

Programsko inženjerstvo

Ak. god. 2020./2021.

Što želiš čitati

Dokumentacija, Rev. 1

Grupa: *Proppers*

Voditelj: *Sven Winkler*

Datum predaje: 17. 11. 2023.

Nastavnik: *Alan Jović*

Sadržaj

1	Dnevnik promjena dokumentacije	3
2	Opis projektnog zadatka	5
2.1	Uvod	5
2.2	Namijena projekta	5
2.3	Funkcionalnost web aplikacije	5
2.4	Dodatne funkcionalnosti	7
2.5	Primjer sličnog rješenja	7
2.5.1	Ekran za prijavu	8
2.5.2	Glavni prozor	9
2.5.3	Lista knjiga	9
2.5.4	Prikaz knjige	10
3	Specifikacija programske potpore	12
3.1	Funkcionalni zahtjevi	12
3.1.1	Obrasci uporabe	14
3.1.2	Sekvencijski dijagrami	20
3.2	Ostali zahtjevi	25
4	Arhitektura i dizajn sustava	26
4.1	Baza podataka	28
4.1.1	Opis tablica	28
4.1.2	Dijagram baze podataka	31
4.2	Dijagram razreda	32
4.3	Dijagram stanja	34
4.4	Dijagram aktivnosti	35
4.5	Dijagram komponenti	37
5	Implementacija i korisničko sučelje	38
5.1	Korištene tehnologije i alati	38
5.2	Ispitivanje programskog rješenja	40

5.2.1 Ispitivanje komponenti	40
5.2.2 Ispitivanje sustava	40
5.3 Dijagram razmještaja	41
5.4 Upute za puštanje u pogon	42
6 Zaključak i budući rad	43
Popis literature	44
Indeks slika i dijagrama	45
Dodatak: Prikaz aktivnosti grupe	46

1. Dnevnik promjena dokumentacije

Kontinuirano osvježavanje

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak. Započeto pisanje opisa projektnog zadatka.	Božidar Pučar	29.10.2023.
0.2	Specifikacija programske potpore bez dijagrama.	Andrej Lovei	29.10.2023.
0.3	Dodan UML dijagram.	Andrej Lovei	29.10.2023.
0.4	Dovršetak opisa projektnog zadatka i generiran pdf.	Božidar Pučar	2.11.2023.
0.5	Dodane slike za opis.	Božidar Pučar	2.11.2023.
0.6	Dodan ER dijagram.	Luka Kitarović	3.11.2023.
0.7	Dodan ER dijagram i relacijski dijagram.	Yu Xing Jin	5.11.2023.
0.8	Dodani sekvencijski dijagrami.	Sven Win- kler	5.11.2023.
0.9	Dodano potpoglavlje Arhitektura i dizajn sustava u dokumentaciju i slike.	Damjan Šarlija	6.11.2023.
0.10	Dodan opis baze podataka i dijagram baze podataka.	Yu Xing Jin	8.11.2023.

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori	Datum
0.11	Maknut italic iz opisa projektnog zadatka.	Božidar Pučar	16.11.2023.
0.12	Dodani opisi sekvencijskih dijagrama te uređena struktura	Andrej Lovei	16.11.2023.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	*	11.09.2013.
1.1	Dodan dijagram stanja	Yu Xing Jin	16.1.2024.
1.2	Dodan dijagram aktivnosti	Yu Xing Jin	17.1.2024.
1.3	Dodan dijagram komponenti	Yu Xing Jin	17.1.2024.
1.4	Dodani opisi dijagrama	Yu Xing Jin	17.1.2024.
1.5	Dodano poglavlje o korištenim tehnologijama i alatima	Andrej Lovei	19.1.2024.
1.6	Dodan dijagram razmještaja s opisom	Vedran Marković	19.1.2024.
1.7	Dodan zaključak	Vedran Marković	21.1.2024.
1.8	Ažuriran prikaz aktivnosti grupe	Damjan Šaralića	21.1.2024.

2. Opis projektnog zadatka

2.1 Uvod

Cilj ovog projektnog zadatka je razvoj programske podrške za stvaranje web aplikacije "Što želiš čitati" koja će korisnicima omogućiti pretragu i pronalaženje knjiga koje se prodaju u određenim trgovinama. Web aplikacija sadržavat će dva tipa korisnika od kojih će normalnim korisnicima biti omogućeno pojednostavljeno pretraživanje knjiga i trgovina koje prodaju određene knjige, dok će registriranim korisnicima biti omogućeno prodavanje i prevođenje knjiga po narudžbi.

2.2 Namijena projekta

Glava zadaća ovog projekta je olakšavanje pronalaženja knjiga koje može biti dugotrajan i mukotrpan proces. Mnoge knjižare i trgovine slične namjene u Republici Hrvatskoj još se nisu prilagodile digitalnom dobu te se oslanjaju na papirnatu oglase i reklamaciju preko televizije ili radija. Ova web aplikacija uvelike bi olakšala i unaprijedila način poslovanja tih trgovina koje bi, kroz par klikova mišem, svoje knjige mogle objaviti na jednom mjestu i pojednostaviti kupcima pronalaženje željenih proizvoda. Osim toga ova web aplikacija također bi služila kao platforma za komunikaciju i suradnju raznih trgovina koje si međusobno mogu slati zahtjeve za prevođenje knjiga.

2.3 Funkcionalnost web aplikacije

Web aplikacija primarno je namijenjena za ljubitelje knjiga i prodavače koji će se dijeliti na dvije vrste korisnika: neregistrirani i registrirani korisnik.

Neregistrirani korisnici aplikaciju će koristiti za pregled ponuda knjiga po trgovinama te njihovo pretraživanje. Ovim korisnicima bit će ponuđeno pretraživanje knjiga pomoću tražilice, odabirom određene trgovine na listi ili odabirom trgovine pomoću karte na kojoj će biti prikazane lokacije svih ponuditelja knjiga koji zadovoljavaju neki kriterij. Neregistrirani korisnici također će moći od izdavača tražiti

da kontaktira stranog izdavača oko prijevoda strane knjige na hrvatski jezik.

