

# Oblikovanje programske potpore

Ak. god. 2015./2016.

## *Virtualna liga - rukomet*

Dokumentacija, Rev. 1

Grupa: *gladijatori*

Voditelj: *Goran Brlas*

Datum predaje: *20. studenog 2015.*

Nastavnik: *Danko Ivošević*

## Popis članova grupe i zaduženja

Goran Brlas – voditelj projekta, koordinacija članova projekta, kontrola kvalitete, upravljanje ljudskim i projektnim resursima, dizajn UML dijagrama

Filip Džidić – dizajn UML dijagrama, arhitektura sustava, baza podataka

Antonio Ivčec – dizajn UML dijagrama, arhitektura sustava

Vinko Kolobara – dizajn UML dijagrama, arhitektura sustava

Vladimir Dragutin Livaja – dizajn UML dijagrama, arhitektura sustava

Lovre Torbarina – dizajn UML dijagrama, arhitektura sustava, baza podataka

Lovro Vidović – dizajn UML dijagrama, arhitektura sustava

## Sadržaj

1. Dnevnik promjena dokumentacije .....	4
2. Opis projektnog zadatka .....	5
3. Pojmovnik .....	7
4. Funkcionalni zahtjevi .....	8
5. Ostali zahtjevi.....	28
6. Arhitektura i dizajn sustava .....	29
6.1.Svrha, opći prioriteti i skica sustava .....	29
6.2.Dijagram razreda s opisom .....	35
6.3.Dijagram objekata .....	39
6.4.Ostali UML dijagrami .....	42
7. Implementacija i korisničko sučelje .....	43
7.1. Dijagram razmještaja.....	43
7.2. Korištene tehnologije i alati .....	43
7.3. Isječak programskog koda vezan za temeljnu funkcionalnost .....	43
7.4. Ispitivanje programskog rješenja .....	43
7.5. Upute za instalaciju .....	43
7.6.Korisničke upute.....	43
8. Zaključak i budući rad .....	44
9. Popis literature .....	45
Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda).....	46
Dodatak B: Dnevnik sastajanja .....	47
Dodatak C: Prikaz aktivnosti grupe.....	49
Dodatak D: Plan rada / Pregled rada i stanje ostvarenja .....	52

## 1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autor(i)	Datum
0.1	Napravljen predložak. Dodan opis projektnog zadatka. Započeo analizu funkcionalnih zahtjeva.	Brlas	26.10.2015.
0.2	Dodan Use Case dijagram.	Brlas	26.10.2015.
0.3	Dodan sekvencijski dijagram.	Vidović	1.11.2015.
0.4	Dodani ostali zahtjevi.	Brlas	10.11.2015.
0.5	Dodan dijagram razreda.	Kolobara	12.11.2015.
0.6	Izmjenjena UC slika, dodan riječnik pojmova.	Brlas	12.11.2015.
0.7	Dodan dijagram ER baze podataka i skica sustava.	Džidić, Torbarina	16.11.2015.
0.8	Dodan dijagram objekata.	Livaja, Ivčec	18.11.2015.
1.0	Dodan zaključak, popis literature te dnevnik sastajanja.	Brlas	18.11.2015.

## 2. Opis projektnog zadatka

Cilj projekta jest razviti web aplikaciju koja će služiti za praćenje rezultata međunarodne rukometne lige, internetski prijenos svih utakmica te nagradne igre za posjetitelje portala, u obliku virtualne lige koja bi koristila igrače iz postojeće baze. Funkcionalnosti koje se želi postići su jednostavna i minimalistička interakcija korisnika sa sustavom koji omogućuje jednostavno pregledavanje rezultata, kao i sudjelovanje u virtualnoj ligi.

Predviđa se izrada portala koji bi se sastojao od web mjesta na kojemu bi se objavljivali rezultati utakmica te bi bila ostvarena programska potpora za prijavu i igranje virtualne lige. Na njemu bi bilo moguće upisati rezultate utakmica zajedno s ostalim, za virtualnu ligu bitnim, događajima. Svi upisani podaci o utakmicama javno bi se objavljivali na portalu udruženja, a dio programskog sustava koji se odnosi na virtualnu ligu to bi pravilno evidentirao kroz odgovarajuće bodovanje izabranih igrača, što će se odraziti na uspješnost prijavljenih natjecatelja virtualne lige.

Osim ručnog upisivanja podataka bit će ostvaren i simulator koji će za prijavljene sastave rukometnih ekipa simulirati odigravanje utakmica, odnosno generirati rezultate i za virtualnu ligu bitne događaje koji će se automatski upisati u sustav te odraziti na bodove natjecatelja.

Kao sustav rukometnog natjecanja pretpostavljen je raspored odigravanja utakmica po sustavu *Bergerovih tablica* ([https://hr.wikipedia.org/wiki/Bergerove\\_tablice](https://hr.wikipedia.org/wiki/Bergerove_tablice)) .

Sve nepredviđene ili krive akcije korisnika također su pokrivene unutar sustava.

U sustavu se, osim prijavljenih natjecatelja, razlikuje još nekoliko vrsta korisnika.

*Posjetitelji portala* koji nisu registrirani natjecatelji mogu samo pratiti rezultate i poredak stvarne rukometne lige.

*Službena osoba* definira proračun natjecatelja, cijene igrača za virtualnu ligu, pravila bodovanja unutar virtualne lige te upisuje događaje utakmica koje se odigravaju.

*Administrator* se brine za učitavanje konfiguracije stvorene od strane službene osobe, samo učitavanje podataka, te provodi simulaciju dijela ili cijelog rukometnog turnira.

Posebna *tehnička komisija* proglašava najboljeg igrača svake utakmice, a njihov ga predstavnik upisuje u sustav.

Natjecatelji virtualne lige registriraju se u sustav navodeći svoje ime, prezime, e-mail adresu, korisničko ime, naziv svoje virtualne ekipe, državu iz koje dolazi te ime rukometne ekipe koju podupire. Nakon toga biraju 7 igrača u svoju virtualnu ekipu unutar dozvoljenog proračuna i po zadanim pozicijama. Nakon svake odigrane utakmice imaju uvid u stanje bodova i bodovanje te prate svoj napredak na ukupnoj ljestvici. Osim ukupne, globalne ljestvice svih natjecatelja, sustav ujedno generira i ljestvice virtualnih ekipa po državama iz koje natjecatelji dolaze i po rukometnim ekipama koje natjecatelji podupiru.

### 3. Pojmovnik

**Apache Derby:** open source relacijska baza podataka izrađena u Javi.

**Apache Tomcat:** open source aplikacijski web poslužitelj za pokretanje i prikaz web stranica.

**CSS (Cascading Style Sheets):** stilski jezik, rabi se za opis prezentacije dokumenta napisanog pomoću markup (HTML) jezika.

**DAO (Data Access Object):** objekt koji pruža apstraktno sučelje za rad s bazom podataka.

**Eclipse:** programska razvojna okolina (IDE) pisana u Javi, a može se koristiti za razvoj aplikacija u raznim programskim jezicima kao što su Java, Ada, C, C++, COBOL, Perl, PHP, Python, R, Ruby, itd. Razvojna okolina (IDE) često se naziva Eclipse ADT (Ada Development Toolkit) za Adu, Eclipse CDT za C/C++, Eclipse JDT za Javu i Eclipse PDT za PHP.

**Git:** distribuirani sustav za upravljanje izvornim kodom.

**Hibernate:** objektno/relacijska tehnologija koja povezuje objekte iz Jave s relacijskim tablicama baze podataka.

**HTML (HyperText Markup Language):** prezentacijski jezik za izradu web stranica.

**Java:** Objektno orijentirani programski jezik koji se može iskoristiti za izradu web aplikacija.

**JSP (JavaServer Pages):** tehnologija koja se koristi za izradu dinamički generiranih web stranica baziranih na HTML, XML ili drugim tipovima dokumenata. Sličan je PHP-u i ASP-u, ali je zasnovan na Javi.

**Servlet:** mali Java program koji se izvršava na web serveru. Obrađuje zahtjeve koje dobiva od web klijenata, najčešće preko HTTP-a.

**SQL:** Programski jezik za manipulaciju bazom podataka.

**UML (Unified/Universal Modeling Language):** standardizirani jezik za slikovno predstavljanje i modeliranje objekata.

## 4. Funkcionalni zahtjevi

### Dionici:

- Posjetitelj
- Natjecatelj
- Službena osoba
- Predstavnik tehničke komisije
- Administrator baze podataka

### Aktori i njihovi funkcionalni zahtjevi:

- Posjetitelj, inicijator
  - Prati rezultate utakmica
  - Prati poredak rukometne lige
  - Može se registrirati i postati natjecatelj
- Natjecatelj, inicijator
  - Registrira se navodeći potrebne podatke
  - Bira 7 igrača u svoju virtualnu ekipu unutar svog proračuna
  - Ostvaruje bodove ovisno o tome kako izabrani igrači igraju
  - Uvid u ukupno stanje bodova, ljestvicu virtualne lige i ostale statističke podatke
- Službena osoba, inicijator
  - Definira proračun i cijene igrača za virtualnu ligu
  - Definira pravila bodovanja unutar virtualne lige
  - Upisuje događaje utakmica koji su relevantni za virtualnu ligu
- Predstavnik tehničke komisije, inicijator
  - Upisuje najboljeg igrača svake utakmice



- Administrator, inicijator
  - Učitava konfiguraciju stvorenu od strane službene osobe
  - Učitava podatke vezane uz ligu
  - Provodi simulaciju utakmica i cijelog turnira
- Baza podataka, sudionik
  - Čuva podatke o svim registriranim korisnicima
  - Čuva podatke o rukometnoj ligi
  - Čuva podatke o virtualnoj ligi

### Opis obrazaca uporabe:

- **UC1** – PogledajStvarnoStanje
  - **Glavni sudionik:** Posjetitelj, Natjecatelj.
  - **Cilj:** Pogledati trenutno stanje u rukometnoj ligi, pojedinačne rezultate utakmica, ostale statističke podatke
  - **Sudionici:** Baza podataka.
  - **Preduvjeti:** Nema.
  - **Rezultat:** Prikaz svih trenutnih stanja u ligi te pojedinačnih utakmica po potrebi.
  - **Željeni scenarij:**
    1. Tablica trenutnog stanja je prikazana prilikom učitavanja stranice.
    2. Ako korisnik želi pogledati pojedinačne rezultate utakmica, treba odabrati tu opciju.
    3. Ako korisnik želi, može pogledati pojedinačni učinak igrača koje je izabrao za virtualnu ligu
    4. Ako korisnik želi, može pregledati sve događaje bitne za virtualnu ligu (strijelce, isključenja i dr.)

5. Sustav vrši pretraživanje baze podataka i prikazuje rezultate utakmica u kronološkom redu, počevši od najnovijih rezultata.

6. Sustav vrši pretraživanje i prikazuje statističke podatke

• **UC2 – RegistrirajNatjecatelja**

- **Glavni sudionik:** Posjetitelj.
- **Cilj:** Registrirati novog natjecatelja u bazu podataka.
- **Sudionici:** Baza podataka.
- **Preduvjeti:** Nema.
- **Rezultat:** Registracija novog korisnika i slanje elektroničke pošte s potvrdom korisniku.
- **Željeni scenarij:**
  1. Korisnik na vrhu stranice odabire poveznicu za prijavu novog korisnika te se otvara stranica za prijavu.
  2. Korisnik unosi sve potrebne podatke te odabire tipku za potvrdu unosa.
  3. Sustav provjerava ispravnost podataka.
  4. Sustav prihvća podatke i u bazu podataka upisuje podatke o novom natjecatelju.
  5. Sustav šalje e-mail korisniku sadržaja o uspješnoj registraciji, kao i potvrdu o odabranim podacima.
- **Mogući drugi scenarij:**
  - Baza podataka ne može upisati novog natjecatelja, zato što potrebni podaci nisu dobro uneseni ili je unesena već postojeća e-mail adresa, naziv virtualne ekipe ili korisničko ime.
    1. Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za prijavu sa obavijesti o kojem tipu greške se radi.
    2. Korisnik modificira potrebne podatke te odabire tipku za završetak unosa ili odustaje od registracije.

**• UC3 – SastaviEkipu**

- **Glavni sudionik:** Natjecatelj.
- **Cilj:** Sastaviti virtualnu ekipu koja se sastoji od 7 igrača.
- **Sudionici:** Baza podataka.
- **Preduvjeti:** Nema. Korisnik već nema izabranu ekipu.
- **Rezultat:** Natjecatelj sastavlja vlastitu ekipu s kojom se onda natječe u virtualnoj ligi.
- **Željeni scenarij:**
  1. Korisnik po poziciji iz padajućeg izbornika odabire igrača ili pretražuje igrača po imenu ili prezimenu, vodeći računa o svom proračunu.
  2. Nakon što su svi igrači odabrani, sustav pohranjuje korisnikovu ekipu u bazu podataka.
- **Mogući drugi scenarij:**
  3. Ukoliko je korisnik prekoračio svoj proračun, sustav ga obavještava te ne prihvaća ekipu.
  4. Ukoliko je korisnik izabrao istog igrača na više pozicija, sustav ga obavještava te ne prihvaća ekipu.
  5. Ukoliko je korisnik izabrao više igrača za istu poziciju, sustav ga obavještava te ne prihvaća ekipu.

**• UC4 – PogledajStanjeVirtualneLige**

- **Glavni sudionik:** Natjecatelj.
- **Cilj:** Dobiti uvid u trenutno stanje virtualne lige.
- **Sudionici:** Baza podataka.
- **Preduvjeti:** Nema.
- **Rezultat:** Natjecatelj dobiva uvid u trenutno stanje virtualne lige.

- **Željeni scenarij:**

1. Natjecatelj u svakom trenutku ima pogled na ukupnu virtualnu ljestvicu.
2. Korisnik može odabrati drukčiji pogled na tablicu:
  - a. Može odabrati koliko bodova je od početka lige do sad osvojio pojedini igrač u vlastitoj ekipi.
  - b. Može odabrati poredak svih natjecatelja koji dolaze iz iste države.
  - c. Može odabrati poredak svih natjecatelja koji podržavaju istu ekipu.
3. Sustav korisniku prikazuje željeni odabir.

