



UNIVERSITÀ DEGLI STUDI DI PALERMO  
DIPARTIMENTO DI INGEGNERIA  
CORSO DI LAUREA TRIENNALE IN  
INGEGNERIA INFORMATICA

## Object Design Document

Sportify™

INGEGNERIA DEL SOFTWARE A.A. 2023/2024



A cura di:

Francesco Fausto Franchina

Francesco Fausto Franchina

Marco Marino

Marco Marino

Giovanni Dolcemascolo

Giovanni Dolcemascolo

Diego Lo Voi

Diego Lo Voi

# SOMMARIO

---

<b>1. Introduzione</b>	3
1.1. Prestazioni e costi	3
1.2. Interfacce	3
1.3. Tempi di risposta	3
1.4. Panoramica DBMS	3
<b>2. Packages</b>	4
2.1. com.sportify	4
2.1.1. API	4
2.1.2. Entity	4
2.1.3. Commons	5
2.1.4. Client	5
2.1.5. Server	6
2.2. Java	7
2.2.1. util.*	7
2.2.2. time.*	7
2.2.3. sql.*	7
2.2.4. javax.mail.*	7
2.3. MySQL – Connector – Java (JDBC)	7
2.4. JavaFX	7
2.4.1. base.*	7
2.4.2. controls.*	7
2.4.3. fxml.*	7
2.4.4. graphics.*	7
2.4.5. swing.*	8
<b>3. Object design UML</b>	8
3.1. Commons	8

3.2. Interfacce	9
3.2.1. Autenticazione	9
3.2.2. Gestione account	9
3.2.3. Gestione prenotazioni	10
3.2.4. Gestione tornei	10
3.2.5. Ricerca utenti	11
3.2.6. Gestione livelli	11
3.2.7. Gestione promozioni	12
3.2.8. Gestione messaggi	13
3.2.9. Gestione notifiche	14
3.3. Controls	15
3.3.1. Autenticazione	15
3.3.2. Gestione account	16
3.3.3. Gestione prenotazioni	16
3.3.4. Gestione tornei	17
3.3.5. Ricerca utenti	18
3.3.6. Gestione livelli	18
3.3.7. Gestione promozioni	19
3.3.8. Gestione messaggi	19
3.3.9. Gestione notifiche	20
3.4. Interfaccia DBMS	21

# 1. Introduzione

---

## 1.1. Prestazioni e costi

Per lo sviluppo del software è stata adottata una strategia volta a massimizzare l'economicità attraverso l'uso di librerie open source, dal momento che, essendo gratuite, consentono di evitare investimenti in licenze software costose. L'utilizzo di tali librerie, inoltre, permette di accelerare i tempi di sviluppo in quanto la maggior parte di queste sono già mature e ampiamente utilizzate, quindi anche ben documentate e supportate da una vasta comunità di sviluppatori.

## 1.2. Interfacce

Le interfacce del software sono state progettate con particolare attenzione all'usabilità e all'estetica, utilizzando JavaFX come framework principale. JavaFX, infatti, offre una vasta gamma di strumenti e componenti per la realizzazione di interfacce grafiche sofisticate, consentendo quindi di garantire una user experience che sia non solo accattivante sul fronte estetico, ma anche intuitiva e accessibile per ogni utente, indipendentemente dal suo livello di competenza. A quanto detto si aggiunge che le interfacce sviluppate con JavaFX hanno il pregio di non gravare pesantemente sulle prestazioni del software.

## 1.3. Tempi di risposta

I tempi di risposta del software sono stati ottimizzati grazie al lavoro svolto sulle interrogazioni SQL. Infatti, per la realizzazione del DB si è tenuto conto delle operazioni da eseguire con maggiore frequenza, in modo da ridurre il carico che ogni interrogazione genera sul DBMS. Tra le ottimizzazioni effettuate figurano l'indicizzazione delle occorrenze di alcune tabella e una strutturazione del modello relazionale che consente di effettuare operazioni di join efficienti. Per garantire la scalabilità del sistema è stata adottata un'architettura di tipo client-server, che ha consentito di applicare una separazione netta delle operazioni da effettuare tra le varie componenti del sistema, il che ha effetti positivi anche sulle operazioni di manutenzione e gestione del software. Entrando nel dettaglio, il server centralizzato gestisce la logica del software e l'accesso controllato ai dati, mentre i client si occupano dell'interazione con l'utente.

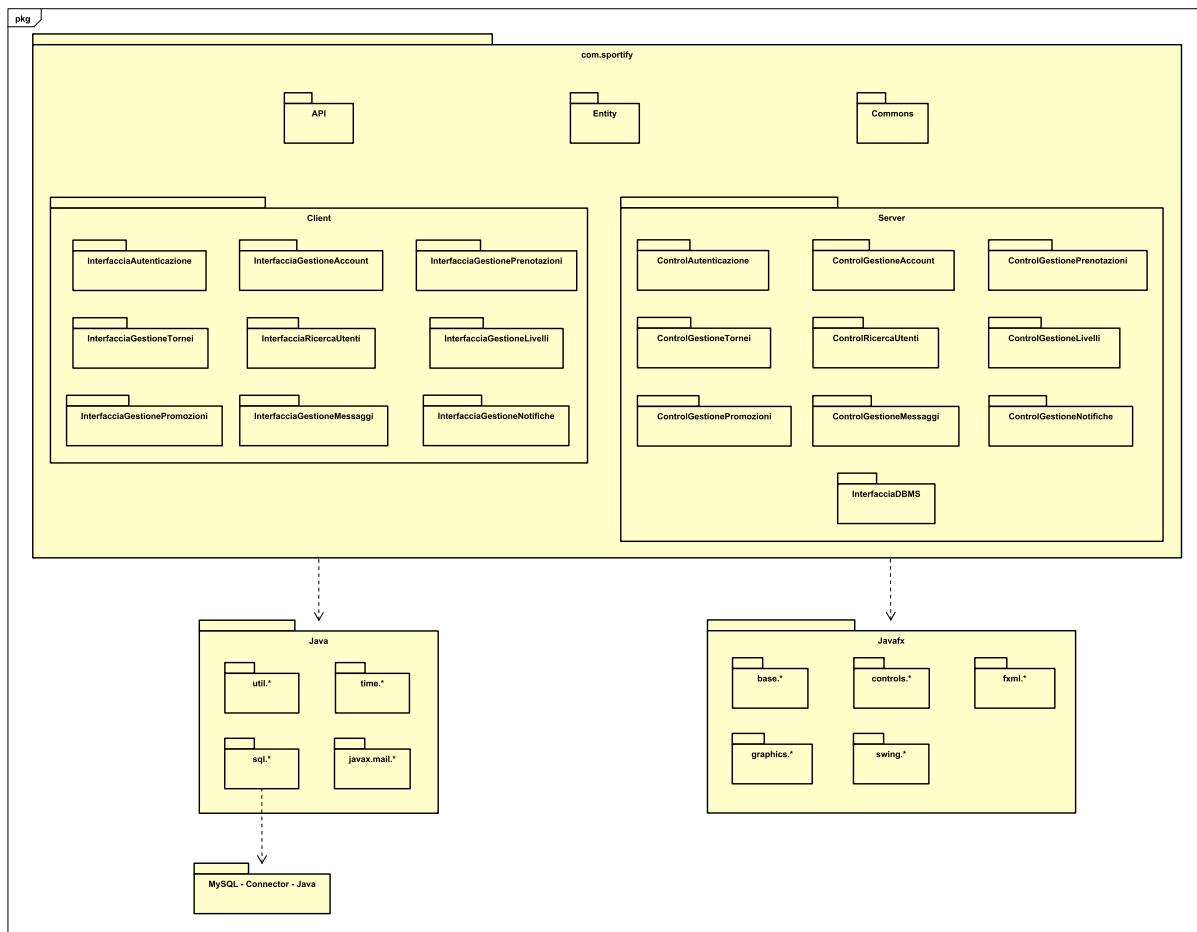
## 1.4. Panoramica DBMS

La scelta del DBMS è ricaduta su MySQL, che gode di elevata robustezza e garantisce una buona affidabilità, oltre ad essere capace di gestire grandi volumi di dati offrendo prestazioni elevate. L'interazione tra il sistema e il DBMS è stata implementata utilizzando MySQL-Connector-Java, una libreria che consente la connessione tra applicazioni Java e MySQL.

attraverso l'uso del protocollo JDBC (Java DataBase Connectivity). JDBC è uno standard Java per la connessione a database relazionali, che fornisce un'API unificata per l'esecuzione di query SQL, l'aggiornamento dei dati e la gestione delle transazioni.

