

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ORGANIZACIJE I INFORMATIKE**  
**VARAŽDIN**

**Tomislav Dorić**

**Tomislav Kačmarčik**

**Valentina Magdić**

**Tihana Poljak**

**Dario Rebrović**

# **APLIKACIJA ZA NOGOMETNI KLUB**

**Projekt iz kolegija Programsko inženjerstvo**

**Varaždin, 2014.**

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ORGANIZACIJE I INFORMATIKE**  
**VARAŽDIN**

**Tomislav Dorić, 39183/10-R**

**Tomislav Kačmarčik, 40588/11-R**

**Valentina Magdić, 38198/09-R**

**Tihana Poljak, 39212/10-R**

**Dario Rebrović, 39221/10-R**

## **APLIKACIJA ZA NOGOMETNI KLUB**

**Projekt iz kolegija Programsko inženjerstvo**

**Nositelj kolegija:**

**Prof. dr.sc. Vjeran Strahonja**

**Mentor:**

**Ivan Švogor, mag.inf.**

**Varaždin, travanj 2014.**

# Sadržaj

1. Uvod.....	1
2. UML dijagrami .....	2
2.1. Dijagram slučajeve korištenja .....	2
2.2. Dijagrami aktivnosti .....	4
2.2.1. Logiranje u sustav .....	4
2.2.2. Pregled igrača.....	5
2.2.3. Planer .....	6
2.2.4. Pregled utakmica.....	7
2.2.5. Pregled timova .....	8
2.2.6. Statistika igrača .....	9
2.3. Dijagrami slijeda.....	10
2.3.1. Logiranje u sustav .....	10
2.3.2. Pregled igrača.....	11
2.3.3. Planer .....	12
2.3.4. Pregled utakmica.....	13
2.3.5. Pregled timova .....	15
2.3.6. Statistika.....	17
3. ERA model.....	18
4. Dijagram klasa .....	20

## 1. Uvod

Po našim saznanjima u Hrvatskoj ne postoji jedinstvena aplikacija kojom bi se pratio rad jednog nogometnog kluba. To se odnosi na nemogućnost brzog i efikasnog praćenja pojedinih nogometaša u jednom klubu. Dosad se to obavljalo na način da trener piše zabilješke o nogometašima na papire te na taj način dolazi do gomilanja istih u kojima se treneri poslije ne mogu snaći. Također, ne postoji aplikacija na kojoj bi se treneri mogli informirati o budućim aktivnostima u klubu, kao na primjer treninzima, utakmicama i skupštinama.

Tehnologija označava način na koji se ostvaruju zadani ciljevi i koraci projekta, pri čemu će nam pomoći razni računalni alati. Za kreiranje dokumentacija i opisivanje dijagrama, slika i tablica koristimo MS Word. Za planiranje razdoblja izrade samog projekta, dodjeljivanje zadataka, planiranje budžeta koristimo MS Project. Visual Paradigm for UML je alat koji smo koristili za kreiranje svih UML dijagrama.

Za izradu same aplikacije koju ćemo raditi u drugoj fazi projekta koristit ćemo Microsoft Visual Studio 2012, a za kreiranje baze podataka Microsoft SQL.

Tehnička dokumentacija sadrži slike i opise UML dijagrama, ERA modela i dijagrama klasa.

## 2. UML dijagrami

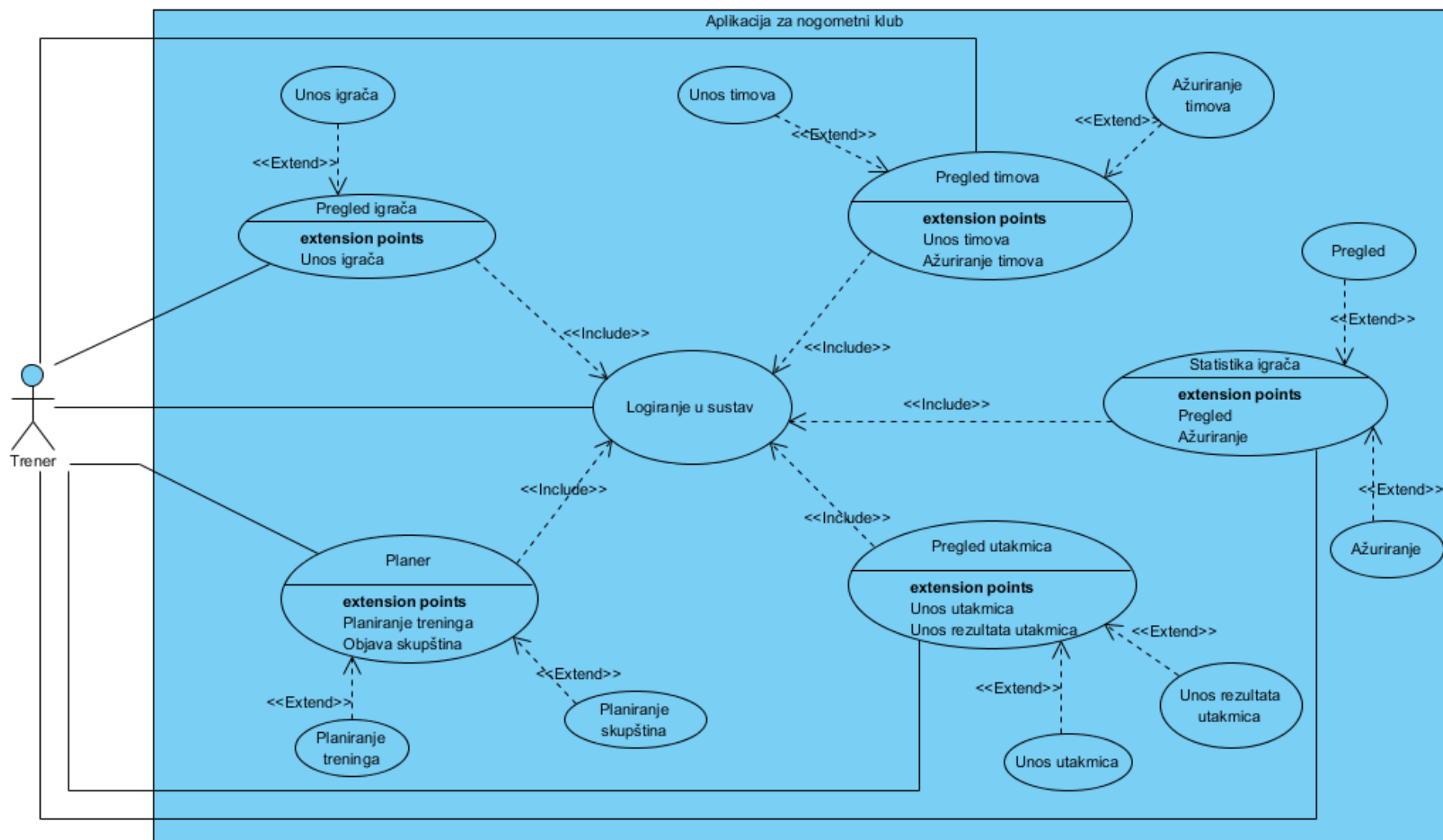
U nastavku tehničke dokumentacije nalaze se dijagrami za daljnji razvoj projekta, a to su dijagram slučajeva korištenja, dijagrami aktivnosti i dijagrami slijeda.

### 2.1. Dijagram slučajeva korištenja

U nastavku slijedi dijagram slučajeva korištenja koji se sastoji od 6 slučajeva korištenja:

- Logiranje u sustav
- Unos igrača
- Pregled timova
- Pregled utakmica
- Statistika igrača
- Planer

Svaki trener koji bude koristio ovu aplikaciju, mora se najprije logirati sa svojim korisničkim imenom. Nakon toga ima mogućnost odabira pregleda igrača u slučaju da želi unijeti nekog novog igrača. Ima mogućnost unositi nove timove, pregledavati dosadašnje timove te ih ažurirati. Također može pregledavati utakmice i unositi rezultate utakmica. Postoji i planer gdje može upisivati datume za buduće treninge i skupštine. Statistika omogućava treneru da vidi koliko pojedini igrač ima zabijenih golova, koje kartone je dobio i sl.

