Requirement

Analysis Document

GameSquare

Sommario

[1. Introduzione 3](#_Toc64884520)

[**1.1 Scopo del sistema** 3](#_Toc64884521)

[**1.2 Ambito del sistema** 3](#_Toc64884522)

[**1.3 Obiettivi e criteri di successo** 4](#_Toc64884523)

[**1.4 Definizioni, acronimi e abbreviazioni** 4](#_Toc64884524)

[**1.5 Riferimenti** 4](#_Toc64884525)

[**1.6 Panoramica** 4](#_Toc64884526)

[2. Sistema corrente 5](#_Toc64884527)

[3. Sistema proposto 5](#_Toc64884528)

[**3.1 Panoramica** 5](#_Toc64884529)

[**3.2 Requisiti funzionali** 6](#_Toc64884530)

[**RF\_AUT: AUTENTICAZIONE** 6](#_Toc64884531)

[**RF\_USER: CONTRIBUTI UTENTE** 6](#_Toc64884532)

[**RF\_NAV: NAVIGAZIONE DATABASE** 6](#_Toc64884533)

[**RF\_ADMIN: AMMINISTRAZIONE** 6](#_Toc64884534)

[**3.3 Requisiti non funzionali** 7](#_Toc64884535)

[**3.3.1 Usabilità** 7](#_Toc64884536)

[**3.3.2 Affidabilità** 7](#_Toc64884537)

[**3.3.3 Prestazioni** 7](#_Toc64884538)

[**3.3.4 Manutenibilità** 7](#_Toc64884539)

[**3.3.5 Implementazione** 7](#_Toc64884540)

[**3.3.6 Interfaccia** 7](#_Toc64884541)

[**3.3.7 Packaging** 8](#_Toc64884542)

[**3.3.8 Legali** 8](#_Toc64884543)

[**3.4 Modelli di sistema** 8](#_Toc64884544)

[**3.4.1 Scenari** 8](#_Toc64884545)

[**3.4.2 Use Case Model** 10](#_Toc64884546)

[**3.4.3 Object Model** 14](#_Toc64884547)

[**3.4.4 Dynamic Model** 21](#_Toc64884548)

1. Introduzione

## **1.1 Scopo del sistema**

GameSquare propone una piattaforma centralizzata e pensata per videogiocatori, a prescindere dalla loro esperienza, e sviluppatori.

Gli utenti registrati potranno aggiungere al proprio profilo i videogiochi in loro possesso, catalogandoli come “acquistati”, “in corso di gioco”, “completati”, e opzionalmente aggiungere a un gioco completato l’accezione di “platinato” (completato al 100%), con possibilità di modificare la categoria in ogni momento. Si potranno inoltre votare i videogiochi che sono stati già aggiunti alla propria lista personale, fornendo un voto in scala decimale. Ogni utente potrà proporre al gestore del catalogo di aggiungere un titolo mancante, fornendo della documentazione che prova l’esistenza dello stesso. Infine, i giocatori avranno un punteggio personale, che aumenterà completando o platinando giochi, e terrà conto della difficoltà degli stessi. Quest’ultima è decisa da un punteggio assegnato al singolo gioco, direttamente proporzionale alla sua presunta difficoltà.

Ogni gioco avrà la sua pagina, in cui sarà possibile visualizzare le informazioni base del titolo, la valutazione generale sulla base dei voti degli utenti e quanto sono stati aggiunti a una determinata categoria rispetto alle altre. La pagina di ogni gioco avrà una sezione dedicate alle discussione più recenti legate al titolo.

Una piattaforma del genere è utile all’utente che vuole avere un’idea d’insieme di un titolo prima di acquistarlo, come anche a coloro che vogliono chiedere a giocatori più esperti come completare sezioni difficoltose, o semplicemente chi vuole discutere di un titolo di suo gradimento.

Inoltre può giovare anche agli sviluppatori, in quanto possono rendersi conto di quante persone apprezzino il loro prodotto tanto da completarlo, e ovviamente leggere e scambiare commenti con la community.

## **1.2 Ambito del sistema**

Il mondo dell’intrattenimento videoludico è diventato negli ultimi anni un fenomeno esteso a centinaia di milioni di persone, grazie a servizi e piattaforme che permettono un semplice accesso a cataloghi molto grandi di videogiochi (e. g. Steam, PSN, Nintendo eShop, etc.), e fenomeni mediatici come la comunità di videogiocatori su YouTube o su Twitch, gli eSports e le molte riviste online specializzate.

Con questa crescita, molti fruitori del prodotto sentono il bisogno di catalogare i propri giochi e analizzarli per avere un’idea di quanti titoli in proprio possesso debbano ancora essere completati (il cosiddetto backlog). Inoltre, la ricerca di una community specifica per un determinato gioco che sia per fare domande riguardo al completamento dello stesso, organizzare tornei in caso di un titolo competitivo, o semplicemente discussione generale, si fa sempre più difficoltosa. Ciò avviene a causa dell’aumento più che lineare della quantità di titoli sul mercato, grazie anche al successo recente di titoli creati da sviluppatori indipendenti e la riscoperta di giochi di generazioni passate (retrogaming).

Un altro aspetto da considerare è la quantità di servizi diversi che offrono qualche tipo di aggregazione per i giocatori, come gruppi Facebook, 4chan, Discord e le miriadi di forum e imageboards che tuttavia non sono centralizzati, e raramente pensati per ospitare questo specifico tipo di utenza.

## **1.3 Obiettivi e criteri di successo**

Obiettivi primari:

* Fornire una piattaforma centralizzata per tenere traccia del proprio catalogo videoludico.
* Compartimentalizzare le discussioni riguardanti uno specifico gioco per rendere più intuitivo lo scambio di opinioni o l’organizzazione di community.
* Offrire agli sviluppatori l’occasione di avere feedback diretto sulle loro opere.

