



Pure Logic
Games

Specifikacija dizajna softvera

Olimijada

Pure Logic:

Stefan Nestorović

Lazar Stanojević

Dragan Đurović

Strahinja Milutinović

Datum kreiranja:

09.06.2018

Poslednja izmena:

16.06.2018

Sadržaj

| | |
|---------------------------------------|----|
| Sadržaj | 2 |
| Istorija izmena..... | 4 |
| Uvod | 5 |
| Namena dokumenta | 5 |
| Namena proizvoda | 5 |
| Opseg projekta | 5 |
| Dodatni materijal | 6 |
| Akrоними и definicije..... | 6 |
| Pregled dokumenta | 8 |
| Opšti opis proizvoda | 9 |
| Klase korisnika i karakteristike..... | 9 |
| Funkcije proizvoda..... | 9 |
| Razmatranje dizajna | 11 |
| Prepostavke | 11 |
| Ograničenja..... | 11 |
| Radno okruženje | 12 |
| Arhitektura sistema | 13 |
| Pregled sistema..... | 13 |
| Obrazloženje dizajna | 15 |
| Dizajn visokog nivoa | 16 |
| Konceptualni pogled | 16 |
| Fizički pogled..... | 18 |
| Dizajn niskog nivoa | 19 |
| Dizajn baza podataka | 19 |
| Model SQLite baze podataka | 19 |
| Šeme MongoDB baze podataka | 22 |
| Dijagrami klasa..... | 24 |
| Klijentski deo..... | 24 |
| Serverski deo..... | 31 |
| Dijagrami slučajeva korišćenja | 35 |

| | |
|---|----|
| Slučaj korišćenja: Registrovanje na sistem | 36 |
| Slučaj korišćenja: Prijava na sistem | 38 |
| Slučaj korišćenja: Pregled igara | 40 |
| Slučaj korišćenja: Održavanje profila | 42 |
| Slučaj korišćenja: Analiza korisnika..... | 44 |
| Slučaj korišćenja: Leaderboard | 44 |
| Slučajevi korišćenja: Pregled igara | 45 |
| Slučajevi korišćenja: Pregled turnira | 46 |
| Slučajevi korišćenja: Pregled botova..... | 47 |
| Slučaj korišćenja: Prijava na turnir | 48 |
| Slučaj korišćenja: Banovanje korisnika | 50 |
| Slučajevi korišćenja: Upravljanje igram | 51 |
| Slučajevi korišćenja: Upravljanje turnirima | 52 |
| Dijagrami sekvenci | 54 |
| Registracija korisnika na sistem | 54 |
| Prijava korisnika na sistem..... | 55 |
| Dodavanje igre..... | 56 |
| Gledanje mečeva | 57 |
| Dodavanje turnira..... | 58 |
| Dizajn korisničkog interfejsa..... | 59 |
| Tehnologije korisničkog interfejsa..... | 59 |
| Slike aplikacije..... | 59 |

Istorija izmena

| Autor | Opis | Datum | Verzija |
|-----------------------|---|------------|---------|
| Stefan Nestorović | Kreiranje dokumenta. Pisanje prva tri poglavija. | 09.06.2018 | 0.1 |
| Stefan Nestorović | Pisanje arhitekture sistema i konceptualnog pogleda. | 10.06.2018 | 0.3 |
| Lazar Stanojević | Pisanje dizajna baze podataka i dizajna korisničkog interfejsa. | 10.06.2018 | 0.5 |
| Stefan Nestorović | Ispravljanje dokumenta prema dobijenim sugestijama. | 12.06.2018 | 0.5 |
| Stefan Nestorović | Pisanje fizičkog pogleda na sistem i završavanje poglavlja o dizajnu baza podataka. | 13.06.2018 | 0.6 |
| Lazar Stanojević | Dodavanje dijagrama klasa za klijentski i serverski deo. Ažuriranje slike dizajna korisničkog interfejsa. | 15.06.2018 | 0.7 |
| Stefan Nestorović | Dodavanje opisa dijagramima klasa. | 15.06.2018 | 0.8 |
| Strahinja Milutinović | Dodavanje dijagrama slučajeva korišćenja i završavanje dijagrama sekvenci. | 15.06.2018 | 0.9 |
| Stefan Nestorović | Dodavanje pojedinih opisa u dizajnu niskog nivoa. Ispravljanje sitnijih grešaka. Export dokumenta u PDF format. | 15.06.2018 | 1.0 |

Uvod

Namena dokumenta

Ovaj dokument pruža detaljan opis dizajna i implementacije Web aplikacije Olimijada, prema zahtevima navedenim u dokumentu *Specifikacija softverskih zahteva projekta Olimijada*.

Namena proizvoda

Ideja projekta Olimijada jeste izrada softverske platforme koja će omogućavati prijavljenim korisnicima da se takmiče u programiranju botova sa drugim korisnicima sistema. U okruženju aplikacije korisnicima će biti ponuđene određene igre, od kojih svaka ima posebno definisana pravila po kojima se igra. U tim igrama neće učestvovati direktno korisnici, već botovi koje korisnici kreiraju i podižu na sistem. Botovi su softverske komponente koje znaju samostalno da se snalaze u nekom softverskom okruženju. Cilj botova u ovom kontekstu jeste da se što bolje snađu u definisanim pravilima neke igre i da u njoj pobede. Korisnici se takmiče sa svojim botovima protiv botova drugih takmičara na turnirima koji se organizuju na nivou igre. Biće omogućen vizuelni prikaz simulacija mečeva na turnirima u realnom vremenu, tako da će korisnici moći da isprate ponašanje svojih botova do najsitnijih detalja.

Cilj proizvoda jeste da privuče sve entuzijaste za programiranje i veštačku inteligenciju, da ih približi ljudima sličnih interesovanja i da ih kroz takmičarsku atmosferu natera na što kreativnije razmišljanje i rešavanje problema. Na ovaj način će korisnici sistema jedni druge podsticati na dalji napredak, dok će im zanimljiv sadržaj i kompleksni problemi koje aplikacija nudi zaokupirati svu pažnju.

Opseg projekta

U ovom odeljku će u kratkim crtama biti izloženo šta je sve od opisane platforme deo posla koji će obaviti razvojni tim *Pure Logic*, a šta nije u opsegu posla na ovom projektu.

Deo krajnjeg proizvoda koji će tim *Pure Logic* razvijati čine mogućnosti:

- Dodavanja novih igara, definisanja novih turnira za određene igre, upravljanja korisničkim nalozima.
- Prijave i registracije korisnika, održavanja korisničkog profila.

- Pregleda igara, turnira, mogućnost otpreme (upload) i testiranja bota, i prijave botova na turnire.
- Praćenja rezultata i gledanja svih mečeva.

Deo krajnjeg proizvoda koji tim *Pure Logic* **neće** razvijati čine:

- Izrada logike i pravila igara koje će postojati.
- Izrada serverske komponente koja će upravljati simulacijom mečeva. Naš deo posla u vezi sa ovom komponentom jedino je prosleđivanje fajlova koji predstavljaju ponašanje botova, i na kraju simulacije, preuzimanje njenih rezultata u vidu koordinata svih objekata koje čine taj meč.

Dodatni materijal

- Specifikacija softverskih zahteva projekta Olimijada
- <http://theaigames.com/> - Primer aplikacije sličan krajnjem cilju projekta
- <https://halite.io/> - Primer aplikacije sličan krajnjem cilju projekta
- <https://angular.io/> - Dokumentacija za Angular 5.
- <https://nodejs.org/en/> - Dokumentacija za Node.js
- <https://www.sqlite.org/docs.html> - Dokumentacija za SQLite
- <https://docs.mongodb.com/> - Dokumentacija za MongoDB
- <https://phaser.io/examples> - Dokumentacija za Phaser

Akronimi i definicije

Radi boljeg razumevanja dokumenta, u sledećoj tabeli su navedeni pojmovi i skraćenice čije značenje nije opšte poznato ili ima posebno značenje u sklopu ovog dokumenta.

| Pojam | Objašnjenje |
|-------|---|
| Bot | Softverska komponenta koja je programirana da se samostalno snalazi u nekom softverskom okruženju. U našem sistemu će fizički biti predstavljena jednim fajlom. |
| Igrač | Jedna instanca (primerak) bota. |

| | |
|-------------------|--|
| Igra | Entitet u sistemu koji poseduje set pravila, kojih svi igrači te igre treba da se pridržavaju, tj. igra je okruženje za botove. Igre mogu biti pojedinačne ili timske, zavisno od toga da li se u igri takmiče jedan igrač protiv drugog igrača, ili više na više. |
| Meč | Jedna instanca (primerak) igre. U meču dolazi do takmičenja igrača dva konkretna korisnika. |
| Turnir | Organizacija više mečeva gde se ukrštaju različiti učesnici za određenu igru... Može biti kup ili liga, što utiče na način uparivanja učesnika. |
| Web | World Wide Web (WWW) je sistem dokumenata na internetu koji su međusobno povezani. |
| HTTP | Hypertext Transfer Protocol (HTTP) je mrežni protokol koji predstavlja glavni i najčešći metod prenosa informacija na Web-u. |
| Framework (okvir) | Softversko okruženje koje korisnicima omogućava olakšani rad u pronalaženju rešenja za specifične probleme korišćenjem ugrađenih funkcija. |
| Server | Računarski sistem koji pruža usluge drugim računarskim sistemima koji se nazivaju klijenti. |
| Front-end | Deo aplikacije koji korisnik može da vidi, i sa kojim može na neki način da vrši interakciju. |
| Back-end | Deo aplikacije koji je nevidljiv za korisnika, a koji obavlja sav posao i rad sa podacima u pozadini. |
| API | Application Programming Interface definiše načine na koje aplikacije mogu da zahtevaju usluge (servise) od biblioteka i/ili operativnih sistema. |
| Open source | Odnosi se na softver čiji je kod dostupan unutar "open source" licence, što znači da je njegov kod dostupan svima besplatno, i da se pored toga dozvoljava njegova promena i poboljšanje. |
| Javascript | Skriptni jezik čija je glavna primena vezana za definisanje funkcionalnosti web stranica na klijentskoj strani. |
| Typescript | Programski jezik koji je strogi nadskup Javascript-a, a koji mu dodaje opcionu staticku tipizaciju i objektnu orijentisanost. |
| JSON | JavaScript Object Notation (JSON) je tekstualno baziran, otvoreni standard, dizajniran za razmenu podataka. |
| Angular 5 | Open source javascript framework za razvoj jednostraničnih web aplikacija. |
| Node.js | Open source javascript framework za izvršavanje javascript koda na serverskoj strani. |
| SQLite | Relaciona baza podataka koja se odlikuje time što sve podatke čita i upisuje u jedan fajl. Otvorenog je koda, nema serverski proces, brza je i pouzdana. |

| | |
|---------------|---|
| MongoDB | Nerelaciona baza podataka koja sve podatke čuva kao JSON dokumente sa dinamičkim šemama. Otvorenog je koda, veoma je brza i skalabilna. |
| HTML5 | Hypertext Markup Language je opisni jezik specijalno namenjen opisu web stranica. |
| HTML5 Canvas | Canvas je poseban HTML element u kome se pomoću kontekstnih objekata mogu vršiti kompleksna grafička iscrtavanja. |
| CSS | Cascading Style Sheets (CSS) je jezik za definisanje izgleda elemenata Web stranica. |
| Phaser | Open source javascript framework za rad sa HTML5 canvas-om. |
| Local storage | Memorija koju pretraživači koriste za čuvanje podataka i nakon njihovog zatvaranja. |
| UML | Unified Modeling Language (UML) je standardni grafički jezik za modelovanje objektno-orientisanog softvera. |
| Linux | Operativni sistem otvorenog koda. |

Pregled dokumenta

Poglavlje 1 – **Uvod**, dat je kratak opis namene proizvoda i misije projekta, uz dodatne reference, akronime i definicije koje se koriste u sklopu celog dokumenta.

Poglavlje 2 - **Opšti opis proizvoda**, date su neke opšte crte i zahtevi koji se tiču proizvoda koji treba izgraditi.

Poglavlje 3 - **Razmatranje dizajna**, tu su date određene pretpostavke koje programer treba da ima u vidu za vreme razvoja, ograničenja koja moraju da se zadovolje kao i opis okruženja u kojem će aplikacija raditi.

Poglavlje 4 - **Arhitektura sistema**, pregled sistema i arhitektonski stil koji će biti u upotrebi, kao i detaljno obrazloženje zašto je baš takvo rešenje najpogodnije.

Poglavlje 5 - **Dizajn visokog nivoa**, gde su dati konceptualni i fizički pogledi na sistem, sa njihovim grubim vizuelnim prikazima.

Poglavlje 6 - **Dizajn niskog nivoa**, daje detaljan opis za svaku od komponenti celog sistema uz prateće dijagrame i objašnjenja.

Poglavlje 7 - **Dizajn korisničkog interfejsa**, služi da prikaže kako gotova aplikacija zapravo treba da izgleda, uz slike i prateća objašnjenja.

Opšti opis proizvoda

Klase korisnika i karakteristike

Aplikacija će zahtevati od korisnika da se regustruje kako bi mogao da učestvuje u aktivnostima za koje je aplikacija primarno i namenjena.

Korisnici aplikacije mogu se podeliti u tri grupe:

- Običan korisnik
- Administrator

Običan korisnik – glavna uloga ovog korisnika jeste da koristi sistem radi sopstvene koristi ili užitka. Ovom korisniku će biti omogućeno da u širokom spektru igara, pronađe one koje ga interesuju, da izvrši otpremu i testiranje svojih botova, da se prijavi na turnire, aktivno učestvuje u njima, da gleda kako svoje, tako i mečeve u kojima on ne učestvuje.

Administrator – ovom korisniku sistema biće omogućeno sve što i običnom korisniku, ali će pored toga imati i mnoge opcije koje će mu omogućavati održavanje aplikacije u najboljem mogućem stanju. Pored pregleda svih korisnika sistema i mogućnosti upravljanja njihovim nalozima, admin će moći da dodaje i uređuje nove igre, da započinje turnire za već postojeće igre i da vrši objave rezultata odigranih turnira i mečeva.

Funkcije proizvoda

Iako će aplikacija zahtevati da korisnik bude registrovan pre nego što mu se dozvoli pristup konkretnom sadržaju i svrsi aplikacije, neregistrovanom korisniku se ipak pruža određena količina, većinom informativnih podataka o samoj aplikaciji. Podrazumeva se da svim funkcionalnostima kojima neregistrovani korisnici imaju pristup, takođe imaju i svi registrovani korisnici.

Funkcionalnosti koje se pružaju **neregistrovanim** korisnicima:

1. Pregled osnovnih informacija o aplikaciji (šta je ideja, zašto se pridružiti...).
2. Pregled svih postojećih igara, osnovne infromacije o svakoj igri ponaosob.
3. Mogućnost registracije na sistem.
4. Pregled često postavljanih pitanja (FAQ - Frequently Asked Questions).

5. Pregled osnovnih informacija o projektu i razvojnom timu.
6. Mogućnost prevoda teksta aplikacije na dva jezika (srpski i engleski).
7. Mogućnost preuzimanja uputstva za upotrebu aplikacije.

Funkcionalnosti koje se pružaju **registrovanim** i prijavljenim korisnicima:

Običan korisnik

1. Održavanje profila (mogućnost pregleda i izmene svih korisničkih podataka).
2. Mogućnost pregleda i pretraživanja svih igara, kao i svih informacija o njima.
3. Mogućnost otpreme, pregleda, brisanja, izmene botova za neku igru.
4. Testiranje izabranog bota u meču protiv samog sebe.
5. Mogućnosti kreiranja igrača, prijave na turnir i povlačenja prijave.
6. Pregled i praćenje svih turnira za neku igru.
7. Pregled informacija o mečevima i njihovim učesnicima, za svaki aktivan turnir.
8. Gledanje svih odigranih mečeva u realnom vremenu.
9. Praćenje turnira i mečeva od interesa.
10. Pregled i pretraga najuspešnijih korisnika sistema (User leaderboard).
11. Uključene su i sve funkcionalnosti koje ima i neregistrovan korisnik.

Administrator

1. Pregled svih registrovanih korisnika, kao i svih informacija za svakog korisnika pojedinačno. Mogućnost zabrane pristupa sistemu korisnicima. Mogućnost dodavanja novih administratora.
2. Postavljanje novih igara (uključuje otpremu fajlova koji sačinjavaju logiku po kojoj će se odvijati simulacija mečeva te igre, kao i uređivanje pravila igre koja će biti ispisana na stranici te igre). Moguća je i izmena svih podataka već postojećih igara.
3. Definisanje novih turnira (kao i mogućnosti brisanja ili produživanja roka prijave ukoliko za taj turnir nije bilo dovoljno prijavljenih).
4. Objavljivanje rezultata turnira i mečeva putem email-a.
5. Uključene su i sve funkcionalnosti koje imaju obični i neregistrovani korisnici.

Razmatranje dizajna

Pretpostavke

Kako je aplikacija koja će biti razvijena u sklopu ovog projekta u stvari Web aplikacija, svakom korisniku će biti potrebna internet konekcija kako bi mogao da joj pristupi i da je koristi.

Kako bi rad aplikacije bio moguć, neophodno je da na serveru na kojem će se aplikacija nalaziti budu instalirani Node.js i MongoDB, kao i sve propratne biblioteke.

U slučaju problema na Web serveru koji pokreće aplikaciju, bilo da je u pitanju neki kvar u sistemu, nestanak struje ili bilo koja druga nepredviđena okolnost, aplikacija neće biti u funkciji i nijedan korisnik neće moći da joj pristupi, sve dok se kvar u sistemu ne reši od strane lica koja su zadužena za održavanje servera.

Ukoliko se iz bilo kojih razloga, veza sa MongoDB bazom izgubi, povezani korisnici će i dalje moći da koriste sve funkcije aplikacije, osim gledanja mečeva. Ako postoje neki turniri čija je simulacija ‘zakazana’ u vreme kada je izgubljena veza sa MongoDB bazom, njihove će se simulacije pomeriti za neko drugo vreme, a time će biti pomereno i vreme odigravanja celog turnira na frontend-u, o čemu će korisnici biti obavešteni (sve ovo je neophodno jer se tok mečeva na turniru zapisuje upravo u MongoDB bazu, a odatle šalje na klijentsku stranu).

Ukoliko dođe do pada servera koji izvršava simulacije mečeva, doći će do pomeranja ‘zakazanih’ simulacija za neko drugo vreme, čime će se i odigravanje mečeva na frontend-u pomeriti, o čemu će korisnici biti obavešteni u sklopu aplikacije. Pad ovog servera neće imati nikakav drugi uticaj na izvršavanje ostatka aplikacije.

Ograničenja

Aplikaciju je potrebno razviti korišćenjem savremenih Web tehnologija, tako da aplikacija bude potpuno funkcionalna kako na fiksnim (desktop računari), tako i na prenosivim uređajima (mobilni telefoni, tableti).

Klijentski deo aplikacije potrebno je razvijati u Angular 5 framework-u, tako da može da se izvršava na svim standardnim Web pretraživačima (sa posebnim nagliskom na Google Chrome i Mozilla Firefox pretraživače na kojima mora biti u potpunosti funkcionalna). Serverski deo aplikacije treba razvijati u Node.js-u. Vizuelno predstavljanje meča može se uraditi u bilo kom javascript framework-u koji radi sa HTML5 canvas-om (npr. Phaser).

Radno okruženje

Serverski deo aplikacije, kao i baze podataka (MongoDB, SQLite) radiće na IMI serveru, koji radi na operativnom sistemu Linux (Ubuntu).

Klijentski deo aplikacije će raditi kako na fiksnim, tako i na prenosivim uređajima koji imaju instaliran neki od standardnih pretraživača (Chrome, Firefox, Edge).

Arhitektura sistema

U ovom poglavlju biće predstavljen arhitektonski stil koji će biti upotrebljen za izradu sistema, kao i sve komponente od kojih će aplikacija biti izgrađena. Pored toga biće dato i obrazloženje takve strukture.

Pregled sistema

Web aplikacija će imati *troslojnu arhitekturu* i svaka komponenta sistema pripadaće tačno jednom sloju.

Slojevi arhitekture su:

1. Prezentacioni sloj

- Predstavlja grafički korisnički interfejs.
- Pripadaju mu komponente za prikaz elemenata u pretraživaču.
- Čine ga mnogi html, css i javascript fajlovi koji zajedno rade u sklopu Angular aplikacije.

2. Logički sloj

- Predstavlja sloj u kojem se vrše sve odluke koje čine logiku aplikacije. Iz ovog sloja se obavljaju zahtevi za novim podacima sloju za pristup podacima, kao i predaja primljenih podataka komponentama prezentacionog sloja radi odgovarajućeg prikaza.
- Čine ga mnogi typescript fajlovi, koji takođe rade u sklopu Angular aplikacije.

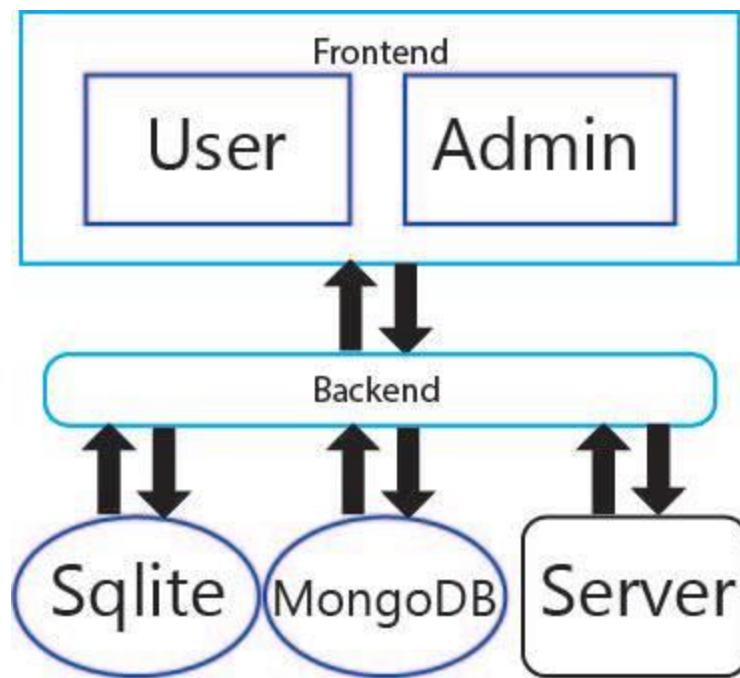
3. Sloj za pristup podacima

- Ovaj sloj ima vezu sa svim bazama podataka aplikacije, tako da se preko njega mogu obavljati sve vrste manipulacije podacima.
 - Čine ga mnoge javascript metode kojima se opslužuju zahtevi, obavljaju razne kalkulacije i manipulacije podacima u bazama, a sve to u sklopu jedne Node.js aplikacije.

Aplikacija će se sastojati iz više povezanih komponenti:

- Front-end aplikacije podeljen je na dva dela, na aplikaciju za običnog korisnika (user) i na aplikaciju za administratora (admin). U zavisnosti od toga koji korisnik se prijavljuje na sistem, biće mu prikazan odgovarajući front-end, sa tim što administrator može u bilo kom trenutku da vrši pregled frontend-a običnog korisnika, dok obrnuto ne važi. Frontend će biti izrađen u Angular 5 tehnologiji.

- SQLite baza podataka u kojoj će se čuvati svi podaci o registrovanim korisnicima, igrama, botovima, turnirima i mečevima.
- MongoDB baza primarno služi za čuvanje podataka o toku svakog odigranog meča. Za tok meča podrazumeva se čuvanje koordinata kretanja svakog objekta u tom meču tokom trajanja celog meča, kao i izmene rezultata na koje to kretanje utiče. Nerelaciona baza podataka je izabrana iz razloga što koordinata objekata, posebno u timskim igrama u trajanju od po nekoliko minuta, može biti pregršt, a nerelacione baze podataka su se pokazale kao veoma efikasne u radu sa velikim količinama podataka.
- Postojaće posebna serverska komponenta čija će glavna uloga biti da vrši simulacije mečeva za koje dobija podatke (igra i igrači oba učesnika) i da nakon završenog posla vrati sve potrebne informacije (u JSON formatu) o toku meča, koje će se zatim smestiti u MongoDB bazu, a jednim delom i u SQLite bazu (samo rezultat meča).
- Serverska logika aplikacije će povezati sve ove komponente i omogućiće upisivanje svih podataka u predviđene baze, kao i čitanje ispravnih podataka i njihovo slanje na klijentsku stranu. Serverska logika će biti izrađena u Node.js tehnologiji.



Slika 1. Arhitektura sistema

Obrazloženje dizajna

Ono u čemu troslojna arhitektura prednjači u odnosu na druge arhitektonске stilove jeste modularnost njenih slojeva. Ona omogućava da se komponente različitih slojeva razvijaju sasvim nezavisno. Omogućava da se čak i nakon izgradnje i povezivanja svih slojeva, izmene u jednom sloju nikako ne odražavaju na druge slojeve; možemo čak i u potpunosti izmeniti tehnologiju u kojoj radi jedan sloj, a da pritom ni na koji način ne prilagodimo rad drugih slojeva, a da ceo sistem ostane u potpunosti funkcionalan. Ovakva modularnost omogućava da aplikacija bude fleksibilna, podložna izmenama i laka za održavanje.