Registrirani korisnici moći će biti samo ponuditelji knjiga te će birati između tri ponuđene vrste ponuditelja (izdavač, antikvarijat i preprodavač). Pri završetku registracije korisnika šalje se zahtjev za odobrenje registracije glavnom administratoru sustava koji treba odobriti zahtjev. Ponuditelji imaju mogućnost ponuditi neograničen broj naslova knjiga i primjeraka knjiga kroz web aplikaciju, te ovisno o vrsti mogu zatražiti da se knjiga na stranom jeziku prevede na hrvatski jezik. Ti zahtjevi ovise o vrsti preprodavača i glase:

- izdavač na temelju prikupljenih želja neregistriranih korisnika smije zatražiti izdavača strane knjige za dozvolu prijevoda knjige sa stranog ili srodnog jezika na hrvatski jezik
- izdavač kroz aplikaciju smije nuditi samo knjige na hrvatskom jeziku
- antikvarijat smije nuditi knjige na stranom jeziku, srodnom jeziku ili hrvatskom jeziku, a smije se adresom nalaziti samo na području Hrvatske
- preprodavač smije nuditi sve vrste knjiga, uključujući i one koje nisu na drugačiji način dobavljive na području Hrvatske, pri čemu mu adresa može biti u Hrvatskoj ali i u zemljama sa srodnim jezikom

Glavni objekti baze web aplikacije bit će knjige koje će imati sljedeće značajke:

- naziv
- autori
- godina izdanja
- izdavač
- kategorija izdavača (domaći, stari)
- žanr
- ISBN
- broj izdanja
- stanje očuvanosti
- tekstni opis
- slika korica
- oznaka vrste knjige
- lista ponuda

Oznaka vrste knjige može biti sljedeća:

- Knjiga je na stranom jeziku (npr. engleski), a ne postoji izdanje na hrvatskom ili srodnom jeziku (bosanski, srpski)
- Knjiga je izdana na hrvatskom jeziku i dobavljiva je na području Hrvatske
- Knjiga je izdana na hrvatskom jeziku, ali nije dobavljiva na području Hrvatske (npr. izdavač je rasprodao izdanje)
- Knjiga je izdana na srodnom jeziku, dobavljiva je samo na njihovom tržištu
- Knjiga je izdana na srodnom jeziku, dobavljiva je u Hrvatskoj, ne postoji na hrvatskom jeziku

Lista ponuda svake knjige treba sadržavati:

- naziv ponuditelja
- broj dostupnih primjeraka
- cijenu knjige kod dotičnog ponuditelja

Ako knjiga nema više dostupnih registriranih ponuditelja, knjigu se treba ukloniti iz baze podataka i ona se ne smije prikazati tijekom pretrage u sustavu.

Uz glavne funkcionalnosti web aplikacije također se očekuje da će registrirani korisnik kao ponuditelj knjiga moći naknadno dodavati bilo koju knjigu koju želi u aplikaciju, a drugim kanalima može nuditi neke druge knjige. Komunikacija između neregistriranih i registriranih korisnika (ponuditelja) neće se odvijati kroz aplikaciju, već će za to služiti drugi kanali komunikacije (npr. e-pošta, telefon, Skype).

2.4 Dodatne funkcionalnosti

Aplikacija će trebati biti izvedena kao web aplikacija prilagođena mobilnom uređaju ili tabletu te će podržavati sustav za registraciju i prijavu korisnika uz pomoć korisničkog imena i lozinke. Sustav također mora moći podržavati rad više korisnika u stvarnom vremenu.

2.5 Primjer sličnog rješenja

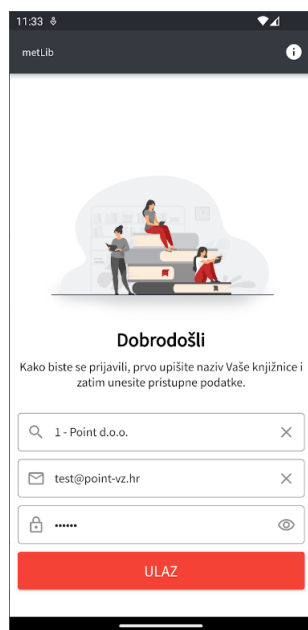
Ova Web aplikacija je slična postojećim web stranicama koje služe za online prodaju knjiga. Dok te stranice primarno funkcioniraju kao web shopovi namijenjeni prodaji bez mogućnosti utjecaja kupaca na izbor knjiga, ova bi aplikacija

omogućila korisnicima pretragu knjiga, lokacija trgovina i mogućnost da zatraže od izdavača prevedenu verziju knjige.

Jedna od takvih web aplikacija je metLib, proizvođača Point d.o.o. koja korisnicima knjižnice omogućuje pretraživanje kataloga knjižnice, te individualno prikaz knjiga koje se trenutno čitaju, arhivu posudbe, listu želja, rezervacije, preporuke... Sličnost koju ova web aplikacija ima s rješenjem našeg projektnog zadatka je što primarno služi kao katalog knjiga za određenu knjižnicu. Naša web aplikacija imala bi sličnu funkcionalnost ali bi bila namijenjena prodavačima knjiga čije bi ponude zajedno prikazivala na jednom mjestu.

2.5.1 Ekran za prijavu

Kad korisnik pokrene aplikaciju on je predložen izborom koji određuje njegovu ulogu u daljem korištenju web aplikacije: hoće li se registrirati ili će je koristiti kao anonimni korisnik. U slučaju da se korisnik odluči prijaviti prvi prozor koji će mu biti od velike važnosti bit će prozor za prijavu u sustav. Primjer jednog takvog prozora može se naći i u web aplikaciji metLib.



Slika 2.1: Primjer prijave u aplikaciju

Web aplikacija ovog projekta imat će prozor za prijavu napravljenu po sličnom principu. Dok prikazana prijava uzima ime knjižnice, adresu e-pošte i lozinku kao što je prikazano na slici 2.1, prijava projektne aplikacije uzimat će korisničko ime i lozinku te će se pomoću tih podataka prijavljivati prodavači knjiga.

2.5.2 Glavni prozor

Web aplikacija srodna projektnoj web aplikaciji ima više funkcionalnosti na glavnom prozoru nego što je predviđeno za projektnu web aplikaciju.

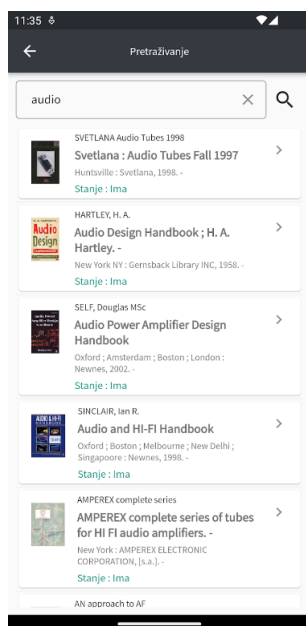


Slika 2.2: Primjer glavnog prozora

Dok prikazani prozor na slici 2.2 uz pretraživanje knjiga ima gumbе koji vode na prozore za prikaz iskaznice, profil korisnika, web katalog, novosti, e-knjige, zbirke i kalendar, projektna web aplikacija primarno će imati prozor za prikaz liste knjiga i tražilicu pomoću koje će neregistrirani korisnici pretraživati bazu podataka unesenih knjiga. Dodatna funkcionalnost bit će mapa pomoću koje će korisnik birati različite prodavaonice ponuditelja nakon čega će mu se prikazati lista knjiga koje nudi ta prodavaonica. Registrirani korisnik umjesto ovog prozora imat će prozor s funkcionalnosti za unos i brisanje knjiga iz baze podataka.

2.5.3 Lista knjiga

Nakon što korisnik klikne na gumb za pretraživanje otvara se novi prozor s tražilicom i listom knjiga. Pretraga knjiga bit će omogućena po značajkama knjige i nazivu ponuditelja. Prozor na slici 2.3 koji prikazuje listu knjiga u web aplikaciji metLib idealni je prikaz onoga što projektna web aplikacija nastoji proizvesti.