## 2. Packages

---



### 2.1. com.spotify

Il package principale del software.

#### 2.1.1. API

Include classi wrapper per API, progettate ad hoc per questo sistema, ragion per cui risultano meno adatte ad essere riutilizzate in altri contesti.

#### 2.1.2. Entity

Include le classi che modellano le entità facenti parte del sistema.

### 2.1.3. Commons

Contiene le interfacce condivise utilizzate in tutto il software.

### 2.1.4. Client

Contiene le componenti che devono essere installate in ciascun nodo client, impiegate per la gestione delle interfacce grafiche. Tali componenti sono:

**Interfaccia Autenticazione:** gestisce le interfacce grafiche necessarie per l'accesso degli utenti al sistema. Fornisce dunque le schermate necessarie per effettuare il login, la registrazione e il recupero dell'account.

**Interfaccia Gestione account:** gestisce le interfacce grafiche necessarie per consentire agli utenti di gestire i propri dati personali. Fornisce dunque le schermate per la visualizzazione dei dati personali, la modifica di e-mail e password, e le schermate per la disconnessione dell'account e la cancellazione dell'account.

**Interfaccia Gestione prenotazioni:** gestisce le interfacce grafiche per consentire agli utenti di prenotare campi da gioco ed effettuare operazioni sulle prenotazioni già esistenti. Fornisce dunque le schermate per visualizzare prenotazioni aperte o chiuse, per la gestione delle prenotazioni aperte dell'utente e per la gestione degli inviti ricevuti dai giocatori.

**Interfaccia Gestione tornei:** gestisce le interfacce grafiche per consentire agli utenti di creare tornei, iscriversi ai tornei aperti, ed effettuare operazioni sui tornei esistenti. Fornisce dunque le schermate per visualizzare i tornei attivi e passati e per gestire i tornei attivi.

**Interfaccia Gestione promozioni:** gestisce le interfacce grafiche per consentire la gestione delle promozioni. Fornisce dunque le schermate che permettono ai gestori di creare, visualizzare e applicare promozioni, e che permettono ai giocatori di visualizzare le promozioni valide.

**Interfaccia Gestione notifiche:** gestisce le interfacce grafiche per consentire la gestione delle notifiche. Fornisce dunque una schermata per la visualizzazione delle notifiche ricevute, nonché dei pop-up di supporto all'interfaccia.

**Interfaccia Ricerca utenti:** gestisce le interfacce grafiche per la ricerca degli utenti. Fornisce le schermate per la ricerca degli utenti, nonché dei pop-up di supporto all'interfaccia.

**Interfaccia Gestione messaggi:** gestisce le interfacce grafiche per la gestione dei messaggi. Fornisce dunque le schermate per la ricerca e la visualizzazione delle chat con gli altri utenti.

**Interfaccia Gestione livelli:** gestisce le interfacce grafiche per la gestione dei livelli dei giocatori. Fornisce dunque le schermate per l’assegnazione dei livelli ai giocatori per gli sport consentiti.

### 2.1.5. Server

Contiene le componenti che devono essere installate nel nodo server, impiegate per la gestione delle control responsabili della logica di business del sistema e l’interfaccia al DBMS. Tali componenti sono:

**Control Autenticazione:** gestisce la logica applicativa relativa all’accesso degli utenti. Implementa le funzionalità per la registrazione di un nuovo utente, per il login e per il recupero dell’account.

**Control Gestione account:** gestisce la logica applicativa relativa alla gestione dei dati personali degli utenti. Implementa quindi le funzionalità necessarie per la visualizzazione dei propri dati, per la modifica di e-mail e password, per la disconnessione dell’account e per la cancellazione dell’account.

**Control Gestione prenotazioni:** gestisce la logica applicativa per l’amministrazione delle prenotazioni degli utenti. Implementa quindi le funzionalità per la creazione e la gestione delle prenotazioni, per la ricerca e la visualizzazione di prenotazioni aperte o chiuse e per la gestione degli inviti ricevuti.

**Control Gestione tornei:** gestisce la logica applicativa per l’amministrazione dei tornei. Implementa quindi le funzionalità per la creazione e la gestione dei tornei, per la visualizzazione dei tornei attivi e passati, per iscriversi ai tornei attivi.

**Control Gestione promozioni:** gestisce la logica applicativa necessaria per la gestione delle promozioni. Implementa le funzionalità per permettere agli utenti di creare, visualizzare e applicare promozioni.

**Control Gestione notifiche:** gestisce la logica applicativa necessaria per la gestione delle notifiche. Implementa quindi le funzionalità per la visualizzazione e la gestione delle notifiche ricevute.

**Control Ricerca utenti:** gestisce la logica applicativa necessaria per la ricerca degli utenti. Implementa quindi le funzionalità per la ricerca degli utenti e la cancellazione degli account dei giocatori.

**Control Gestione messaggi:** gestisce la logica applicativa necessaria per la gestione dei messaggi. Implementa quindi le funzionalità per la ricerca e la visualizzazione delle chat con gli altri utenti.

**Control Gestione livelli:** gestisce la logica applicativa necessaria per la gestione dei livelli dei giocatori. Implementa quindi le funzionalità per l’assegnazione dei livelli ai giocatori per gli sport consentiti.

**Interfaccia DBMS:** gestisce la comunicazione tra il sottosistema server e il DBMS. Fornisce tutte le funzionalità necessarie per l'interazione con il DBMS mediante operazioni di inserimento, aggiornamento e cancellazione di dati memorizzati.

## 2.2. Java

Questo package contiene tutte le librerie di Java impiegate nella realizzazione del software.

### 2.2.1. util.\*

Contiene tutte le strutture dati utilizzate.

### 2.2.2. time.\*

Contiene le classi utilizzate per gestire le date e gli orari.

### 2.2.3. sql.\*

Contiene le classi utilizzate per gestire le comunicazioni con il DBMS.

### 2.2.4. javax.mail.\*

Contiene le classi utilizzate per l'invio di e-mail tramite protocollo SMTP.

## 2.3. MySQL – Connector – Java (JDBC)

Questo package fornisce tutti i driver e le classi necessarie per consentire alle applicazioni Java di interagire con il database MySQL in modo efficiente e sicuro.

## 2.4. JavaFX

### 2.4.1. base.\*

Contiene le classi di base per realizzare interfacce grafiche con JavaFX.

### 2.4.2. controls.\*

Contiene le componenti di controllo dell'interfaccia utente quali, ad esempio, tasti e campi di testo.

### 2.4.3. fxml.\*

Include il supporto ai file scritti in linguaggio FXML, che offrono una modalità alternativa di definire i layout delle schermate.

### 2.4.4. graphics.\*

Contiene le classi per il rendering delle componenti grafiche delle schermate (finestre, forme geometriche, ...) e di effettistica e animazioni ad esse correlate.

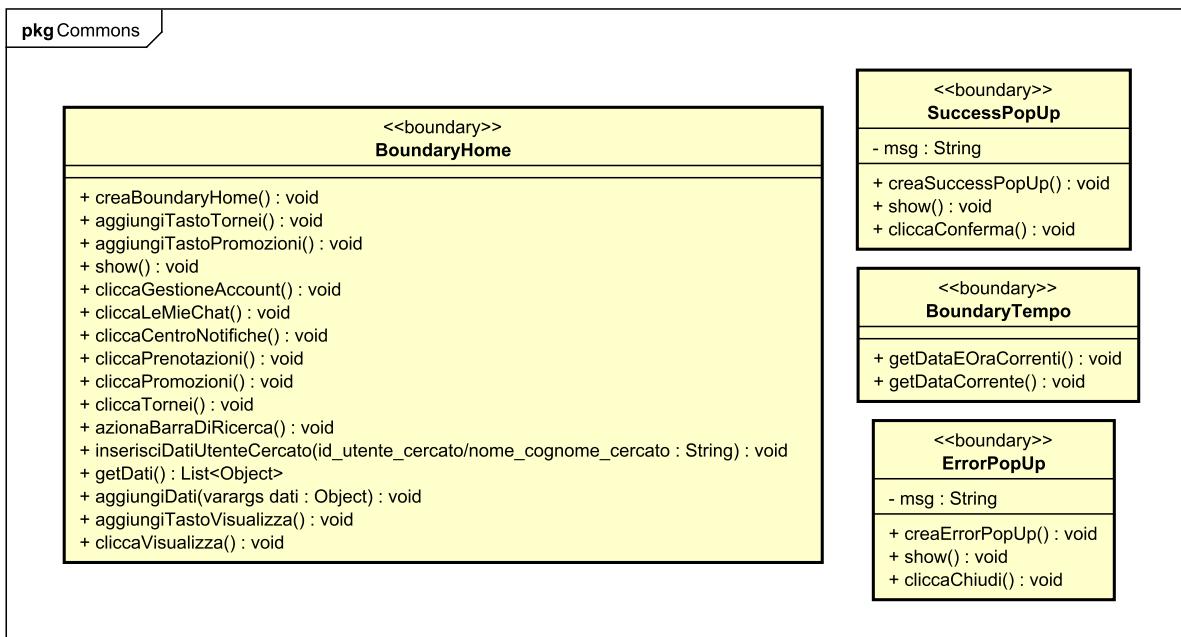
### 2.4.5. swing.\*

Include il supporto alle librerie del framework swing in JavaFX.

