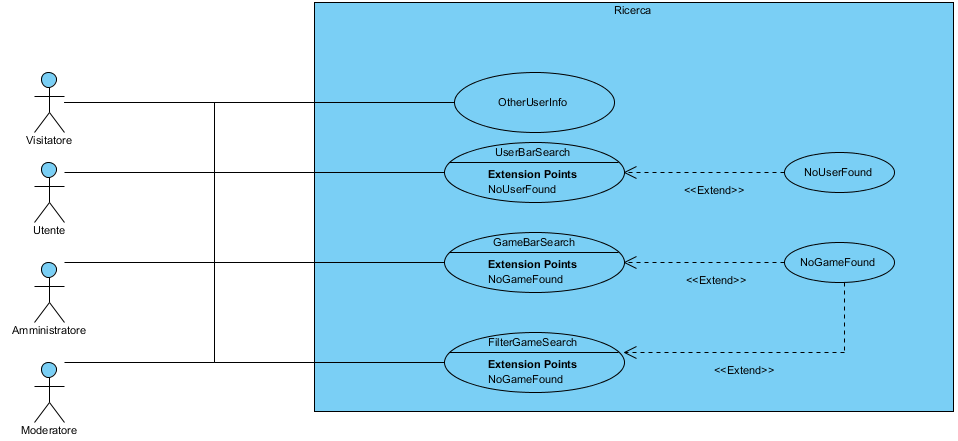
|  |  |
| --- | --- |
| ID | UC\_1.5 |
| Use case name | **ErrorLogin** |
| Participating actor | Iniziato da *Utente/Amministratore/Moderatore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore. 2. L’*Attore* visualizza il messaggio di errore. 3. L’*Attore* inserisce email e password e clicca accedi 4. Il *Sistema* controlla che i campi siano corretti e che l’*attore* non sia bannato. 5. Il *Sistema* mostra la pagina successiva in base al ruolo dell’*Utente*. |
| Entry condition | * L’*Utente/Amministratore/Moderatore* inserisce username e password che non corrispondono ad un account registrato oppure l’Utente è bannato e clicca sul pulsante accedi |
| Exit condition | * L’*Utente/Amministratore/Moderatore* riesce ad autenticarsi nel sito |
| Extension Point | * Al passo 4 se l’*Utente/Amministratore/Moderatore* inserisce nuovamente delle credenziali errate, fare riferimento a UC\_1.5 |

|  |  |
| --- | --- |
| ID | UC\_1.6 |
| Use case name | **ErrorRegistration** |
| Participating actor | Iniziato da *Visitatore* |
| Flow of events | 1. L’*Attore* visualizza il messaggio di errore del sistema. 2. L’*Attore* inserisce di nuovo i dati nel form di registrazione e clicca sul bottone “Registra” 3. Il *Sistema* controlla che:  * Il campo che contiene l’e-mail contenga effettivamente un’e-mail * Le credenziali inserite non corrispondano ad un profilo già registrato nel sistema * La password rispetti i criteri richiesti  1. Il *Sistema* informa l’*Attore* con un pop- up dell’avvenuta registrazione dopo essere stato reindirizzato alla pagina di login |
| Entry condition | * Il *Visitatore* inserisce delle credenziali che non rispettano il formato richiesto e clicca il tasto per sottomettere il form   OR   * Il *Visitatore* tenta di inserire un’username o un’email già presenti nel database |
| Exit condition | Il *Visitatore* completata la registrazione con successo |
| Extension Point | * Al passo 3 se il *Visitatore* inserisce nuovamente delle credenziali errate, fare riferimento a UC\_1.6 |

|  |  |
| --- | --- |
| ID | UC\_1.7 |
| Use case name | **ErrorModifyPassword** |
| Participating actor | Iniziato da *Utente/Amministratore/Moderatore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore. 2. L’*Attore* visualizza il messaggio di errore. 3. L’*Attore* inserisce la nuova password e clicca modifica. 4. Il *Sistema* controlla che i campi siano corretti. 5. Il *Sistema* mostra la pagina successiva in base al ruolo dell’*Utente*. |
| Entry condition | * L’*Utente/Amministratore/Moderatore* deve essere loggato.   AND   * L’*Utente/Amministratore/Moderatore* inserisce i dati che non coincidono con quelli richiesti |
| Exit condition | * L’*Utente/Amministratore/Moderatore* modifica la password. |
| Extension Point | * Al passo 3 se l’*Utente/Amministratore/Moderatore* inserisce nuovamente delle credenziali errate, fare riferimento a UC\_1.7 |

|  |  |
| --- | --- |
| ID | UC\_1.8 |
| Use case name | **UserInfo** |
| Participating actor | Iniziato da *Utente/Moderatore/Amministratore* |
| Flow of events | 1. L’*Attore* accede alla propria area utente per visualizzare i dati dell’account e le recensioni fatte da lui 2. Il *Sistema* mostra:  * Username * Email * Modifica password * Pulsante per modificare password * Lista delle proprie recensioni * Pulsante vicino ad ogni recensione per eliminarla |
| Entry condition | * L*’ Utente/Moderatore/Amministratore* deve essere loggato*.* |
| Exit condition | * Viene visualizzata la pagina di informazioni della persona loggata. |

**Ricerca**

****

|  |  |
| --- | --- |
| ID | UC\_2.1 |
| Use case name | **GameBarSearch** |
| Participating actor | Iniziato da *Visitatore/Moderatore/Amministratore/Utente* |
| Flow of events | 1. Il *Sistema* mostra:  * la barra di ricerca * un toggle per scegliere che elementi cercare * dei filtri per filtrare le ricerche  1. L’*Attore* seleziona nel toggle al lato della barra di ricerca il tipo di elementi da ricercare (giochi o utenti) inizia poi a digitare nella barra il titolo di un videogioco. 2. Il *Sistema* effettua la ricerca nel proprio DB e restituisce i risultati disponibili. |
| Entry condition | * Il *Visitatore/Moderatore/Amministratore/Utente* si trova nella pagina del sito dedicata alla ricerca.   AND   * Il *Visitatore/Moderatore/Amministratore/Utente* digita qualcosa nella barra |
| Exit condition | * Il *Visitatore/Moderatore/Amministratore/Utente* *Amministratore* ha trovato il Videogioco che cercava. |
| Extension Point | * Al passo 3 se la ricerca non produca alcun risultato fare riferimento a UC\_2.5 |

|  |  |
| --- | --- |
| ID | UC\_2.2 |
| Use case name | **FilterGameSearch** |
| Participating actor | Iniziato da *Visitatore/Moderatore/Amministratore/Utente* |
| Flow of events | 1. Il *Sistema* mostra i seguenti filtri:  * Voto generale in ordine decrescente * Voto generale in ordine crescente * Genere * Titolo in ordine alfabetico crescente * Titolo in ordine alfabetico decrescente * Piattaforma  1. L’*Attore* seleziona i filtri relativi in base ai risultati desiderati dalla ricerca (l’*Utente* può anche scrivere nella barra di ricerca e lasciare. 2. L’*Attore* cerca gli elementi in base ai filtri disponibili selezionati dall’*Utente* e restituisce i risultati. |
| Entry condition | * Il *Visitatore/Moderatore/Amministratore/Utente* si trova nella pagina del sito dedicata alla ricerca.   AND   * Il *Visitatore/Moderatore/Amministratore/Utente* digita qualcosa nella barra |
| Exit condition | * Il *Visitatore/Moderatore/Amministratore/Utente* ha trovato il Videogioco che cercava. |
| Extension Point | * Al passo 3 se la ricerca non produca alcun risultato fare riferimento a UC\_2.5 |

|  |  |
| --- | --- |
| ID | UC\_2.3 |
| Use case name | **GuestInfoGame** |
| Participating actor | Iniziato dall’*Visitatore* |
| Flow of events | 1. L’*Attore* clicca su un titolo e viene aperta la pagina ad esso relativa 2. Il *Sistema* mostra:  * Il titolo del Videogioco * La copertina dei Videogioco * La media dei voti per ogni campo * La lista delle recensioni degli utenti |
| Entry condition | * Il *Visitatore* si trova nella pagina di ricerca e clicca su un *Videogioco*. |
| Exit condition | * Viene visualizzata la pagina di informazioni del *Videogioco*. |

|  |  |
| --- | --- |
| ID | UC\_2.4 |
| Use case name | **UserInfoGame** |
| Participating actor | Iniziato da Admin/*Moderatore/Utente* |
| Flow of events | 1. L’*Attore* clicca su un titolo e viene aperta la pagina ad esso relativa 2. Il *Sistema* mostra:  * Il titolo del Videogioco * La copertina dei Videogioco * La media dei voti per ogni campo * Un bottone per aggiungere nuovi voti o un bottone per eliminare una votazione inserita dall’utente * La lista delle recensioni degli utenti * Un bottone per aggiungere una nuova recensione e un bottone per eliminare una recensione vicino ad ogni recensione scritta dall’Utente |
| Entry condition | * Il *Moderatore/Admin/Utente* si trova nella pagina di ricerca e clicca su un *Videogioco*. |
| Exit condition | * Viene visualizzata la pagina di informazioni del *Videogioco*. |

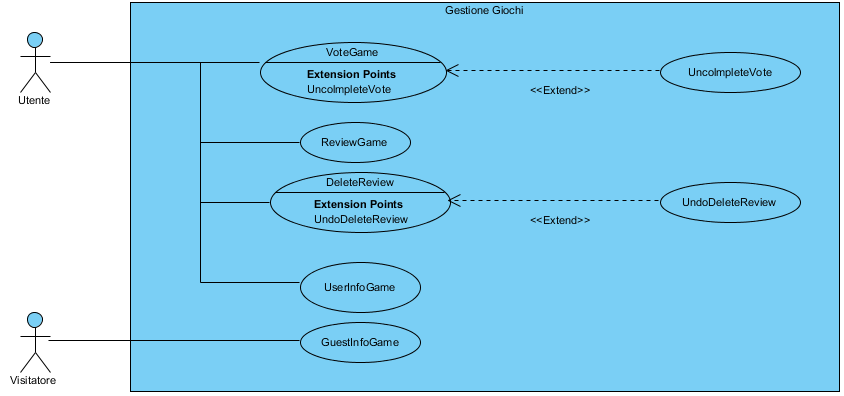
|  |  |
| --- | --- |
| ID | UC\_2.5 |
| Use case name | **NoGameFound** |
| Participating actor | Iniziato dall’*Amministratore/Utente/Moderatore/Visitatore* |
| Flow of events | * Il sistema mostra un messaggio di errore restando sulla stessa pagina |
| Entry condition | * L’utente effettua una ricerca ma non trova nessuna corrispondenza |
| Exit condition | * L’ *Amministratore/Utente/Moderatore/Visitatore* visualizza il messaggio e rimane sulla pagina della ricerca |

