

## *PIATTAFORMA GAMING*

# **PROGETTO OBJECT-ORIENTED SOFTWARE DESIGN**

---

Studenti:

Davide Petrini	244311
Marco Pistilli	244308

# OBIETTIVO

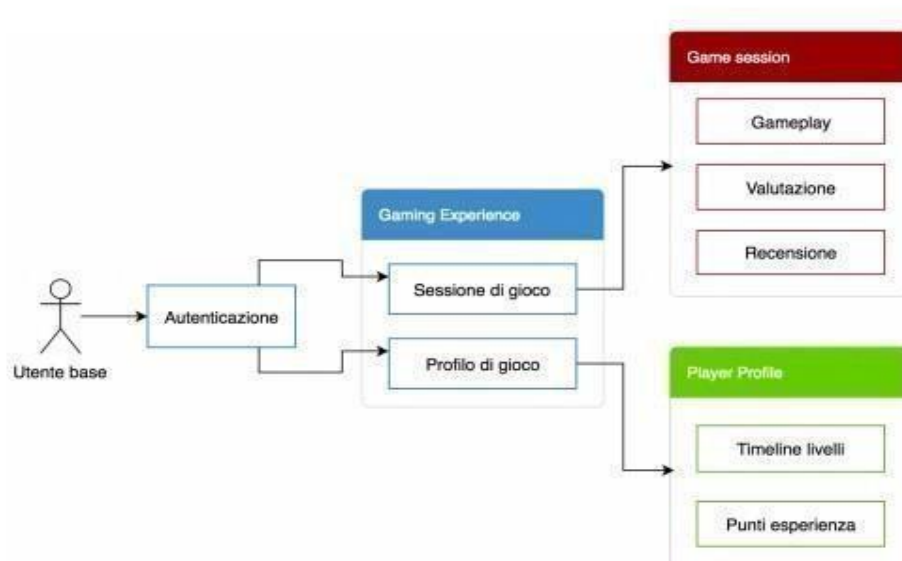
Il progetto si propone di realizzare una piattaforma gaming che coinvolga gli utenti a giocare al fine di guadagnare punti esperienza e nuovi livelli di gioco.

## ATTORI DEL SISTEMA

### UTENTE

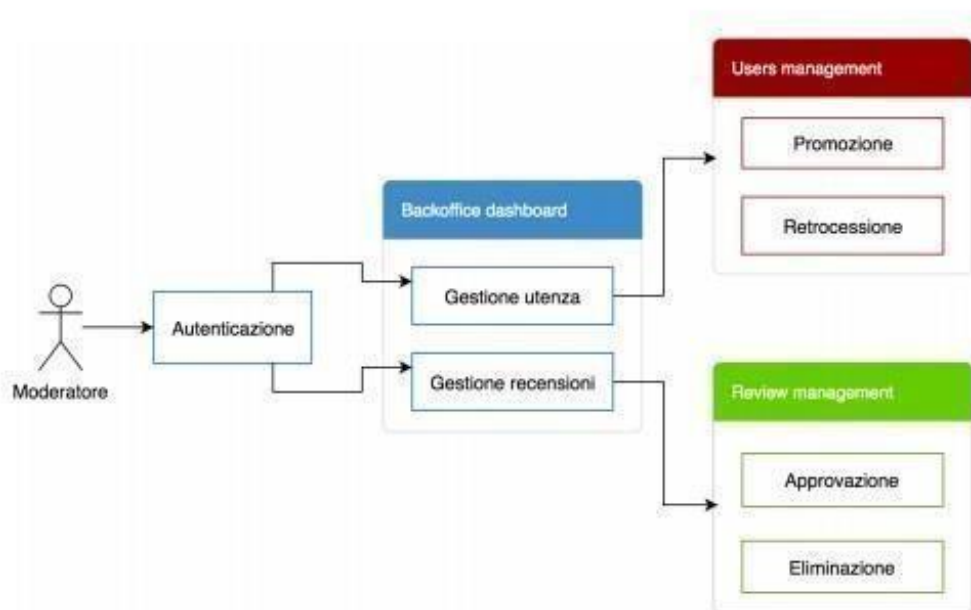
Attore che può utilizzare il sistema solo dopo essersi registrato e dopo avere eseguito l'accesso tramite il login. L'utente può controllare e modificare le sue informazioni anagrafiche oppure visualizzare i suoi progressi (livello e trofei).

Per aumentare i suoi progressi l'utente accede ad una lista di giochi e ne sceglie uno a suo piacimento che in seguito può valutare con recensioni o voti.



### MODERATORE

Il moderatore è un attore a cui è consentito eseguire le stesse attività dell'utente, ma ha dei privilegi. In particolare, può accettare o rifiutare le recensioni effettuate dagli utenti e promuovere utenti a moderatori o retrocedere moderatori a utenti.



## AMMINISTRATORE

L'amministratore, a differenza degli altri attori, non prende parte ai giochi dunque non possiede livelli o trofei.

Il suo compito è quello di garantire l'efficienza del sistema, interagire con il database, promuovere a moderatore gli utenti, retrocederli o eliminarli.

# REQUISITI

---

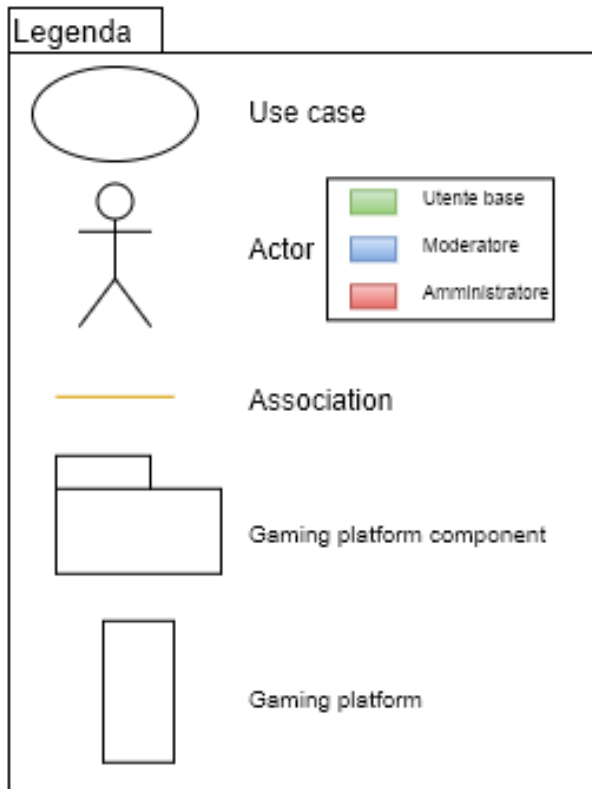
## ELENCO DEI REQUISITI

- *REGISTRAZIONE*: consente ad un utente di registrarti al sistema
- *AUTENTICAZIONE*: permette di effettuare il login da un utente registrato
- *VISUALIZZAZIONE PROFILO*: consente all'utente autenticato di visualizzare le informazioni personali e la sezione gaming
- *LISTA GIOCHI*: elenco di giochi tra cui l'utente può scegliere per giocare
- *SESSIONE DI GIOCO*: funzionalità connesse al gioco raggruppate come segue:
  - Partecipazione al gioco: consente di scegliere il gioco desiderato;
  - Attribuzione esperienza: assegnazione punti EXP per ogni partita di un determinato gioco effettuato dall'utente;
  - Aumento del livello: feature che, raggiunti i punti EXP necessari aumenta il livello dell'utente;
  - Otteniamo achievements: ogni volta che si raggiunge un nuovo livello viene assegnato un premio all'utente;
- *AREA VALUTAZIONE*: consente all'utente di recensire un gioco che può essere espressa sotto forma di:
  - Voto: valutazione numerica
  - Recensione: valutazione tramite commento
  - Validazione: funzionalità consentita solo ai moderatori che consiste nel accettare o rifiutare una recensione di altri utenti

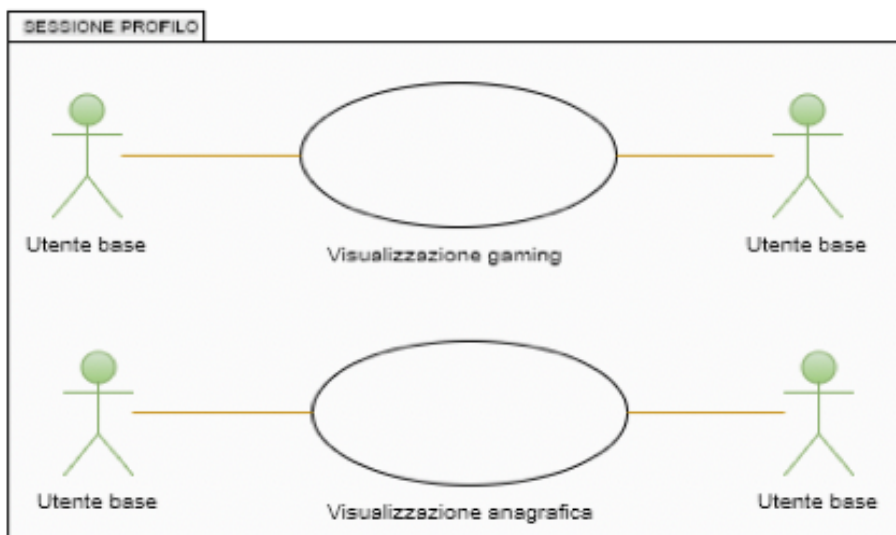
# USES CASES

Tecnica che valuta ogni requisito focalizzandosi sugli attori che interagiscono col sistema. Sono rappresentati in modello UML tramite gli *Use Cases Diagrams*.