- **UC5 – DefinirajPočetnuKonfiguraciju**

- **Glavni sudionik:** Službena osoba.
- **Cilj:** Definirati proračun natjecatelja i cijene igrača za virtualnu ligu te pravila bodovanja unutar virtualne lige.
- **Sudionici:** Administrator.
- **Preduvjeti:** Liga još nije počela.
- **Rezultat:** Definiran proračun, cijene te pravila bodovanja unutar virtualne lige.
- **Željeni scenarij:**
  1. Službena osoba definira proračun, cijene te pravila bodovanja unutar virtualne lige.
  2. Definirani podaci izvode se u odgovarajućem formatu.
  3. Definirani podaci učitavaju se kao početna konfiguracija od strane administratora.

- **UC6 – UpišiDogađaje**

- **Glavni sudionik:** Službena osoba.
- **Cilj:** Upisati događaje utakmica koje se odigravaju.

- **Sudionici:** Baza podataka.
- **Preduvjeti:** Nema.
- **Rezultat:** Upisani događaji utakmica koje se odigravaju.
- **Željeni scenarij:**
  1. Službena osoba odabire utakmicu za koju želi upisati događaje.
  2. U bazu podataka se upisuju događaji relevantni za virtulanu ligu za odabranu utakmicu te se automatski osvježava stanje u tablici virtualne lige.

- **UC7 – ProglasiMVP**

- **Glavni sudionik:** Predstavnik tehničke komisije.
- **Cilj:** Upisati igrača utakmice za odabranu utakmicu.
- **Sudionici:** Baza podataka.
- **Preduvjeti:** Nema.
- **Rezultat:** Upisan igrač utakmice za odabranu utakmicu u bazu podataka.
- **Željeni scenarij:**
  1. Predstavnik tehničke komisije odabire utakmicu za koju želi proglasiti glavnog igrača ukoliko on nije automatski proglašen od strane simulatora.
  2. U bazu podataka se upisuje igrač utakmice te se automatski osvježava stanje na tablici virtualne lige.

- **UC8 – AdministrirajBazuPodataka**

- **Glavni sudionik:** Administrator.
- **Cilj:** Učitati podatke ili početnu konfiguraciju u bazu podataka.
- **Sudionici:** Baza podataka.

- **Preduvjet:** Ako se želi učitati početna konfiguracija, ona prvo mora biti definirana od strane službene osobe.
- **Rezultat:** Učitani podatci ili početna konfiguracija.
- **Željeni scenarij:**
  1. Administrator iz izbornika odabire želi li učitati početnu konfiguraciju ili podatke vezane uz rukometnu ligu ili natjecatelje.
  2. Željeni podatci ili konfiguracija učitavaju se u bazu podataka.

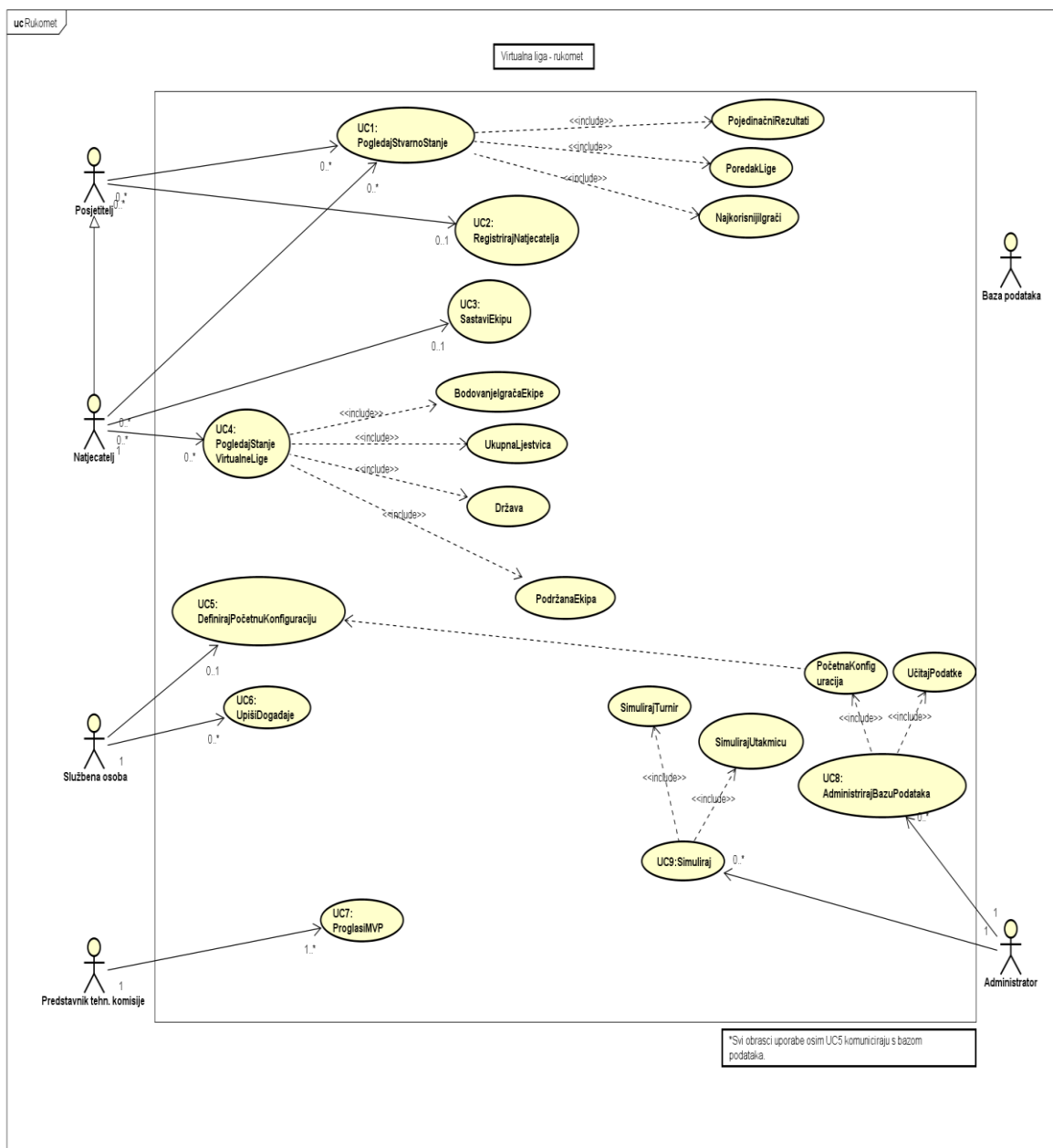
- **UC9 – Simuliraj**

- **Glavni sudionik:** Administrator.
- **Cilj:** Simulirati jednu utakmicu ili cijeli turnir.
- **Sudionici:** Baza podataka.
- **Preduvjet:** Nema.
- **Rezultat:** Simulirana jedna utakmica ili cijeli turnir.
- **Željeni scenarij:**
  1. Administrator iz izbornika odabire želi li simulirati jednu utakmicu ili cijeli turnir.
  2. Ako simulira jednu utakmicu, onda odabire točnu utakmicu iz izbornika te pokreće simulator. Ako se simulira cijeli turnir, odabere se ta opcija i pokreće simulator.
  3. Rezultat simulacije se unosi u bazu podataka i automatski se osvježava stanje na tablici rukometnih ekipa, kao i stanje tablice virtualne lige.

- **UC10 – UlogirajSe**

- **Glavni sudionik:** Natjecatelj, Službena osoba, Predstavnik tehn. komisije, Administrator.
- **Cilj:** Ulogirati se u sustav.
- **Sudionici:** Baza podataka.

- **Preduvjet:** Nema.
- **Rezultat:** Korisnik ulogiran u sustav te su mu dodijeljena prava sukladna sa njegovim ovlastima.
- **Željeni scenarij:**
  1. Korisnik na vrhu stranice odabire poveznicu za prijavu novog korisnika te se otvara stranica za prijavu.
  2. Korisnik unosi potrebne podatke.
  3. Korisnik se ulogira u sustav.
- **Mogući drugi scenarij:**
  1. Ukoliko je korisnik unio pogrešne podatke, sustav mu to javlja odgovarajućom porukom.



Slika 4.1. Obrazac uporabe za UC1 do UC10.



## Sekvencijski dijagrami

### Obrazac uporabe UC1

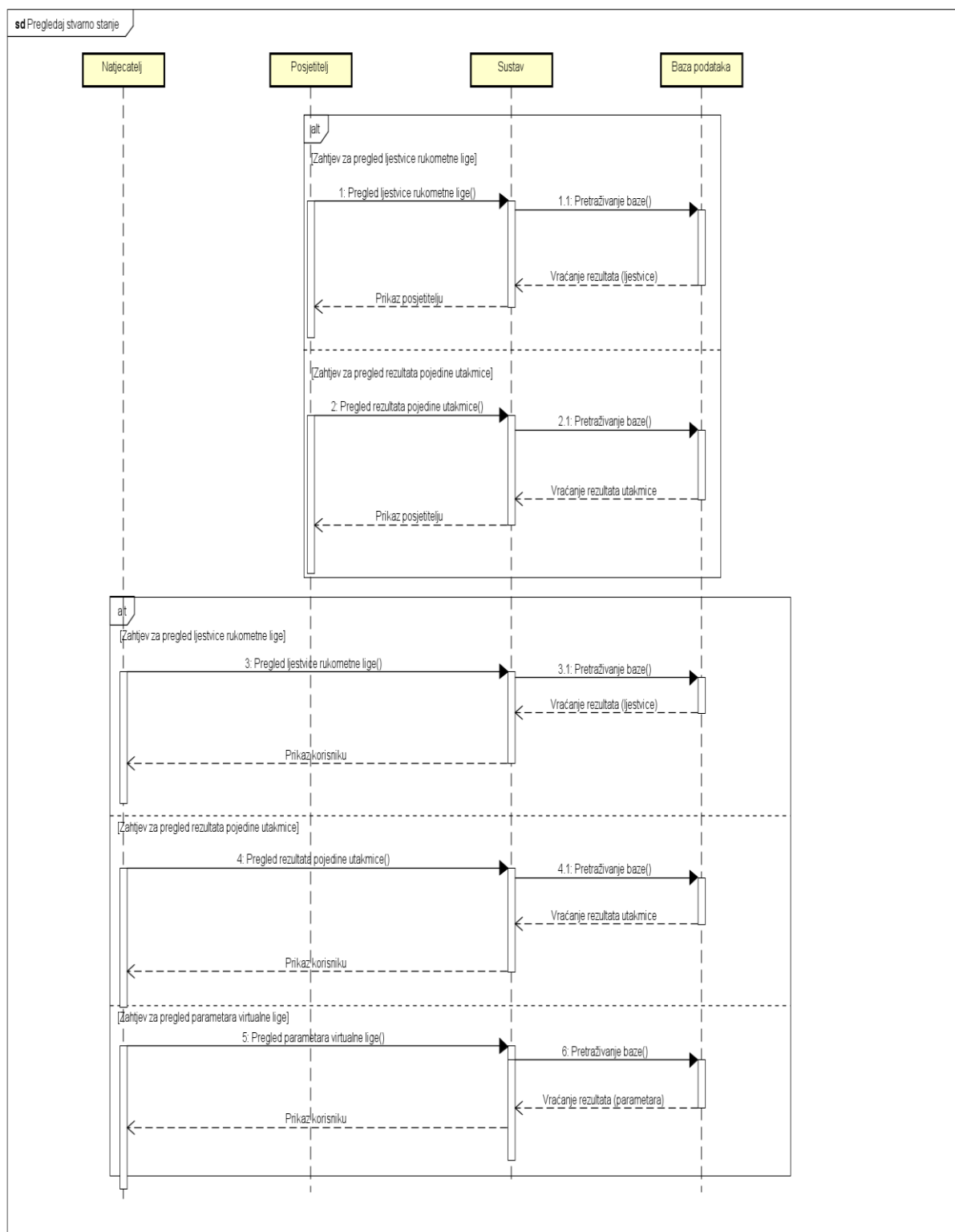
Pregled stvarnog stanja rukometne lige može biti izvršen od strane korisnika virtualne lige (registrirane osobe koja je ujedino i natjecatelj) te od strane posjetitelja portala (neregistrirane osobe).

Korisnik, uz praćenje rezultata stvarne rukometne lige, ima mogućnost praćenja parametara potrebnih za ravnopravno natjecanje u virtualnoj ligi: stanja na ljestvici virtualne lige te učinka igrača koje je izabrao za sudjelovanje u virtualnoj ligi.

Korisnik (natjecatelj) i posjetitelj imaju mogućnost gledanja rezultata svake utakmice odabirom potrebne opcije koja će omogućiti prikaz rezultata.

Korisnik učitava stranicu te mu se nudi prikaz tablice rukometne lige, rezultata utakmice, statističkih podataka, ljestvice stvarne rukometne lige te tablice najkorisnijih igrača.