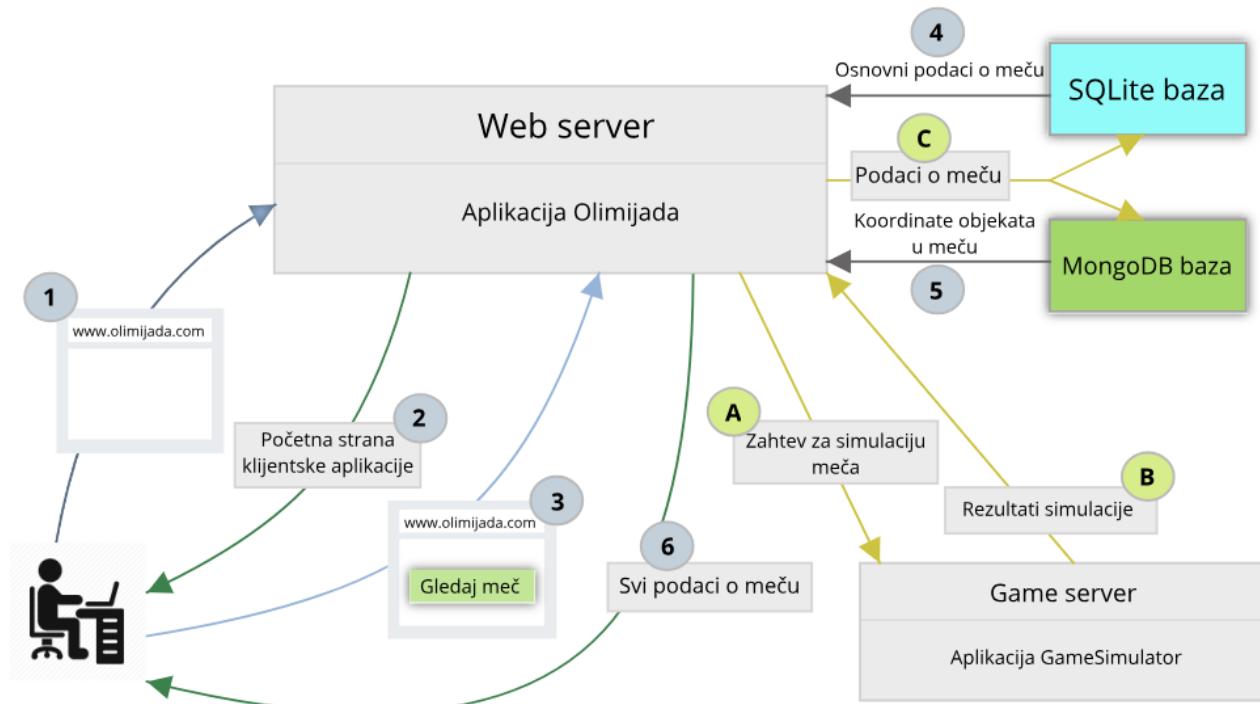
U sistemu će se aplikacija za izvršavanje simulacija mečeva nalaziti na posebnom serveru, odvojenom od Web servera koji će opsluživati korisničke zahteve. Ovakva arhitektura ima više prednosti:

- Kompleksna izračunavanja koja izvršava aplikacija za simulaciju mečeva neće usporavati brzinu odgovora na zahteve glavne aplikacije, a time ni ugrožavati korisnički doživljaj prilikom upotrebe aplikacije. Takođe, sama izračunavanja će se neometano izvršavati u sopstvenom okruženju.
- Ukoliko dođe do nekakvih problema u aplikaciji za simuliranje mečeva, to neće imati za posledicu i pad aplikacije za opsluživanje korisnika. Korisnicima će i dalje biti omogućena upotreba aplikacije, ali će biti obavešteni da je došlo do kvara u sistemu i da se zbog toga mečevi trenutno ne mogu simulirati... Zamolićemo korisnike da strpljivo sačekaju dok se problem ukloni kako bi nastavili sa takmičenjem. Kreiranje novih turnira i prijave ne postojeće turnire biće onemogućene za vreme trajanja kvara, dok će turniri čija je simulacija mečeva zakazana biti privremeno pauzirani.

Dizajn visokog nivoa

U ovom poglavlju je prikazana arhitektura sistema na visokom nivou. Prikazane su komponente iz kojih se sastoji čitav sistem kao i njihova međusobna povezanost.

Konceptualni pogled



Slika 2. Konceptualni dijagram sistema

Svrha konceptualnog pogleda jeste da slikovito objasni od kojih delova se sistem sastoji, kao i da prikaže kako svi ti delovi međusobno komuniciraju (na visokom nivou apstrakcije).

Na slici 2. je data jedna priča na osnovu koje se može zaključiti najopštiji način rada sistema. Redosled operacija je opisan brojevima i slovima. Brojevima je opisan jedan slučaj upotrebe aplikacije od strane korisnika (gledanje izabranog meča), dok je slovima označen proces obavljanja simulacije jednog meča.

Objašnjenje prema numeričkim oznakama:

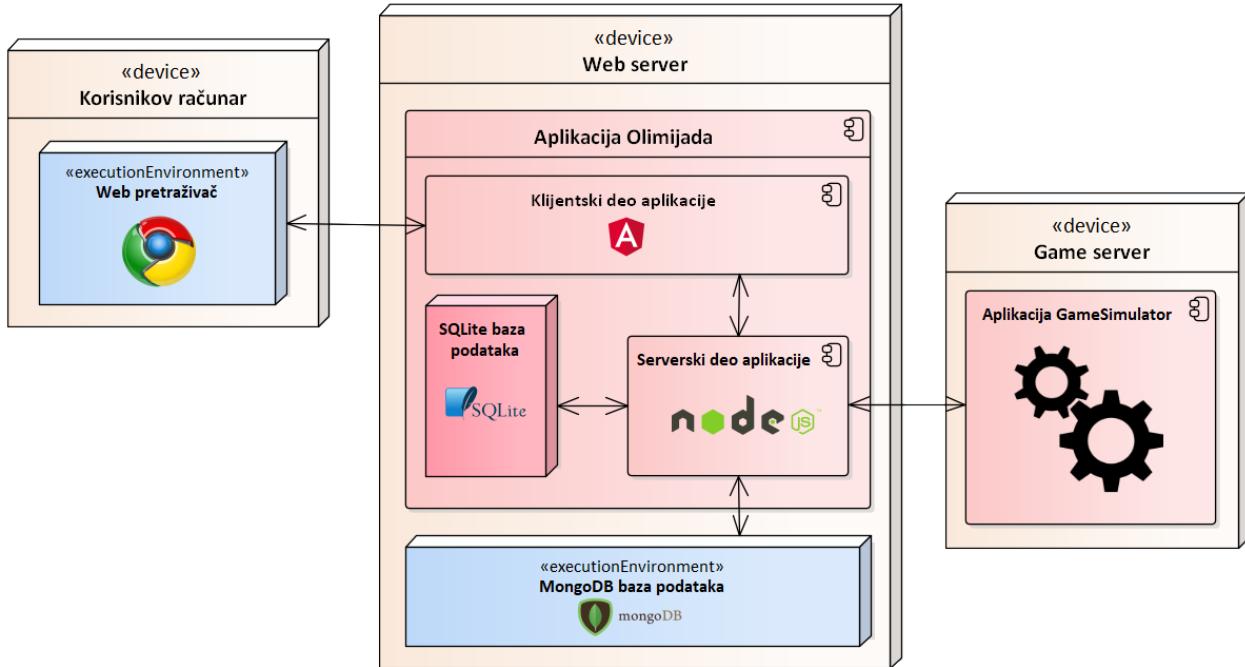
- 1) Korisnik ukucava Web adresu naše aplikacije i time šalje zahtev na Web server koji pokreće aplikaciju Olimijada.

- 2) U serverskoj aplikaciji se vrši obrada tog zahteva i kao odgovor klijentu se vraća index.html fajl, koji čini osnovu svake Angular aplikacije. Nakon prijema odgovora i obrade fajla od strane pretraživača, korisniku bi trebalo da se prikaže početna strana aplikacije.
- 3) Nakon razgledanja aplikacije, korisnik je odlučio da pogleda određeni meč. On klikom na dugme *Gledaj meč* ponovo šalje zahtev serverskoj aplikaciji.
- 4) Na serveru se vrši analiza zahteva iz koje se uviđa koji meč korisnik tačno želi da pogleda. Zatim se iz SQLite baze vrši prikupljanje opštih podataka o tom meču, poput informacija o korisnicima koji su u njemu bili suprotstavljeni, kao i podaci o njihovim timovima i igračima.
- 5) Iz MongoDB baze se vrši preuzimanje podataka o samom toku meča. Tu spadaju koordinate kretanja za sve igrače koji su učestvovali u tom meču, kao i niz vrednosti koji predstavlja izmene u rezultatu meča tokom vremena.
- 6) Sve te informacije o meču se sa servera šalju nazad klijentu. Na klijentovoј strani se primljeni podaci obrađuju i konačno koriste za vizuelizaciju meča.

Objašnjenje prema slovnim oznakama:

- A. Na slici 2. je napisano da se serveru šalje zahtev za simulacijom direktno iz serverske aplikacije, bez uticaja korisnika. Taj zahtev se zapravo generiše automatski, u vreme koje je korisnik sam izabrao kao pogodno. U samom zahtevu će se nalaziti fajlovi koji opisuju logiku igre za koju se meč igra, kao i fajlovi logike svih igrača prijavljenih za taj meč.
- B. Nakon obavljene simulacije, aplikaciji Olimijada se šalju rezultati te simulacije, koji sadrže sve potrebne podatke o toku meča.
- C. Serverska aplikacija Olimijada obrađuje primljene podatke, i u MongoDB bazu upisuje sve podatke koji se tiču samog toka meča, dok u SQLite bazu upisuje podatke o rezultatu i pobedniku meča. Korisnici koji su učestvovali u tom meču bivaju obavešteni o njegovom ishodu putem email-a i preporučuje im se da pogledaju rezultate na našem sajtu za više detalja.

Fizički pogled



Slika 3. Fizički prikaz sistema (deployment diagram)

Na slici 3. je dat fizički prikaz sistema, odnosno prikaz svih komponenti od kojih će se sistem sastojati i način na koji su one povezane.

Sa korisničke strane, zahtevi su da korisnikov računar ima pristup Internetu i da ima instaliran neki od standardnih Web pretraživača (preporučeno Google Chrome ili Mozilla Firefox), kako bi mogao da pokrene aplikaciju.

Postojaće jedan Web server na kojem će biti smešten klijentski, i pokrenut serverski deo aplikacije *Olimijada*. Na tom Web serveru je neophodno da bude instaliran Node.js i da bude instalirana i startovana MongoDB baza. Serverski deo aplikacije *Olimijada* će koristiti i jednu SQLite bazu podataka koja, strukturno gledano, predstavlja samo jedan fajl, tako da za nju nije potreban serverski proces.

Na Game serveru će raditi aplikacija *GameSimulator*, čija je glavna uloga u ovom sistemu da izvršava simulacije mečeva. Za simulacije će koristiti ulazne podatke koje dobija putem veze sa serverskim delom aplikacije *Olimijada*, a rezultate obavljenih simulacija će vraćati istom tom vezom, nazad ka aplikaciji *Olimijada*, koja će dobijene podatke da obradi i sačuva u bazama podataka.

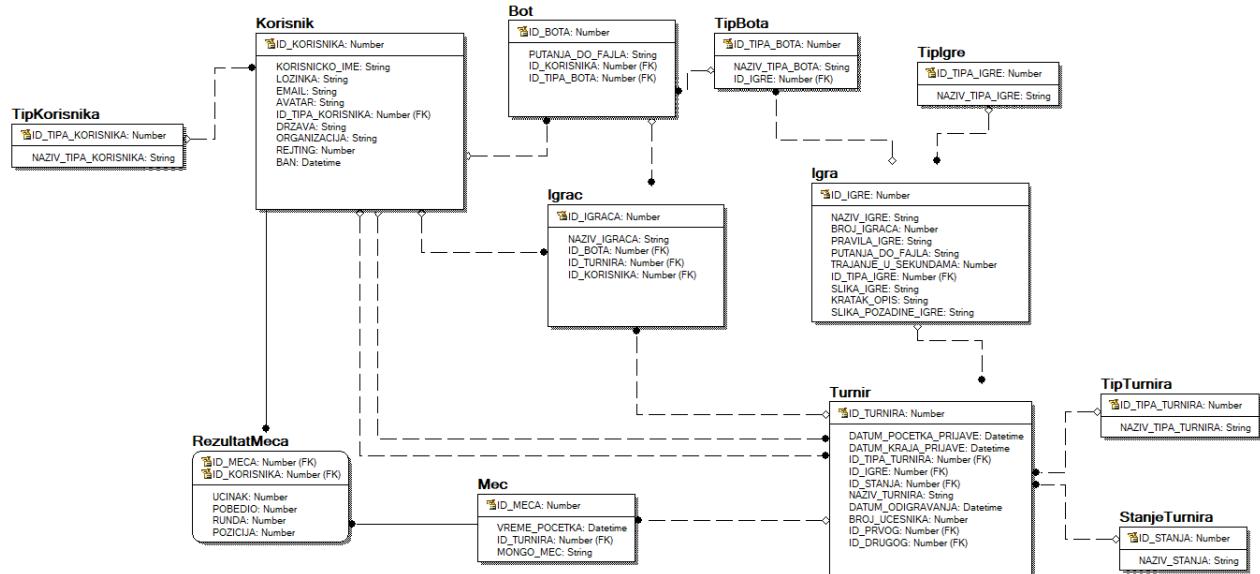
Dizajn niskog nivoa

Dizajn baza podataka

Aplikacija će koristiti dve različite baze za skladištenje podataka – jednu relacionu SQLite bazu i jednu nerelacionu MongoDB bazu podataka. Razlog za upotrebu dva tipa baza podataka jeste u tome što se u samom sistemu istovremeno javljaju podaci sa konzistentnim formatom i strukturom, kao i mnogobrojni podaci koji mogu imati raznolike forme koje jedna relaciona baza teško može, ili uz velike gubitke u performansama, da isprati.

Model SQLite baze podataka

U nastavku je dat model relacione (SQLite) baze podataka, sa opisom svih njenih tabela i atributa:



Slika 4. Model SQLite baze podataka sa relacijama između tabela

| Korisnik | | |
|----------------|--------------|-------------------------------------|
| Naziv atributa | Tip podataka | Opis |
| id_korisnika | Number | Jedinstveni identifikator korisnika |
| korisnicko_ime | String | Korisničko ime |
| lozinka | String | Enkriptovana šifra korisnika |
| email | String | Email korisnika |

| | | |
|-------------------|----------|--|
| avatar | String | Putanja do slike korisnika na serveru. |
| id_tipa_korisnika | Number | Tip korisnika |
| drzava | String | Država porekla korisnika |
| organizacija | String | Organizacija korisnika (firma, škola, fakultet...) |
| rejting | Number | Rejting korisnika. Stiče se osvajanjem turnira. |
| ban | DateTime | Datum do kog je korisnik banovan. Popunjeno jedino u slučaju da je korisnik banovan. |

| TipKorisnika | | |
|----------------------|--------------|---|
| Naziv atributa | Tip podataka | Opis |
| id_tipa_korisnika | Number | Jedinstveni identifikator tipa korisnika |
| naziv_tipa_korisnika | String | Naziv tipa korisnika (Administrator, običan korisnik) |

| Bot | | |
|------------------|--------------|----------------------------------|
| Naziv atributa | Tip podataka | Opis |
| id_bota | Number | Jedinstveni identifikator bota |
| putanja_do_fajla | String | Putanja do fajla bota na serveru |
| id_korisnika | Number | Korisnik kome pripada bot |
| id_tipa_bota | Number | Tip bota |

| TipBota | | |
|-----------------|--------------|---------------------------------------|
| Naziv atributa | Tip podataka | Opis |
| id_tipa_bota | Number | Jedinstveni identifikator tipa bota |
| naziv_tipa_bota | String | Naziv tipa bota (definisani su igrom) |
| id_igre | Number | Igra |

| Igrac | | |
|----------------|--------------|----------------------------------|
| Naziv atributa | Tip podataka | Opis |
| id_igraca | Number | Jedinstveni identifikator igrača |
| naziv_igraca | String | Naziv igrača |
| id_bota | Number | Bot |
| id_turnira | Number | Turnir |
| id_korisnika | Number | Korisnik |

| Igra | | |
|----------------------|--------------|--|
| Naziv atributa | Tip podataka | Opis |
| id_igre | Number | Jedinstveni identifikator igre |
| naziv_igre | String | Naziv igre |
| broj_igraca | Number | Broj igrača u timu |
| pravila_igre_en | String | Pravila igre na engleskom jeziku |
| pravila_igre_srb | String | Pravila igre na srpskom jeziku |
| putanja_do_fajla | String | Putanja do fajla igre na serveru |
| trajanje_u_sekundama | Number | Prosečno vreme trajanja igre u sekundama |
| id_tipa_igre | Number | Tip igre |
| slika_igre | String | Putanja do slike igre na serveru |
| kratak_opis_en | String | Kratak opis igre na engleskom jeziku |
| kratak_opis_srb | String | Kratak opis igre na srpskom jeziku |
| slika_pozadine_igre | String | Putanja do slike za pozadinu igre na serveru |

| TipIgre | | |
|---------------------|--------------|--|
| Naziv atributa | Tip podataka | Opis |
| id_tipa_igre | Number | Jedinstveni identifikator tipa igre |
| naziv_tipi_igre_en | String | Naziv tipa igre na engleskom jeziku (Može biti npr. Action, Strategy, Blocks...) |
| naziv_tipi_igre_srb | String | Naziv tipa igre na srpskom jeziku (Akcija, Strategija,...) |

| TipTurnira | | |
|--------------------|--------------|--|
| Naziv atributa | Tip podataka | Opis |
| id_tipa_turnira | Number | Jedinstveni identifikator tipa turnira |
| naziv_tipi_turnira | String | Naziv tipa turnira (Kup ili Liga) |

| StanjeTurnira | | |
|----------------|--------------|---|
| Naziv atributa | Tip podataka | Opis |
| id_stanja | Number | Jedinstveni identifikator stanja turnira |
| naziv_stanja | String | Naziv stanja turnira (može biti da je turnir još uvek u fazi prijava, zatim da su prijave završene, ali turnir još uvek nije odigran, i na kraju da je turnir odigran). |

| Turnir | | |
|--------|--|--|
|--------|--|--|

| Naziv atributa | Tip podataka | Opis |
|-----------------------|--------------|---|
| id_turnira | Number | Jedinstveni identifikator turnira |
| naziv_turnira | String | Naziv turnira |
| datum_pocetka_prijave | DateTime | Datum i vreme početka prijave za turnir |
| datum_kraja_prijave | DateTime | Datum i vreme kraja prijave za turnir |
| id_tipa_turnira | Number | Jedinstveni identifikator tipa turnira |
| id_stanja | Number | Stanje turnira |
| datum_odigravanja | DateTime | Datum i vreme početka odigravanja turnira |
| broj_ucesnika | Number | Broj korisnika koji se mogu prijaviti na turnir |
| id_prvog | Number | Id korisnika koji je osvojio prvo mesto na turniru |
| id_drugog | Number | Id korisnika koje je osvojio drugo mesto na turniru |

| Mec | | |
|----------------|--------------|--------------------------------|
| Naziv atributa | Tip podataka | Opis |
| id_meca | Number | Jedinstveni identifikator meča |
| vreme_pocetka | DateTime | Datum i vreme početka meča |
| id_turnira | Number | Turnir |

| RezultatMeca | | |
|----------------|--------------|---|
| Naziv atributa | Tip podataka | Opis |
| id_meca | Number | Meč |
| id_korisnika | String | Korisnik |
| ucinak | Number | Postignut učinak korisnika u meču |
| pobedio | Number | Označava pobjednika meča: 1-pobedio 0-ako i drugi učesnik meča ima 0 onda je nerešeno, inače je izgubio |
| runda | Number | Runda turnira u kojoj je odigran meč |

Šeme MongoDB baze podataka

Jedna od odlika nerelacionih baza podataka jeste ta što one ne moraju imati strogo definisanu strukturu poput relacionih baza. MongoDB je jedna nerelaciona baza podataka, i u njoj se podaci čuvaju po kolekcijama. Kolekcija predstavlja kontejner za dokumente, koji predstavljaju ništa drugo, nego JSON objekte, koji mogu imati bilo

kakvu strukturu. Međutim, ovim JSON objektima se, pomoću šema, može definisati određena struktura, pre svega radi lakše obrade ovakvih objekata u serverskoj aplikaciji.

U našoj MongoDB bazi postojaće dve kolekcije:

- KretanjeObjekataMeca - čiji će dokumenti čuvati koordinate kretanja igrača tokom trajanja meča.
- RezultatiMeca – čiji će dokumenti čuvati podatke o izmenama glavnog rezultata tokom trajanja meča.

Šema dokumenata iz kolekcije KretanjeObjekataMeca:

```
{  
    id_meca: Number,  
    ucesnici:  
    [{  
        id_korisnika: Number,  
        igraci:  
        [{  
            id_igraca: Number,  
            koordinate: [{x: Number, y: Number}]  
        }]  
    }]  
}
```

Šema dokumenata iz kolekcije RezultatiMeca:

```
{  
    id_meca: Number,  
    ucesnici:  
    {[  
        id_korisnika: Number,  
        rezultati: [Number]  
    ]}  
}
```

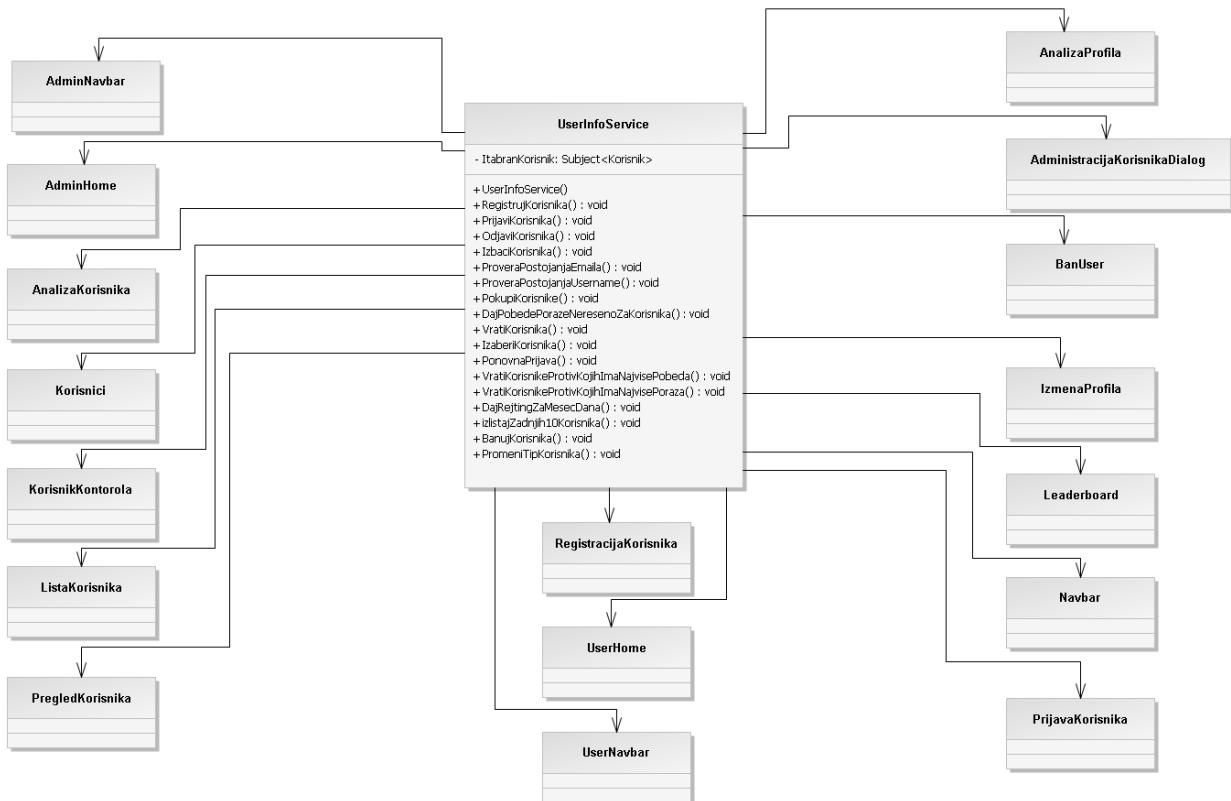
Bitno je napomenuti da ove šeme **ne ograničavaju** bazu tako da se unos podataka mora obaviti u jednom od ova dva formata. One služe da olakšaju rad sa objektima koji ispunjavaju ove formate, dok će se aplikacija prilagođavati za rad sa objektima koji ove strukturne formate ne ispunjavaju.

Dijagrami klase

Klijentski deo

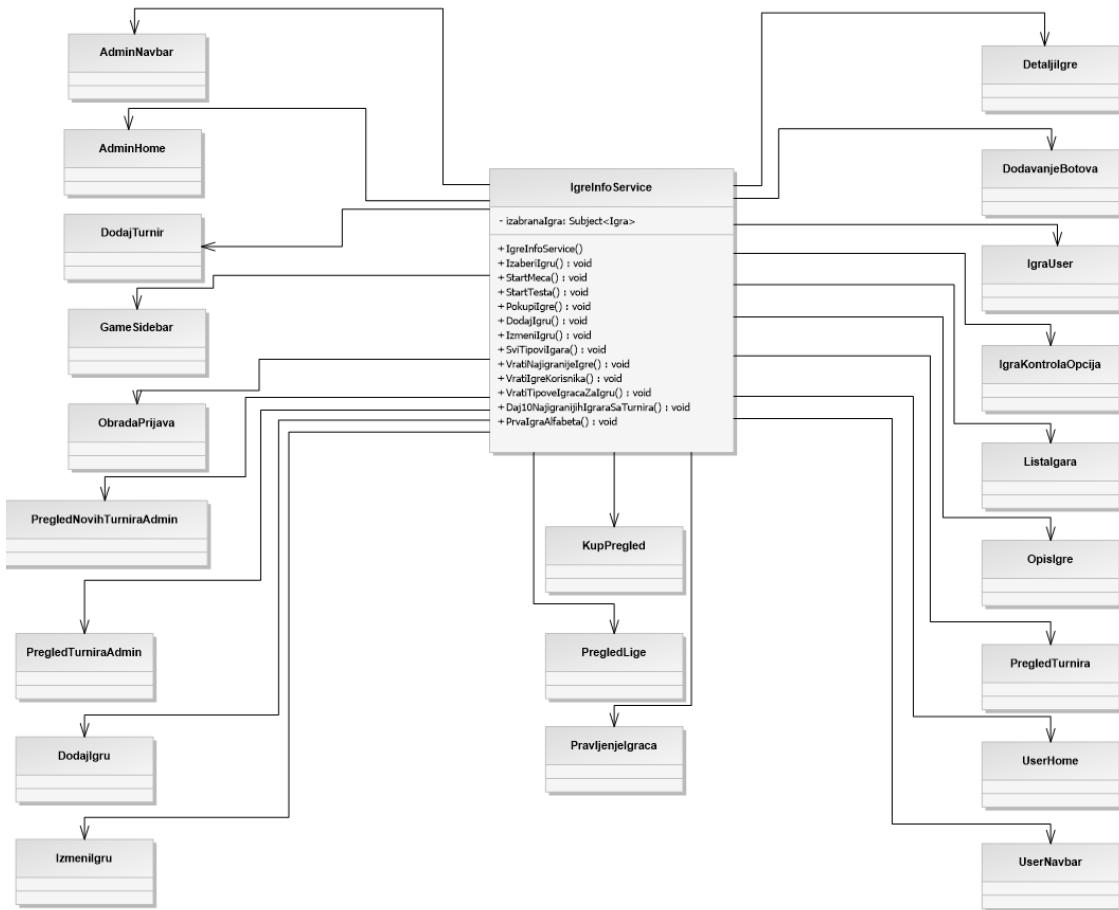
Klijentska strana aplikacije je izgrađena iz više funkcionalno različitih delova. Postoje *komponente* koje u sebi sadrže strukturu (html), izgled (css) i logiku (typescript) nekog dela aplikacije. Postoje *servisi*, čija je najčešća uloga da na zahtev komponenti šalju zahteve za podacima backend-u aplikacije, i da primljene podatke prosleđuju komponentama, koje te podatke koriste za prikaz. Pored uloge medijatora između komponenti i backend-a aplikacije, servisi se koriste i za povezivanje više komponenti koje za ispunjavanje nekog cilja moraju da koriste iste podatke. Ti se podaci mogu čuvati u servisima.

U ovom odeljku biće navedeni servisi koje će aplikacija imati, zajedno sa komponentama koje će te servise koristiti.



