

# Programsko inženjerstvo

Ak. god. 2023./2024.

## BytePit

Dokumentacija, Rev. <1 ili 2>

Grupa: *Looney Codes*

Voditelj: *Vedran Ćutić*

Datum predaje: <dan>. <mjesec>. <godina>.

Nastavnik: *Hrvoje Nuić*

# Sadržaj

<b>1</b>	<b>Dnevnik promjena dokumentacije</b>	<b>3</b>
<b>2</b>	<b>Opis projektnog zadatka</b>	<b>5</b>
<b>3</b>	<b>Specifikacija programske potpore</b>	<b>11</b>
3.1	Funkcionalni zahtjevi . . . . .	11
3.1.1	Obrasci uporabe . . . . .	13
3.1.2	Sekvencijski dijagrami . . . . .	24
3.2	Ostali zahtjevi . . . . .	30
<b>4</b>	<b>Arhitektura i dizajn sustava</b>	<b>31</b>
4.1	Baza podataka . . . . .	32
4.1.1	Opis tablica . . . . .	32
4.1.2	Dijagram baze podataka . . . . .	33
4.2	Dijagram razreda . . . . .	34
4.3	Dijagram stanja . . . . .	35
4.4	Dijagram aktivnosti . . . . .	36
4.5	Dijagram komponenti . . . . .	37
<b>5</b>	<b>Implementacija i korisničko sučelje</b>	<b>38</b>
5.1	Korištene tehnologije i alati . . . . .	38
5.2	Ispitivanje programskog rješenja . . . . .	39
5.2.1	Ispitivanje komponenti . . . . .	39
5.2.2	Ispitivanje sustava . . . . .	39
5.3	Dijagram razmještaja . . . . .	40
5.4	Upute za puštanje u pogon . . . . .	41
<b>6</b>	<b>Zaključak i budući rad</b>	<b>42</b>
	<b>Popis literature</b>	<b>43</b>
	<b>Indeks slika i dijagrama</b>	<b>44</b>

**Dodatak: Prikaz aktivnosti grupe**

**45**

# 1. Dnevnik promjena dokumentacije

## *Kontinuirano osvježavanje*

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Preuzet i postavljen predložak.	Nikola Vlahović	20.10.2023.
0.2	Dodan opis projektnog zadatke. Upisani članovi.	Marko Varga	21.10.2023.
0.3	Nadopisan opis projektnog zadatka.	Marina Hrbud	22.10.2023.
0.4	Dodani funkcionalni zahtjevi	Lara Marčec	23.10.2023.
0.5	Dodani opisi obrazaca uporabe	Nikola Vlahović, Lara Marčec	29.10.2023.
0.6	Dodani dijagrami obrazaca uporabe	Jakov Novak, Marko Varga	29.10.2023.
0.7	Dodani opisi ostalih zahtjeva	Lara Marčec	30.10.2023.
0.8	Dodani sekvencijski dijagrami	Lara Marčec	2.11.2023.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
0.9	Dodan opis arhitekture i dizajna sustava	Antonio Glavaš	4.11.2023.

*Moraju postojati glavne revizije dokumenata 1.0 i 2.0 na kraju prvog i drugog ciklusa. Između tih revizija mogu postojati manje revizije već prema tome kako se dokument bude nadopunjavao. Očekuje se da nakon svake značajnije promjene (dodatka, izmjene, uklanjanja dijelova teksta i popratnih grafičkih sadržaja) dokumenta se to zabilježi kao revizija. Npr., revizije unutar prvog ciklusa će imati oznake 0.1, 0.2, ..., 0.9, 0.10, 0.11.. sve do konačne revizije prvog ciklusa 1.0. U drugom ciklusu se nastavlja s revizijama 1.1, 1.2, itd.*

## 2. Opis projektnog zadatka

Projektni zadatak jest razvoj programske podrške za ostvarenje mrežne aplikacije BytePit. Aplikacija predstavlja platformu za rješavanje programerskih zadataka i provođenje natjecanja u programiranju. Aplikacija BytePit pruža ključne mogućnosti kao što su:

- *registracija natjecatelja i voditelja natjecanja*
- *dijeljenje osmišljenih zadataka iz natjecateljskog programiranja*
- *rješavanje zadataka i evaluacija rješenja na temelju testnih primjera*
- *kreiranje i provođenje natjecanja.*

Zbog ovih je mogućnosti aplikacija BytePit od posebnog interesa za odgojno-obrazovne ustanove koje alat mogu koristiti kao dodatak u nastavi predmeta povezanih uz programiranje (provođenje ispita ili domaćih zadaća), sudionike natjecanja u programiranju koji na platformi mogu u slobodno vrijeme vježbati i usavršavati svoje vještine te udruge i druga tijela koja se bave organizacijom natjecanja iz programiranja ili pak promocijom računalne znanosti. IT kompanije bi ovu aplikaciju mogle upotrebljavati kako bi ispitale relevantna znanja i programerske sposobnosti potencijalnih novih zaposlenika.

Aplikacija na početnoj stranici mora nuditi mogućnost registracije novim korisnicima. Proces registracije, uz mogućnost odabira uloge (voditelj natjecanja ili natjecatelj), zahtijeva unos korisničkog imena, fotografije, lozinke, imena, prezimena i adrese e-pošte. Za završetak registracije potrebno je otvoriti poveznicu dobivenu putem e-pošte. Voditeljevu registraciju mora dodatno odobriti administrator, koji pored toga ima ovlasti za upravljanje registriranim korisnicima i njihovim podacima, uključujući dodjelu prava i promjene osobnih podataka.

Tri su osnovna tipa korisnika aplikacije:

- **natjecatelji** — Natjecatelji su osnovni korisnici aplikacije i njezina glavna ciljna skupina. Osnovne usluge koje im se pružaju su sudjelovanje u programerskim natjecanjima, rješavanje zadataka i praćenje svog napretka. Na zahtjev mogu pristupiti zadacima (tijekom ili izvan termina natjecanja) . Mogu

slati programska rješenja na evaluaciju, a nakon natjecanja pratiti i rangiranje s obzirom na ostale natjecatelje. Na profilima natjecatelja, vidljive su različite statistike, uključujući broj točno riješenih zadataka, broj pokušaja rješavanja zadataka te prikaz osvojenih pehara za natjecanja. Natjecatelji mogu organizirati i virtualna natjecanja, koja se temelje na prošlim natjecanjima u kalendaru. Ta natjecanja su aktivna samo za njih, a rangiraju se prema službenim rezultatima originalnih natjecanja. Također, mogu kreirati virtualna natjecanja s nasumično odabranim zadacima, ravnomjerno raspoređenim prema težini zadataka.

- **voditelji** — Voditelji natjecanja organiziraju i upravljaju programerskim natjecanjima na platformi. To uključuje postavljanje natjecanja, dodavanje zadataka i praćenje rezultata sudionika. Prilikom kreiranja novih natjecanja moraju postaviti parametre, kao što su datum objave i trajanje natjecanja, dodati zadatke i po želji slike pehara. Dodavanje programerskih zadataka u natjecanja uključuje i postavljanje njihovih karakteristika – bodova i vremenskog ograničenja, ali i testnih primjera. Profili voditelja sadrže popis zadataka koje su učitali, s mogućnošću sortiranja.
- **administratori** — Administratori imaju ovlasti za upravljanje cijelom platformom, uključujući i korisničke račune. Upravljanje registriranim korisnicima i njihovim podacima uključuje dodjelu prava (primjerice potvrdu registracije voditelja) i promjene osobnih podataka korisnika. Administrator ima ovlasti za uređivanje svih zadataka i natjecanja pri čemu se se ne mijenjaju prethodno ostvareni rezultati.

Središnji dijelovi aplikacije su *zadaci* i *natjecanja*. Programerski zadaci su ključna komponenta aplikacije i pružaju izazove korisnicima koji žele razvijati svoje programerske sposobnosti. Svaki zadatak dolazi s nazivom, jedinstvenim opisom, brojem bodova koji se mogu osvojiti i vremenskim ograničenjem za rješavanje. Zadatak može biti postavljen kao privatan ili javan, a svaki privatni zadatak dodan u natjecanje, po završetku istog, postaje javan. Natjecatelji biraju zadatke koji ih zanimaju (ili koji su dio natjecanja) i šalju svoje rješenje u obliku izvornog programskog koda. Evaluacija rješenja (omogućena samo za jedan programski jezik) temelji se na usporedbi generiranih izlaza programa s očekivanim izlazima iz testnih primjera. Dodijeljeni broj bodova proporcionalan je postotku točnih primjera. Za svaki zadatak dostupan je popis svih natjecatelja koji su poslali rješenje za taj zadatak.

tak. Ovaj popis sadrži informacije o broju točno riješenih primjera, prosječnom vremenu izvršavanja po primjeru te omogućava pristup učitanim rješenjima. Važno je napomenuti da opcija za pristup učitanim rješenjima aktivira samo za one natjecatelje koji su uspješno i potpuno riješili zadatak.

Natjecanja predstavljaju srce platforme, potičući korisnike da pokažu svoje programerske vještine i natječu se s drugima. Voditelji natjecanja imaju ključnu ulogu u organizaciji ovih događaja. Oni kreiraju i definiraju ključne elemente natjecanja. Natjecatelji se prijavljuju za sudjelovanje u natjecanjima, a tijekom trajanja rješavaju zadatke i šalju svoja rješenja. Početkom natjecanja prijavljeni natjecatelji mogu pristupiti svim zadacima koji se nalaze u njemu. Nakon završetka natjecanja, rezultati se prikazuju na profilima natjecatelja, a osvajači pehara nagrađuju se za svoje uspjehe. Također, natjecatelji imaju mogućnost pregledavanja svih rješenja koja su drugi natjecatelji poslali za isti zadatak.