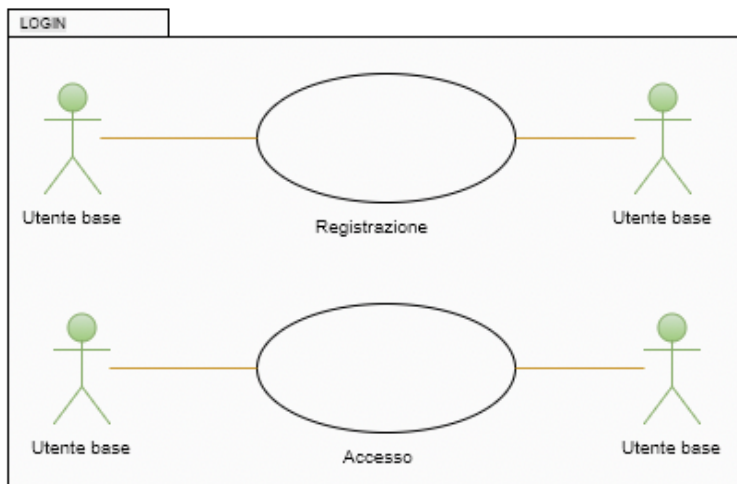
## LEGENDA



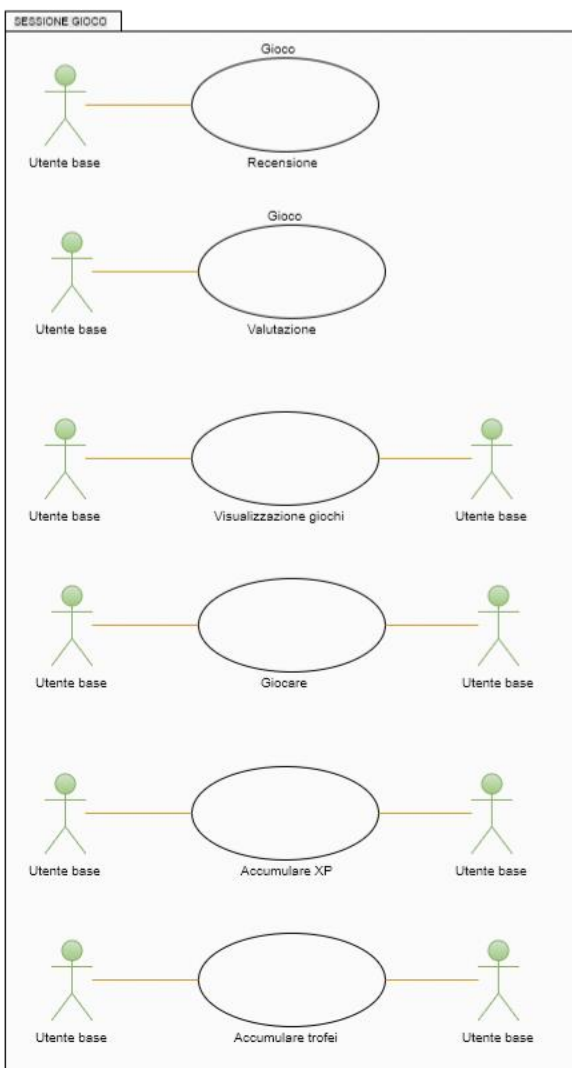
## SESSIONE PROFILO



## LOGIN

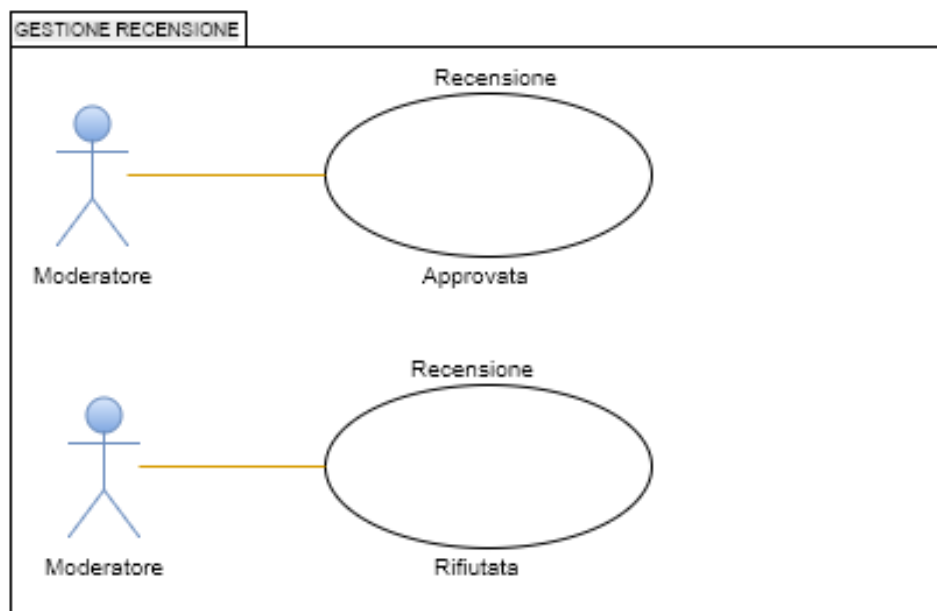
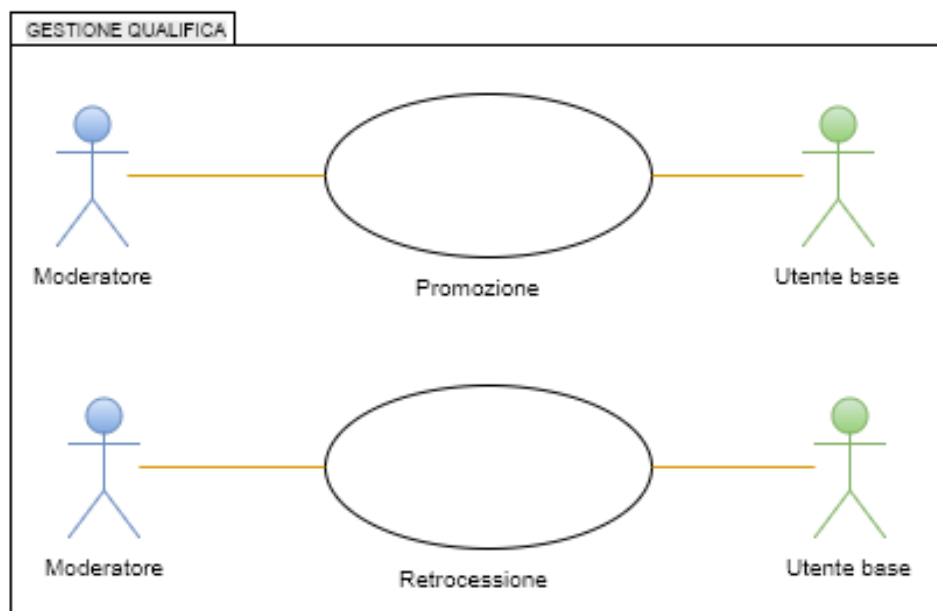


## SESSIONE GIOCO

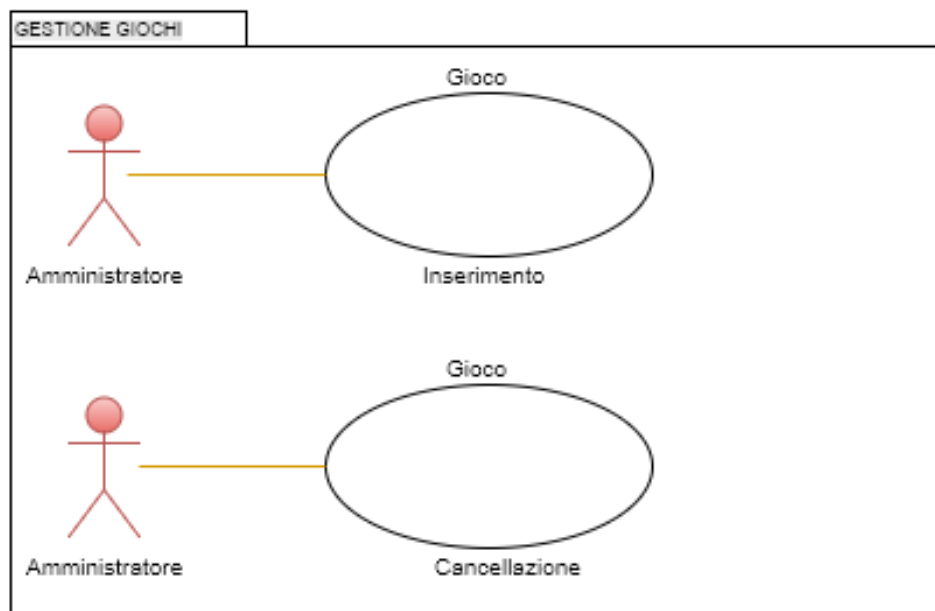
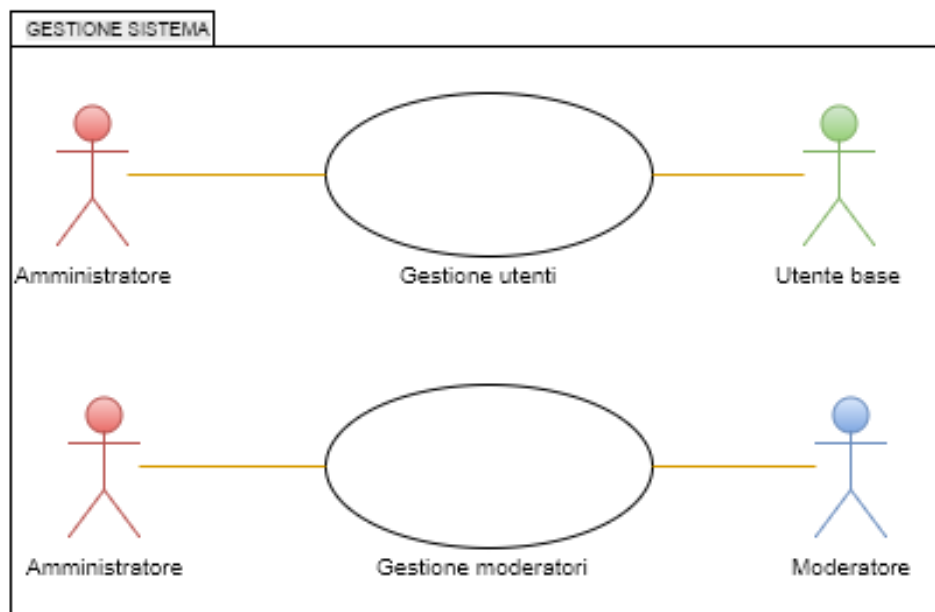


## MODERATORE

Il moderatore, oltre a possedere i cases dell'Utente base, è esteso con i seguenti:

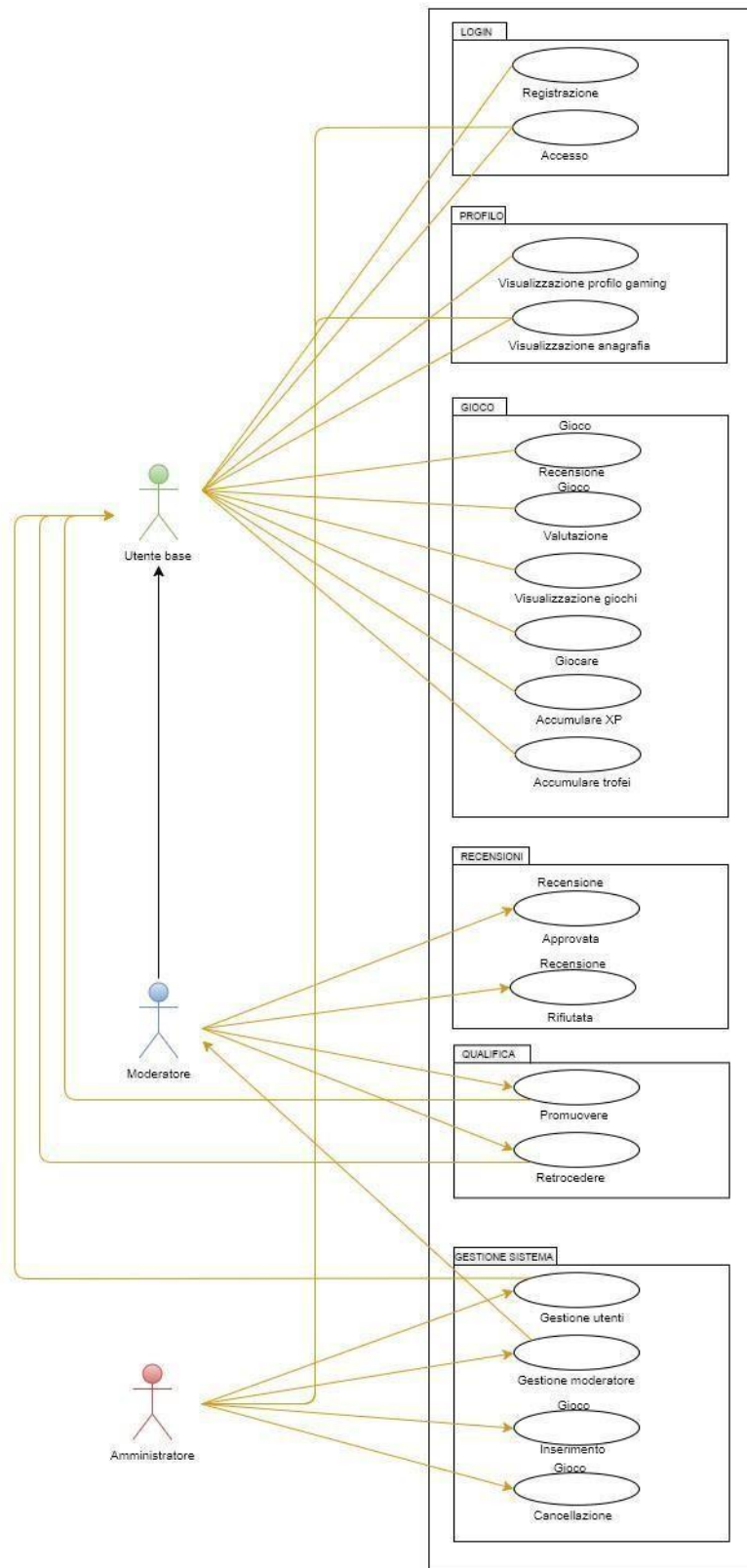


## AMMINISTRATORE





# GAMING PLATFORM



## SCENARI:

**REGISTRAZIONE:** un utente che vuole entrare a far parte della piattaforma ha la possibilità di registrarsi, inserendo dati anagrafici e username con la quale sarà riconosciuto in gioco.

**LOGIN:** per accedere alla piattaforma, un qualsiasi attore dovrà inserire username e password compilati nella registrazione.

**PROFILO PERSONALE:** ogni attore può visualizzare il proprio profilo personale, che registra i dati anagrafici dello stesso; in questa sezione si potranno modificare i dati inseriti.

**PROFILO GAMING:** viene associato un profilo gaming ad ogni moderatore e utente base dove vengono registrati i progressi ottenuti nella piattaforma. In particolare, il livello, punti esperienza, trofei e la timeline.

**GAMING:** lista dei giochi a cui moderatori e utenti base possono accedere. Per effettuare una partita è sufficiente cliccare sul tasto *Gioca*.

Dalla lista giochi è possibile anche effettuare la recensione dei giochi e visualizzarne la media dei voti.

**LIVELLO:** il livello di un utente base o moderatore aumenta in base all'esperienza fornita dal gioco al quale si partecipa.

**RECENSIONE:** una recensione, prima di essere pubblicata, deve essere approvata da un moderatore. Le recensioni scritte dai moderatori sono pubblicate automaticamente.

**SISTEMA:** il funzionamento del sistema deve essere garantito dall'amministratore. La piattaforma deve funzionare correttamente sia per la sessione gaming sia per le attività di moderazione.

## SYSTEM DESIGN

---

### OVERVIEW

Il sistema sarà suddiviso nella parte server-side, composta dal database, e nella parte client-side, rappresentante l'applicazione che sarà utilizzata dagli attori.

---

### CLIENT-SIDE

In questa sezione il team si occupa della implementazione degli attori, della componente grafica dell'applicazione e della relazione client-server; si può perciò ricorrere al pattern MVC per implementare tale sistema.

Il Model View Controller è un pattern architetturale formato da, come suggerisce il nome, tre componenti:

- *Model*: contiene lo stato dell'applicazione, definisce la logica implementativa dei dati e delle operazioni che possono essere svolte su di essi; in Java, seguendo la politica della programmazione ad oggetti, incapsula le classi

che definiscono le parti principali della piattaforma.