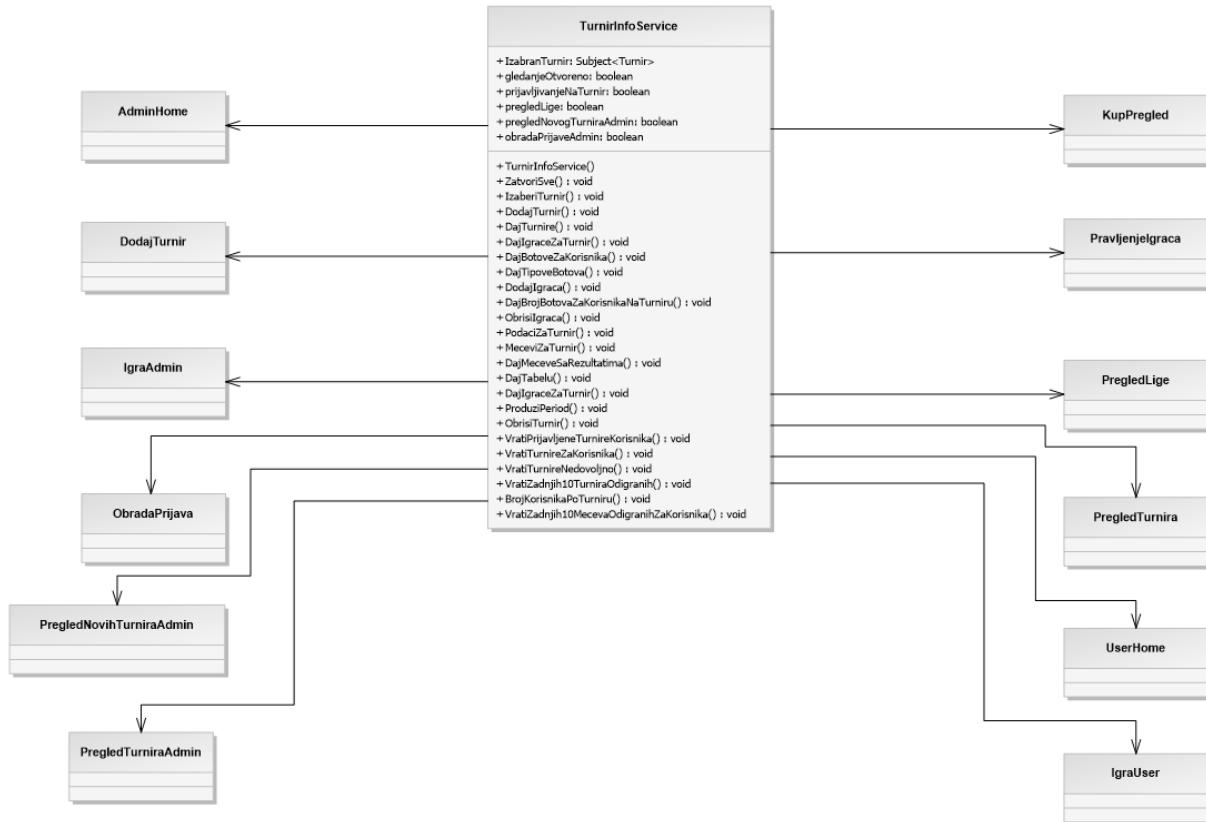
Slika 5. *UserInfo* servis i komponente koje ga koriste

UserInfo servis sadrži metode za rad sa podacima o korisnicima. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Neki od primera upotrebe ovog servisa jesu: registracija, prijava korisnika, prikaz liste svih korisnika, prikaz podataka o svakom korisniku ponaosob, prikaz leaderboard-a.



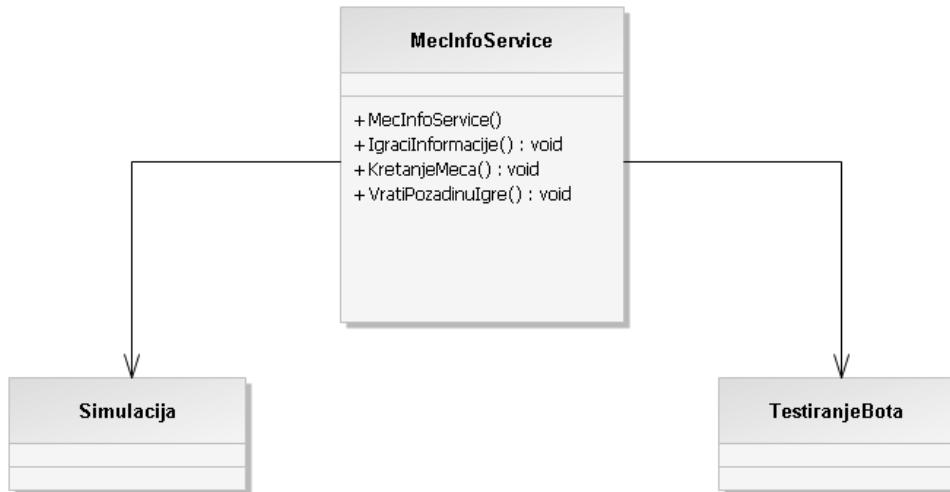
Slika 6. *IgreInfo* servis i komponente koje ga koriste

Kada su u pitanju podaci o igrama, uvek će se koristiti *IgreInfo* servis. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Ovaj servis se koristi pri bilo kakvoj manipulaciji sa podacima igara (listing igara, izmena, dodavanje,...).



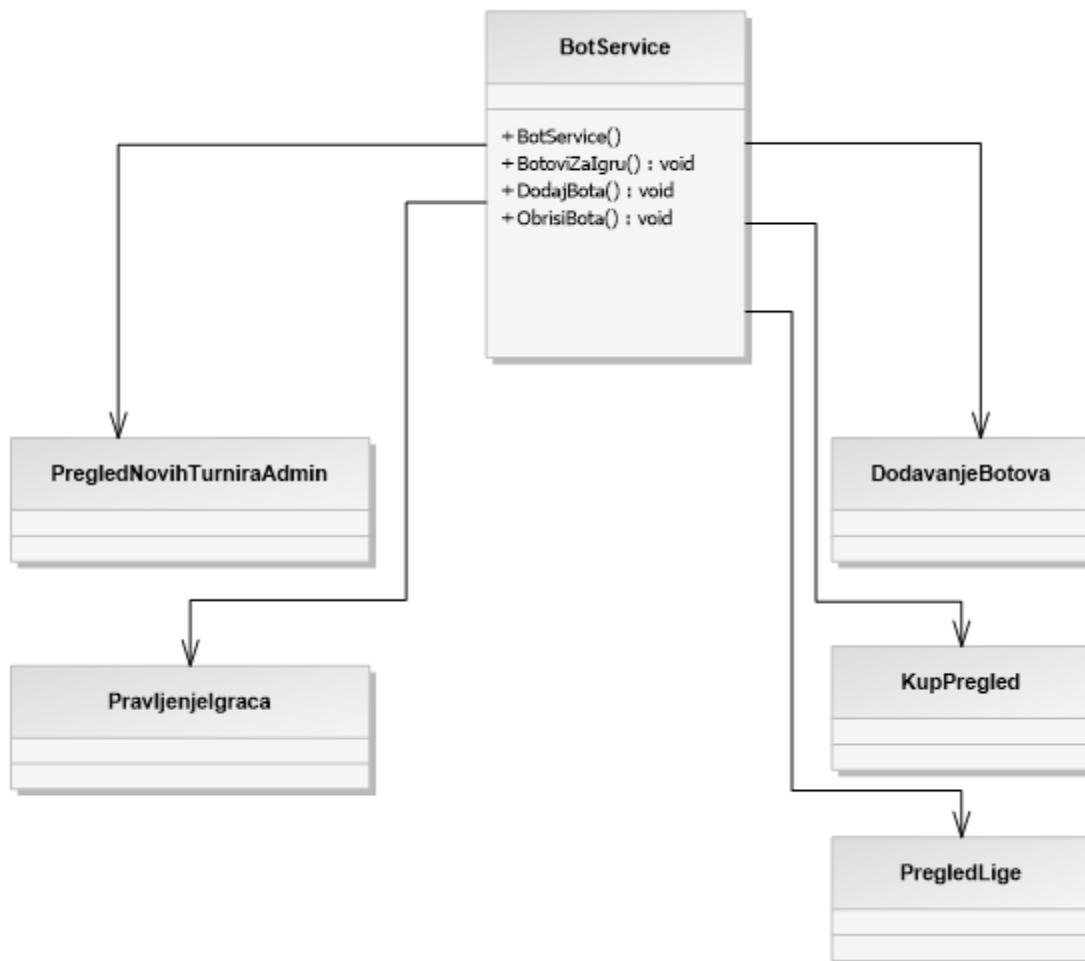
Slika 7. TurnirInfo servis i komponente koje ga koriste

TurnirInfo servis sadrži metode za rad sa podacima o turnirima. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Neki od primera upotrebe ovog servisa jesu: pregled turnira, pregled mečeva za ligu, pregled mečeva za kup, kreiranje turnira, prijava na turnir,...



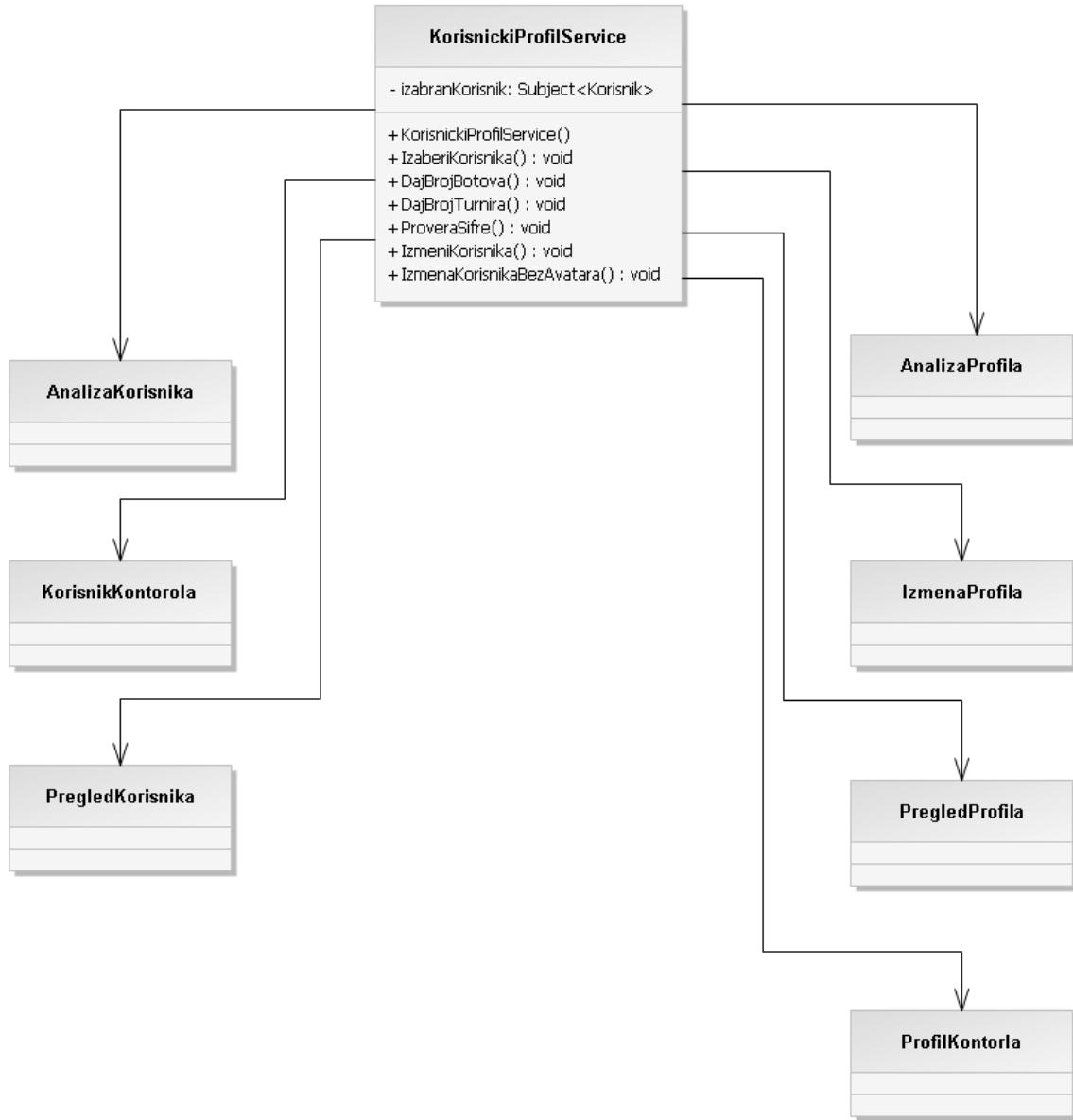
Slika 8. MecInfo servis i komponente koje ga koriste

MecInfo servis sadrži metode za rad sa podacima o mečevima. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Ovaj servis se koristi pri povlačenju informacija o meču prilikom vizuelizacije meča i prilikom testiranja bota.



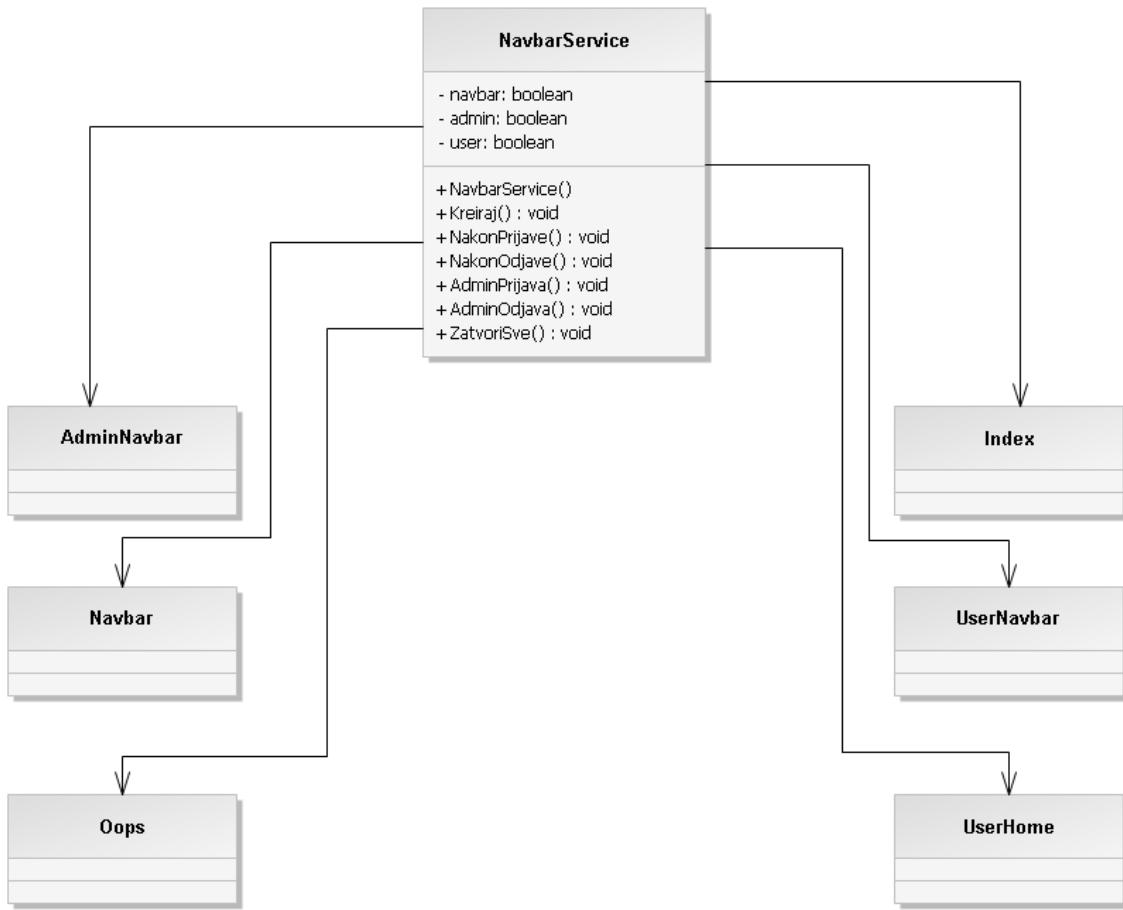
Slika 9. Bot servis i komponente koje ga koriste

Bot servis sadrži metode za rad sa podacima o botovima. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Ovaj servis se koristi pri upload-u i brisanju botova, prijavi na turnir i pregledu mečeva turnira.



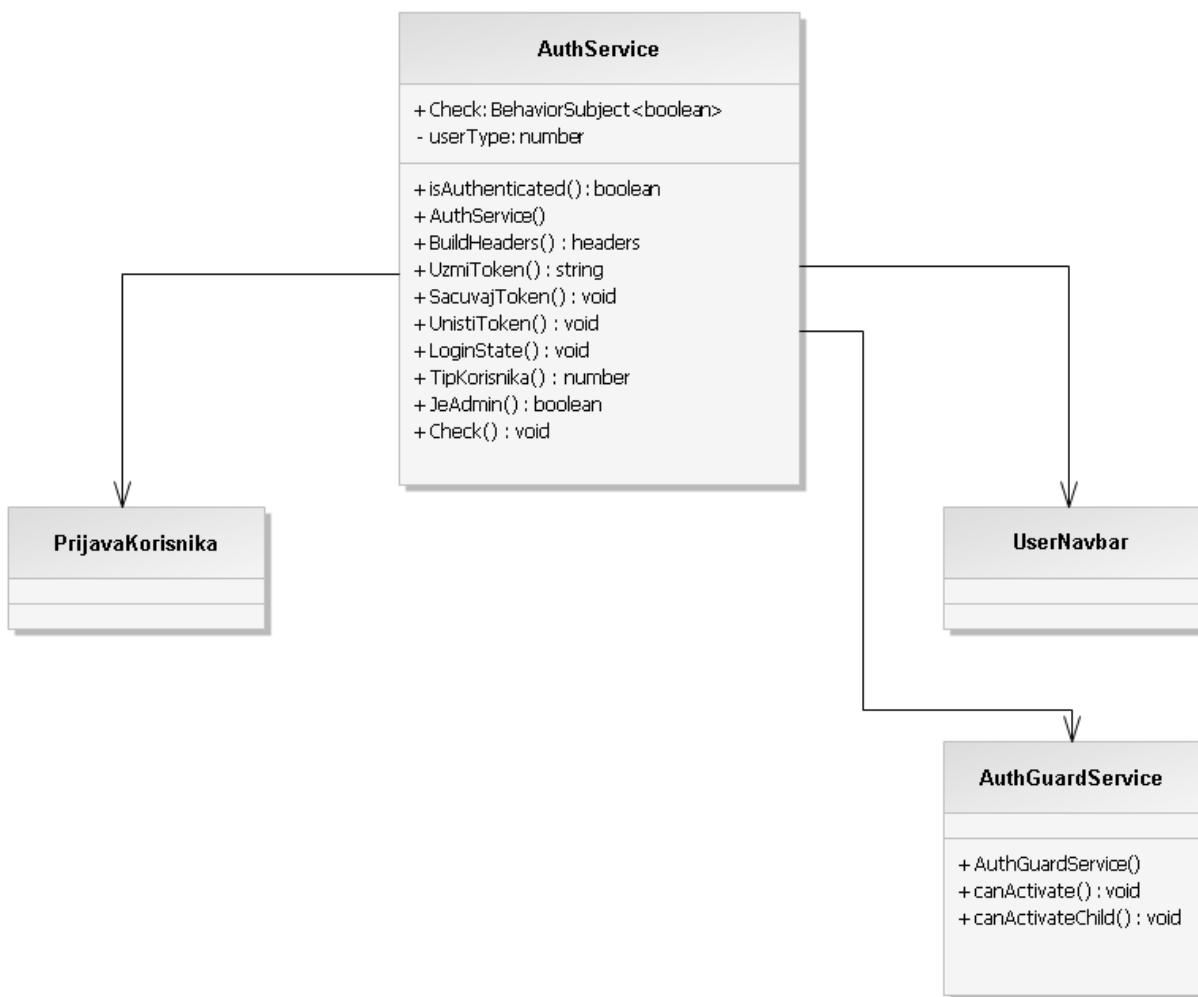
Slika 10. *KorisnickiProfil* servis i komponente koje ga koriste

KorisnickiProfil servis sadrži metode za rad sa podacima koji se pojavljuju na profilnim stranama. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Ovaj servis se koristi pri izmeni profila, pregledu informacija na profilu, pregledu analiziranih korisničkih podataka (statistika)...



Slika 11. Navbar servis i komponente koje ga koriste

Navbar servis sadrži metode za rad sa izmenama navigacije u aplikaciji. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Bilo je reči o tome kako se servisi ne koriste samo za komunikaciju između komponenti i servera, već i za komunikaciju između samih komponenti. Ovaj servis je primer takve upotrebe. Naime, u aplikaciji se često zbog različitih vrsta korisnika vrši smena navigacionih barova, kao i sakrivanje određenih delova navigacije dok se korisnik nalazi na određenim delovima aplikacije. Ovaj servis omogućava baš takvu vrstu manipulacije.

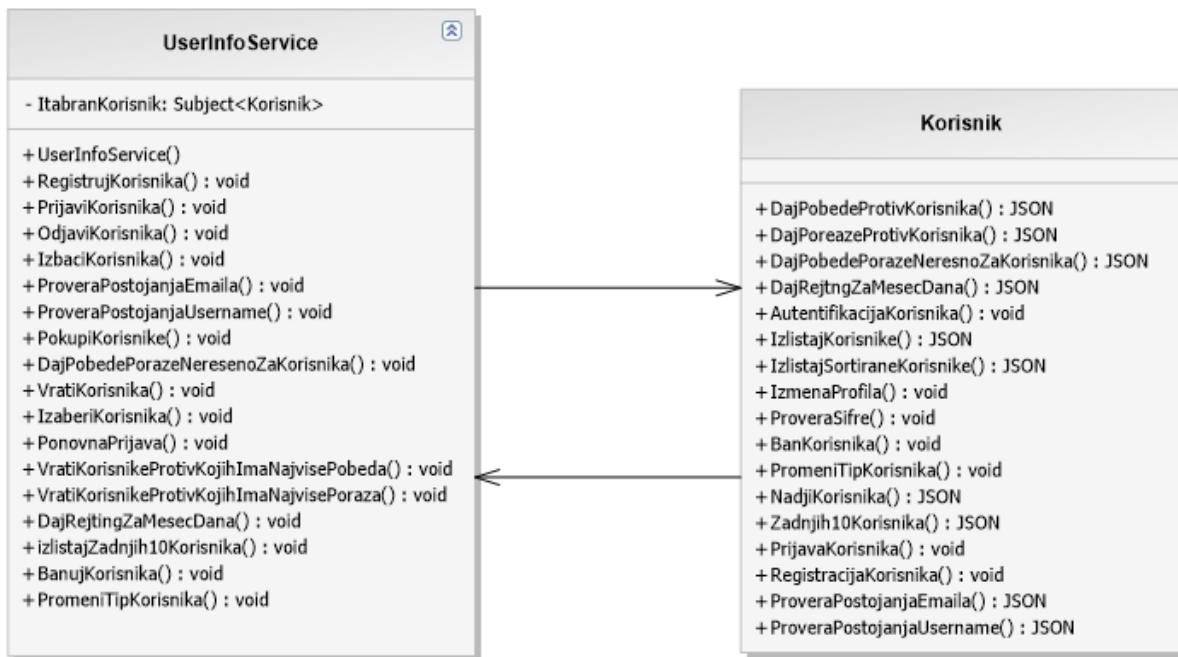


Slika 12. Auth servis i komponente koje ga koriste

Auth servis sadrži metode za rad sa svim sigurnosnim proverama koje se obavljaju na frontend-u. Sve komponente prikazane na dijagramu koriste neke od metoda ovog servisa. Glavne funkcije ovog servisa su: provera da li je korisnik prijavljen, provera da li je korisnik administrator, manipulacije sa web token-om, upravljanje pristupom... Glavna funkcija ovog servisa, koja se postiže u njenom zajedničkom radu sa *AuthGuard* servisom, jeste provera da li korisnik ima pravo pristupa sadržaju koji zahteva. To se obavlja putem `Check()` metode, koja vrši serversku proveru identiteta i dekodiranje poslatog tokena. Ukoliko se ispostavi da korisnik nema pristup, izvršiće se redirekcija korisnika i njegovo obaveštavanje prikladnim porukama. Ova provera se vrši pri svakom korisnikovom zahtevu i time obezbeđuje zaštitu od neautorizovanog pristupa aplikaciji (npr. da običan korisnik pokuša da pristupi stranama za administratora).

Serverski deo

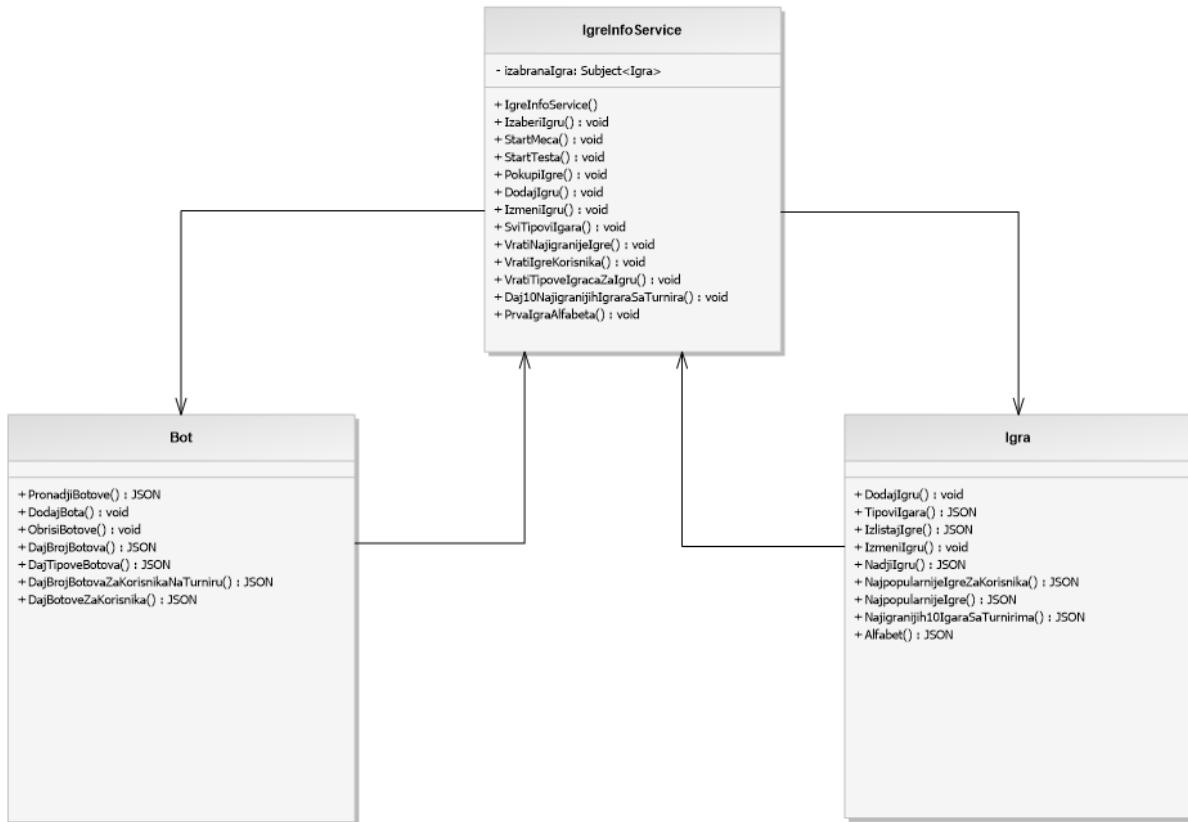
Serverski deo aplikacije će se sastojati od skripti napisanih u javascript jeziku, koje će obavljati sve zahteve na serveru. Skripte su podeljene po svojim oblastima namene na određene klase koje će u ovom odeljku biti opisane. Za svaki servis



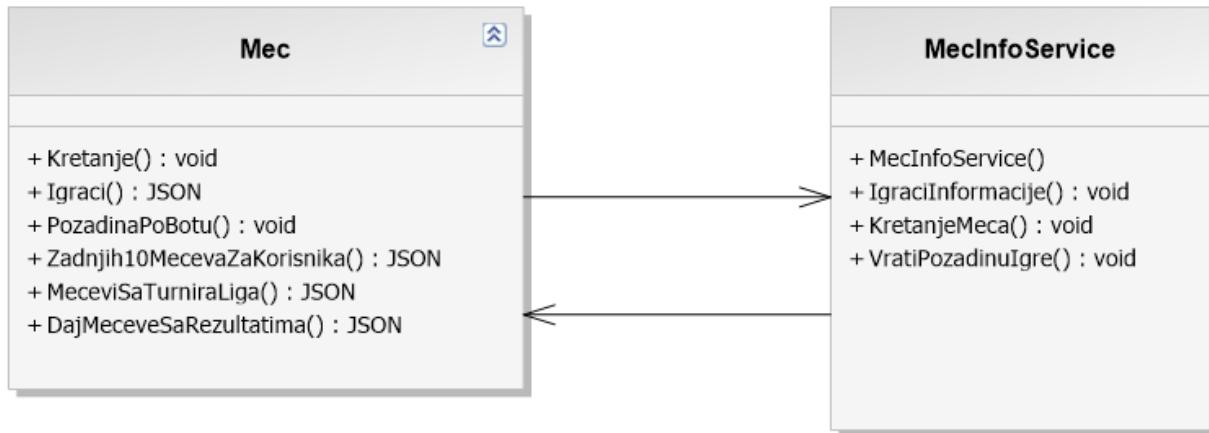
Slika 13. Veza između klase *Korisnik* na serverskoj strani i *Userinfo* servis na klijentskoj strani

U prethodnom odeljku koji se bavio klasama na klijentskoj strani bilo je reči o tome kako servisi igraju ulogu medijatora između komponenti i servera. Na slici 13. je prikazana veza između servisa i klase na serverskoj strani, kao i metode koje ta klasa ima. Svaka od datih metoda obavlja neku vrstu manipulacije sa podacima, koja je potrebna kako bi se odgovorilo na zahtev sa klijenta. Može se lako uočiti da je povratna vrednost ovih metoda, ukoliko ona postoji, u stvari jedan JSON objekat. Dakle, komunikacija između serverskog i klijentskog dela aplikacije obavljaće se pomoću JSON formata.