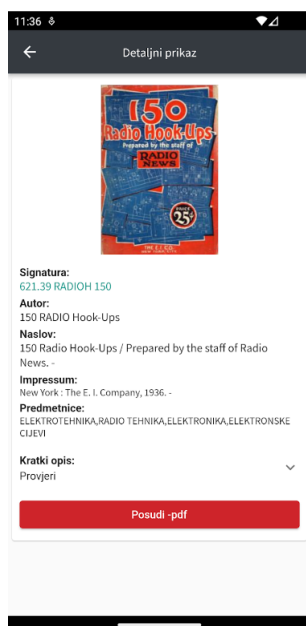


Slika 2.3: Primjer liste knjiga

Neregistrirani korisnici na ovom će prozoru moći pretraživati knjige po značajkama i nazivu ponuditelja, dok će registrirani korisnici imati popis knjiga koje oni prodaju.

2.5.4 Prikaz knjige

Nakon odabira knjige u aplikaciji metLib otvara se ekran sa slike 2.4 koji prikazuje korice knjige, signaturu, autora, naslov, impresum, predmetnice i kratki opis.



Slika 2.4: Primjer pregleda odabrane knjige

Projektna web aplikacija također će sadržavati sličan feature gdje će neregistrirani korisnik odabirom knjige iz liste knjiga biti prebačen na prozor koji će sadržavati osnovne podatke o odabranoj knjizi: naziv, autore, godinu izdanja, izdavača, kategoriju izdavača, žanr, ISBN, broj izdanja, stanje očuvanosti, tekstni opis, sliku korica, oznaku vrste knjige i listu ponuda.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Kupci
2. Ponuditelji
 - (a) Izdavač
 - (b) Antikvarijat
 - (c) Preprodavač
3. Administrator
4. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani korisnik (inicijator) može:
 - (a) pretraživati ponude knjiga
 - i. po značajkama knjige (naziv, autori, godina izdanja, izdavač, kategorija izdavača (domaći, strani), žanr, ISBN, broj izdanja, stanje očuvanosti, tekstni opis, slika korica, oznaka vrste knjige i lista ponuda)
 - ii. po nazivu ponuditelja (izlistati sve knjige dotičnog)
 - iii. na karti (npr. OpenStreetMap) gdje su označene lokacije svih ponuditelja knjiga
 - (b) zatražiti od izdavača da kontaktira stranog izdavača oko prijevoda strane knjige na hrvatski jezik
 - (c) vidjeti popis dostupnih ponuda za odabranu knjigu, uključujući naziv ponuditelja, broj dostupnih primjeraka i cijenu knjige kod svakog ponuditelja

2. Svaki ponuditelj (izdavač, antikvarijat, preprodavač) (inicijator) može:
 - (a) ponuditi neograničeni broj naslova knjiga i primjeraka knjiga
 - (b) uklanjati primjerke već postojeće knjige
3. Izdavač (inicijator) može:

zatražiti izdavača strane knjige za dozvolu prijevoda knjige sa stranog ili srodnog jezika na hrvatski jezik
4. Antikvarijat (inicijator) može:

nuditi knjige na stranom jeziku, srodnom jeziku ili hrvatskom jeziku
5. Preprodavač (inicijator) može:

nuditi sve vrste knjiga, uključujući i one koje nisu na drugačiji način dobavljive na području Hrvatske
6. Administrator (inicijator) može:
 - (a) odobriti registracije ponuditelja
 - (b) upravljati korisničkim računima
 - (c) riješavati sporove i pratiti komunikaciju između neregistriranih korisnika i ponuditelja
7. Baza podataka (sudionik):
 - (a) pohranjuje sve podatke o korisnicima i njihovim ovlastima
 - (b) pohranjuje sve podatke o knjigama kao i zahtjeve za prijevod za svaku knjigu na stranom jeziku

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 - Pregled knjiga na karti

1. **Glavni sudionik:** Neregistrirani korisnik
2. **Cilj:** Pregledati dostupne knjige uz pomoć karte
3. **Sudionici:** Baza podataka
4. **Preduvjet:** -
5. **Opis osnovnog tijeka:**
 - (a) Karta je prikazana prilikom učitavanja aplikacije
 - (b) Neregistrirani korisnik odabire ponuditelja na karti
 - (c) Izlistavaju se sve knjige dotičnog ponuditelja

UC2 - Registracija

1. **Glavni sudionik:** Neregistrirani korisnik
2. **Cilj:** Stvoriti korisnički račun za pristup sustavu
3. **Sudionici:** Baza podataka
4. **Preduvjet:** -
5. **Opis osnovnog tijeka:**
 - (a) Korisnik odabire opciju za registraciju
 - (b) Korisnik unosi potrebne korisničke podatke i bira između 3 navedene vrste ponuditelja
 - (c) Korisnik prima obavijest o uspješnoj registraciji nakon administrator-skog pregleda
6. **Opis mogućih odstupanja:**
 - 2.a Odabir već zauzetog korisničkog imena i/ili e-maila, unos korisničkog podatka u nedozvoljenom formatu ili pružanje neispravnoga e-maila ili odabir adrese koja se ne podudara sa pravilima aplikacije

- i. Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za registraciju
- ii. Korisnik mijenja potrebne podatke te završava unos ili odustaje od registracije

UC3 - Prijava u sustav

1. **Glavni sudionik:** Ponuditelj
2. **Cilj:** Autentifikacija korisnika
3. **Sudionici:** Baza podataka
4. **Preduvjet:** Registracija
5. **Opis osnovnog tijeka:**
 - (a) Unos korisničkog imena i lozinke
 - (b) Potvrda o ispravnosti unesenih podataka
 - (c) Pristup funkcijama registriranih korisnika, ovisno o tipu ponuditelja
6. **Opis mogućih odstupanja:**
 - 2.a Neispravno korisničko ime/lozinka
 - i. Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za prijavu

UC4 - Pretraživanje knjiga po značajkama knjige

1. **Glavni sudionik:** Neregistrirani korisnik
2. **Cilj:** Pronalazak željene knjige
3. **Sudionici:** Baza podataka
4. **Preduvjet:** -
5. **Opis osnovnog tijeka:**
 - (a) Unos značajki knjige u odgovarajuće polje
 - (b) Izlistavanje knjiga koje se podudaraju sa tim značajkama

6. Opis mogućih odstupanja:

2.a Nepodudaranje sa ijednom knjigom u bazi podataka

- i. Sustav obavještava korisnika o tome da knjiga ili ne postoji ili nije dostupna

UC5 - Pretraživanje knjiga po nazivu ponuditelja

1. **Glavni sudionik:** Neregistrirani korisnik

2. **Cilj:** Izlistavanje knjiga određenog ponuditelja

3. **Sudionici:** Baza podataka

4. **Preduvjet:** -

5. Opis osnovnog tijeka:

- (a) Unos naziva ponuditelja u odgovarajuće polje
- (b) Izlistavanje knjiga koje su na ponuditeljevom popisu