## 3. Object design UML

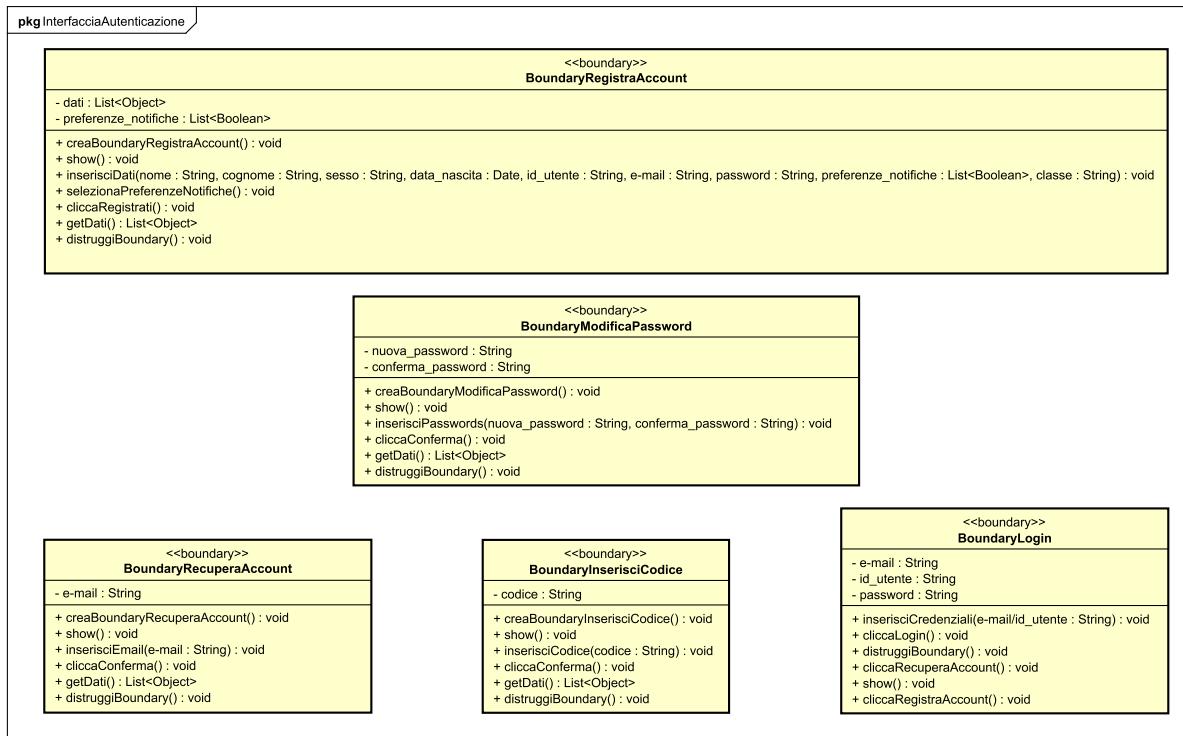
---

### 3.1. Commons

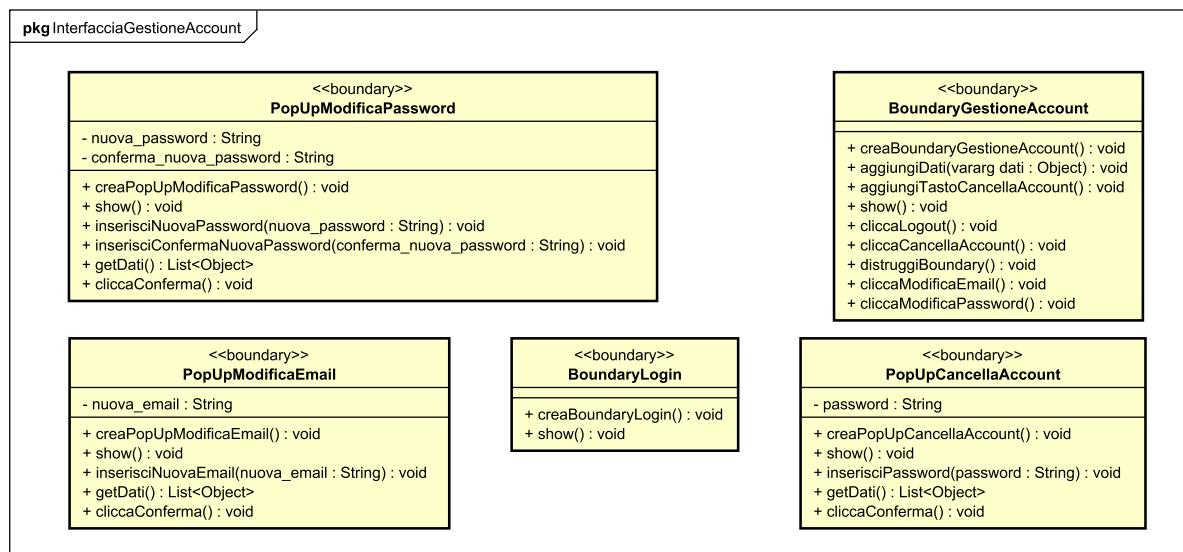


## 3.2. Interfacce

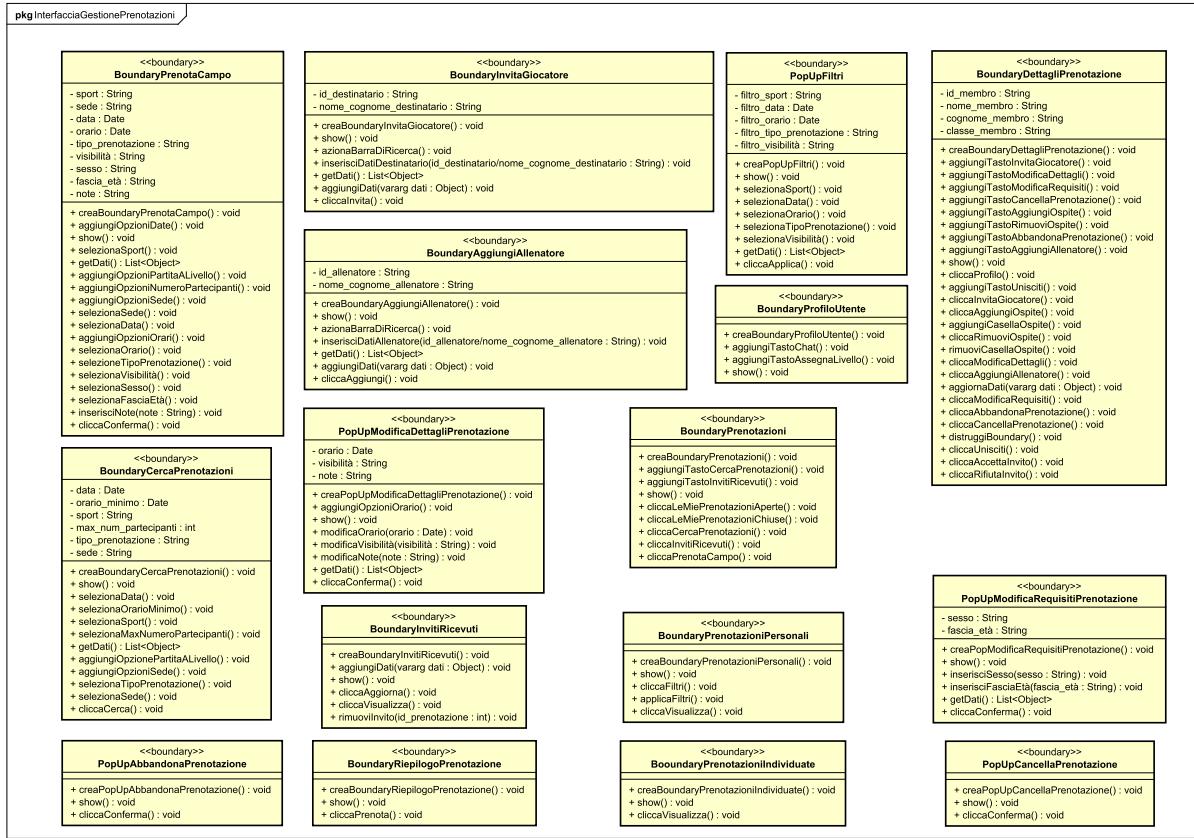
### 3.2.1. Autenticazione



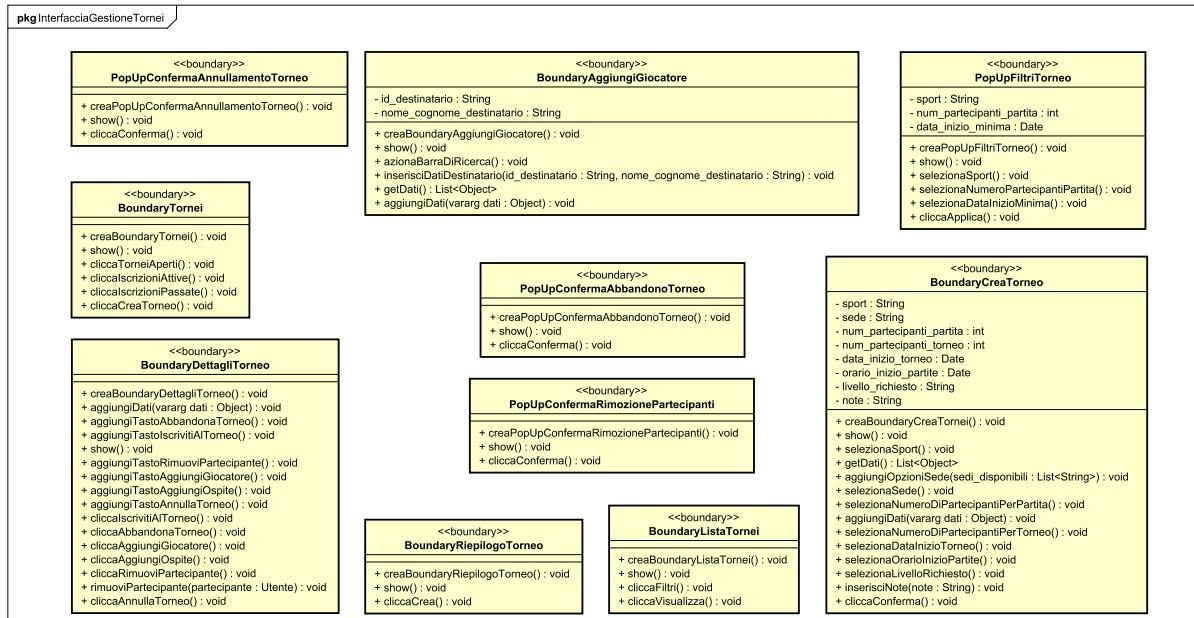