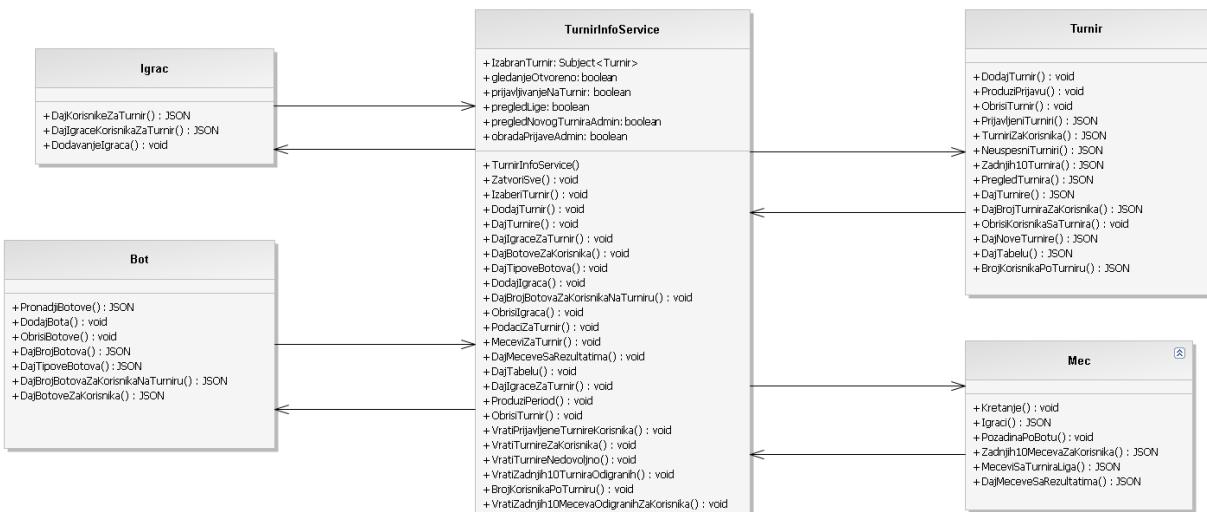
Važno je još napomenuti da klase na serverskom delu aplikacije ni na koji način ne zavise od implementacije servisa na klijentskoj strani, i obrnuto. Svrha ovih dijagrama jeste da se pokaže koje klase i koji servisi treba da međusobno komuniciraju, dok je njihov način implementacije sasvim nezavisan za obe strane aplikacije (front i back).



Slika 14. IgreInfo servis i klase koje koristi na backend-u



Slika 15. Klasa Mec koja opslužuje zahteve MecInfo servisa



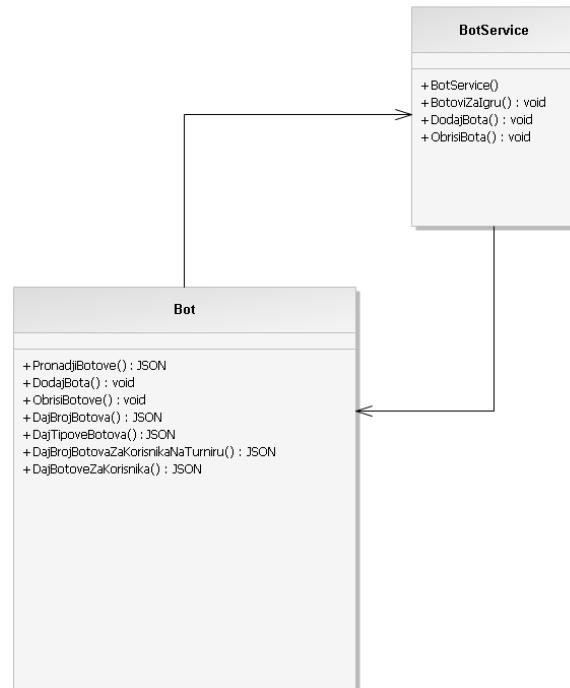
Slika 16. Servis TurnirInfo i klase koje koristi na backend-u



Slika 17. Servis KorisnickiProfil i klase koje koristi na backend-u



Slika 18. Servis Auth koristi podatke korisnika za autentifikaciju preko klase Korisnik



Slika 19. Bot servis koristi klasu Bot za pristup podacima o botovima

Dijagrami slučajeva korišćenja

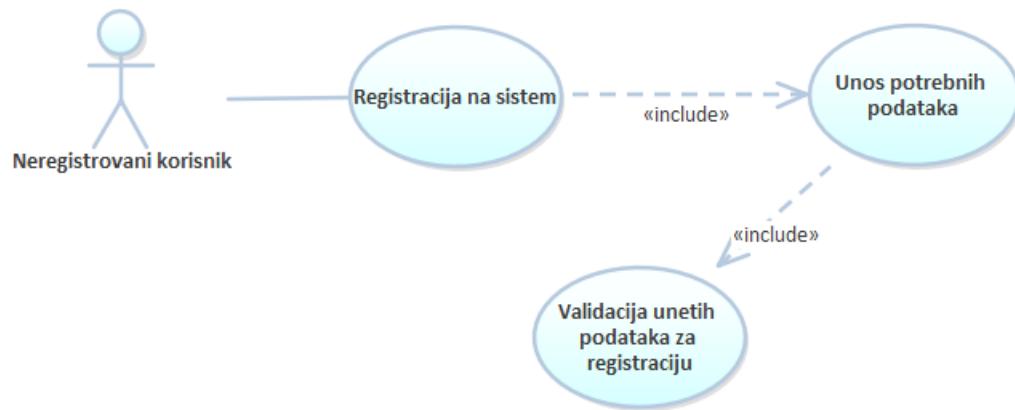
Na slici su prikazani opšti slučajevi korišćenja aplikacije Olimijada. U nastavku teksta će svaki od slučajeva korišćenja biti detaljno objašnjen.

Dijagrami slučajeva korišćenja za neregistrovanog korisnika



Slika 20. Opšti slučajevi korišćenja za neregistrovanog korisnika

Slučaj korišćenja: Registrovanje na sistem



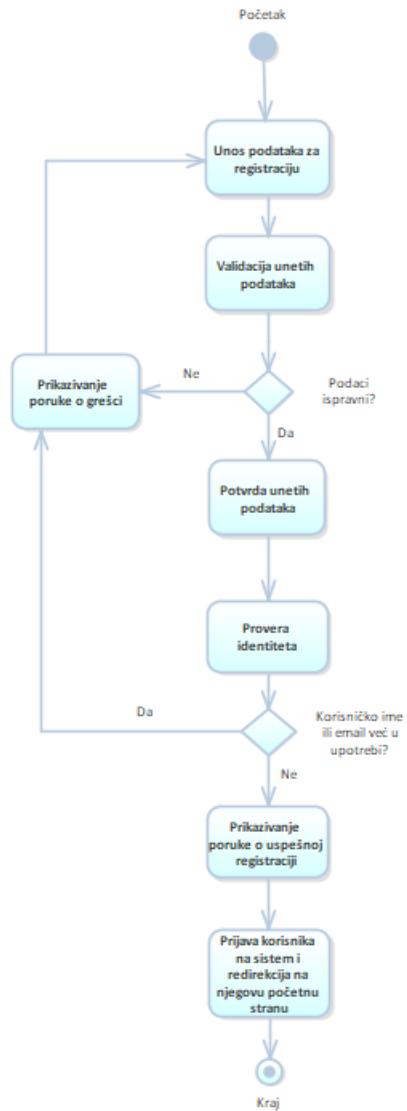
Slika 21. Slučaj korišćenja: Registracija

Kratak opis: Mogućnost registracije neregistrovanog korisnika.

Opis slučaja korišćenja:

Korisnik će imati mogućnost pravljenja korisničkog naloga. Nakon klika na polje „Registruj se“ u glavnom meniju otvorice se forma za unos podataka. Korisnik mora da unese ime, prezime, elektronsku poštu i lozinku u ispravnom formatu, u suprotnom će porukom ispod polja za unos biti obavešten o grešci. Ukoliko je sve podatke uneo ispravno i registracija prođe uspešno, korisnik će biti obavešten porukom. Ukoliko korisnički nalog već postoji korisnik će biti obavešten porukom sa mogučnošću ponovnog registrovanja.

Glavni tok događaja:



Slika 22. Dijagram aktivnosti registracije korisnika

Definisanje uslova:

Pred-uslov: Startovanje aplikacije.

Post-uslov: Regularnost baze podataka.

Alternativni tokovi događaja:

Ukoliko prilikom registracije korisnik unese korisničko ime koje već postoji, neće biti u mogućnosti da kreira nalog. Ukoliko prilikom registracije korisnik unese email koji je već u upotrebi, takođe neće biti u mogućnosti da kreira nalog.

Zahtevi:

Potrebno je korisnika obavestiti ukoliko unese već postojeće korisničko ime ili već upotrebljeni email.

Slučaj korišćenja: Prijava na sistem

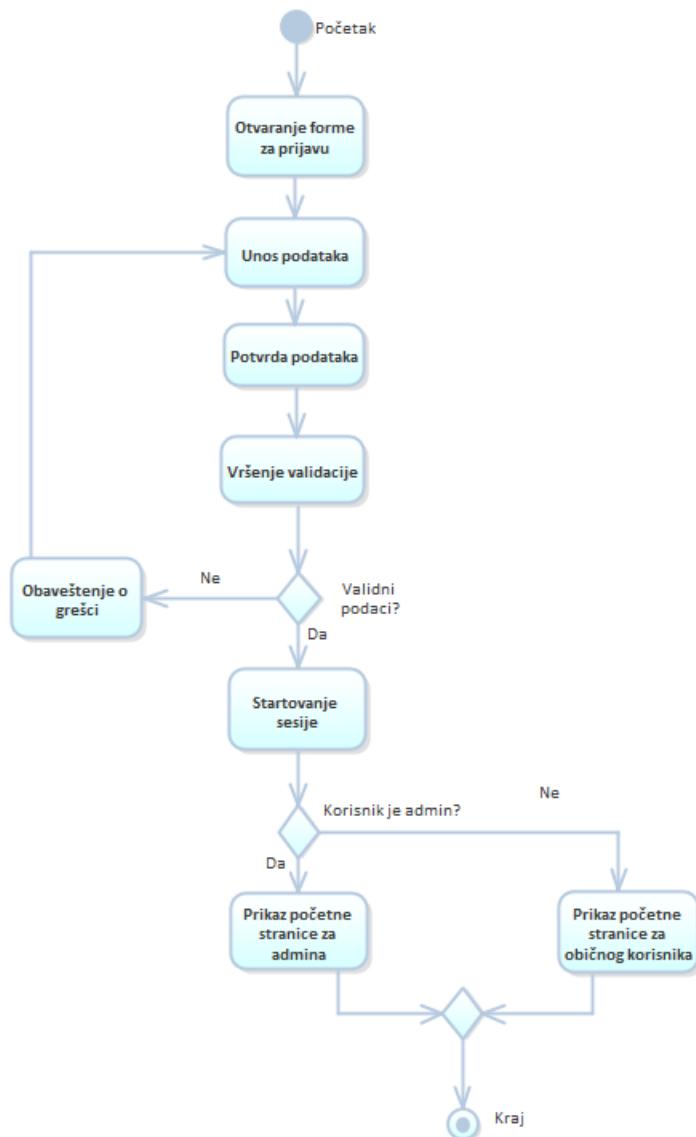
Kratak opis: Prijavljanje korisnika (ili administratora) na sistem.

Opis slučaja korišćenja:

Prilikom pokretanja aplikacije otvara se forma za prijavljivanje korisnika na sistem. Neophodno je uneti korisničko ime/email i šifru. Ukoliko kombinacija korisničkog imena i šifre nije validna pojavljuje se upozorenje i omogućava se korisniku da se ponovo prijavi na sistem. Ukoliko je kombinacija validna otvara se početna strana tog korisnika.

Glavni tok događaja:

Korisnik pokreće web aplikaciju Olimijada. Pojavljuje se forma za prijavljivanje na sistem. Nakon što korisnik unese korisničko ime/email i šifru, vrši se provera validnosti unetih podataka. Ukoliko su podaci nevalidni, pojavljuje se poruka o tome i korisniku se omogućava ponovno prijavljivanje na sistem. Ako su uneti podaci validni pojavljuje se početna stranica.



Slika 23. Dijagram aktivnosti prijave

Definisanje uslova:

Pred-uslov: Startovanje aplikacije.

Post-uslov: Regularnost baze podataka.

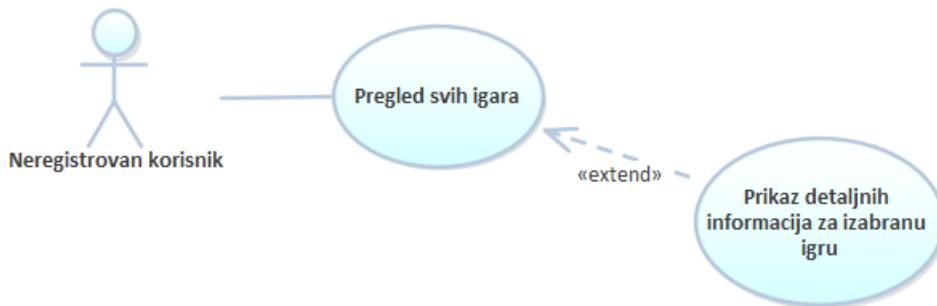
Aleternativni tokovi događaja:

Ukoliko korisničko ime i šifra nisu validni korisnik dobija upozorenje i pruža mu se mogućnost ponovnog prijavljivanja.

Zahtevi:

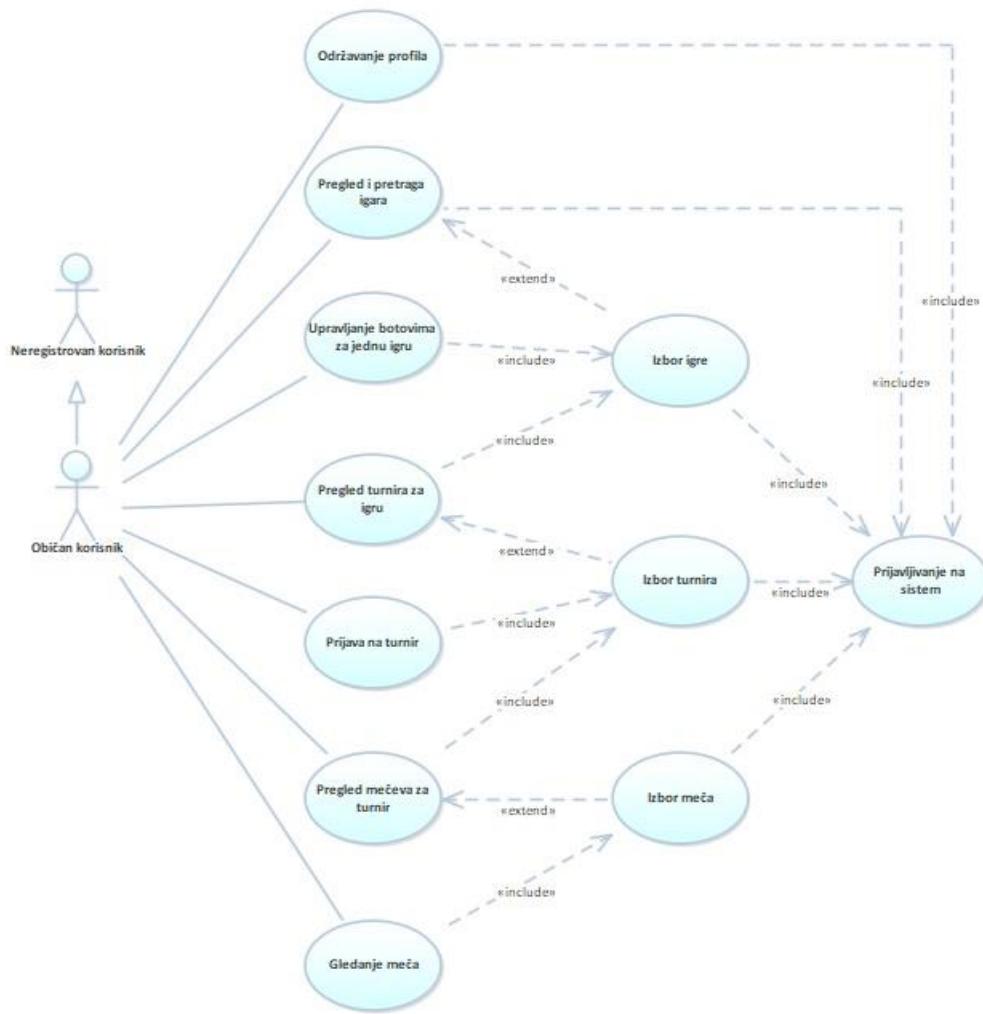
Potrebno je korisnika obavestiti o neuspešnom prijavljivanju.

Slučaj korišćenja: Pregled igara



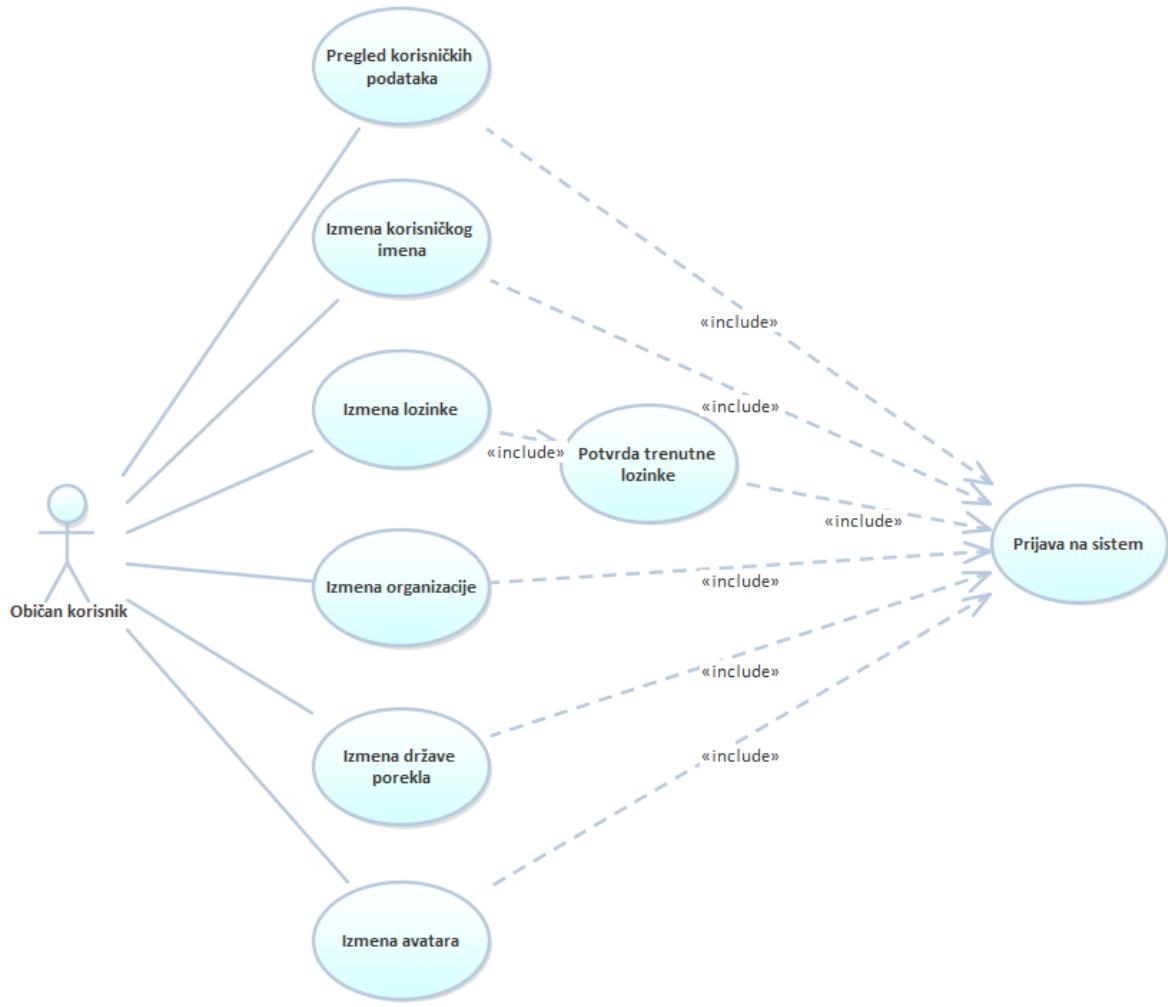
Slika 24. Slučaj korišćenja: Pregled igara

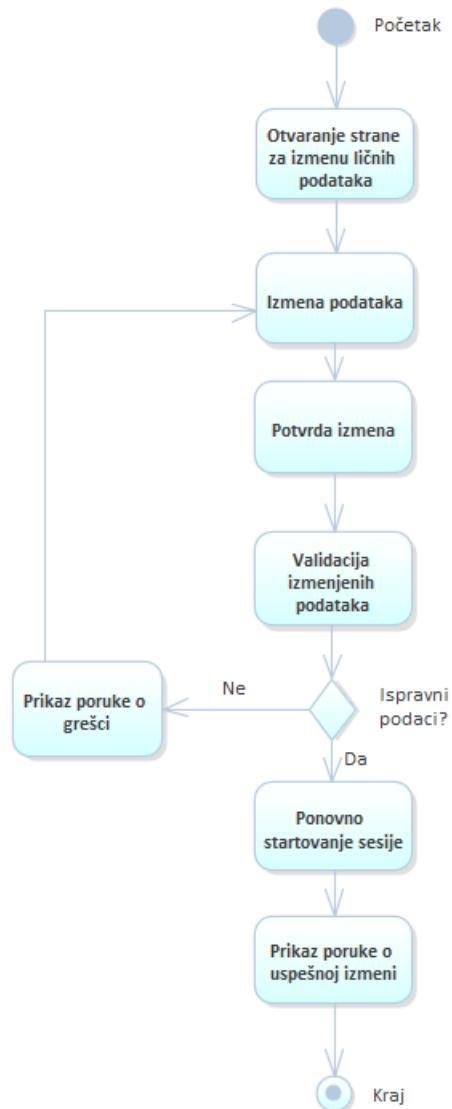
Dijagrami slučajeva korišćenja za običnog korisnika



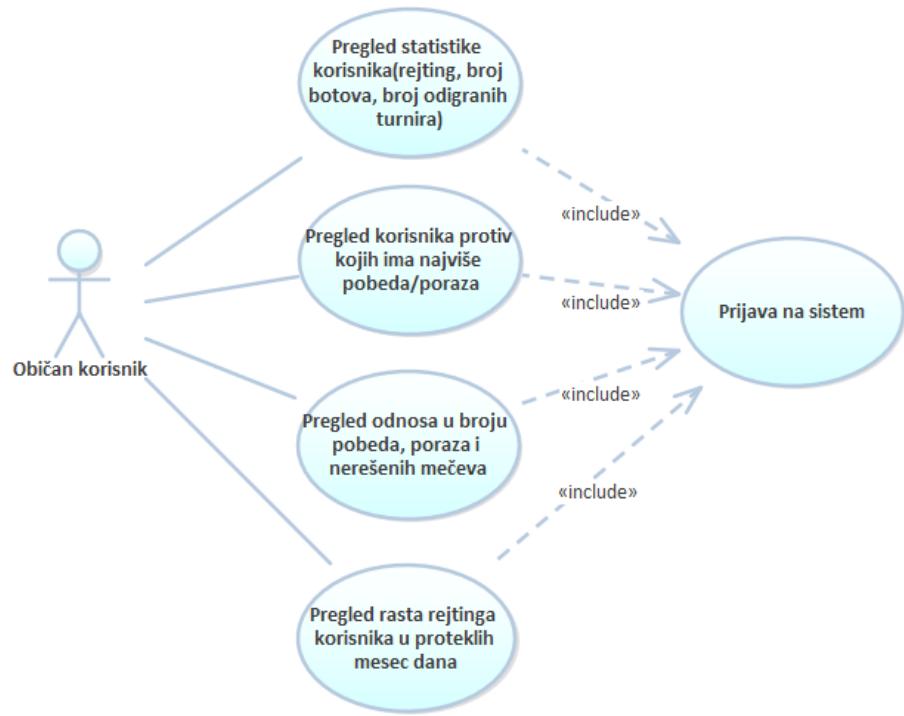
Slika 25. Opšti slučajevi korišćenja za običnog korisnika