Rješenje za ovu aplikaciju biti će ostvareno pomoću razvojnog okvira Springboota i JavaScript biblioteka React kojom će biti modelirano korisničko sučelje.

Moguća poboljšanja za aplikaciju BytePit uključuju unaprjeđenje korisničkog sučelja radi olakšane navigacije, uvođenje ocjena i komentara za zadatke i natjecanja, implementaciju sustava za korisničku komunikaciju putem chatova ili foruma, bolje statističke podatke i analitiku na korisničkim profilima, integraciju s razvojnim alatima, raznolikost vrsta zadataka, napredniju evaluaciju rješenja s podrškom za različite programske jezike, te dodatak edukativnih materijala kako bi se potaknulo učenje i razvoj programerskih vještina među korisnicima.

Primjeri sličnih rješenja kroz koje ćemo proći su:

- *Edgar*
- *Sphere online judge*
- *Codeforces*

Kao i u sustavima *Edgar*, *SPOJ* i *Codeforces*, korisnici će moći pregledavati zadatke na stranici, te će moći na svom profilu vidjeti statistiku u odnosu na preostale natjecatelje, te svoj rank. Također mogu na kalendaru vidjeti vrijeme odvijanja budućih natjecanja.



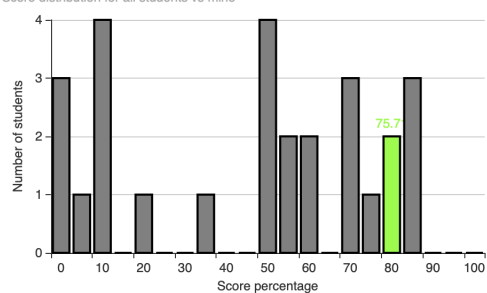
## list of classical problems

ID	NAME	QUALITY	USERS	ACC %	DIFFICULTY C I
1	Life, the Universe, and Everything	👍 294	207076	32.89	{ }
2	Prime Generator	👍 616	75846	15.94	15 13
3	Substring Check (Bug Funny)	👍 9	1057	7.99	41 13
4	Transform the Expression	👍 257	41908	49.17	1 1
5	The Next Palindrome	👍 450	19715	10.82	25 18
6	Simple Arithmetics	👍 10	2978	12.49	30 9
7	The Bulk!	👍 6	476	16.32	30 24
8	Complete the Sequence!	👍 30	4249	38.15	16 19
9	Direct Visibility	👍 4	247	12.48	30 21
10	Complicated Expressions	👍 11	1515	32.43	16 17
11	Factorial	👍 171	56448	44.98	1 1
12	The Game of Master-Mind	👍 1	717	49.89	17 17

Slika 2.1: Sustav SPOJ - pregled zadatka na stranici

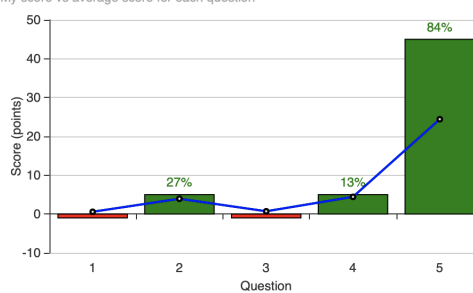
## Exam score percentage distribution (n = 31)

Score distribution for all students vs mine



## Per Question

My score vs average score for each question



Rank

#6 /31

83%

Percentile rank

Score

53 /70

76%

Percentage

Slika 2.2: Sustav Edgar - pregled statistike ispita

<a href="#">MAIN</a> <a href="#">ACMSGURU</a> <a href="#">PROBLEMS</a> <a href="#">SUBMIT</a> <a href="#">STATUS</a> <a href="#">STANDINGS</a> <a href="#">CUSTOM TEST</a>							
Contest status							
#	When	Who	Problem	Lang	Verdict	Time	Memory
<a href="#">229301890</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	~JAAT~	<a href="#">1883A - Morning</a>	GNU C++17	Accepted	31 ms	0 KB
<a href="#">229301889</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	<a href="#">Part_time_Ray</a>	<a href="#">1850C - Word on the Paper</a>	GNU C++17 (64)	Accepted	15 ms	0 KB
<a href="#">229301887</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	bkifhr9	<a href="#">1873B - Good Kid</a>	GNU C++17	Time limit exceeded on test 1	1000 ms	0 KB
<a href="#">229301886</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	manish36	<a href="#">486A - Calculating Function</a>	GNU C++17	Accepted	15 ms	0 KB
<a href="#">229301885</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	<a href="#">regain0002</a>	<a href="#">1888D2 - Dances (Hard Version)</a>	GNU C++17	Wrong answer on test 2	46 ms	37600 KB
<a href="#">229301883</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	P.N.V.Sumanasree	<a href="#">50A - Domino piling</a>	GNU C++20 (64)	Accepted	30 ms	0 KB
<a href="#">229301882</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	<a href="#">_SAK_</a>	<a href="#">1883E - Look Back</a>	GNU C++17	Accepted	93 ms	1200 KB
<a href="#">229301881</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	ValeriaDubinets	<a href="#">1808A - Lucky Numbers</a>	GNU C++20 (64)	Time limit exceeded on test 2	1000 ms	0 KB
<a href="#">229301880</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	Huihuilucky	<a href="#">1791C - Prepend and Append</a>	GNU C++17	Accepted	31 ms	0 KB
<a href="#">229301879</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	<a href="#">xgyxgy</a>	<a href="#">803B - Distances to Zero</a>	GNU C++20 (64)	Wrong answer on test 13	15 ms	800 KB
<a href="#">229301878</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	vjudge3	<a href="#">1538C - Number of Pairs</a>	Clang++17 Diagnostics	Wrong answer on test 2	514 ms	2100 KB
<a href="#">229301876</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	imaginesabab	<a href="#">520A - Pangram</a>	GNU C++17	Accepted	15 ms	0 KB
<a href="#">229301875</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	<a href="#">gdaaay</a>	<a href="#">1883C - Raspberries</a>	GNU C++20 (64)	Accepted	15 ms	0 KB
<a href="#">229301874</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	rohitrojo	<a href="#">1863A - Channel</a>	GNU C++20 (64)	Wrong answer on test 1	15 ms	0 KB
<a href="#">229301873</a>	Oct/22/2023 16:35 <sup>UTC+2</sup>	<a href="#">3inf</a>	<a href="#">1884C - Medium Design</a>	GNU C++17	Accepted	499 ms	7300 KB

Slika 2.3: Sustav Codeforces - pregled statusa na natjecanju

## Calendar

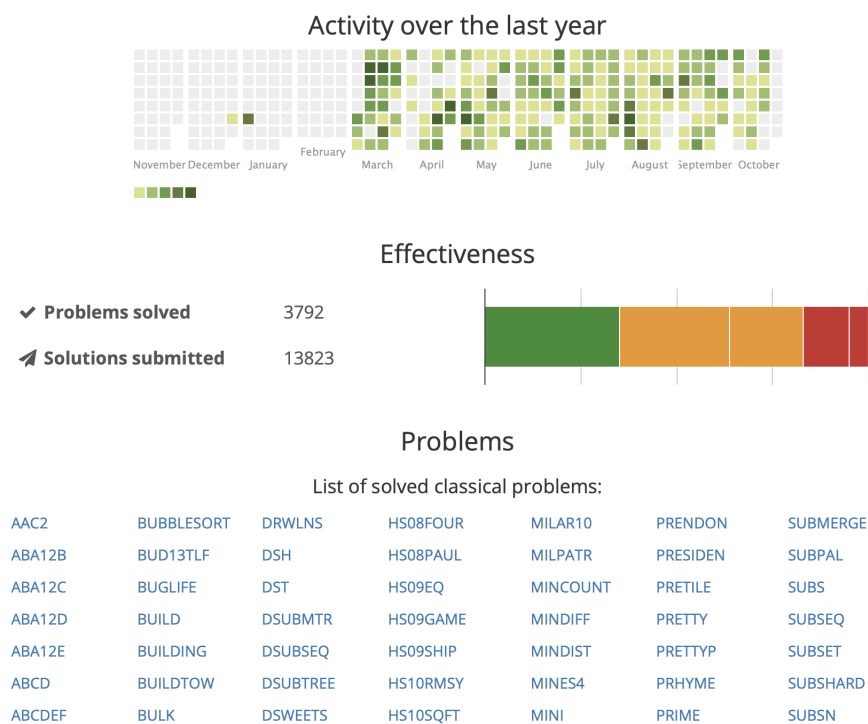
<b>Programming Contests Calendar</b> Today <span>◀</span> <span>▶</span> <b>October 2023</b> <span>▼</span> <span>Print</span> <span>Week</span> <span>Month</span> <span>Agenda</span> <span>▼</span>							
Sun	Mon	Tue	Wed	Thu	Fri	Sat	
Oct 1	2	3	4	5	6	7	
8	9	10	11	12	13	14	
11:05am CF R (Div. 1, based on 11:05am CF R 902)	4:35pm Educational CF R 156	8:15pm Stream: Teaching Ch		4:35pm CF R 903 (Div. 3)			
15	16	17	18	19	20	21	
9:45pm Stream: No code CF							
22	23	24	25	26	27	28	
9:05am CF R 904 (Div. 2) 1:05pm CF R 905						4:35pm CF R	
29	30	31	Nov 1	2	3	4	
	3:35pm CF R (Div. 2)						

Events shown in time zone: Central European Time - Belgrade



Slika 2.4: Sustav Codeforces - kalendar nadolazećih natjecanja

Ukoliko natjecatelj zanima profil nekog drugog natjecatelja, klikom na ime profila će se otvoriti stranica na kojoj je moguće vidjeti statistiku natjecatelja: broj točno riješenih zadataka, pehari, broj svih pokrenutih zadataka. *Codeforces* dodatno daje prikaz aktivnosti natjecatelja u vremenskom periodu od jedne godine.