6. Opis mogućih odstupanja:

2.a Nepodudaranje sa ijednim ponuditeljem u bazi podataka

- i. Sustav obavještava korisnika o tome da ponuditelj ili ne postoji ili nema nijednu knjigu na trenutnoj ponudi

UC6 - Zatražiti izdavača da kontaktira stranog izdavača oko prijevoda strane knjige na hrvatski

1. **Glavni sudionik:** Neregistrirani korisnik

2. **Cilj:** Zatražiti prijevod strane knjige na hrvatski jezik

3. **Sudionici:** Baza podataka

4. **Preduvjet:** Izdavač postoji i pronađen je kroz search

5. Opis osnovnog tijeka:

- (a) Neregistrirani korisnik upiše značajke strane knjige
- (b) Sustav obavještava korisnika da je upit poslan

UC7 - Dodavanje knjige

1. **Glavni sudionik:** Ponuditelj
2. **Cilj:** Dodati dostupnu knjigu
3. **Sudionici:** Baza podataka
4. **Preduvjet:** Ponuditelj je prijavljen u sustav
5. **Opis osnovnog tijeka:**
 - (a) Ponuditelj odabere opciju za dodavanje primjerka knjige
 - (b) Ponuditelj upisuje sve značajke knjige
 - (c) Potvrda o upisu knjige u bazu podataka kao nove knjige ili kao još jedan primjerak već postojeće

UC8 - Uklanjanje knjige

1. **Glavni sudionik:** Ponuditelj
2. **Cilj:** Izbrisati primjerak knjige
3. **Sudionici:** Baza podataka
4. **Preduvjet:** Ponuditelj je prijavljen u sustav
5. **Opis osnovnog tijeka:**
 - (a) Ponuditelj odabere opciju za brisanje primjerka knjige
 - (b) Ponuditelj upisuje sve značajke knjige
 - (c) Potvrda o smanjenju primjeraka knjige
6. **Opis mogućih odstupanja:**
 - 2.a Ponuditelj nije niti nudio knjigu ili je već na 0 primjeraka
 - i. Sustav obavještava ponuditelja o nemogućnosti uklanjanja primjerka određene knjige

UC9 - Uređivanje knjige

1. **Glavni sudionik:** Ponuditelj
2. **Cilj:** Urediti knjigu

3. **Sudionici:** Baza podataka

4. **Preduvjet:** Ponuditelj je prijavljen u sustav

5. **Opis osnovnog tijeka:**

- (a) Ponuditelj odabere opciju za uređivanje knjige
- (b) Ponuditelj upisuje promjene značajke knjige
- (c) Ponuditelj sprema promjene
- (d) Sustav potvrđuje promjene

6. **Opis mogućih odstupanja:**

2.a Ponuditelj nije niti nudio knjigu ili je već na 0 primjeraka

- i. Sustav obavještava ponuditelja o nemogućnosti uklanjanja primjerka određene knjige

UC10 - Neregistrirani korisnik vidi podatke za odabranu knjigu

1. **Glavni sudionik:** Ponuditelj

2. **Cilj:**

3. **Sudionici:** Baza podataka

4. **Preduvjet:** Knjiga postoji u bazi podataka

5. **Opis osnovnog tijeka:**

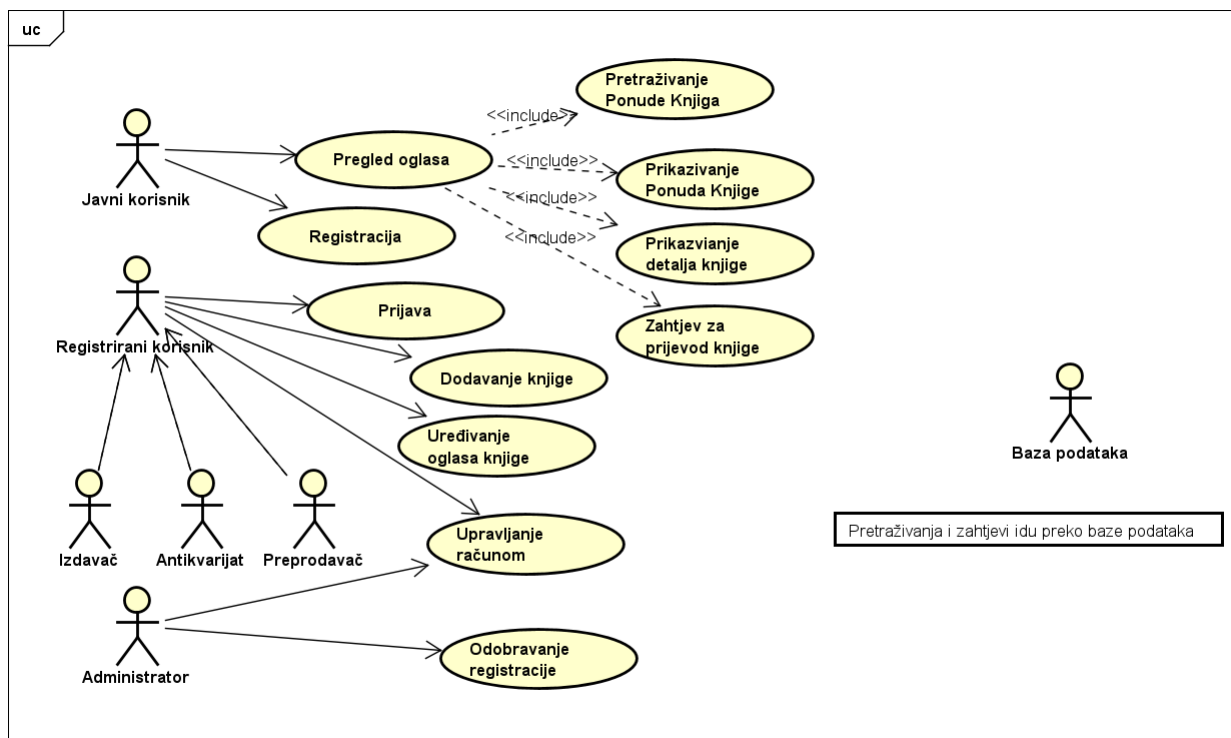
- (a) Neregistrirani korisnik klikne na knjigu
- (b) Ispisuje se popis dostupnih ponuda za odabranu knjigu, uključujući naziv ponuditelja, broj dostupnih primjeraka i cijenu knjige kod svakog ponuditelja.
- (c) Potvrda o smanjenju primjeraka knjige

6. **Opis mogućih odstupanja:**

2.a Ponuditelj nije niti nudio knjigu ili je već na 0 primjeraka

- i. Sustav obavještava ponuditelja o nemogućnosti uklanjanja primjerka određene knjige