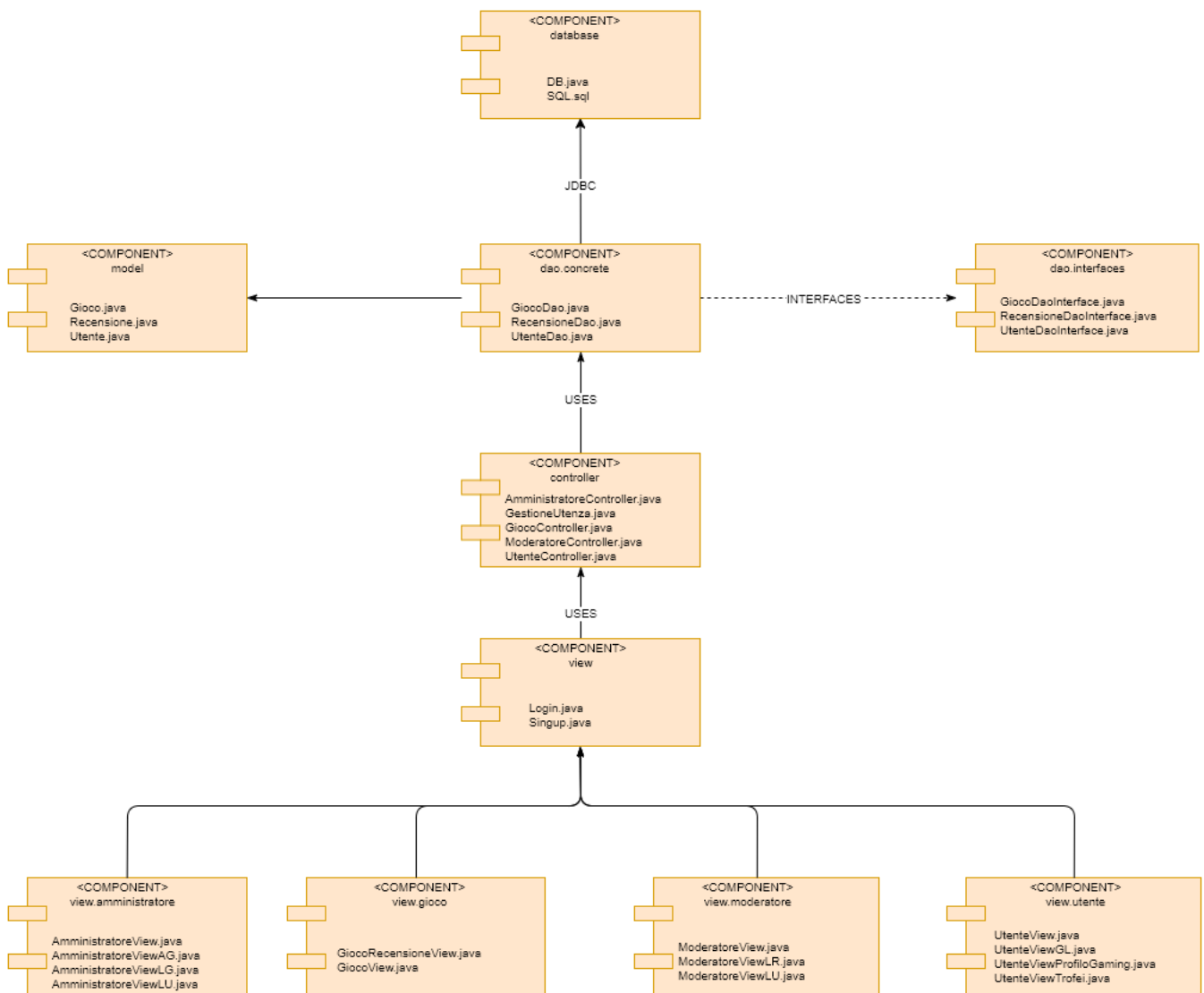
- *View*: gestisce l'output dei dati, quindi tutto ciò che concerne la parte grafica dell'applicazione. In Java, sarà principalmente sfruttato il framework Swing. Data la scelta della strategia *pull model*, la *View* richiede gli aggiornamenti quando lo ritiene necessario ed inoltre delega al *Controller* l'esecuzione dei processi richiesti dall'utente dopo averne catturato gli input e la scelta delle eventuali schermate da presentare.
- *Controller*: componente che gestisce le azioni dell'utente che attua sulla *View*, e le comunica al *Model*. Nella piattaforma, ha inoltre il compito di richiamare i metodi descritti nelle classi DAO, quest'ultime implementate per interagire con il database

Presentiamo quindi un altro pattern, il Data Access Object, per la gestione della persistenza dei dati. In Java è implementato come un insieme di interfacce che specificano i metodi di cui le classi concrete effettueranno l'overriding, composti da queries che verranno eseguite dal DBMS. Per permettere il collegamento fra classi DAO e DBMS è stato impiegato un *Java DataBase Connector*, un driver che appunto consente l'accesso e la gestione dei dati su un DB dal programma Java che lo utilizza, indipendentemente dal tipo di DBMS impiegato.

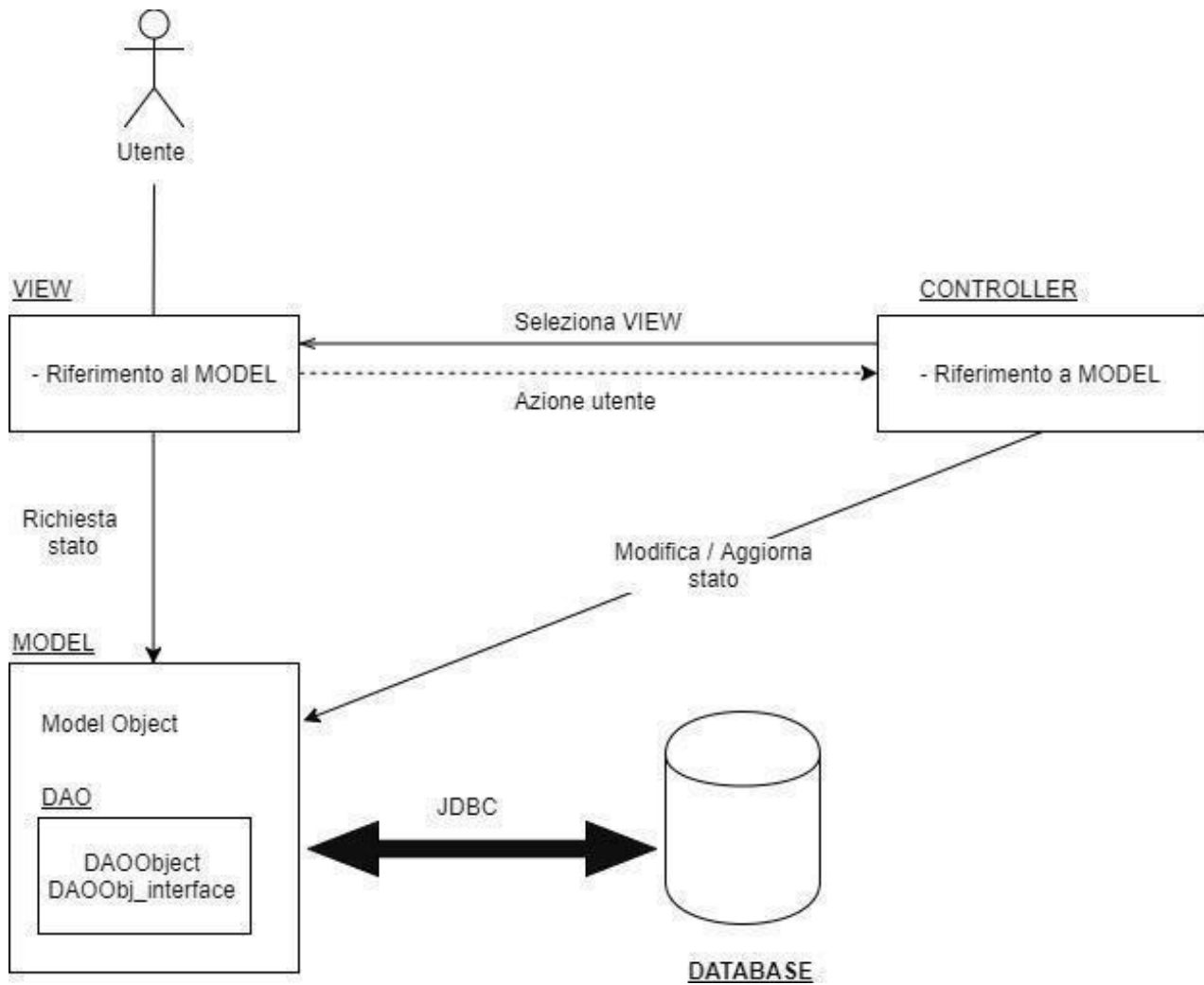
L'implementazione dei patterns descritti garantisce un'ottima manutenibilità del sistema e raggiungere una maggiore astrazione dei componenti; di conseguenza, l'architettura del sistema è la seguente

# MODELLO ARCHITETTURA DEL SISTEMA:

## COMPONENT DIAGRAM

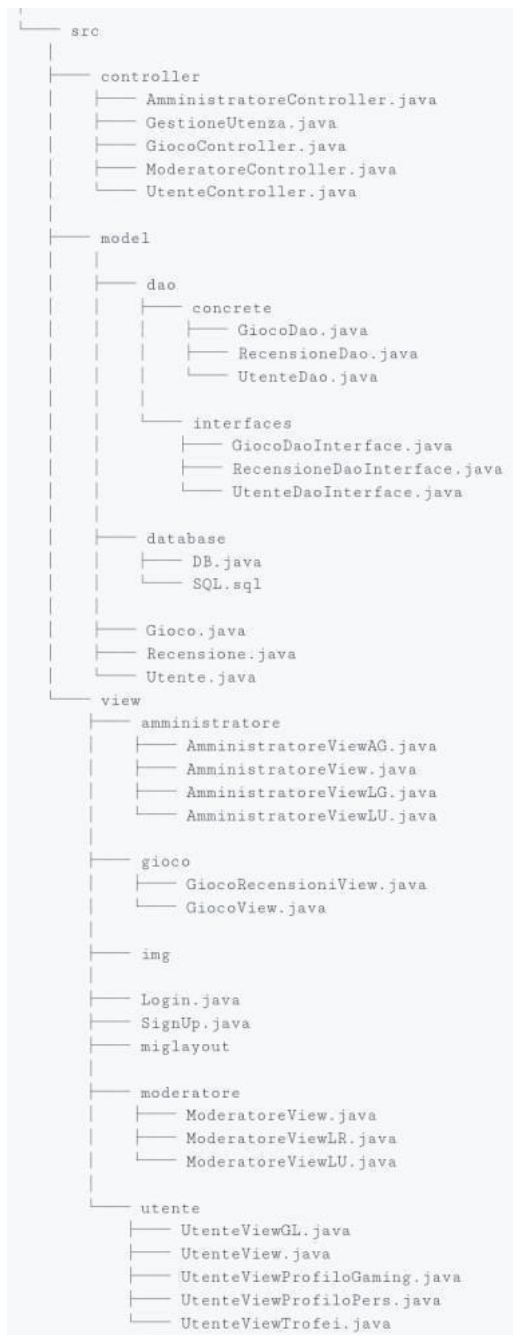


## GRAFICO DELLE INTERAZIONI FRA I SEGUENTI DESIGN PATTERNS



## SOFTWARE DESIGN

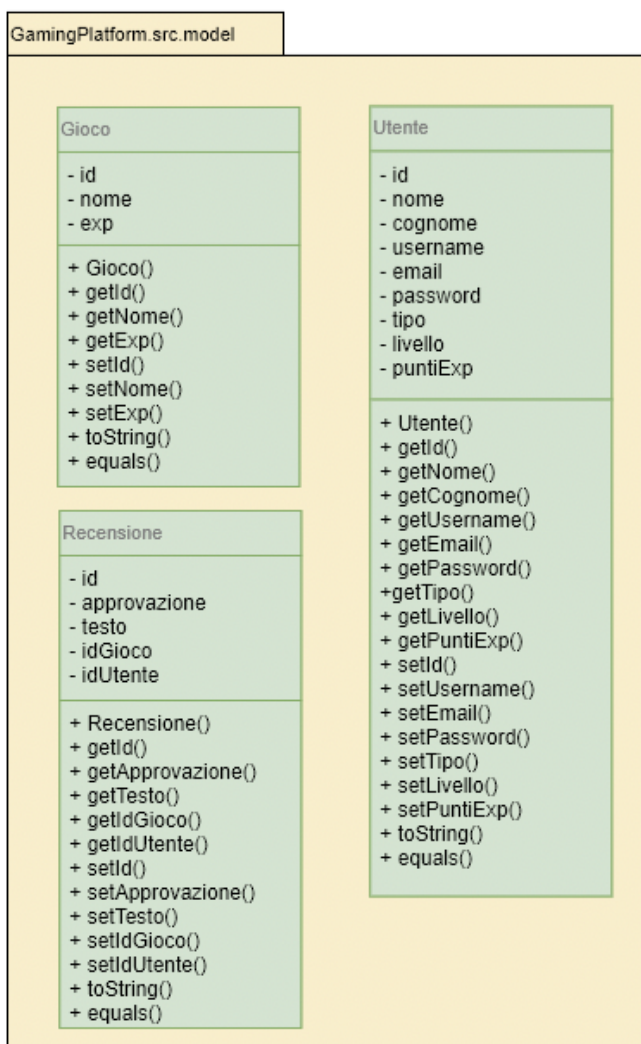
L'implementazione della piattaforma è stata ideata in una suddivisione in moduli, per ampliare la manutenibilità del sistema e separare logicamente i componenti; in Java sono stati quindi impiegati diversi packages che incapsulano le relative classi.



## CLASS DIAGRAMS

Sono un tipo di diagrammi di modello UML; li utilizziamo per rappresentare logicamente i package.

### MODEL'S CLASS DIAGRAM



Come da preambolo, il precedente package incapsula il componente *Model* del pattern MVC: esso include le tre classi principali della piattaforma

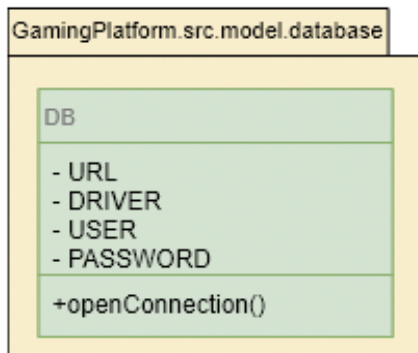
- *Gioco*: ogni istanza determina appunto un gioco presente nella piattaforma caratterizzato dall'ID, dal nome e dai



punti exp che fornisce. Sono a disposizione i metodi del costruttore, i getters e i setters, nonché il metodo *toString()* che ritorna informazioni relative al gioco

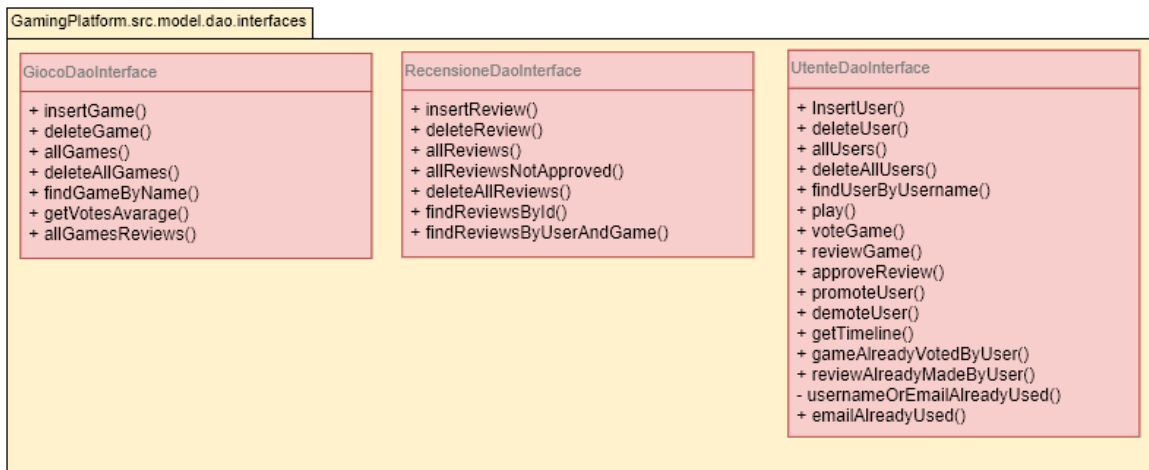
- *Utente*: anche per questa classe, ogni istanza rappresenta un utente registrato nel database; è composto dai dati anagrafici e dai dati relativi alla piattaforma gaming, particolare attenzione all'attributo *Tipo* che specifica la natura dell'utente, quindi se amministratore, moderatore o utente base. Come nella classe precedente, sono disponibili i metodi del costruttore, i getters e i setters, il metodo *toString()* ed *equals()* che controlla l'uguaglianza fra due Utenti
- *Recensione*: classe che modella il concetto logico di recensione; ha come attributi il proprio ID, un valore utilizzato dai moderatori per controllare se è stata approvata o meno, il testo e gli ID del giocatore che l'ha composta e il gioco che giudica. Anche qui notiamo il costruttore, i getters ed i setters, e il metodo informativo