Slika 2.5: Sustav SPOJ - profil natjecatelja

## 3. Specifikacija programske potpore

### 3.1 Funkcionalni zahtjevi

#### Dionici:

1. Voditelji
2. Natjecatelji
3. Administrator
4. Razvojni tim

#### Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (inicijator) može:
  - (a) vidjeti kalendar s budućim natjecanjima
  - (b) pregledati dostupne zadatke na stranici
  - (c) pregledati profile natjecatelja i voditelja
  - (d) registrirati se u sustav stvaranjem novog korisničkog računa pri čemu odabire jednu od uloga (natjecatelj ili voditelj), a potrebni su mu: korisničko ime, fotografija, lozinka, ime, prezime te email adresa
2. Natjecatelj (inicijator) može:
  - (a) sudjelovati na natjecanju
  - (b) vidjeti rang listu natjecatelja na natjecanju kojeg je i sam bio sudionik
  - (c) vidjeti popis svih učitanih rješenja od ostalih sudionika za prethodno završena natjecanja
  - (d) pristupiti rješavanju već objavljenih zadataka
  - (e) izraditi virtualno natjecanje odabirom nekog prošlog natjecanja ili nasumičnim generiranjem zadataka
  - (f) vidjeti vlastiti profil s osobnim podacima, statistikom o broju točno riješenih zadataka, broju isprobanih zadataka te prikaz pehara za osvojena natjecanja
3. Aktivni natjecatelj (inicijator) može:

- (a) rješavati zadatke i slati datoteke s programskim kodom tijekom natjecanja na kojem se natječe
- (b) osvojiti bodove na natjecanju s obzirom na potrošeno vrijeme za rješavanje zadatka i postotak točnih primjera
- (c) na temelju postignuća za prva tri mjesta osvojiti pehar koji je vidljiv na vlastitom profilu

4. Voditelj (inicijator) može:

- (a) pregledati dostupne zadatke na stranici
- (b) izraditi novi zadatak pri čemu treba definirati: naziv zadatka, broj bodova, vremensko ograničenje, tekst zadatka i primjere za evaluaciju
- (c) organizirati novo natjecanje pri čemu treba definirati: vrijeme početka i završetka, broj zadataka, koji zadaci će biti aktivni te po želji sličicu pehara
- (d) uređivati vlastite prethodno objavljene zadatke te natjecanja
- (e) vidjeti vlastiti profil s osobnim podacima, popisom učitanih zadataka s mogućnošću sortiranja te kalendar s popisom objavljenih natjecanja

5. Administrator (inicijator) može:

- (a) uređivati sve zadatke i natjecanja
- (b) potvrditi voditelja prilikom registracije
- (c) vidjeti popis svih registriranih korisnika i njihovih osobnih podataka
- (d) mijenjati dodijeljena prava i osobne podatke

6. Baza podataka (sudionik) može:

- (a) pohranjuje sve podatke o korisnicima i njihovim ovlastima te zadacima i natjecanjima
- (b) pohranjuje rezultate natjecanja, rješenja zadataka i statistiku natjecatelja

### 3.1.1 Obrasci uporabe

#### *dio 1. revizije*

#### Opis obrazaca uporabe

*Funkcionalne zahtjeve razraditi u obliku obrazaca uporabe. Svaki obrazac je potrebno razraditi prema donjem predlošku. Ukoliko u nekom koraku može doći do odstupanja, potrebno je to odstupanje opisati i po mogućnosti ponuditi rješenje kojim bi se tijekom obrasca vratio na osnovni tijek.*

#### UC1 - Pregled kalendara

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregledati kalendar s nadolazećim natjecanjima
- **Sudionici:** baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
  1. Korisnik otvara početnu stranicu web aplikacije
  2. Prikazuje se kalendar
  3. Korisnik odabire određeni datum
  4. Aplikacija prikazuje popis nadolazećih natjecanja za odabrani datum
  5. Korisnik odabire natjecanje
  6. Prikazuju se detalji i informacije o odabranom natjecanju

#### UC2 - Pregled zadataka

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregledati zadatke završenih natjecanja
- **Sudionici:** baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
  1. Korisnik odabire završeno natjecanja
  2. Prikazuje se lista zadatka povezana s odabranim natjecanjem
  3. Korisnik odabire specifičan zadatak
  4. Aplikacija prikazuje detalje zadatka

#### UC3 - Pregled profila natjecatelja i voditelja

- **Glavni sudionik:** Korisnik

- **Cilj:** Pregledati profile natjecatelja i voditelja
- **Sudionici:** baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
  1. Korisnik otvara profil određenog korisnika klikom na njihovo korisničko ime
  2. Ako je otvoren profil natjecatelja, aplikacija prikazuje informacije o broju točno riješenih zadataka, broju isprobanih zadataka te osvojene pehare
  3. Ako je otvoren profil voditelja, aplikacija prikazuje popis učitanih zadataka i kalendar s popisom objavljenih natjecanja

#### UC4 - Registracija

- **Glavni sudionik:** Korisnik
- **Cilj:** Stvoriti korisnički račun
- **Sudionici:** baza podataka, administrator
- **Preduvjet:** korisnik nije prethodno registriran ili prijavljen u sustav
- **Opis osnovnog tijeka:**
  1. Korisnik odabire opciju za registraciju
  2. Korisnik popunjava obrazac za registraciju s potrebnim podacima
  3. Korisnik na svoju e-mail adresu prima obavijest i zahtjev za potvrdu registracije
  4. Ako je korisnik odabrao ulogu "voditelj", sustav dodatno šalje obavijest administratoru
  5. Administrator potvrđuje registraciju voditelja
- **Opis mogućih odstupanja:**
  - 2.a Već zauzeto korisničko ime ili e-mail adresa, uneseni podaci u nedozvoljenom formatu ili neispravna e-mail adresa
    1. Sustav obavještava korisnika o neuspjelom unosu i prikazuje relevantne poruke o greškama
    2. Korisnik mijenja potrebne podatke te završava unos ili odustaje od registracije
  - 5.a Administrator odbija zahtjev za registraciju voditelja natjecanja:
    1. Sustav obavještava korisnika putem e-maila

#### UC5 - Prijava

- **Glavni sudionik:** Korisnik

- **Cilj:** Dobiti pristup korisničkom sučelju
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik je registriran u sustav, ali nije prijavljen
- **Opis osnovnog tijeka:**
  1. Korisnik unosi korisničko ime i lozinku
  2. Sustav potvrđuje ispravnosti unesenih podataka
  3. Korisniku je omogućen pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
  - 2.a Neispravno korisničko ime/lozinka
    1. Sustav obavještava korisnika o neuspjehom upisa i vraća ga na stranicu za registraciju

#### UC6 - Pregled profila

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregledati vlastiti profil
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik je prijavljen
- **Opis osnovnog tijeka:**
  1. Korisnik pristupa opciji "Moj profil" klikom na ikonu profila
  2. Aplikacija prikazuje podatke o korisniku (ime, fotografija, lozinka, ime, prezime i e-mail adresa)

#### UC7 - Pregled vlastite statistike

- **Glavni sudionik:** Korisnik (natjecatelj)
- **Cilj:** Pregledati vlastitu statistiku unutar web aplikacije
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik je prijavljen
- **Opis osnovnog tijeka:**
  1. Korisnik pristupa svojem profilu i odabire opciju "Moje statistike"
  2. Aplikacija prikazuje statistike korisnika (broj točno riješenih zadataka, broj isprobanih zadataka, pehare za osvojena natjecanja)

#### UC8 - Pregled vlastitih zadataka

- **Glavni sudionik:** Korisnik (voditelj)
- **Cilj:** Pregledati popis svih vlastito objavljenih zadataka
- **Sudionici:** baza podataka



- **Preduvjet:** korisnik je prijavljen
- **Opis osnovnog tijeka:**
  1. Voditelj odabire opciju za prikaz vlastitih zadataka
  2. Prikazuju se svi objavljeni zadaci tog voditelja

#### UC9 – Sudjelovanje na natjecanju

- **Glavni sudionik:** Korisnik (natjecatelj)
- **Cilj:** Pristupiti natjecanju
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik je prijavljen i postoji natjecanje u tijeku
- **Opis osnovnog tijeka:**
  1. Prikazuju se informacije o natjecanju u tijeku i natjecatelj potvrđuje da želi sudjelovati na natjecanju
  2. Natjecatelju se prikazuju zadaci koji su dio natjecanja te preostalo vrijeme do kraja natjecanja

#### UC10 – Prijenos datoteke rješenja

- **Glavni sudionik:** Korisnik (natjecatelj, aktivni natjecatelj)
- **Cilj:** Prenijeti datoteku kao rješenje nekog zadatka na natjecanju
- **Sudionici:** baza podataka
- **Preduvjet:** natjecatelj je pristupio natjecanju ili rješavanju zadataka
- **Opis osnovnog tijeka:**
  1. Korisnik nakon napisanog rješenja zadatka odabire opciju za predaju rješenja
  2. Korisnik odabire datoteku s rješenjem koju želi predati i potvrđuje odabir
  3. Datoteka se prenosi, predaje kao rješenje i šalje na evaluaciju
- **Opis mogućih odstupanja:**
  - 2.a Odabir pogrešne datoteke
    1. Aktivni natjecatelj odabire opciju za brisanje datoteke
    2. Aktivni natjecatelj ponovno kreće s 1. korakom iz osnovnog tijeka

#### UC11 – Pregled rang liste

- **Glavni sudionik:** Korisnik (natjecatelj)
- **Cilj:** Pregledati rang listu nekog natjecanja
- **Sudionici:** baza podataka