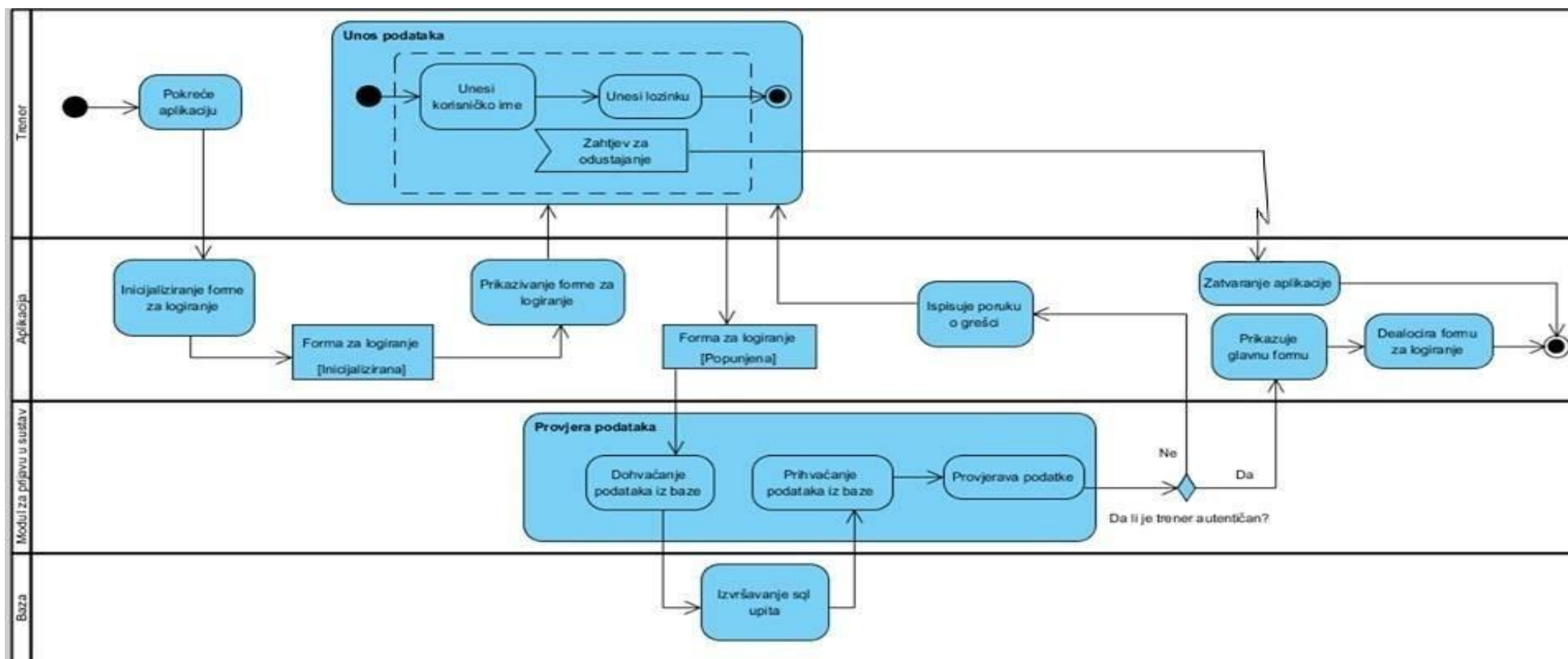


Slika 2.1. Dijagram slučajeva korištenja

## 2.2. Dijagrami aktivnosti

### 2.2.1. Logiranje u sustav

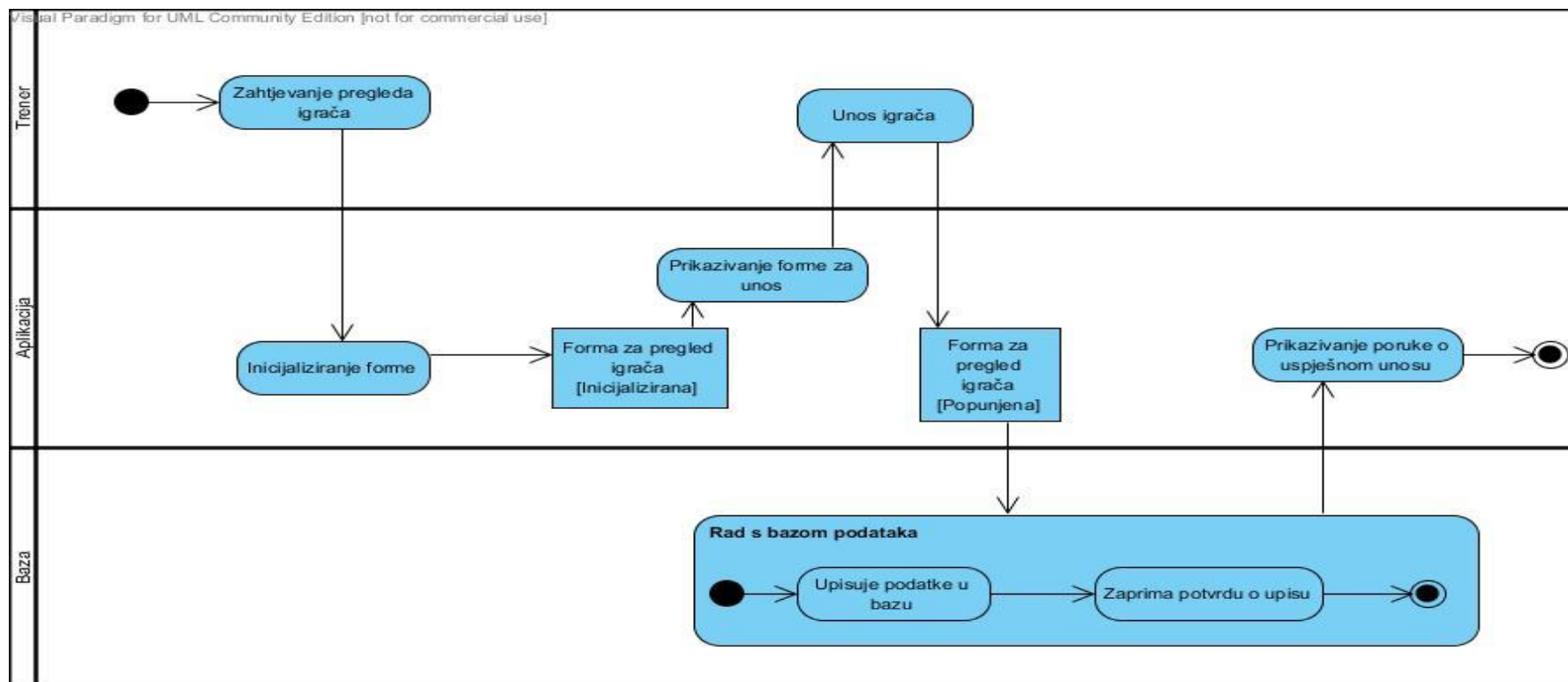
Prilikom pokretanja aplikacije treneru se prikaže forma za logiranje gdje on upiše svoje korisničko ime i lozinku. Podaci se provjeravaju u bazi i nakon toga upitom se provjerava da li postoji taj trener. Ako su podaci točni prikazuje se glavna forma, a ako nisu ispisuje se poruka o pogrešci te se ponovno prikazuje forma za prijavu.



Slika 2.2.1: Dijagram aktivnosti- Logiranje u sustav

## 2.2.2. Pregled igrača

Ovaj dijagram aktivnosti predstavlja način na koji trener pregledava i unosi igrače u bazu. Prvi trenerov korak prilikom izbora pregleda igrača, nakon što je logiran u sustav je slanje zahtjeva za pregled igrača. Nakon toga sustav inicijalizira modul za pregled, aplikacija pokazuje treneru formu u kojoj trener unosi sve igrače koje je mislio te se tada ti podaci od aplikacije šalju dalje na upis u bazu podataka. Nakon što je baza podataka uspješno obavila upis trenerovih igrača, šalje poruku o uspješnom upisu koju trener vidi. Zadnji korak je taj da sustav dealocira bazu.