Prikaz svih potrebnih parametara za sudjelovanje u virtualnoj ligi vrši se uz pomoć sustava i baze podataka. Sustav vrši pretraživanje baze podataka i prikazuje rezultate utakmica u kronološkom redu, počevši od najnovijih rezultata. Na isti se način prikazuju i statistički podatci vezani za pojedinu utakmicu.



powered by Astah

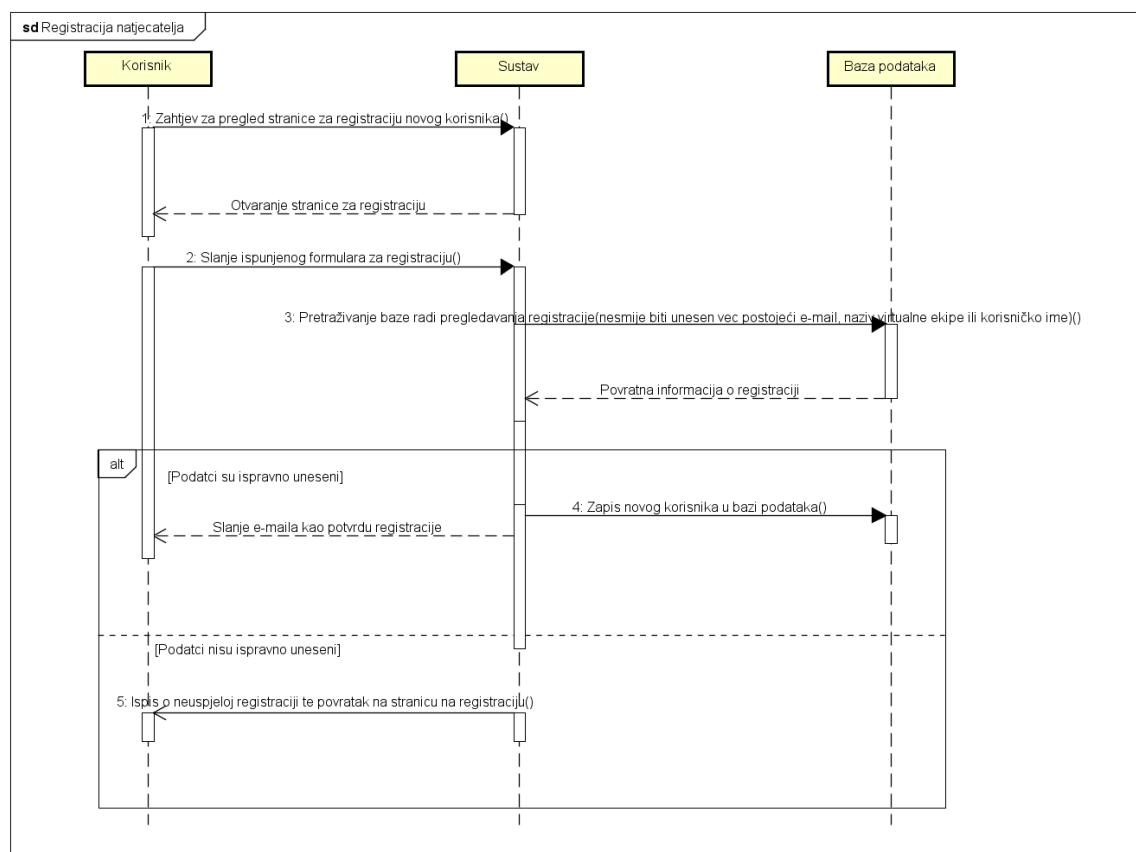
Slika 4.2. Sekvencijski dijagram za UC1 (PogledajStvarnoStanje)

### **Obrazac uporabe UC2**

Korisnik na vrhu stranice odabire poveznicu za prijavu novog korisnika koja mu nudi mogućnost registriranja. Stranica sadrži formular za registraciju. Za samu registraciju nužno je upisati: ime, prezime, e-mail adresu, korisničko ime, šifru, naziv svoje virtualne ekipe, državu iz koje dolazi te ime rukometne ekipe koju podupire.

Korisnik unosi potrebne podatke te potvrđuje unos podataka. Sustav provjerava ispravnost unesenih podataka te, ako su svi podatci dobro uneseni, šalje e-mail korisniku kojim se potvrđuje uspješna registracija te se korisnik upisuje u bazu podataka.

Ako uneseni podatci nisu ispravno uneseni ili je unesena već postojeća e-mail adresa, naziv virtualne ekipe ili korisničko ime, sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za prijavu sa obavijesti o kojem tipu greške se radi.



powered by Astah

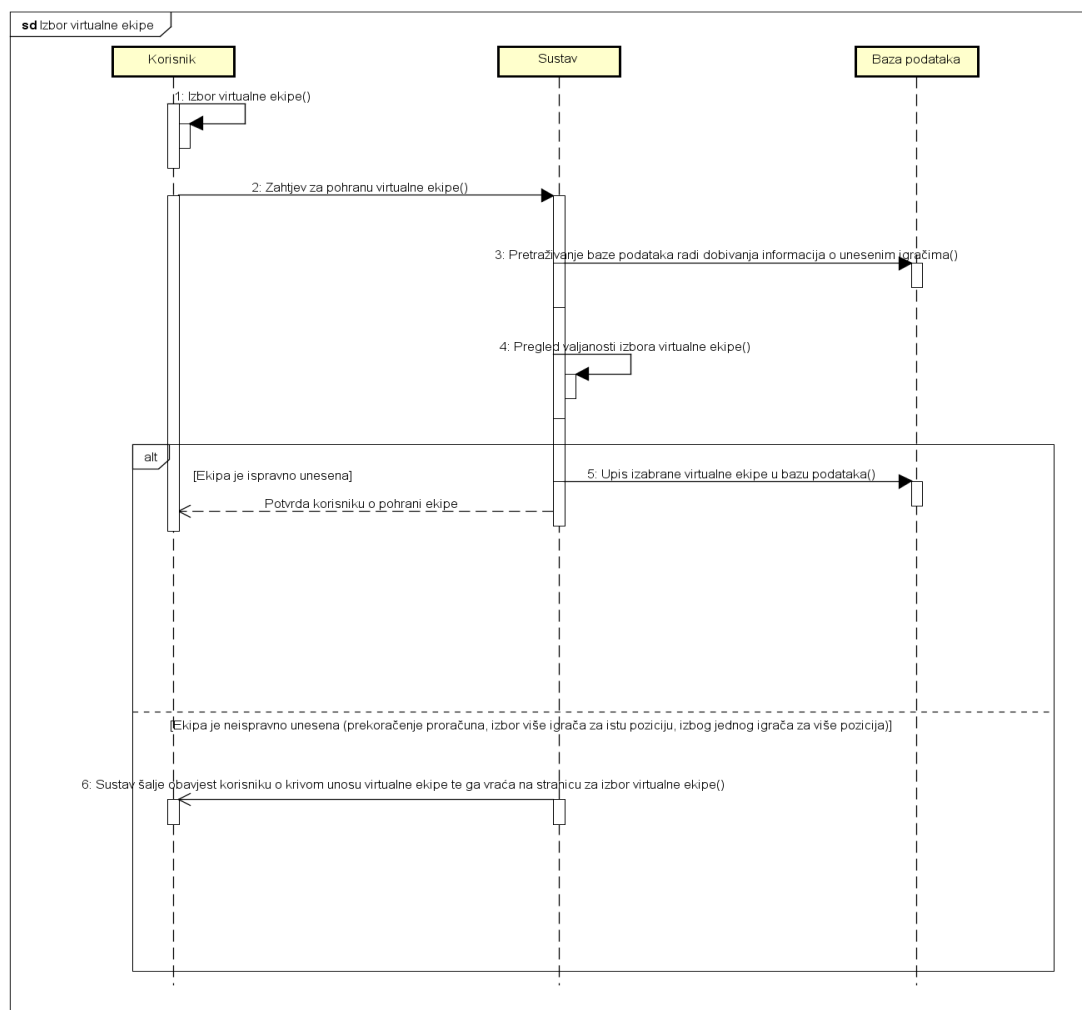
**Slika 4.3. Sekvencijski dijagram za UC2 (RegistrirajNatjecatelja)**

**Obrazac uporabe UC3**

Nakon uspješne registracije korisnik je proslijeđen na stranicu koja nudi mogućnost izbora vlastite ekipe s kojom se natječe u virtualnoj ligi. Korisnik odabire igrače pretražujući po imenu i prezimenu željenog igrača.

Nakon što su svi korisnikovi igrači odabrani, sustav čita podatke iz baze podataka za odabranu ekipu. U slučaju da je korisnik prilikom izbora virtualne ekipe prekoračio proračun, izabrao istog igrača na više pozicija ili više igrača za istu poziciju, sustav ga obavještava o grešci te ga vraća na stranicu koja nudi mogućnost izbora ekipe za sudjelovanje u virtualnoj ligi.

Ukoliko je korisnik izabrao vlastitu ekipu na pravilan način, ekipa se zapisuje u bazu podataka.



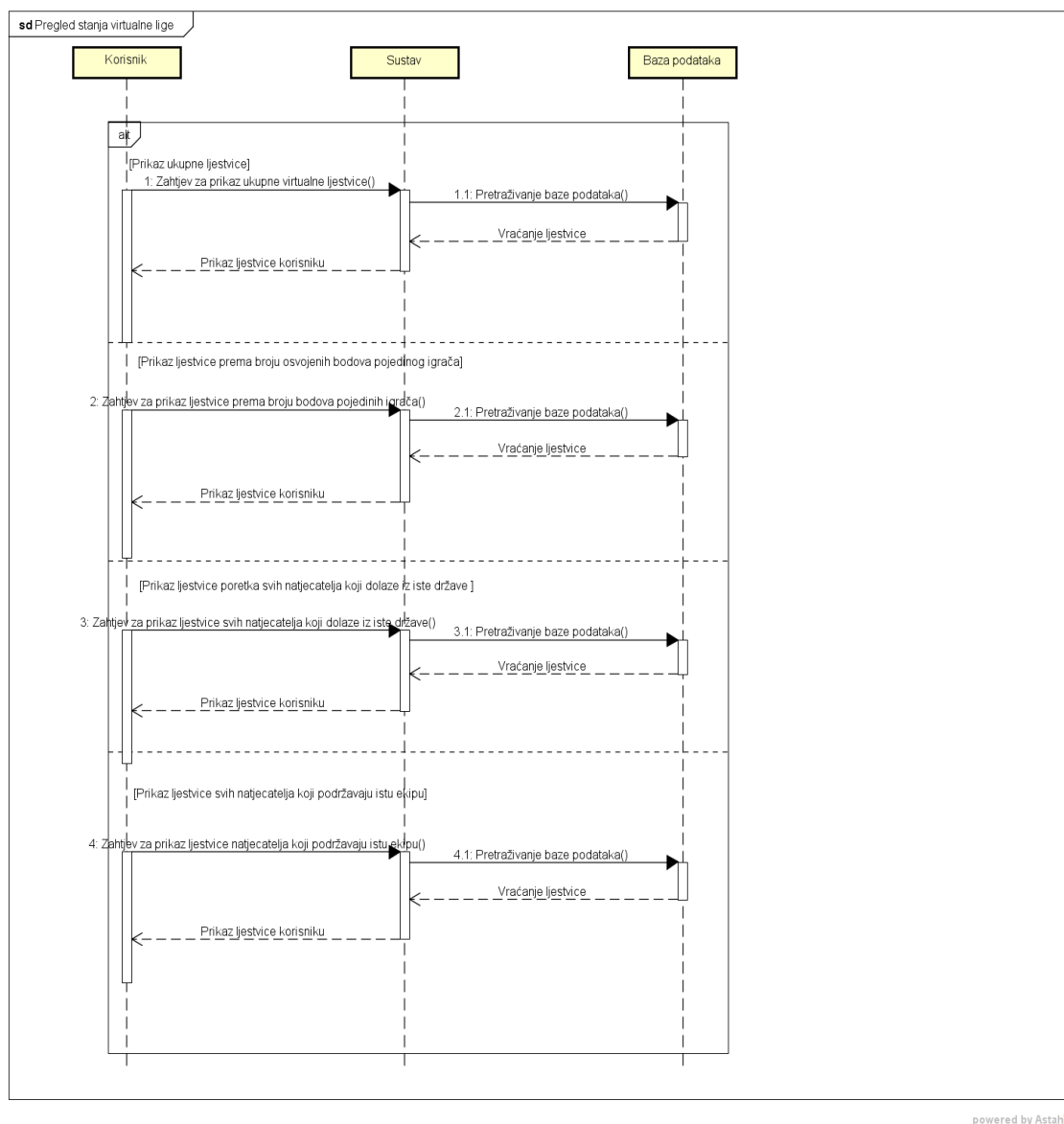
powered by Astah

**Slika 4.4. Sekvencijski dijagram za UC3 (SastaviEkipu)**

### **Obrazac uporabe UC4**

Sustav nudi mogućnost prikaza stanja na ukupnoj ljestvici svakom natjecatelju u bilo kojem trenutku (sustav šalje upit bazi podataka koja vraća ukupnu ljestvicu). Također, korisnik može izabrati i drugačiji pogled na tablicu. Može odabrati pogled stanja bodova igrača u vlastitoj ekipi, poredak svih natjecatelja koji dolaze iz iste države te poredak svih natjecatelja koji podržavaju istu ekipu.