- **Preduvjet:** korisnik je prijavljen kao natjecatelj
- **Opis osnovnog tijeka:**
  1. Natjecatelj odabire opciju za prikaz prošlih natjecanja na kojima je sudjelovao
  2. Odabire natjecanje za koje želi vidjeti rang listu ostalih sudionika i odabire prikaz rang liste
  3. Otvara se popis s prikazom postignuća ostalih sudionika po broju ostvarenih bodova

#### UC12 – Pregled rješenja zadataka

- **Glavni sudionik:** Korisnik (natjecatelj)
- **Cilj:** Vidjeti tuđa rješenja zadataka s nekog natjecanja
- **Sudionici:** Baza podataka
- **Preduvjet:** Natjecatelj je sudjelovao na natjecanju koje je završilo
- **Opis osnovnog tijeka:**
  1. Natjecatelj odabire opciju za prikaz prošlih natjecanja na kojima je sudjelovao
  2. Natjecatelj odabire natjecanje za koje želi vidjeti predana rješenja
  3. Prikazuju se informacije o natjecanju s popisom svih predanih rješenja
  4. Odabirom opcije za prikaz natjecanja po zadacima prikazuje se pregled zadataka s tog natjecanja
  5. Odabirom zadatka otvara se tekst zadatka s prikazom svih predanih rješenja drugih sudionika uz informacije o uspješnosti predanog rješenja

#### UC13 – Rješavanje zadataka

- **Glavni sudionik:** Korisnik (natjecatelj)
- **Cilj:** Vježbanje već objavljenih zadataka
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik je prijavljen kao natjecatelj
- **Opis osnovnog tijeka:**
  1. Natjecatelj odabire opciju za prikaz svih objavljenih zadataka
  2. Prikazuje se popis svih zadataka
  3. Odabirom zadatka koji želi rješavati otvara se sučelje slično onom na natjecanju koje natjecatelju omogućuje učitavanje datoteke kao rješenja zadataka i evaluiranje istog

#### UC14 – Izrada virtualnog natjecanja

- **Glavni sudionik:** Korisnik (natjecatelj)
- **Cilj:** Vježbanje simuliranjem pravog natjecanja
- **Sudionici:** baza podataka
- **Preduvjet:** postoji barem jedno završeno natjecanje i/ili barem jedan javno vidljiv zadatak u bazi
- **Opis osnovnog tijeka:**
  1. Natjecatelj odabire opciju “virtualno natjecanje”
  2. Otvara se prikaz s dvije mogućnosti odabira
  3. Natjecatelj odabire jednu od dvije opcije - stvaranje natjecanja pokretanjem nekog prošlog natjecanja iz kalendara ili stvaranje natjecanja nasumičnim odabirom zadataka iz baze zadataka
  4. Natjecanje se stvara i natjecatelj može krenuti s rješavanjem zadataka

#### UC15 – Unos novog zadatka

- **Glavni sudionik:** Korisnik (voditelj)
- **Cilj:** Stvoriti novi zadatak u bazi postojećih zadataka
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik je prijavljen kao voditelj
- **Opis osnovnog tijeka:**
  1. Voditelj odabire opciju za unos novog zadatka
  2. Otvara se forma gdje voditelj upisuje naziv zadatka, broj bodova koji nosi, vremensko ograničenje izvršavanja, tekst zadatka i primjere za evaluaciju
  3. Voditelj odabire želi li zadatak stvoriti kao privatan ili javan
  4. Voditelj potvrđuje unos zadatka
  5. Zadatak se pohranjuje u bazu ostalih zadataka s informacijom o autoru zadatka
- **Opis mogućih odstupanja:**
  - 2.a Voditelj ostavlja prazno neko polje u formi i pokušava predati takav zadatak
    1. Sustav ga obavještava o neispravnom pokušaju predaje zadatka
    2. Sustav omogućuje ponovno ispunjavanje forme u svrhu ispravne predaje

#### UC16 – Uređivanje zadatka

- **Glavni sudionik:** Korisnik (voditelj)

- **Cilj:** Uređivanje postojećeg zadatka
- **Sudionici:** baza podataka
- **Preduvjet:** postoji zadatak koji je prijavljeni voditelj unio u bazu
- **Opis osnovnog tijeka:**
  1. Voditelj odabire opciju “moji zadaci”
  2. Otvara se pregled svih zadataka koje je voditelj unio u sustav
  3. Voditelj odabire opciju uređivanja zadataka
  4. Otvara se forma slična onoj kod unosa novog zadatka koja voditelju omogućuje izmjenu podataka te spremanje istih

### UC17 – Organiziranje natjecanja

- **Glavni sudionik:** Korisnik (voditelj)
- **Cilj:** Organizirati novo natjecanje
- **Sudionici:** baza podataka
- **Preduvjet:** korisnik je prijavljen
- **Opis osnovnog tijeka:**
  1. Voditelj odabire opciju za organiziranje novog natjecanja
  2. Otvara se forma gdje voditelj odabire vrijeme početka i završetka natjecanja, broj zadataka, koji zadaci će biti aktivni te po želji učitava sličicu pehara
  3. Voditelj potvrđuje podatke o novom natjecanju
  4. Natjecanje se dodaje u kalendar natjecanja
- **Opis mogućih odstupanja:**
  - 2.a Voditelj ne ispunjava neko polje u formi i pokušava potvrditi natjecanje
    1. Sustav ga obavještava o neispravnosti ispunjene forme
    2. Sustav omogućuje ponovno ispunjavanje forme u svrhu ispravne predaje

### UC18 – Uređivanje natjecanja

- **Glavni sudionik:** Korisnik (voditelj)
- **Cilj:** Uređivanje postojećeg natjecanja
- **Sudionici:** baza podataka
- **Preduvjet:** postoji natjecanje koje je prijavljeni voditelj organizirao, a ono nije u tijeku ili završeno
- **Opis osnovnog tijeka:**
  1. Voditelj odabire opciju “moja natjecanja”

2. Otvara se pregled svih natjecanja koje je voditelj organizirao
3. Voditelj odabire opciju uređivanja natjecanja kojemu želi izmijeniti podatke
4. Otvara se pregled sličan onom prilikom organiziranja novog natjecanja koji voditelju omogućuje izmjenu i spremanje novih postavki natjecanja

#### **UC19 – Pregled svih korisnika u bazi**

- **Glavni sudionik:** Administrator
- **Cilj:** Pregled svih registriranih korisnika u bazi
- **Sudionici:** baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju prikaza korisnika u bazi
  2. Otvara se popis svih registriranih BytePit korisnika

#### **UC20 – Izmjena osobnih podataka**

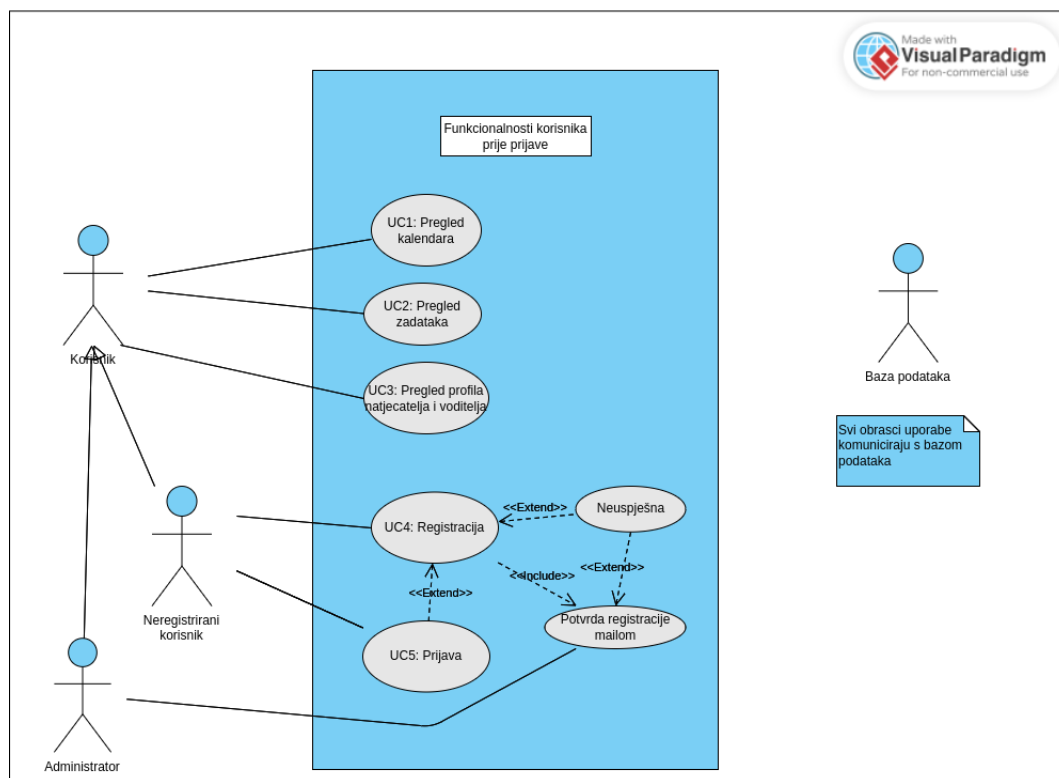
- **Glavni sudionik:** Administrator
- **Cilj:** Uređivanje podataka nekog korisnika
- **Sudionici:** baza podataka
- **Preduvjet:** postoji barem jedan registrirani korisnik u bazi
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju prikaza korisnika u bazi
  2. Otvara se popis svih registriranih BytePit korisnika
  3. Odabirom korisnika prikazuju se njegovi osobni podaci s mogućnošću izmjene istih uključujući i izmjenu dodijeljene uloge korisniku