|  |  |
| --- | --- |
| ID | UC\_2.6 |
| Use case name | **UserBarSearch** |
| Participating actor | Iniziato da *Visitatore/Moderatore/Amministratore/Utente* |
| Flow of events | 1. Il *Sistema* mostra:  * la barra di ricerca * un toggle per scegliere che elementi cercare * dei filtri per filtrare le ricerche  1. *L’ Attore* seleziona nel toggle al lato della barra di ricerca il tipo di elementi da ricercare (giochi o utenti) inizia poi a digitare nella barra l’username di un utente. 2. Il *Sistema* effettua la ricerca nel proprio DB e restituisce i risultati disponibili. |
| Entry condition | * Il *Moderatore/Amministratore/Utente/Visitatore* si trova nella pagina del sito dedicata alla ricerca.   AND   * Il *Moderatore/Amministratore/Utente* digita qualcosa nella barra |
| Exit condition | * Il *Moderatore/Amministratore/Utente/Visitatore* ha trovato l*’Utente che* cercava. |
| Extension Point | * Al passo 3 se la ricerca non produca alcun risultato fare riferimento a UC\_2.7 |

|  |  |
| --- | --- |
| ID | UC\_2.7 |
| Use case name | **NoUserFound** |
| Participating actor | Iniziato dall’*Amministratore/Utente/Moderatore/Visitatore* |
| Flow of events | * Il *Sistema* mostra un messaggio di errore restando sulla stessa pagina |
| Entry condition | * L’ *Amministratore/Utente/Moderatore/Visitatore* effettua una ricerca ma non trova nessuna corrispondenza |
| Exit condition | * L’ *Amministratore/Utente/Moderatore/Visitatore* visualizza il messaggio e rimane sulla pagina della ricerca |

|  |  |
| --- | --- |
| ID | UC\_2.8 |
| Use case name | **OtherUserInfo** |
| Participating actor | Iniziato da *Visitatore/Utente/Moderatore/Amministratore* |
| Flow of events | 1. L’*Attore* clicca su un utente in modo da vedere la pagina con le sue informazioni 2. Il *Sistema* mostra:  * Username * Lista delle recensioni fatte |
| Entry condition | * Il *Visitatore/Utente/Moderatore/Amministratore* deve essere loggato*.* |
| Exit condition | * Viene visualizzata la pagina di informazioni della persona loggata. |

**Gestione Giochi**

****

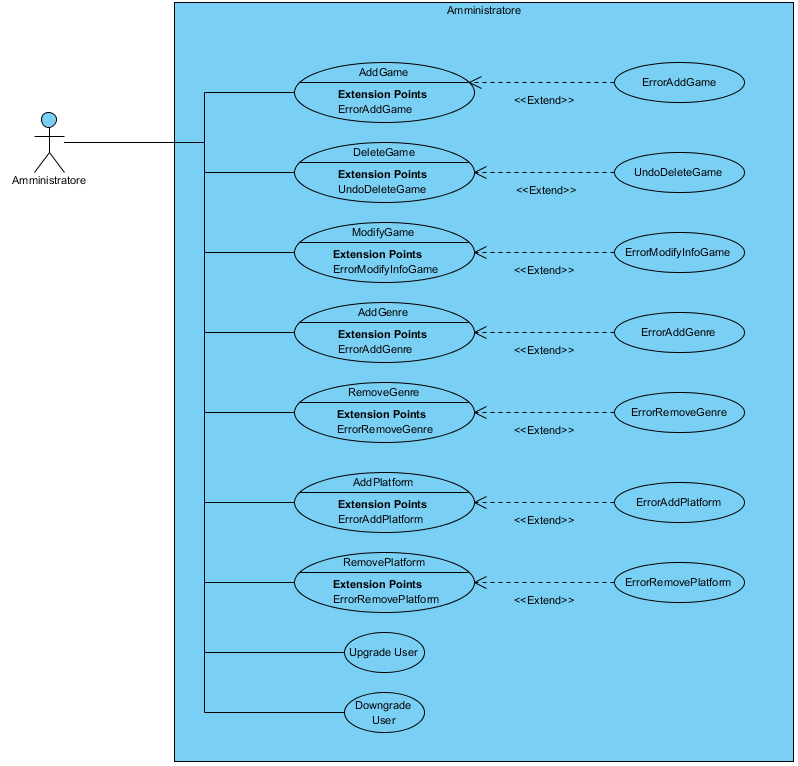
|  |  |
| --- | --- |
| ID | UC\_3.1 |
| Use case name | **ReviewGame** |
| Participating actor | Iniziato dall’ *Utente/Amministratore/Moderatore* |
| Flow of events | 1. L’*Attore* scrive una nuova recensione nell’area di testo e clicca sul pulsante per aggiungerla tra le recensioni. 2. Il *Sistema* aggiunge la nuova recensione a quelle già esistenti |
| Entry condition | * L’ *Utente/Amministratore/Moderatore* ha effettuato l’accesso a GamEvaluate.   AND   * L’ *Utente/Amministratore/Moderatore* è all’interno dell’area informativa del gioco da votare. |
| Exit condition | * L’ *Utente/Amministratore/Moderatore* scrive una nuova recensione |
| Quality requirements | * l’*Utente* viene informato tramite *pop-up* del successo o insuccesso*.* |

|  |  |
| --- | --- |
| ID | UC\_3.2 |
| Use case name | **DeleteReview** |
| Participating actor | Iniziato dall’ *Utente* |
| Flow of events | 1. L’*Attore* Individua la recensione scritta da lui stesso che vuole eliminare e clicca sul pulsante che permette di eliminarla 2. *Il Sistema* mostra una finestra dove gli chiede se è sicuro di volerla eliminare e due bottoni per dare una risposta affermativa o negativa 3. L’*Attore* clicca sul pulsante per dare una risposta affermativa 4. Il *Sistema* elimina la recensione e chiude la finestra |
| Entry condition | * L’ *Utente* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Utente* è all’interno dell’area informativa del gioco da votare.   AND   * L’*Utente* ha già recensito il titolo. |
| Exit condition | * La rimozione della recensione scritta dall’*Utente* va a buon fine. |
| Quality requirements | Nel caso di fallimento l’*Utente* viene informato tramite *pop-up.* |
| Extension Point | Al passo 3 se l’utente clicca sul pulsante per dare una risposta negativa, fare riferimento a UC\_3.5 |

|  |  |
| --- | --- |
| ID | UC\_3.3 |
| Use case name | **VoteGame** |
| Participating actor | Iniziato dall’ *Utente/Amministratore/Moderatore* |
| Flow of events | 1. Il *Sistema* presenterà un form con la possibilità di assegnare un voto ai singoli aspetti del gioco tramite appositi slider. 2. L’*Attore* scorre i vari slider e assegna i voti. Una volta finito preme il tasto di conferma 3. Il *Sistema* verifica che tutti i campi siano stati inseriti 4. Il *Sistema* calcola una media rispetto alla media già presente per ogni campo e la aggiorna ((voto inserito + media presente )/ numero di votazioni) 5. Il *Sistema* mostra un pop up all’*Utente* con l’avvenuta conferma della votazione |
| Entry condition | * L’ *Utente/Amministratore/Moderatore* ha effettuato l’accesso a GamEvaluate.   AND   * L’ *Utente/Amministratore/Moderatore* è all’interno dell’area informativa del gioco da votare.   AND   * L’ *Utente/Amministratore/Moderatore* non ha effettuato ancora una votazione per il Videogioco in questione |
| Exit condition | * La votazione va a buon fine. |
| Extension Point |  |

|  |  |
| --- | --- |
| ID | UC\_3.4 |
| Use case name | **UndoDeleteReview** |
| Participating actor | Iniziato da *Utente* |
| Flow of events | 1. Il *Sistema* chiude la finestra di dialogo 2. Il *Sistema* mostra l’area informativa del Videogioco |
| Entry condition | * L’*Utente* è collegato alla pagina di visualizzazione del gioco e clicca sul pulsante per non eliminare la recensione   AND   * L’*Utente* ha effettuato l’accesso |
| Exit condition | * L’*Utente* non ha eliminato la recensione scritta da lui |

**Amministratore**

****

|  |  |
| --- | --- |
| ID | UC\_4.1 |
| Use case name | **AddGame** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* mostra un form con:  * Titolo del gioco * Descrizione * URL immagine * Genere * Piattaforma * Un pulsante per aggiungere il Videogioco  1. L’*Attore* compila il form e preme il pulsante 2. Il *Sistema* elabora i dati inseriti e controlla che tutti i campi siano corretti e non vuoti 3. Il *Sistema* aggiunge il Videogioco alla lista dei Videogiochi |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* si trova sulla pagina per aggiungere un Videogioco |
| Exit condition | * Il *Videogioco* viene inserito con successo. |
| Quality requirements | * Nel caso di fallimento l’*Amministratore* viene informato tramite *pop-up.* |
| Extension Point | * Al passo 4 se l’*Amministratore* inserisce delle credenziali che non rispettano il formato corretto o se non ha compilato qualche campo e clicca il tasto per sottomettere il form, fare riferimento a UC\_4.6 |

|  |  |
| --- | --- |
| ID | UC\_4.2 |
| Use case name | **DeleteGame** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. L’*Attore* clicca il pulsante per eliminare il Videogioco 2. Il *Sistema* Mostra un messaggio che chiede se è sicuro di voler eliminare il *Videogioco* e due pulsanti, uno per dare una risposta affermativa e l’altro per dare una risposta negativa 3. L’*Attore* clicca sul pulsante per dare una risposta affermativa 4. Il *Sistema* elimina il Videogioco e mostra all’utente un messaggio di avvenuta eliminazione |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* si trova sulla pagina di informazioni di un Videogioco |
| Exit condition | * Il *Videogioco* viene eliminato con successo. |
| Extension Point | * se l’*Amministratore* clicca sul pulsante per dare una risposta negativa, fare riferimento a UC\_4.8 |

|  |  |
| --- | --- |
| ID | UC\_4.3 |
| Use case name | **AdminInfoGame** |
| Participating actor | Iniziato da *Amministratore* |
| Flow of events | 1. L’*Attore* clicca su un titolo e viene aperta la pagina ad esso relativa 2. Il *Sistema* mostra:  * Il titolo del Videogioco * La copertina dei Videogioco * La media dei voti per ogni campo * Un bottone per aggiungere un nuovo voto * La lista delle recensioni degli utenti * Un bottone per aggiungere una nuova recensione e un bottone per eliminare una recensione vicino ad ogni recensione scritta dall’Utente |
| Entry condition | * L’*Amministratore* si trova nella pagina di ricerca e clicca su un *Videogioco*. |
| Exit condition | * Viene visualizzata la pagina di informazioni del *Videogioco*. |