Slučaj korišćenja: Održavanje profila

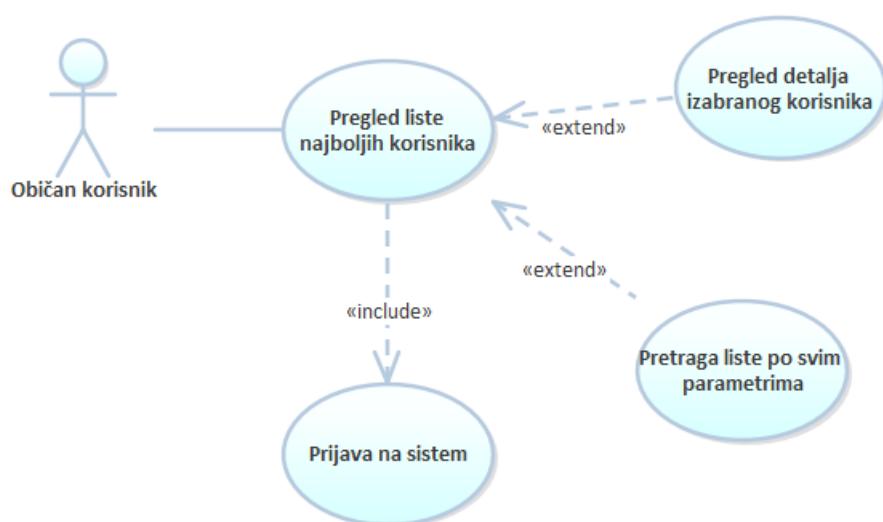




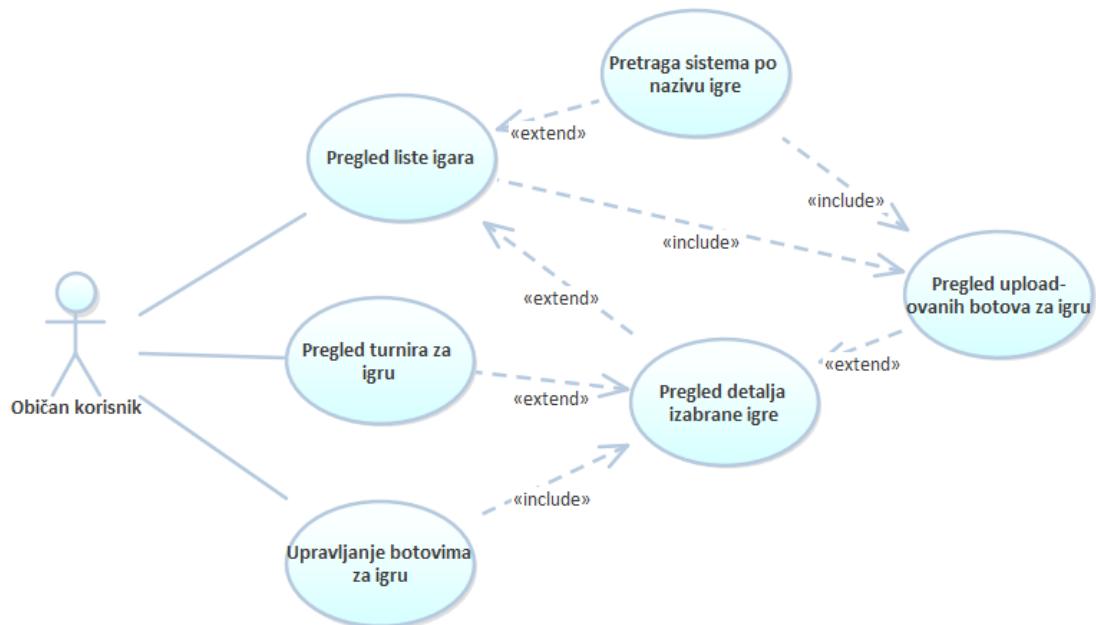
Slučaj korišćenja: Analiza korisnika



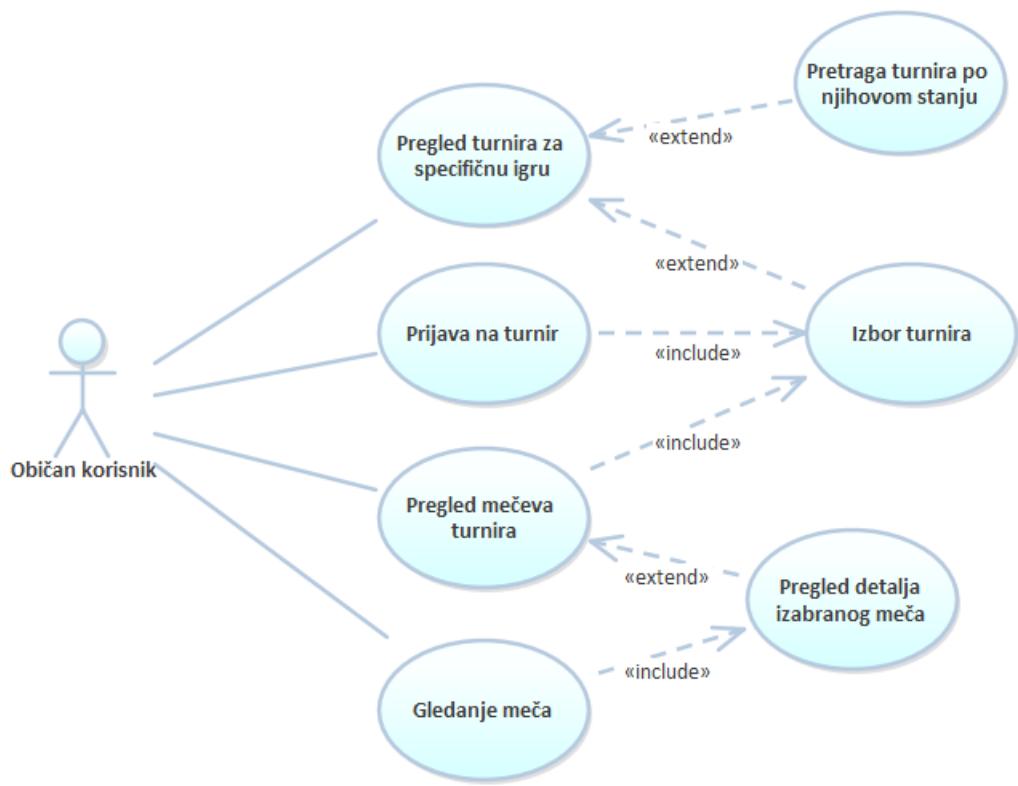
Slučaj korišćenja: Leaderboard



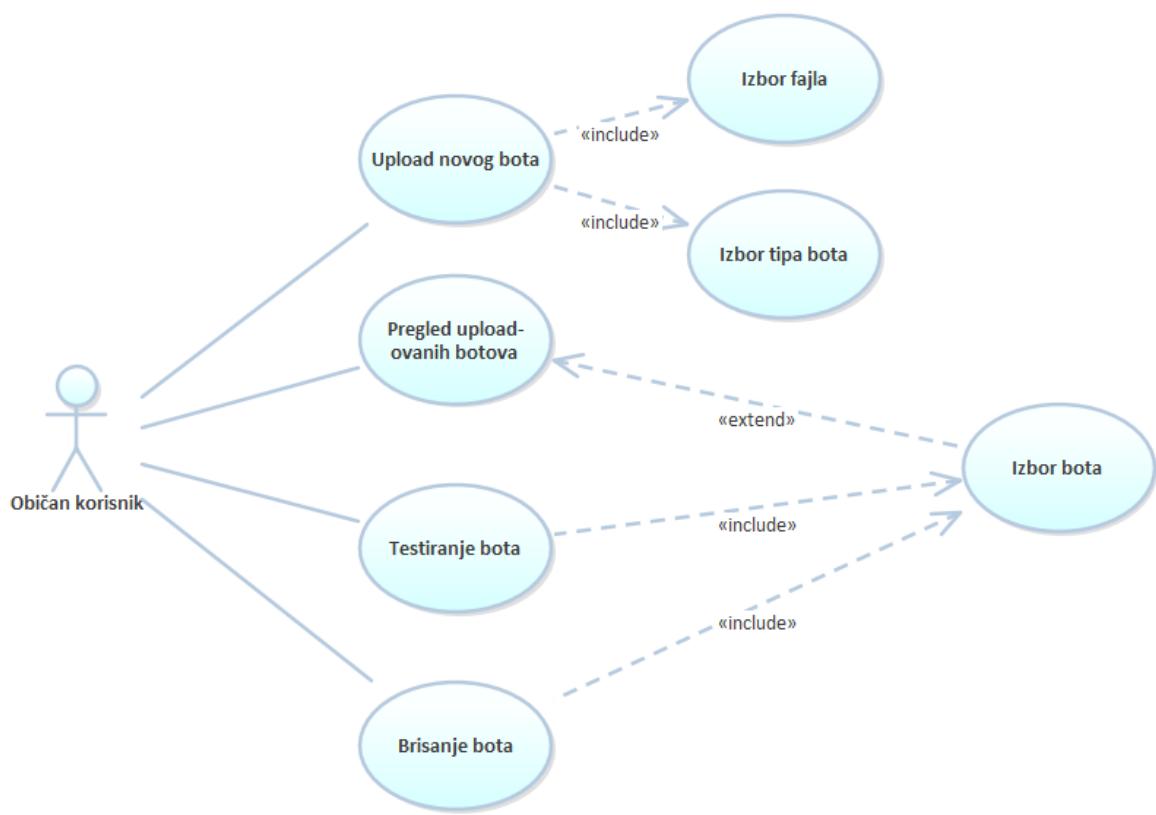
Slučajevi korišćenja: Pregled igara



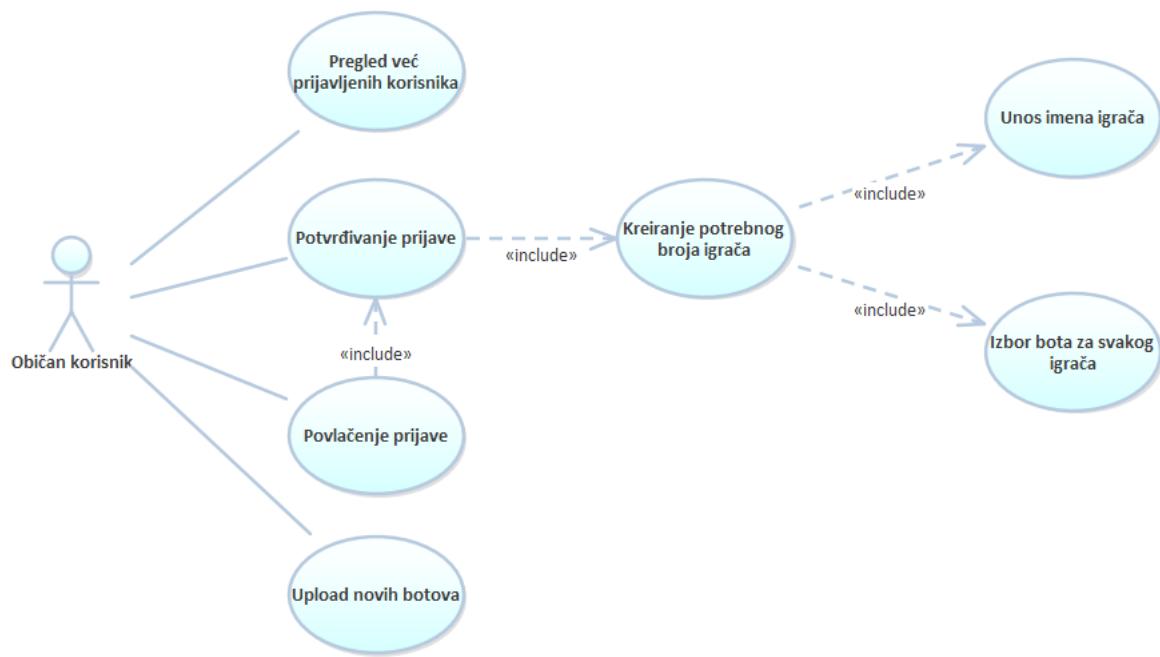
Slučajevi korišćenja: Pregled turnira



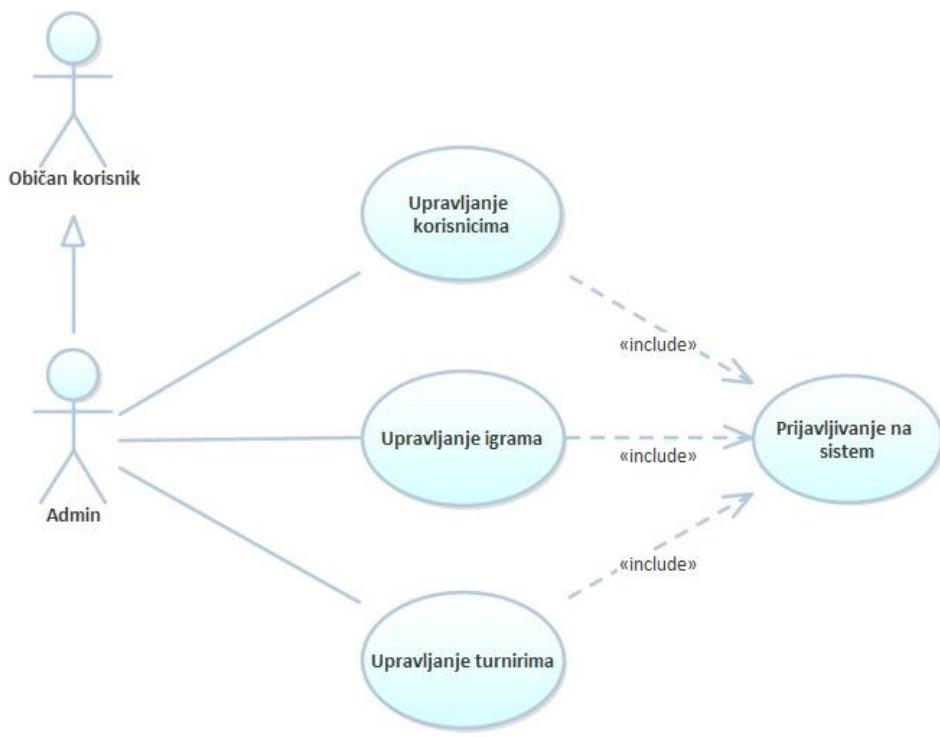
Slučajevi korišćenja: Pregled botova



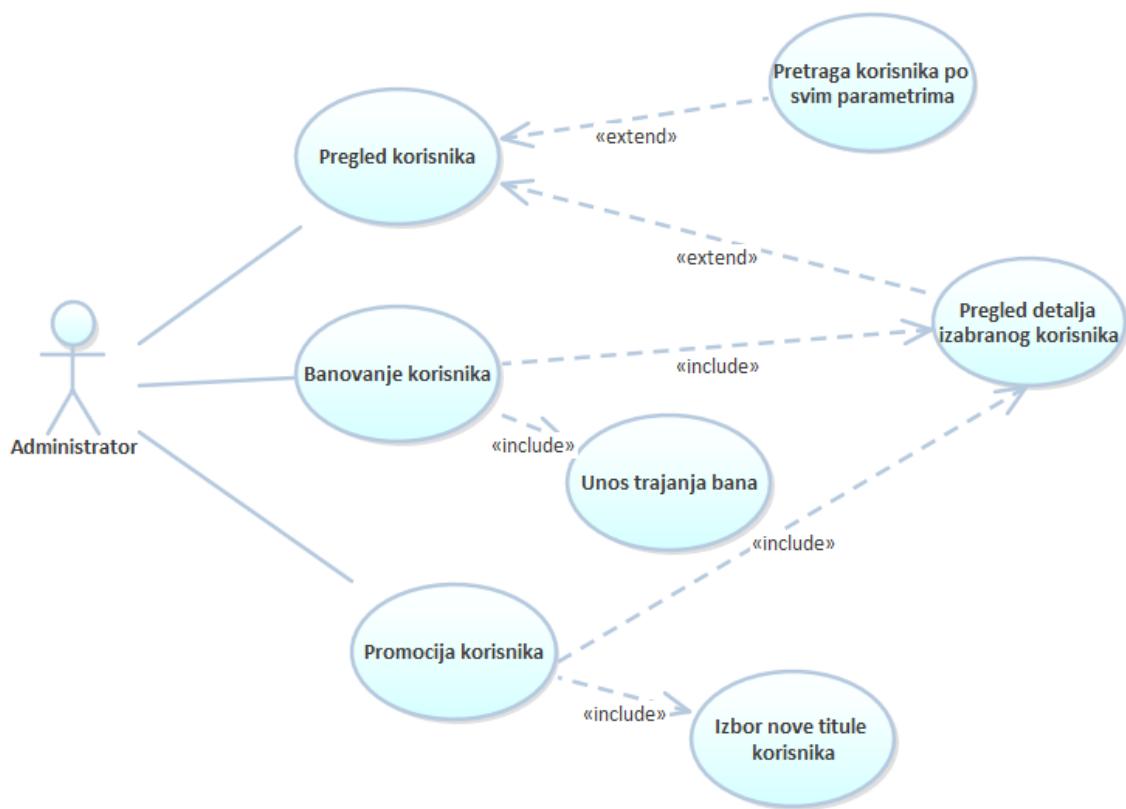
Slučaj korišćenja: Prijava na turnir



Dijagrami slučajeva korišćenja za administratora



Slika 26. Opšti slučajevi korišćenja za administratora



Slika 27. Slučajevi korišćenja vezani za upravljanje korisnicima

Slučaj korišćenja: Banovanje korisnika

Kratak opis:

Onemogućavanje pristupa sistemu određenom registrovanom korisniku.

Opis slučaja korišćenja:

Omogućiti administratoru da udalji izabranog korisnika sa sistema na vremenski period koji on izabere.

Glavni tok događaja:

U pregledu korisnika administrator vrši izbor korisnika kojeg želi da banuje. Zatim iz padajuće liste bira opciju za ban, nakon čega se otvara novi prozor u kojem se traži da admin unese datum do kojeg želi da korisnik bude banovan. Nakon ispravnog unosa datuma i potvrde istog, administrator će dobiti poruku o uspešno izvedenom banovanju, dok izabrani korisnik više neće moći da se uloguje na sistem dok ban ne istekne.

Definisanje uslova:

Pred-uslov: Korisnik je prijavljen na sistem kao administrator.

Post-uslov: Regularnost baze podataka.

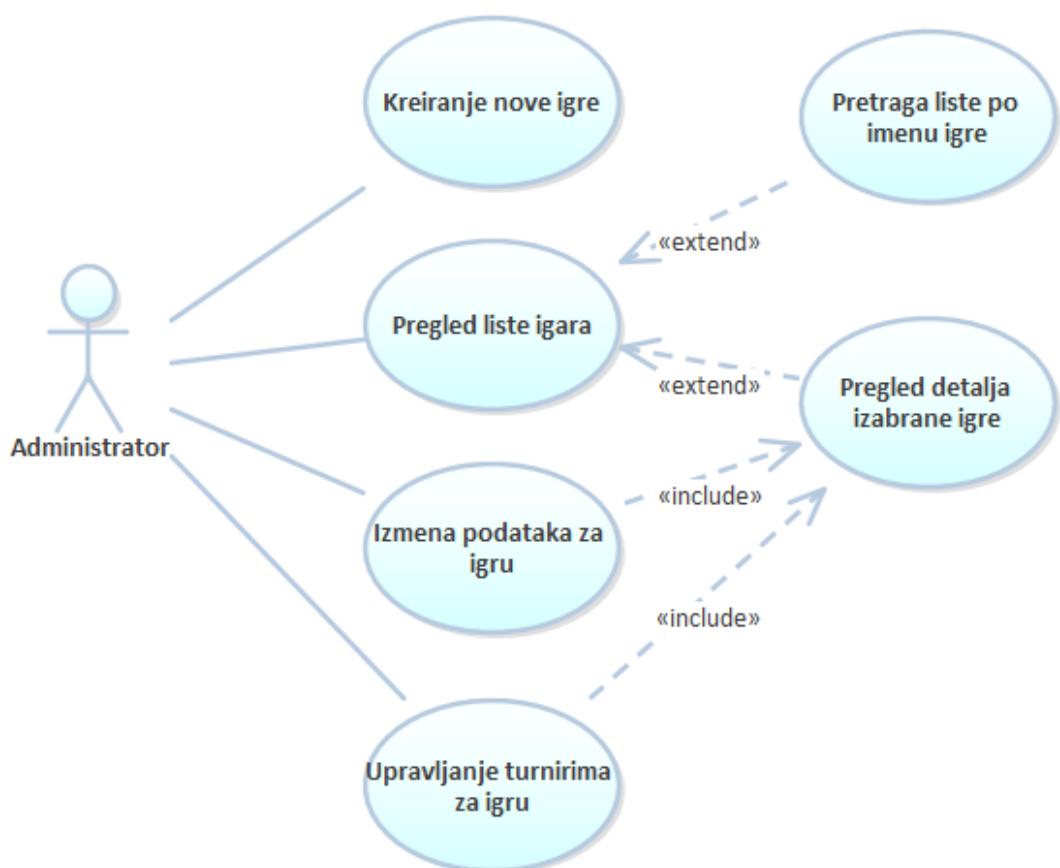
Alternativni tokovi događaja:

/

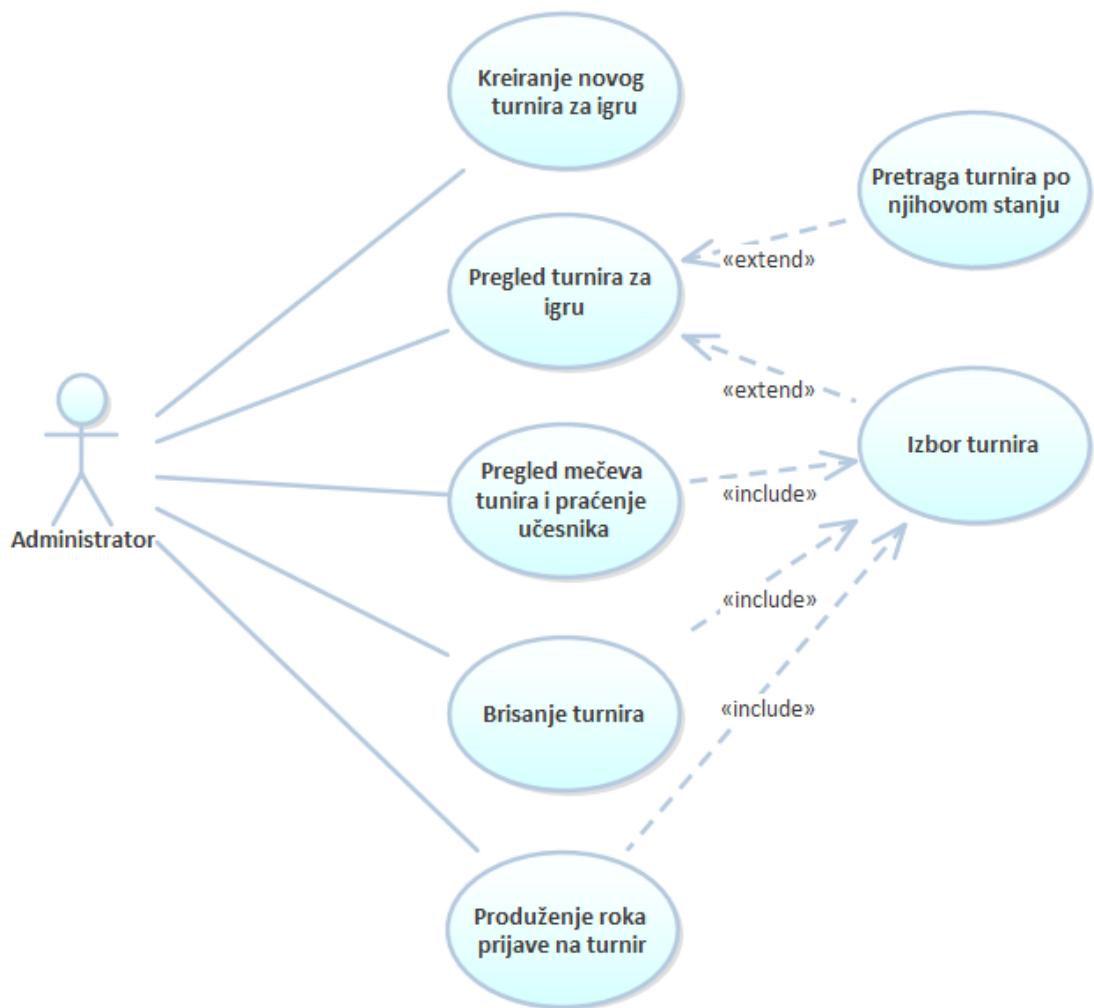
Zahtevi:

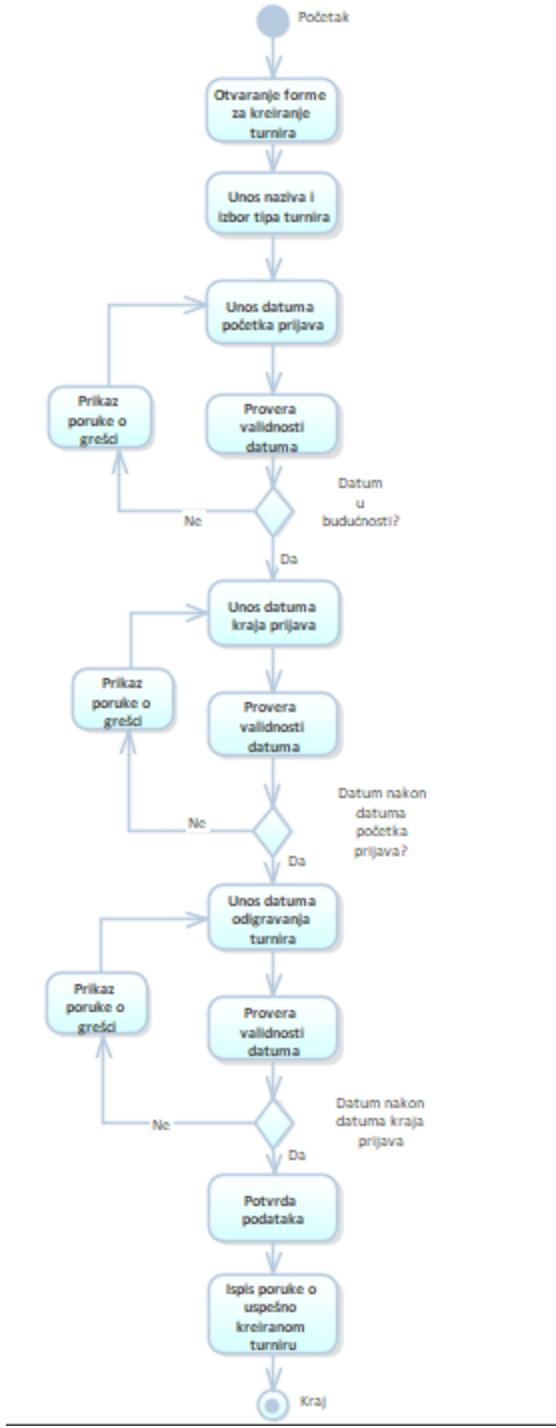
/

Slučajevi korišćenja: Upravljanje igrama



Slučajevi korišćenja: Upravljanje turnirima

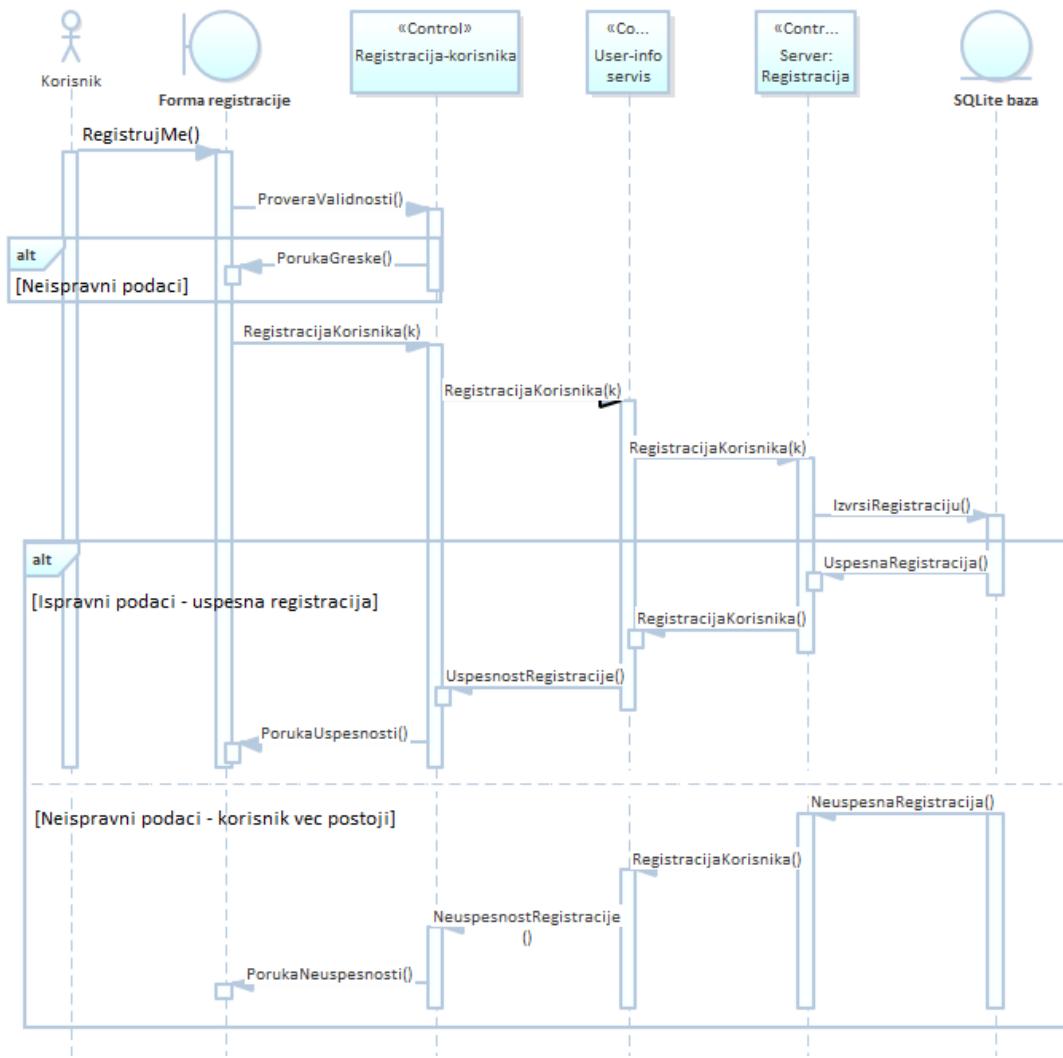




Dijagrami sekvenci

U ovom poglavlju biće predstavljeni dijagrami sekvenčni aplikacije Olimijada. Komunikacija između klijentskih i serverskih komponenti se obavlja pomoću web servisa koji su prikazani na dijagramima. Klijentska komponenta, preko servisa zove funkciju odgovarajućeg web servisa koji potom komunicira sa serverskom komponentom.