## DB'S CLASS DIAGRAM



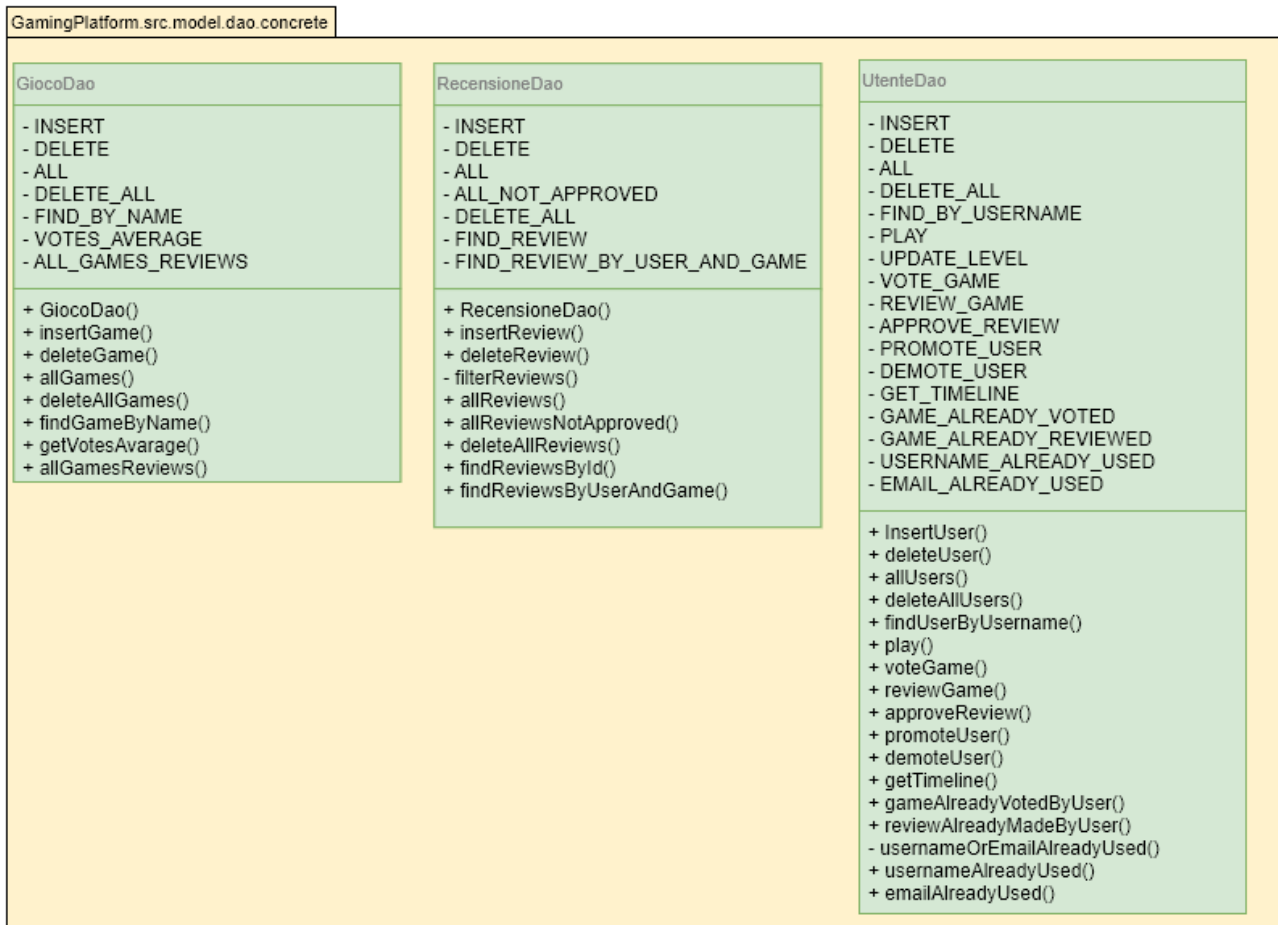
Classe adibita al DataBase, contiene l'URL, un riferimento al JDBC impiegato, i dati di autenticazione al DB e, di conseguenza, il metodo che stabilisce la connessione

## DAO'S CLASS DIAGRAM – INTERFACES



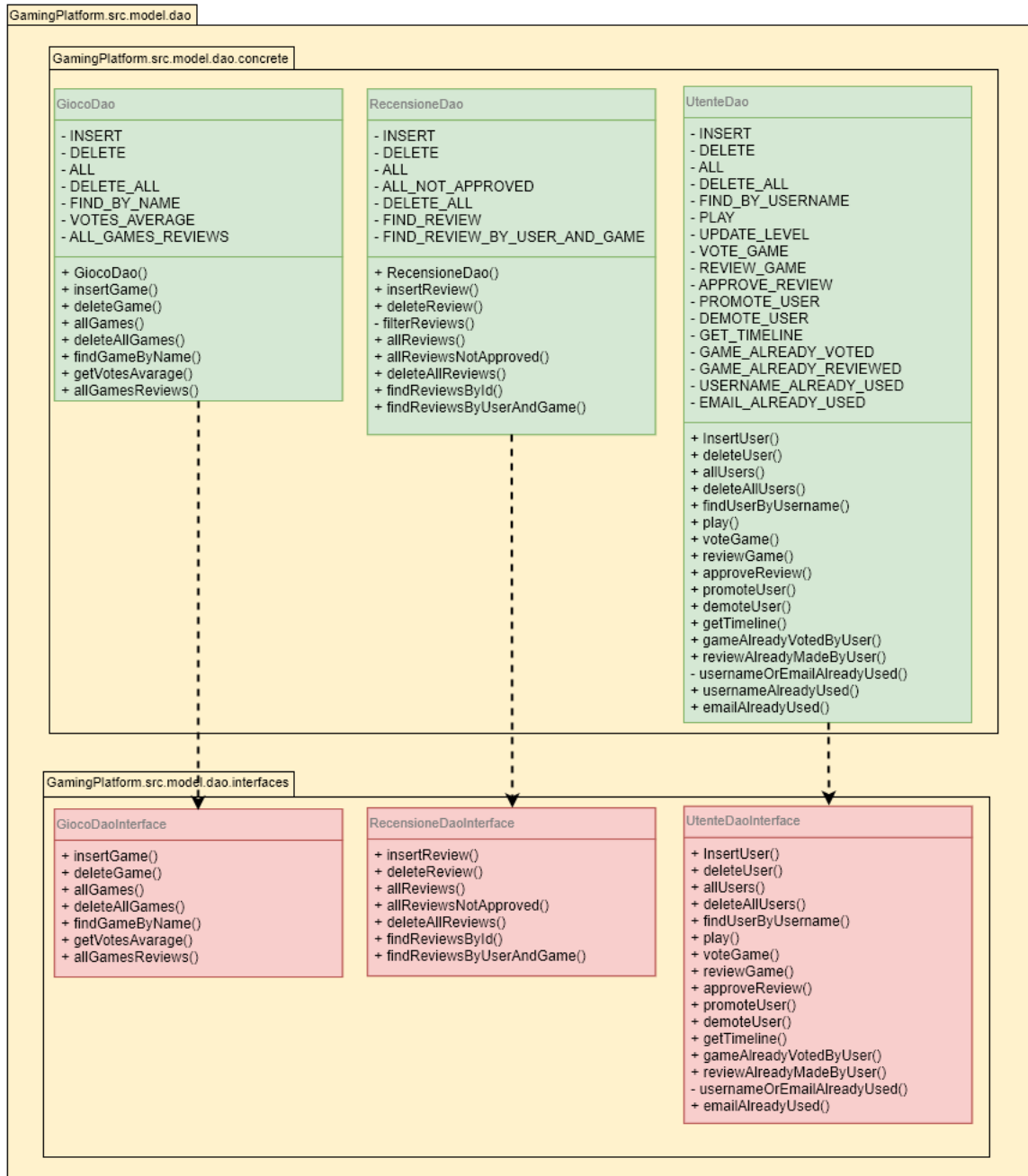
Package che contiene le interfacce delle classi implementate per il pattern DAO; ognuna di esse specifica i metodi sfruttabili dalle classi concretes che le implementeranno per effettuare queries al DB

## DAO'S CLASS DIAGRAM – CONCRETE



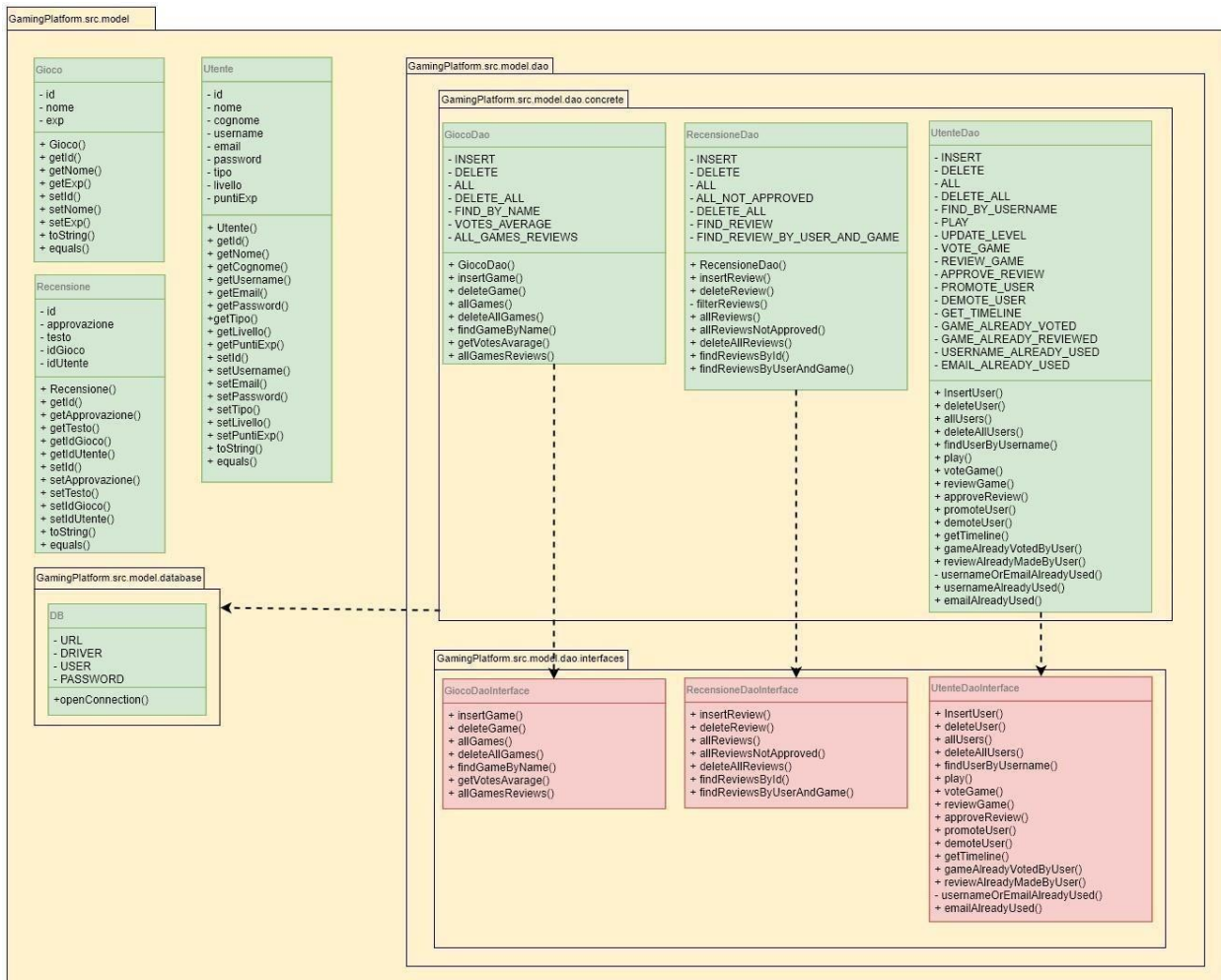
Come descritto nelle interfacce, queste classi implementano le DAO'S interfaces: negli attributi sono contenute le queries utili alla gestione del sistema, che saranno richiamate dai metodi specifici definiti nelle classi stesse