|  |  |
| --- | --- |
| ID | UC\_4.3 |
| Use case name | **ModifyGame** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. L’*Attore* clicca sul bottone per modificare i dati del Videogioco 2. Il *Sistema* mostra un form da riempire con i nuovi valori da inserire, ed un bottone “Conferma” 3. L’*Attore* inserisce i nuovi dati e clicca “Conferma” 4. Il *Sistema* Controlla che il campo non sia vuoti e che l’input rispetti il formato richiesto 5. Il *Sistema* modifica i campi aggiornati |
| Entry condition | * L’ *Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* si trova sulla pagina di visualizzazione del Videogico |
| Exit condition | * Il *Videogioco* viene modificato con successo. |
| Extension Point | Al passo 4 se l’*Amministratore* inserisce un campo che non rispetta il formato corretto o se lascia il campo vuoto e clicca sul pulsante per sottomettere il form, fare riferimento a UC\_4.7 |

|  |  |
| --- | --- |
| ID | UC\_4.4 |
| Use case name | **UpgradeUser** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* mostra:  * Email dell’*Utente* * Username dell’*Utente* * Ruolo dell’*Utente* * Recensioni dell’*Utente* * Pulsante per cambiare ruolo dell’*Utente* * Pulsante per Bannare l’*Utente*  1. L’*Attore* preme sul pulsante per cambiare ruolo all’*Utente* e lo innalza a *Moderatore* |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* è presente sulla pagina di informazioni di un *Utente* |
| Exit condition | * *L’Utente* viene nominato *Moderatore* con successo. |

|  |  |
| --- | --- |
| ID | UC\_4.5 |
| Use case name | **DowngradeUser** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* mostra:  * Email dell’*Utente* * Username dell’*Utente* * Ruolo dell’*Utente* * Recensioni dell’*Utente* * Pulsante per cambiare ruolo dell’*Utente* * Pulsante per Bannare l’*Utente*  1. L’*Attore* preme sul pulsante per cambiare ruolo al *Moderatore* e lo abbassa ad un *Utente* |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* è presente sulla pagina di informazioni di un *Utente* |
| Exit condition | * Il *Moderatore* viene nominato *Utente* con successo. |

|  |  |
| --- | --- |
| ID | UC\_4.6 |
| Use case name | **ErrorAddGame** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore 2. L’*Attore* visualizza il messaggio di errore 3. L’*Attore* Inserisce le informazioni richieste e clicca aggiungi 4. Il *Sistema* controlla che i campi siano corretti 5. Il *Sistema* mostra la pagina successiva in base al ruolo dell’utente |
| Entry condition | * L’A*mministratore* inserisce i dati che non corrispondono al formato richiesto e clicca su aggiungi 0R l’immagine supera le dimensioni consentite   AND   * L’A*mministratore* deve essere loggato |
| Exit condition | * L’A*mministratore* aggiunge il *Videogioco* |
| Extension Point | Al passo 4 se l’*Amministratore* inserisce dei dati errati, fare riferimento a UC\_4.6 |

|  |  |
| --- | --- |
| ID | UC\_4.7 |
| Use case name | **ErrorModifyInfoGame** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore 2. L’*Attore* visualizza il messaggio di errore 3. L’*Attore* Inserisce le informazioni richieste e clicca modifica 4. Il *Sistema* controlla che i campi siano corretti 5. Il *Sistema* mostra la pagina successiva in base al ruolo dell’utente |
| Entry condition | * L’A*mministratore* inserisce i dati che non corrispondono al formato richiesto e clicca su aggiungi   AND   * L’A*mministratore* deve essere loggato |
| Exit condition | * L’A*mministratore* modifica il *Videogioco* |
| Extension Point | Al passo 4 se *l’Amministratore* inserisce dei dati errati, fare riferimento a UC\_4.7 |

|  |  |
| --- | --- |
| ID | UC\_4.8 |
| Use case name | **UndoDeleteGame** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* annulla l’eliminazione del gioco, chiude la finestra e resta sulla pagina di visualizzazione del gioco |
| Entry condition | * L’A*mministratore* clicca sul pulsante per dare una risposta negativa sull’eliminazione di un *Videogioco* |
| Exit condition | * Il gioco non viene eliminato |

|  |  |
| --- | --- |
| ID | UC\_4.9 |
| Use case name | **AddGenre** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* mostra un form con:  * Nome del genere * Descrizione * Un pulsante per aggiungere il genere  1. L’*Attore* compila il form e preme il pulsante 2. Il *Sistema* elabora i dati inseriti e controlla che tutti i campi siano non vuoti e controlla se il nome del Genre non è già esistente nel DB 3. Il *Sistema* aggiunge il Genere alla lista dei Generi |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* si trova sulla pagina per gestire i Generi |
| Exit condition | * Il *Genere* viene inserito con successo. |
| Quality requirements | * Nel caso di fallimento l’*Amministratore* viene informato tramite *pop-up.* |
| Extension Point | * Al passo 3 se *l’Amministratore* inserisce dei dati errati oppure tenta di aggiungere un genere che già esiste, fare riferimento a UC\_4.11 |

|  |  |
| --- | --- |
| ID | UC\_4.10 |
| Use case name | **DeleteGenre** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* mostra una lista con tutti i generi presenti e un pulsante affianco ad ogni genere per eliminarlo 2. L’*Attore* clicca il pulsante per eliminare il Genere 3. Il *Sistema* elimina il Genere e mostra all’utente un messaggio di avvenuta eliminazione |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* si trova sulla pagina per gestire i generi |
| Exit condition | * Il *Genere* viene eliminato con successo. |
| Extension Point | * se l’*Amministratore* tenta di eliminare un genere che alcuni *Videogiochi* hanno ancora come attributo, fare riferimento a UC\_4.12 |

|  |  |
| --- | --- |
| ID | UC\_4.11 |
| Use case name | **ErrorAddGenre** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore 2. L’*Attore* visualizza il messaggio di errore 3. L’*Attore* modifica le informazioni precedentemente inserite e preme aggiungi 4. Il *Sistema* controlla che non esista già un genere con lo stesso nome 5. Il *Sistema* mostra la pagina successiva in base al ruolo dell’utente mostrando un messaggio di conferma |
| Entry condition | * L’A*mministratore* inserisce i dati che non corrispondono al formato richiesto o non inserisce dei dati e clicca su aggiungi   AND   * L’A*mministratore* inserisce un genere già esistente   AND   * L’A*mministratore* deve essere loggato |
| Exit condition | * L’A*mministratore* aggiunge il *Genere* |
| Extension Point | Al passo 4 se l’*Amministratore* inserisce dei dati errati, fare riferimento a UC\_4.11 |

|  |  |
| --- | --- |
| ID | UC\_4.12 |
| Use case name | **ErrorRemoveGenre** |
| Participating actor | Iniziato da *Amministratore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore. 2. L’*Attore* visualizza il messaggio di errore. 3. Il *Sistema* mostra la pagina successiva in base al ruolo dell’*Utente*. |
| Entry condition | * L’*Amministratore* deve essere loggato.   AND   * L’*Amministratore* inserisce prova a eliminare un genere che alcuni *Videogiochi* hanno ancora come attributo oppure il formato dei dati inserito è errato (campi vuoti) |
| Exit condition | * L’*Amministratore* modifica la password. |
| Extension Point | * Al passo 3 se l’*Amministratore* tenta di nuovo di eliminare un genere che alcuni giochi lo hanno come attributo, fare riferimento a UC\_4.12 |

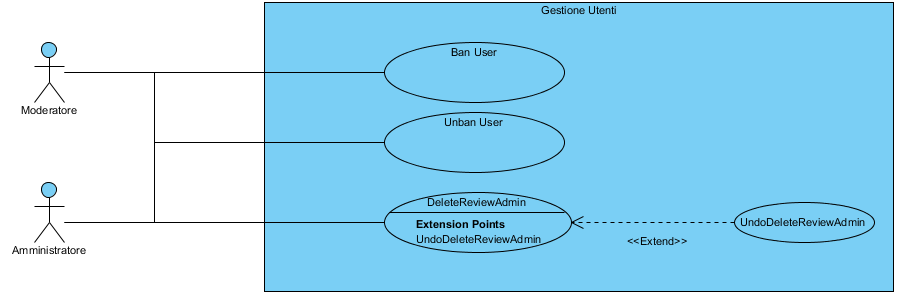
|  |  |
| --- | --- |
| ID | UC\_4.13 |
| Use case name | **AddPlatfom** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* mostra un form con:  * Nome della piattaforma * Pulsante per aggiungere la piattaforma  1. L’*Attore* compila il form e preme il pulsante 2. Il *Sistema* elabora i dati inseriti e controlla che non ci siano altre piattafome con lo stesso nome inserite (non case sensitive) 3. Il *Sistema* aggiunge il Genere alla lista dei Generi |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* si trova sulla pagina per gestire le Piattaforme |
| Exit condition | * La Piattaforma viene inserito con successo. |
| Quality requirements | * Nel caso di fallimento l’*Amministratore* viene informato tramite *pop-up.* |
| Extension Point | * Al passo 3 se *l’Amministratore* non inserisce dei dati oppure tenta di aggiungere una piattaforma che già esiste, fare riferimento a UC\_4.15 |

|  |  |
| --- | --- |
| ID | UC\_4.14 |
| Use case name | **DeletePlatform** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* mostra una lista con tutte le piattaforme presenti e un pulsante affianco ad ogni piattaforma per eliminarlo 2. L’*Attore* clicca il pulsante per eliminare la Piattaforma 3. Il *Sistema* elimina la piattaforma e mostra all’utente un messaggio di avvenuta eliminazione |
| Entry condition | * L’*Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * L’*Amministratore* si trova sulla pagina per gestire le piattaforme |
| Exit condition | * La piattaforma viene eliminata con successo. |
| Extension Point | * se l’*Amministratore* tenta di eliminare una piattaforma che alcuni *Videogiochi* hanno ancora come attributo, fare riferimento a UC\_4.16 |

|  |  |
| --- | --- |
| ID | UC\_4.15 |
| Use case name | **ErrorAddPlatform** |
| Participating actor | Iniziato dall’*Amministratore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore 2. L’*Attore* visualizza il messaggio di errore 3. L’*Attore* modifica le informazioni precedentemente inserite e preme aggiungi 4. Il *Sistema* controlla che non esistano altre Piattaforme con lo stesso nome 5. Il *Sistema* mostra la pagina successiva in base al ruolo dell’utente |
| Entry condition | * L’A*mministratore* inserisce i dati che non corrispondono al formato richiesto e clicca su aggiungi   AND   * L’A*mministratore* inserisce un genere già esistente   AND   * L’A*mministratore* deve essere loggato |
| Exit condition | * L’A*mministratore* aggiunge la Piattaforma |
| Extension Point | Al passo 4 se l’*Amministratore* inserisce dei dati errati, fare riferimento a UC\_4.15 |

|  |  |
| --- | --- |
| ID | UC\_4.16 |
| Use case name | **ErrorRemovePlatform** |
| Participating actor | Iniziato da *Amministratore* |
| Flow of events | 1. Il *Sistema* invia un messaggio di errore. 2. L’*Attore* visualizza il messaggio di errore. 3. Il *Sistema* mostra la pagina successiva in base al ruolo dell’*Utente*. |
| Entry condition | * L’*Amministratore* deve essere loggato.   AND   * L’*Amministratore* inserisce prova a eliminare una piattaforma che alcuni *Videogiochi* hanno ancora come attributo oppure non inserisce la piattaforma da eliminare |
| Exit condition | * L’*Amministratore* elimina la Piattaforma |