Criteri di successo:

* Il sito deve essere online 24/7.
* Deve avere un tempo di risposta basso.

## **1.4 Definizioni, acronimi e abbreviazioni**

NA

## **1.5 Riferimenti**

NA

## **1.6 Panoramica**

Il seguente documento “GameSquare RAD” è diviso in sezioni ed ha la seguente composizione:

**- Sezione di Introduzione:**

In questa sezione è spiegato in breve qual è l'ambito del progetto e lo scopo del sistema che verrà implementato definendo gli obiettivi e i criteri di successo del sistema.

**-Sezione Sistema Corrente:**

Questa sezione resterà vuota in quanto il sistema proposto verrà creato da zero e non andrà a sostituire alcun servizio pubblico esistente.

**-Sezione SISTEMA PROPOSTO:**

In questa sezione viene inizialmente data una panoramica del sistema che deve essere realizzato e cosa andrà ad implementare.

Si procede con la sottosezione dei Requisiti Funzionali del sistema dove vengono identificate le funzionalità che Il sistema deve offrire.

I Requisiti Non Funzionali seguono il modello FURPS+, composto da: Usabilità, Affidabilità, Prestazioni, Manutenibilità, Implementazione, Interfaccia, Packaging, Legali. Seguono poi gli scenari, gli use-case e gli use-case diagram.

2. Sistema corrente

NA

3. Sistema proposto

## **3.1 Panoramica**

Il sistema da noi proposto è un sito web che offrirà servizi sia a utenti facenti parte della community videoludica, siano essi sviluppatori, nella forma di discussioni con i fruitori del loro prodotto, o i fruitori stessi, nella forma di liste per tenere traccia del proprio catalogo, discussioni con altri utenti e valutazioni.

L’utente registrato standard potrà:

* Visionare il proprio profilo con i relativi dati più rilevanti.
* Accedere alla lista da lui creata e modificarla a piacimento.
* Valutare i giochi presenti nella propria lista.
* Creare o contribuire a discussioni già esistenti, oltre che richiederne la creazione.
* Proporre l’aggiunta di un gioco, se dispone delle prove della sua esistenza.

Lo sviluppatore potrà:

* Visionare il proprio profilo con una sezione dedicata ai propri giochi sviluppati.
* Creare o contribuire a discussioni già esistenti.
* Aggiungere il proprio il gioco.

I moderatori potranno:

* Moderare e rimuovere qualsiasi tipo di contenuto (discussioni, commenti, etc.) qualora questo sia inopportuno.
* Accettare o eliminare richieste di creazione di discussioni o di cancellazione di contenuti inopportuni da parte degli utenti.

I gestori catalogo potranno:

* Accettare o eliminare richieste da parte degli utenti di aggiunta di un gioco al sito.
* Aggiungere essi stessi un gioco al sito, oltre che modificarne le informazioni.

Gli utenti non registrati potranno:

* Navigare il sito e cercare giochi tramite ricerca testuale, parametri o titoli correlati.
* Visualizzare la pagina di un prodotto e le informazioni relative, comprese le discussioni a riguardo.

## **3.2 Requisiti funzionali**

### **RF\_AUT: AUTENTICAZIONE**

* **RF\_AUT\_01 – Registrazione** –Priorità Media

Il sistema deve permettere la registrazione di un utente.

* **RF\_AUT\_02 – Login** –Priorità Alta

Il sistema deve permettere l’accesso di un utente.

* **RF\_AUT\_03 – Log out** –Priorità Alta

Il sistema deve permettere all’utente di terminare la propria sessione.

* **RF\_AUT\_04 – Recupero Password** –Priorità Bassa

Il sistema deve permettere all’utente di recuperare la propria password.

* **RF\_AUT\_04 – Modifica Password** –Priorità Bassa

Il sistema deve permettere all’utente di modificare la propria password.

### **RF\_USER: CONTRIBUTI UTENTE**

* **RF\_USER\_01 – Aggiungi Gioco alla Lista** –Priorità Alta

Il sistema deve permettere all’utente di aggiungere un qualsivoglia gioco alla sua lista, selezionando una categoria a piacimento.

* **RF\_USER\_02 – Vota Gioco** –Priorità Alta

Il sistema deve permettere all’utente di poter aggiungere una valutazione ad un gioco.

* **RF\_USER\_03 – Commenta** –Priorità Bassa

Il sistema deve permettere all’utente di contribuire a una discussione tramite un commento.

* **RF\_USER\_03 – Like** –Priorità Bassa

Il sistema deve permettere all’utente di mettere “mi piace” a un commento.

* **RF\_USER\_04 – Richiesta Discussione** –Priorità Media

Il sistema deve permettere all’utente di richiedere la creazione di una discussione.

* **RF\_USER\_05 – Richiesta Aggiunta Gioco** –Priorità Alta

Il sistema deve permettere all’utente di richiedere l’aggiunta di un gioco ai gestori del catalogo.

* **RF\_USER\_06 – Segnalazione contenuto** –Priorità Bassa

Il sistema deve permettere all’utente di segnalare ai moderatori contenuti ritenuti inopportuni o non pertinenti**.**

### **RF\_NAV: NAVIGAZIONE**

* **RF\_NAV\_01 – Cerca Elemento** –Priorità Media

Il sistema permette all’utente di eseguire una ricerca testuale o tramite parametri sui giochi.

* **RF\_NAV\_02 – Visualizza Elemento** –Priorità Alta

Il sistema deve far visualizzare la pagina dedicata di un gioco con le relative informazioni.

### **RF\_ADMIN: AMMINISTRAZIONE**

* **RF\_ADMIN\_01 – Valuta Richiesta Discussione** –Priorità Media

Il moderatore deve poter valutare (accettare o rifiutare) richieste di creazione di discussioni da parte degli utenti.