Dijagrami obrazaca uporabe

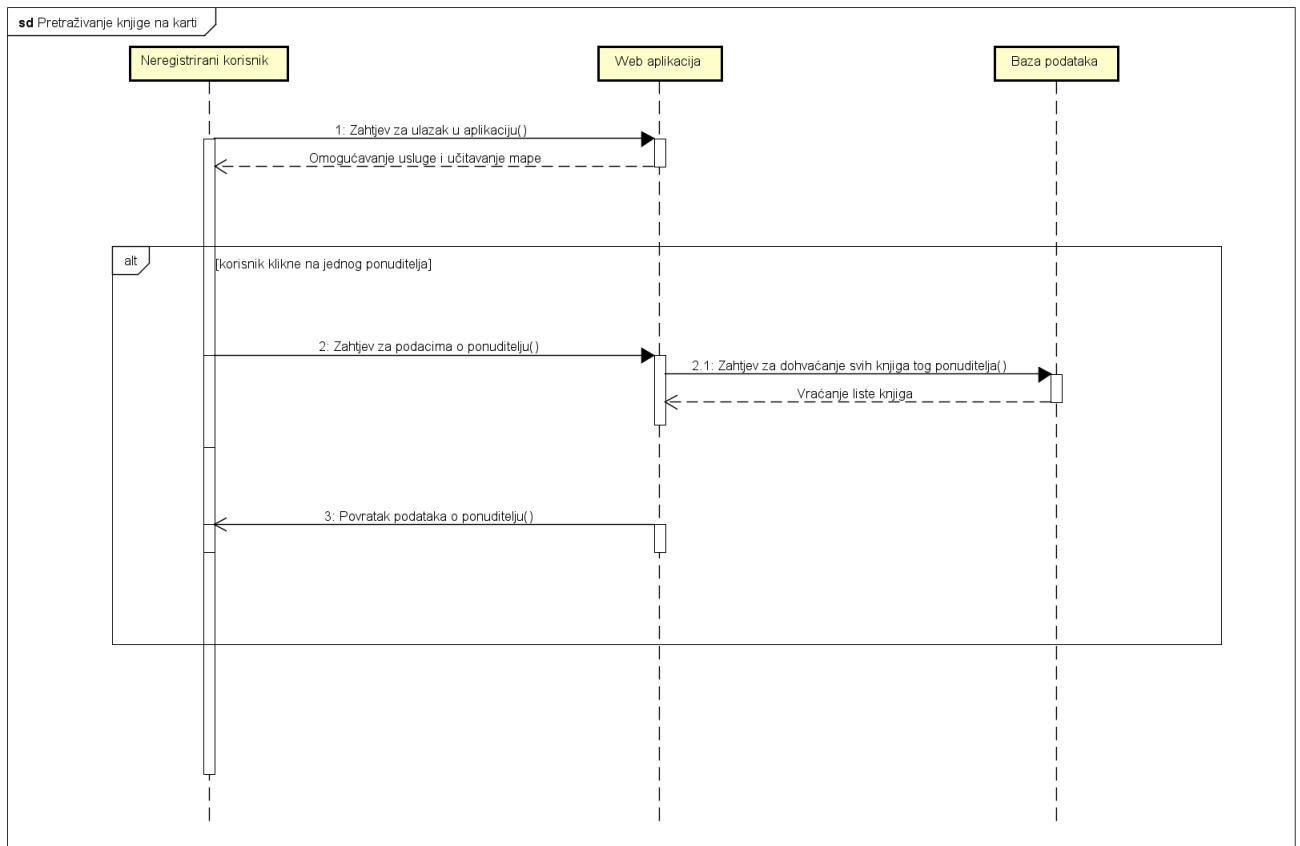


Slika 3.1: Dijagram obrasca uporabe

3.1.2 Sekvencijski dijagrami

Obrazac uporabe UC1 - Pretraživanje knjige na karti

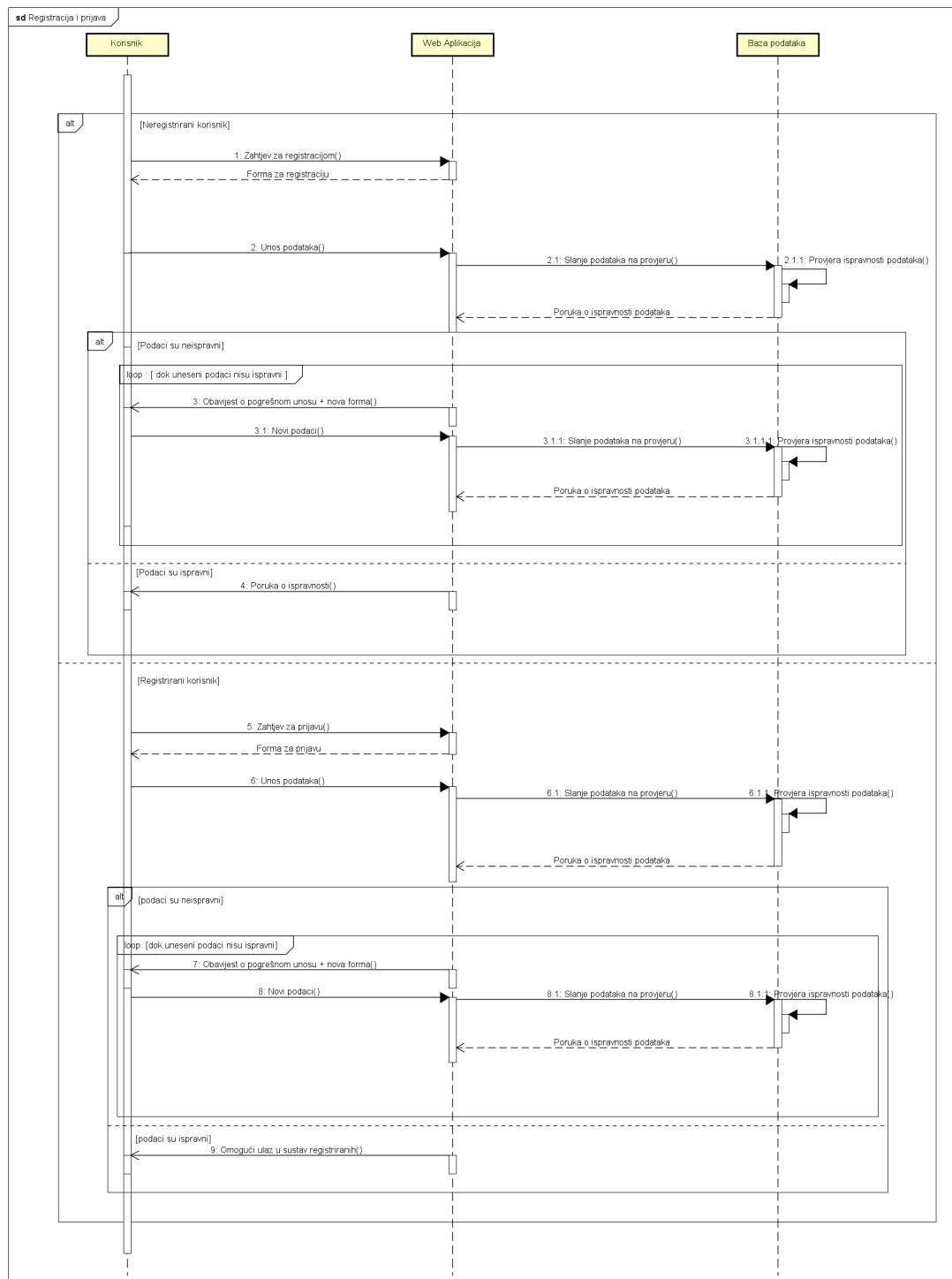
Neregistrirani korisnik šalje zahtjev za ulazak u aplikaciju koja učitava mapu. Neregistrirani korisnik šalje zahtjev za podacima o ponuditelju na kojeg klikne, poslužitelj iz baze podataka dohvaća sve knjige tog ponuditelja te ih vraća ponuditelju.