**Moderatore**



|  |  |
| --- | --- |
| ID | UC\_5.1 |
| Use case name | **DeleteReviewAdmin** |
| Participating actor | Iniziato dal *Moderatore/Amministratore* |
| Flow of events | 1. L’*Attore* preme sul tasto “x” vicino alla recensione per eliminarla 2. Il *Sistema* Mostra un messaggio che chiede se è sicuro di voler eliminare la recensione e due bottoni, uno per dare una risposta affermativa e l’altro per dare una risposta negativa 3. L’*Attore* seleziona il pulsante di conferma 4. Il *Sistema* elimina la recensione |
| Entry condition | * Il *Moderatore/Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * Il *Moderatore/Amministratore e* è all’interno dell’area informativa del gioco in cui è presente la recensione da eliminare. |
| Exit condition | * La rimozione della recensione va a buon fine e Il *Moderatore* ne viene informato con un *pop-up*. |
| Extension Point | * Al passo 3 se l’utente clicca sul pulsante per dare una risposta negativa, fare riferimento a UC\_5.4 |

|  |  |
| --- | --- |
| ID | UC\_5.2 |
| Use case name | **BanUser** |
| Participating actor | Iniziato dal *Moderatore/Amministratore* |
| Flow of events | 1. L’*Attore* preme sul pulsante per bannare l’*Utente* 2. Il *Sistema* banna l’*Utente* |
| Entry condition | * Il *Moderatore/Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * Il *Moderatore/Amministratore* è sulla pagina informativa dell’*Utente* |
| Exit condition | * *L’Utente* viene bannato con successo. |
| Extension Point |  |

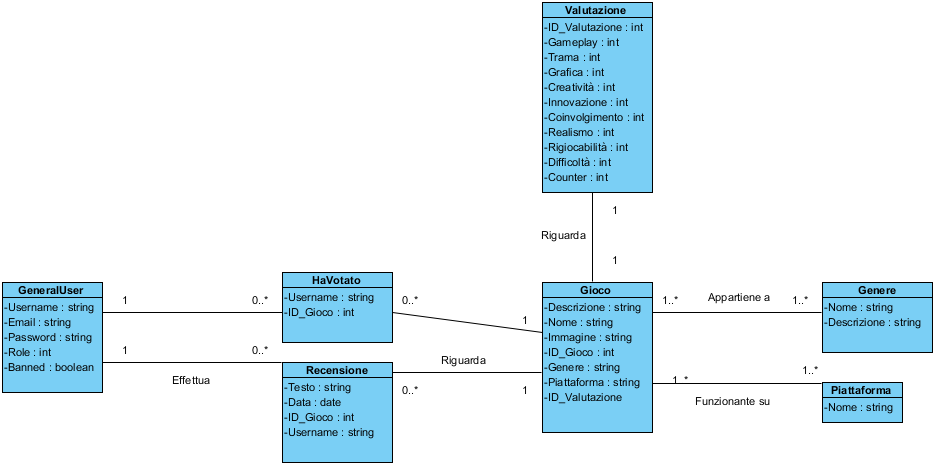
|  |  |
| --- | --- |
| ID | UC\_5.3 |
| Use case name | **UnBanUser** |
| Participating actor | Iniziato dal *Moderatore/Amministratore* |
| Flow of events | 1. L’*Attore* preme sul pulsante per bannare l’*Utente* 2. Il *Sistema* banna l’*Utente* |
| Entry condition | * Il *Moderatore/Amministratore* ha effettuato l’accesso a GamEvaluate.   AND   * Il *Moderatore/Amministratore* è sulla pagina informativa dell’*Utente* |
| Exit condition | * *L’Utente* viene bannato con successo. |
| Extension Point |  |

|  |  |
| --- | --- |
| ID | UC\_5.4 |
| Use case name | **UndoDeleteReviewAdmin** |
| Participating actor | Iniziato dal *Moderatore* |
| Flow of events | 1. Il *Sistema* chiude la finestra di dialogo. 2. Il *Sistema* mostra la pagina di visualizzazione del *Videogioco*. |
| Entry condition | * *Il Moderatore* ha effettuato l’accesso a GamEvaluate. * *Il Moderatore* è all’interno dell’area informativa del gioco che è stato recensito da un altro *Utente*. * *Il Moderatore* preme sul pulsante annulla eliminazione recensione. |
| Exit condition | * *Il Moderatore* non ha eliminato la recensione selezionata. |
| Quality requirements | N.A. |