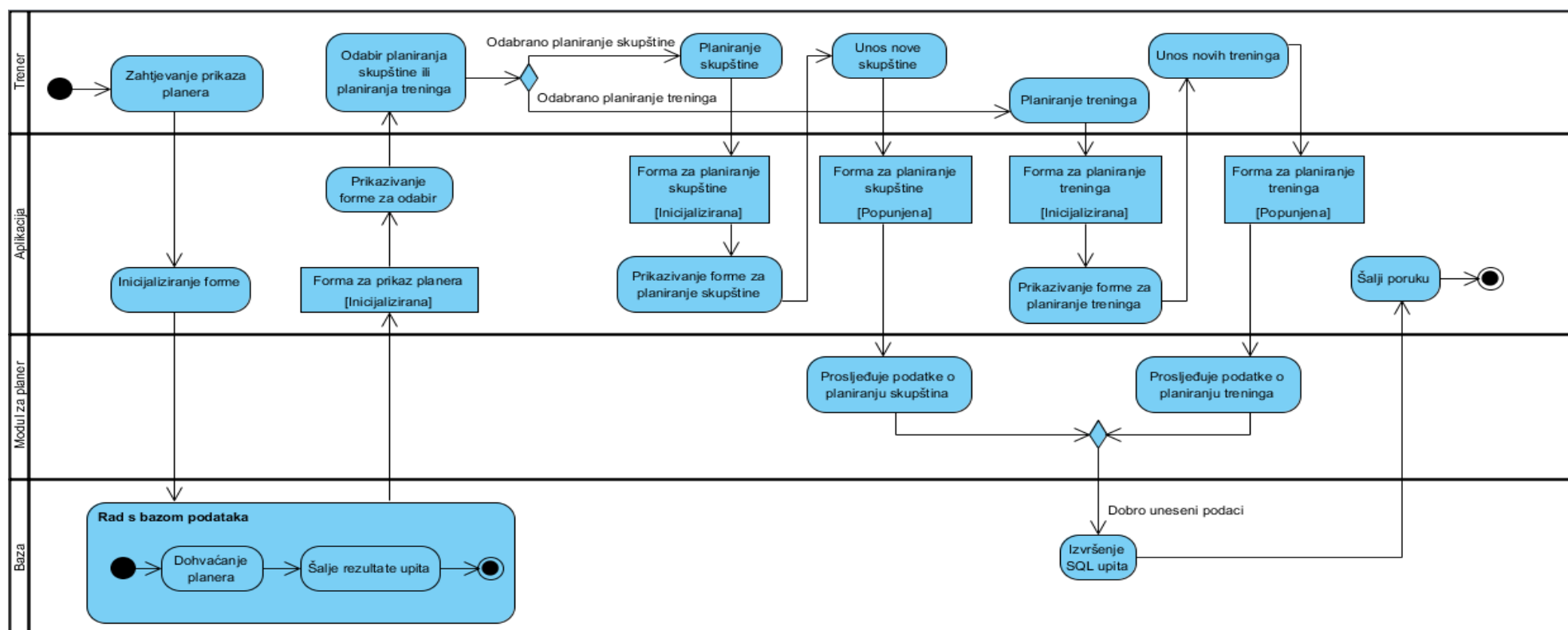


Slika 2.2.2: Dijagram aktivnosti- Pregled igrača



### 2.2.3. Planer

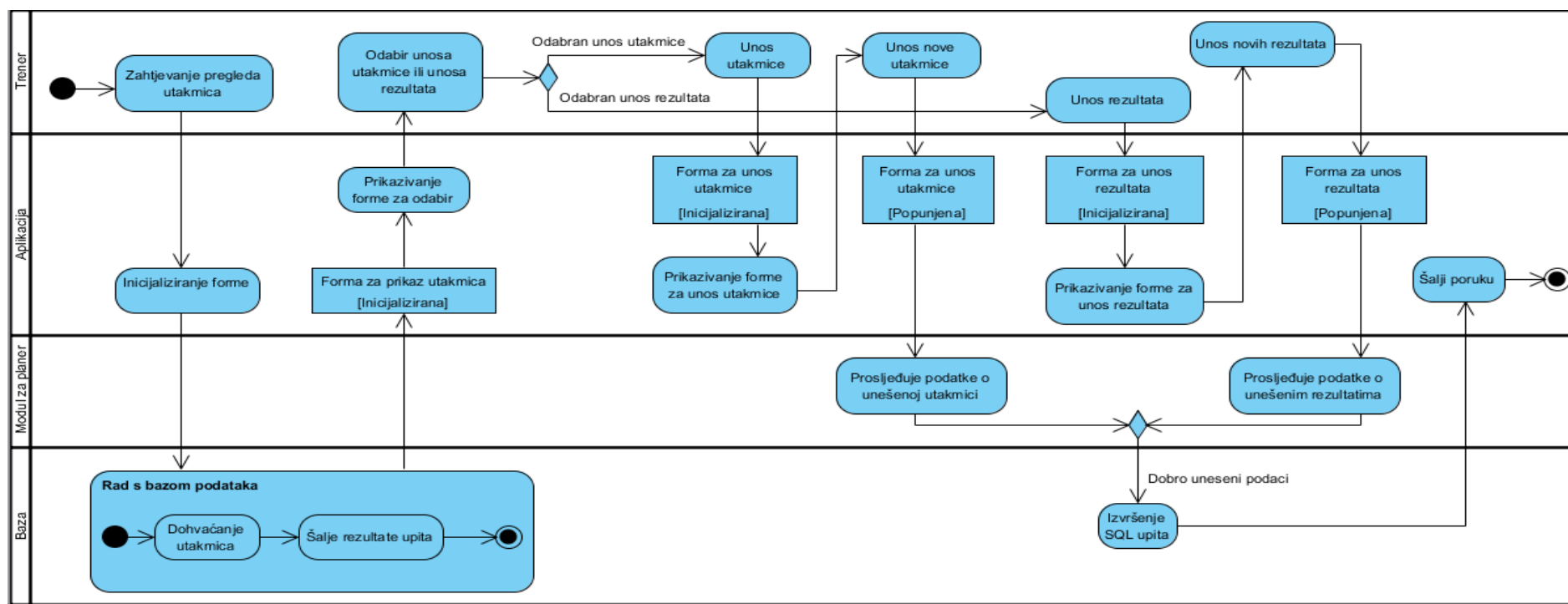
Ovaj dijagram predstavlja način na koji trener planira skupštinu ili trening. Prvi trenerov korak prilikom izbora planera, nakon što je logiran u sustav je slanje zahtjeva za prikaz planera. Nakon toga se dohvaćaju svi dosadašnji podaci iz baze i aplikacija prikazuje formu za odabir. Tu trener ima mogućnost odabrati planiranje skupštine ili planiranje treninga. Ako trener izabere opciju planiranje skupštine, dobiva formu za unos u koju unosi skupštinu. Ako je izabrana opcija planiranje treninga, trener dobiva formu u koju unosi treninge. Nakon uspješno unesenih podataka modul te podatke prosljeđuje do baze. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom ažuriranju.



