

Model podataka i perzistencije

# HigherLower

*Članovi tima:*

*Jovan Dojčinović 18134*

*Filip Dojčinović, 18135*

*Naziv tima:*

*gang of D*

*Datum: 28..12.2023..*

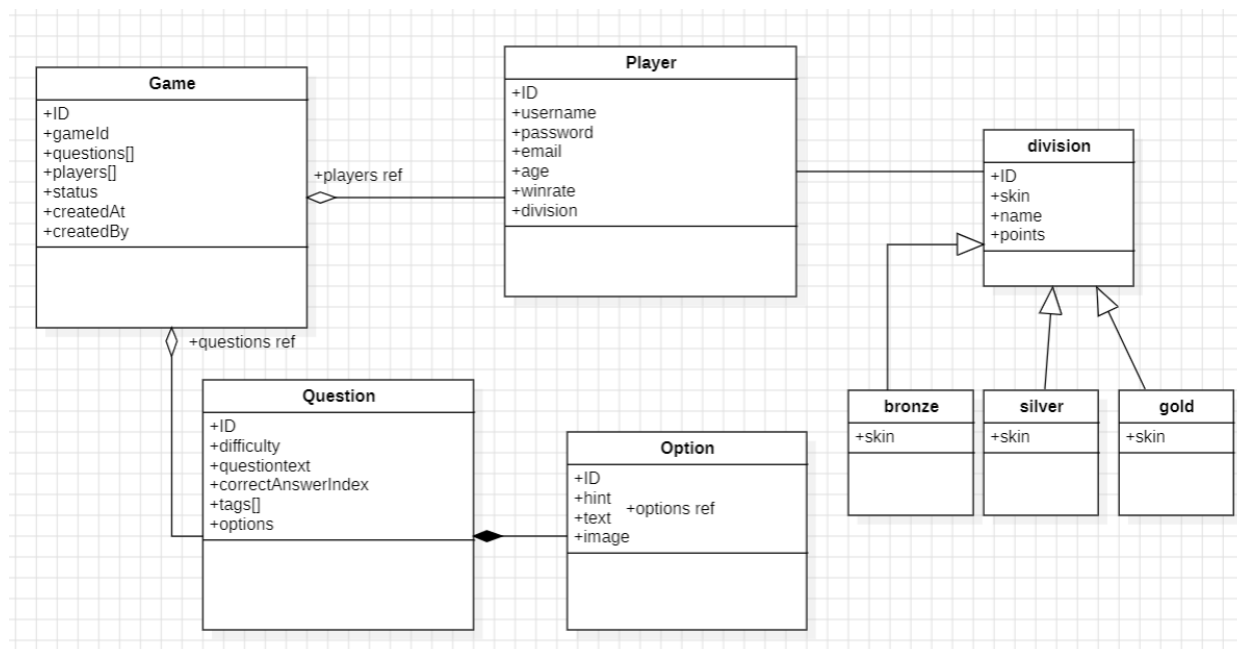
# SADRŽAJ

<b>1</b>	<b>MODEL PODATAKA .....</b>	<b>3</b>
1.1	DIJAGRAM KLASA.....	3
1.2	KLASE.....	3
1.2.1	<i>Game</i> .....	3
1.2.2	<i>Question</i> .....	3
1.2.3	<i>Player</i> .....	4
1.2.4	<i>Option</i> .....	4
1.2.5	<i>Division</i> .....	4
<b>2</b>	<b>MODEL PERZISTENCIJE.....</b>	<b>5</b>
2.1	DIJAGRAM ENTITETA .....	5
2.2	ENTITETI.....	6
2.2.1	<i>Player</i> .....	6
2.2.2	<i>Question</i> .....	5
2.2.3	<i>Game</i> .....	5
<b>3</b>	<b>MEHANIZAM MAPIRANJA.....</b>	<b>6</b>

# 1 MODEL PODATAKA

## 1.1 DIJAGRAM KLASA

Na sledećem dijagramu klasa predstavljen je model podataka sistema Higher Lower:



### 1.2.1 Game Klasa:

Game klasa predstavlja osnovnu strukturu igre. Sadrži informacije o trenutnom statusu igre, učesnicima i nizu nasumično odabranih pitanja. Dodatno, poseduje posebnu logiku za obradu pitanja nakon odgovora igrača. Status igre može biti jedan od tri stringa: "waiting", "inProgress", "completed".

### 1.2.2 Question Klasa:

Question klasa opisuje pitanje u igri sa nizom opcija i rednim brojem tačnog odgovora. Nakon odgovora igrača, pitanje se obrađuje posebnom logikom koja ocenjuje težinu pitanja, prilagođavajući dalji tok igre.

### 1.2.3 Player Klasa:

Player klasa predstavlja igrača učesnika u igri. Igrač može učestvovati u više igara istovremeno, ali se unutar jedne igre pojavljuje samo jednom. Ovo omogućava praćenje statistike igrača na nivou svake pojedinačne igre.