Registracija korisnika na sistem

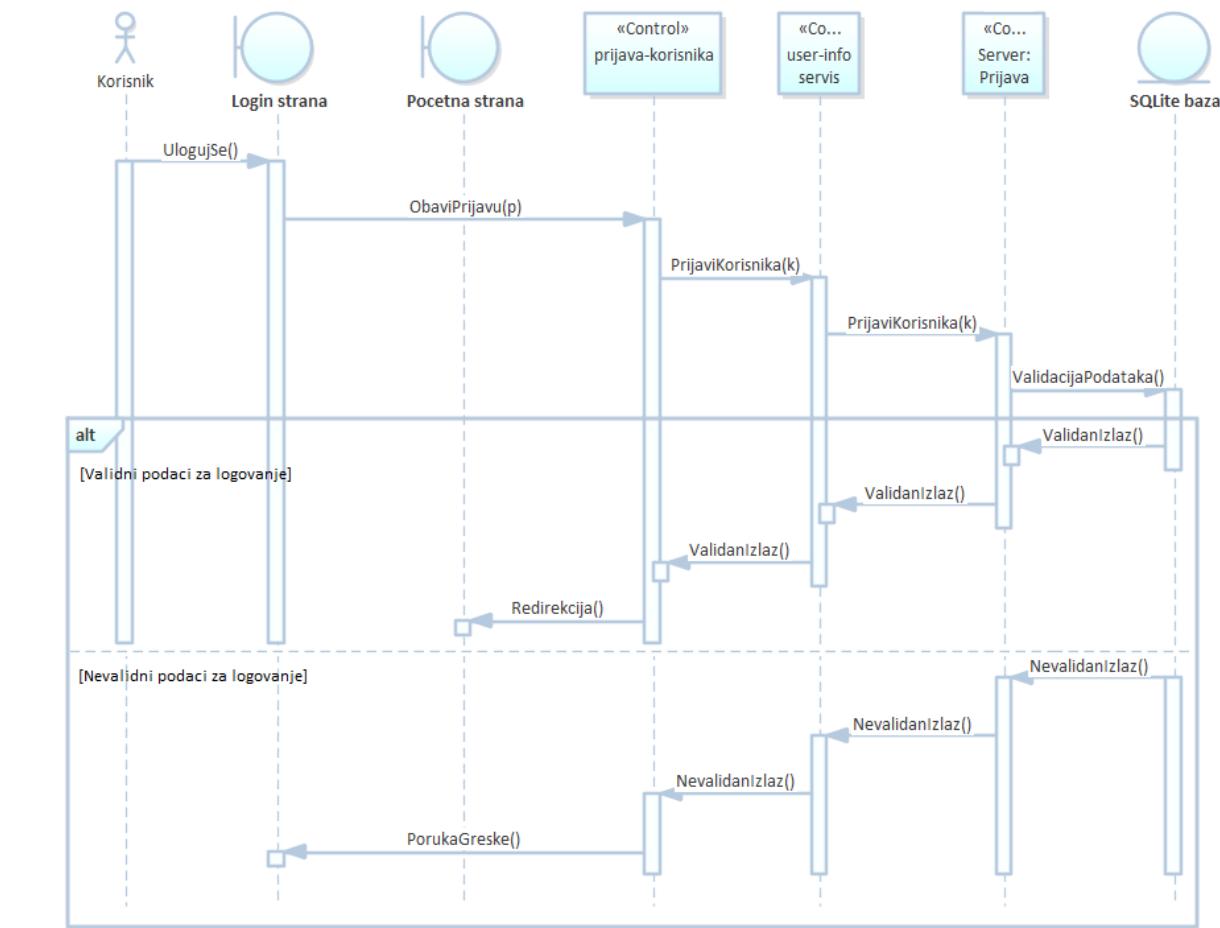


Slika 13. Dijagram sekvenčni: Registracija korisnika na sistem

Na slici 13. prikazan je dijagram sekvenčni registracije korisnika na sistem. Nakon što korisnik klikne na dugme *RegistrujMe* vrši se prosleđivanje unetih podataka pomoću klijentske komponente *Registracija-korisnika* i web servisa user-info-service do servera. Server proverava u bazi podataka validnost unetih podataka i na osnovu toga vraća

odgovarajuću poruku klijentskoj komponenti koja, ukoliko je kombinacija validna, registruje korisnika na sistem, a u suprotnom ga obaveštava o pogrešnoj kombinaciji.

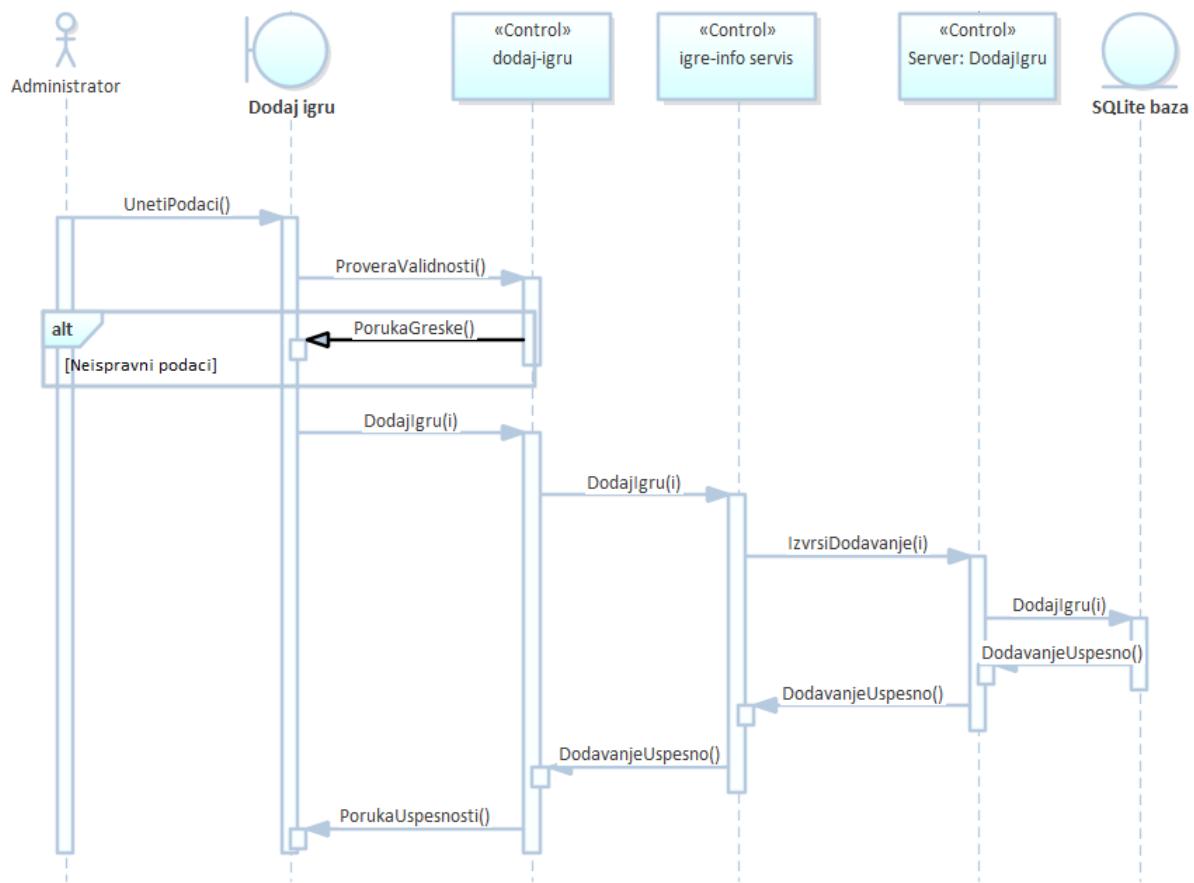
Prijava korisnika na sistem



Slika 14. Dijagram sekvenci: Prijavljivanje korisnika na sistem

Na slici 14. prikazan je dijagram sekvenci prijavljivanja korisnika na sistem. Nakon što korisnik klikne na dugme *UlogujSe* vrši se prosleđivanje korisničkog imena i šifre pomoću klijentske komponente *prijava-korisnika* i web servisa do servera. Sever proverava u bazi podataka ispravnost unetih parametara i na osnovu toga vraća odgovarajuću poruku klijentskoj komponenti koja, ukoliko je kombinacija validna, redirektuje korisnika na početnu stranu, a u suprotnom ga obaveštava o pogrešnoj kombinaciji.

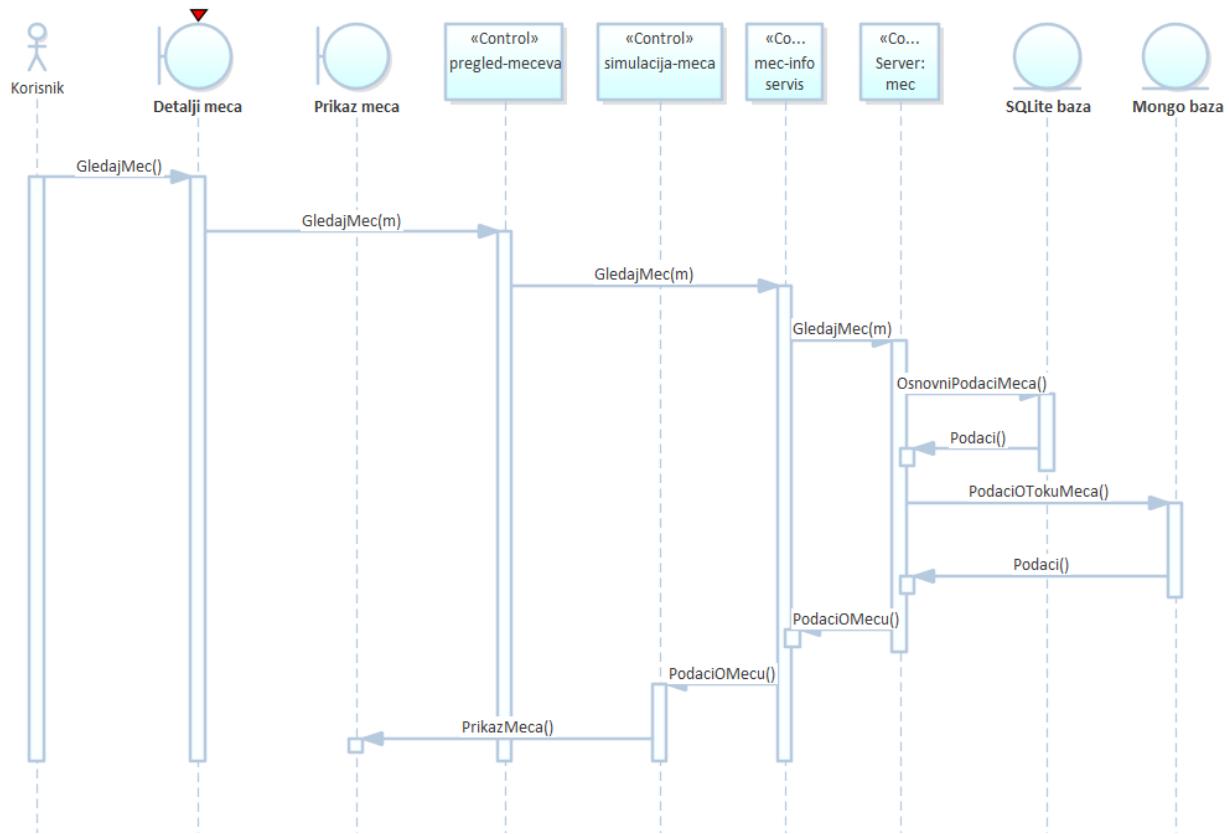
Dodavanje igre



Slika 15. Dijagram sekvenci: Dodavanje igre

Na slici 15. prikazan je dijagram sekvenci dodavanja igre. Nakon što administrator unese podatke o novoj igri u klijentskoj komponenti dodaj-igru se proveravaju uneti podaci. Ukoliko su podaci validni komponenta preko web servisa šalje podatke serveru koji ih prosleđuje bazi. Nakon toga server vraća poruku o uspešnosti dodavanja igre. Ukoliko podaci nisu validni, administrator će dobiti poruku o grešci.

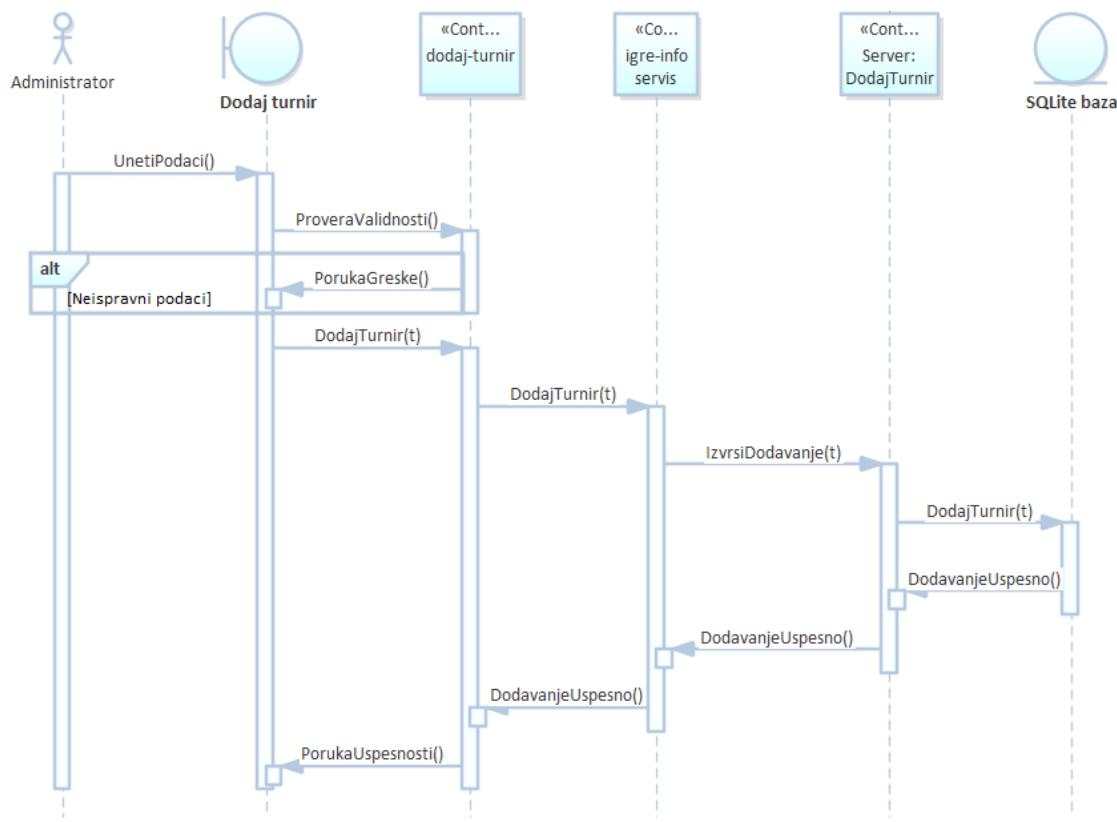
Gledanje mečeva



Slika 16. Dijagram sekvenci: Gledanje meča

Na slici 16. prikazan je dijagram sekvenci gledanja meča. Nakon što korisnik izabere meč koji želi da gleda i klikne na dugme GledajMec klijentska komponenta pregled-meceva preko web servisa šalje zahtev serveru koji traži osnovne podatke meča od SQLite baze, nakon što dobije osnovne podatke, server zahteva podatke o toku meča od MongoDB baze. Kada je server dobio tražene informacije vraća podatke o meču klijentskoj komponenti simulacija-meča koja omogućava korisniku da vidi meč.

Dodavanje turnira



Slika 17. Dijagram sekvenci: Dodavanje turnira

Na slici 17. prikazan je dijagram sekvenci dodavanja turnira. Nakon što administrator uneše podatke o novom turniru u klijentskoj komponenti dodaj-turnir se proveravaju uneti podaci. Ukoliko su podaci validni komponenta preko web servisa šalje podatke serveru koji ih prosleđuje bazi. Nakon toga server vraća poruku o uspešnosti dodavanja turnira. Ukoliko podaci nisu validni, administrator će dobiti poruku o grešci.

Dizajn korisničkog interfejsa

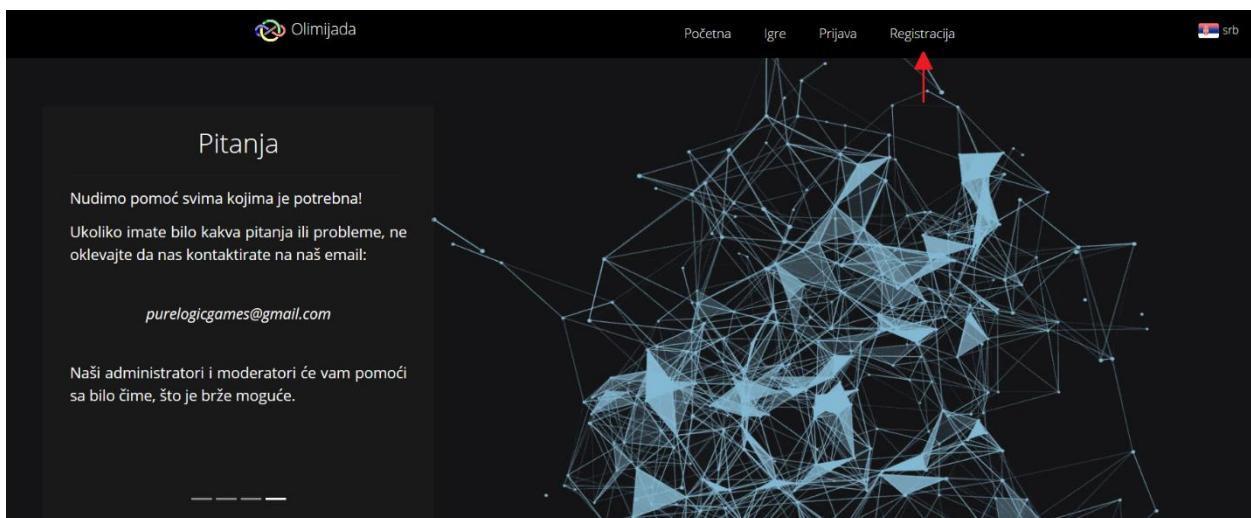
Tehnologije korisničkog interfejsa

Radi ostvarivanja responsive dizajna, u razvoj svih grafičkih komponenti biće integrisana upotreba ngBootstrap tehnologija.

U skladu sa najsavremenijim trendovima Web tehnologija, za prikaz svih korisničkih kontrola biće korišćen Angular Material dizajn.

Slike aplikacije

U nastavku ovog poglavlja biće prikazano kako će izgledati delovi aplikacije, kao i kratak opis načina njihovog funkcionisanja.



Klikom na „Registracija“ otvorice vam se forma za registraciju.

Registracija

* E-mail:

Uneti email je neispravan.

* Korisničko ime:

Korisničko ime mora imati između 6 i 12 karaktera.

* Lozinka:

Lozinka mora da sadrži barem 8 karaktera, od toga barem: 1 veliko slovo, 1 malo slovo i 1 broj.

* Potvrди lozinku:

Unete lozinke se ne poklapaju.



5
4
3
2
1

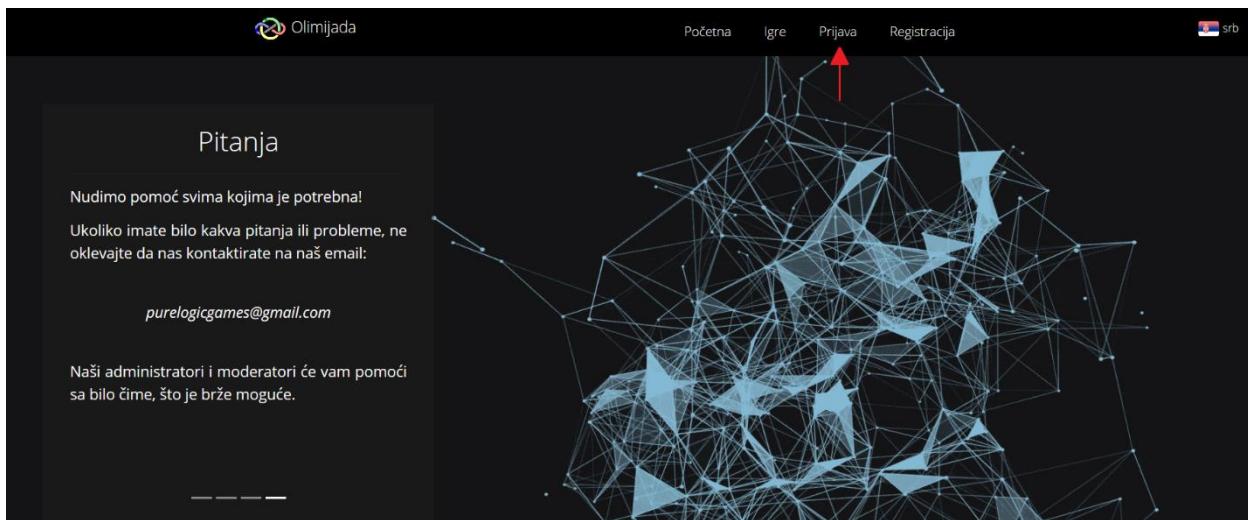
Registruj se ← 6

Forma za registraciju sadrži polja za unos:

1. Email adrese
2. Korisničkog imena
3. Lozinke
4. Potvrde lozinke
5. Slike korisnika

Ukoliko je niste ispoštovali pravila za unos, ili ako je korisničko ime zauzeto ili ako ste se već prijavili sa datom email adresom, sistem će obavestiti korisnika o grešci. Ukoliko su sva polja imaju validan unos, klikom na dugme 6 (regustruj se) korisnik se registruje i dobija obaveštenje o uspešnoj registraciji.

Nakon uspešne registracije korisnik može da se prijavi na sitem. Klikom na dugme „Prijava“.



Kada korisnik klikne da dugme prijava, otvara mu se forma za prijavu.

A screenshot of a login form titled 'Prijava'. The form contains two input fields: one for 'Korisničko ime/email:' and another for 'Lozinka:', both with placeholder text 'Unesite korisničko ime/email' and 'Unesite lozinku' respectively. Below the inputs is a blue 'Prijavi se' button. Three red arrows numbered 1, 2, and 3 point to these three elements from the right side of the image.

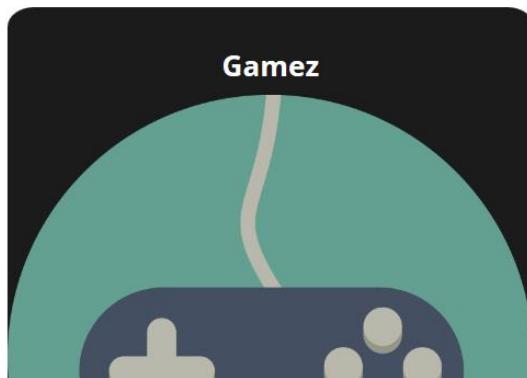
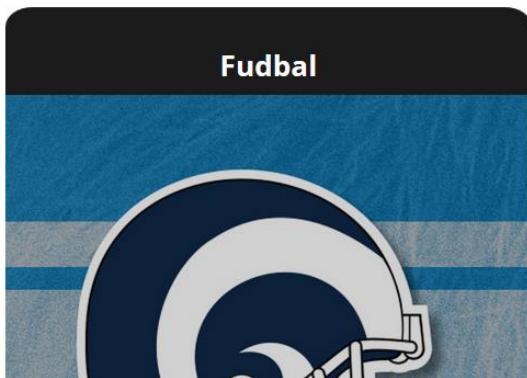
Na stranici se nalaze:

1. Polje za unos korisničkog imena ili email adrese
2. Polje za unos lozinke
3. Dugme kojim se vrši prijavljivanje na sistem, kada korisnik klikne na dugme vrši se prijava korisnika, ukoliko su neki od podataka loši korisnik dobija obaveštenje ukoliko jesu podaci dobri korisnik dobija obaveštenje o uspešnom prijavljivanju



U svakom trenutku korisniku se pruža mogućnost promene jezika (srpski/engleski).