#### **UC21 – Brisanje korisnika**

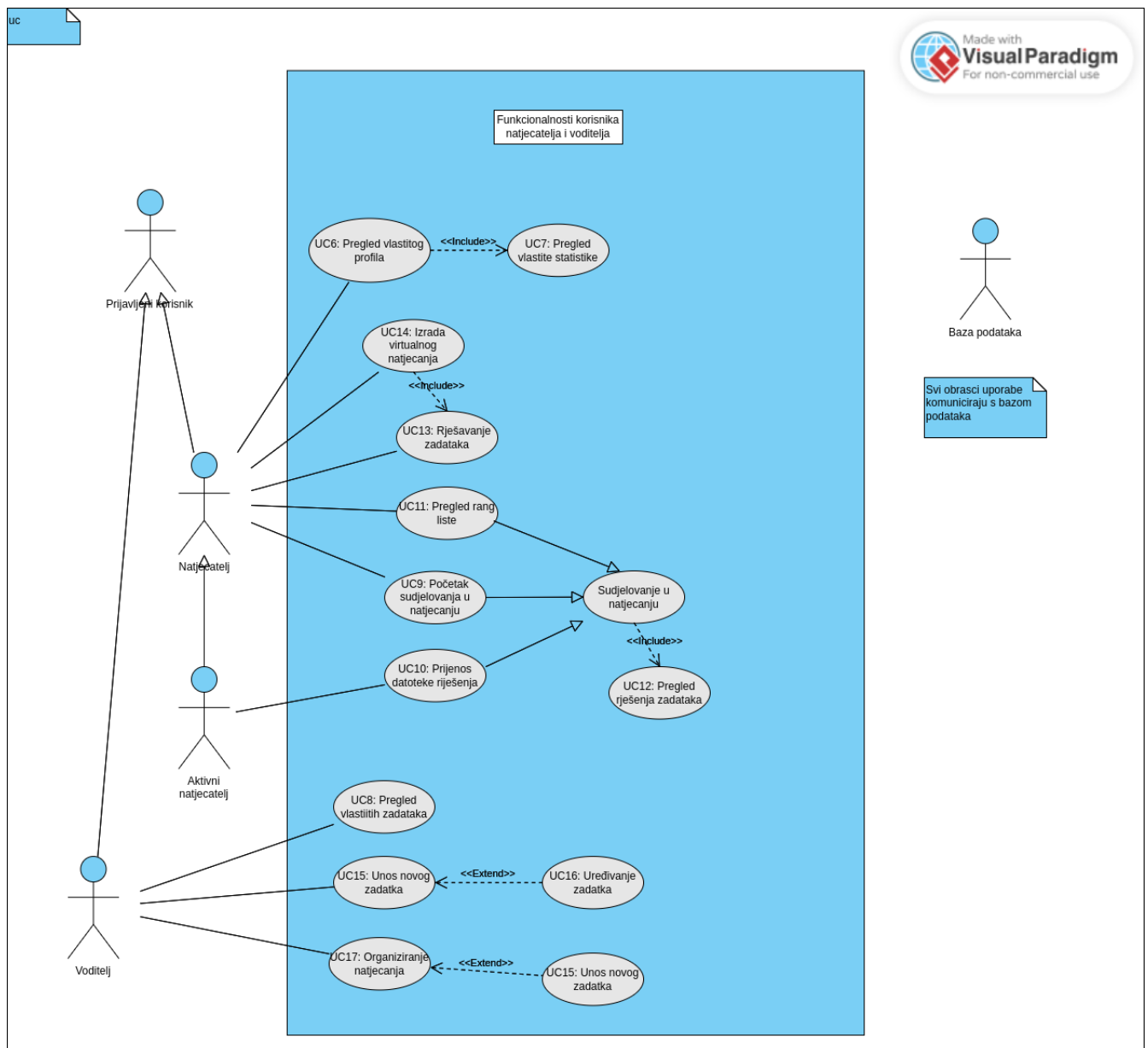
- **Glavni sudionik:** Administrator
- **Cilj:** Brisanje postojećeg korisnika iz baze podataka
- **Sudionici:** baza podataka
- **Preduvjet:** postoji barem jedan registrirani korisnik u bazi
- **Opis osnovnog tijeka:**
  1. Administrator odabire opciju prikaza korisnika u bazi
  2. Otvara se popis svih registriranih BytePit korisnika
  3. Administrator odabire i potvrđuje opciju brisanja korisnika
  4. Korisnik se briše iz baze podataka

## Dijagrami obrazaca uporabe

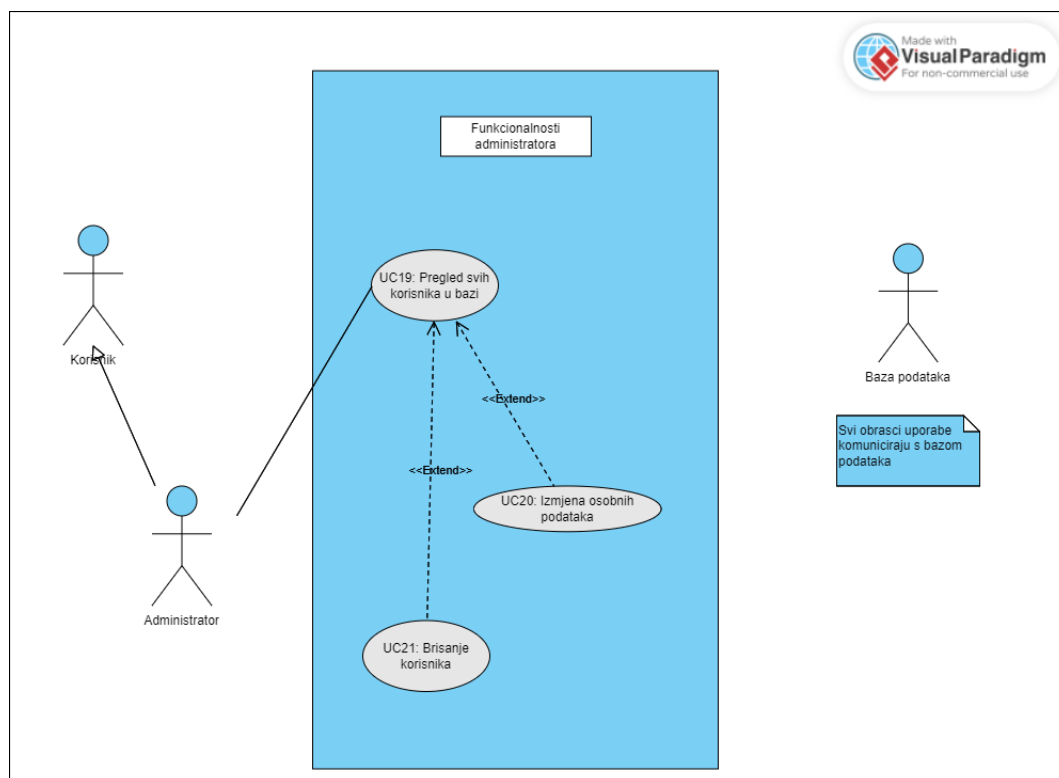
Prikazati odnos aktora i obrazaca uporabe odgovarajućim UML dijagramom. Nije nužno nacrtati sve na jednom dijagramu. Modelirati po razinama apstrakcije i skupovima srodnih funkcionalnosti.



Slika 3.1: Obrasci uporabe - funkcionalnosti za neprijavljene korisnike



Slika 3.2: Obrasci uporabe - funkcionalnosti za natjecatelje i voditelje



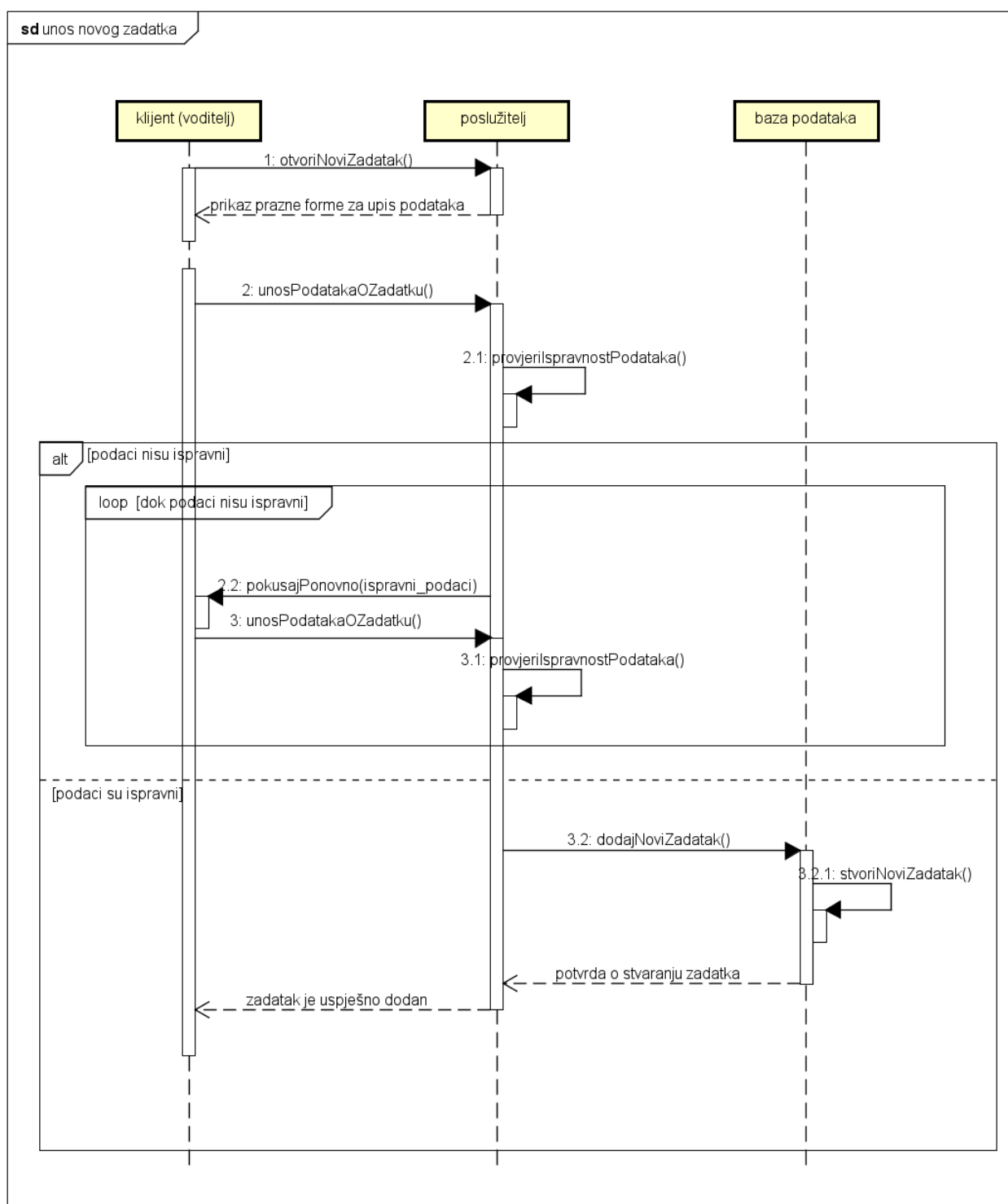
Slika 3.3: Obrasci uporabe - funkcionalnosti za administratore