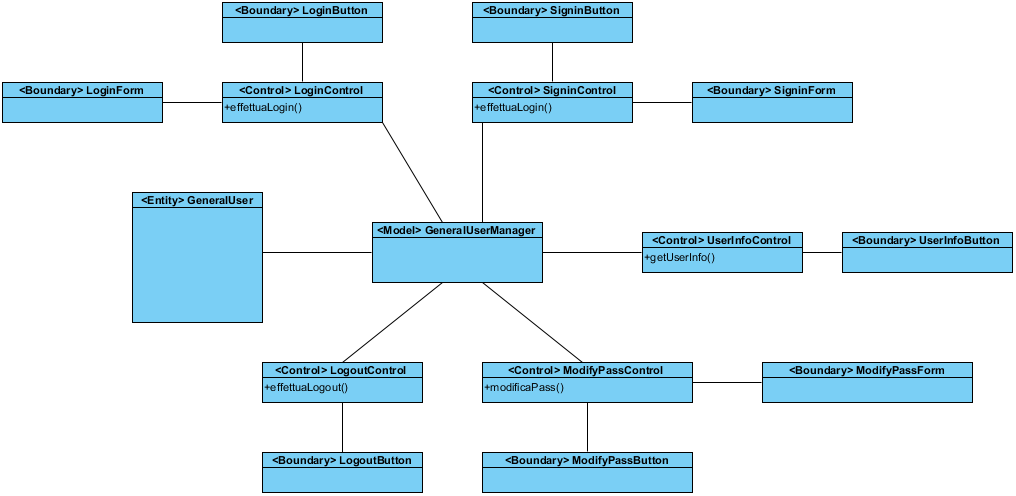
**3.4.3 Object model**

**Class Diagrams**

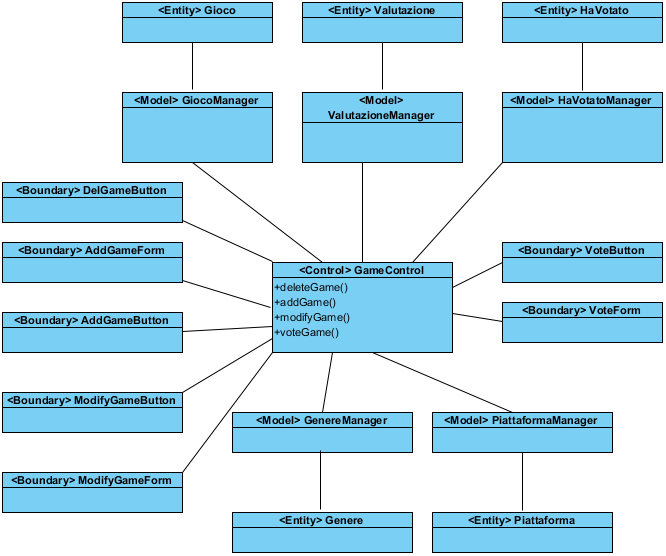
Entity



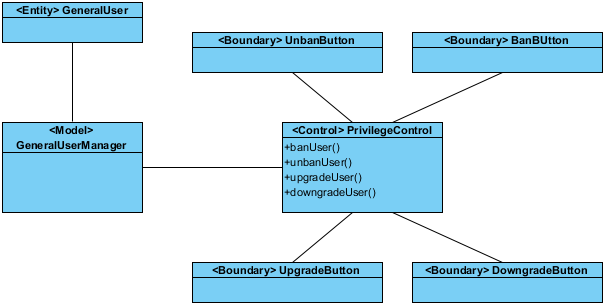
Gestione Account



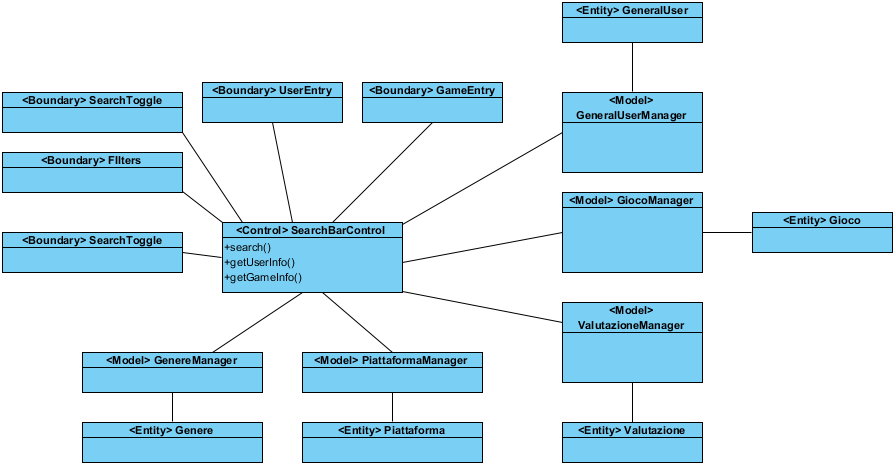
Gestione Giochi



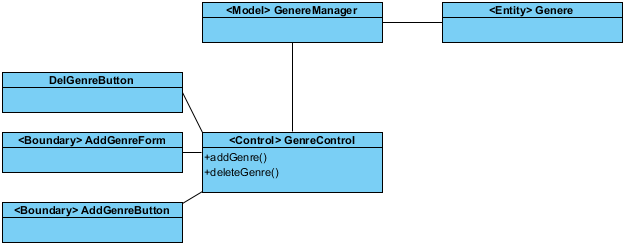
Gestione Utenti



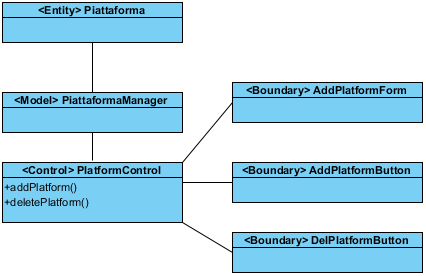
Gestione Ricerca

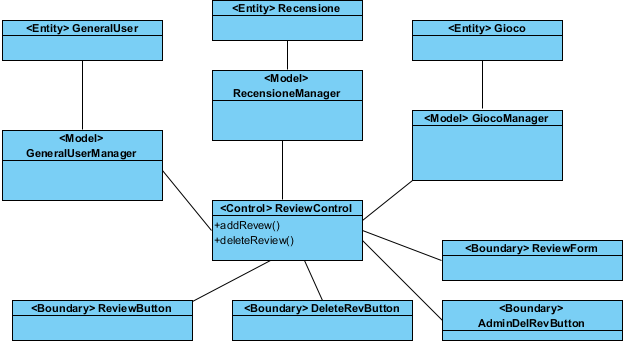


Gestione Generi