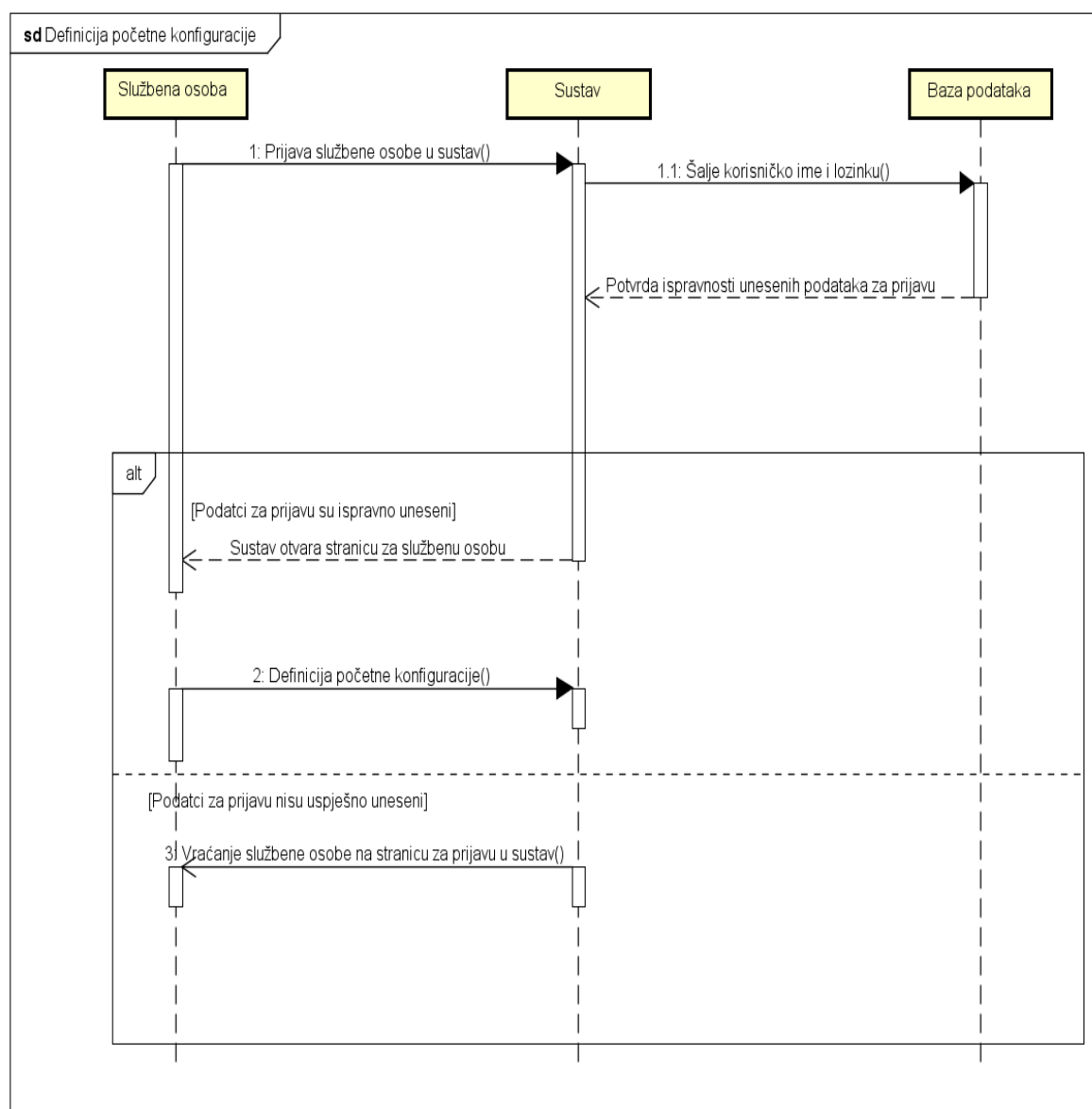
Nakon odabira željenog prikaza tablice sustav šalje upit bazi podataka za prikaz specifičnog oblika tablice korisniku te se korisniku prikaže tražena tablica.



**Slika 4.5. Sekvencijski dijagram za UC4 (PogledajStanjeVirtualneLige)**

**Obrazac uporabe UC5**

Službena osoba prijavljuje se na sustav i odlazi na posebnu stranicu za definiranje parametara važnih za sudjelovanje u virtualnoj ligi. Definira proračun svakog natjecatelja, cijene igrača te pravila bodovanja unutar virtualne lige popunjavanjem odgovarajućih polja na stranici. Stvara se poveznica za izvoz konfiguracije virtualne lige u određenom formatu. Ukoliko je krivo unesena lozinka ili korisničko ime službena osoba se vraća na početnu stranicu za prijavu u sustav.

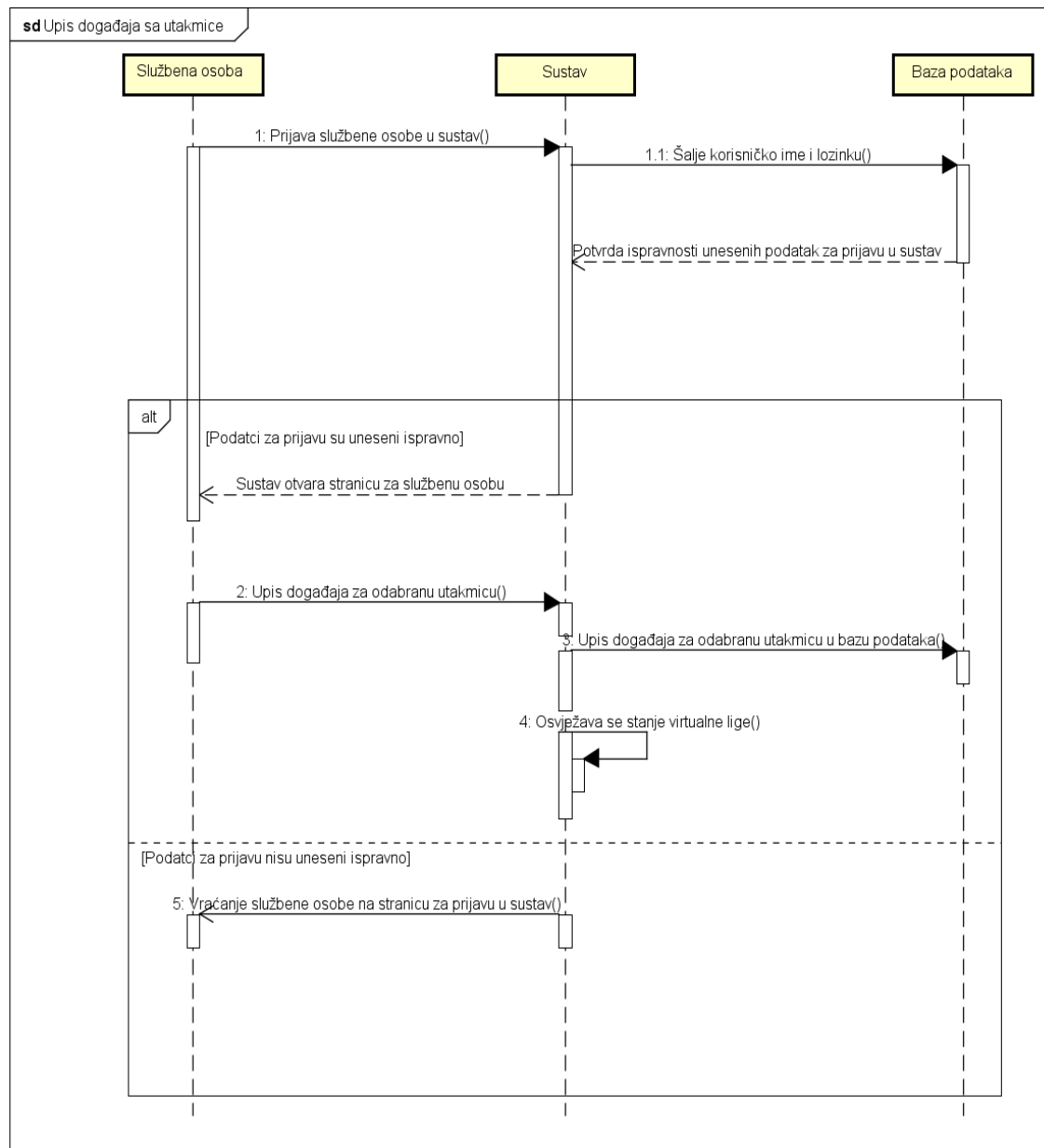


powered by Astah

**Slika 4.6. Sekvencijski dijagram za UC5 (Definiraj Početnu Konfiguraciju)**

**Obrazac oprabe UC6**

Službena osoba prijavljuje se u sustav. U slučaju pogrešnog unosa lozinke ili korisničkog imena sustav vraća službenu osobu na stranicu za prijavu u sustav. Nakon uspješne prijave u sustav službena osoba bira utakmicu za koju želi upisati događaje. U bazu podataka se upisuju događaji relevantni za virtulanu ligu za odabranu utakmicu te se automatski osvježava stanje u tablici virtualne lige.



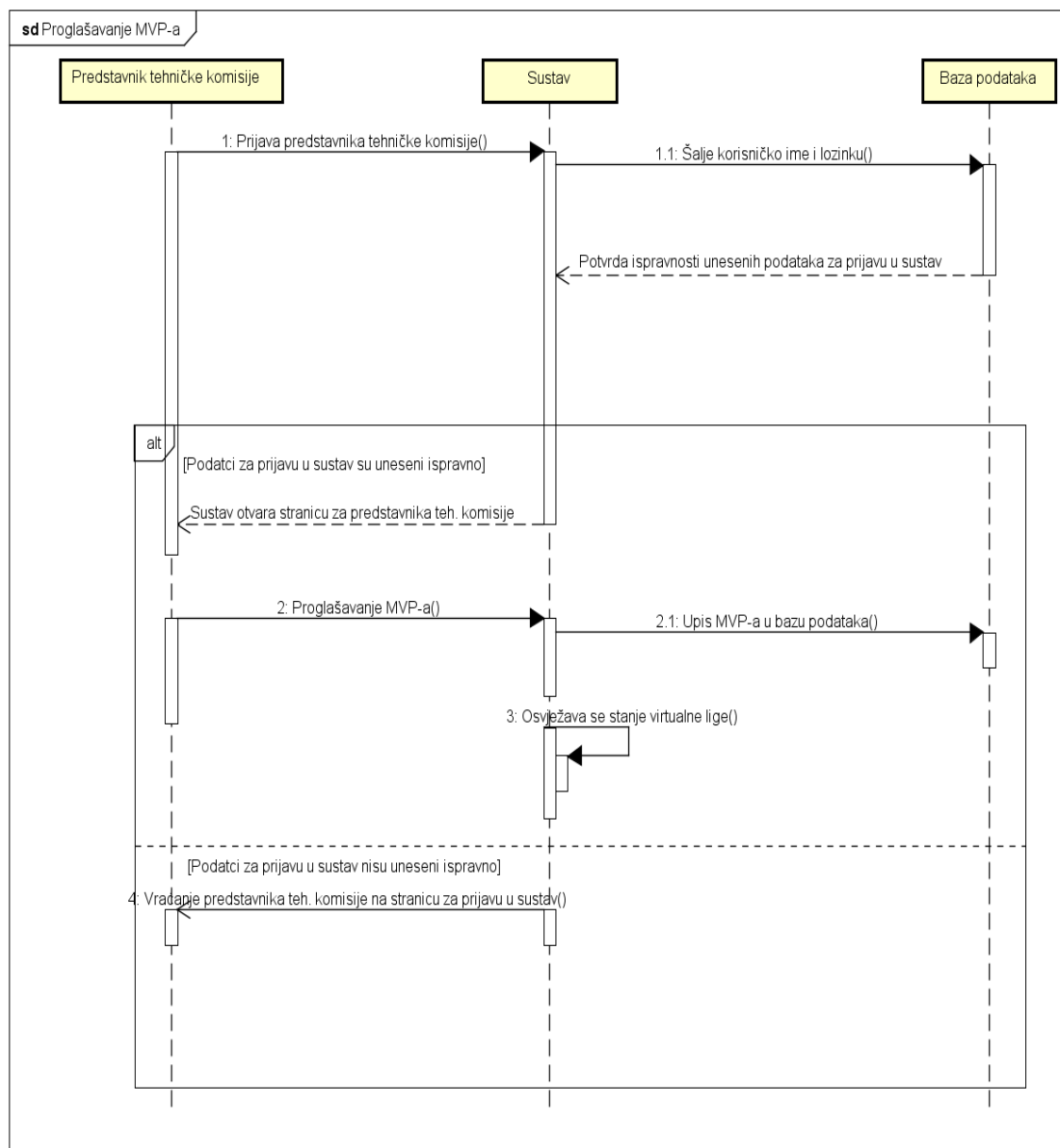
powered by Astah

**Slika 4.7. Sekvencijski dijagram za UC6 (UpisDogađaje)**

**Obrazac uporabe UC7**

U sustav se prijavljuje predstavnik tehničke komisije. U slučaju krivog unosa korisničkog imena ili lozinke korisnik se prosljeđuje na stranicu za prijavu u sustav.

Predstavnik tehničke komisije odabire utakmicu za koju želi proglasiti glavnog igrača ukoliko on nije automatski proglašen od strane simulatora. U bazu podataka upisuje se igrač utakmice te se automatski osvježava stanje na tablici virtualne lige.



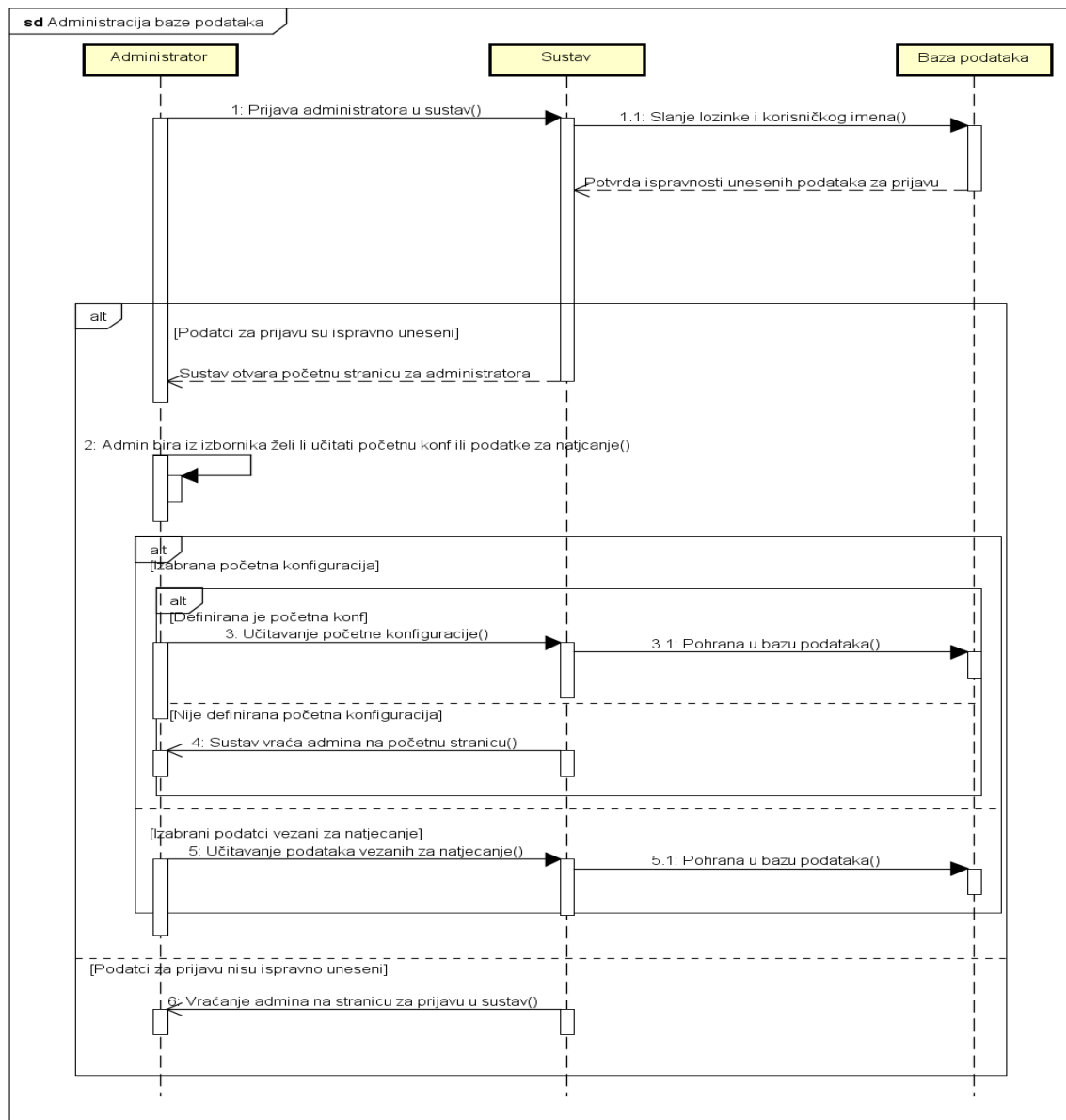
powered by Astah

**Slika 4.8. Sekvencijski dijagram za UC7 (Proglasavanje MVP)**



**Obrazac uporabe UC8**

Administrator se prijavljuje u sustav. U slučaju krivog unosa korisničkog imena ili lozinke korisnik se prosljeđuje na stranicu za prijavu u sustav. Nakon uspješne prijave u sustav administrator iz izbornika odabire želi li učitati početnu konfiguraciju ili podatke vezane uz rukometnu ligu ili natjecatelje. Željeni podatci ili konfiguracija učitavaju se u bazu podataka.