Slika 2.2.3: Dijagram aktivnosti- Planer

## 2.2.4. Pregled utakmica

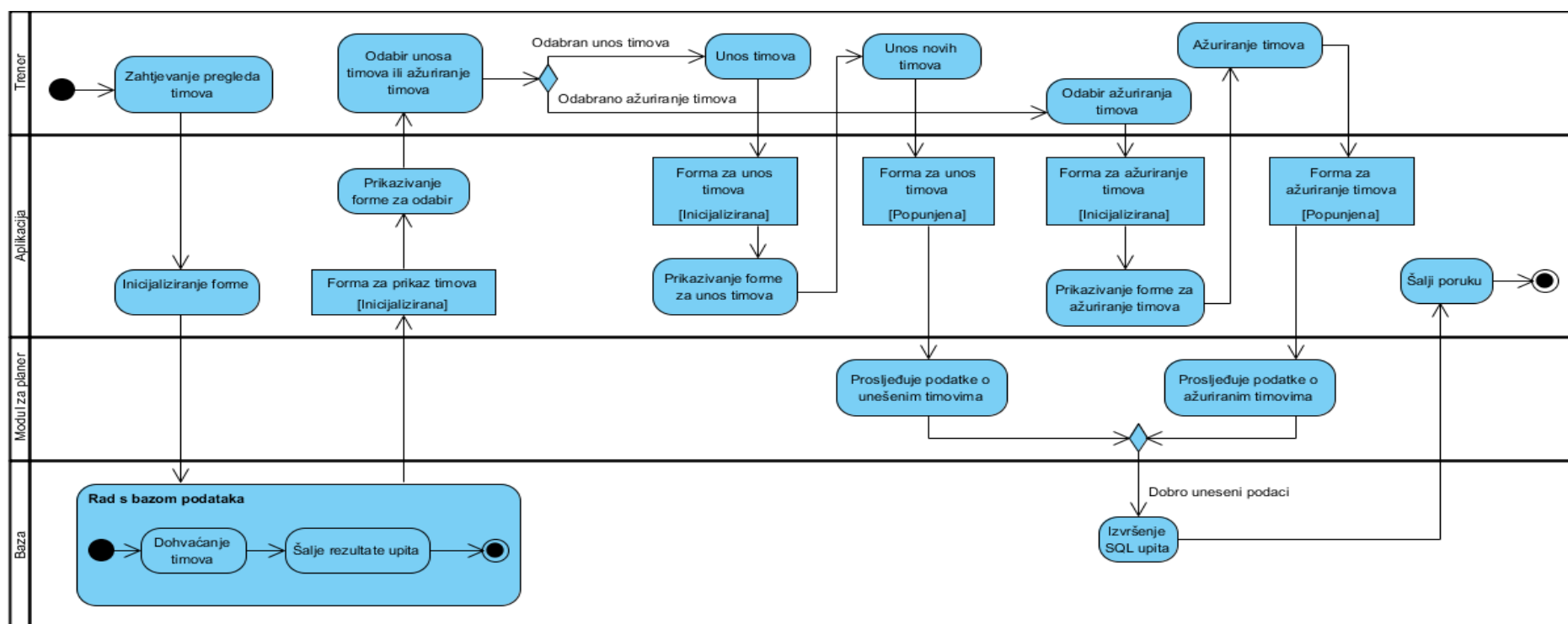
Ovaj dijagram predstavlja način na koji trener unosi utakmice ili rezultate utakmica. Prvi trenerov korak prilikom pregleda utakmica, nakon što je logiran u sustav je slanje zahtjeva za prikaz utakmica. Nakon toga se dohvaćaju svi dosadašnji podaci iz baze i aplikacija prikazuje formu za odabir. Tu trener ima mogućnost odabrati unos utakmice ili unos rezultata. Ako trener izabere opciju unos utakmice, dobiva formu za unos u koju unosi utakmicu. Ako je izabrana opcija unos rezultata, trener dobiva formu u koju unosi rezultate trenutno odigrane utakmice. Nakon uspješno unesenih podataka modul te podatke prosljeđuje do baze. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom unosu utakmica ili uspješnom unosu rezultata.



Slika 2.2.4: Dijagram aktivnosti- Pregled utakmica

## 2.2.5. Pregled timova

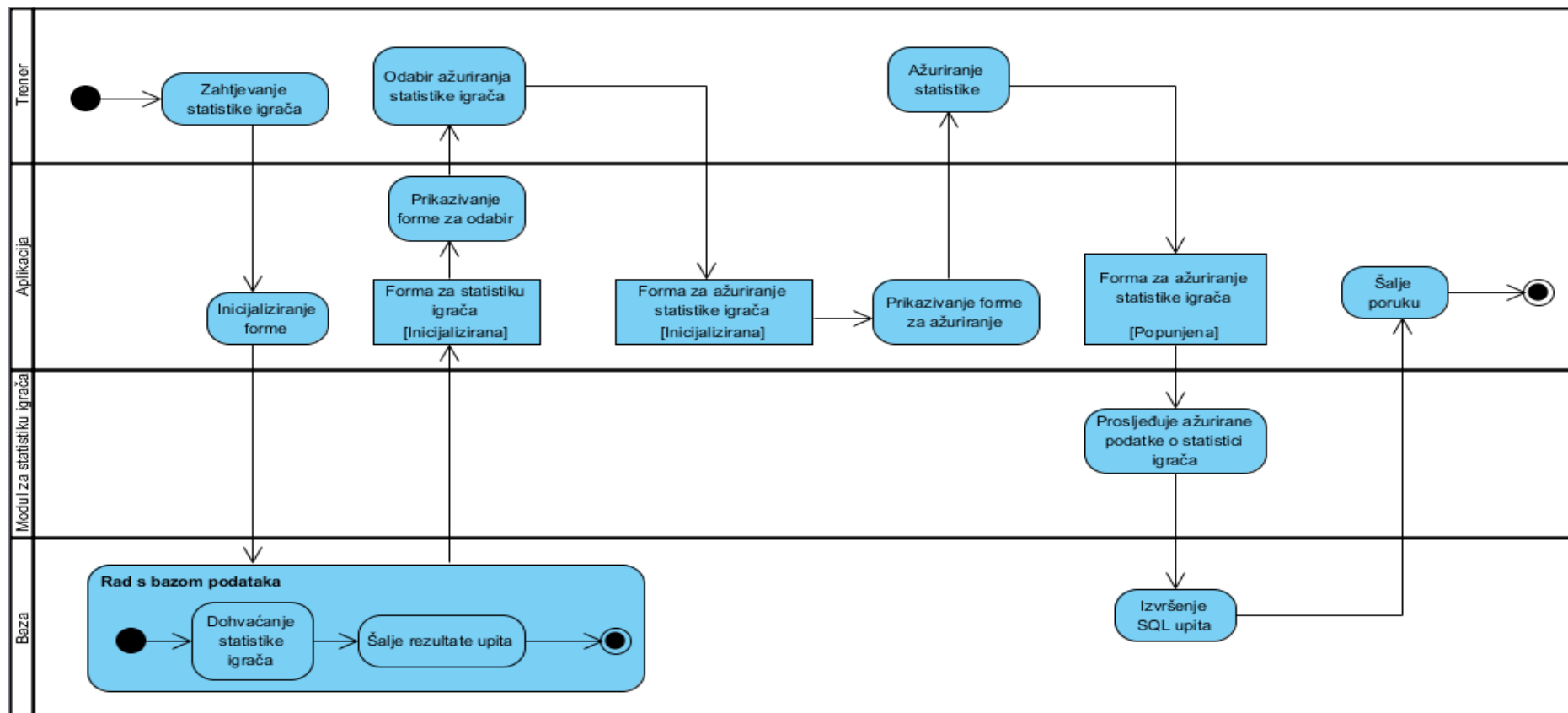
Ovaj dijagram predstavlja način na koji trener unosi nove timove ili ažurira postojeće. Prvi trenerov korak prilikom pregleda timova, nakon što je logiran u sustav je slanje zahtjeva za pregled timova. Nakon toga se dohvaćaju svi dosadašnji podaci iz baze i aplikacija prikazuje formu za odabir. Tu trener ima mogućnost odabrati unos timova ili ažuriranje timova. Ako trener izabere opciju unos timova, dobiva formu za unos u koju unosi timove. Ako je izabrana opcija ažuriranje timova, trener dobiva formu u gdje ažurira već unesene timove. Nakon uspješno unesenih podataka modul te podatke prosljeđuje do baze. Dalje se izvršava SQL upit u bazi te dobivamo prikaz o uspješnom unosu timova ili ažuriranju.



Slika 2.2.5: Dijagram aktivnosti- Pregled timova

## 2.2.6. Statistika igrača

Ovaj dijagram predstavlja način na koji trener ažurira statistiku. Prvi trenerov korak prilikom izbora statistike, nakon što je logiran u sustav je slanje zahtjeva za statistiku igrača. Nakon toga dohvaćaju se svi dosadašnji podaci iz baze gdje trener vidi dosadašnju statistiku igrača. Trener ima mogućnost odabira ažuriranja gdje dobiva formu gdje izvršava ažuriranje. Nakon toga se ažurirani podaci proslijeđuju do baze. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom ažuriranju statistike.