* **RF\_ADMIN\_02 – Creazione Discussione** –Priorità Media

Il moderatore e gli sviluppatori di videogiochi devono poter creare discussioni relative a determinati giochi, senza che esse necessitino di approvazione.

* **RF\_ADMIN\_03 – Valuta Segnalazione**–Priorità Bassa

Il moderatore deve poter valutare segnalazioni di contenuti inappropriati da parte degli utenti.

* **RF\_ADMIN\_04 – Valuta Richiesta aggiunta gioco** –Priorità Alta

Il gestore catalogo deve poter valutare richieste per l’aggiunta di nuovi giochi al sito da parte degli utenti

* **RF\_ADMIN\_05 – Aggiungi o modifica Gioco** –Priorità Media

Il gestore del catalogo deve poter modificare o aggiungere giochi al sistema.

* **RF\_ADMIN\_06 – Rimuovi Elemento** –Priorità Media

Il moderatore deve poter rimuovere elementi creati dall’utenza che ritiene non opportuni.

## **3.3 Requisiti non funzionali**

### **3.3.1 Usabilità**

Il sistema deve:

* Permettere di accedere alle funzioni principali (ricerca di un gioco, accesso al profilo utente) in modo intuitivo e dalla homepage.
* Contenere una sezione “Aiuto” che funge da breve guida per l’utente.

### **3.3.2 Affidabilità**

Il servizio, in quanto sito web, dovrà essere online 24/7 con basso rischio di crash del servizio di hosting.

Inoltre, agli utenti non sarà richiesto di fornire dati sensibili in quanto non utili ai fini dell’utilizzo di GameSquare.

### **3.3.3 Prestazioni**

* Il tempo di risposta di una query deve essere basso, in condizioni medie minore o uguale a un secondo.
* Non c’è un limite realistico ai giochi che è possibile aggiungere al database.
* Non c’è un limite realistico ai giochi che l’utente può avere nella sua libreria.
* Il servizio deve funzionare correttamente su dispositivi con diverse risoluzioni e aspect ratio.

### **3.3.4 Manutenibilità**

* In futuro ci sarà, per ogni gioco, una sezione dei giochi consigliati.
* La rimozione dei bug sarà a carico degli sviluppatori.
* Le modifiche e le migliorie saranno a carico di sviluppatori e progettisti, in base alle direttive e alle richieste dell’utenza.

### **3.3.5 Implementazione**

* La parte back-end deve essere programmata in Java.
* Il database deve essere programmato in SQL.
* La parte front-end deve essere strutturata tramite CSS e JavaScript.

### **3.3.6 Interfaccia**

* Il sistema deve interagire con un database di videogiochi importandone il contenuto.
* Il contenuto del database sarà gestito dalle API fornite dallo stesso.
* Il sito si affiderà a un sito di hosting per essere online.

### **3.3.7 Packaging**

Il sistema, in quanto sito web dovrà essere supportato da tutti i browser più usati.

### **3.3.8 Legali**

NA

## **3.4 Modelli di sistema**

### **3.4.1 Scenari**

#### **-Scenario 1:**

***Nome Scenario:*** ricercaTorneo

***Attori Partecipanti:*** Nico:utente

***Descrizione:*** Nico, un giocatore di Tekken 7, vuole partecipare al suo primo torneo competitivo online.

***Flusso di eventi:***

1. Si reca sulla homepage di GameSquare e cerca il gioco interessato.
2. Sulla pagina del gioco in questione e clicca sulla sezione “discussioni”.
3. Filtra le discussioni per argomento selezionando “competitivo”.
4. Nota varie organizzazioni di tornei online, fra cui una per giocatori con poca esperienza in tale ambito.
5. Esprime il suo desiderio di partecipare in quanto orari e modalità gli sono convenienti.
6. L’autore del thread risponde a Nico e si mettono d’accordo sulle modalità di partecipazione.

#### **-Scenario 2:**

***Nome Scenario:*** communityFeedback

***Attori Partecipanti:*** Lorenzo:utente

***Descrizione:*** Lorenzo, uno sviluppatore indipendente, ha pubblicato il suo primo videogioco da pochi giorni e sta cercando del feedback a riguardo.

***Flusso di eventi:***

1. Cerca il proprio gioco sulla piattaforma, visualizzandone la pagina.
2. Nota che l’85% delle persone che lo hanno acquistato lo hanno anche giocato, indice di alta curiosità da parte del pubblico.
3. Nota anche che il 70% delle persone che lo hanno giocato lo hanno anche completato, indice forse di scarsa longevità.
4. Si dirige poi nella sezione discussioni, dove nonostante un generale elogio ha conferma della scarsa difficoltà del titolo.
5. Comincia, infine, egli stesso una discussione dove si annuncia come creatore del titolo in questione e richiede consigli e feedback direttamente ai propri acquirenti.

#### **-Scenario 3:**

***Nome Scenario:*** richiestaAggiuntaGioco

***Attori Partecipanti:*** Francesco:utente

***Descrizione:*** Francesco, un appassionato di videogiochi meno conosciuti, non riesce a trovare su GameSquare il titolo “Garage: Bad Dream Adventure”.

***Flusso di eventi:***

1. Si reca sull’apposita pagina contenente il form per proporre l’aggiunta di un gioco.
2. Inserisce i requisiti, ossia il nome del gioco e una fonte attendibile della sua esistenza, in questo caso un riferimento a un video del gameplay caricato per intero su YouTube.
3. Dopo qualche giorno, la sua proposta viene approvata e il gioco aggiunto al database.
4. Francesco aggiunge il gioco al suo catalogo di titoli completati.