### 3.2.2. Gestione account



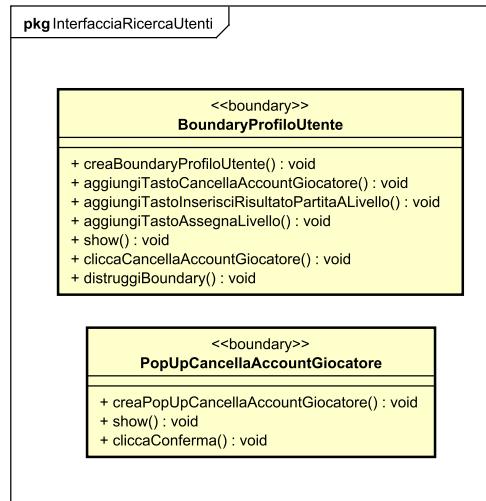
### 3.2.3. Gestione prenotazioni



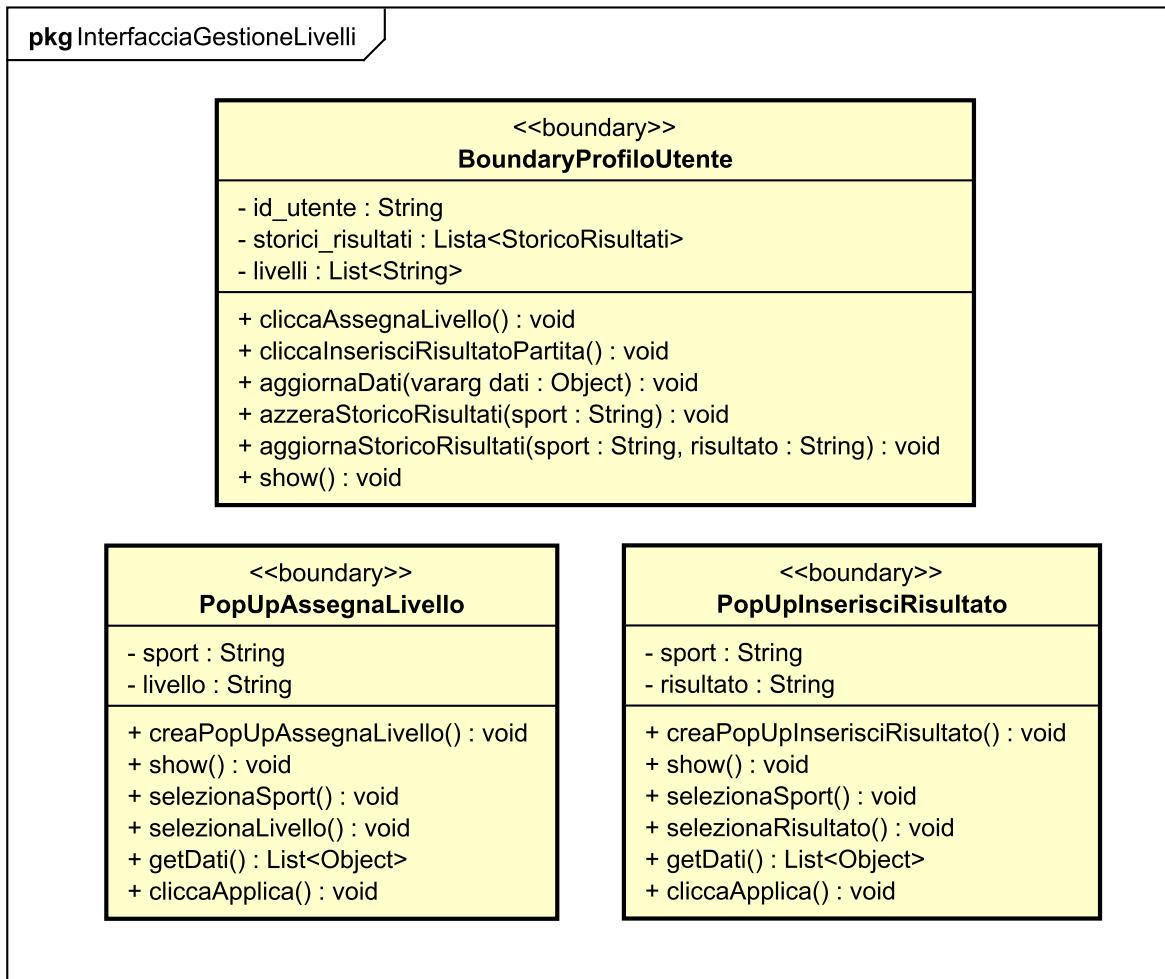
### 3.2.4. Gestione tornei



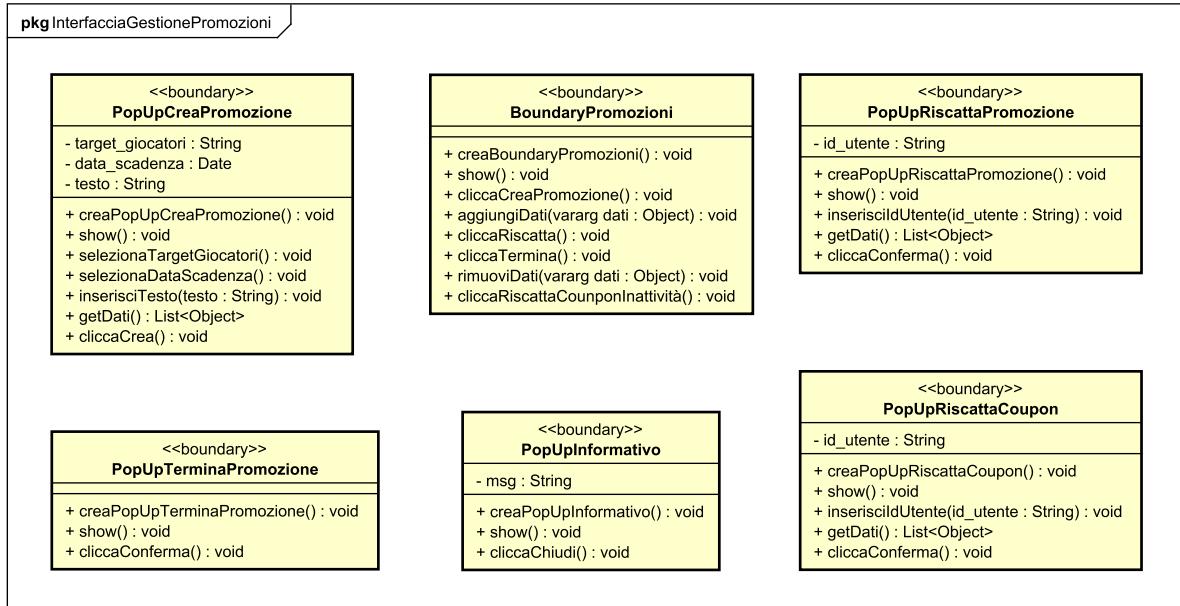