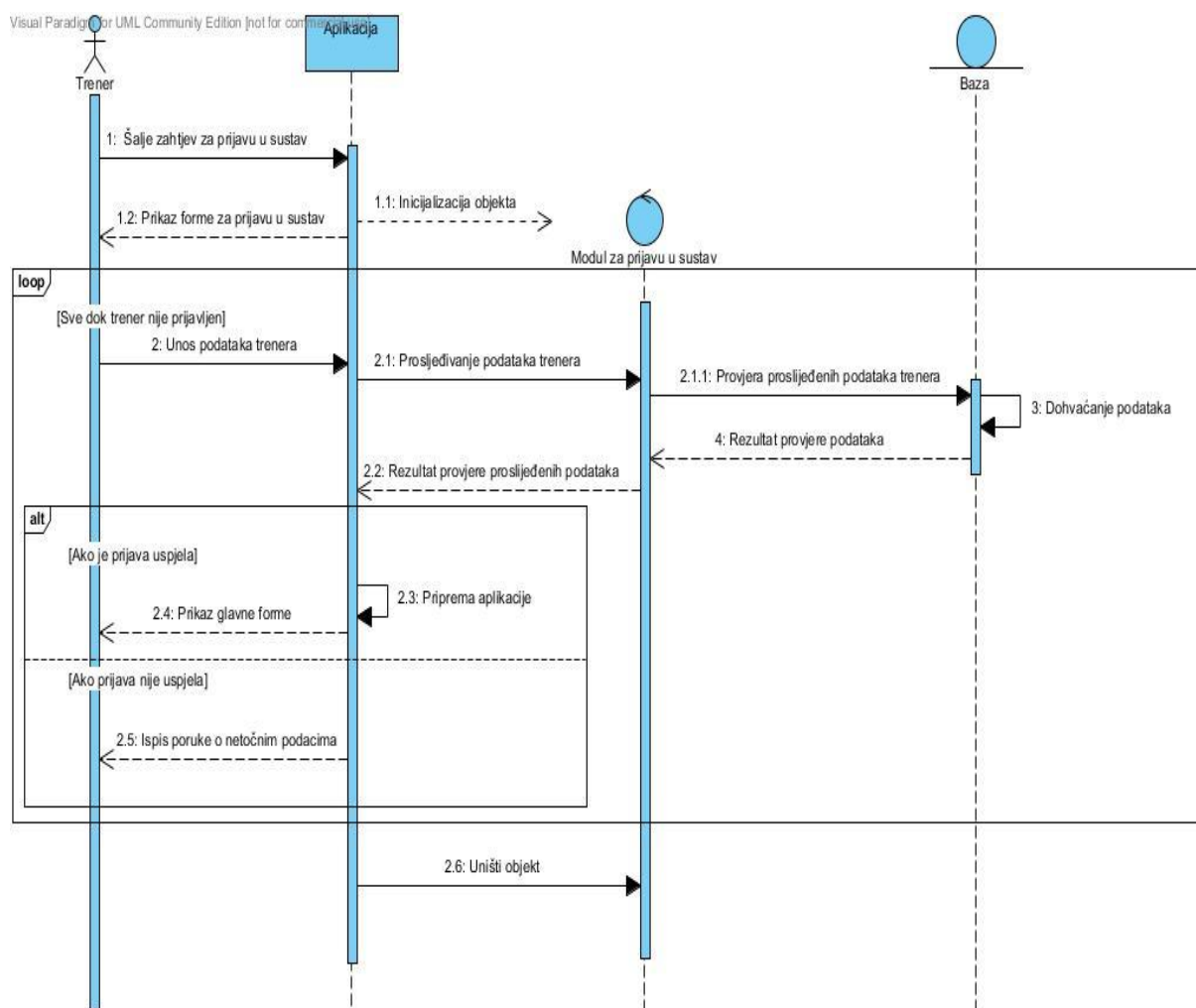


Slika 2.2.6: Dijagram aktivnosti- Statistika igrača

## 2.3. Dijagrami slijeda

### 2.3.1. Logiranje u sustav

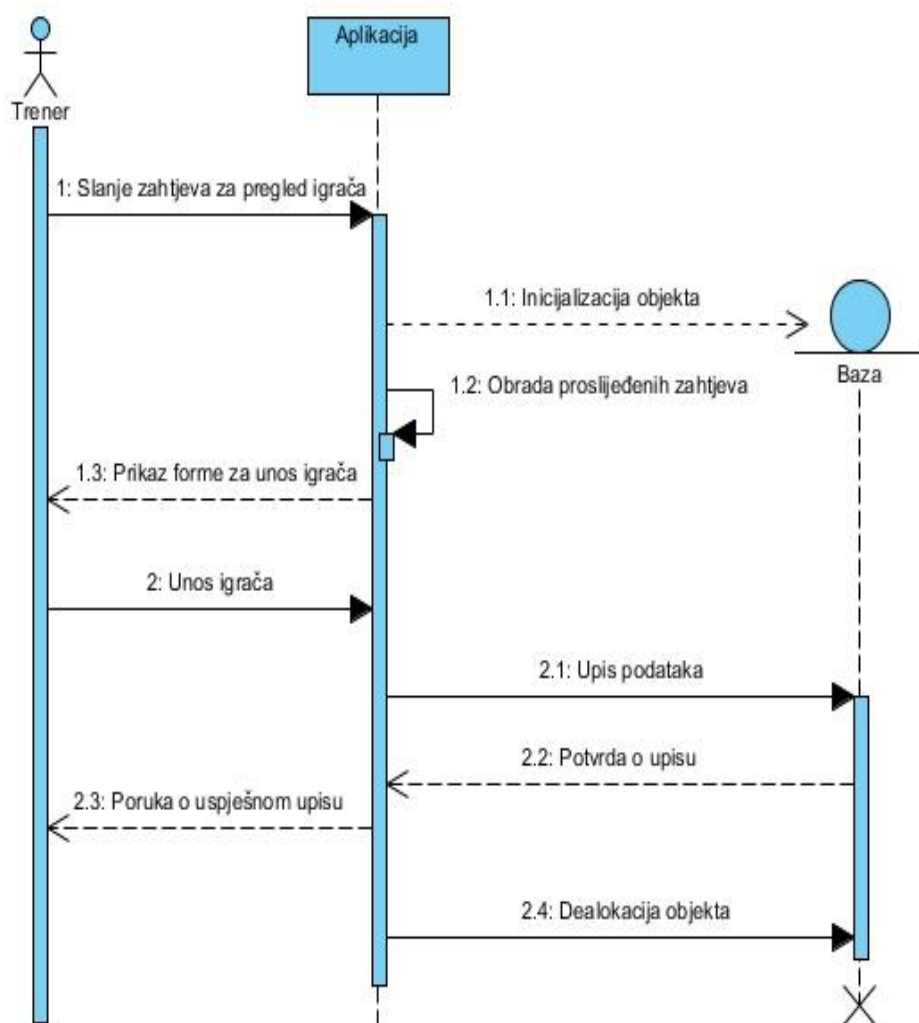
Prvi trenerov korak kod logiranja u sustav je slanje zahtjeva za logiranje sustavu. Nakon toga sustav inicijalizira modul za logiranje, aplikacija pokazuje treneru formu u kojoj se trener prijavljuje u sustav te pokreće petlju koja se vrti sve dok korisnik nije logiran. Na početku petlje sustav korisniku prikazuje formu za logiranje u koji korisnik unosi svoje podatke. Sustav podatke prosljeđuje modulu koji ih provjerava u bazi te nakon toga vraća status rezultata sustavu. Ako je prijava uspjela sustav pripremi aplikaciju te korisniku prikazuje glavnu formu. U slučaju da prijava nije uspjela sustav ispisuje korisniku poruku o grešci te se vraća na početak petlje. Nakon što je korisnik logiran sustav dealocira modul za logiranje. Slika 2.3.1. prikazuje dijagram slijeda Logiranje u sustav.