## DAO'S CLASS DIAGRAM



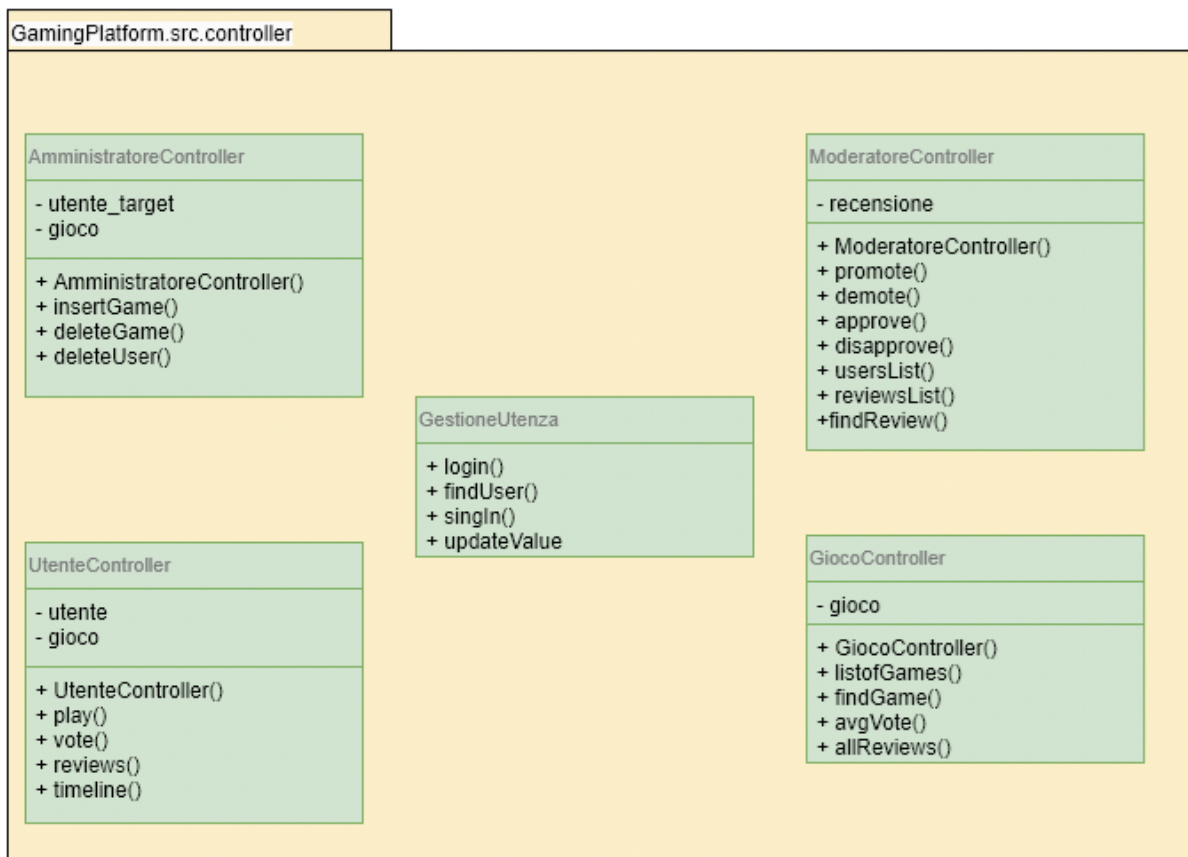
Class diagram che sottolinea il collegamento fra interfacce e classi concrete.

## INTERAZIONI DAO-DB



Ling tra componenti DAO e DB: chiaramente, per eseguire le suddette queries è necessario passare per il canale definito dalla classe *database*.

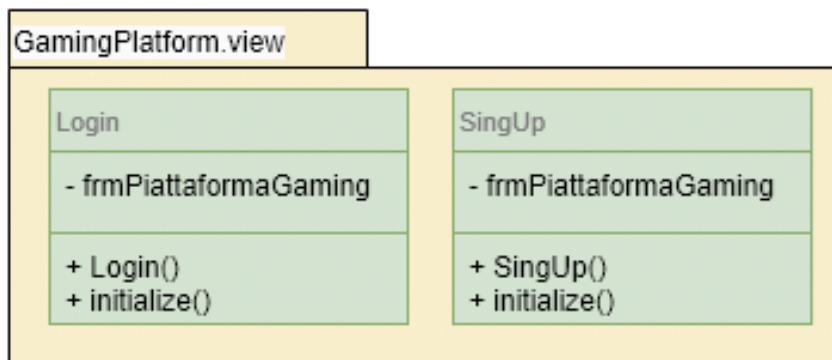
## CONTROLLER'S CLASS DIAGRAM



Il sopra illustrato package implementa il componente *Controller* del pattern MVC; qui viene sviluppata la logica di controllo, dunque la divisione in sottoelementi facilita l'implementazione.

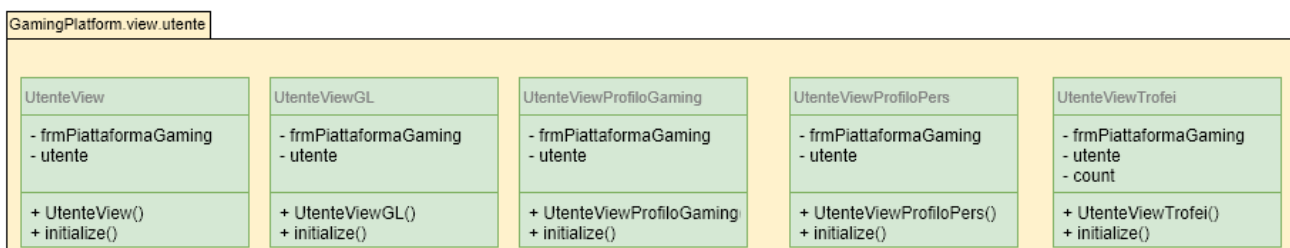
Ogni coppia *<Componente>Controller* incapsula le operazioni correlate a tale componente – i.e. *UtenteController* mette a disposizione i metodi per giocare, votare, recensire un gioco; *GiocoController* riguarda le azioni svolte su un gioco, come listare l'insieme dei giochi, ottenere una media dei voti relativi ad esso o la lista delle recensioni pubblicate -

## VIEW'S CLASS DIAGRAM



Seguendo la gerarchia dei package, abbiamo in un primo livello di view la GUI relativa alla registrazione e al login – in seguito sarà riportato un CD completo del componente *View*.

## UTENTE'S VIEW'S CLASS DIAGRAM - VIEW' SUBPACKAGE –



Questo package rappresenta l'ambito del *View* riguardante gli utenti base; in particolare *UtenteViewGL* implementa la grafica per lista giochi ai quali può partecipare;

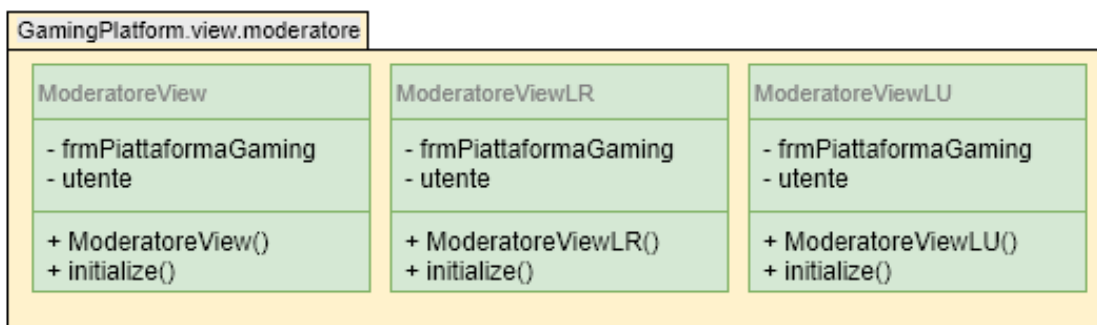
*UtenteViewProfiloGaming* la GUI per il profilo con i progressi svolti nella piattaforma;

*UtenteViewProfiloPers* mostra i dati personali dell'utente, dalla quale possono anche essere modificati – richiamando metodi del controller -;

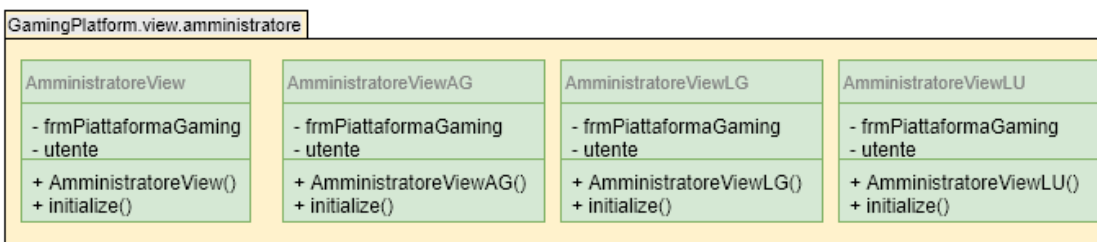
*UtenteViewTrofei* è la bacheca dei premi ottenuti.

## MODERATORE'S VIEW'S CLASS DIAGRAM - VIEW' SUBPACKAGE -

Come per l'utente base, è stata implementata la *View* per i moderatori, in cui *ModeratoreViewLR* mostra la lista delle recensioni da approvare, mentre *ModeratoreViewLU* la lista degli utenti da gestire per la promozione o retrocessione



## AMMINISTRATORE'S VIEW'S CLASS DIAGRAM - VIEW' SUBPACKAGE –



Implementazione della *View* per l'amministratore:

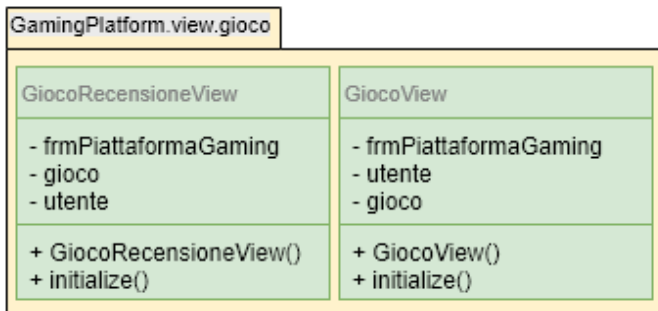
*AmministratoreViewAG* rappresenta la GUI per l'aggiunta di un gioco dalla piattaforma;

*AmministratoreViewLG* mostra la lista dei giochi, dalla quale si può effettuare la rimozione;

*AmministratoreViewLU* è l'interfaccia che permette la visualizzazione degli utenti registrati alla piattaforma.

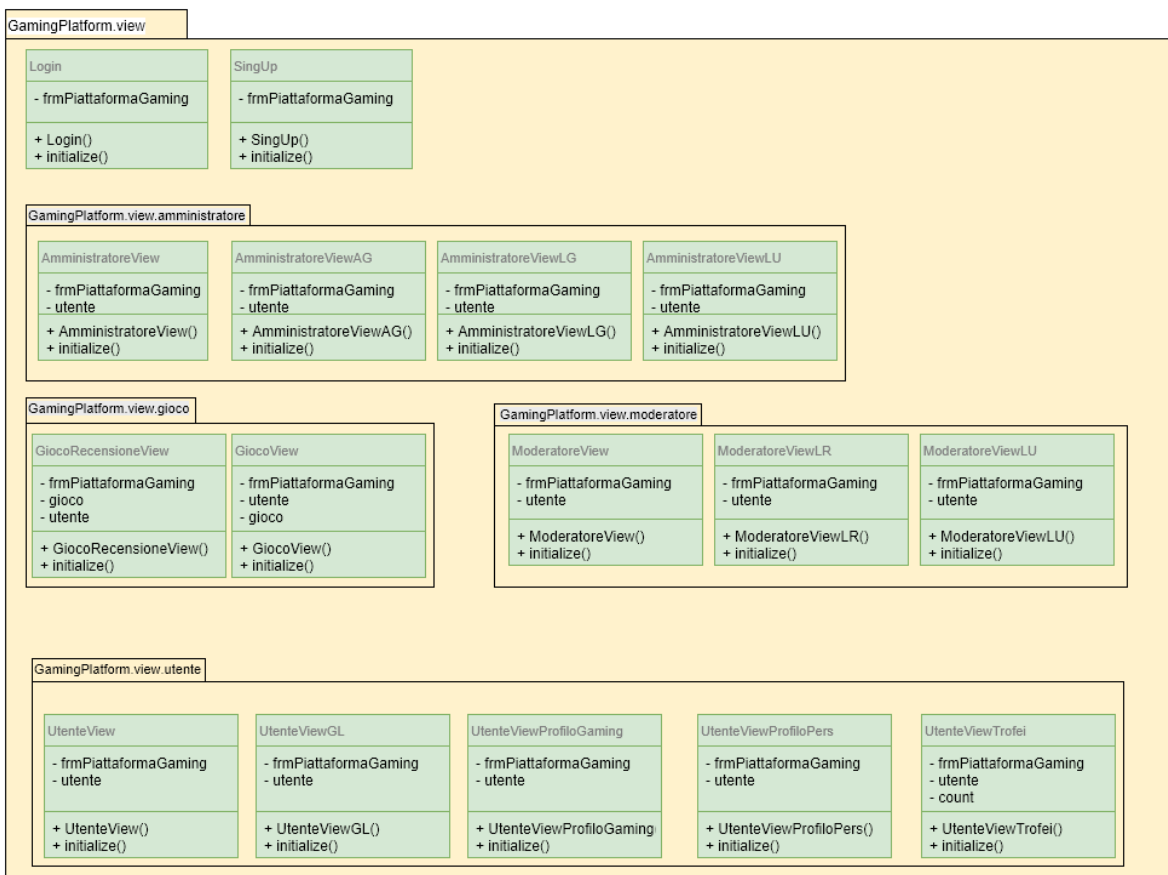


## GIOCO'S VIEW'S CLASS DIAGRAM - VIEW' SUBPACKAGE -



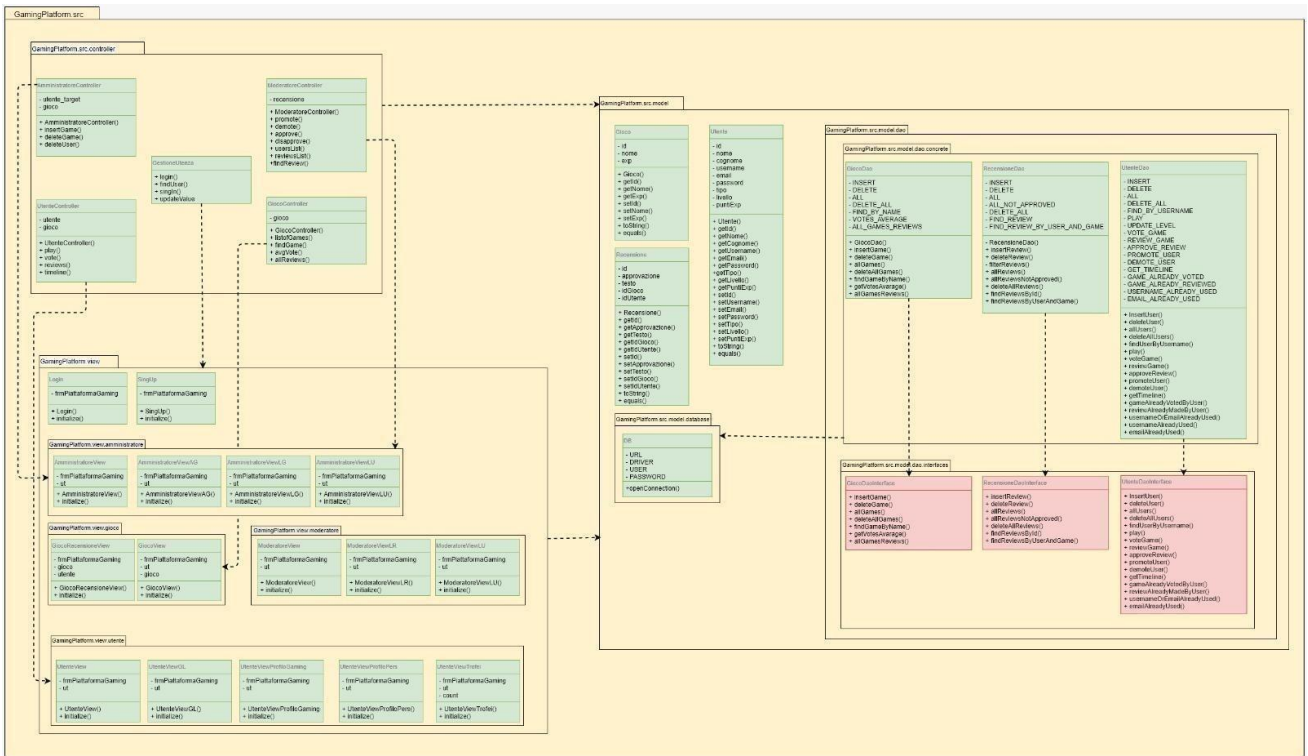
GUI per la visualizzazione della pagina gioco (*GiocoView*) e per la lista delle recensioni pubblicate (*GiocoRecensioniView*)