### 3.2.5. Ricerca utenti



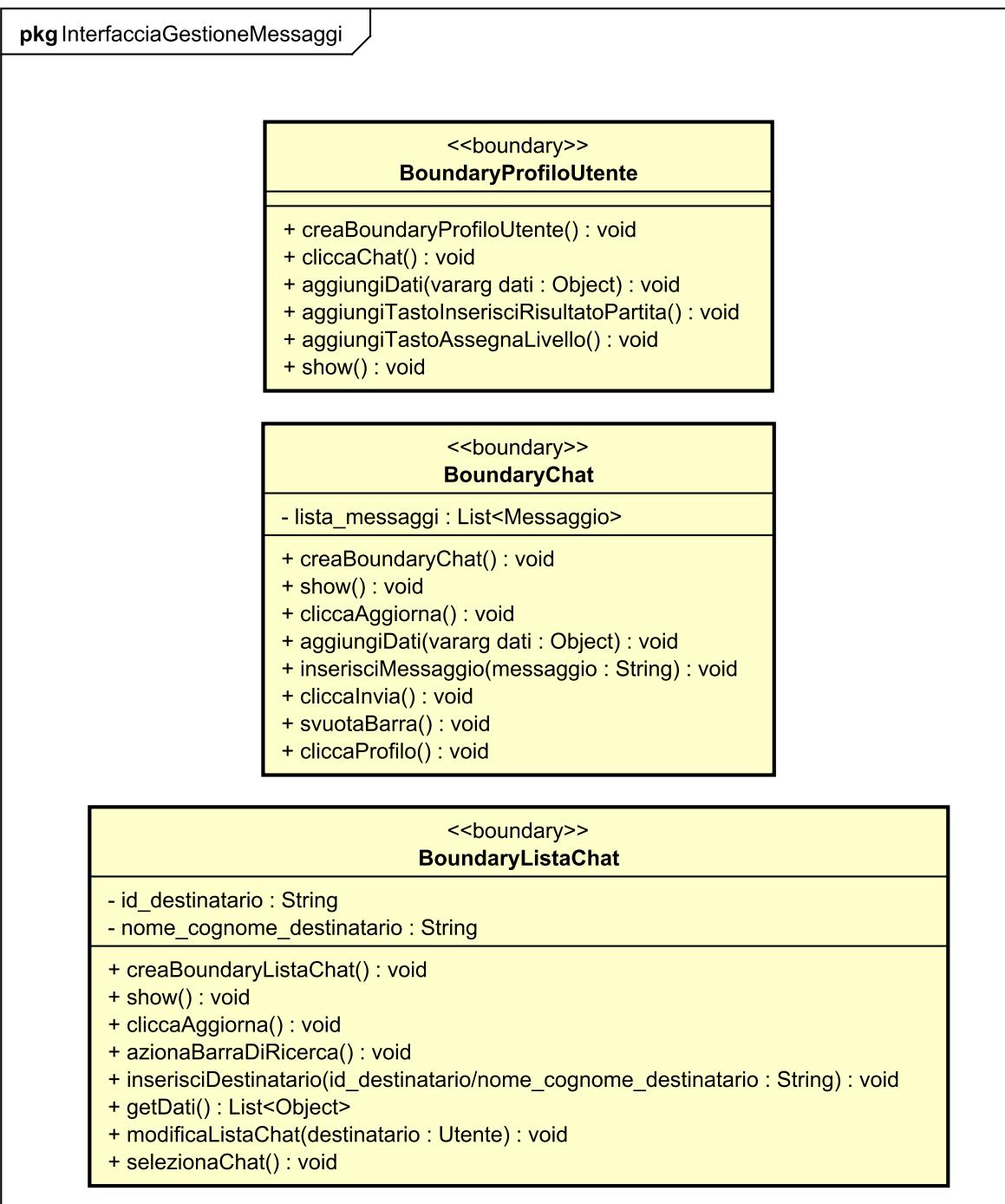
### 3.2.6. Gestione livelli



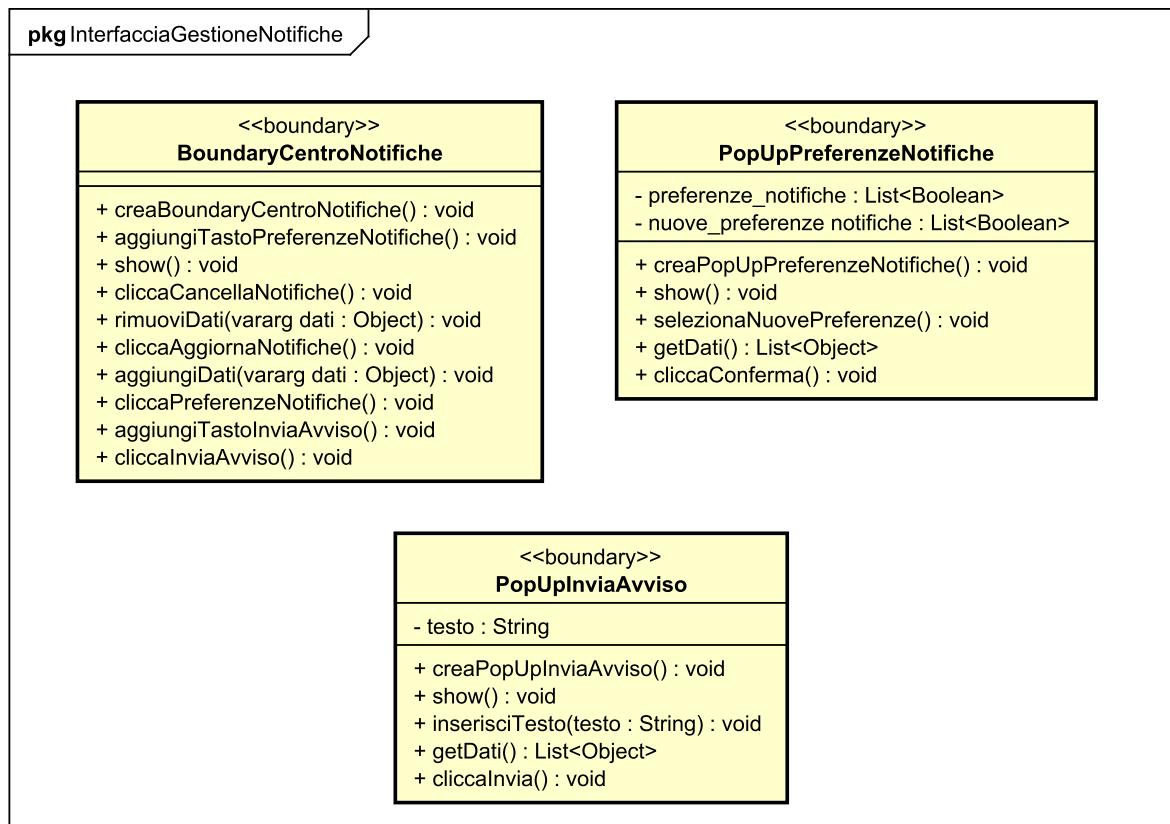
### 3.2.7. Gestione promozioni



### 3.2.8. Gestione messaggi

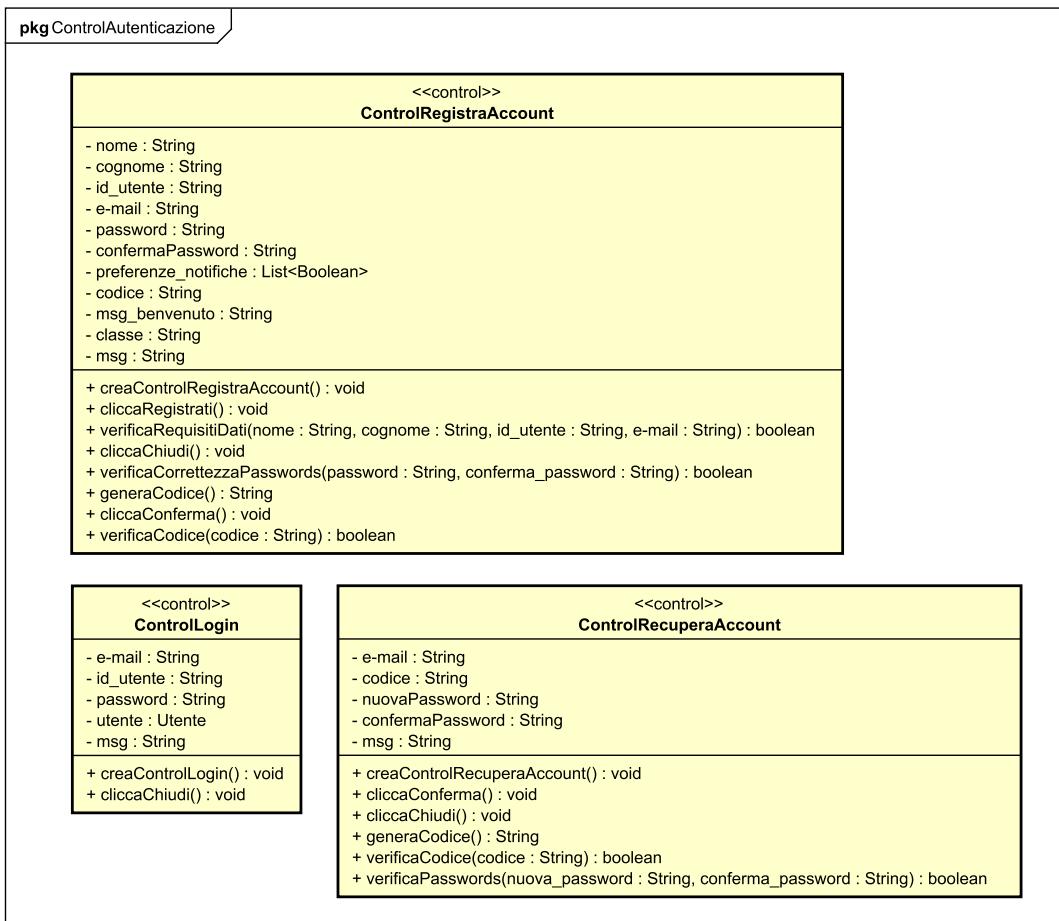