Slika 3.2: Sekvencijski dijagram za UC1

Obrazac uporabe UC2-3 - Registracija i prijava u sustav

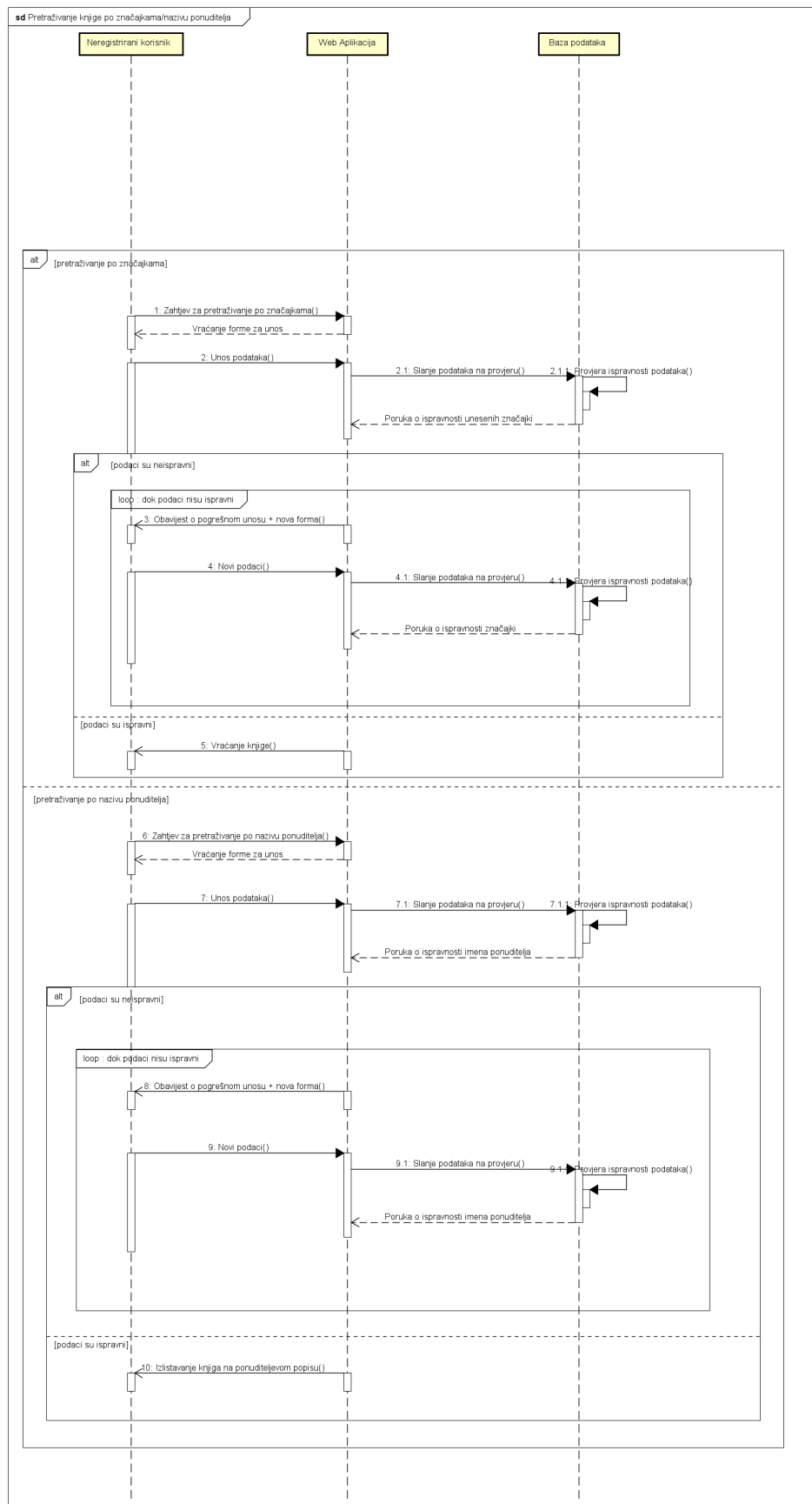
Neregistrirani korisnik šalje zahtjev za registracijom u aplikaciju. Poslužitelj vraća formu za registraciju. Neregistrirani korisnik upisuje podatke, poslužitelj šalje podatke na provjeru pomoću baze podataka. Ako su podaci neispravni, ponavlja se postupak upisa podataka, ako nisu, poslužitelj šalje poruku o ispravnosti. Ako se registrirani korisnik prijavljuje, sve je isto, samo nakon provjere i ako su podaci točni, omogućuje ulaz u sustav registriranih korisnika.



Slika 3.3: Sekvencijski dijagram za UC2-3

Obrazac uporabe UC4-5 - Pretraživanje knjige po značajkama knjige i nazivu ponuditelja

Neregistrirani korisnik šalje zahtjev za pretraživanjem po značajkama, poslužitelj mu vraća formu za unos. Provjeravaju se uneseni podaci te ako nisu ispravni, daje se nova forma za ispunjavanje, a ako jesu, vraćaju se knjige sa tim značajkama. Ista stvar se događa sa pretraživanjem po nazivu ponuditelja, samo se ovdje vraćaju knjige na ponuditeljevom popisu.