## FULL VIEW



Come anticipato, il CD intero per il componente *View* del pattern MVC

# SRC CLASS DIAGRAM



Class Diagram dell'intera piattaforma, che evidenzia i links fra i diversi componenti del sistema

# DATABASE

Il Database è fondamentale nella piattaforma, è collegato ad essa tramite un driver JDBC e garantisce persistenza e fruibilità dei dati. È composto da tabelle che modellano i concetti principali, implementato come un database relazionale. La sua composizione è la seguente:

## Tables (4)

- ▷ gioco
- ▷ recensione
- ▷ timeline
- ▷ utente

Nello specifico:

## LA TABELLA UTENTE

	idUtente	nome	cognome	username	email	password	tipo	livello	puntiExp
1	1	Marco	Pistilli	Assacross	Assacross@gmail.com	1234	amministratore	0	0
2	2	Davide	Petrini	King	King@gmail.com	1234	amministratore	0	0
3	13	Amministratore	Amministratore	Amministratore	Amministratore	Amministratore	amministratore	0	0
4	4	Fabio	Di Carlo	Fhumoo	Fabio@gmail.com	1234	moderatore	2	200
5	11	Luca	Ciancaglion	Ciancs	ciancs@gmail.com	12345678	moderatore	0	0
6	12	Piero	Piero	Piero	a@gmail.com	12345678	moderatore	0	0
7	14	Moderatore	Moderatore	Moderatore	Moderatore@moderatore.it	moderatore	moderatore	0	0
8	5	Simone	Marrone	Sbrown	Simone@gmail.com	1234	utente	3	300
9	6	Orazio	De Cesaris	Rowmans	row@gmail.com	1234	utente	4	401
10	7	Diego	Piermarini	Diego	Diego@gmail.com	1234	utente	5	500
11	15	Utente	Utente	Utente	Utente@Utente.it	utente	utente	0	0

## LA TABELLA GIOCO

	<b>idGioco</b>	nome	exp
1	2	Poker	50
2	3	Tris	10
3	4	Scopa	15
4	5	Briscola	15
5	6	Tresette	40
6	7	Scopone	20
7	8	Ruba Mazzo	20
8	9	Piramide	10
9	10	Risiko	50
10	11	Morra	20
11	12	Scacchi	25

## LA TABELLA RECENSIONE

	<b>idRecensione</b>	approvazione	testo	utente	gioco	valutazione
1	27	0		4	1	1
2	28	1	Review of Fhumoo:This ...	4	6	5
3	29	1	Review of Fhumoo: I like it	4	5	4
4	30	1	Review of Moderatore:I ...	14	8	3
5	31	0	Review of Utente:I like t...	15	12	5

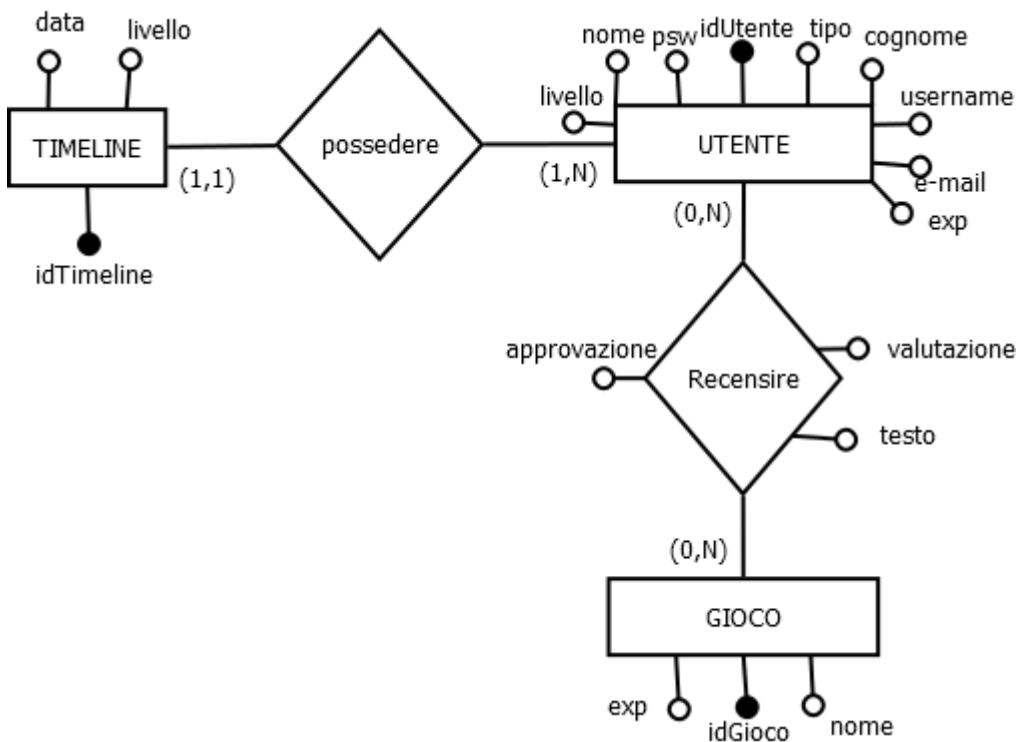
## LA TABELLA TIMELINE

	<b>idTimeline</b>	utente	livello	data
1	6	2	0	2018-04-05
2	11	1	0	2018-05-12
3	27	1	5	2018-05-16
4	29	6	4	2018-05-16
5	30	7	5	2018-05-16
6	31	5	3	2018-05-16
7	32	4	1	2018-05-16
8	33	1	0	2018-05-16
9	34	4	2	2018-05-16
10	35	11	0	2018-05-17
11	36	12	0	2018-05-17
12	37	15	0	2018-05-18
13	38	14	0	2018-05-18
14	39	13	0	2018-05-18

-Il codice sql è disponibile nel file SQL.sql – contiene inoltre i triggers per gli aggiornamenti della timeline per la registrazione, la cancellazione e il raggiungimento di un nuovo livello –

Il DATABASE inoltre è lo strumento col quale l'amministratore ha la completa gestione dei dati della piattaforma: da qui può svolgere altre attività, oltre quelle disponibili nell'applicazione tramite interfaccia grafica, come la query per promuovere un utente base a moderatore e viceversa.

## MODELLO E-R



## MODELLO RELAZIONALE

-Gli attributi sottolineati rappresentano le chiavi primarie dell'entità-

Utente (idUtente, Nome, Cognome, Username, Email, Password, Tipo, Livello, PuntiExp)

Gioco (idGioco, Nome, Exp)

Recensione (ID, Approvazione, Testo, Gioco, Utente, Valutazione)

*Gioco e Utente chiavi esterne delle relative entità*

Timeline (idTimeline, Livello, Utente)

*Livello e Utente chiavi esterne delle relative entità*

# IMMAGINI ESPLICATIVE DELLA PIATTAFORMA GAMING

## LOGIN



The image shows a web browser window titled "Piattaforma Gaming". The page has a light gray background and features the title "Gaming Platform" in a bold, black, sans-serif font. Below the title, there are two input fields: "Username" and "Password", each with a corresponding text label to its left. The "Username" field is a simple white rectangle with a thin gray border. The "Password" field is a white rectangle with a thin gray border. Below these fields, there are two buttons: "Log In" and "Register". Both buttons are white with a thin gray border and a slight shadow. The "Log In" button is positioned to the left of the "Register" button.

Pagina di apertura dalla quale si può eseguire l'autenticazione se si è già registrati, inserendo username e password, altrimenti dal bottone *Register* si può accedere alla pagina di registrazione. I campi sono case-sensitive; in generale, se le stringhe non combaciano ad una coppia username/password presente sul database, verrà mostrato un messaggio di errore nell'autenticazione.

## REGISTRAZIONE



The image shows a registration form window titled "Registrazione". It contains six input fields and two buttons. The fields are labeled "Name", "Surname", "Username", "Password", "Confirm Password", and "E-mail". The buttons are labeled "Cancel" and "Confirm".

Field Label	Input Type
Name	Text
Surname	Text
Username	Text
Password	Text
Confirm Password	Text
E-mail	Text

Buttons: Cancel, Confirm

Da questa schermata è possibile eseguire la Registrazione alla piattaforma gaming; username ed E-Mail dovranno essere univoci, quest'ultima inoltre deve rispettare i caratteri di una mail modello ([stringa@dominio.estensione](#))  
 Ogni campo è soggetto a controllo sulla lunghezza per vincoli sul database; un messaggio di errore verrà visualizzato in caso di errore.



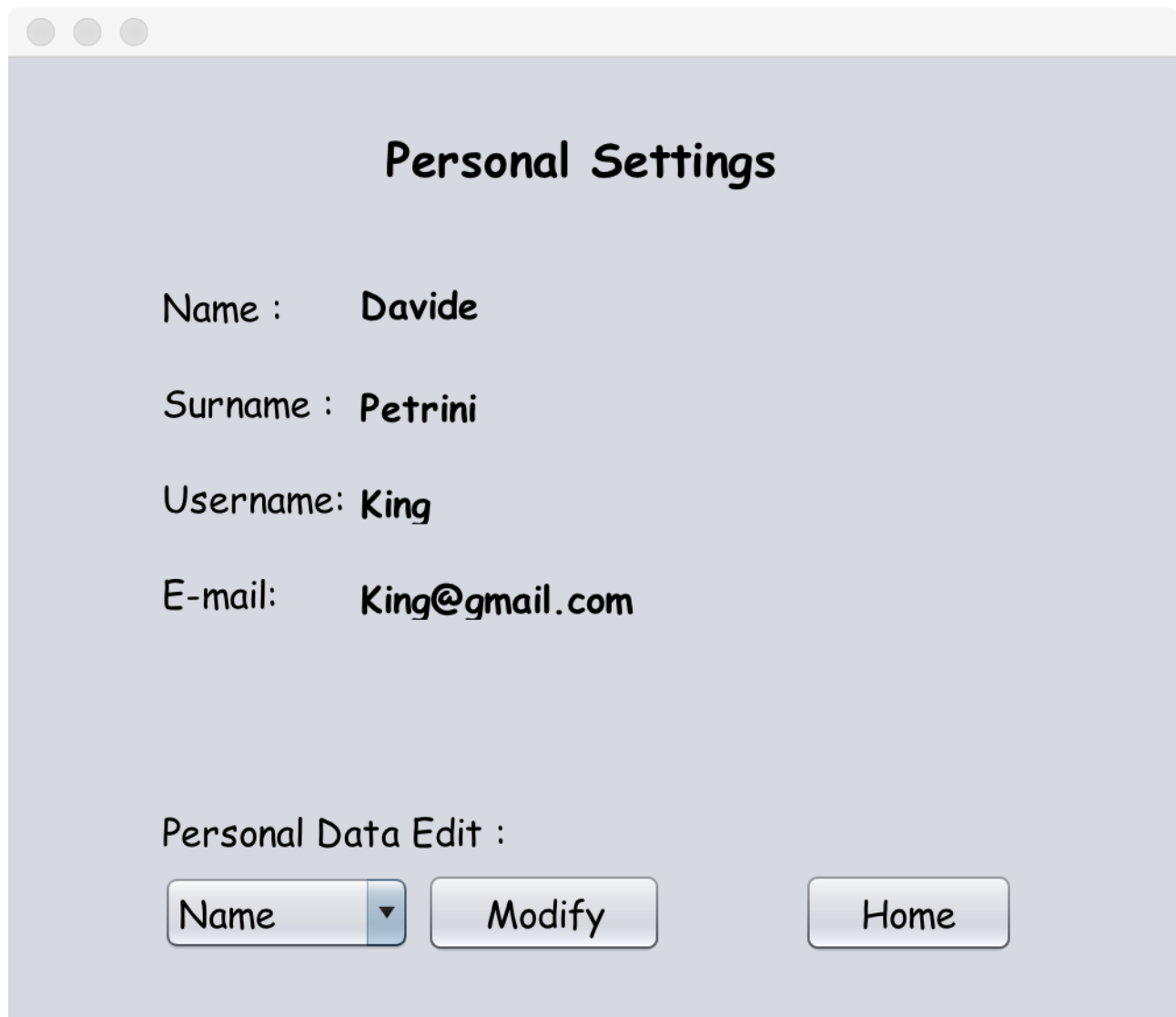
---

## HOME UTENTE BASE



Vista della homepage Utente: come da nome, i bottoni permettono di accedere al Profilo personale, Profilo gaming, scelta del gioco, eseguire il logout.