### 3.2.9. Gestione notifiche

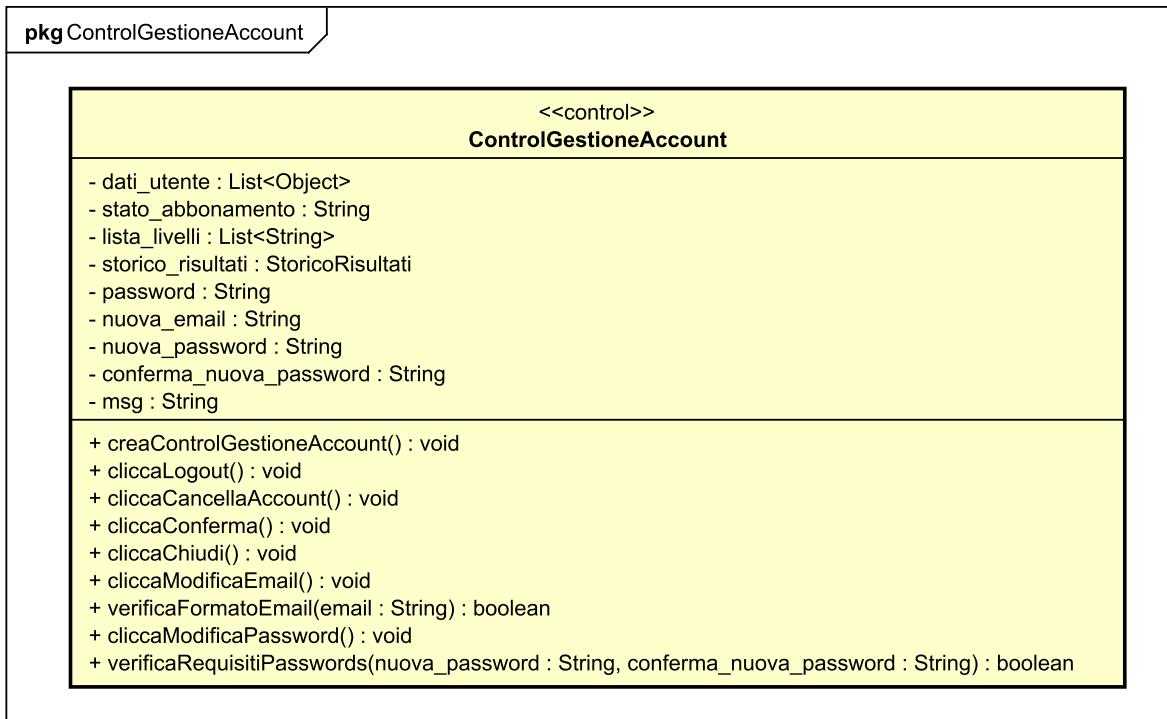


## 3.3. Controls

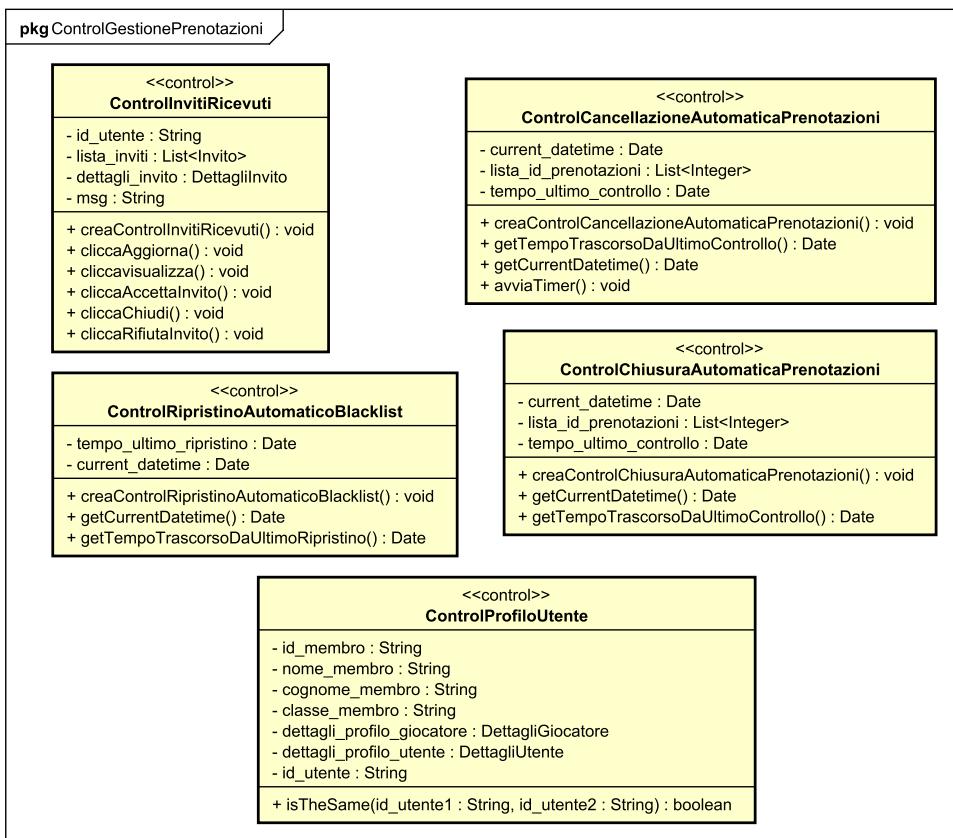
### 3.3.1. Autenticazione



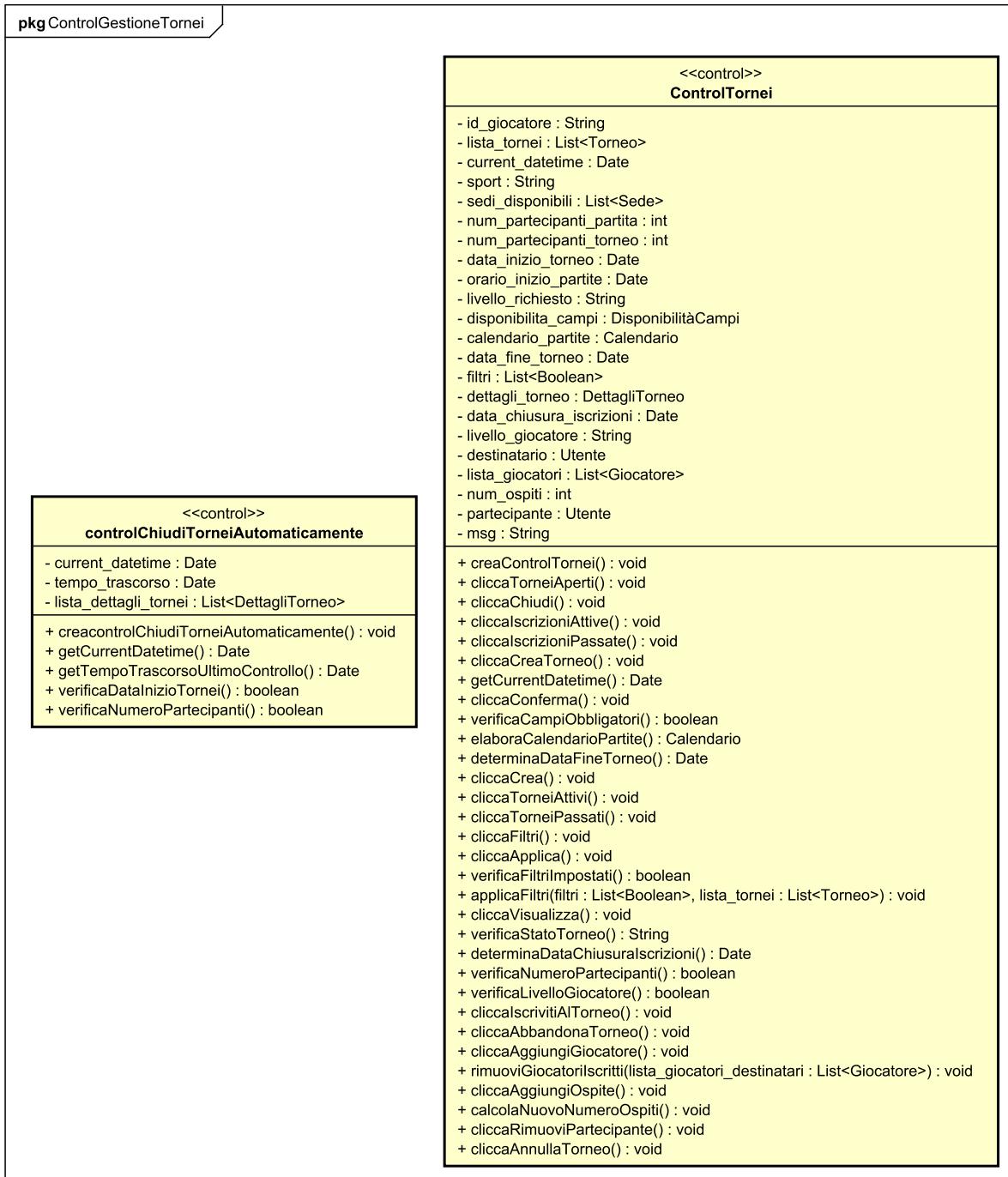