Slika 3.4: Sekvencijski dijagram za UC4-5

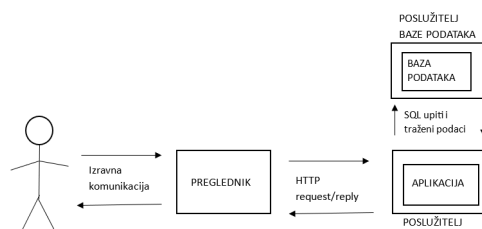
3.2 Ostali zahtjevi

- Sustav treba podržavati rad više korisnika u stvarnom vremenu
- Ponuditelj može dodati bilo koju knjigu za koju želi da bude ponuđena u aplikaciji, a drugim kanalima može nuditi neke druge knjige
- Komunikacija neregistriranih korisnika (onih koji traže neku knjigu) i ponuditelja ne odvija se kroz aplikaciju, već za to služe drugi kanali komunikacije (npr. e-pošta, telefon)
- Korisničko sučelje i sustav moraju podržavati hrvatsku abecedu (dijakritičke znakove) pri unosu i prikazu tekstualnog sadržaja
- Aplikacija treba biti izvedena kao web aplikacija prilagođena (engl. Responsive) mobilnom uređaju ili tabletu
- Registrirani korisnici trebaju joj pristupati uz pomoć korisničkog imena i lozinke
- Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava

4. Arhitektura i dizajn sustava

Kako bi se omogućio ispravan rad web aplikacije te interakcija između korisnika i nje potrebno je nekoliko osnovnih komponenata. To su:

- Web preglednik
- Web aplikacija
- Web poslužitelj
- Baza podataka i njezin poslužitelj



Slika 4.1: Prikaz komunikacije između navedenih dijelova sustava

Web preglednik (klijent) šalje zahtjeve poslužitelju te prima odgovore kako bi prikazao sadržaj potreban korisniku. Nakon što dohvati programski kod te označni (markup) tekst potreban za prikaz web stranice, on prevodi taj kod i prikazuje ga korisniku na unaprijed definiran način. Način na koji će preglednik prikazati određeni element ili strukturu markup teksta nije propisan samim markup kodom, već ga određuje sam preglednik. Osim osnovnog koda stranice (.html), preglednik dohvaća i .css i .js datoteke te multimedijske sadržaje kao što su slike, videozapisi, zvuk i sl.

Sama komunikacija između preglednika i poslužitelja odvija se protokolom HTTP (Hypertext transfer protokol). Nakon što poslužitelj primi HTTP zahtjev, prosljeđuje ga web aplikaciji koja ga zatim obrađuje. Rezultat obrade se, nakon toga, vraća poslužitelju, koji ga prosljeđuje natrag korisniku u obliku HTTP odgovora. Tijekom obrade zahtjeva, ovisno o njegovoj prirodi, web aplikacija ponekad

treba pristupiti bazi podataka. Bazi podataka se pristupa putem upita (npr. SQL). Koristit ćemo relacijsku bazu podataka i sustav PostgreSQL. Naime, s navedenom vrstom baza smo najbolje upoznati i imamo najviše iskustva u radu s njima. Osim toga, smatramo da nam omogućava ostvarenje svih planiranih funkcionalnosti potrebnih za našu aplikaciju. PostgreSQL se također temelji na modelu klijent – poslužitelj. Poslužitelj je u ovom slučaju povezan s bazom podataka te prima upite koje šalje web aplikacija. Server ih procesira, pribavlja tražene informacije iz baze podataka te ih šalje aplikaciji. Aplikacija tada može dovršiti započetu operaciju tj. obraditi zahtjev korisnika/preglednika.

Za izradu aplikacije koristit ćemo programski jezik Java. Naime, to je jezik s kojim je većina članova našeg tima najbolje upoznata. Kako bismo ostvarili funkcionalnost backenda za našu aplikaciju, uz programski jezik Java koristit ćemo Spring radni okvir (framework). Za frontend ćemo koristiti JavaScript i njegov React library koji će nam omogućiti jednostavniju i elegantniju izradu web stranice.

Arhitektura sustava temeljit će se na MVC konceptu (model-controller-view). Ovaj način razvoja aplikacije ima brojne prednosti, ponajviše glede odvojenog razvoja dijelova aplikacije. S obzirom na to da su odgovornosti razdvojene kad se primjenjuje ova arhitektura, mnogo je lakše zasebno razvijati svaki dio te ga testirati odvojeno od ostalih. Olakšano je pronalaženje pogrešaka u programskom kodu te vršenje izmjena nad postojećim dijelovima projekta, bez većeg kompromitiranja postojećih funkcionalnosti. Osim toga, ovakav pristup olakšao bi rad na skalabilnosti aplikacije, ako bi to bilo potrebno u bilo kojem trenutku. Još jedan od razloga za korištenje arhitekture temeljene na MVC načelu je činjenica da ju podržava Spring radni okvir koji koristimo. Dijelovi ove arhitekture su sljedeći:

- **Model** – obuhvaća logiku i funkcionalnost aplikacije u smislu obrade podataka i upravljanja istima. Prima podatke od controllera i na temelju toga izvršava potrebne radnje nad podacima te vraća rezultat natrag.
- **Controller** – poveznica između model i view komponenata. Prihvaća zahtjeve poslane od strane korisnika te određuje kako odgovoriti na njih. Te informacije prosljeđuje modelu. Kad od modela dobije odgovor, rezultat obrade vraća korisniku putem view komponente. Podatci koje controller dobiva od korisnika mogu biti klikovi na gumbе, poslani obrasci i sl.

- **View** – odnosi se na front end stranu aplikacije. To su dijelovi aplikacije koje korisnik vidi i s kojima izravno komunicira. View pribavlja podatke od modela te ih prikazuje na zaslonu računala na način razumljiv čovjeku. To su HTML, CSS datoteke i JS skripte na strani klijenta.

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

ponuditelj		
korisnickoIme	VARCHAR	Skup znakova koji se koriste za prepoznavanje i pristup
lozinka	VARCHAR	Lozinka se koristi u kombinaciji s korisničkim imenom
e-posta	VARCHAR	Način kontaktiranja
adresa	VARCHAR	Podatci o mjestu na kojem ponuditelj radi
brojTelefona	INT	Način kontaktiranja

knjige		
id	INT	Koristi za identifikaciju knjige
naziv	VARCHAR	Naslov knjige
autor	VARCHAR	Autor knjige
godinaIzdanja	INT	Godina izdanja
kategorijaIzdavaca	ENUM	Postoji dvije kategorije izdavača: domaći, strani
zanr	VARCHAR	Oblik književnog djela
ISBN	INT	Medunarodni standardni književni broj jedinstveni je identifikator knjiga
brojIzdanja	INT	Broj izdanja
stanjeOcuvanosti	VARCHAR	Stanje očuvanosti
tekstniOpis	VARCHAR	Sadržaj knjige
slikaKorica	IMAGE	Slika korica knjige
oznakaVrsteKnjige	ENUM	Oznaka vrste knjige može biti: strani-jezik, hrvatski-jezik-dobavljiva, hrvatski-jezik-nedobavljiva, srodni-jezik-nedobavljiva, srodni-jezik-dobavljiva
izdavac	VARCHAR	Naziv izdavača

listaPonuda		
id	INT	Koristi za identifikaciju knjige
brojDostupnihPrimjeraka	INT	Trenutni broj dostupnih primjeraka od određenog ponuditelja
cijenaKnjige	FLOAT	Cijena knjige

zahtjev		
id	INT	Koristi za identifikaciju knjige

ponuditi		
id	INT	Koristi za identifikaciju knjige
korisnickoIme	VARCHAR	Skup znakova koji se koriste za prepoznavanje i pristup