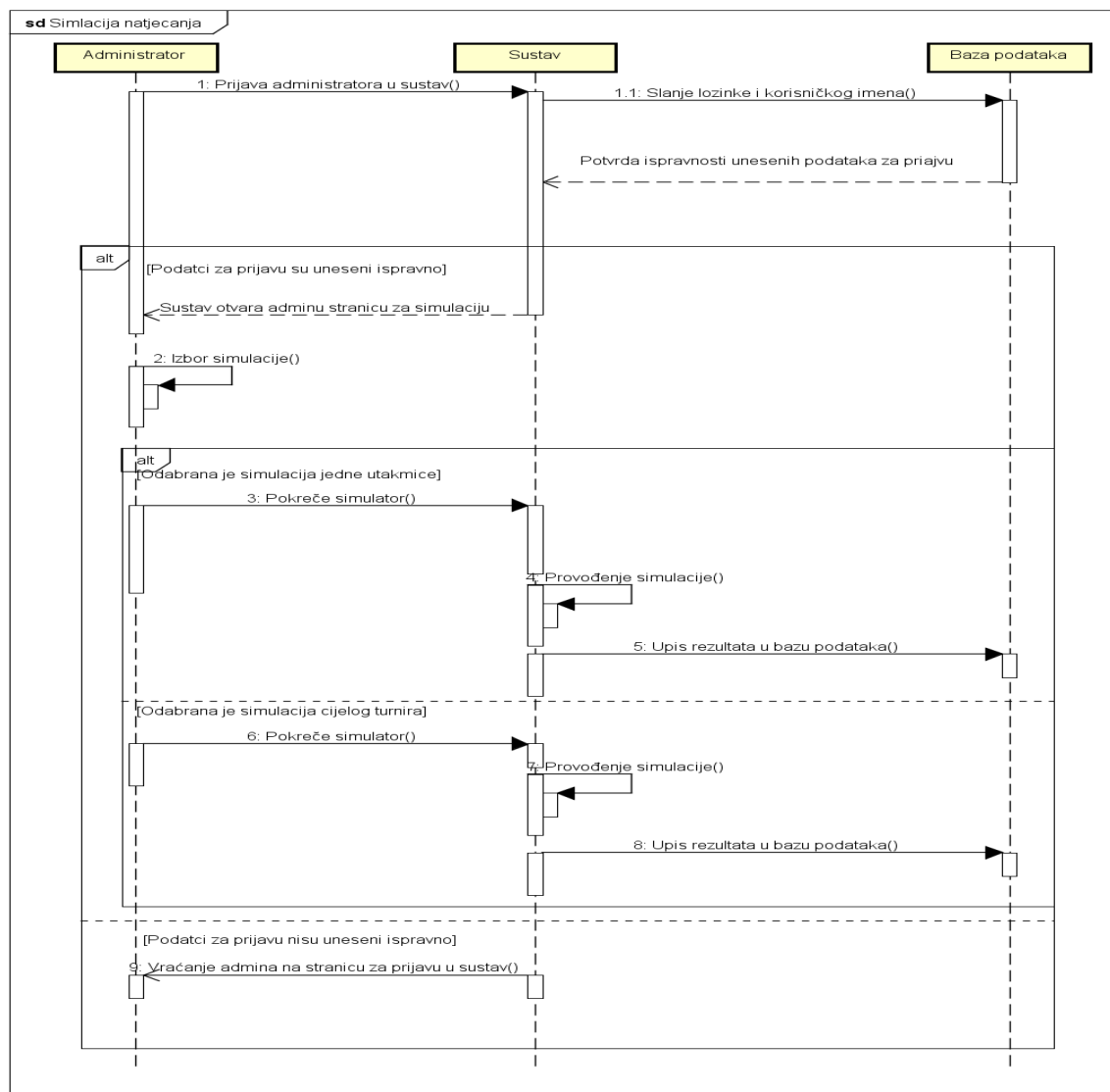


powered by Astah

**Slika 4.9. Sekvencijski dijagram za UC8 (AdministrirajBazuPodataka)**

**Obrazac uporabe UC9**

Administrator se prijavljuje u sustav. U slučaju krivog unosa korisničkog imena ili lozinke prosljeđuje se na stranicu za prijavu u sustav. Nakon uspješne prijave u sustav administrator iz izbornika odabire želi li simulirati jednu utakmicu ili cijeli turnir. Ako simulira jednu utakmicu, onda odabire točnu utakmicu iz izbornika te pokreće simulator. Ako se simulira cijeli turnir, odabere se ta opcija i pokreće simulator. Rezultat simulacije se unosi u bazu podataka i automatski se osvježava stanje na tablici rukometnih ekipa, kao i stanje tablice virtualne lige.



powered by Astah

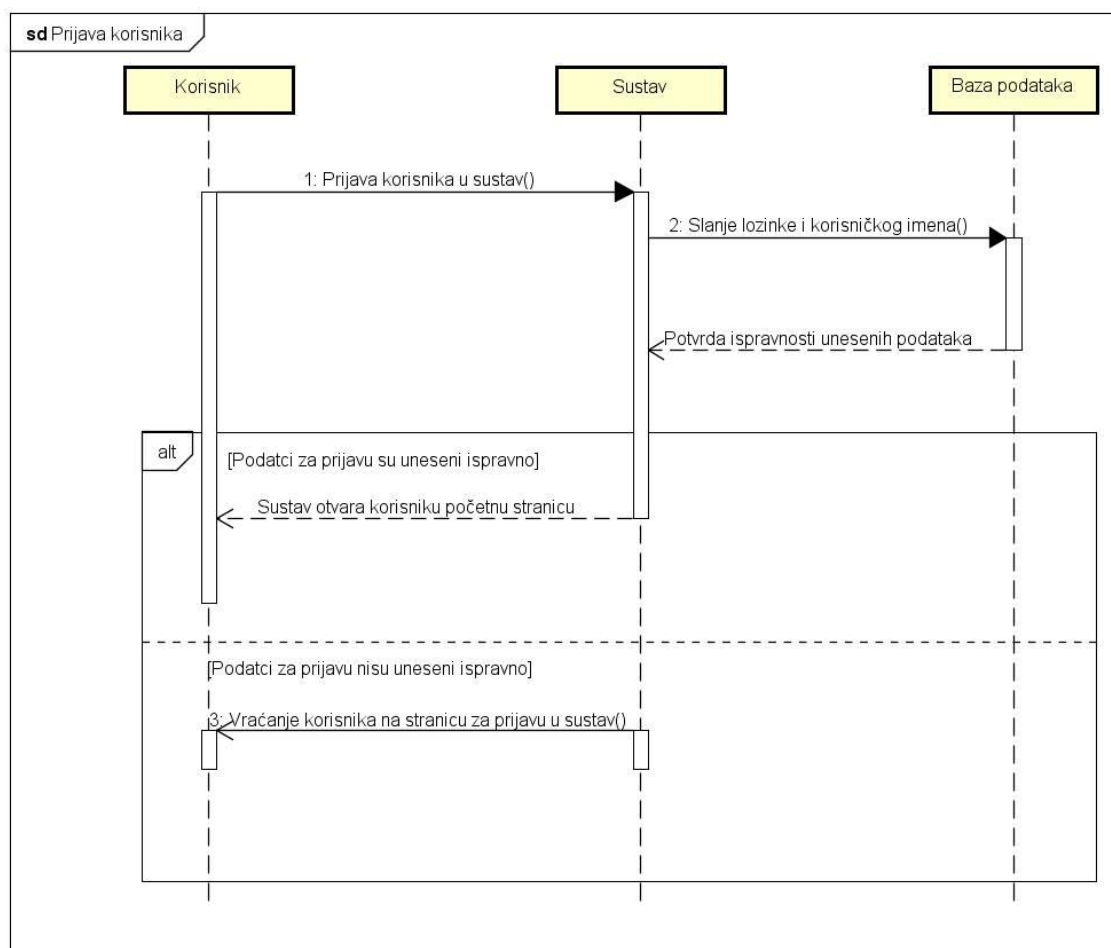
**Slika 4.10. Sekvencijski dijagram za UC9 (Simuliraj)**

**Obrazac uporabe UC10:**

Korisnik (natjecatelj, službena osoba, predstavnik tehničke komisije, administrator) na vrhu stranice odabire poveznicu za prijavu novog korisnika te se otvara stranica za prijavu. Korisnik unosi potrebne podatke za prijavu u sustav (korisničko ime i lozinku) te stisne gumb za prijavu u sustav.

Ukoliko je dobro unesena lozinka i korisničko ime korisnik je uspješno prijavljen u sustav i sustav otvara početnu stranicu.

Ukoliko je korisnik prilikom prijave napisao pogrešno korisničko ime ili lozinku sustav mu javlja informaciju o grešci te ga usmjerava na stranicu za prijavu u sustav.



powered by Astah

**Slika 4.11. Sekvencijski dijagram za UC10 (UlogirajSe)**

## 5. Ostali zahtjevi

- Sustav mora podržavati paralelni rad više korisnika,
- Korisničko sučelje i sustav trebaju podržavati hrvatske znakove,
- Sustav mora biti realiziran tako da omogućuje jednostavno korištenje bez potrebe za detaljnim uputama,
- Pretraga baze podataka koju inicira korisnik ne smije trajati duže od 10 sekundi,
- Korisničko sučelje mora omogućiti više različitih razina dostupnosti ovisno o dodijeljenim korisničkim ovlastima,
- Opisana aplikacija mora biti prilagođena izvedbi na Webu,
- Nepravilno i nepredviđeno ponašanje korisnika ne smije narušiti funkcionalnost i rad sustava,
- Nadogradnja ili uvođenje potpuno novih funkcionalnosti ne smije narušiti postojeće funkcionalnosti kao ni sustav u cjelini,
- Cjelokupni sustav i priložena dokumentacija moraju biti izvedeni do siječnja 2016. godine.

## 6. Arhitektura i dizajn sustava

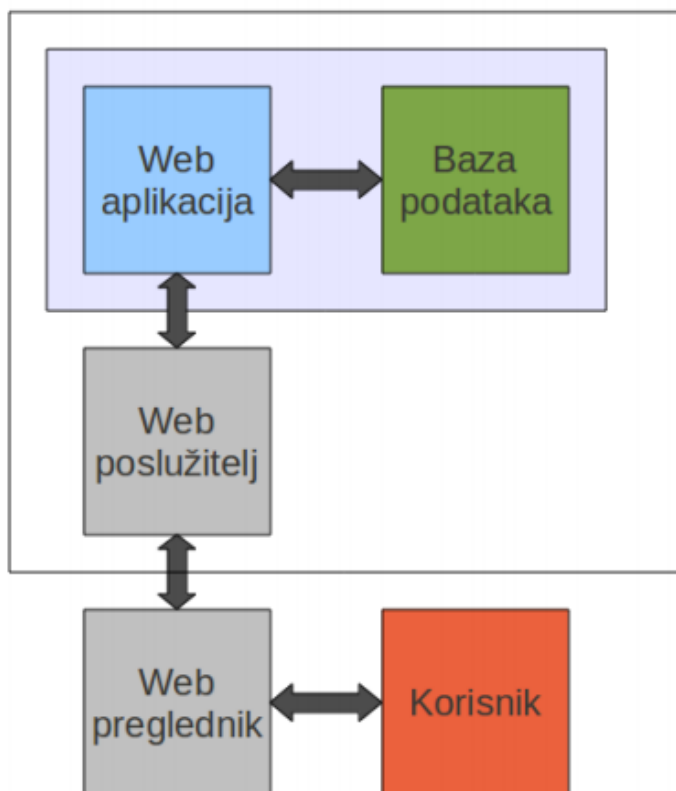
### 6.1. Svrha, opći prioriteti i skica sustava

Najidealnija arhitektura našeg sustava jest **web aplikacija**, jer ne zahtjeva od korisnika nikakvu instalaciju posebne programske podrške osim standardnog web preglednika kojeg većina korisnika ima instaliranog na svojim računalima. Dakle, naša će aplikacija morati zadovoljavati standarde današnjih web poslužitelja.