## PROFILO PERSONALE



**Personal Settings**

Name : **Davide**

Surname : **Petrini**

Username: **King**

E-mail: **King@gmail.com**

Personal Data Edit :

Name ▼ Modify Home

Pagina relativa al profilo personale: è uguale per tutti e tre gli attori. Sono mostrate le informazioni personali inserite al momento della registrazione; eventualmente possono essere modificate dal menù a tendina. Può essere anche cambiata la username, che è mostrata nel profilo gaming.

## PROFILO GAMING UTENTE

### Gaming Profile

Username: Diego

Level: 5

Exp: 500

Trophies: [View Trophies](#)

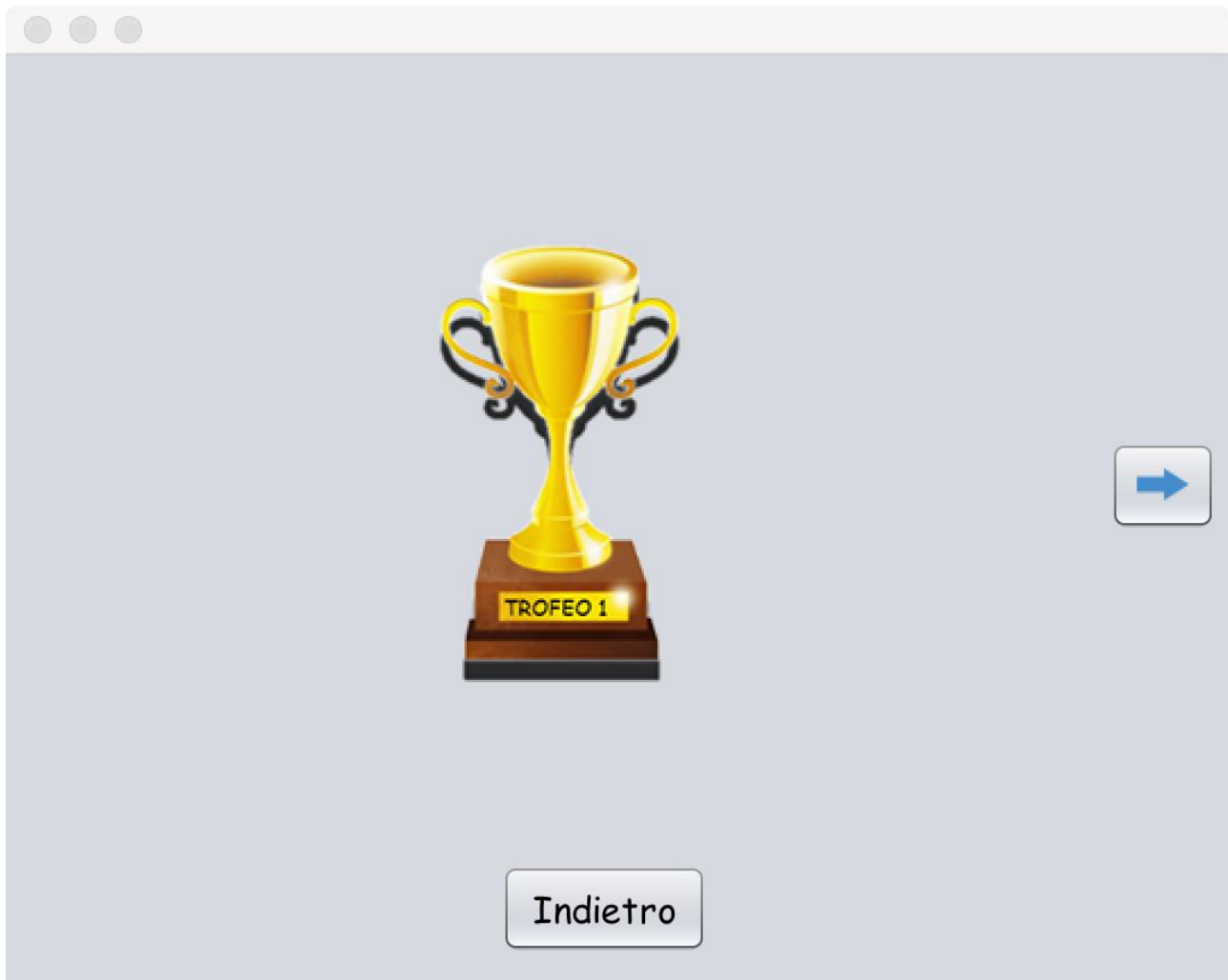
Date	Level
2018-05-16	5

Timeline:

[Home](#)

Profilo che mostra i progressi nel gioco; anch'esso uguale per l'utente base e il moderatore. Inoltre è presente il bottone *View Trophies* dal quale si può accedere alla schermata di visualizzazione trofei.

## VISUALIZZAZIONE TROFEI



Pagina per la visualizzazione dei trofei: se le frecce compaiono, vuol dire che ci sono ancora trofei da scorrere. Se non si è guadagnato nessun trofeo, non si potrà accedere alla pagina.

## LISTA GIOCHI UTENTE



Lista giochi a disposizione dell'utente base e del moderatore – si ricorda che l'amministratore non prende parte ai giochi -; selezionando uno dei giochi e cliccando sul bottone *Go* si accede alla pagina relativa al gioco selezionato.

## PAGINA GIOCO



The screenshot shows a web browser window with a light gray background. At the top, there are three small gray circles representing window controls. The main content area has a light gray background. It displays the game name 'Tresette' and the player's experience '40'. Below this is a large white text input field. Under the input field are two buttons: 'Review' on the left and 'View Review' on the right. Below these buttons is the text 'Average Vote: 5'. Under this text is a small spinner control showing the number '1' and a 'Vote' button. At the bottom of the page are two buttons: 'Play' on the left and 'Back' on the right.

Name: Tresette      Experience: 40

[Review](#)      [View Review](#)

Average Vote: 5

[Vote](#)

[Play](#)      [Back](#)

Ogni gioco ha la sua pagina esplicativa: contiene il nome, l'esperienza fornita ad ogni giocata, la media dei voti assegnati, i bottoni per accedere alla lista delle recensioni approvate.

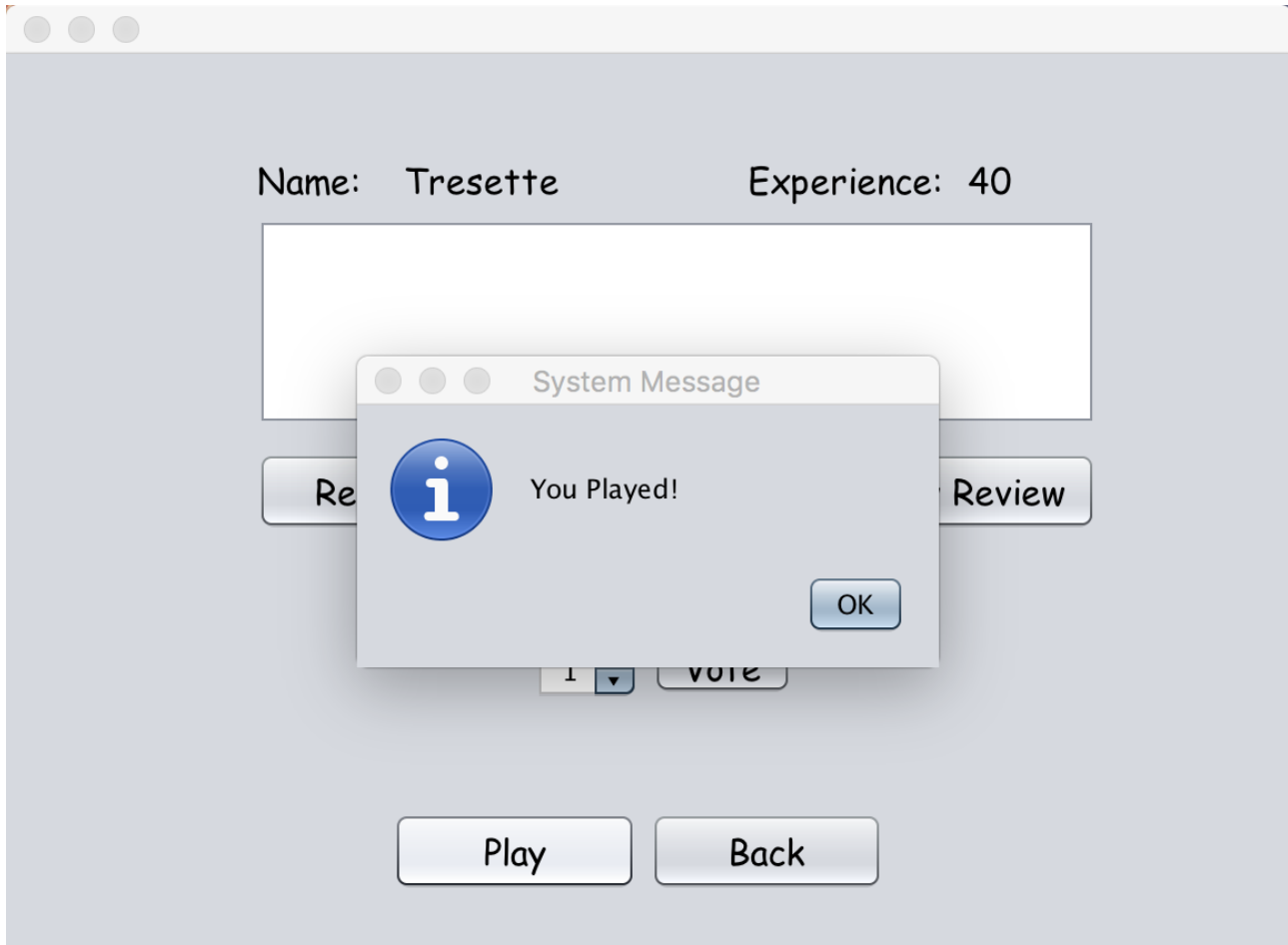
Per assegnare un voto – scelto dall'opzione che può essere incrementata tramite frecce in un range 1-5 e cliccando sul bottone *Vote*, per inviare la recensione composta nel campo testo si deve cliccare sul bottone *Review*, per giocare al gioco prescelto invece su *Play*.

## LISTA RECENSIONI APPROVATE PER UN GIOCO



Il bottone *View Review* della pagina gioco conduce a questa view, che contiene informazioni sullo scrittore ed il testo della recensione.

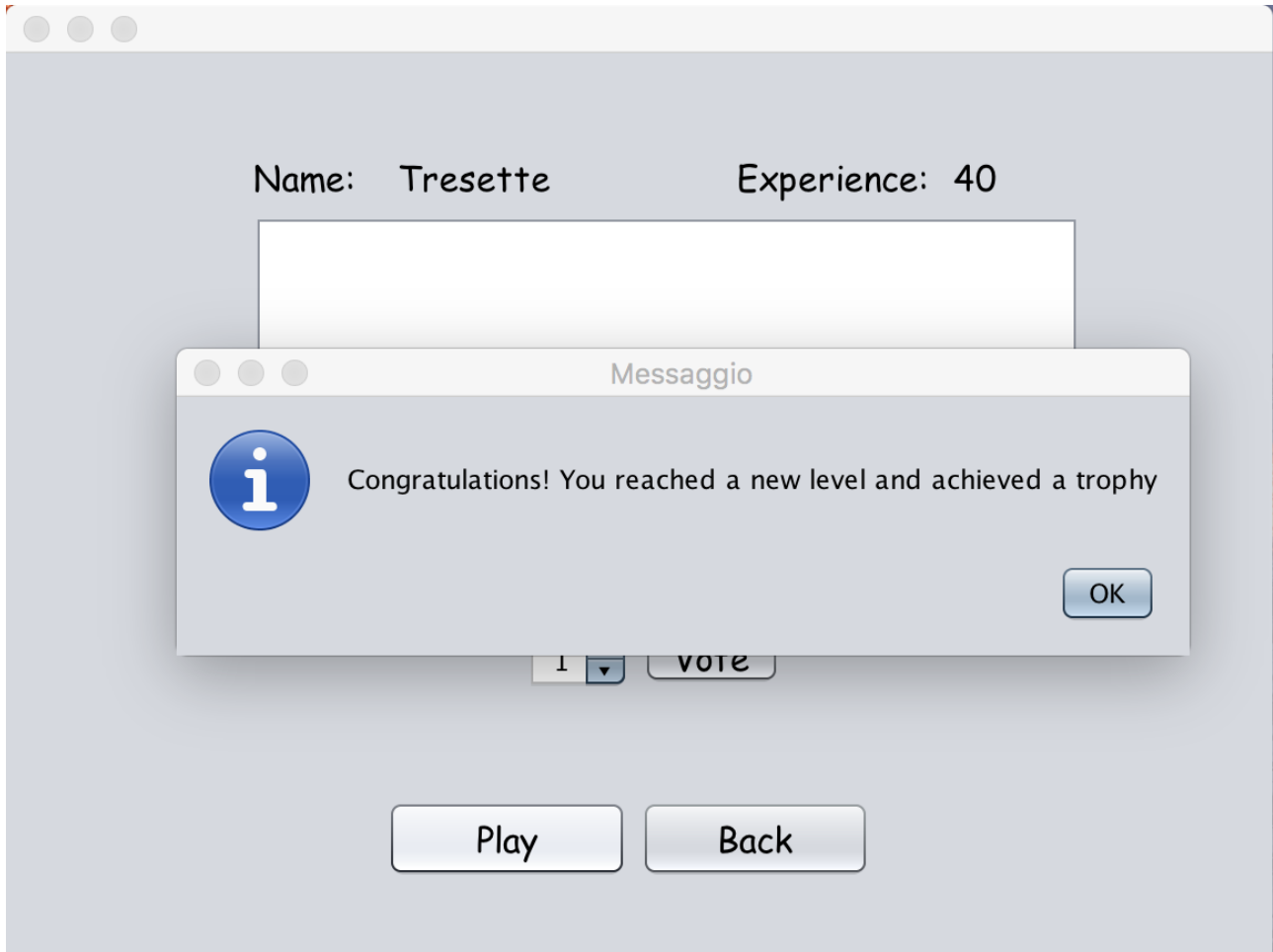
## GIOCARRE AD UN GIOCO



Il click sul bottone *Gioca* produce una bet, e il messaggio di avvenuta giocata avvisa l'utente.