#### **-Scenario 4:**

***Nome Scenario:*** eliminazioneContenuto

***Attori Partecipanti:*** Piero:admin

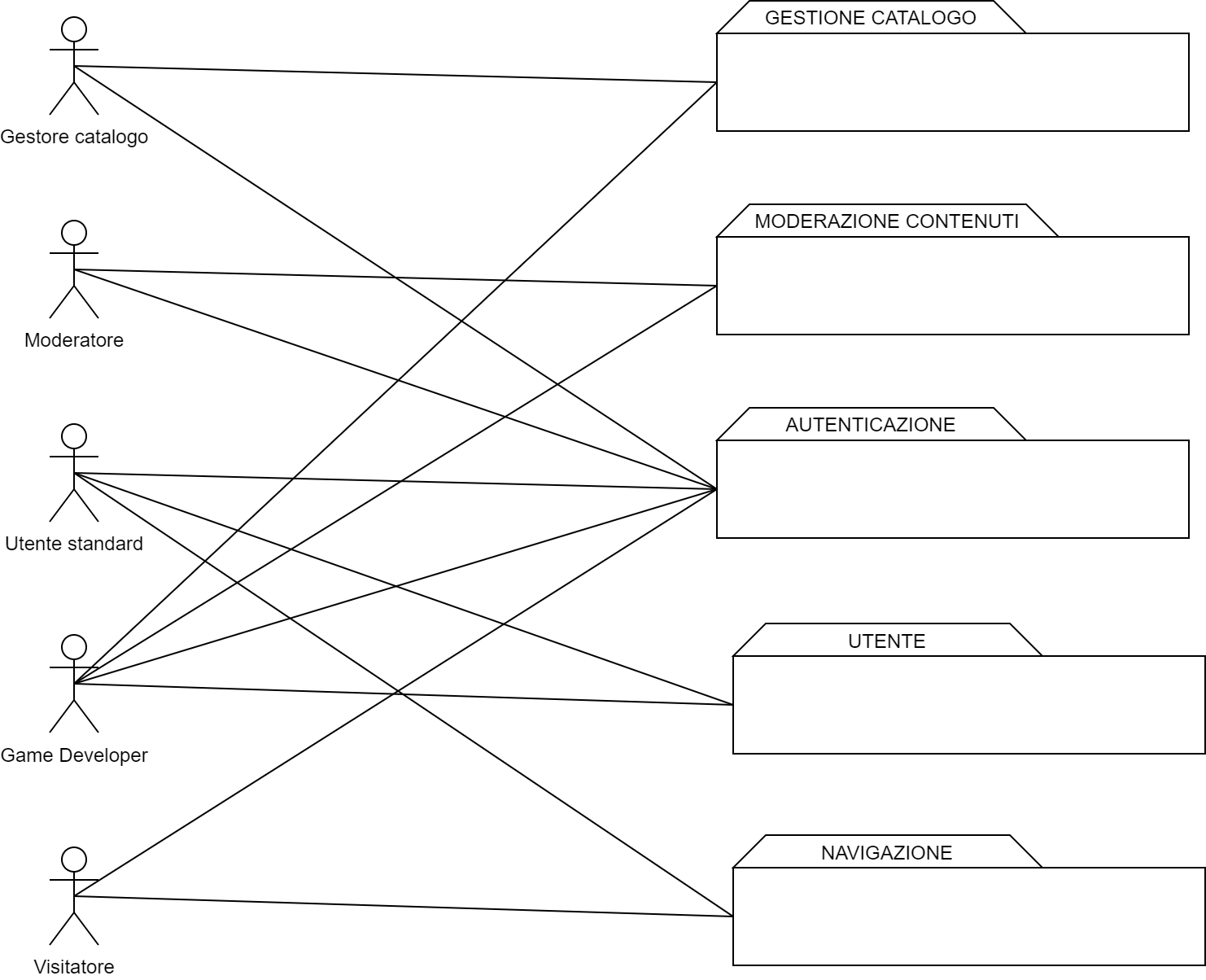
***Descrizione:*** Piero, un admin di GameSquare, nota che un contenuto è inopportuno.

***Flusso di eventi:***

1. Molti utenti hanno segnalato una discussione come non opportuna, a causa della mancanza di contributi significativi alla community e all’eccessiva volgarità.
2. L’admin nota le molte segnalazioni e decide di rimuovere il contenuto nocivo.
3. Tramite un apposito tasto elimina il contenuto inviando una segnalazione privata all’utente che lo ha creato incoraggiandolo a soppesare meglio le proprie parole la prossima volta.