### 3.3.2. Gestione account



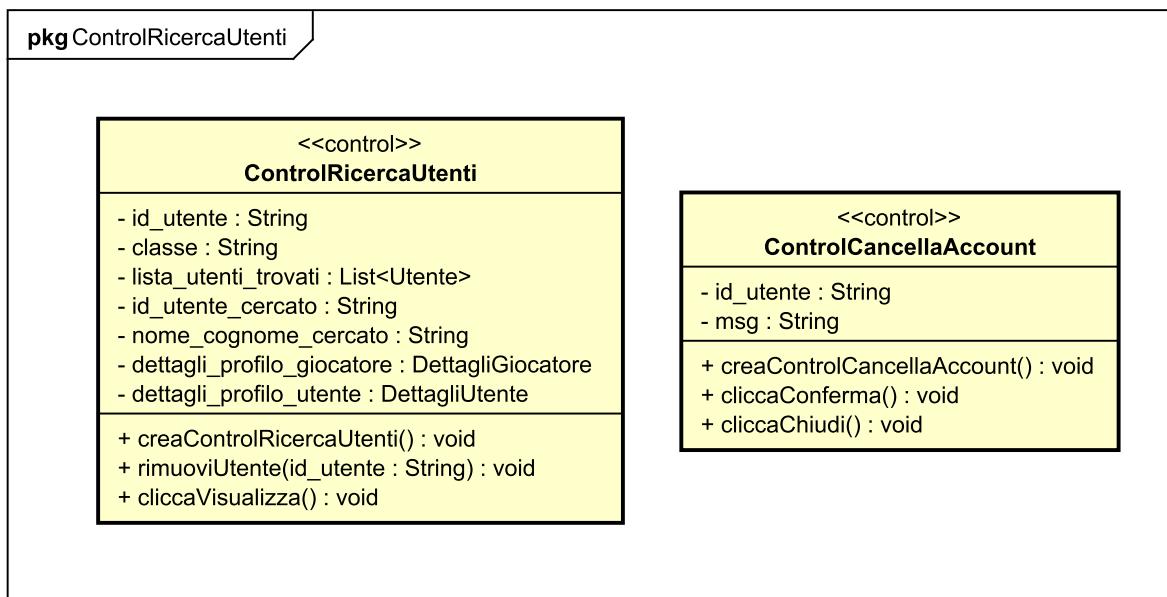
### 3.3.3. Gestione prenotazioni



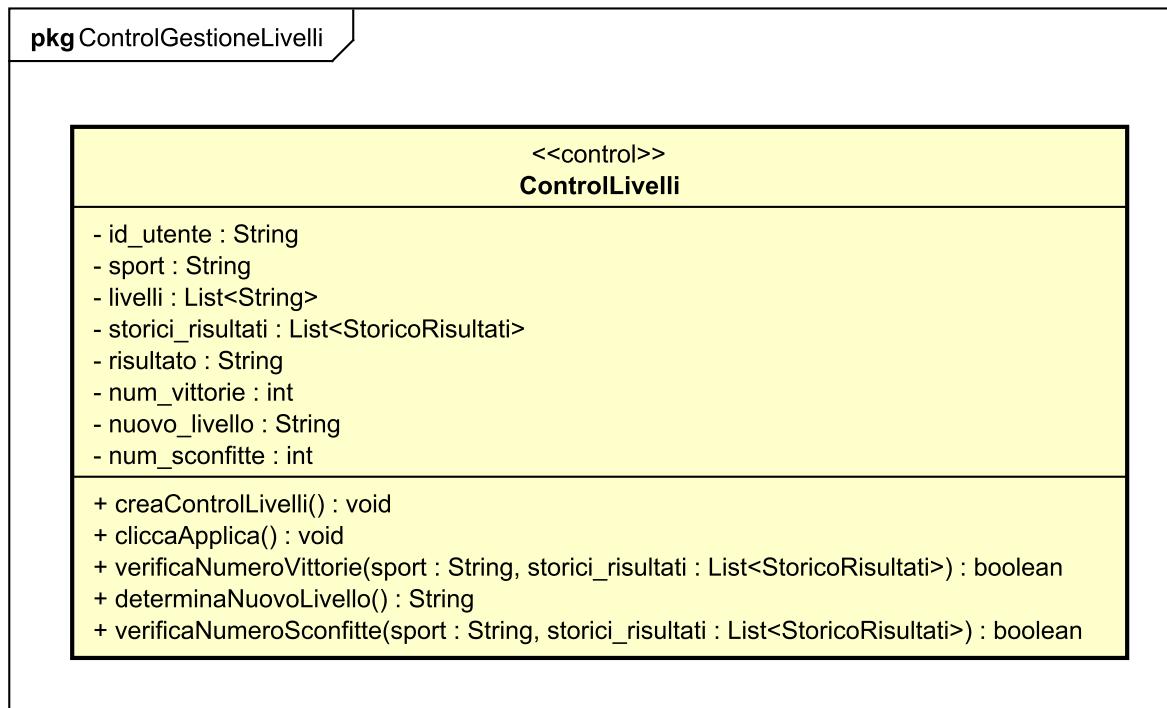
### 3.3.4. Gestione tornei



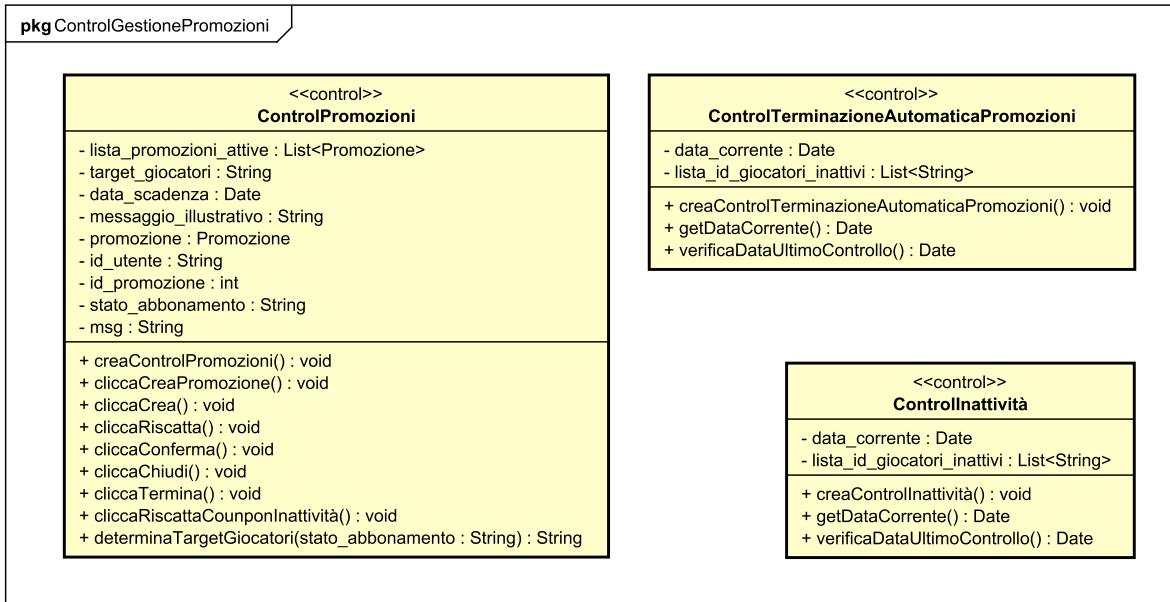
