

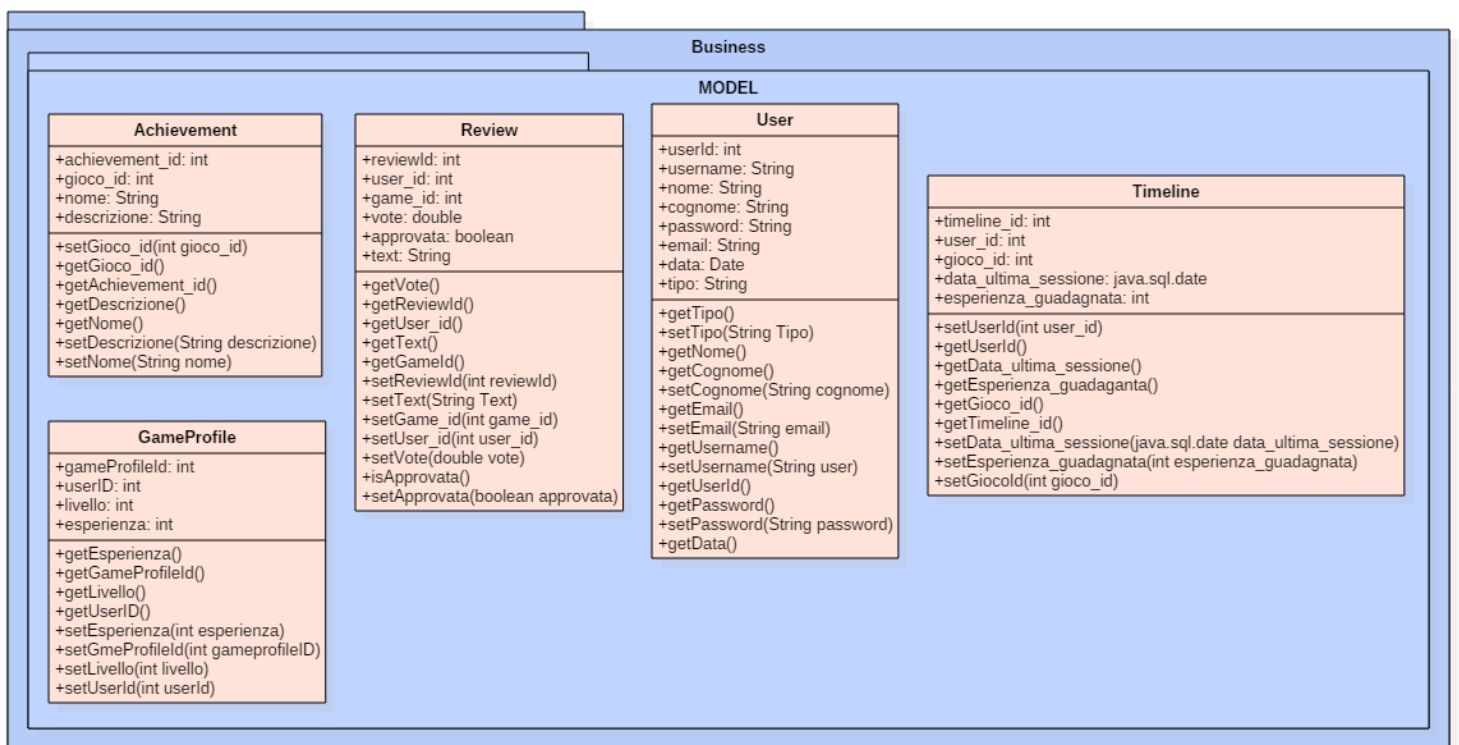
CLASS DIAGRAM

L'elemento di modello principale del class diagram è la classe.

Una classe rappresenta una categoria di entità (istanze), nel caso particolare dette oggetti; il nome della classe indica la categoria di entità descritta in essa. Ogni classe è corredata da un insieme di attributi (che descrivono le caratteristiche o lo stato degli oggetti) e operazioni (che ne descrivono il comportamento).

Abbiamo deciso di strutturare il nostro sistema in tre package: Business, Listener e Presentation. Business a sua volta è diviso in Model e Implementation. In Model troviamo le classi principali, le quali rappresentano i protagonisti del sistema (Gli Oggetti) ovvero:

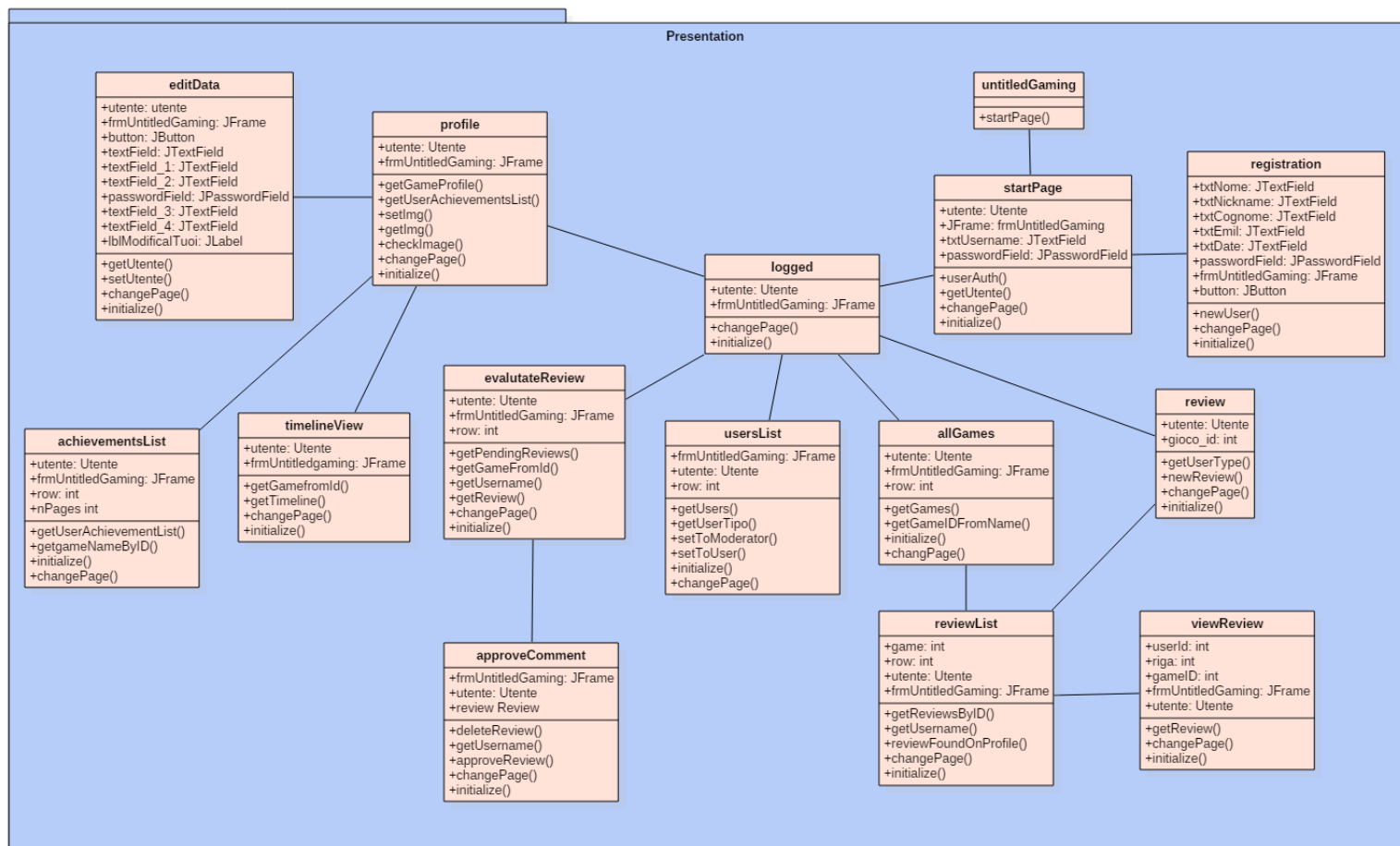
- Achievement;
- GameProfile;
- Review;
- User;
- Timeline;



Per interfacciarsi al sistema ogni utente ha a disposizione una propria interfaccia grafica (GUI), la quale a sua volta è rappresentata da una classe.

Le classi Swing sono definite nel pacchetto javax.swing, il cui nome javax indica estensione standard a Java, ogni interfaccia è stata progettata per risultare il più possibile user-friendly.

Queste classi sono contenute nel package “Presentation”. Ogni oggetto grafico è predisposto ad essere sollecitato in qualche modo dall’utente e ad ogni sollecitazione genera eventi che vengono gestite dalle apposite interfacce (Interfaces).



Nel package “Implementation” troviamo le classi che servono a interagire con il database:

- **“ReviewManagement”**: adibita a gestire i metodi riguardanti le recensioni. Tra le altre funzioni è possibile aggiungere o eliminare un oggetto di tipo “recensione” e gestirne i vari aspetti.
- **“UserManagement”**: essa gestisce i metodi riguardanti l’utenza e quindi l’aggiunta di un nuovo utente, la promozione o retrocessione di esso ecc.

- **“TimelineManagement”**: essa gestisce la cronologia di gioco dell’utente andando a fare un update nel database ogni volta che l’utente terminerà una sessione di gioco e nel momento in cui l’utente lo richiede, mostra i dati relativi all’ultimo update (quindi ultima sessione di gioco) dello stesso utente.
- **“AchievementsManager”**: essa gestisce i metodi relativi all’attribuzione degli achievement gestendo la lista presente nel database e sbloccandoli all’utente che ha soddisfatto un determinato controllo (in questo caso per meriti dovuti ai vari giochi), inoltre è atto anche a visualizzare la descrizione di ogni singolo achievement o di una lista nel momento in cui l’utente lo richiede, ecc
- **“DBManager”**: è una classe che serve a instaurare una connessione tra il database e il sistema. La connessione è effettuata tramite un oggetto Connection dichiarato *static*, in modo da avere un’unica connessione al database per tutto il sistema .All’interno della classe sono presenti altre funzioni relative al check di password (tramite verifica degli hash), username (per evitare che vengano registrati 2 utenti con lo stesso username) ed email (per far si che venga inserito un indirizzo email valido).

In ogni chiamata viene gestita un’eccezione, tramite la creazione di un nuovo oggetto della classe **“BusinessException”**, che si trova nel package Business. Questa classe si occupa di avvertire, tramite un messaggio di errore, l’utente nel caso si verifichi un qualsiasi problema nell’applicazione.