### **3.4.2 Use Case Model**

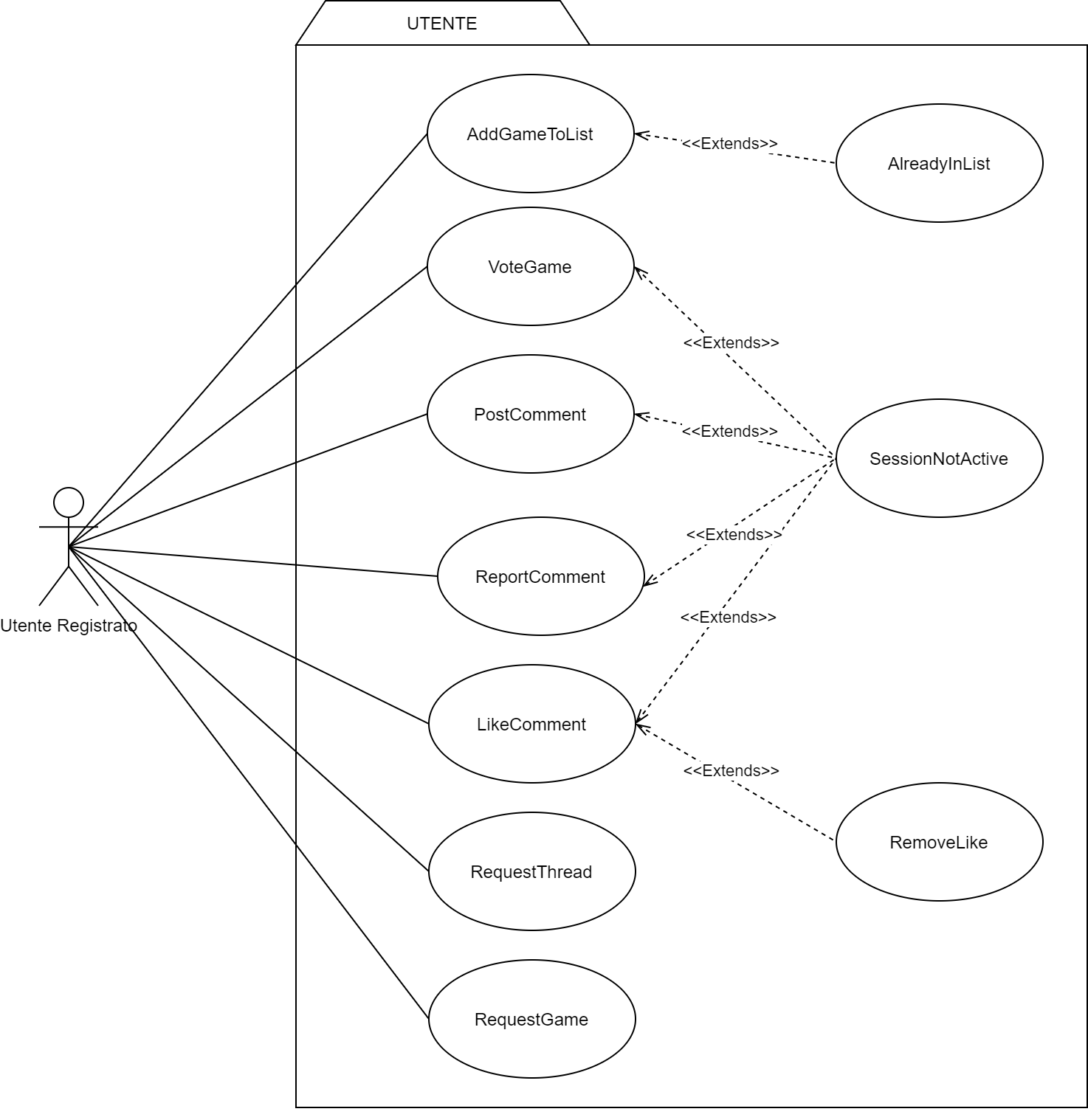
#### **-UCD\_GENERALE**



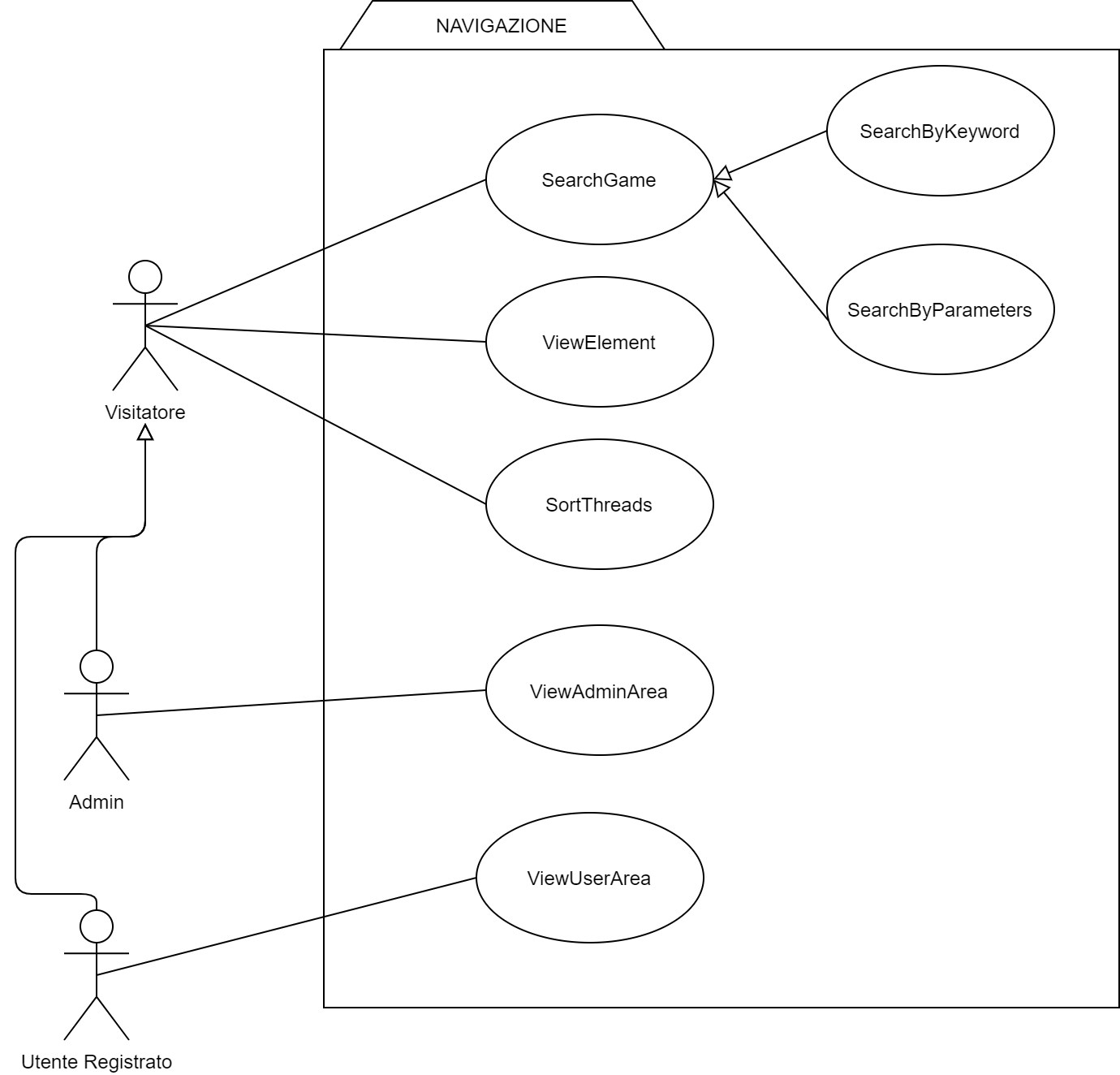
#### **-UCD\_ADMIN**



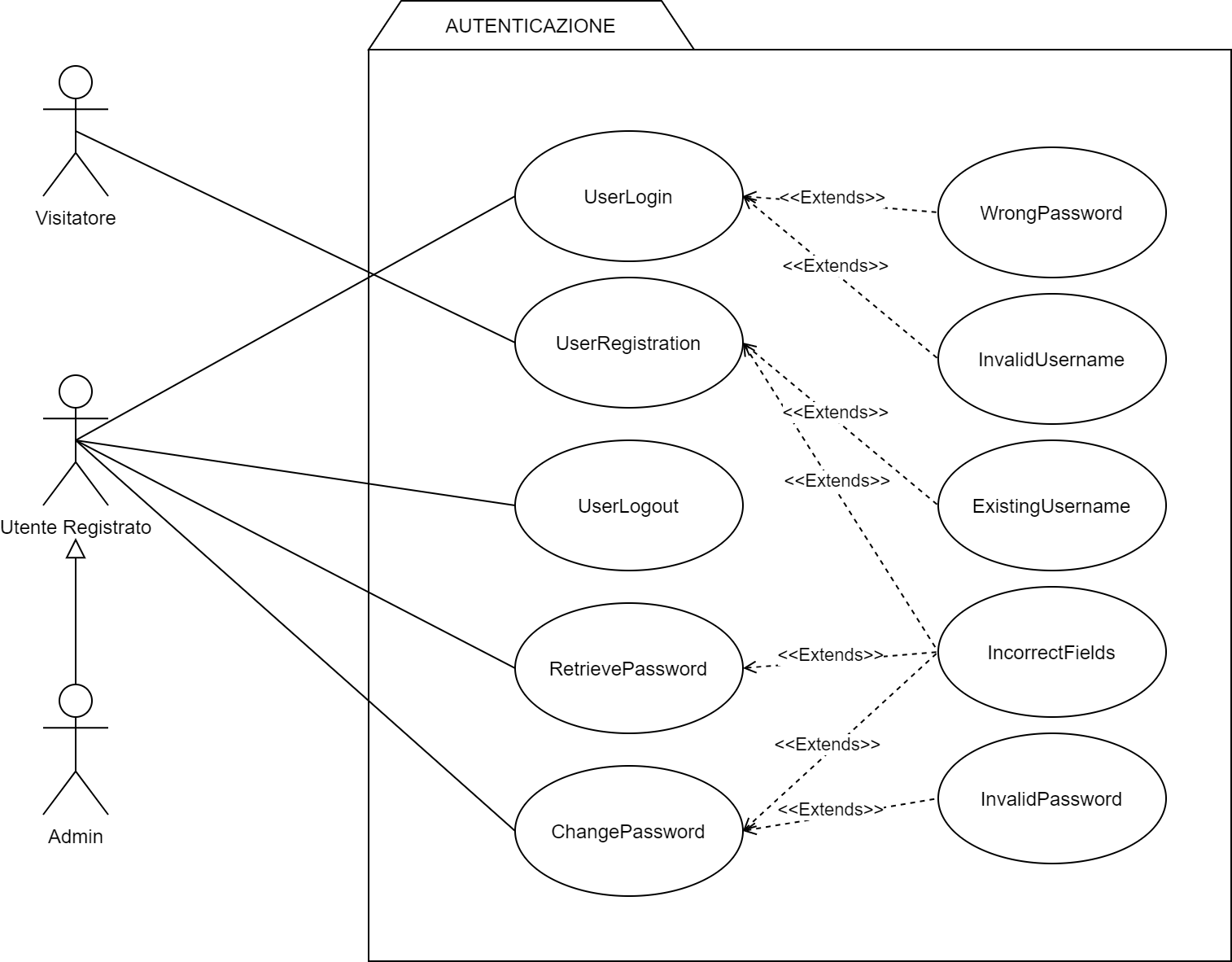
#### **-UCD\_USER**



#### **-UCD\_NAV**



#### **-UCD\_AUT**



### 

### **3.4.3 Object Model**

|  |  |  |
| --- | --- | --- |
| Nome oggetto | Tipologia | Descrizione |
| Registration\_Boundary | Boundary | Form da compilare per la registrazione, con campi da riempire e submit |
| Login\_Boundary | Boundary | Pagina per effettuare il login da utente registrato o da admin |
| Notification\_boundary | Boundary | Notifica a schermo che appare per segnalare un cambiamento o un errore |
| Homepage | Boundary | Pagina principale visualizzata se non si ha una sessione attiva |
| LoggedHomePage | Boundary | Pagina principale a cui si viene reindirizzati una volta attivata la sessione |
| UserAreaButton | Boundary | Area accessibile dall’utente registrato, con tutte le informazioni su di esso |
| Like\_Boundary | Boundary | Icona che viene creata quando viene messo like a un commento da un utente; se cliccata permetterà di rimuoverlo |
| Comment\_boundary | Boundary | Uno specifico commento in una discussione |
| SearchBar\_boundary | Boundary | Barra di ricerca per effettuare una ricerca di giochi per parola chiave |
| GameSort\_Boundary | Boundary |  |
| ResultsPage | Boundary | Pagina che mostra i risultati di una ricerca effettuata nel sito |
| GameLink\_boundary | Boundary | Link che reindirizza alla pagina di uno specifico gioco |
| GamePage | Boundary | Pagina che mostra tutte le informazioni relative ad un determinato gioco, oltre a consentire le operazioni sullo stesso, da utente registrato |
| ThreadPage | Boundary | Pagina che mostra una determinata discussione relativa ad un gioco, con tutti i commenti a seguito |
| ThreadForm | Boundary | Form da compilare per richiedere una nuova discussione (utente registrato) o crearne una (admin) |
| GameRequestForm | Boundary | Form con cui l’utente registrato può richiedere l’aggiunta di un nuovo gioco |
| ReportRequestForm | Boundary | Form con cui l’utente registrato può richiedere la segnalazione di un commento inopportuno |
| Logout\_Boundary | Boundary | Permette di terminare la propria sessione attiva |
| RetrievePassword\_boundary | Boundary | Form da compilare per indicare l’indirizzo e-mail presso il quale inviare il link di recupero della password |
| PasswordModifier\_boundary | Boundary | Form da compilare per la modifica della password |
| ThreadRequestPage | Boundary | Pagina accessibile ai moderatori con le richieste di nuove discussioni da parte degli utenti |