### 1.2.4 Option Klasa:

Option klasa predstavlja opciju u okviru pitanja. Svaka opcija ima svoj tekst, hint (dodatak ili podsticaj koji igračima pomaže u odgovaranju na pitanje) i sliku. Opcije čine sastavni deo strukture pitanja u igri, pružajući dodatne informacije ili vizuelni materijal koji može uticati na odabir igrača prilikom odgovaranja na pitanje.

### 1.2.5. Division klasa

Predstavlja meru uspešnosti korisnika u igrama. Omogućava korisniku da koristi određene avatare, skinove i teme. Implementiran je kao apstraktna klasa.

**1.2.5.1 Bronze Klasa:**

Predstavlja klasu izvedenu iz klase divison. Definise svoj skin, kao i svoju temu

**1.2.5.2. Silver Klasa:**

Predstavlja klasu izvedenu iz klase divison. Definise svoj skin, kao i svoju temu

**1.2.5.3. Gold Klasa:**

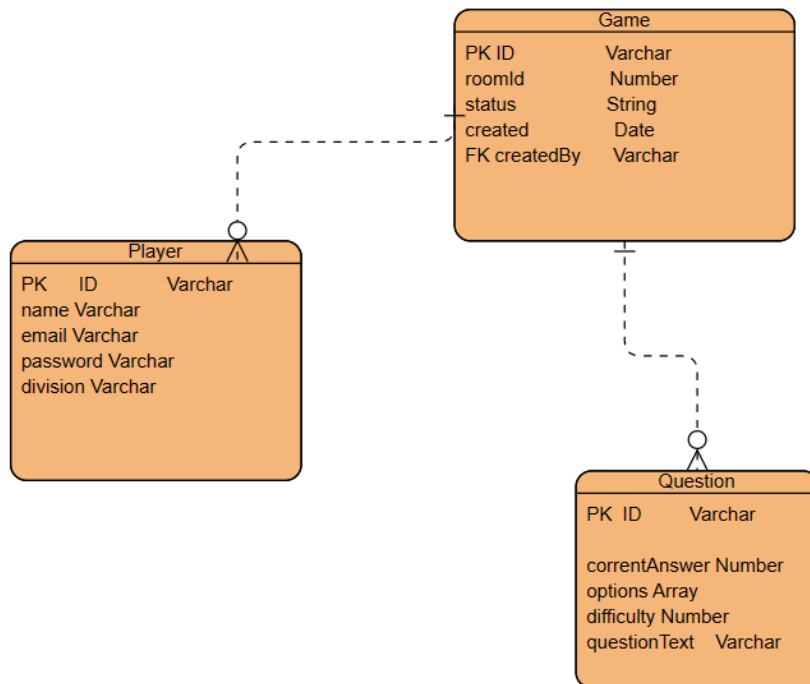
Predstavlja klasu izvedenu iz klase divison. Definise svoj skin, kao i svoju temu

## 2 MODEL PERZISTENCIJE

---

### 2.1 DIJAGRAM ENTITETA

Na sledećem dijagramu prikazani su entiteti sistema:



## **2.2 ENTITETI**

### **2.2.1 Player (Igrač)**

Entitet Player koristi se za čuvanje podataka o igračima. Sadrži ID igrača, koji je povezan sa određenim korisnikom. Jedan korisnik može učestvovati u više igara, pa se njegov ID može povezati sa više igrača. Svaki igrac ima jedinstveni username, kao i podatke neophodne za autorizaciju i autentifikaciju.

### **2.2.2 Question (Pitanje)**

Entitet Question čuva podatke o pitanjima u igri. Svako pitanje ima ID i određeni broj odgovora. Takođe, prati redni broj tačnog odgovora.

### **2.2.3 Game (Igra)**

Entitet Game predstavlja igru koja se trenutno odvija. Sadrži ID igre, informacije o igračima koji učestvuju u igri, pitanjima koja su postavljena i trenutnom statusu igre (waiting, inProgress, completed). Takođe, prati ko je kreirao igru i kada je igra kreirana.

### 3 MEHANIZAM MAPIRANJA

---

Higher&Lower koristi MongoDB kao bazu podataka, a za mapiranje podataka i komunikaciju sa bazom, koristi se Mongoose. Mongoose je biblioteka za MongoDB koja pruža sredstva za definisanje modela podataka, validaciju i olakšava interakciju sa MongoDB kolekcijama.

Projekat se oslanja na arhitekturu koja uključuje modele Mongoose, servise na nivou entiteta za logiku poslovanja i kontrolere koji pružaju usluge klijentima. Mongoose modeli definišu strukturu podataka koja se preslikava na MongoDB kolekcije.

Data Transfer Objects (DTOs) koriste se za prenošenje podataka između različitih delova sistema. U slučaju Higher&Lower, servisi koriste DTO-ove za komunikaciju između Mongoose modela i kontrolera. DTO-ovi predstavljaju jednostavne objekte podataka prilagođene potrebama klijentskih delova sistema, čime se postiže efikasna komunikacija i odvajanje slojeva aplikacije.