Za rad aplikacije bit će bitna baza podataka u kojoj ćemo bilježiti i čuvati sve važne podatke potrebne za rad i funkcioniranje sustava. Aplikacija će koristiti prikladni DAO za pristup bazi podataka.

Najvažniji podsustavi arhitekture našeg sustava su:

1. Web poslužitelj
2. Web aplikacija
3. Baza podataka



Slika 6.1.1 Podsustavi arhitekture sustava

## Web poslužitelj

Web poslužitelj temeljni je dio bilo kojeg sustava na arhitekturi web aplikacije. Njegova je osnovna zadaća procesiranje zahtjeva korisnika kroz HTTP, osnovni mrežni protokol kojim se dijele informacije na World Wide Web-u.

Danas postoje razne kvalitetne programske implementacije web poslužitelja. Naša aplikacija konkretno će se izvoditi na *Apache Tomcat* web poslužitelju.

Upute za instalaciju i korištenje nalazit će se u prilogu napisanom nakon implementacije.

## Web aplikacija

Web aplikacija je prozor kroz koji korisnik komunicira sa sustavom. Ona korisničke zahtjeve oblikuje u operacije sustava. Aplikacija kroz bazu podataka upravlja sa svim podacima potrebnim za rad sustava. Web aplikacija sama stvara i puni bazu podataka po potrebi.

Još jedna zadaća web aplikacija je vizualna prezentacija sustava korisniku. Za to se oslanja na web preglednik. Sama aplikacija priprema vizualne informacije u obliku HTML dokumenata dok web preglednik prikazuje te HTML dokumente u obliku web stranica.

## Baza podataka

Zadaća baze podataka je brza i jednostavna pohrana, izmjena i dohvat podataka za daljnju obradu.

Naša baza podataka sastoji se od sljedećih entiteta:

- Utakmica
- Događaj
- Igrači
- Pozicija
- Tip događaja

- Virtualne ekipe
- Korisnik
- Klubovi
- Ogranak
- Poredak lige
- Kolo
- Poredak virtualne lige

**Utakmica** – Ovaj entitet sadržava sve važne informacije o utakmicama koje će se odigravati u ligi. Sadrži attribute: Šifra domaćina, Šifra gosta, Datum održavanja utakmice, Šifra najboljeg igrača u utakmici, Šifra utakmice te šifre više događaja koji su se odigrali u utakmici.

Ovaj entitet je u vezi One-to-One s entitetom. **Igrači** preko atributa šifre najboljeg igrača te u vezi One-to-Many s entitetom **Događaj** preko šifre događaja.

**Događaj** – Ovaj entitet sadržava informacije o odigranim događajima u utakmicama. Sadrži attribute: Šifra događaja, Ime događaja, Šifra tipa događaja te Šifra igrača koji je izvršio taj događaj.

Ovaj entitet je u vezi Many-to-One s entitetom **Utakmica** preko šifre događaja, One-to-One s entitetom **Tip Događaja** preko šifre tipa i u vezi One-to-One s entitetom **Igrači** preko šifre igrača koji je sudjelovao u događaju.

**Igrači** – Ovaj entitet sadržava osnovne informacije o igračima u rukometnim klubovima. Svaki igrač ima attribute: Šifra igrača, Ime igrača, Prezime igrača, Šifra kluba u kojem igra, Starost igrača, Cijena igrača, Šifra pozicije te ostvareni bodovi.

Ovaj entitet je u One-to-One vezama s entitetima **Utakmica**, **Događaj** i **Pozicija** preko atributa šifre igrača i šifre pozicije. Također je u vezi Many-To-One s entitetom **Klubovi** preko šifre igrača.

**Pozicija** – Ovaj entitet sadržava sve moguće pozicije u kojima igrač može igrati u rukometu. Pozicija ima attribute Šifre pozicije i Ime pozicije.

Ovaj entitet je u vezi One-To-One s entitetom **Igrači** preko atributa šifre pozicije.

**Tip događaja** – Ovaj entitet sadržava sve moguće tipove događaja koje se mogu dogoditi u jednoj utakmici. Tip događaja sadrži attribute Šifre tipa i Ime tipa.

U vezi je One-To-One s entitetom **Događaj** preko atributa šifre tipa.

**Virtualne ekipe** – Predstavlja virtualne ekipe koje svaki korisnik može stvoriti da bi se natjecao u virtualnoj ligi. Sadrži attribute: Šifra ekipe, Naziv ekipe, Trenutni proračun.

Preko atributa šifre ekipe entitet je u vezi One-To-One s entitetom **Korisnik**.

**Korisnik** - Entitet korisnik služi za pohranjivanje i čuvanje podataka koje opisuju korisnika u našem opisanom sustavu web aplikacije za rukometnu ligu. Sadrži slijedeće attribute: korisničko ime, korisnička lozinka, šifra korisnika, ime, prezime, e-mail, država, šifra ekipe, šifra podržanog kluba, tip korisnika i šifra virtualne ekipe.

Entitet je u vezi One-To-One s entitetom **Virtualna Ekipa** preko atributa šifra ekipe, u vezi Many-To-One s entitetom **Klubovi** te u vezi Many-To-One s entitetom **Poredak virtualne lige**.

**Klubovi** - Entitet Klubovi služi nam za pohranjivanje i čuvanje podataka koje opisuju klub u našoj rukometnoj ligi. Sadrži attribute: Šifra kluba, šifra ogranaka, ime kluba, vrijednost kluba, broj igrača, ostvareni bodovi.

Entitet je u vezi One-To-Many s entitetom **Korisnik** preko atributa šifre kluba, Many-To-One s entitetom **Ogranak** preko šifre ogranka i Many-To-One vezi s entitetom **Poredak Lige** preko šifre kluba.



**Ogranak** - Entitet ogranak predstavljanja ogranak rukometne lige. Sadrži dva atributa: Šifra ogranka i Ime ogranka.

Preko atributa Šifra ogranka entitet je u vezi One-To-Many s entitetom **Klubovi**.

**Poredak lige** - Entitet Poredak Lige predstavlja poredak rukometne lige. Sastoji se od dva atributa: Šifra kluba i Šifra kola. Preko atributa Šifra kluba entitet je u vezi One-To-Many s entitetom **Klubovi**, a preko atributa šifre kola je u vezi One-To-Many s entitetom **Kolo**.

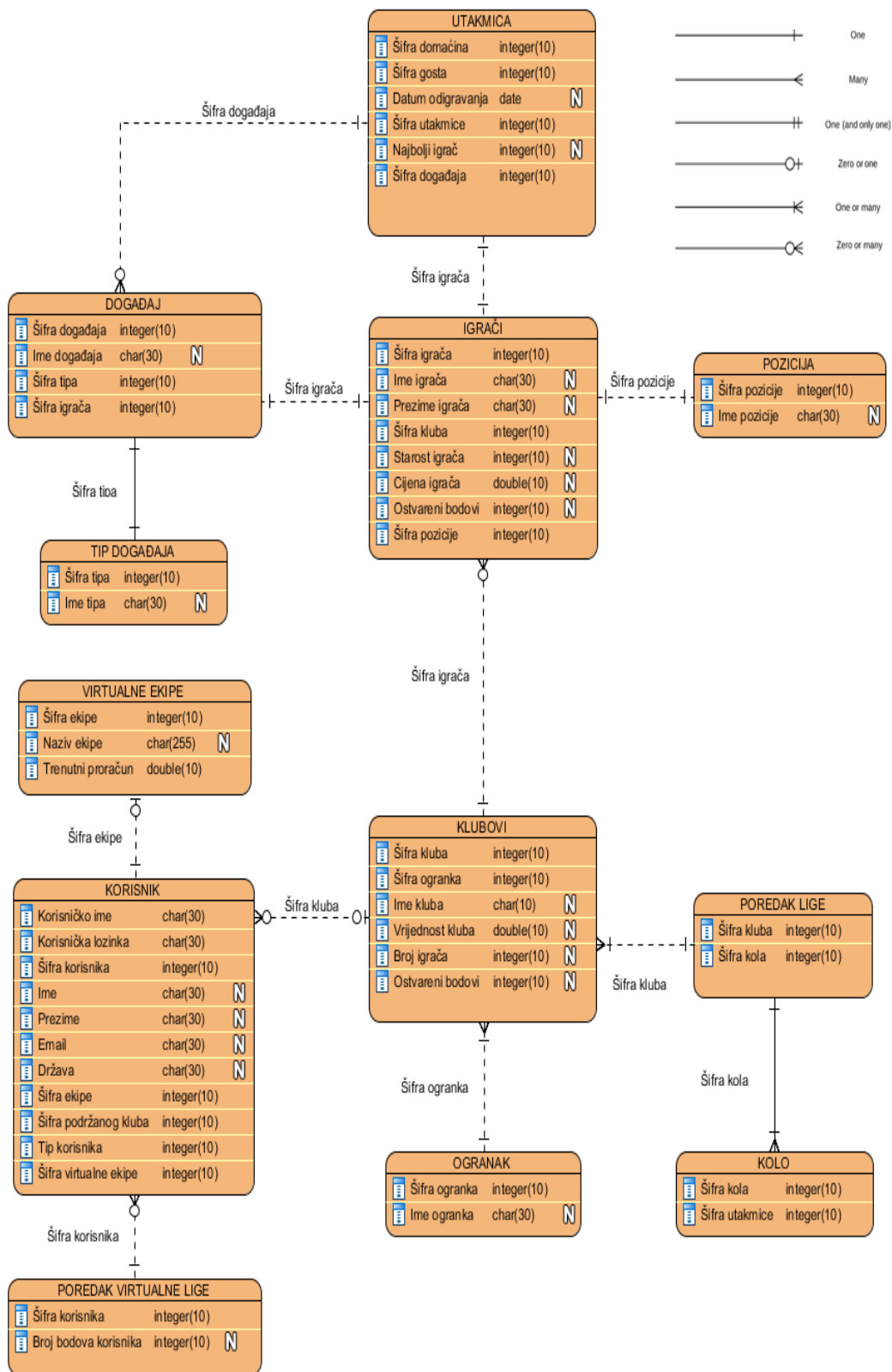
**Kolo** - Entitet Kolo predstavlja jedno kolo odigravanja utakmica rukometne lige.

Sastoji se od dva atributa: Šifra kola i Šifra utakmice.

Preko atributa šifre koja je u vezi Many-To-One s entitetom **Poredak lige**.

**Poredak virtualne lige** - Entitet Poredak virtualne lige predstavlja poredak virtualnih ekipa virtualne lige. Sastoji se od dva atributa: Šifra korisnika i Broj bodova korisnika.

Preko atributa šifre korisnika entitet je u vezi One-To-Many s entitetom **Korisnik**.



Slika 6.1.2 ER dijagram baze podataka

## 6.2. Dijagram razreda s opisom

Razredi su podijeljeni u tri velike cjeline: podpaket **modeli**, podpaket **dao** i podpaket **simulator**. U podpaketu **modeli** nalaze se svi potrebni modeli podataka koji se koriste u izradi aplikacije, u podpaketu **dao** nalaze razredi zaduženi za komunikaciju s bazom podataka, a u podpaketu **simulator** nalazi se sve potrebno za simulaciju utakmica.

Podpaket **modeli** je podijeljen također u dvije skupine.

Prva skupina zove se **liga** i sadrži sve razrede koji se odnose na samu implementaciju virtualne i prave rukometne lige.

**Liga** je osnovni razred koji predstavlja stvarnu rukometnu ligu. Sadrži klubove koji sudjeluju, te listu kola.

**Kolo** predstavlja jedno kolo lige. Sadrži listu utakmica koje se održavaju u tom kolu kao i **Ligu** u kojoj se nalazi.

**Utakmica** predstavlja jednu utakmicu. Sadrži **Klub** domaćina, **Klub** gosta, datum odigravanja utakmice, listu **Događaja** na utakmici te najboljeg **Igrača** (mvp u dijagramu) i **Kolo** u kojem se odigrava.

**Klub** predstavlja jedan od klubova koji se natječu u stvarnoj ligi. Sadrži naziv kluba, ogranak u kojem se nalazi, vrijednost kluba i listu **Igrača** koji nastupaju za njega.

**Igrač** predstavlja jednog igrača koji sadrži ime, prezime, godine, vrijednost i **Poziciju** na kojoj igra. Uz to, ima i **Klub** kojem pripada.