| GameRequestPage | Boundary | Form che permette la richiesta di un nuovo gioco |
| ReportRequestPage | Boundary | Form che permette la segnalazione di un commento |
| NewGame\_boundary | Boundary | Form che permette all’admin di inserire un nuovo gioco nel database |
| GameUpdaterForm | Boundary | Form che permette all’admin di aggiornare i dati relativi ad un determinato gioco |
| Registration\_Control | Control | Si occupa di controllare il corretto inserimento dei campi in un form per quando riguarda la registrazione di un nuovo utente e la sua effettiva operazione |
| Login\_Control | Control | Si occupa di tutte le operazioni di controllo per il login di un utente registrato |
| Logout\_Control | Control | Si occupa di tutte le operazioni di controllo per il logout di un utente registrato avente una sessione attiva |
| RetrievePassword\_Control | Control | Si occupa di tutte le operazioni di controllo per la richiesta del recupero della password di un utente registrato |
| ChangePassword\_Control | Control | Si occupa di tutte le operazioni di controllo per la richiesta da parte di un utente registrato avente sessione attiva di cambiare la password |
| Search\_Control | Control | Consente al sistema di effettuare una ricerca tra tutti i giochi o thread presenti nel sistema secondo i parametri o la keyword inserita dall’utente, oppure per un gioco specifico nel caso di indirizzamento tramite link |
| Game\_Control | Control | Si occupa di tutte le operazioni di controllo per l’aggiunta di un nuovo gioco da parte di un admin e per la sua creazione e/o update |
| Request\_Control | Control | Si occupa di tutte le operazioni di controllo per la richiesta da parte di un utente di aggiungere un nuovo gioco, di creare un nuovo thread riguardo un gioco oppure di segnalare un commento o thread sgradito |
| Thread\_Control | Control | Si occupa di tutte le operazioni di controllo per l’aggiunta di un nuovo thread da parte di un admin e per la sua creazione |
| Vote\_control | Control | Si occupa di tutte le operazioni di controllo per la votazione di un qualsiasi gioco aggiunto ad una lista da parte di un utente |
| Comment\_Control | Control | Si occupa delle operazioni di controllo per l’aggiunta di un commento a una specifica discussione da parte dell’utente registrato |
| Like\_control |  | Si occupa delle operazioni di controllo per l’aggiunta di un like a un commento, come la verifica della sessione dell’utente e se un like era già stato inserito da un utente specifico |