### 3.3.5. Ricerca utenti



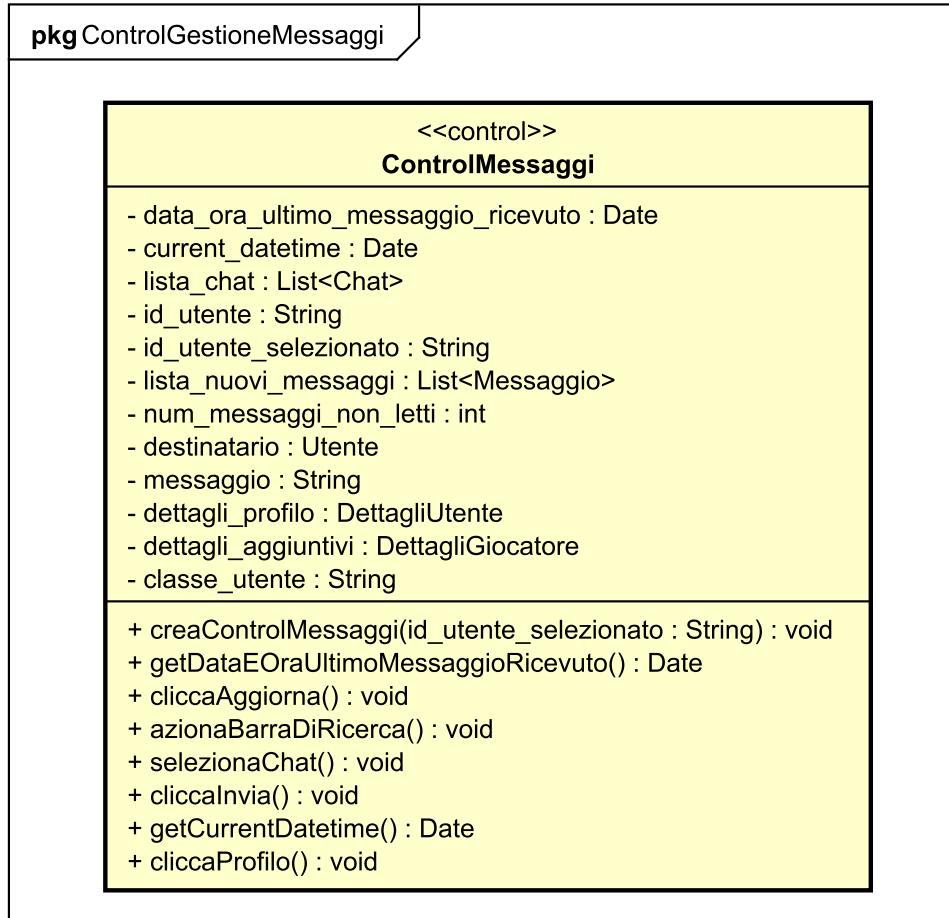
### 3.3.6. Gestione livelli



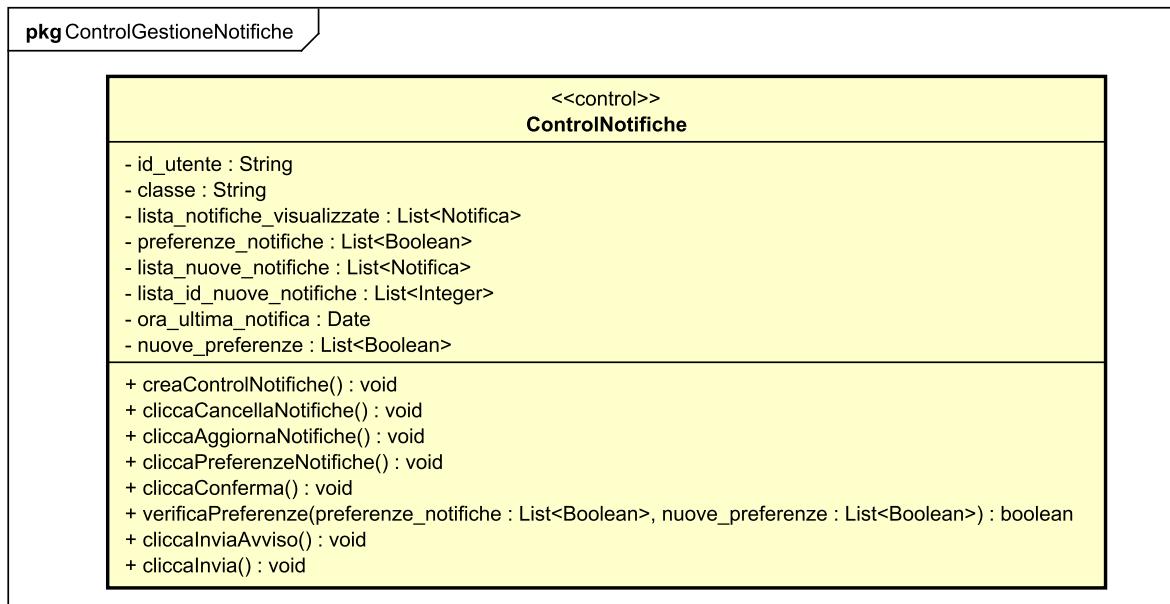
### 3.3.7. Gestione promozioni



### 3.3.8. Gestione messaggi



### 3.3.9. Gestione notifiche



### 3.4. Interfaccia DBMS