A screenshot of a page titled "Pitanja" (Questions). The page contains text about providing help and an email address for contact. On the left, there is a sidebar with the "Olimijada" logo and a "purelogicgames@gmail.com" email address. The main content area features a large blue network graph. The header at the top includes the "Igre" link, which is highlighted with a red arrow. Other header links are "Početna", "Prijava", and "Registracija". A small "srpski" language option is also visible in the top right corner.



Klikom na Igre korisnik ostvaruje mogućnost prikaza svih igara kao i osnovnih podataka o njima.

Od podataka na ovoj stranici korisnik je u mogućnosti da vidi ime igre, tip igre kao i kraći opis igre. Klikom na jednu od igara ima mogućnost da vidi više podataka o toj igri.

[Nazad na listu igara](#)

Defactus

--- Adventure ---

Defactus is one of the most interesting and difficult games to code a bot for. While Defactus has only a few rules, the game can be quite complicated, as there are countless ways the game can play out.

 **Timska igra** (2 igrača)

Prosečno trajanje meča 4 minut(a)



Pravila igre



General Defactus rules

White is always first to move and players take turns alternately moving one piece at a time. Movement is required. If a player's turn is to move, he is not in check but has no legal moves, this situation is called "Stalemate" and it ends the game in a draw.
Each type of piece has its own method of movement. A piece may be moved to another position or may capture an opponent's piece, replacing on its square (en passant).

Ova stranica sadrži podatke o:

- Imenu igre
- Tipu igre
- Broju igrača igre
- Prosečno vreme igre
- Pravilima igre

Klikom na polje 1 korisnik može otvoriti podatke o pravilima igre.

Klikom na dugme 2 korisnik će se vratiti na listu igara.

Tip korisnika može biti admin i običan korisnik. Admin ima sledeći prikaz početne strane.

🏠 Početna

Turniri bez dovoljnog broja učesnika:

| Naziv turnira | Igra | Registracija istekla | Tip turnira | Image |
|---------------|--------|----------------------|-------------|-------|
| Delta league | Delta | 6/5/18, 7:32 PM | League | |
| Fudbal Cup | Fudbal | 6/5/18, 7:33 PM | Cup | |
| Fudbal League | Fudbal | 6/5/18, 7:59 PM | League | |
| Gamez Cup | Gamez | 6/5/18, 8:59 PM | Cup | |

Nedavno registrovani korisnici:

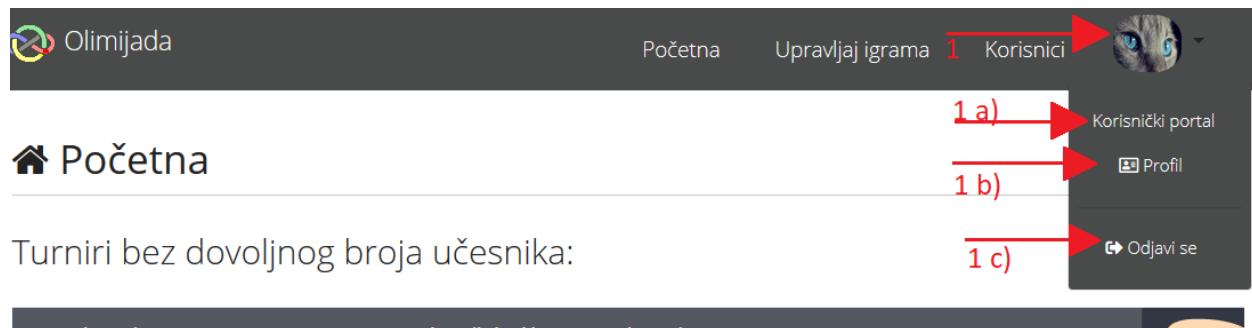
| | | | |
|--------------------------------|-------------------------------------|--------------------------------|--------------------------------|
| Toma123 | Tijana | Hozumite | Ziolin |
| Država: | Država: Serbia | Država: | Država: |
| Organizacija: | Organizacija: IMI | Organizacija: | Organizacija: |
| Email: acika298@gmail.com | Email: ristovic.tijana.kg@gmail.com | Email: lazajfiju44@gmail.com | Email: nestorovic.96@gmail.com |
| Anotheruser | Whatsapp | Rooter | Totalies |
| Država: | Država: | Država: | Država: |
| Organizacija: | Organizacija: | Organizacija: | Organizacija: |
| Email: nestorovic.96@gmail.com | Email: nestorovic.96@gmail.com | Email: nestorovic.96@gmail.com | Email: nestorovic.96@gmail.com |

Kratka istorija turnira:

| Naziv turnira | Igra | Datum odigravanja | Tip turnira | Image |
|------------------------------|-------|-------------------|-------------|-------|
| Delta super liga | Delta | 6/6/18, 1:00 AM | League | |
| Delta super liga pomozi boze | Delta | 6/6/18, 12:25 AM | League | |

Podaci koje admin vidi na svojoj početnoj a ne nalaze se svi na gore navedenoj slici redom su:

- Turniri bez dovoljno prijavljenih učesnika, svako polje ima mogućnost da se klikom na njega otvari stranica turnira
- 10 poslednjih registrovanih korisnika svako polje ima mogućnost klikom na njega otvaranje stranice pregleda korisnika
- 10 poslednje odigranih turnira svako polje ima mogućnost da se klikom na njega otvari stranica turnira
- 10 najpopularnijih igrara

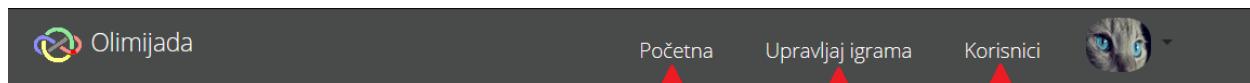


Klikom na polje 1 korisnik (admin) otvara padajući meni, koji ima sledeće izbore:

- a) Korisnički portal – mogućnost da admin predje u deo aplikacije i dalje korišćenje kao običan korisnik
- b) Profil – ovo polje otvara mogućnost da korisnik vidi i menja svoje podatke
- c) Odjavi se – odjava korisnika

Pregled sopstvenog profila u kome korisnik može da menja podatke o sebi ali takođe i da vidi svoju statistiku i analizu izgleda na sledeći način.

U adminovom meniju se nalaze i sledeće opcije.



Početna

Turniri bez dovoljnog broja učesnika:

| Naziv turnira | Igra | Registracija istekla | Tip turnira | |
|---------------|--------|----------------------|-------------|--|
| Delta league | Delta | 6/5/18, 7:32 PM | League | |
| Fudbal Cup | Fudbal | 6/5/18, 7:33 PM | Cup | |

1. Klikom na dugme početna admin ima mogućnost da se u bilo kom trenutku vrati na početnu stranu.
2. Klikom na dugme Upravlja igrama adminu se otvara lista igara sa podacima o igrama, mogućnost menjanja tih podataka, upravljanje turnirima
3. Klikom na dugme Korisnici admin može videti listu svih registrovanih korisnika, kao i unapredjenje ili unazadovanje korisnika.

Defactus is one of the most interesting and difficult games to code a bot for. While Defactus has only a few rules, the game can be quite complicated, as there are countless ways the game can play out.

Timska igra (2 igrača)

Prosečno trajanje meča: 4 minut(a)

Pravila igre

Na ovoj stranici admin može videti:

1. Dodaj igru – otvara formu za dodavanje nove igre
2. Pretraga igara – pretraga po nazivu
3. Lista trenutnih igrara
4. Izmena podataka igre – otvara formu za promenu podataka igre
5. Turniri – otvara listu turnira za datu igru
6. Kreiraj turnir – otvara formu za pravljenje novog turnira

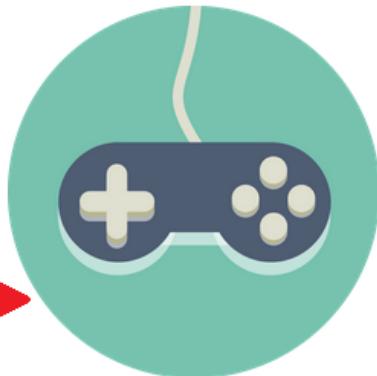
Dodaj igru

12 → Nazad na igre

* Naziv igre:

Unesite naziv igre

Dodaj ikonicu igre:



* Tip igre:

Izaberite tip igre ▾

-  Igra za jednog igrača
-  Timska igra

Tipovi botova

| | |
|-----------------|-------------------|
| Naziv tipa bota | Dodaj bota |
|-----------------|-------------------|

Pozadina igre:



* Prosečno vreme trajanja igre u minutima:

Unesite vreme trajanja u minutima

8

* Izaberite fail igre

Browse... No file selected.

Kratak opis igre:

13 → EN

Pravila igre:

Enter text here...

14 EN

Kreisrat Iggy

11

10

Forma za dodavanje igre sadrži:

1. Naziv igre
2. Padajući meni iz koga se izabere tip igre
3. Odabir da li je igra timska ili pojedinačna ako je timska otvor se polje za unos broja igrača

Igra za jednog igrača

Timska igra

* Broj igrača u timu:

Unesite broj igrača

4. Polje za unos tipova botova za igru
5. Prosečno vreme trajanje igre
6. Polje za odabir fajla igre
7. Polje za dodavanje slike igre
8. Polje za dodavanje slike pozadine igre
9. Polje za unos kratkog opisa igre
10. Polje za unos pravila igre

Kada se popune podaci o igri admin može dodati igru klikom na dugme 11.

Klikom na dugme 12 admin se vraća na stranicu upravljanja igrama.

Kada se otvori forma za menjanje podataka igre otvor se sledeća stranica.

Polja u izmeni igre su ista kao i polja za dodavanje igre, samo sa popunjениm podacima izabrane igre.

Dugme 13 i 14 služi da admin prebaci na kratak opis odnosno pravila igre na drugi jezik i upise podatke za taj drugi jezik.

[Detalji igre](#)[Izmeni podatke igre](#)[Kreiraj turnir](#)[Turniri](#)

Izmeni podatke igre Defactus

Naziv igre:

Defactus

Tip igre:

Adventure

x ▾

 Igra za jednog igrača

 Timska igra

Broj igrča u timu:

2



Tipovi botova: 

DPS 

Tank 

Healer 

igrac defactus 

Naziv tipa bota

Dodaj bota

Prosečno vreme trajanja igre u minutima:

3



Izaberite fajl igre

No file selected.

Kratak opis igre:

EN

Defactus je jedna od najinteresantnijih i najtežih igara kada je upitanju pravljenje bota za nju. Iako Defactus ima samo par pravila, igra može biti jako komplikovana, pošto postoji mnogo načina na koje se igra može igrati.

Pozadina igre:



Promeni sliku igre:

No file selected.

Pravila igre:

EN



Pravila Defactus-a

Belja je uvek prva koja se pomera i igrači se polako menjaju po jednom komadu. Pokret je neophodan. Ako je okretanje igrača a pokret, on nije u čekanju, ali nema pravnih poteza, ova situacija se zove "Stalemate" i završava se kao nerešeno.

Klikom na gore pomenuto dugme za dodavanje turnira otvara se stranica sa sledećom formom.

[Detalji igre](#)[Izmeni podatke igre](#)[Kreiraj turnir](#)[Turniri](#)

Defactus turnir

* Naziv turnira:



* Tip turnira:

- Kup
- Liga



* Registracija za turnir počinje:



* Registracija za turnir se završava:



* Turnir počinje:

[Kreiraj turnir](#)

1. Naziv turnira
2. Odabir tipa turnira Kup – Liga
3. Vreme početka prijave na turnir
4. Vreme kraja prijave na turnir
5. Vreme početka turnira

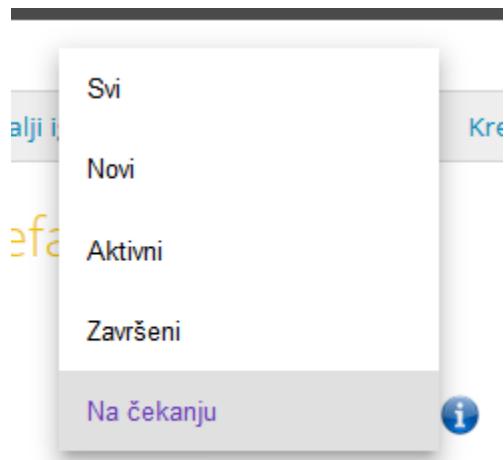
Cela forma se potvrđuje klikom na dugme 6.

Klikom na dugme turniri otvara se stranica za pregled turnira.



Trenutno ne postoje turnriri za ovu kategoriju.

1. Otvara se lista sa opisom turnira (Na čekanju, novi, u toku, završeni, svi)
2. Otvara stranicu sa formom za pravljenje novog turnira za izabranu igru



Nerešeni turniri - Započni, produži ili izbriši ove turnire!

Novi turniri - Vidi informacije turnira!

Predstojeći turniri - Pogledaj postave turnira u najavi!

Završeni turniri - Pogledaj odlične partie!

Defactus odmah
 Cup Turnir je počeo: 6/5/18, 11:32 PM 4 igrača Otvori 2

Defactus league odmah
 League Turnir je počeo: 6/5/18, 11:43 PM 4 igrača Otvori 2

Defactus cup 2 odmah
 Cup Turnir je počeo: 6/5/18, 11:50 PM 4 igrača Otvori 2

Defactus liga super
 League Turnir je počeo: 6/5/18, 11:52 PM 4 igrača Otvori 2

Klikom na 1 otvara se lista turnira za kliknuto stanje turnira. Klikom na dugme 2 otvara se turnir zavisno od tipa turnira Cup ili Liga.

Izgled stranice lige izgleda kao na sledećoj slici.

DEFACTUS LEAGUE ODMAH

Turnir je počeo:
Jun 5, 2018, 11:43:00 PM

Rang lista:

| Pozicija | Korisničko ime | Pobede | Nerešene | Izgubljene | Bodovi | Mečevi |
|----------|-----------------|--------|----------|------------|--------|--------|
| 1. | Rooter | 2 | 0 | 0 | 6 | |
| 2. | Gregory | 1 | 0 | 1 | 3 | |
| 3. | Admin | 0 | 0 | 2 | 0 | |
| 4. | Whatsapp | 0 | 0 | 0 | 0 | |

Mečevi za svako kolo:

| Korisnik 1 | Rezultat | Korisnik 2 |
|------------|----------|------------|
| Admin | 20 : 100 | Gregory |
| Rooter | 50 : 50 | Whatsapp |

« Prethodno kolo 1 2 3 Sledеće kolo » 1

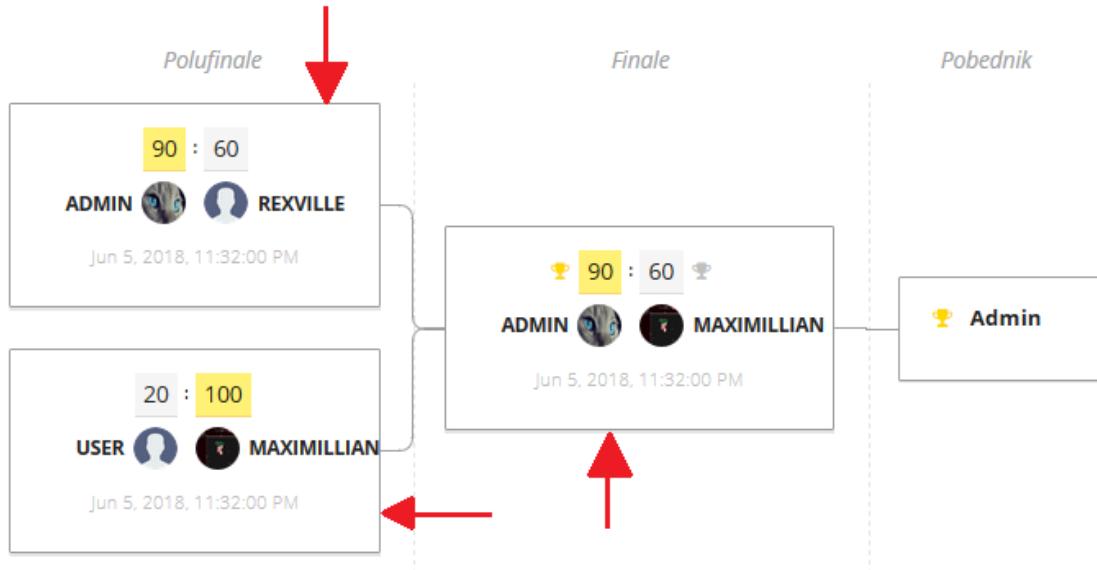
U navigaciji 1 korisnik može izabrati kolo lige i videti mečeve iz tog kola.

Na dugme 2, tj. Svako polje kao sto su označena ova dva polja korisnik može videti detalje tog meča.

Izgled kupa je prikazan na sledećoj slici.

DEFACTUS ODMAH

Turnir je počeo:
Jun 5, 2018, 11:32:00 PM

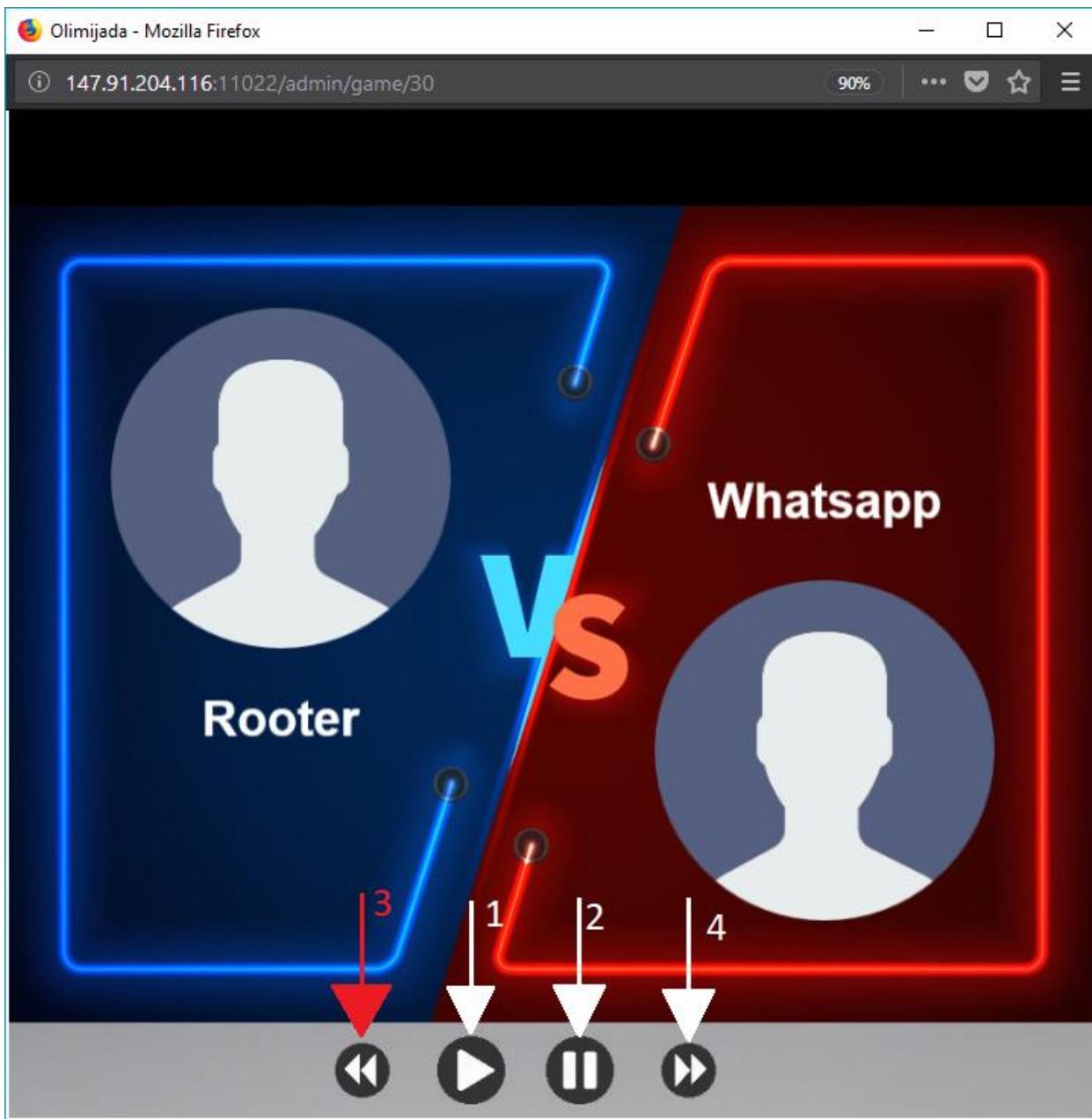


Klikom na navedena polja otvara se pregled meča kao i kod pregleda lige.



Klikom na imena korisnika otvara se pregled korisnika stranica sa njihovim podacima.

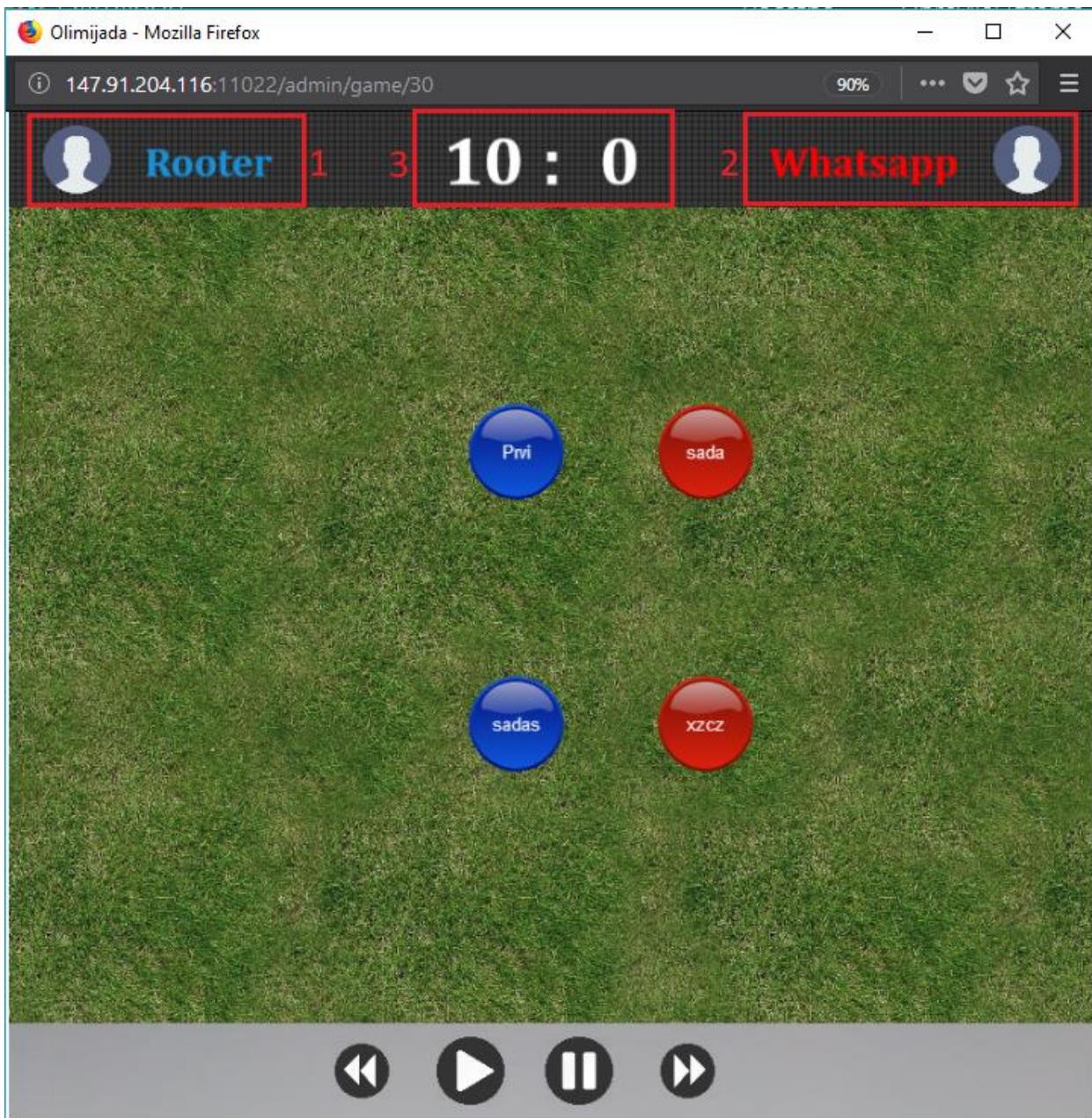
Klikom na dugme 2 otvara se prikaz meča u novom prozoru.



Opcije ove stranice su:

1. Početak prikaza meča
2. Stopiranje meča
3. Premotavanje meča unazad
4. Premotavanje meča unapred

Kada kliknemo na početak meča, otvara se sledeći prikaz.



U poljima:

1. Naziv i slika prvog korisnika
2. Naziv i slika drugog korisnika
3. Rezultat koji se menja u toku meča

U sredini se nalaze igrači korisnika.

Ukoliko se turnir za neki turnir vreme prijave završilo a nema dovoljno korisnika admin ima sledeće opcije.

Registracioni period za turnir **Proba** je istekao...

Nema dovoljno prijavljenih učesnika za ovaj turnir

Da li želite da:

Produžite registracioni period

Ovo će dati više vremena novim korisnicima da čuju o ovom turniru i da mu se pridruže.
Postavite datum novog registracionog perioda i datum startovanja novog turnira u poljima ispod

* Nova registracija se završava: **1**

2018-06-10T19:39

Obrišete ovaj turnir zauvek.

Ovo će obrisati sve učesnike koji su se prijavili za ovaj turnir.
Svi učesnici će biti informisani putem email-a, sa našim izvinjenjem, o brisanju turnira.

* Novi turnir počinje: **2**

2018-06-10T19:41

Produži registracioni period! 3

Izбриши turnir! 4

Broj registrovanih učesnika: **0 / 4**

Korisničko ime

Država

Organizacija

Rejting:

5
↓

Ukoliko admin želi da produži vreme prijave za turnir koristi polja 1 i 2.

1. Novo vreme kraja prijave na turnir
2. Novo vreme početka turnira

Sve to može učiniti na dugme 3.

Ukoliko želi da obriše turnir onda treba kliknuti na dugme 4.

Ispod se može videti tabela sa korisnicima koji su se prijavili do tada na turnir.

Klikom na dugme korisnici adminu se otvara sledeća stranica.