Slika 2.3.1: Dijagram slijeda- Logiranje u sustav

### 2.3.2. Pregled igrača

Ovaj dijagram slijeda koji je prikazan na slici 2.3.2. predstavlja način na koji trener pregledava i unosi igrače u bazu. Prvi trenerov korak prilikom izbora pregleda igrača, nakon što je logiran u sustav je slanje zahtjeva za pregled igrača. Nakon toga sustav inicijalizira modul za pregled, aplikacija pokazuje treneru formu u kojoj trener unosi sve igrače koje je mislio te se tada ti podaci od aplikacije šalju dalje na upis u bazu podataka. Nakon što je baza podataka uspješno obavila upis trenerovih igrača, šalje poruku o uspješnom upisu koju trener vidi. Zadnji



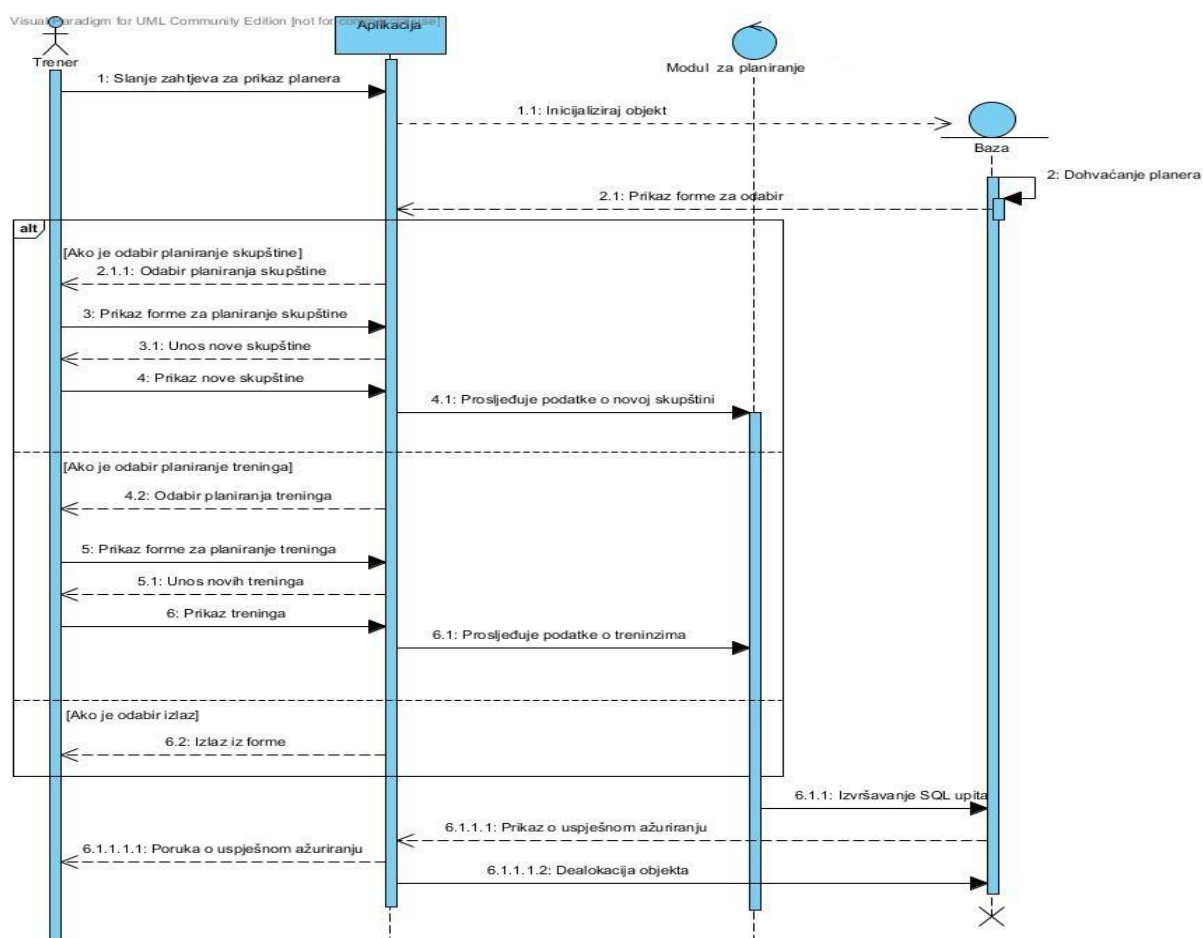
korak je taj da sustav dealocira bazu.

Slika 2.3.2: Dijagram slijeda- Pregled igrača

### 2.3.3. Planer

Ovaj dijagram slijeda predstavlja način na koji trener planira skupštinu ili trening. Prvi trenerov korak prilikom izbora planera, nakon što je logiran u sustav je slanje zahtjeva za prikaz planera. Nakon toga sustav inicijalizira modul za planiranje. Aplikacija prikazuje formu za odabir. Tu trener ima tri mogućnosti: 1. planiranje skupštine, 2. planiranje treninga i 3. izlaz iz forme, tom opcijom se odmah završava cijeli proces.

Ako trener izabere opciju pod brojem jedan dobiva formu za unos planiranje skupštine u koju unosi skupštinu te to šalje aplikaciji koja dalje prosljeđuje podatke o novoj skupštini do modula. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom ažuriranju. Ako je izabrana opcija pod brojem dva, trener dobiva formu za planiranje treninga u koju unosi treninge te to šalje aplikaciji koja dalje prosljeđuje podatke o treninzima do modula. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom ažuriranju. Slika 2.3.3. prikazuje ovaj dijagram slijeda.



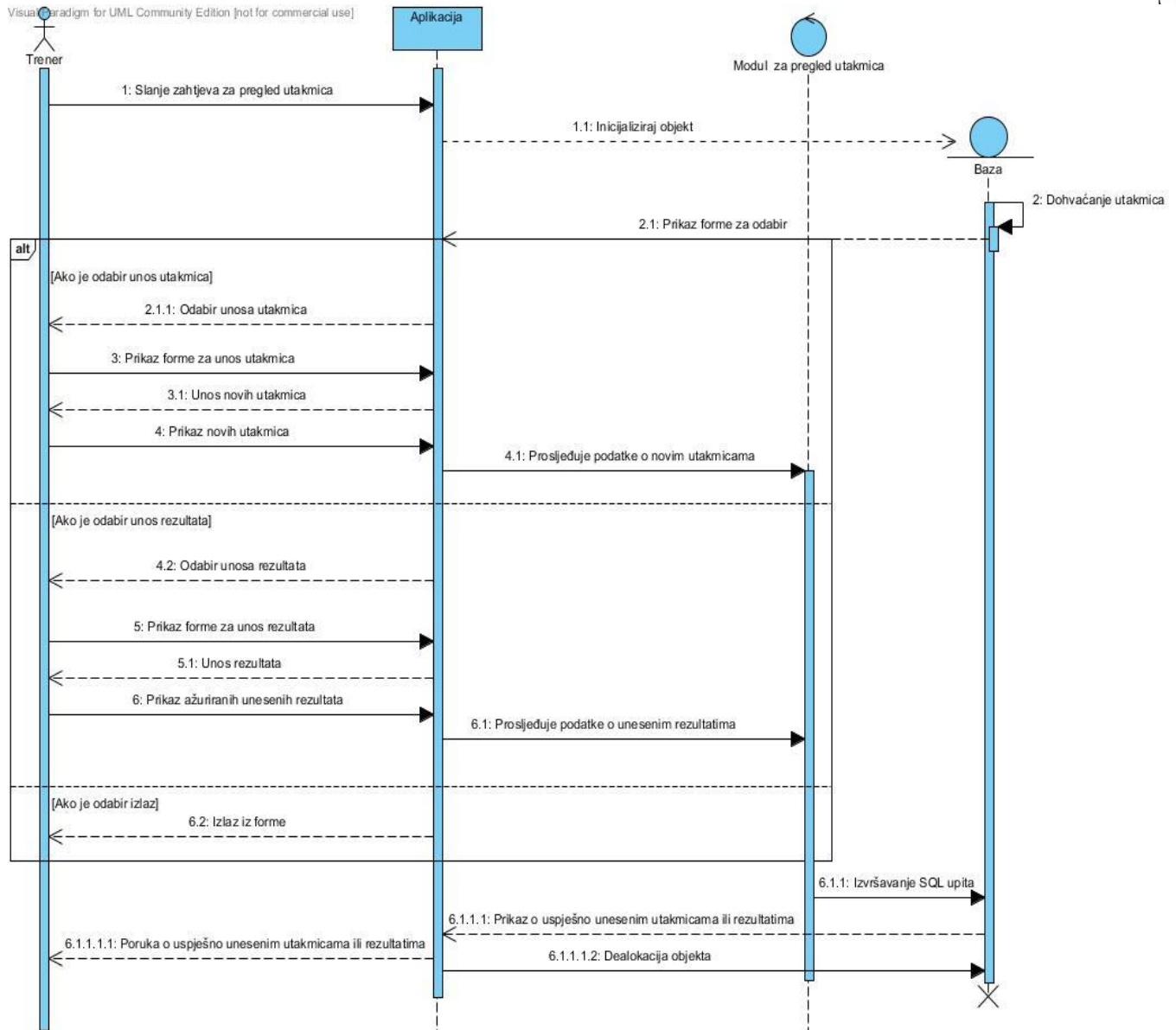
### Slika 2.3.3: Dijagram slijeda- Planer

#### 2.3.4. Pregled utakmica

Ovaj dijagram slijeda prikazan na slici 2.3.4. predstavlja način na koji trener unosi utakmice ili rezultate. Prvi trenerov korak prilikom izbora pregleda utakmica, nakon što je logiran u sustav je slanje zahtjeva za pregled utakmica. Nakon toga sustav inicijalizira modul za pregled utakmica. Aplikacija prikazuje formu za odabir. Tu trener ima tri mogućnosti: 1. unos utakmica , 2. unos rezultata i 3. izlaz iz forme, tom opcijom se odmah završava cijeli proces.