### 3.1.2 Sekvencijski dijagrami

#### UC15 – unos novog zadatka

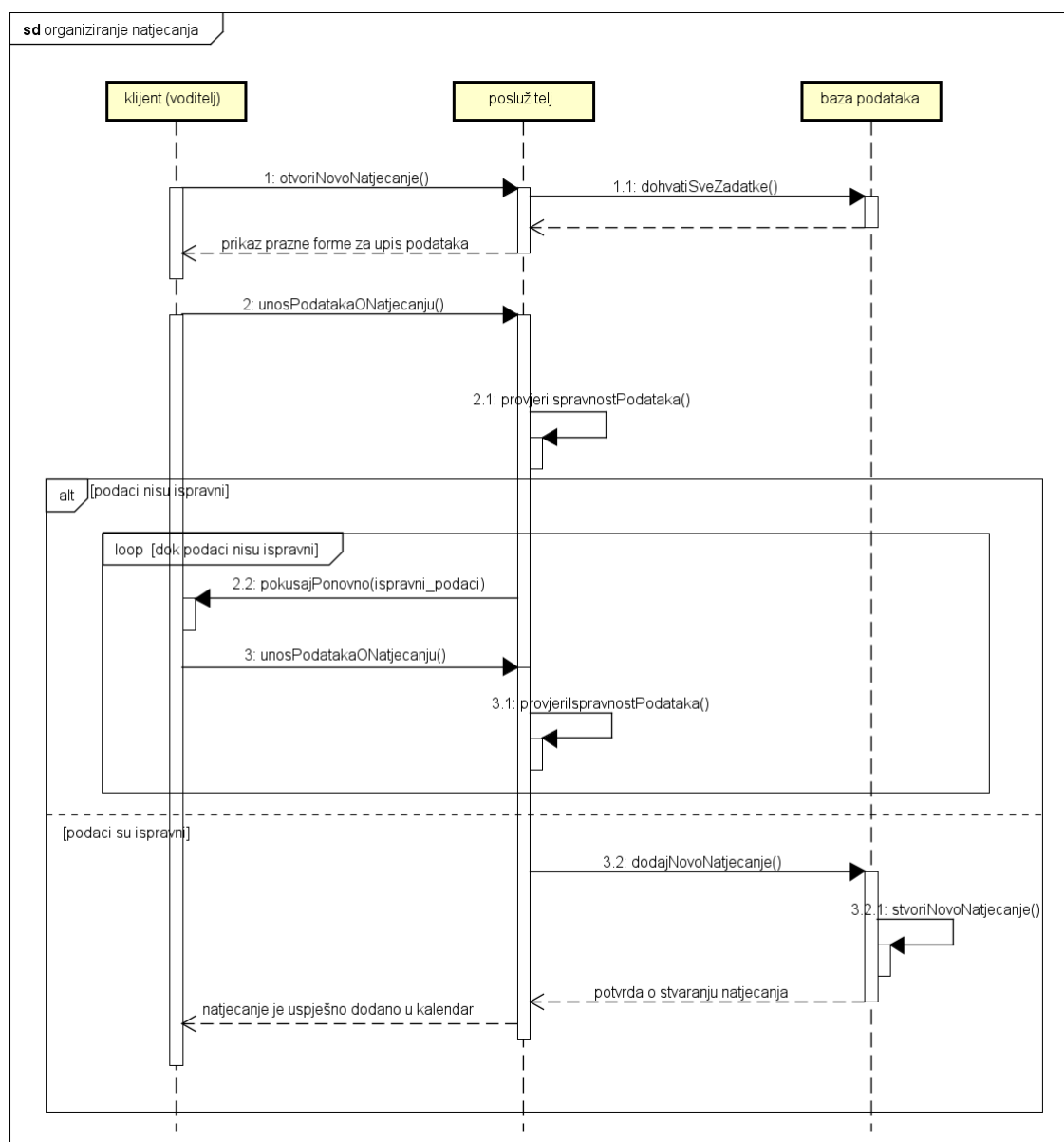
Korisnik prijavljen u sustav kao voditelj odabire opciju za stvaranje novog zadatka. Poslužitelj prikazuje formu s praznim poljima koje voditelj treba ispuniti podacima o zadatku. Točnije, potrebno je navesti naziv zadatka, broj bodova koje nosi, vremensko ograničenje izvršavanja, tekst zadatka i primjere za evaluaciju, a moguće je i odabrati opciju da zadatak bude stvoren kao privatan. Voditelj upisuje navedene podatke te ih šalje poslužitelju koji prvo provjerava da su svi podaci ispravno uneseni te da nema polja koja su ostala prazna. U slučaju neispravnosti podataka, poslužitelj prikazuje relevantnu poruku o problemu i ponovno omogućuje ispunjavanje forme. Nakon uspješne provjere, podaci se šalju bazi podataka koja ih sprema i time stvara novi zadatak. Ako sve prođe bez problema, baza podataka šalje potvrdu poslužitelju o uspješnom stvaranju novog zadatka, a poslužitelj zatim javlja korisniku da je njegov zahtjev uspješno proveden.



Slika 3.4: Sekvencijski dijagram za obrazac uporabe - unos novog zadatka

**UC17 – organiziranje novog natjecanja**

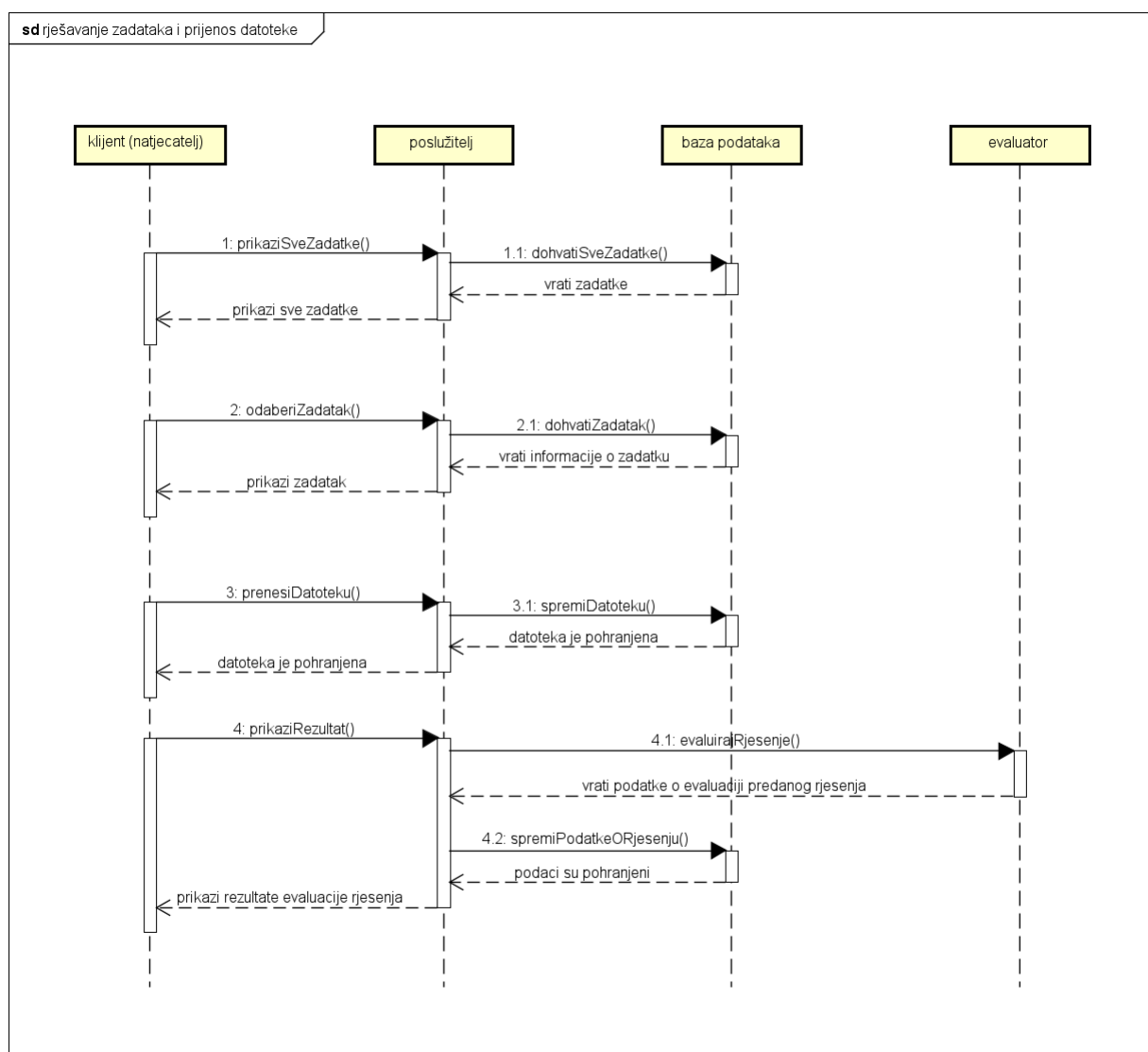
Korisnik prijavljen u sustav kao voditelj odabire opciju za stvaranjem novog natjecanja. Poslužitelj prikazuje formu s praznim poljima koje je voditelj dužan ispuniti informacijama o natjecanju. Potrebno je unijeti vrijeme početka i završetka natjecanja, broj zadataka, odabrati zadatke koji će biti aktivni te po želji učitati sličicu pehara kojom se nagrađuju najbolji natjecatelji. Nakon što voditelj upiše podatke i preda ih poslužitelju na obradu, poslužitelj će prvo provjeriti ispravnost dobivenih vrijednosti. U slučaju da je voditelj izostavio nešto i ostavio polje praznim, sustav će ga obavijestiti i zatražiti ponovni unos podataka. Nakon uspješno ispunjene forme i uspješno primljenih podataka, poslužitelj će ih proslijediti bazi podataka i zatražiti stvaranje novog natjecanja. Nakon pohrane, ako sve prođe bez problema, baza podataka bi trebala poslati povratnu informaciju poslužitelju o uspješnom stvaranju novog natjecanja, a poslužitelj bi zatim trebao javiti korisniku da je njegov zahtjev uspješno proveden.