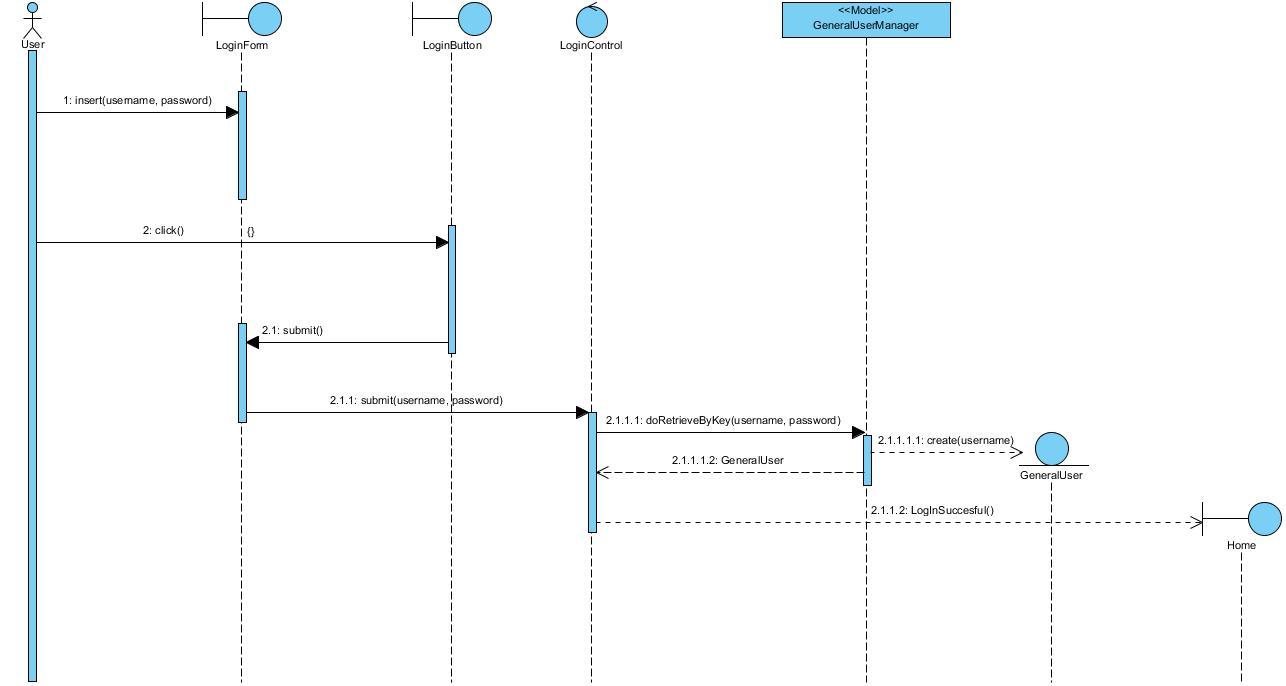
Gestione Piattaforme



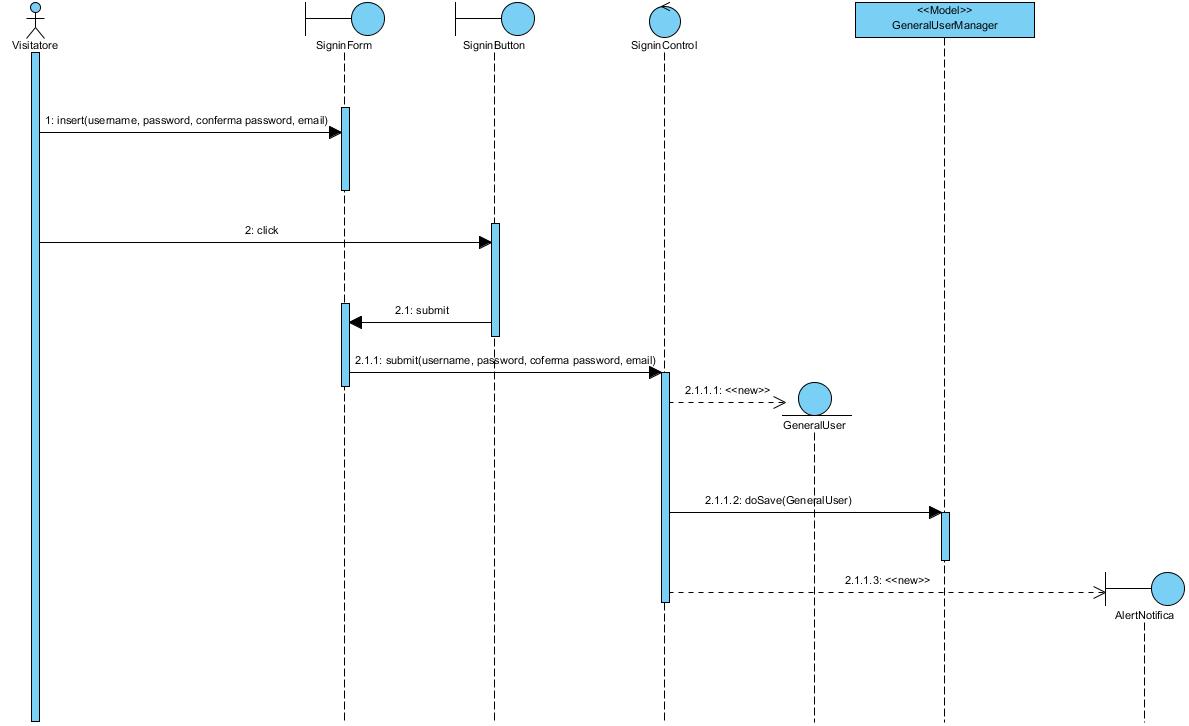
Gestione Recensioni

**Sequence Diagrams**

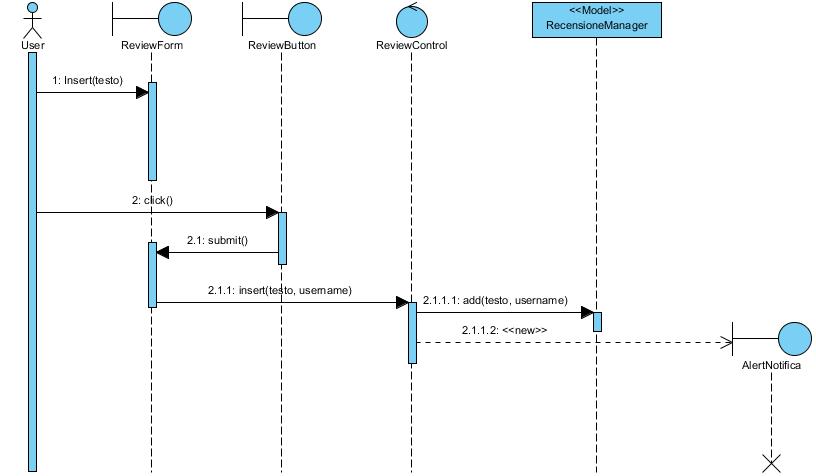
Log in

****

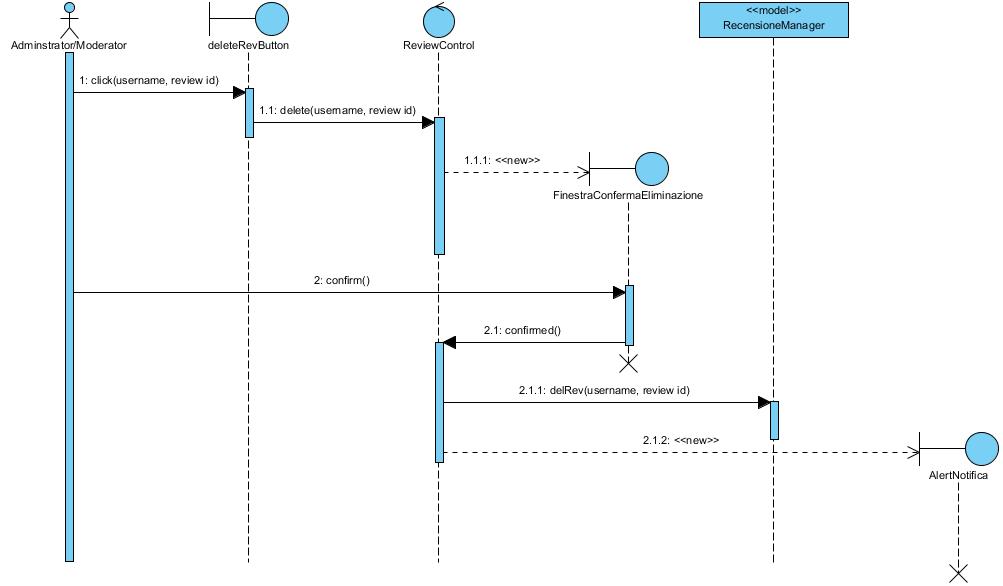
Sign in

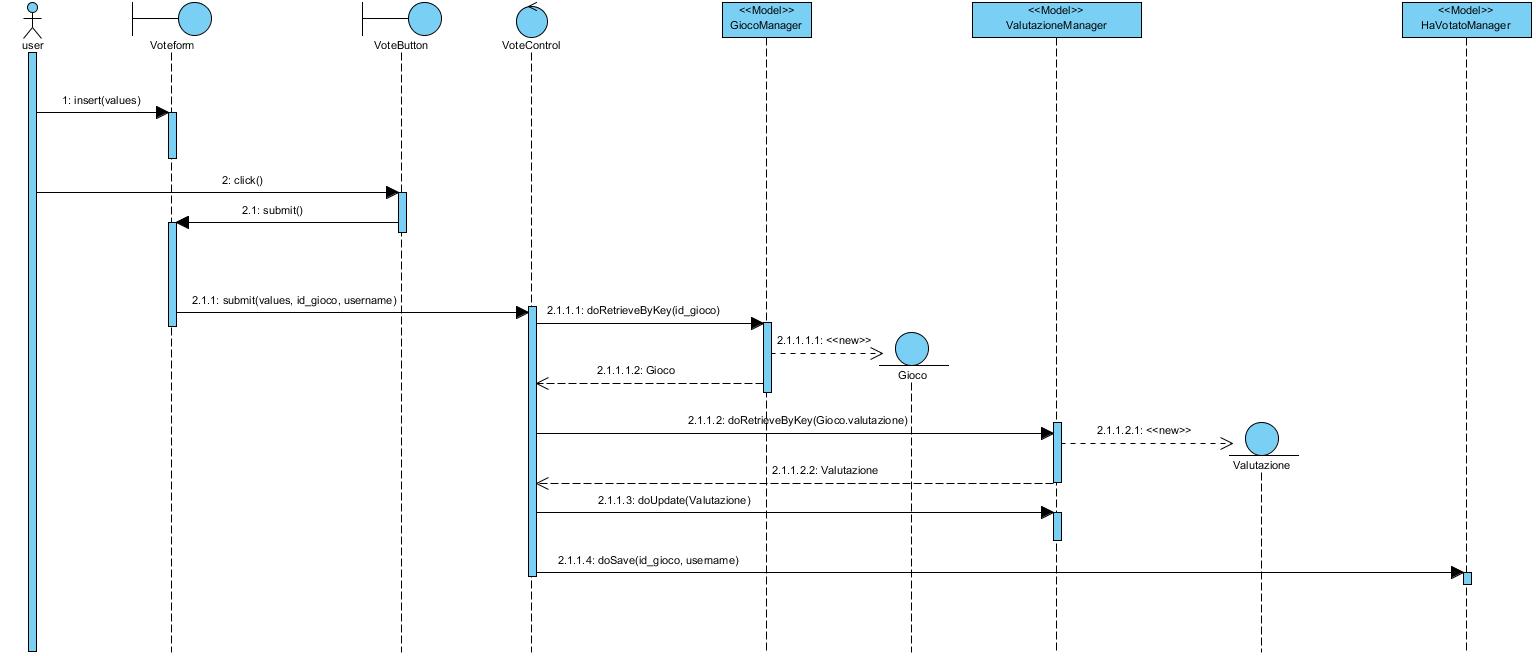


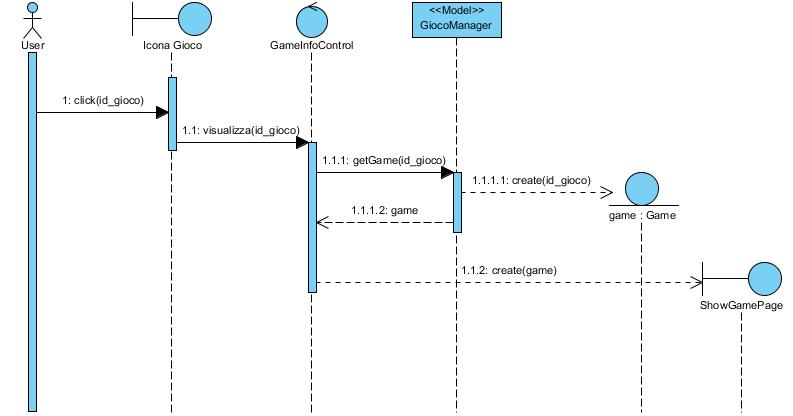
Review game

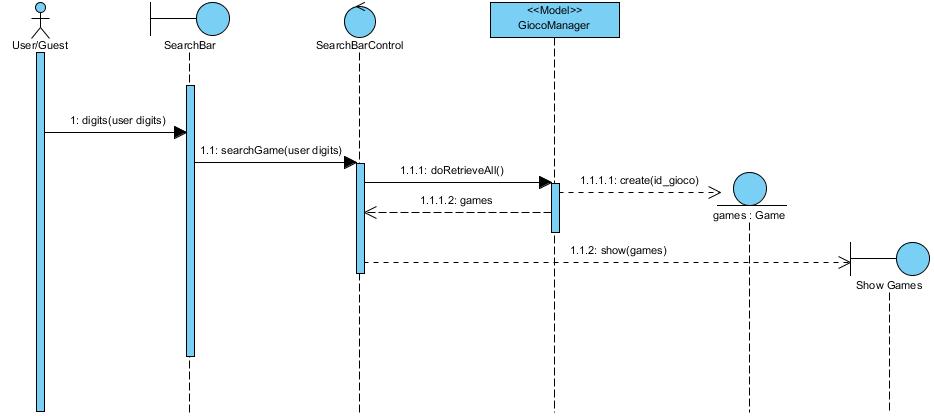


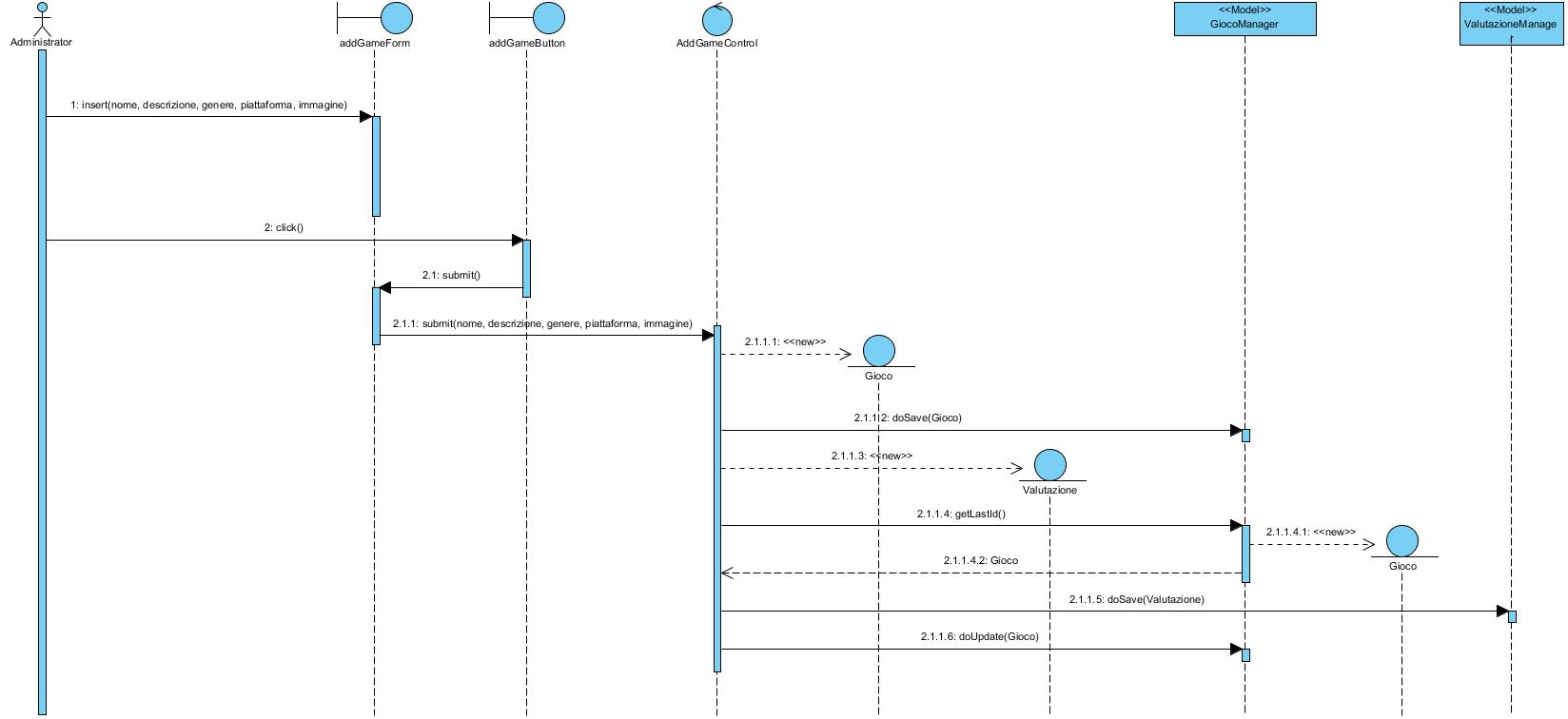
Delete review Admin