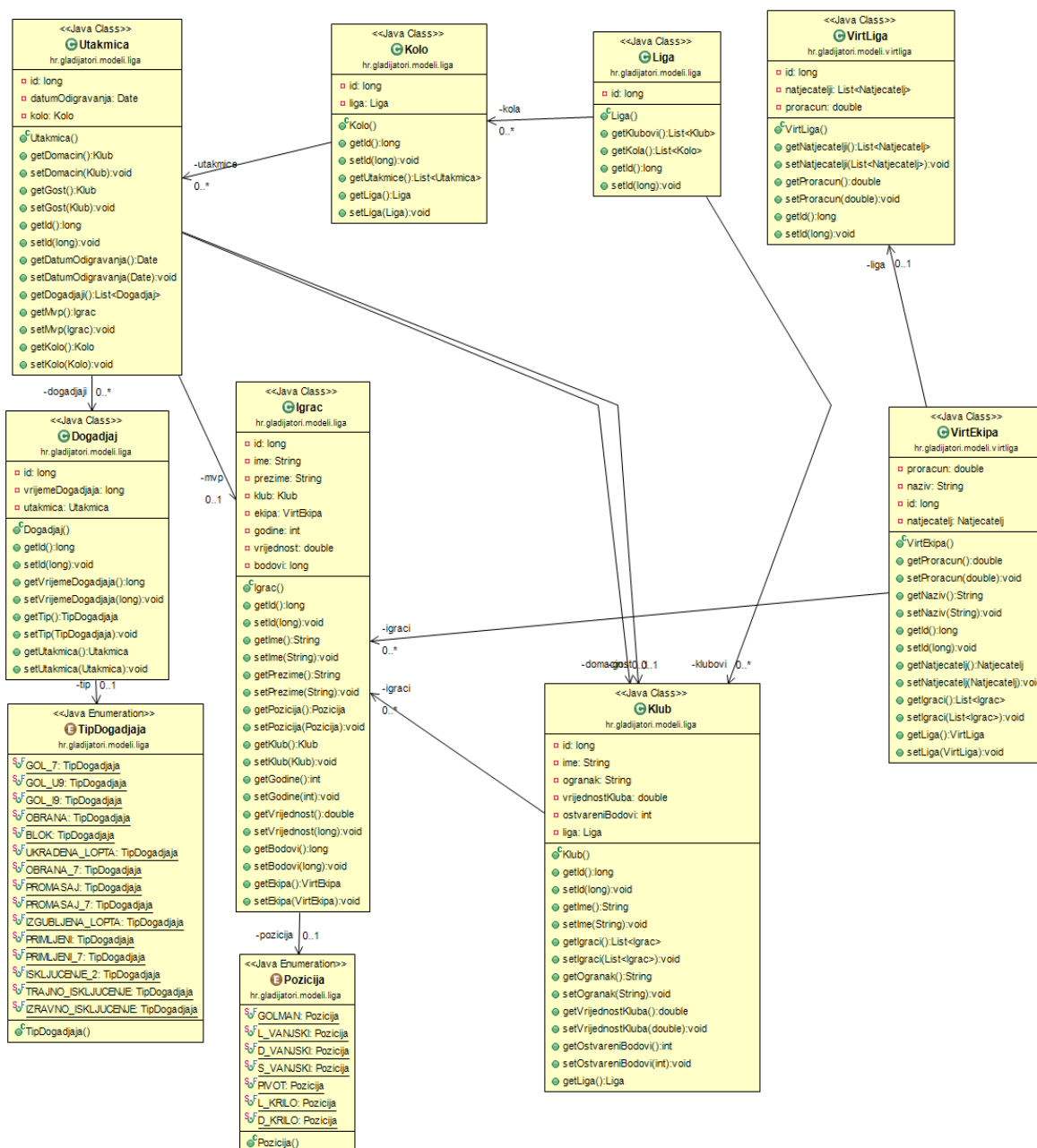
**Pozicija** je posebni enum koji se odnosi na moguće različite pozicije koje igrač može igrati.

**Događaj** se odnosi na pojedini događaj koji se može dogoditi tijekom utakmice. Sadrži vrijeme događaja (sekundu u kojoj se dogodio), **TipDogađaja** koji određuje koja se vrsta događaja dogodila i **Utakmicu** na kojoj se dogodio.

**TipDogađaja** je enum koji sadrži sve moguće događaje koji se mogu dogoditi na utakmici.

**VirtLiga** predstavlja virtualnu ligu u kojoj se **Natjecatelji** natječu. Sadrži listu **Natjecatelja** koji sudjeluju, te predefinirani proračun za svaku **VirtEkipu**.

**VirtEkipa** predstavlja jednu ekipu koja sudjeluje u virtualnoj ligi. Sastoji se od trenutnog proračuna, naziva, te liste **Igrača** koji su prijavljeni za tu ekipu i **VirtLiga** kojoj pripada.



Slika 6.2.1 – Dijagram razreda podpaketa modeli.liga

Druga skupina modela se odnosi na korisnike. Sadrži izvorni razred **Korisnik** iz kojeg su izvedeni podrazredi koji se odnose na pojedinu vrstu korisnika koje razlikujemo. Svaki od izvedenih korisnika ima posebne metode i razinu dozvola u

aplikaciji. Uveden je i poseban **TipKorisnika** koji služi za jednostavno razlikovanje različitih korisnika.

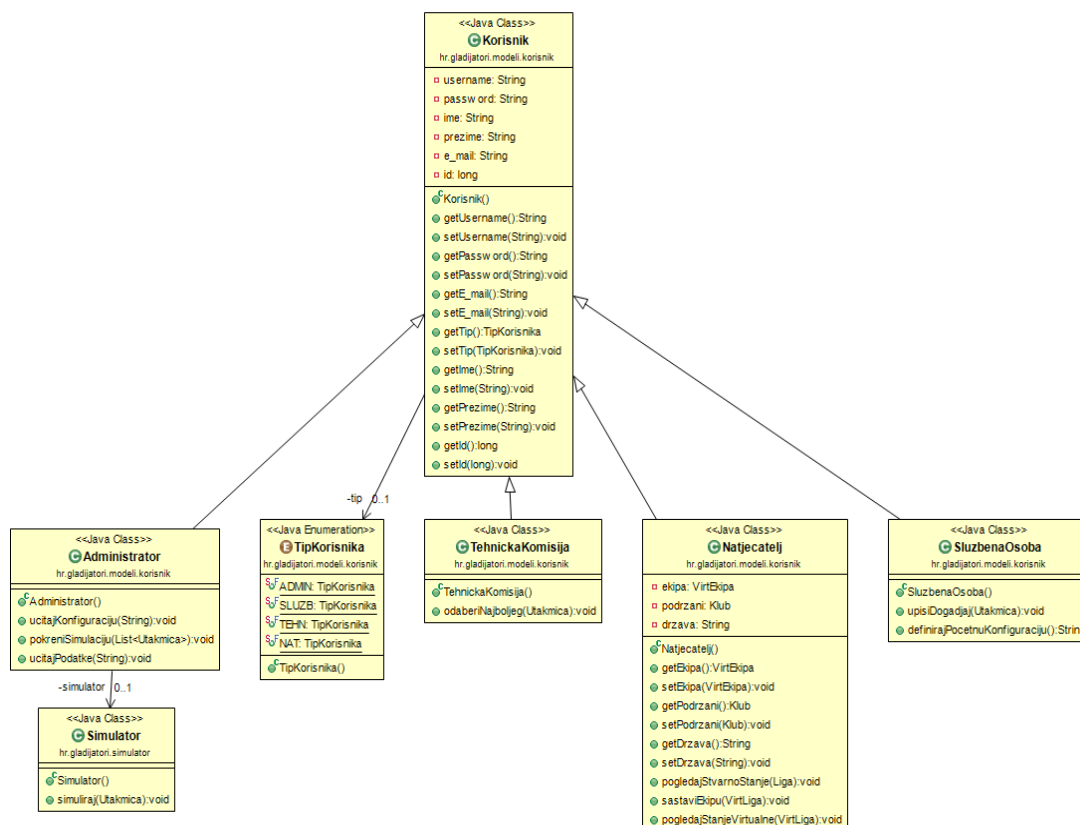
**Korisnik** kao izvorni razred sadrži sve zajedničke atribute (korisničko ime, lozinku, ime, prezime, e-mail) korisnika i nudi metode za postavljanje i dohvaćanje istih.

Tako **Administrator** ima metode za učitavanje konfiguracije (kao parametar primi putanju do datoteke u kojoj se nalazi konfiguracija), pokretanje **Simulatora** (kao parametar prima sve utakmice koje treba simulirati) i učitavanje podataka lige (kao parametar prima putanju do datoteke u kojoj se nalaze podatci). Posebni atribut kojeg **Administrator** ima je **Simulator** kojeg koristi za simulaciju utakmica.

**TehnickaKomisija** ima metodu za biranje najboljeg igrača (kao parametar prima utakmicu na kojoj treba odabrati najboljeg igrača).

**Natjecatelj** ima metode za provjeru stanja lige (kao parametar prima ligu koju provjerava), sastavljanje ekipe (kao parametar prima virtualnu ligu u kojoj sudjeluje) i provjeru stanja virtualne lige (kao parametar prima virtualnu ligu koju provjerava). Također, ima i dodatne atribute koje samo on posjeduje. To su *ekipa*, *podrzani* i *drzava*. *Ekipa* je njegova ekipa u virtualnoj ligi, *podrzani* je stvarni klub kojeg podržava, a *drzava* je država iz koje dolazi.

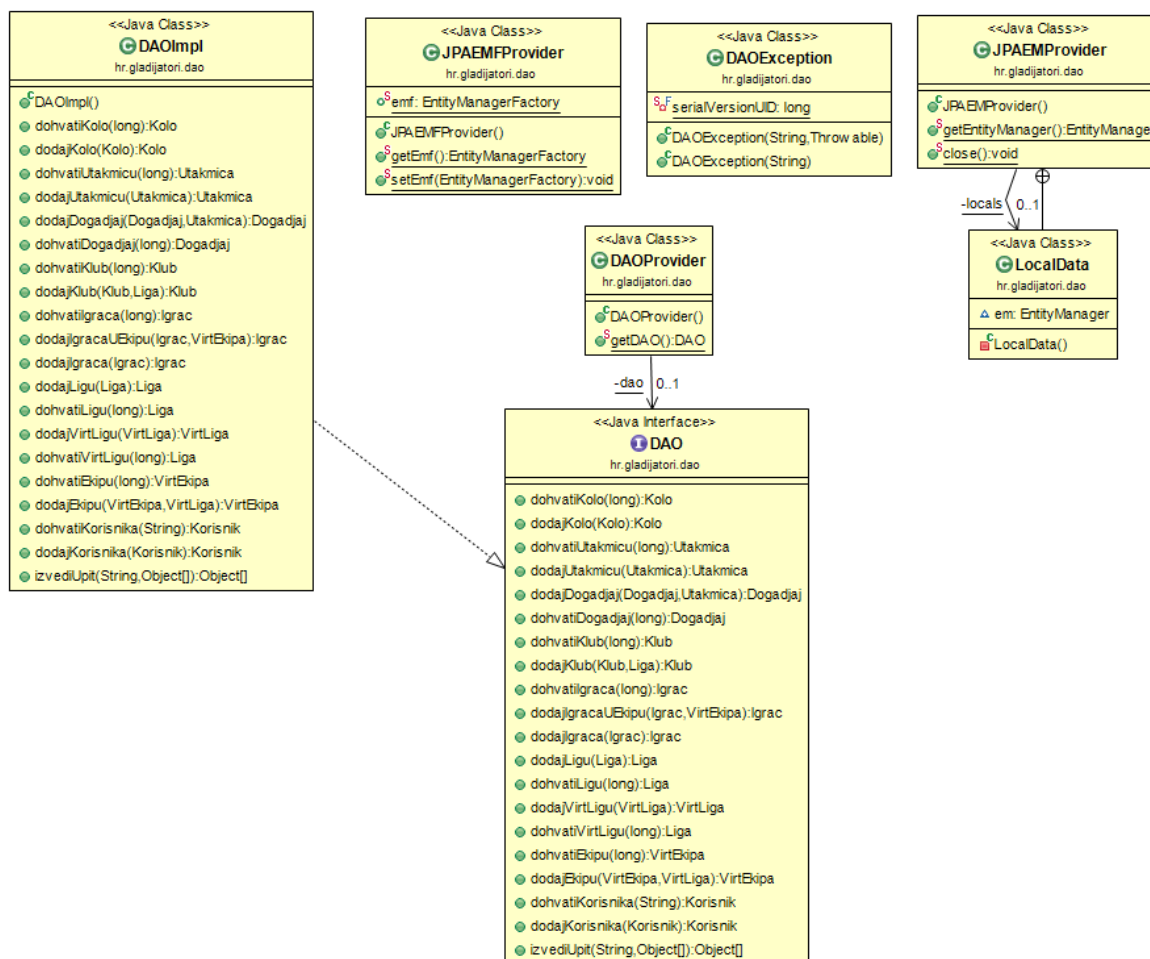
**SluzbenaOsoba** ima metode za upisivanje događaja neke utakmice (kao parametar prima utakmicu na koju upisuje) i definiranje početne konfiguracije (zapisuje u datoteku i vraća putanju do nje).



Slika 6.2.2 – Dijagram razreda podpaketa modeli.korisnik i simulator

Druga velika cjelina cijele aplikacije je **dao**, odnosno dio za upravljanje podacima i povezivanje s bazom podataka. Ključno sučelje je **DAO** koje sadrži sve potrebne metode za upravljanje bazom podataka. Te se metode odnose na dodavanje objekata u bazu, dohvaćanje objekata u bazu, ali i mogućnost izvođenja korisničkog SQL upita. **DAOImpl** je razred koji implementira sve potrebne metode u skladu s korištenom bazom podataka.

Cijeli **DAO** sloj zamišljen je da radi kao Singleton, postojala bi samo jedna instanca **DAOImpl** koja se može dohvatiti iz bilo kojeg dijela programa, a taj objekt bi pružao **DAOProvider**. Ostali razredi služe za inicijalizaciju i prekid veze s bazom podataka.



Slika 6.2.3 – Dijagram razreda podpaketa DAO

Treća i zadnja cjelina je **simulator**. Sastoji se od potrebnih razreda za simulaciju utakmice.