Ako trener izabere opciju pod brojem jedan dobiva formu za unos novih utakmica u koju unosi utakmice te to šalje aplikaciji koja dalje prosljeđuje podatke o novim utakmicama do modula. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom ažuriranju. Ako je izabrana opcija pod brojem dva, trener dobiva formu za unos rezultata u koju unosi rezultate te to šalje aplikaciji koja dalje prosljeđuje podatke o unesenim rezultatima do modula. Dalje se izvršava SQL upit u bazi te dobivamo prikaz o uspješnom unesenim utakmicama ili rezultatima.



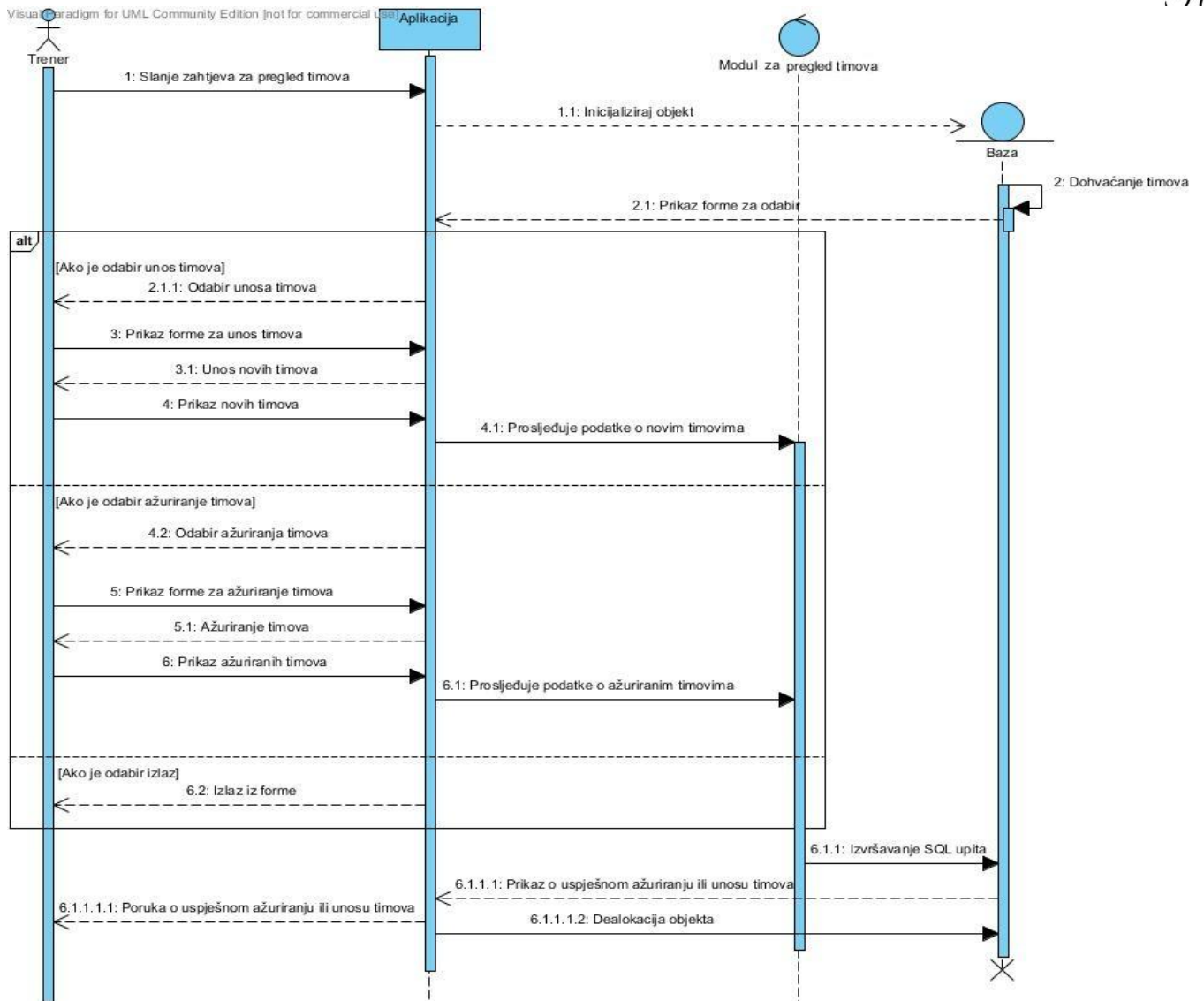


Slika 2.3.4: Dijagram slijeda- Pregled utakmica

### 2.3.5. Pregled timova

Ovaj dijagram slijeda predstavlja način na koji trener unosi timove ili ažurira. Prvi trenerov korak prilikom izbora pregleda timova, nakon što je logiran u sustav je slanje zahtjeva za pregled timova. Nakon toga sustav inicijalizira modul za pregled timova. Aplikacija prikazuje formu za odabir. Tu trener ima tri mogućnosti: 1. unos novih timova , 2. ažuriranje timova i 3. izlaz iz forme, tom opcijom se odmah završava cijeli proces.

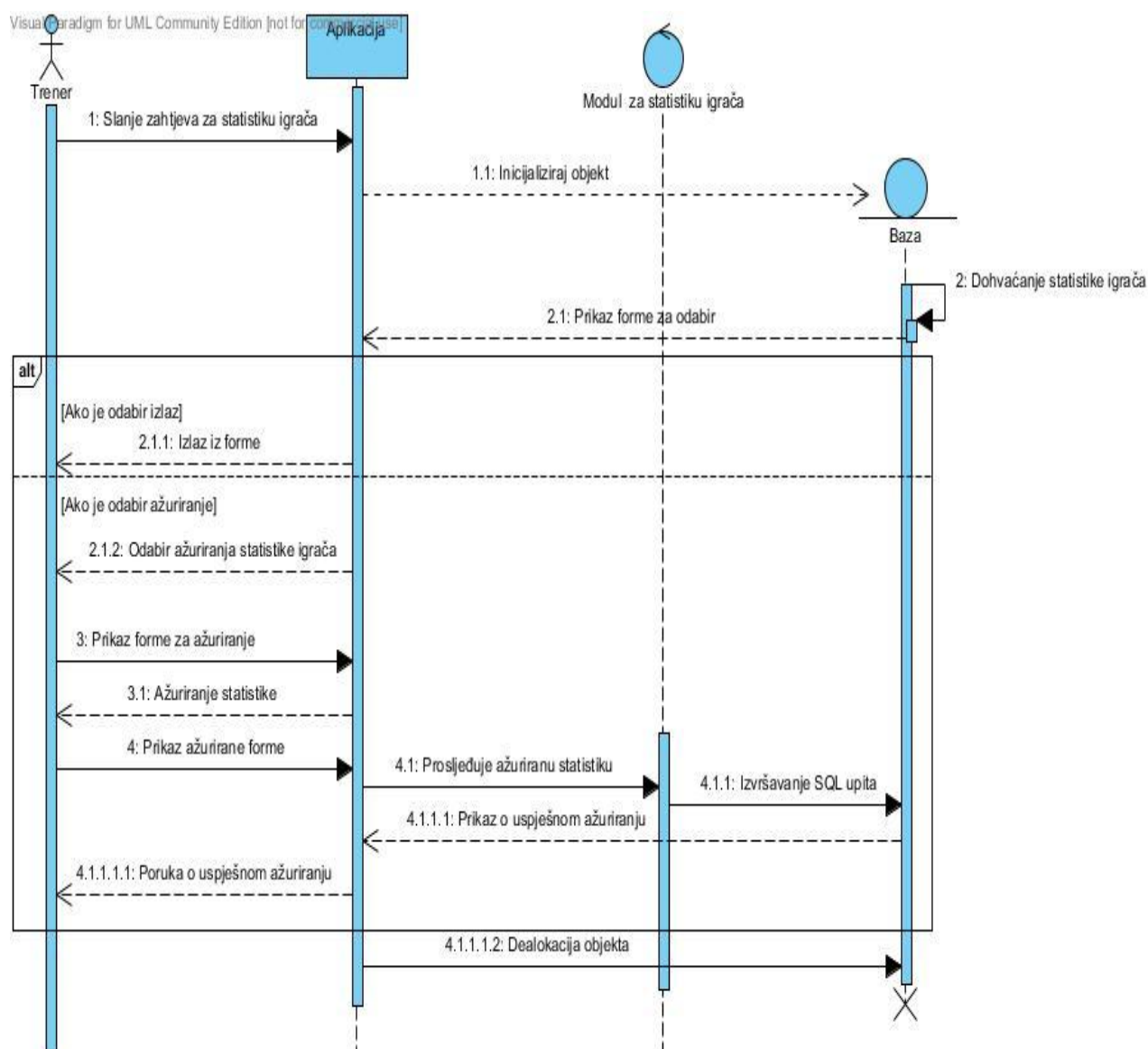
Ako trener izabere opciju pod brojem jedan dobiva formu za unos novih timova u koju unosi timove te to šalje aplikaciji koja dalje prosljeđuje podatke o novim timovima do modula. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom ažuriranju. Ako je izabrana opcija pod brojem dva, trener dobiva formu za ažuriranje timova u koju unosi rezultate te to šalje aplikaciji koja dalje prosljeđuje podatke o ažuriranim timovima do modula. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom unesenim utakmicama ili rezultatima. Na slici 2.3.5. je prikazan ovaj dijagram slijeda.