| List\_Control | Control | Consente al sistema di organizzare le liste di un determinato utente, verificando la presenza di oggetti ed eventualmente aggiungendoli, togliendoli o spostandoli da una lista all’altra, calcolando conseguentemente il punteggio associato |
| Game | Entity | Rappresenta un singolo gioco |
| GameSet | Entity | Rappresenta un insieme di giochi |
| User | Entity | Rappresenta un singolo utente registrato, con tutte le sue informazioni e il suo punteggio abilità |
| Moderatore | Entity | Rappresenta quel tipo di admin che si occupa dell’aspetto social del sito. Quindi creazione e rimozione di discussioni e/o commenti, segnalazioni e richieste di nuove discussioni |
| Sviluppatore | Entity | Rappresenta un user con più privilegi. Può creare discussioni e aggiungere giochi senza richiederli tramite un form e ha una sua categoria nell’area utente dedicata ai giochi sviluppati |
| Gestore Catalogo | Entity | Rappresenta quel tipo di admin che si occupa di gestire il catalogo di giochi di GameSquare, accettare o rifiutare richieste di nuovi giochi, aggiungerli, modificarli o eliminarli |
| List | Entity | Rappresenta una lista di giochi di un determinato utente |
| Vote | Entity | Rappresenta un voto aggiunto da un utente a uno specifico gioco |
| Like | Entity | Rappresenta un like inserito da un utente in un commento specifico |
| Comment | Entity | Rappresenta un commento aggiunto da un giocatore in una determinata discussione |
| Thread | Entity | Rappresenta una singola discussione creata da un utente per un gioco |
| ThreadSet | Entity | Rappresenta un insieme di discussioni per un gioco |
| GameRequest | Entity | Rappresenta una richiesta per l’aggiunta di un nuovo gioco inviata a un admin da un utente registrato |
| ThreadRequest | Entity | Rappresenta una richiesta per l’aggiunta di un nuovo thread inviata a un admin da un utente registrato |
| ReportRequest | Entity | Rappresenta una segnalazione di un commento o un thread inviata a un admin da un utente registrato |
| User\_manager | Manager | Si occupa della gestione dei dati persistenti relativi all’entità “User” |
| Game\_manager | Manager | Si occupa della gestione dei dati persistenti relativi all’entità “Game” |
| Thread\_manager | Manager | Si occupa della gestione dei dati persistenti relativi all’entità “Thread” |
| List\_manager | Manager |  |
| Vote\_manager | Manager |  |
| Comment\_manager | Manager |  |
| Like\_manager | Manager |  |
| GameRequest\_manager | Manager |  |
| ThredRequest\_manager | Manager |  |
| ReportRequest\_manager | Manager |  |
| Session\_manager | Manager |  |
| Field\_manager | Manager |  |