Slika 3.5: Sekvencijski dijagram za obrazac uporabe - organiziranje novog natjecanja

**UC10 i UC13 – rješavanje zadataka i prijenos datoteke**

Korisnik prijavljen u sustav kao natjecatelj odabire opciju za prikaz svih objavljenih zadataka. Poslužitelj prikazuje popis svih javnih zadataka. Natjecatelj zatim bira zadatak koji želi pokušati riješiti pri čemu se otvara sučelje s tekstom zadatka i opcijom za prijenos datoteke rješenja. Natjecatelj odabire datoteku koju želi prenijeti kao rješenje i potvrđuje svoj odabir, a poslužitelj prosljeđuje datoteku do baze podataka koja ju sprema za prikaz prilikom nekih drugih funkcionalnosti. Na klijentov zahtjev predana datoteka se predaje evaluatoru koji pomoću definiranih primjera ulaza i izlaza određuje točnost rješenja zadatka i vraća ih poslužitelju. Poslužitelj sve rezultate sprema u bazu podataka i prikazuje klijentu u aplikaciji.

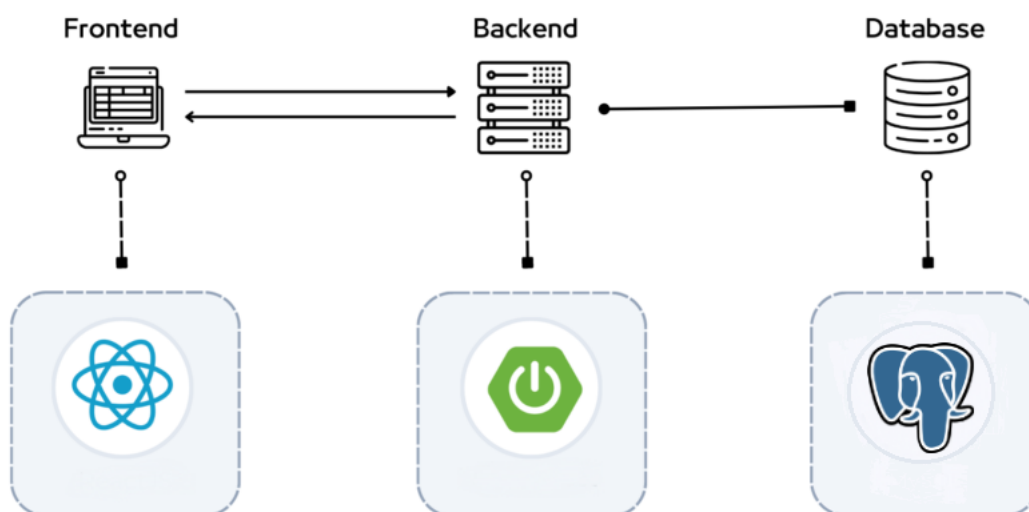


Slika 3.6: Sekvencijski dijagram za obrasce uporabe - rješavanje zadataka i prijenos datoteke

## 3.2 Ostali zahtjevi

- Sustav treba dopustiti višekorisnički rad u stvarnom vremenu.
- Sustav treba biti jednostavan i intuitivan za korištenje tako da djeca nemaju problema sa razumijevanjem i snalaženjem na aplikaciji.
- Sustav se treba izgraditi kao mrežna aplikacija pomoću objektno-orijentiranih jezika.
- Sustav treba biti prilagođen za hrvatski jezik i abecedu prilikom prikaza i unosa tekstualnog sadržaja, uključujući i dijakritičke znakove.
- Pristup sustavu treba biti omogućen putem javne mreže.
- Sustav treba omogućiti evaluaciju rješenja zadataka za minimalno jedan programski jezik.
- Sustav treba biti takav da se sve funkcije izvršavaju brzo, ne duže od nekoliko sekundi, uključujući i evaluaciju rješenja zadataka.
- Baza podataka sustava mora biti kvalitetno i ispravno povezana sa sučeljem aplikacije.

## 4. Arhitektura i dizajn sustava



Slika 4.1: Prikaz arhitekture sustava

Arhitektura sustava može se podijeliti na tri glavna podsustava, a to su **Web poslužitelj**, **Web aplikacija** i **baza podataka**.

- **Web poslužitelj** prima zahtjeve od klijenata putem interneta, obrađuje ih i pruža resurse poput web stranice, slike, videa i datoteke kao odgovor. Za distribuciju resursa najčešće se koriste protokoli kao što su HTTP (Hypertext Transfer Protocol) ili HTTPS (HTTP Secure).
- **Web aplikacija** je program koji se izvršava na web pregledniku (Google Chrome, Mozilla Firefox, Safari itd.) i pruža korisnicima mogućnost izvršavanja željenih zahtjeva, odnosno interakciju s određenim uslugama i funkcionalnostima web aplikacije. Prilikom obrade zahtjeva pristupa se bazi podataka i korisniku se odgovor vraća kao HTML dokument.
- **Baza podataka** je organizirani skup podataka namijenjen za efikasno upravljanje, ažuriranje, pretraživanje i dohvat podataka. Uloge baze podataka u web aplikaciji su pohrana podataka, očuvanje integriteta podataka i osiguravanje dosljednosti, upravljanje transakcijama itd.



Za izradu naše web aplikacije odabrali smo **Spring Boot** (open-source Java framework) i **React** (open-source JavaScript library). Odabrana razvojna okruženja su IntelliJ IDEA i Eclipse IDE za Spring Boot, odnosno Visual Studio Code i WebStorm za React. Za izradu baze podataka koristimo PostgreSQL. Arhitektura, koja je podržana Spring Boot-om, temelji se na **MVC (Model-View-Controller)** konceptu koji strogo odvađa model, akcije i prezentaciju, olakšava razvoj i održavanje aplikacije te čini aplikaciju prilagodljivom i jednostavnom za proširenje.

- **Model** - predstavlja poslovnu logiku, odnosno dinamičke strukture podataka, mijenja pogled na zahtjev kontrolera. Modeli u pravilu predstavljaju podatke(objekte) koje aplikacija obrađuje.
- **View** - ono što klijent vidi, odnosno korisničko sučelje potrebno za interakciju s aplikacijom kao što su dijagrami, linkovi, slike, tablice itd.
- **Controller** - presreće zahtjeve klijenata i prilagođava model, odnosno obavještava model o promjeni zahtjeva korisnika i u skladu sa zahtjevima daje prikladan View.

## 4.1 Baza podataka

### *dio 1. revizije*

*Potrebno je opisati koju vrstu i implementaciju baze podataka ste odabrali, glavne komponente od kojih se sastoji i slično.*

#### 4.1.1 Opis tablica

*Svaku tablicu je potrebno opisati po zadanom predlošku. Lijevo se nalazi točno ime varijable u bazi podataka, u sredini se nalazi tip podataka, a desno se nalazi opis varijable. Svjetlozelenom bojom označite primarni ključ. Svjetlo plavom označite strani ključ*

korisnik - ime tablice		
IDKorisnik	INT	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

korisnik - ime tablice		
korisnickoIme	VARCHAR	
email	VARCHAR	
ime	VARCHAR	
primjer	VARCHAR	

#### 4.1.2 Dijagram baze podataka

*U ovom potpoglavlju potrebno je umetnuti dijagram baze podataka. Primarni i strani ključevi moraju biti označeni, a tablice povezane. Bazu podataka je potrebno normalizirati. Podsjetite se kolegija "Baze podataka".*

## 4.2 Dijagram razreda

*Potrebno je priložiti dijagram razreda s pripadajućim opisom. Zbog preglednosti je moguće dijagram razlomiti na više njih, ali moraju biti grupirani prema sličnim razinama apstrakcije i srodnim funkcionalnostima.*