Slika 2.3.5: Dijagram slijeda- Pregled timova

### 2.3.6. Statistika

Dijagram slijeda predstavlja način na koji trener ažurira statistiku. Prvi trenerov korak prilikom izbora statistike, nakon što je logiran u sustav je slanje zahtjeva za statistiku igrača. Nakon toga sustav inicijalizira modul za pregled timova. Aplikacija prikazuje formu za odabir. Tu trener ima dvije mogućnosti: 1. izlaz iz forme, tom opcijom se odmah završava cijeli proces, 2. ažuriranje, tu dobiva formu za ažuriranje statistike u koju unosi promjene te to šalje aplikaciji koja dalje prosljeđuje podatke s ažuriranom statistikom do modula. Dalje se izvršava željeni SQL upit u bazi te dobivamo prikaz o uspješnom ažuriranju te se izvršava dealokacija. Slika 2.3.6. prikazuje ovaj dijagram slijeda.



Slika 2.3.6: Dijagram slijeda- Statistika

### 3. ERA model

Za potrebnu aplikaciju smo kreirali i ERA model koji je prikazan na slici 3. Na ERA modelu su prikazani entiteti, veze i atributi.

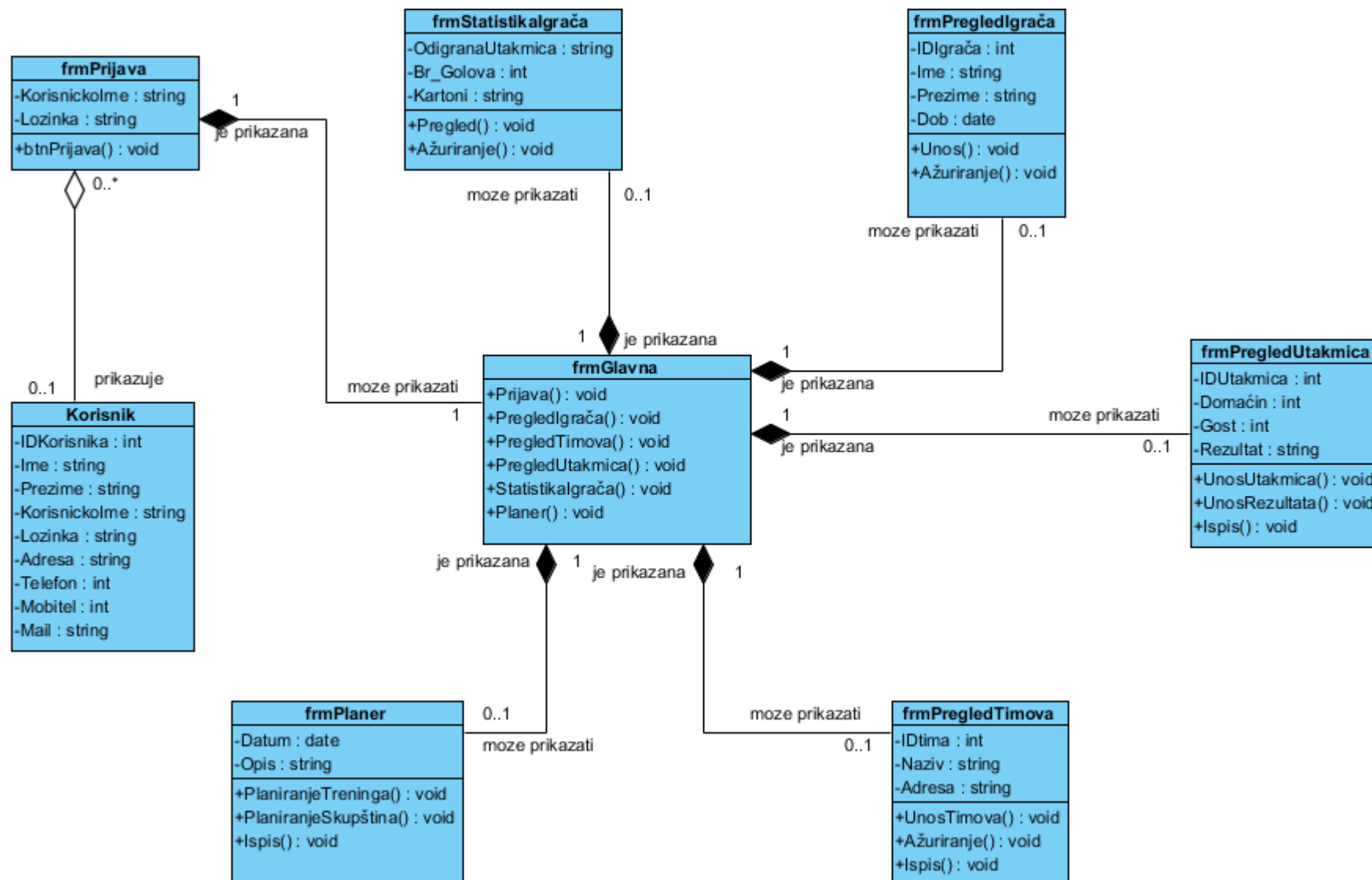
Era model se sastoji od 12 entiteta od kojih su 3 slaba entiteta, a to su: utakmica, događaj i zapis. Svaka ekipa se veže za entitet korisnik koji predstavlja trenera, osim toga ekipa se veže i na entitet utakmica gdje se može vidjeti da na svakoj utakmici može igrati gost i domaćin. Također u jednoj ekipi može igrati više igrača. Na entitet korisnik se veže i entitet događaj koji predstavlja planiranje budućih utakmica i skupština. Na entitet zapis se veže događaj na utakmici koji predstavlja događaje (npr. U kojoj minuti je koji igrač dobio crveni karton).

19

## 4. Dijagram klasa

Dijagram klasa je dijagram u kojem smo prikazali klase, atribute i funkcije koje ćemo koristiti prilikom izrade aplikacije. Sastoji se od 8 klasa. Od klase frmPrijava kreće cijela aplikacija, što znači da se korisnik mora ulogirati kako bi mogao koristiti aplikaciju. Kada se korisnik logira otvara mu se Glavna forma.

Na frmGlavna su kompozicijom povezane frmStatistikaIgrača, frmPregledIgrača, frmPregledUtakmica, frmPregledTimova, frmPlaner, što znači da ako se obriše glavna forma brišu se i svi dijelovi. Ako se obriše neki dio, cjelina (frmGlavna) ostaje. Na frmPrijava je agregacijom povezan Korisnik.



Slika 4. Dijagram klasa