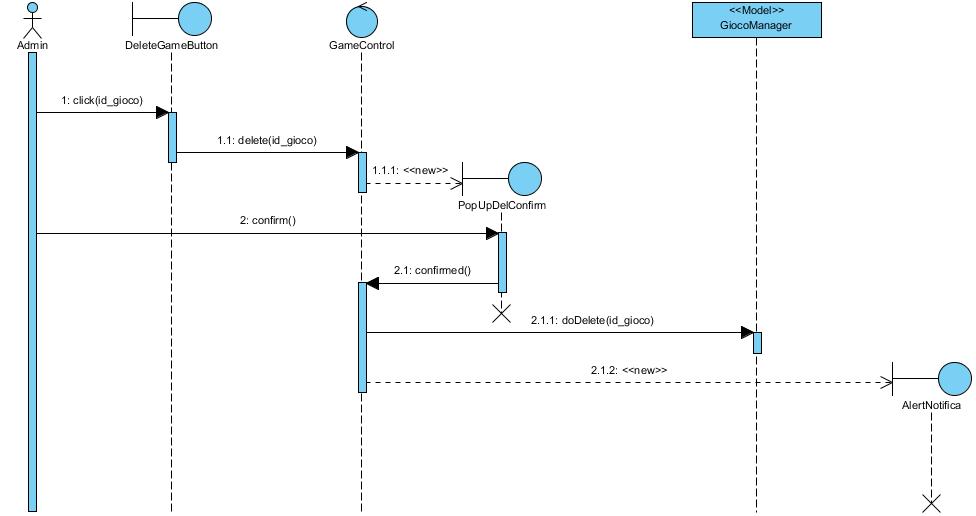


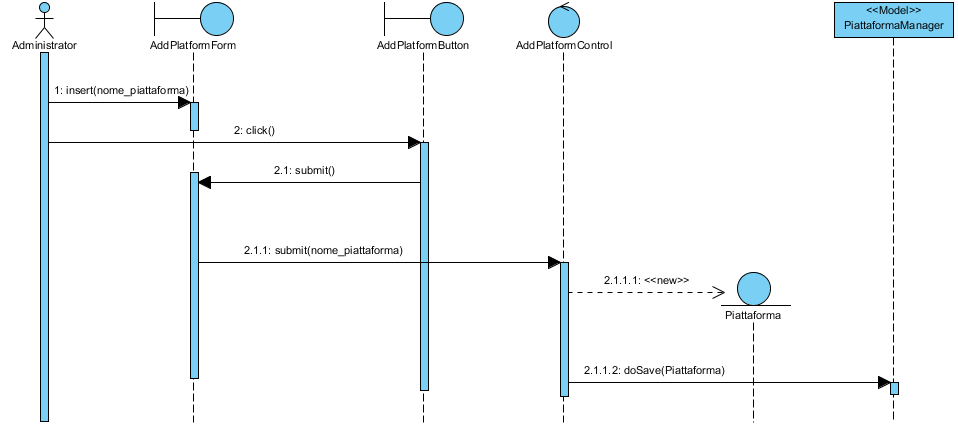
Vote Game

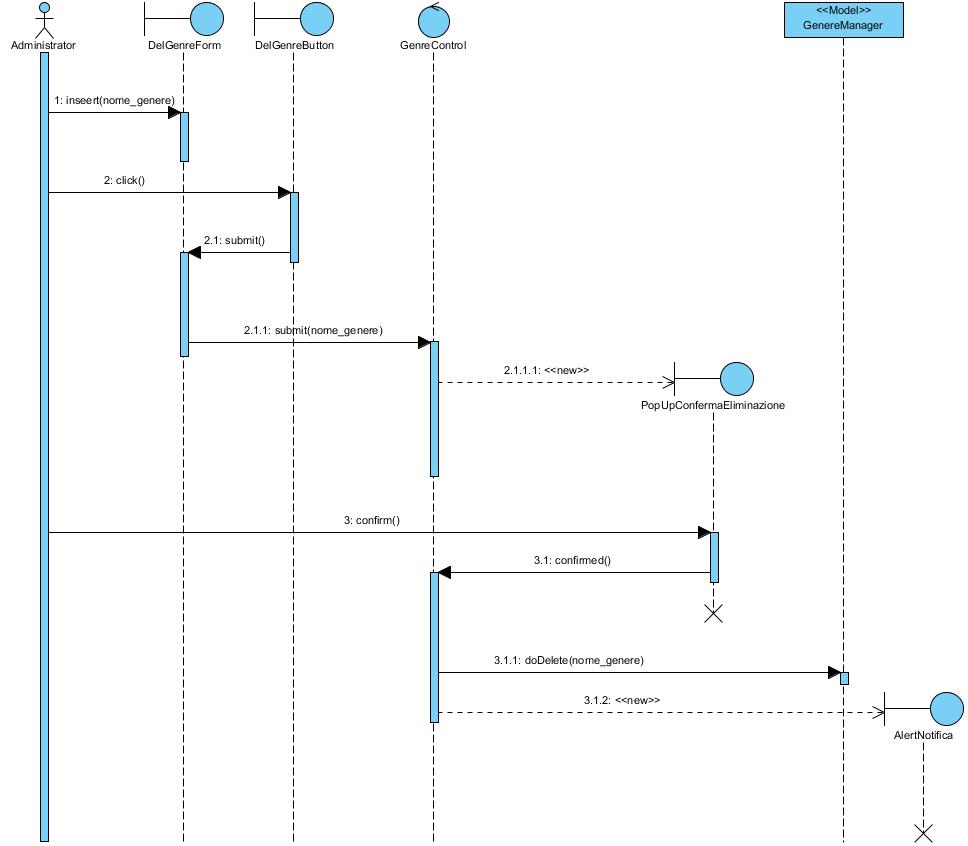
User info game

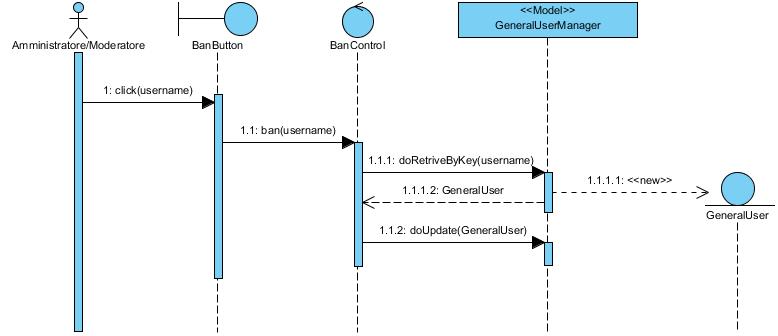
Game bar search

Add game

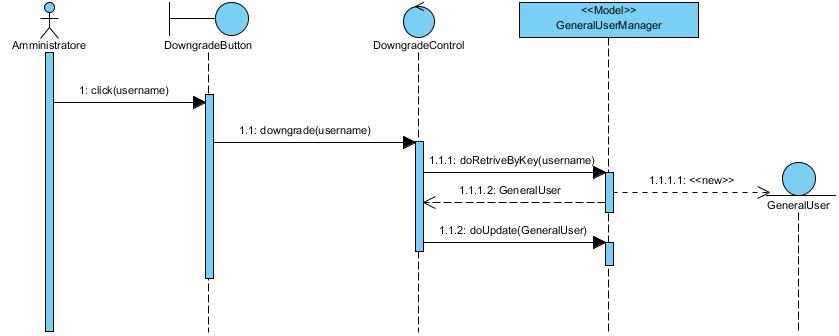
Delete game

Add platform

Delete Genre

Ban user

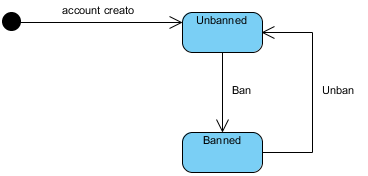
Downgrade user



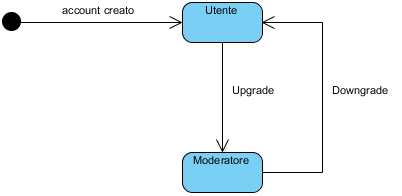
**3.4.4 Dynamic model**

**Statechart Diagrams**

Stato account

