

# **Arhitektura i projektovanje softvera**

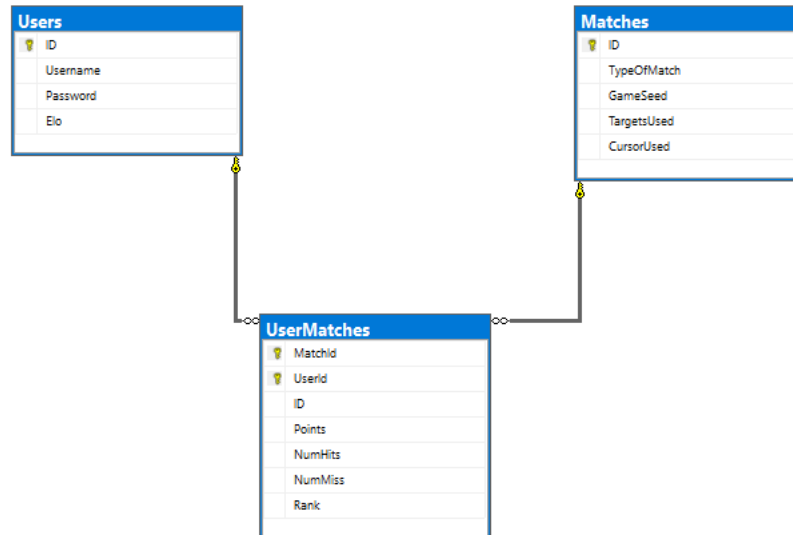
**Naziv projekta:  
AIMPro**

**Model podataka i perzistencije**

Članovi tima:  
Milan Janković 16114  
Marko Krstić 16153  
Stefan Stanojević 16361

Softvverski proizvod AIMPro pamti korisnike, kao i njihove potrebne atribute kao što su username i password za njihovu autentifikaciju i njihov trenutni rating, takodje je potrebno pamti i istoriju odigranih meceva i postignute rezultate u njima. To je realizovano preko tri tabele:

- User - pamti podatke vezane za korisnika
  - Id tipa integer je jedinstveni broj korisnika
  - Username tipa string je njegov nickname koji se prikazuje
  - Password kao niz bajtova je šifra preko koje potvrđuje svoj identitet
  - Elo tipa integer je korisnikov rezultat postignut u igri
- Match - partija koja je odigrana
  - Id tipa integer je jedinstveni broj partije
  - Tip meca koji je odigran
  - Gameseed je niz bajtova preko koga moze da rekreira partiju
  - Mete koje su se koristile u partiji
  - Kursori koji su korišćeni u partiji
- UserMatch - vrši preslikavanje n:m tablea User i Match i pamti postignute rezultate odgovarajućeg korisnika u odigranoj partiji
  - MatchID tipa integer je strani ključ i pokazuje za koju partiju su podaci
  - UserID tipa integer je strani ključ i pokazuje koji korisnik je postigao odgovarajuće rezultate
  - Points tipa integer je broj postignutih poena u partiji
  - NumHits tipa integer je broj pogodaka korisnika u partiji
  - NumMisses tipa integer je broj promasaja korisnika u partiji
  - Rank tipa integer je mesto koje je osvojio korisnik u partiji



AIMPro koristi entity framework za mapiranje objekata u bazi code first metodom. Implementirane su 3 klase koje se mapiraju u bazi: Match, UserMatch, User. Klasa UserContext je klasa zaduzena za mapiranje objekata. DataBaseAPI klasa koja sadrži potrebne funkcije za rad sa bazom podataka:

- AddUser -dodaje novog korisnika u bazi podatkaa
- UpdateUsers - koja dodaje novi mec i rezultate meca odgovarajucem korisniku
- GetProfileStatistics - vraca statistiku odgovarajuceg korisnika
- GetMatchHistory - vraca istoriju meceva za odgovarajuceg korisnika

- RemoveUser - brise korisnika iz baze
- GetUser - vraca korisnika iz baze podataka sa istorijom meceva
- GetUserWithoutHistory - vraca korisnika bez istorije meceva

Korisnici mogu da pristupe bazi podataka preko odgovarajucih funkcija Server API. Funkcije vracaju podatke iz baze podataka preko klasa User, RoomState i Profile. Komponenta Query je zaduzena za poziv odgovarajucih funkcija DataBaseAPI kako bi vratila odgovarajuce podatke korisniku. Dok GameLogic je zaduzena za upis novih podataka generisanih tokom igranja igre u bazi podataka.