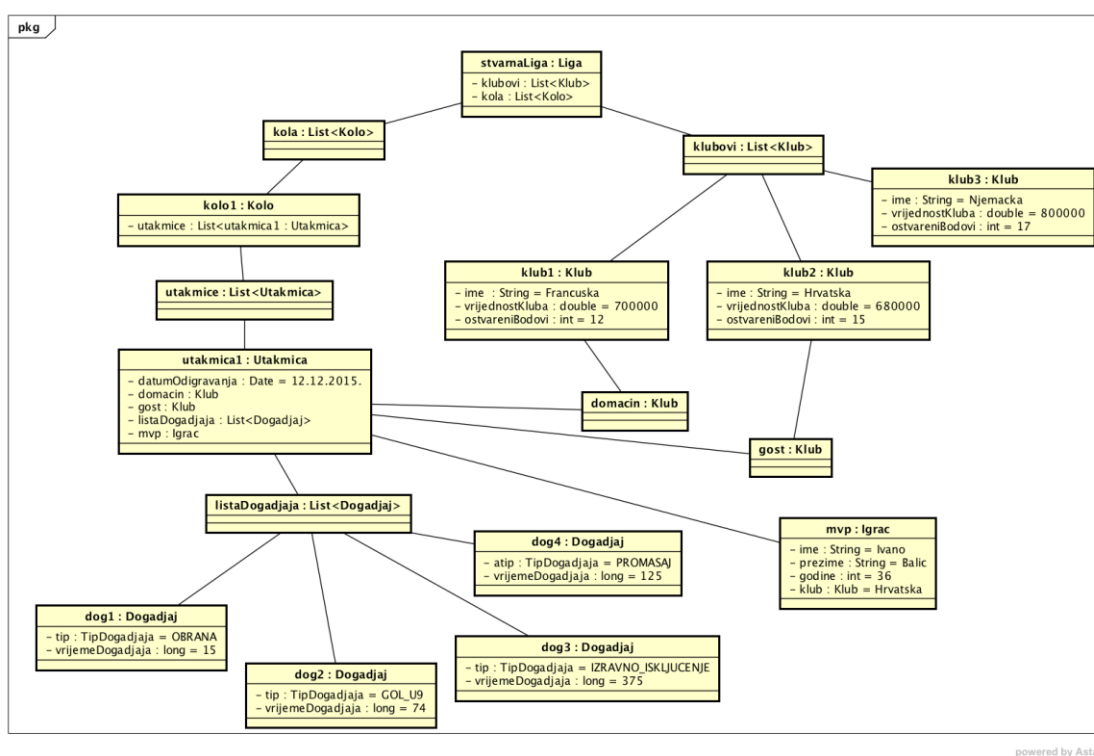
### 6.3. Dijagram objekata

Kako bismo prikazali ponašanje sustava u stvarnom vremenu, prikazat ćemo kako naš razredni model izgleda u dvije specifične situacije. Za prvi primjer uzeli smo jednu stvarnu ligu koja se sastoji od liste **Klubova** i liste **Kola**. Lista **Kola** sadrži instance razreda **Kolo**, a između ostalog i prikazani objekt **kolo1**. Svako kolo ima listu utakmica koje sadrži, i tako **kolo1** sadrži instancu razreda **Utakmica**, **utakmica1**. **Utakmica1** se sastoji od datuma održavanja, dva **Kluba** (domaćin i gost), liste **Događaja** utakmice te najboljeg igrača. Lista instanci **Događaj** utakmice, puni se dok se utakmica odvija, a svaki **Događaj** ima svoj određeni tip (GOL,

OBRANA, PROMASAJ) te varijablu vrijemeDogadjaja koja označava u kojoj sekundi utakmice se događaj dogodio. Varijabla **mvp** označava najboljeg igrača utakmice i pokazuje na instancu razreda **Igrac** koja sadrži igračevo ime, prezime, njegove godine te njegov klub.

Uz to, svaka liga ima i listu instanci razreda **Klub**, u kojoj se nalaze svi klubovi koji trenutno igraju u toj ligi, sa svojim podacima: ime, vrijednostKluba, ostvareniBodovi.

Ovi objekti interno sadrže još neke varijable (npr. Liga, Klub, Utakmica, Igrac i Dogadjaj imaju varijablu id koja služi za rukovanje podacima iz baze podataka) koje, zbog sažetosti prikaza, nismo ovdje prikazali.

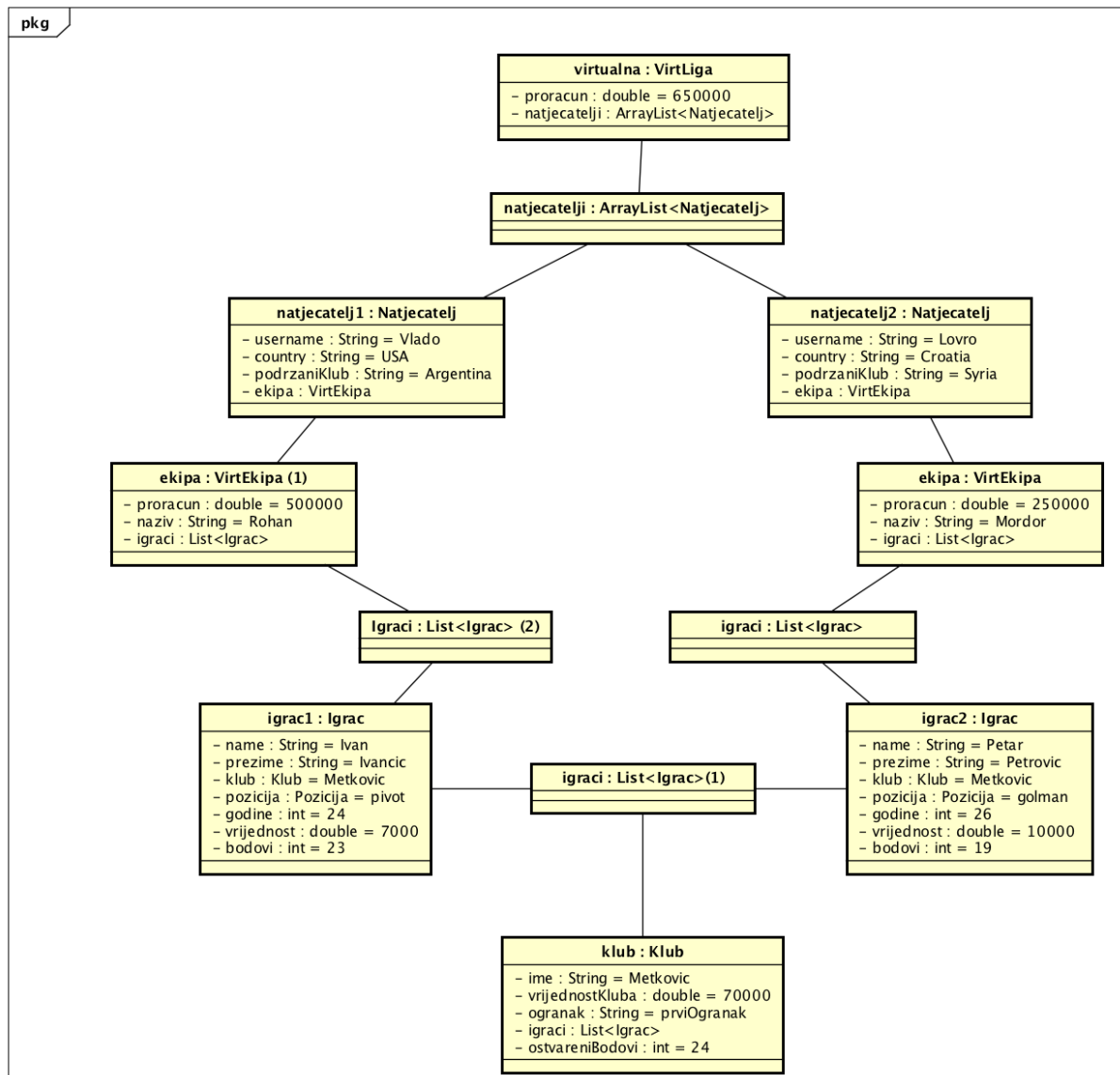


Slika 6.3.1 – Dijagram objekata Stvarna liga

Kao drugi primjer, prikazat ćemo jednu virtualnu ligu. Za razliku od stvarne, virtualna liga se izvodi između registriranih **Korisnika** na sustavu, koji slažu svoje **Ekipe**, stvarnim **Igracima**, čije performanse rastu ili padaju ovisno o njihovom učinku u stvarnoj **Ligi**. Tako i prikazana **VirtLiga**, virtualna, ima svoj proračun te sadrži listu **Natjecatelja** koji u njoj sudjeluju. **Natjecatelji** su korisnici, koji se prijavljuju na sustav. Klasa **Natjecatelj** sadrži osobne podatke korisnika (username, country,



podržaniKlub te još neke koje na dijagramu nisu prikazane poput password, e-mail) i ekipu koju je **Natjecatelj** odabrao. **Ekipa** natjecatelja je instanca razreda **VirtEkipa**.



powered by Astah

Slika 6.3.2. - Dijagram objekata Virtualna Liga

Ona sadrži proračun i naziv ekipe te listu **Igraca**. Svaki od **Igraca** ime svoje ime, prezime, stvarni **Klub** u kojem igra, svoju poziciju na kojoj igra, vrijednost (cijenu), te bodove koje je do sada ostvario u stvarnoj **Ligi**. Na dijagramu smo prikazali slučaj kada dva različita **Natjecatelja** odaberu **Igrače** iz istog stvarnog **Kluba**.

## *6.4. Ostali UML dijagrami*

## **7. Implementacija i korisničko sučelje**

*7.1. Dijagram razmještaja*

*7.2. Korištene tehnologije i alati*

*7.3. Isječak programskog koda vezan za temeljnu funkcionalnost*

*7.4. Ispitivanje programskog rješenja*

*7.5. Upute za instalaciju*

*7.6. Korisničke upute*

## 8. Zaključak i budući rad

Analiziran je projektni zadatak, napravljeni su dijagrami obrazaca uporabe, sekvencijski dijagrami, model sustava, dijagrami razreda te dijagrami objekata.

Daljni rad na projektu biti će usmjeren na implementaciju svih dijelova potrebnih za efikasno izvođenje web-stranice namijenjene ostvarenju virtualne rukometne lige.

## 9. Popis literature

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. Oblikovanje programske potpore, FER ZEMRIS, <http://www.zemris.fer.hr/predmeti/opp>
3. Apache Tomcat 7, <http://tomcat.apache.org/tomcat-7.0-doc>
4. Apache Derby 10.8, <http://db.apache.org/derby/manuals/>. Marsic, „Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. Jboss Hibernate, <http://www.hibernate.org/docsUML2ClassDiagramGuidelines>, <http://www.agilemodeling.com/style/classDiagram.htm>
6. Astah Community, <http://astah.net/editions/community>

## **Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda)**

## **Dodatak B: Dnevnik sastajanja**

### **Sastanak1: 19.10.2015. - FER**

**Prisutni:** Goran Brlas, Filip Džidić, Antonio Ivčec, Vinko Kolobara, Vladimir Dragutin Livaja, Lovre Torbarina, Lovro Vidović

**Sažetak:** Upoznavanje članova tima i inicijalna diskusija o zadatku. Analiza ključnih točaka zadatka i formiranje potrebnih objašnjenja funkcionalnosti.

**Zaključci:** Najavljen početak na formalnom opisivanju problema, funkcijskih i ostalih zahtjeva. Rad na UML dijagramima odgođen dok svi članovi grupe nisu upoznati s teorijskom pozadinom i potrebnim alatima.

### **Sastanak2: 21.10.2015. - FER**

**Prisutni:** Goran Brlas, Filip Džidić, Antonio Ivčec, Vinko Kolobara, Vladimir Dragutin Livaja, Lovre Torbarina, Lovro Vidović

**Sažetak:** Sastanak s asistentom. Dodatna analiza projekta te rješavanje nedoumica oko projekta.

**Zaključci:** Početni dogovor oke podjele posla, dogovoreno da će svaki član tima raditi na određenom dijelu dokumentacije u 1. ciklusu.

### **Sastanak3: 12.11.2015. - FER**

**Prisutni:** Goran Brlas, Filip Džidić, Antonio Ivčec, Vinko Kolobara, Vladimir Dragutin Livaja, Lovre Torbarina, Lovro Vidović

**Sažetak:** Dogovor oko arhitekture sustava. Napravljen model razreda te model baze podataka.

**Zaključci:** Podjela posla oko izrade dijagrama na temelju napravljenog modela.

#### **Sastanak4: 15.11.2015. - FER**

**Prisutni:** Filip Džidić, Lovre Torbarina

**Sažetak:** Izrada dokumentacije arhitekture sustava te modela baze podataka.

#### **Sastanak5: 17.11.2015. - FER**

**Prisutni:** Antonio Ivčec, Vladimir Dragutin Livaja

**Sažetak:** Izrada dokumentacije dijagrama objekata.



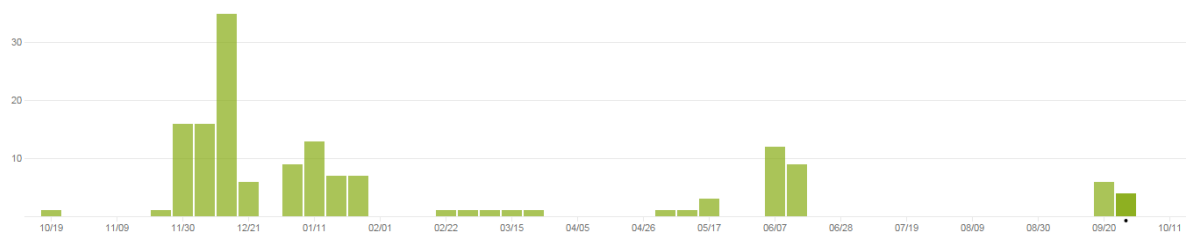
## Dodatak C: Prikaz aktivnosti grupe

### Članovi grupe (abecednim redom)

Popis aktivnosti	Goran Brlas	Filip Džidić	Antonio Ivčec	Vinko Kolobara	Vladimir Dragutin Livaja	Lovre Torbarina	Lovro Vidović
Upravljanje projektom	100%						
Opis projektnog zadatka	100%						
Rječnik pojmova	40%	10%	10%	10%	10%	10%	10%
Opis funkcionalnih zahtjeva	50%						50%
Opis ostalih zahtjeva	100%						
Arhitektura i dizajn sustava							
Svrha, opći prioriteti i skica sustava		50%				50%	
Dijagram razreda s opisom							
Dijagram objekata			50%		50%		
Ostali UML dijagrami				100%			
Implementacija i korisničko sučelje							

Dijagram razmještaja							
Korištene tehnologije i alati							
Isječak programskog kôda							
Ispitivanje programskog rješenja							
Upute za instalaciju							
Korisničke upute							
<b>Plan rada</b>							
<b>Pregled rada i stanje ostvarenja</b>	100%						
<b>Zaključak i budući rad</b>	100%						
<b>Popis literature</b>	100%						
<b>Dodaci</b>							
Indeks							
Dnevnik sastajanja	100%						

## Pregled pohrana kroz vrijeme trajanja projekta (primjer):



*Dodatak D: Plan rada / Pregled rada i stanje ostvarenja*

Plan rada za rev. 2 je implementirati sustav ostvarenja virtualne rukometne lige, te nadopuniti dokumentaciju svim potrebnim informacijama.

Ostvareno u rev. 1:

- Opis projektnog zadatka,
- Funkcionalni zahtjevi,
- Ostali zahtjevi,
- Arhitektura i dizajn sustava