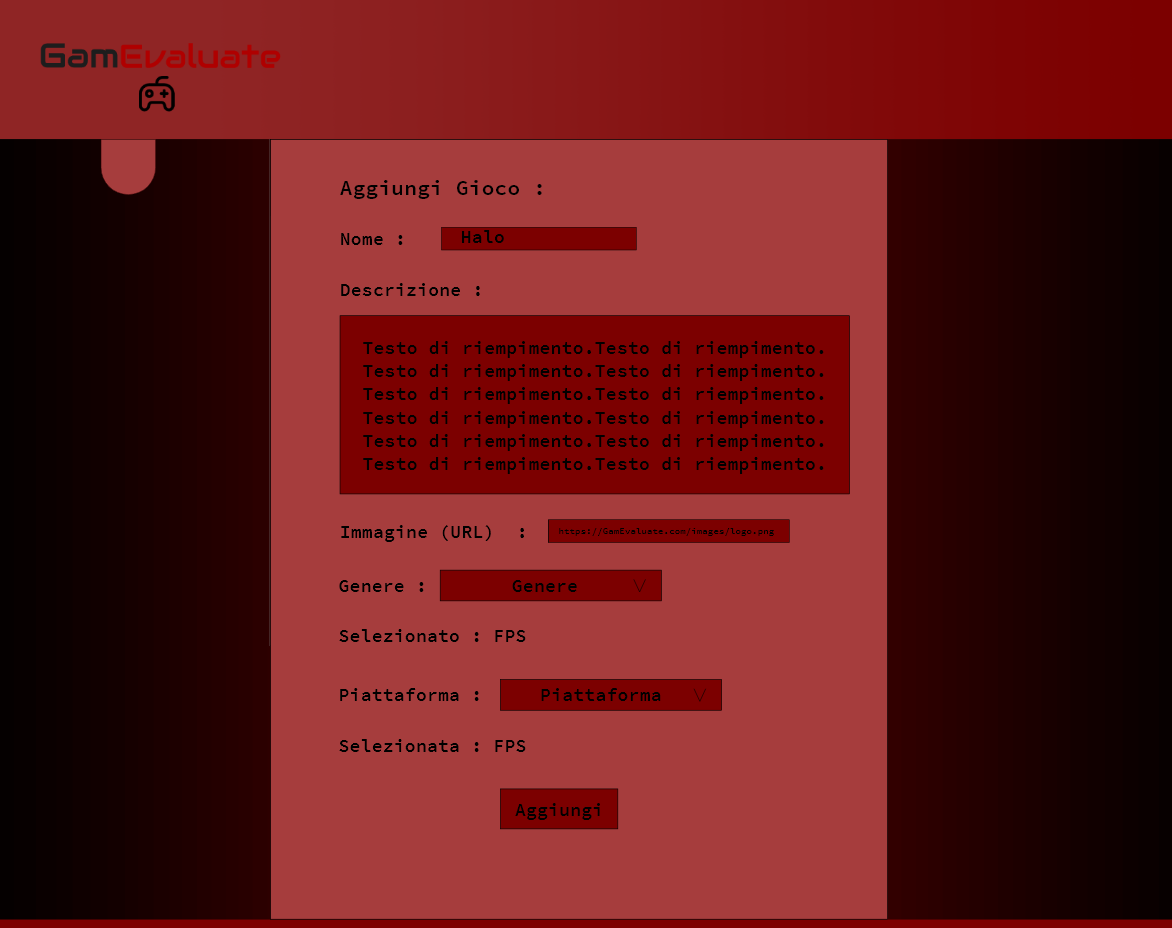


Ruolo account

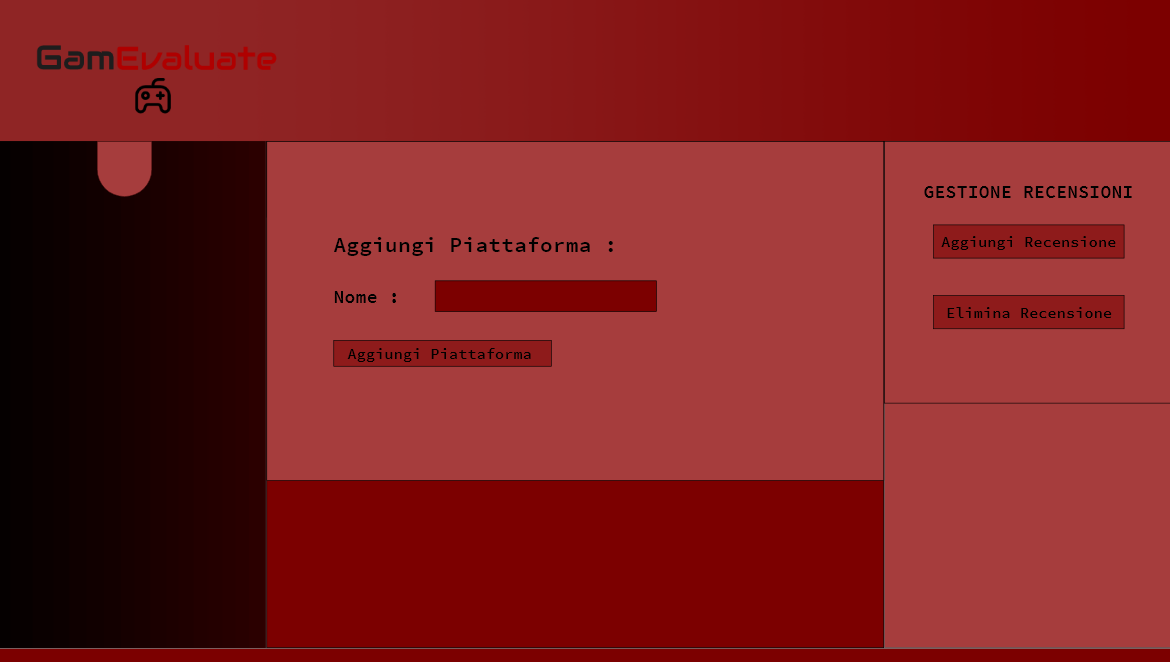


* + 1. **Mock-ups**

Add Game

****

Add platform



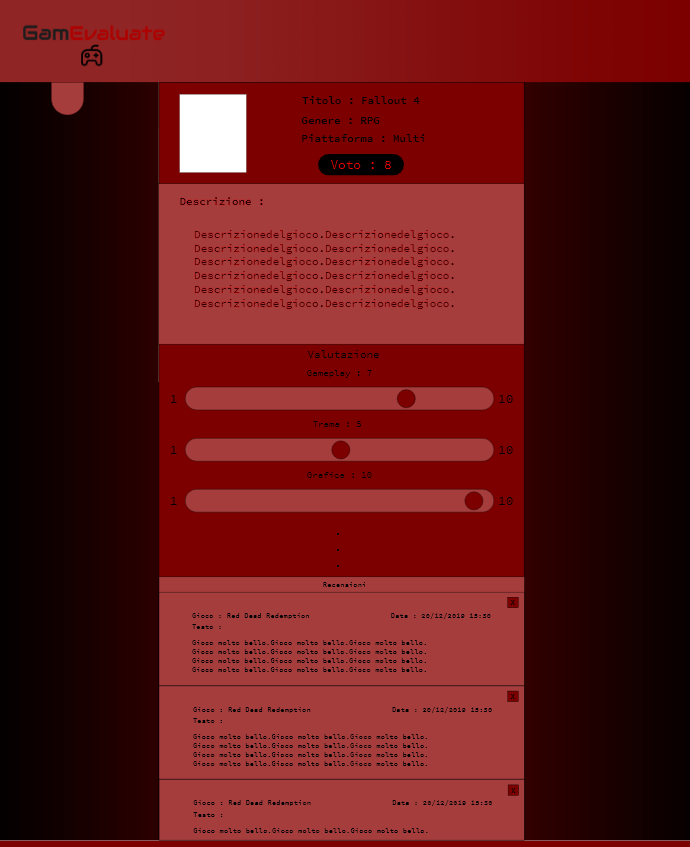
Delete Genre

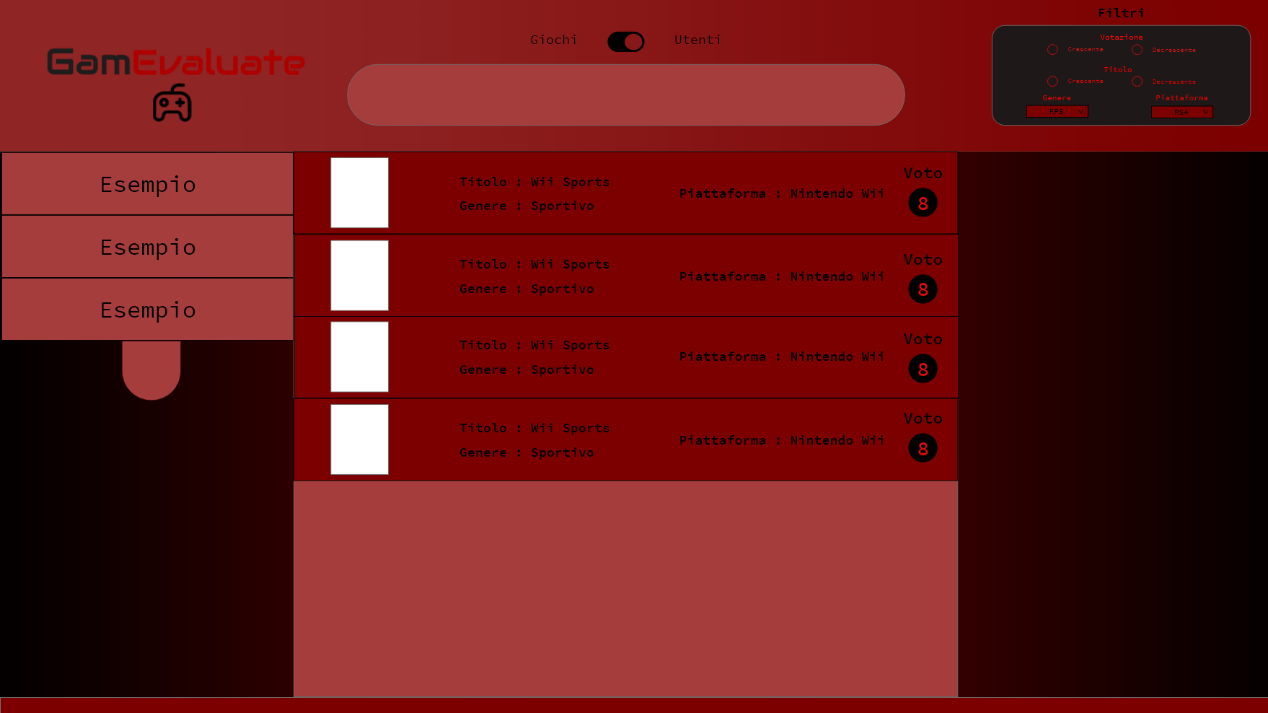
Add Genre

****

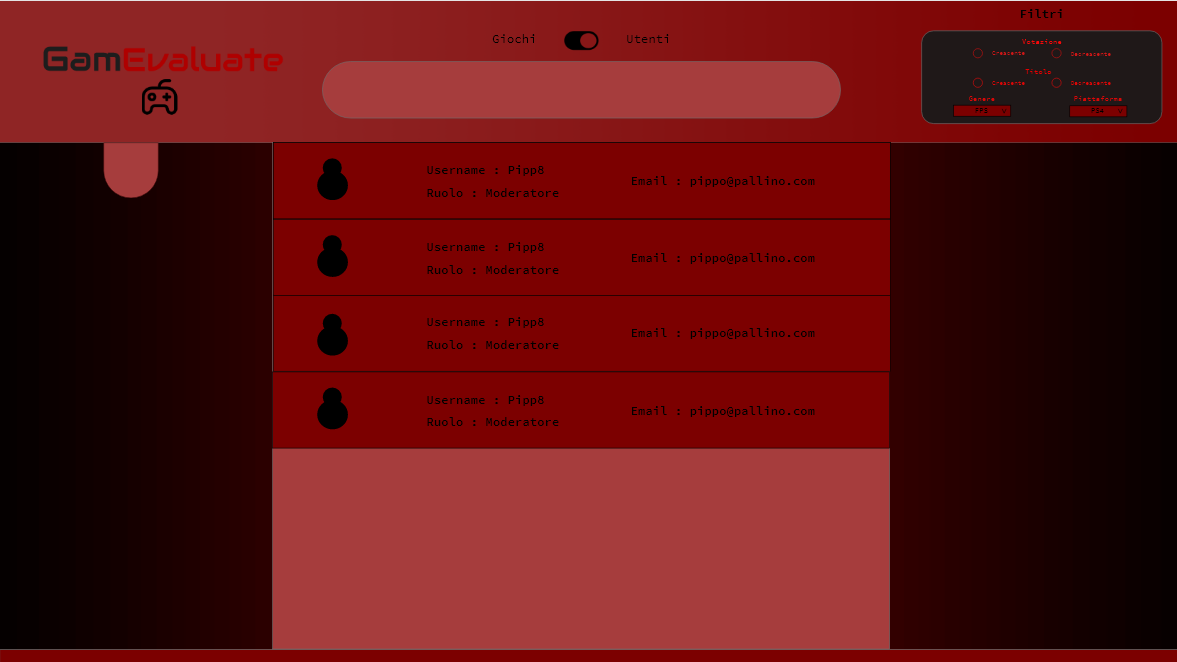
Delete Platform

****

Info game

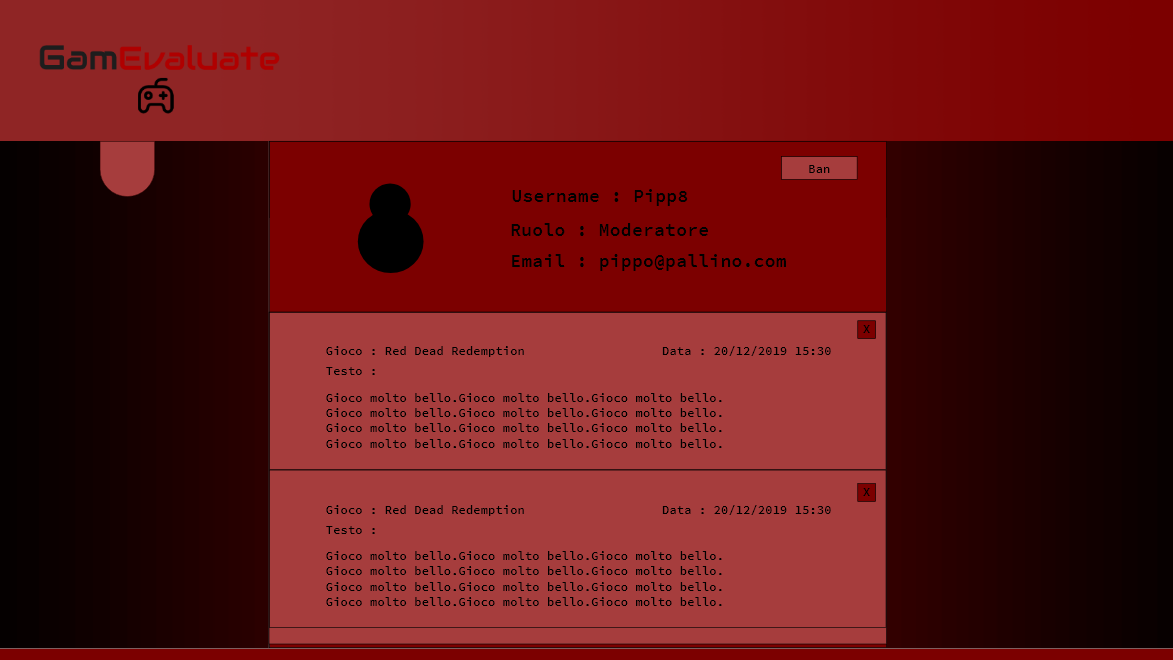
Home games

Home utenti



User info admin



User info moderator

User info