## AVVISO AUMENTO DI LIVELLO



Se la giocata comporta l'aumento di livello, viene notificato tramite un messaggio.

---

## HOME MODERATORE

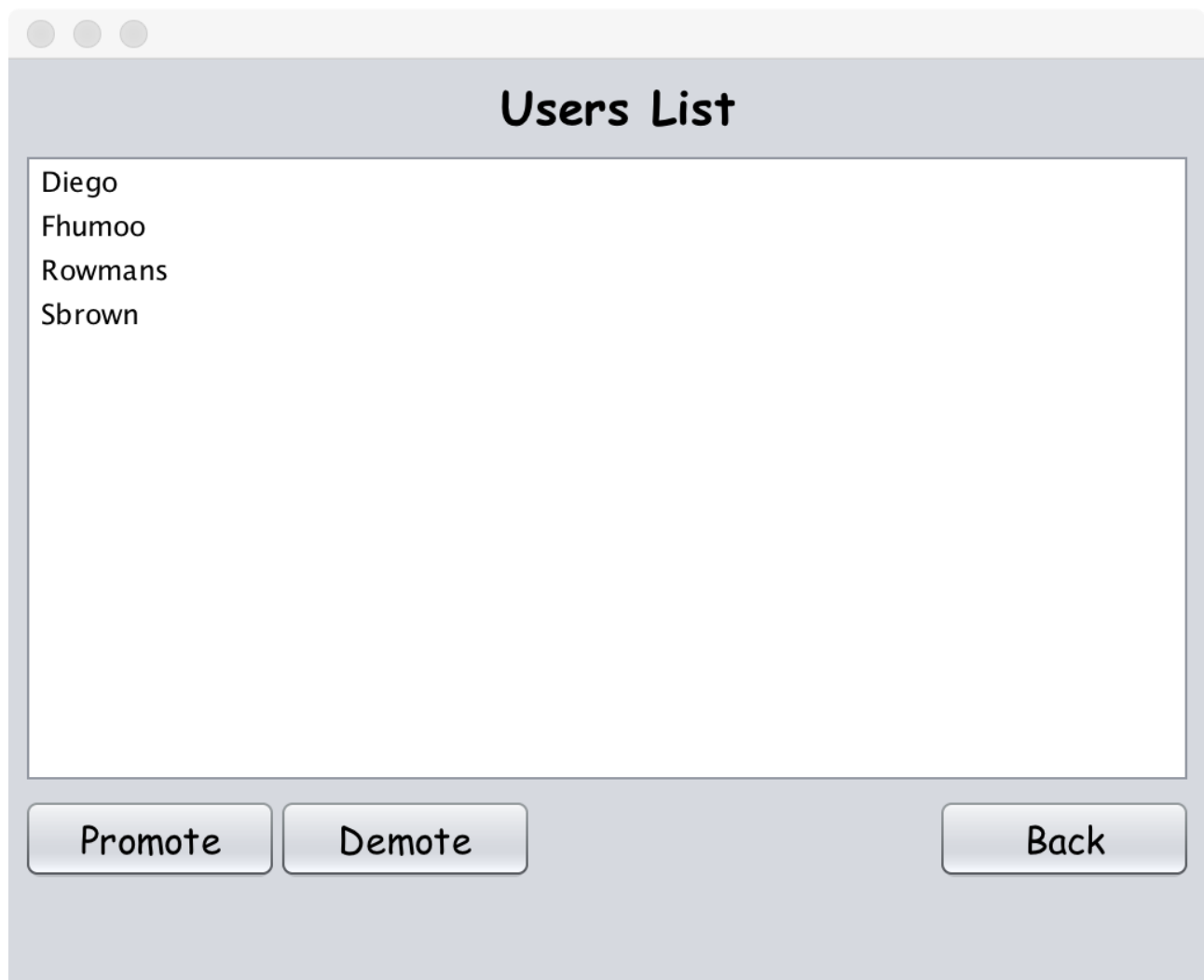


Pagina di benvenuto del moderatore.

Oltre alle features presenti anche nell'Utente base, esso ha a disposizione altri due bottoni: la lista utenti e la lista recensioni da approvare

---

## LISTA UTENTI MODERATORE



La lista utenti visibile dal moderatore permette di promuovere a moderatore un utente base o, viceversa, retrocedere un moderatore ad utente base. Selezionando L'username dell'account che si vuole promuovere/retrocedere e cliccare su uno dei due bottoni *Promote* o *Demote*.

## LISTA RECENSIONI DA APPROVARE



The image shows a window titled "Review List" with a light gray header and a white content area. The content area contains a single review entry: "29 -- Review of Diego: BelloBello". Below the content area, there are three buttons: "Accept", "Decline", and "Back". The "Accept" and "Decline" buttons are positioned side-by-side on the left, and the "Back" button is on the right.

Review List	
29 -- Review of Diego: BelloBello	

Accept Decline Back

Se è presente una recensione da approvare, il tasto *Lista recensioni* presente nella homepage del moderatore potrà essere cliccabile e porterà alla pagina con la lista delle recensioni. In quest'ultima, selezionando una delle recensioni da approvare, si potrà scegliere se approvarla o negarla, come da immagine.

## HOME AMMINISTRATORE



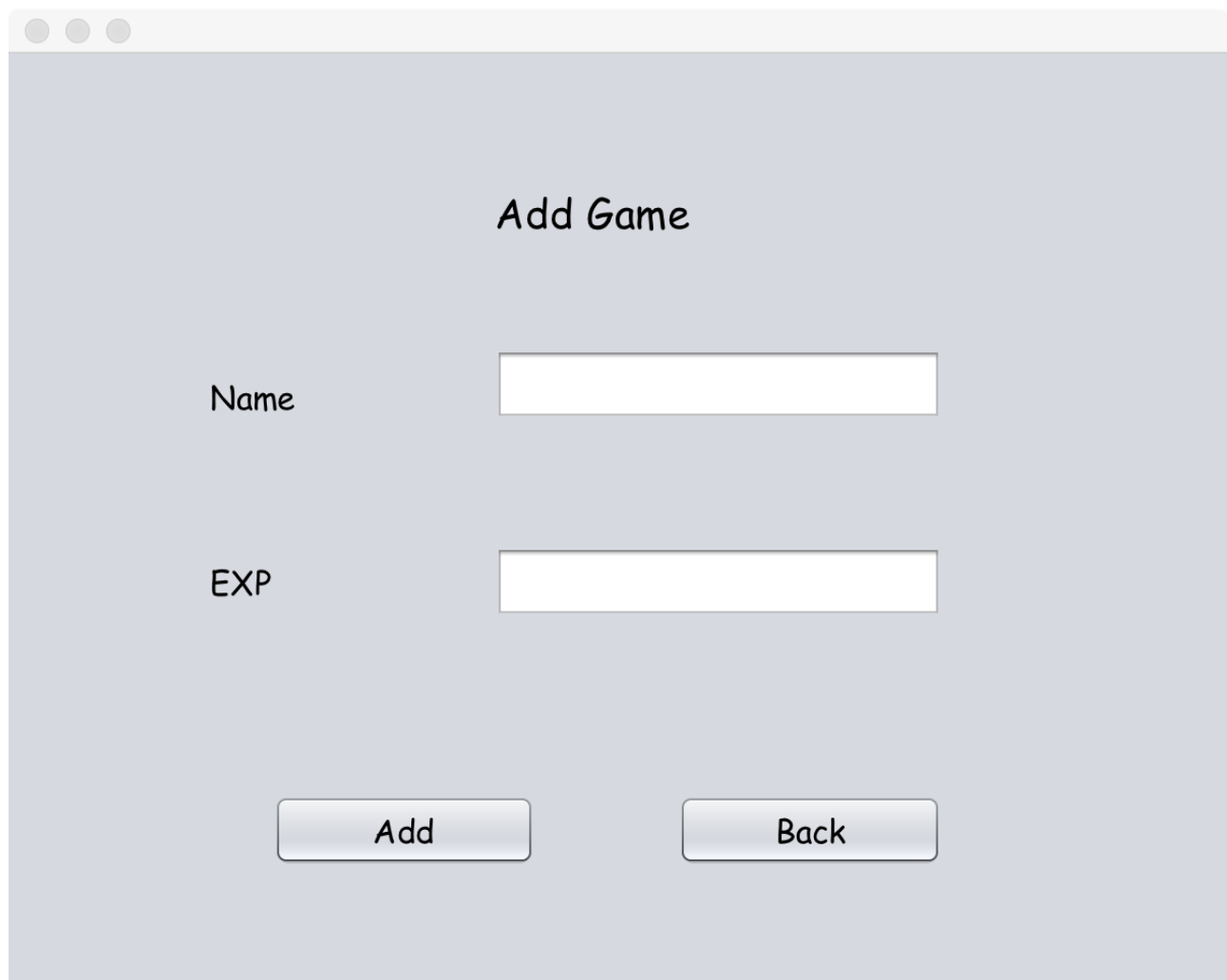
Homepage dell'amministratore

## LISTA GIOCHI AMMINISTRATORE



Il click sul bottone *Edit Games* nella homeview dell'amministratore produrrà la seguente schermata, composta dalla lista dei giochi presenti nella piattaforma, seguiti dal bottone *Delete* per un'eventuale rimozione del gioco. Inoltre è presente il tasto *Add Game* per poter effettuare l'inserimento di un nuovo gioco

## INSERIMENTO DI UN GIOCO



A screenshot of a web application window titled "Add Game". The window has a light gray background and a title bar with three small circles. The form contains two input fields: "Name" and "EXP", each with a corresponding text label to its left. Below the input fields are two buttons: "Add" and "Back".

*Add Game*

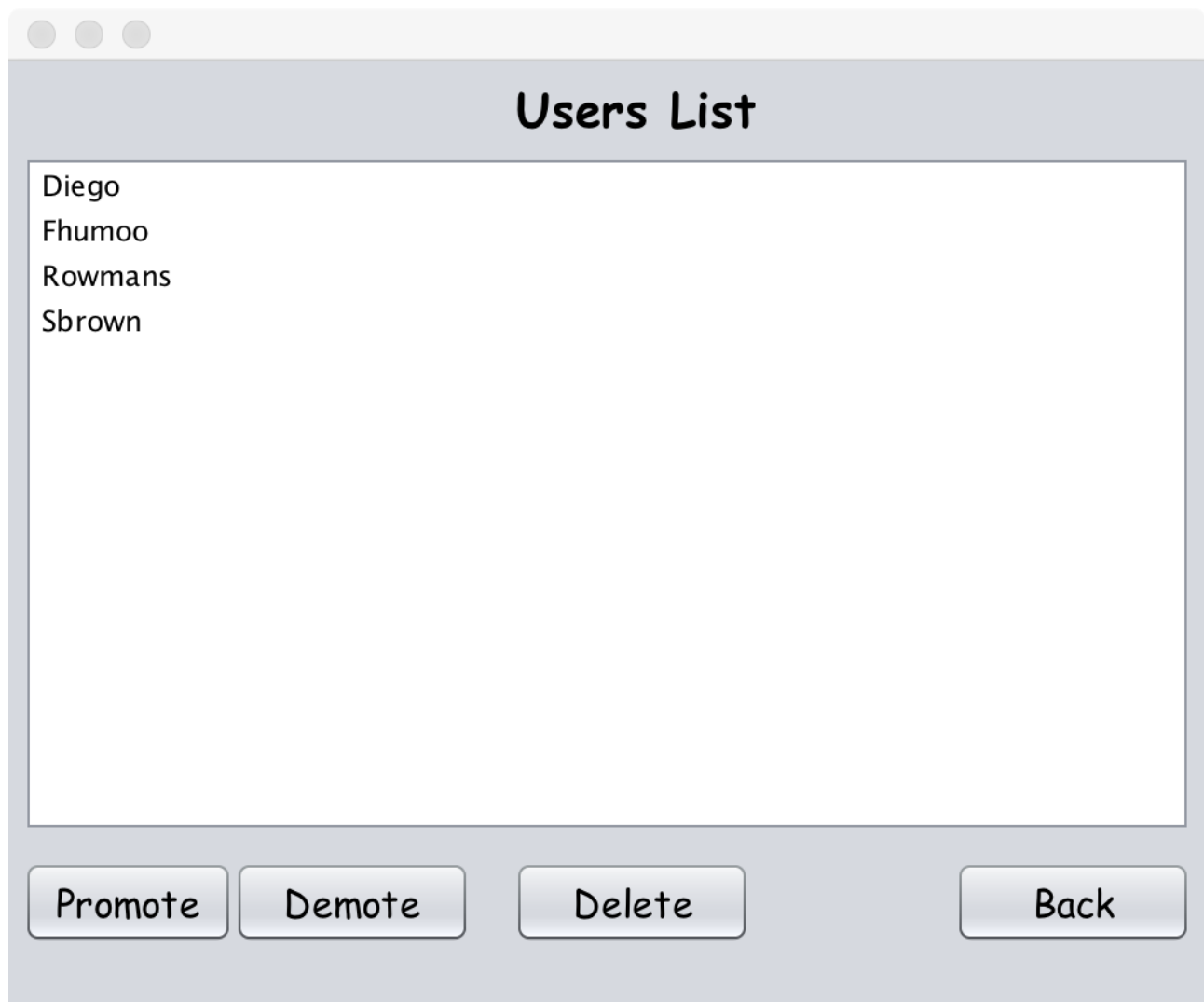
Name

EXP

Il click infatti riporterà a questa schermata, nella quale si inseriranno il nome ed i punti EXP del nuovo gioco; se il gioco è già presente, verrà segnalato un messaggio di errore.

---

## LISTA UTENTI AMMINISTRATORE



Da questa lista l'amministratore potrà eliminare un utente dalla piattaforma oltre alle funzionalità che ha il moderatore.