### **dio 1. revizije**

*Prilikom prve predaje projekta, potrebno je priložiti potpuno razrađen dijagram razreda vezan uz **generičku funkcionalnost** sustava. Ostale funkcionalnosti trebaju biti idejno razrađene u dijagramu sa sljedećim komponentama: nazivi razreda, nazivi metoda i vrste pristupa metodama (npr. javni, zaštićeni), nazivi atributa razreda, veze i odnosi između razreda.*

### **dio 2. revizije**

*Prilikom druge predaje projekta dijagram razreda i opisi moraju odgovarati stvarnom stanju implementacije*

## 4.3 Dijagram stanja

### *dio 2. revizije*

*Potrebno je priložiti dijagram stanja i opisati ga. Dovoljan je jedan dijagram stanja koji prikazuje **značajan dio funkcionalnosti** sustava. Na primjer, stanja korisničkog sučelja i tijekom korištenja neke ključne funkcionalnosti jesu značajan dio sustava, a registracija i prijava nisu.*

## 4.4 Dijagram aktivnosti

### *dio 2. revizije*

*Potrebno je priložiti dijagram aktivnosti s pripadajućim opisom. Dijagram aktivnosti treba prikazivati značajan dio sustava.*

## 4.5 Dijagram komponenti

### *dio 2. revizije*

*Potrebno je priložiti dijagram komponenti s pripadajućim opisom. Dijagram komponenti treba prikazivati strukturu cijele aplikacije.*

## 5. Implementacija i korisničko sučelje

### 5.1 Korištene tehnologije i alati

#### *dio 2. revizije*

*Detaljno navesti sve tehnologije i alate koji su primijenjeni pri izradi dokumentacije i aplikacije. Ukratko ih opisati, te navesti njihovo značenje i mjesto primjene. Za svaki navedeni alat i tehnologiju je potrebno **navesti internet poveznicu** gdje se mogu preuzeti ili više saznati o njima.*

## 5.2 Ispitivanje programskog rješenja

### *dio 2. revizije*

*U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti na razini komponenti i na razini cijelog sustava s prikazom odabranih ispitnih slučajeva. Studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete.*

### 5.2.1 Ispitivanje komponenti

*Potrebno je provesti ispitivanje jedinica (engl. unit testing) nad razredima koji implementiraju temeljne funkcionalnosti. Razraditi **minimalno 6 ispitnih slučajeva** u kojima će se ispitati redovni slučajevi, rubni uvjeti te izazivanje pogreške (engl. exception throwing). Poželjno je stvoriti i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane. Potrebno je priložiti izvorni kôd svih ispitnih slučajeva te prikaz rezultata izvođenja ispita u razvojnom okruženju (prolaz/pad ispita).*

### 5.2.2 Ispitivanje sustava

*Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium<sup>1</sup>. Razraditi **minimalno 4 ispitna slučaja** u kojima će se ispitati redovni slučajevi, rubni uvjeti te poziv funkcionalnosti koja nije implementirana/izaziva pogrešku kako bi se vidjelo na koji način sustav reagira kada nešto nije u potpunosti ostvareno. Ispitni slučaj se treba sastojati od ulaza (npr. korisničko ime i lozinka), očekivanog izlaza ili rezultata, koraka ispitivanja i dobivenog izlaza ili rezultata.*

*Izradu ispitnih slučajeva pomoću radnog okvira Selenium moguće je provesti pomoću jednog od sljedeća dva alata:*

- *dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita*
- ***Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsko sučelje.*

*Detalji o korištenju alata Selenium bit će prikazani na posebnom predavanju tijekom semestra.*

---

<sup>1</sup><https://www.seleniumhq.org/>



## 5.3 Dijagram razmještaja

### *dio 2. revizije*

Potrebno je umetnuti **specifikacijski** dijagram razmještaja i opisati ga. Moguće je umjesto specifikacijskog dijagrama razmještaja umetnuti dijagram razmještaja instanci, pod uvjetom da taj dijagram bolje opisuje neki važniji dio sustava.

## 5.4 Upute za puštanje u pogon

### *dio 2. revizije*

*U ovom poglavlju potrebno je dati upute za puštanje u pogon (engl. deployment) ostvarene aplikacije. Na primjer, za web aplikacije, opisati postupak kojim se od izvornog kôda dolazi do potpuno postavljene baze podataka i poslužitelja koji odgovara na upite korisnika. Za mobilnu aplikaciju, postupak kojim se aplikacija izgradi, te postavi na neku od trgovina. Za stolnu (engl. desktop) aplikaciju, postupak kojim se aplikacija instalira na računalo. Ukoliko mobilne i stolne aplikacije komuniciraju s poslužiteljem i/ili bazom podataka, opisati i postupak njihovog postavljanja. Pri izradi uputa preporučuje se **naglasiti korake instalacije uporabom natuknica** te koristiti što je više moguće **slike ekrana** (engl. screenshots) kako bi upute bile jasne i jednostavne za slijediti.*

*Dovršenu aplikaciju potrebno je pokrenuti na javno dostupnom poslužitelju. Studentima se preporuča korištenje neke od sljedećih besplatnih usluga: Amazon AWS, Microsoft Azure ili Heroku. Mobilne aplikacije trebaju biti objavljene na F-Droid, Google Play ili Amazon App trgovini.*

## 6. Zaključak i budući rad

### *dio 2. revizije*

*U ovom poglavlju potrebno je napisati osvrt na vrijeme izrade projektnog zadatka, koji su tehnički izazovi prepoznati, jesu li riješeni ili kako bi mogli biti riješeni, koja su znanja stečena pri izradi projekta, koja bi znanja bila posebno potrebna za brže i kvalitetnije ostvarenje projekta i koje bi bile perspektive za nastavak rada u projektnoj grupi.*

*Potrebno je točno popisati funkcionalnosti koje nisu implementirane u ostvarenoj aplikaciji.*

# Popis literature

## *Kontinuirano osvježavanje*

*Popisati sve reference i literaturu koja je pomogla pri ostvarivanju projekta.*

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, Software engineering book", Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>

# Indeks slika i dijagrama

2.1	Sustav SPOJ - pregled zadataka na stranici . . . . .	8
2.2	Sustav Edgar - pregled statistike ispita . . . . .	8
2.3	Sustav Codeforces - pregled statusa na natjecanju . . . . .	9
2.4	Sustav Codeforces - kalendar nadolazećih natjecanja . . . . .	9
2.5	Sustav SPOJ - profil natjecatelja . . . . .	10
3.1	Obrasci uporabe - funkcionalnosti za neprijavljene korisnike . . . .	21
3.2	Obrasci uporabe - funkcionalnosti za natjecatelje i voditelje . . . .	22
3.3	Obrasci uporabe - funkcionalnosti za administratore . . . . .	23
3.4	Sekvencijski dijagram za obrazac uporabe - unos novog zadatka . .	25
3.5	Sekvencijski dijagram za obrazac uporabe - organiziranje novog natjecanja . . . . .	27
3.6	Sekvencijski dijagram za obrasce uporabe - rješavanje zadataka i prijenos datoteke . . . . .	29
4.1	Prikaz arhitekture sustava . . . . .	31

# Dodatak: Prikaz aktivnosti grupe

## Dnevnik sastajanja

### *Kontinuirano osvježavanje*

*U ovom dijelu potrebno je redovito osvježavati dnevnik sastajanja prema predlošku.*

#### 1. sastanak

- Datum: 18. listopada 2023.
- Prisustvovali: Vedran Ćutić, Antonio Glavaš, Marina Hrbud, Lara Marčec, Jakov Novak, Marko Varga, Nikola Vlahović
- Teme sastanka:
  - upoznavanje
  - analiza zadatka
  - postavljanje GitHub-a

#### 2. sastanak

- Datum: 20. listopada 2023.
- Prisustvovali: Vedran Ćutić, Antonio Glavaš, Marina Hrbud, Lara Marčec, Jakov Novak, Marko Varga, Nikola Vlahović
- Teme sastanka:
  - uvodni sastanak s asistentom i demonstratoricom
  - razrješavanje upita o osnovnim funkcionalnostima aplikacije
  - dogovor oko alata i tehnologija

#### 3. sastanak

- Datum: 31. listopada 2023.
- Prisustvovali: Vedran Ćutić, Antonio Glavaš, Marina Hrbud, Lara Marčec, Jakov Novak, Marko Varga, Nikola Vlahović
- Teme sastanka:
  - pregled odrađenih zadataka
  - dogovor o vizualnom izgledu aplikacije
  - raspodjela daljnjih poslova i dogovor internih rokova za iste

## Tablica aktivnosti

### Kontinuirano osvježavanje

*Napomena: Doprinosi u aktivnostima treba navesti u satima po članovima grupe po aktivnosti.*

	Vedran Ćutić	Antonio Glavaš	Marina Hrbud	Lara Marčec	Jakov Novak	Marko Varga	Nikola Vlahović
Upravljanje projektom							
Opis projektnog zadatka			2			2	
Funkcionalni zahtjevi				1			
Opis pojedinih obrazaca				2			1
Dijagram obrazaca					2	1	
Sekvencijski dijagrami				3			
Opis ostalih zahtjeva				1			
Arhitektura i dizajn sustava		2					
Baza podataka							
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							
Upute za puštanje u pogon							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Vedran Čutić	Antonio Glavaš	Marina Hrbud	Lara Marčec	Jakov Novak	Marko Varga	Nikola Vlahović
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>							
<i>npr. izrada početne stranice</i>							
<i>izrada baze podataka</i>							
<i>spajanje s bazom podataka</i>							
<i>back end</i>							



## Dijagrami pregleda promjena

### *dio 2. revizije*

*Prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je na kraju projekta generirane grafove s gitlaba prenijeti u ovo poglavlje dokumentacije. Dijagrami za vlastiti projekt se mogu preuzeti s [gitlab.com](https://gitlab.com) stranice, u izborniku Repository, pritiskom na stavku Contributors.*