The screenshot shows a table of users with columns: Korisničko ime, Email, Tip korisnika, and several buttons (Pogledaj, Banuj korisnika, Unapredi/Snizi čin korisnika). A search bar at the top has a red arrow pointing to it labeled '2'. A red arrow labeled '1' points to the 'Pogledaj' button for the first user. A red arrow labeled '3' points to the gear icon for the second user. A red arrow labeled '4' points to the 'Banuj korisnika' button for the second user. A red arrow labeled '5' points to the 'Unapredi/Snizi čin korisnika' button for the second user. The table includes users like Admin, Anotheruser, Gregory, Hozumite, Maximillian, Ravenue, Rexville, Rooter, Tijana, and Toma123.

| Pretraži korisnike | | | |
|--------------------|------------------------------|---------------|---|
| Korisničko ime | Email | Tip korisnika | |
| Admin | nestorovic.96@gmail.com | Admin | Pogledaj 1 |
| Anotheruser | nestorovic.96@gmail.com | User | Pogledaj 3 4 |
| Gregory | mecak95@gmail.com | User | Pogledaj 5 |
| Hozumite | lazafiju44@gmail.com | User | Pogledaj |
| Maximillian | lazafiju44@gmail.com | User | Pogledaj |
| Ravenue | mecak95@gmail.com | User | Pogledaj |
| Rexville | lazafiju44@gmail.com | User | Pogledaj |
| Rooter | nestorovic.96@gmail.com | User | Pogledaj |
| Tijana | ristovic.tijana.kg@gmail.com | User | Pogledaj |
| Toma123 | acika298@gmail.com | Admin | Pogledaj |

« Prethodna strana 1 2 Sledeća strana »

1. Klikom na dugme 1 korisnik može otvoriti pregled korisnika
2. Klikom na polje 2 otvaraju se kriterijumi za pretragu korisnika
3. Klikom na zupcanik otvara se mogućnost da se neki korisnik banuje ili da mu se promeni čin
4. Banovanje korisnika
5. Promena čina korisnika

The screenshot shows a search form with fields for: Korisničko ime (1), Email (2), Tip korisnika (3), Država (4), Organizacija (5), Min rejting (6), and Max rejting (7). There are also dropdown menus for Tip korisnika and Država.

Sakrij prozor za pretragu

| | | | |
|--|---|---|---|
| Korisničko ime: 1 | Email: 2 | Tip korisnika: 3 | Država: 4 |
| <input type="text" value="Korisničko ime"/> | <input type="text" value="Email"/> | <input type="text" value="Tip korisnika"/> | <input type="text" value="Država"/> |
| Organizacija: 5 | | Min rejting: 6 | Max rejting: 7 |
| <input type="text" value="Organizacija"/> | | <input type="text" value="Min rejting"/> | <input type="text" value="Max rejting"/> |

Korisničko ime Email Tip korisnika

Pretraživanje korisnika se može izvršiti po sledećim kriterijumima:

1. Korisničkim imenima
2. Email adresama
3. Tipovima korisnika
4. Državama
5. Organizacijma
6. Minimalnom rejtingu
7. Maksimalnom rejtingu korisnika

Forma za banovanje izgleda na sledeći način.

Ban Anotheruser

* Ban za korisnika traje do:

Polje za unos datuma do kad je korisnik banovan kao i samo dugme za banovanje.

Admin ima opciju i da promeni čin korisnika iz admina u moderatora i iz moderatora u admina.

Unapredi/Snizi čin korisnika Anotheruser 

Admin Moderator

1. Korisničko ime
2. Email adresa
3. Naziv organizacije
4. Odabir države
5. Unos trenutne lozinke
6. Unos nove lozinke
7. Potvrda nove lozinke
8. Promena slike

Svi navedeni podaci se mogu izmeniti i sačuvati klikom na dugme 9. Klikom na dugme 10 podaci se vraćaju u stanje pre izmene podataka.

Sve ovo se može videti i menjati kada smo na stanici „Izmena profila“.

U poljima:

11. Broj botova korisnika
12. Broj turnira na kojima je učestvovao korisnik ili učestvuje
13. Rejting korisnika

Podaci pod brojevima 11, 12, i 13 se mogu videti i u korisničkoj analizi (15).

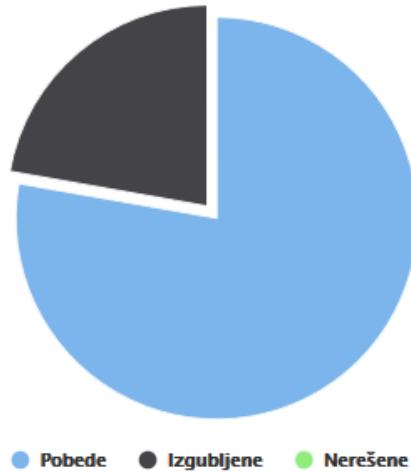
Drugar  1

| | | |
|---|--|-------|
| 1 |  User | 1 / 0 |
| 2 |  Rexville | 1 / 0 |
| 3 |  Whatsapp | 1 / 0 |
| 4 |  Hozumite | 1 / 0 |
| 5 |  Maximillian | 3 / 0 |

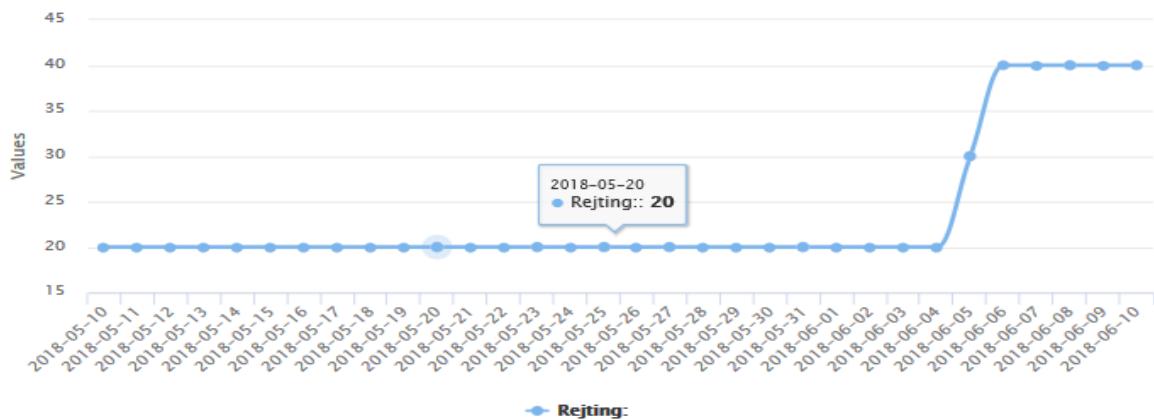
Rival  2

| | | |
|---|---|-------|
| 1 |  Gregory | 0 / 1 |
| 2 |  Rooter | 0 / 1 |

Procenat uspešnosti  3



Rast rejtinga 4

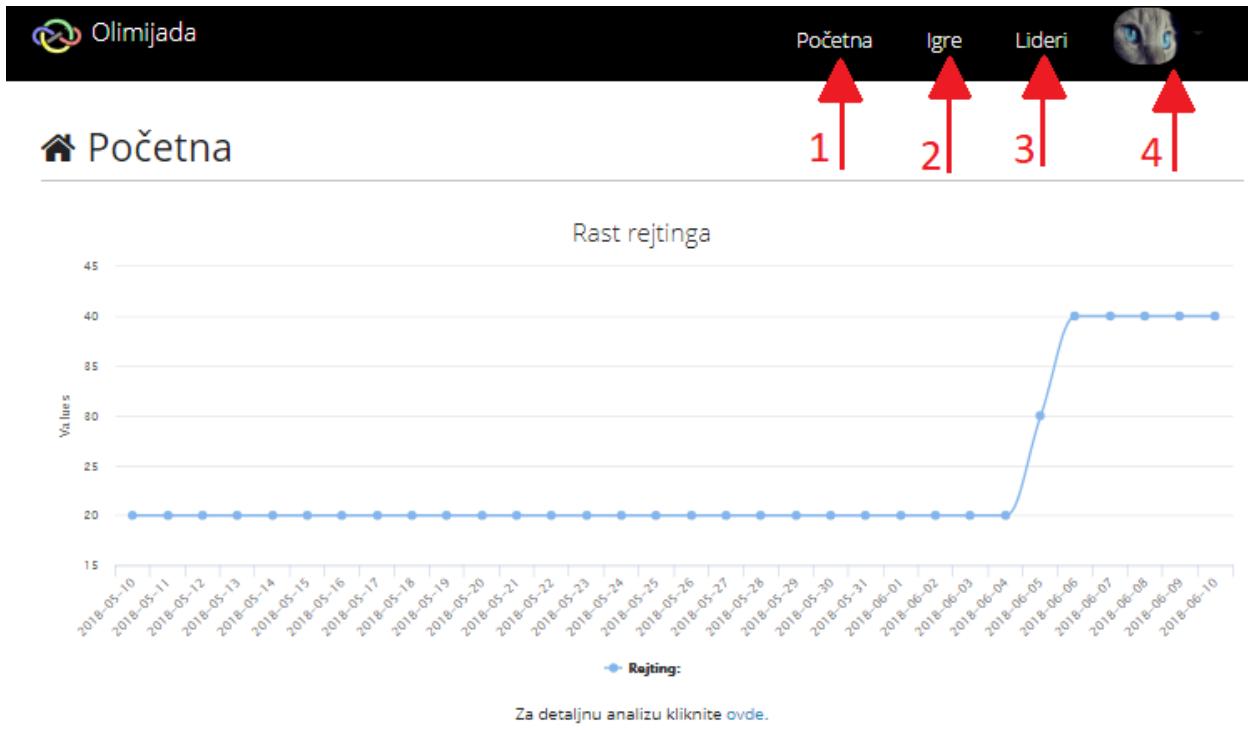


Dobijate 10 poena za svako prvo mesto na turniru i 5 za svako drugo mesto.

Na prethodne dve slike korisnik može videti:

1. Listu od 5 korisnika protiv kojih ima najviše pobjeda
2. Listu od 5 korisnika protiv kojih ima najviše poraza
3. Dijagram koji pokazuje odnos pobjeda, poraza i nerešenih mečeva
4. Dijagram koji pokazuje kako se rejting korisnika kretao u prethodnih mesec dana

Početna strana običnog korisnika:



Igre koje ste najviše igrali:

| | |
|---|---|
| Fudbal  | Defactus  |
| Fudbal is a game played by two teams of eleven players. Players kick the ball to each other and try ... | Defactus is one of the most interesting and difficult games to code a bot for. While Defactus has on... |

Najpopularnije igre:

| | | | |
|--|--|---|---|
| Fudbal  | Defactus  | Delta  | Gamez  |
| Fudbal is a game played by two teams of eleven players. Players kick ... | Defactus is one of the most interesting and difficult games to code a... | The game might seem a bit easy, but solving the game proves to be qui... | Clear all the bubbles and get yourself out of trouble! Destroy the bo... |

Na početnoj strani korisnik može videti podatke sa slike ali i neke koje nisu na njoj.

Podaci koje korisnik može videti redom su:

- Rejting korisnika za proteklih mesec dana
 - 3 igre koje je najviše igrao
 - 4 najpopularnije igre
 - Turnire na kojima je prijavljen, svako polje ima mogućnost da se klikom na njega otvori stranica turnira
 - Turnire koje korisnika možda mogu da zanimaju na osnovu igara, tipova igara koje je najviše igrao..., svako polje ima mogućnost da se klikom na njega otvori stranica turnira
 - Poslednji mečevi korisnika, svako polje ima mogućnost da se klikom na njega otvori novi prozor prikazivanje meča (objašnjeno gore)
-
1. Dugme Početna, korisnik u bilo kom trenutku klikom na njega se može vratiti na početnu stranu
 2. Igre, otvara listu igara sa podacima o njima, kao i turnirima
 3. Lideri, listu korisnika sortiranu po rejtingu sa mogućnosti pretraživanja kao gore navedenom formatu

Lista najuspešnijih korisnika

Sakrij prozor za pretragu

| | | | |
|---|------------------------------------|--|--|
| Korisničko ime: | Email: | Tip korisnika: | Država: |
| <input type="text" value="Korisničko ime"/> | <input type="text" value="Email"/> | <input type="button" value="Tip korisnika"/> | <input type="button" value="Država"/> |
| Organizacija: | | Min rejting: | Max rejting: |
| <input type="text" value="Organizacija"/> | | <input type="button" value="Min rejting"/> | <input type="button" value="Max rejting"/> |

| | Korisničko ime | Email | Tip korisnika | Država | Organizacija | Rejting |
|-----|---|-------------------------|---------------|--------|----------------|---------|
| 1. |  User | lazafiju44@gmail.com | User | Serbia | PMF Kragujevac | 125 |
| 2. |  Maximillian | lazafiju44@gmail.com | User | Serbia | Ubuntu | 55 |
| 3. |  Admin | nestorovic.96@gmail.com | Admin | Serbia | PMF Kragujevac | 40 |
| 4. |  Ravenue | mecak95@gmail.com | User | Serbia | Rockstar | 25 |
| 5. |  Rexville | lazafiju44@gmail.com | User | Serbia | Ubisoft | 25 |
| 6. |  Whatsapp | nestorovic.96@gmail.com | User | | | 20 |
| 7. |  Gregory | mecak95@gmail.com | User | Serbia | Rockstar | 15 |
| 8. |  Ziolin | nestorovic.96@gmail.com | User | | | 15 |
| 9. |  Rooter | nestorovic.96@gmail.com | User | | | 10 |
| 10. |  Hozumite | lazafiju44@gmail.com | User | | | 5 |

« Prethodna strana 1 2 Sledeća strana »

Svaki red tabele ima mogućnost da se klikom na njega otvori stranica pregled korisnika. Stranica pregleda korisnika izgleda na sledeći način. Prva strana koja se otvara je sa podacima korisniku.

1 → Korisnički podaci ← 2

The image shows a user profile interface. At the top, there are two tabs: "Korisnički podaci" (Basic User Data) and "Korisnička analiza" (User Analysis). A red arrow labeled '1' points from the left towards the "Korisnički podaci" tab. Another red arrow labeled '2' points from the "Korisnička analiza" tab back towards the right. Below the tabs is a large circular profile picture placeholder. To the right of the picture, the word "User" is displayed in orange. Below "User" are the user's details: "Serbia" and "PMF Kragujevac". Further down, there are sections for "Botova dodato:" (0) and "Turnira odigrano:" (0). At the bottom, the user's rating is shown as "Rejting: 125".

| | |
|--------------------|----------------------|
| Korisničko ime: | User |
| Email: | lazafiju44@gmail.com |
| Tip korisnika: | User |
| Država: | Serbia |
| Organizacija: | PMF Kragujevac |
| Botova dodato: | Turnira odigrano: |
| 0 | 0 |
| Rejting: 125 | |

1. Otvara stranicu sa osnovnim podacima o korisniku
2. Otvara stranicu sa analizom korisnika, kao na sledećoj slici, podaci koji se ovde mogu videti su kao i kada korisnik udje na svoj profil.

Korisnički podaci Korisnička analiza



User

Serbia

PMF Kragujevac

Botova dodato: 26 Turnira odigrano: 24

Rejting:
125

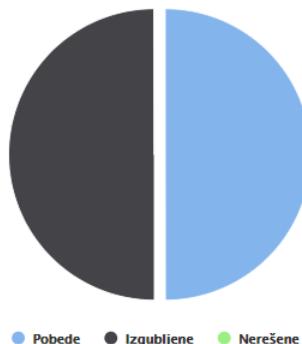
Drugar

| | | |
|---|----------|-------|
| 1 | Rexville | 1 / 0 |
| 2 | Ravenue | 1 / 0 |
| 3 | Whatsapp | 1 / 0 |
| 4 | Ziolin | 1 / 0 |

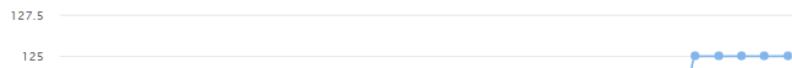
Rival

| | | |
|---|-------------|-------|
| 1 | Admin | 0 / 1 |
| 2 | Gregory | 0 / 1 |
| 3 | Maximillian | 0 / 2 |

Procenat uspešnosti



Rast rejtinga



Klikom na dugme Igre u meniju korisniku se otvara sledeća stranica.

The screenshot shows a web-based game selection interface. At the top, there is a navigation bar with tabs: 'Pretraži igre' (highlighted with a red arrow 1), 'Detalji igre', 'Botovi', and 'Turniri'. Below the navigation bar is a sidebar on the left containing four game cards: 'Defactus' (selected, highlighted with a red box and arrow 2), 'Delta', 'Fudbal', and 'Gamez'. To the right of the sidebar is a large circular icon of a video game controller. Below the icon, the game name 'Defactus' and its genre '--- Adventure ---' are displayed. At the bottom of the page, there are two green buttons: 'Turniri' (with a red arrow 5 pointing to it) and 'Botovi' (with a red arrow 4 pointing to it). Red numbers 3, 4, and 5 also point to the 'Turniri' button.

Defactus is one of the most interesting and difficult games to code a bot for. While Defactus has only a few rules, the game can be quite complicated, as there are countless ways the game can play out.

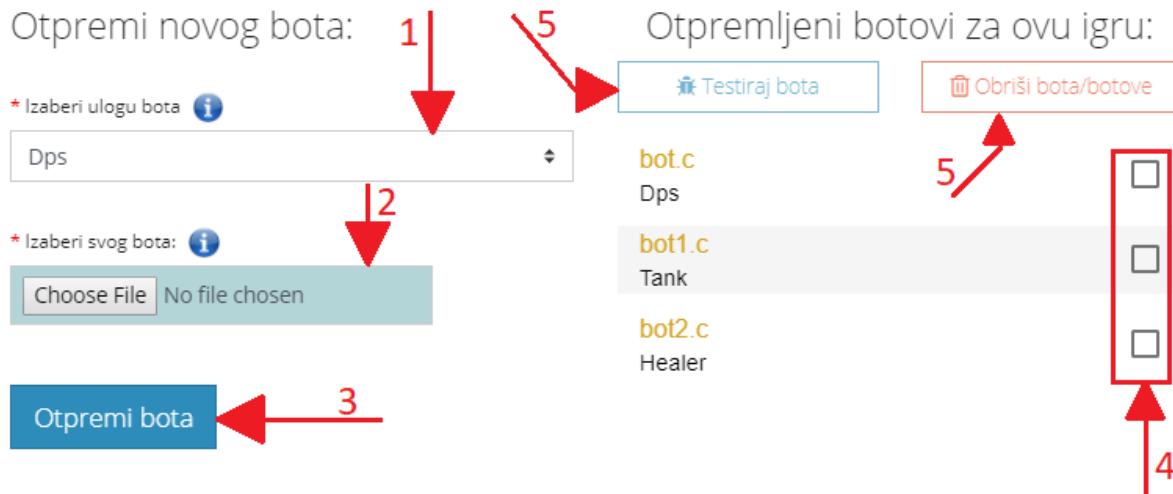
Timska igra (2 igrača)

Prosečno trajanje meča: **4 minut(a)**

Pravila igre

1. Pretraživanje igara po nazivu
2. Lista igra u kojoj korisnika bira igre
3. Otvara stranicu detalji igre kao što je prikazana na slici
4. Otvara stranicu za dodavanje botova, brisanje i testiranje botova
5. Otvara stranicu za pregled turnira

Stranica za dodavanje botova prikazuje sledeću formu.



Ako korisnik želi da doda bota to mora učiniti na sledeći način.

Prvo popuniti polje 1 i 2.

1. Izbor tipa bota/uloge bota
2. Dodavanje fajla bota

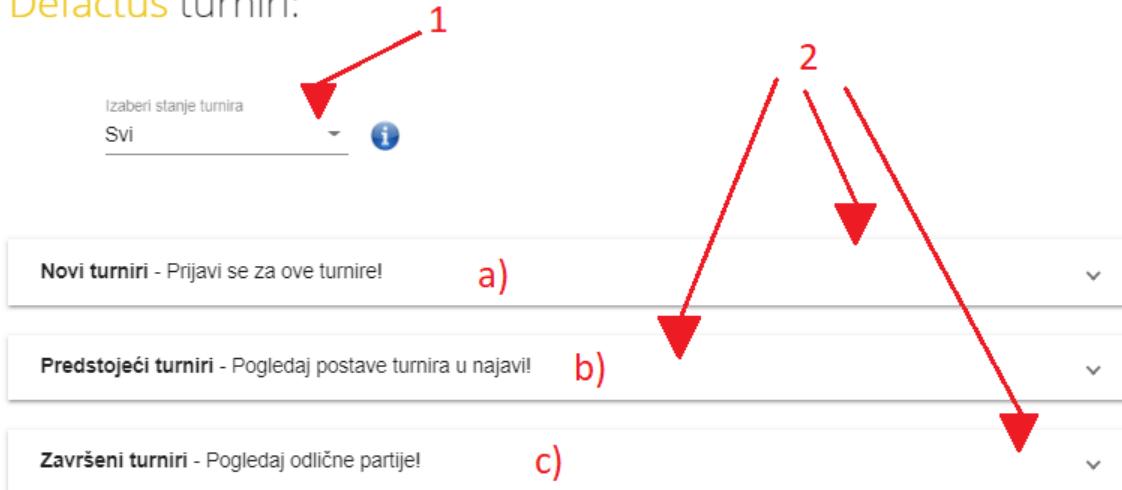
Kada sve to završi korisnik mora pritisnuti dugme 3 da bi izvršio dodavanje bota.

Štikliranjem jednog bota u polju 4 korisnik može testirati svog bota, a štikliranjem jednog ili više botova korisnik može obrisati te botove.

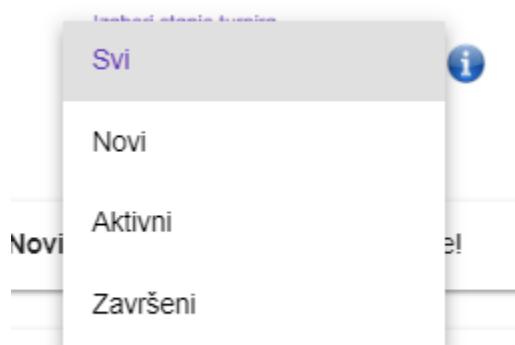
Desno možemo videti listu trenutnih botova korisnika za izabranu igru, sa nazivom fajla bota i tipom/ulogom bota.

Klikom na turniri otvara se pregled turnira.

Defactus turniri:



Klikom na polje 1 otvara se padajuća lista sa svim stanjima turnira (Novi, Aktivni, Završeni, i lista svih tunira).



Predstojeći turniri - Pogledaj postave turnira

Klikom na neka od polja 2 otvara se lista turnira za izabrano stanje.

- Novi turniri – turniri kojima i dalje traje prijava
- Aktivni turniri – turniri koji su u toku igranja
- Završeni turniri – turniri koji su završeni

Pregled izabranog turnira koji su završeni i koji su aktivni izgleda kao u gore navednom delu. Prijava na turnir zahteva popunjavanje sledeće forme.

PROBA **1**

Krajnji datum: **2**
Aug 10, 2018, 7:29:00 PM

Turnir počinje: **3**
Aug 10, 2018, 7:31:00 PM

Napravi igrače i registruj se

Prijavljeni korisnici

Ime igrača

Nema prijavljenih korisnika za ovaj turnir.

4

- bot.c** - Dps
- bot1.c** - Tank
- bot2.c** - Healer

5

Dodaj novog bota

6

Dodaj igrača

7

Broj igrača: 1/2

Tim

Bot

Pera

bot.c - DPS



8

Prijava se na turnir

9

Polja o podacima turnira:

1. Naziv turnira
2. Datum kraja prijave na turnir
3. Datum početka turnira

Da bi dodao igrača a zatim i prijavio na turnir, korisnik je u obavezi da popuni polja 4 i 5.

4. Naziv igrača
5. Odabir bota

Ukoliko korisniku nijedan od trenutnih botova ne odgovara on ima mogućnost dodavanja bota bez prelaska na drugu stranicu. Klikom na dugme 6 otvara se mali prozor kao na sledećoj slici.

bot.c - Dps

bot1.c - Tank

bot2.c - Healer

1

2

3

4

* Izaberite ulogu bota i

Dps

* Izaberite svog bota i

Choose File No file chosen

Otpremi bota Zatvori prozor

Dodaj igrača

Broj igrača: 1/2

1. Odabir tipa bota/uloga bota
2. Dodavanje fajla bota

Da bi dodao bota korisnik mora pritisnuti dugme 3. Ako želi zatvoriti prozor pritisnuti dugme 4.

Kada je korisnik popunio polje 4 i 5 može dodati igrača na dugme 7. U tabeli 8 korisnik može videti broj igrača koliko je do sad dodaš i listu igrača, koje može obrisati na „X“.

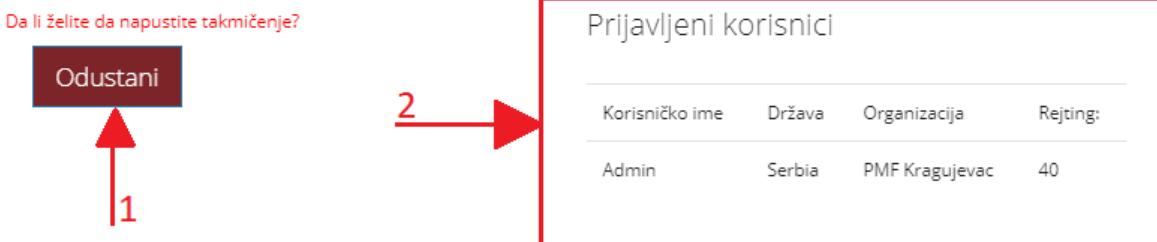
Da bi se korisnik prijavio mora imati tačno određeni broj igrača, koji piše u tabeli. Nakon ispunjavanja gore navedneih uslova prijavljivanje na turnir korisnik vrši na dugme 9.

Ukoliko je korisnik već prijavljen na turnir i želi da prijavi druge igrače, on prvo mora odustati od turnira.

PROBA

Krajnji datum:
Aug 10, 2018, 7:29:00 PM

Turnir počinje:
Aug 10, 2018, 7:31:00 PM



Odustajanje od turnira može uraditi na dugme 1. U tabeli 2 korisnik može videti sve korisnike koji su prijavljeni na taj turnir.

Korisnik na svom email-u dobija obaveštenje kada se završi prijava na turnir na koji se on prijavio, kada se završi turnir na kom on učestvuje, kada se turnir obriše a on je bio prijavljen, kada se produži prijava za turnir.



U futeru korisnik može videti sledeće podatke.



Klikom na polje 1 korisnik može skinuti korisničko upustvo.

Klikom jedno od polja 3 korisnik može da otovri naše stranice na društvenim mrežama.

Klikom na polje 2 korisniku će se otvoriti stranica o članovima tima koji su pravili aplikaciju.

O nama

Pure Logic tim je novoosnovani tim koji sačinjavaju četiri člana.

Na čelu tima je Stefan Nestorović, a ostali članovi su Lazar Stanojević, Dragan Đurović i Strahinja Milutinović

Stefan Nestorović - Student treće godine informatike na Prirodno-matematičkom fakultetu u Kragujevcu, sa prosekom preko 9.7. Na poziciji team - leadera uspešno je organizovao tim na mnogobrojnim zadacima.