#### **3.4.3.1 Class Diagrams**

##### **- CD\_AUT**



##### **- CD\_NAV**



##### **- CD\_USER**



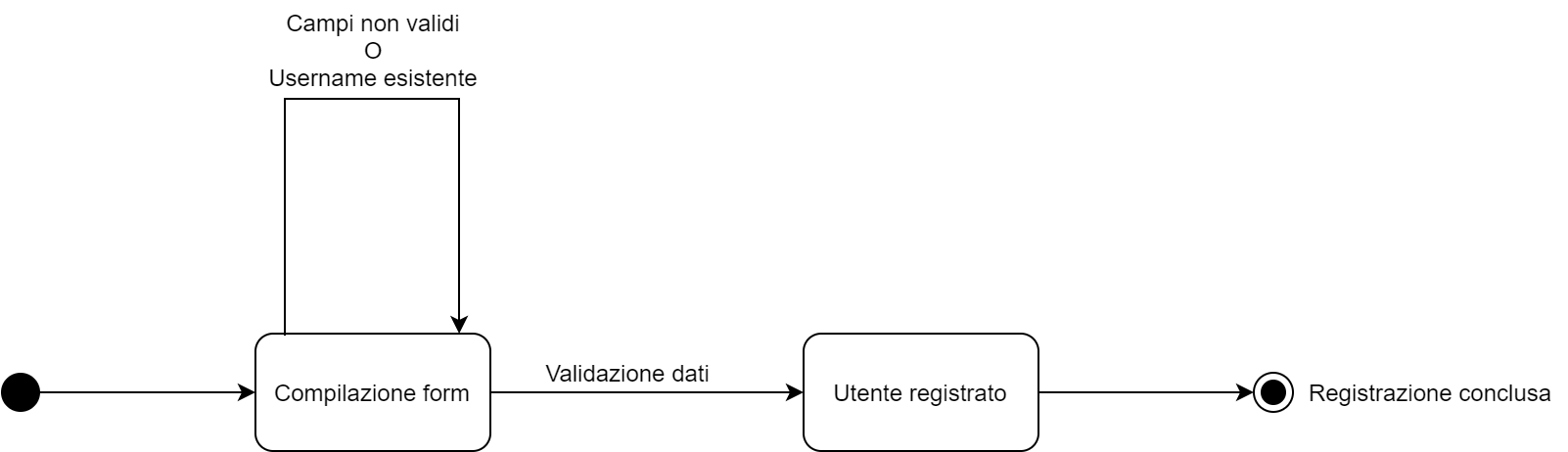
##### **- CD\_ADMIN**



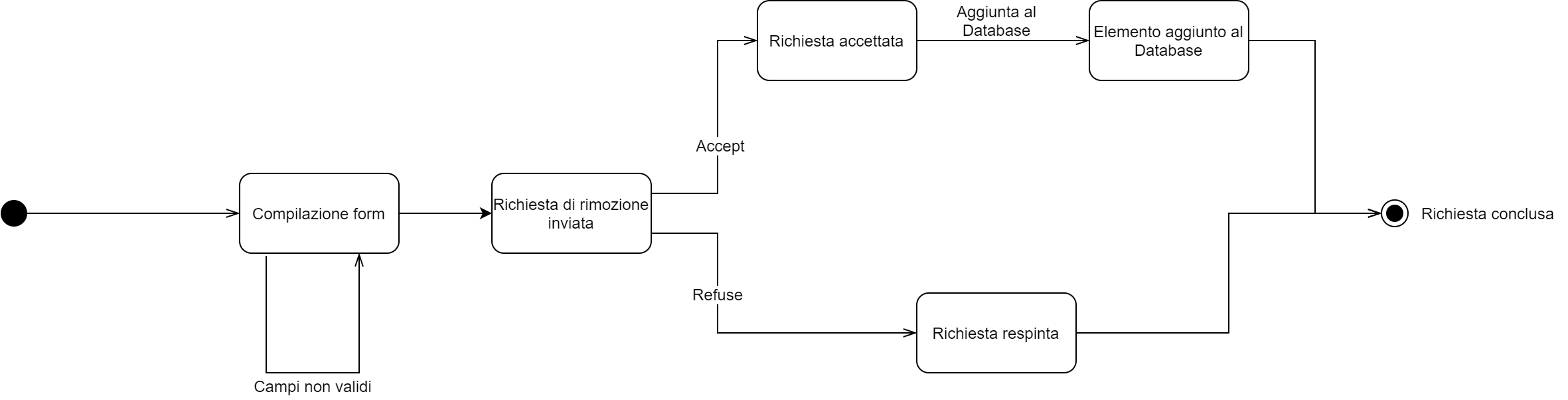
### **3.4.4 Dynamic Model**

#### 3.4.4.1 Statechart Diagrams

##### **- STATECHART REGISTRAZIONE UTENTE:**



##### **- STATECHART AGGIUNTA ELEMENTO:**



##### **- STATECHART RIMOZIONE ELEMENTO:**