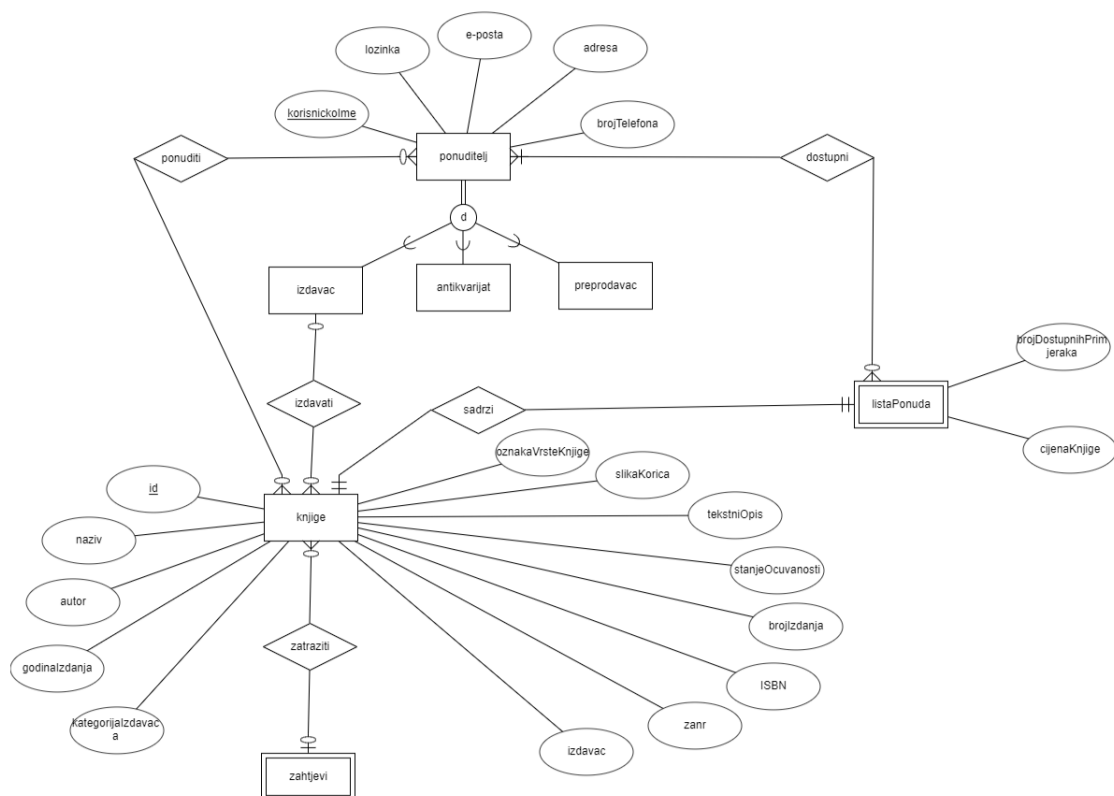
dostupni		
korisnickoIme	VARCHAR	Skup znakova koji se koriste za prepoznavanje i pristup

izdavac		
korisnickoIme	VARCHAR	Skup znakova koji se koriste za prepoznavanje i pristup

antikvarijat		
korisnickoIme	VARCHAR	Skup znakova koji se koriste za prepoznavanje i pristup

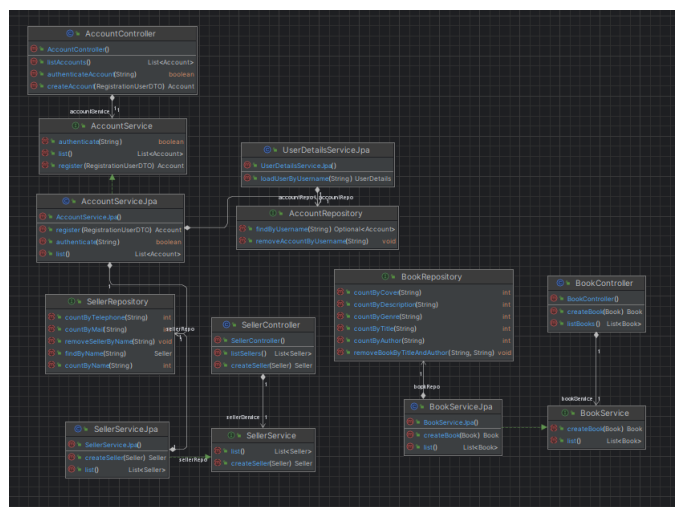
preprodavac		
korisnickoIme	VARCHAR	Skup znakova koji se koriste za prepoznavanje i pristup

4.1.2 Dijagram baze podataka



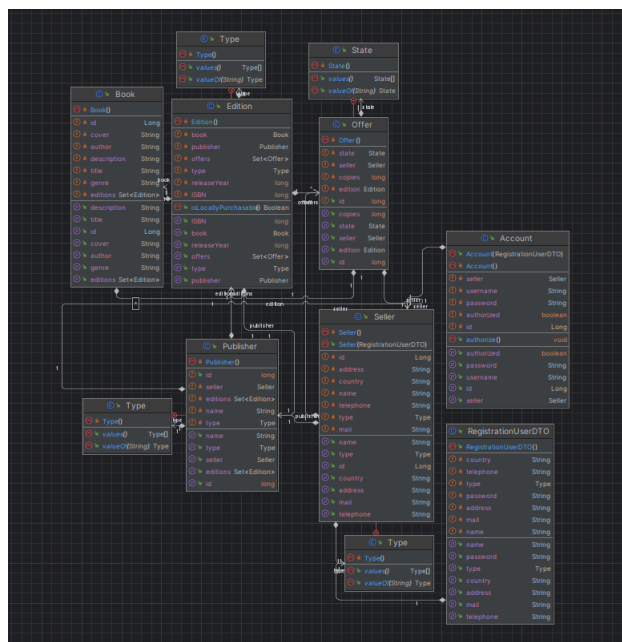
Slika 4.2: ER dijagram

4.2 Dijagram razreda



Slika 4.3: Dijagram razreda - odnos komponenti

Slika 4.3 prikazuje tok obrade HTTP zahtjeva od frontenda do baze podataka. Razred `AccountController` prima HTTP zahtjev poslan od strane frontenda, korisnika. Kako bi dogovorio na njih uspostavlja komunikaciju s Razredom `AccountServiceJpa`. Razred `AccountServiceJpa` zatim uspostavlja kontakt s `AccountService` koji zatim obavlja poslove vezane za korisnički račun. Isto tako Razred `AccountServiceJpa` uspostavlja komunikaciju s `AccountRepository` koji služi kao sučelje za pohranu i dohvat podatka iz baze. Isto tako postoji za `Seller` i `Book`. Imaju svoje `Controllere` za komunikaciju HTTP. Koji onda uspostavljaju komunikaciju sa `SellerServiceJpa` za `Seller` i `BookServiceJpa` za `Book`. Te `Book` i `Seller` imaju isto svaki svoj `Repository` koji služe za pohranu i dohvat podatka iz baze.



Slika 4.4: Dijagram razreda - repozitoriji i servisi

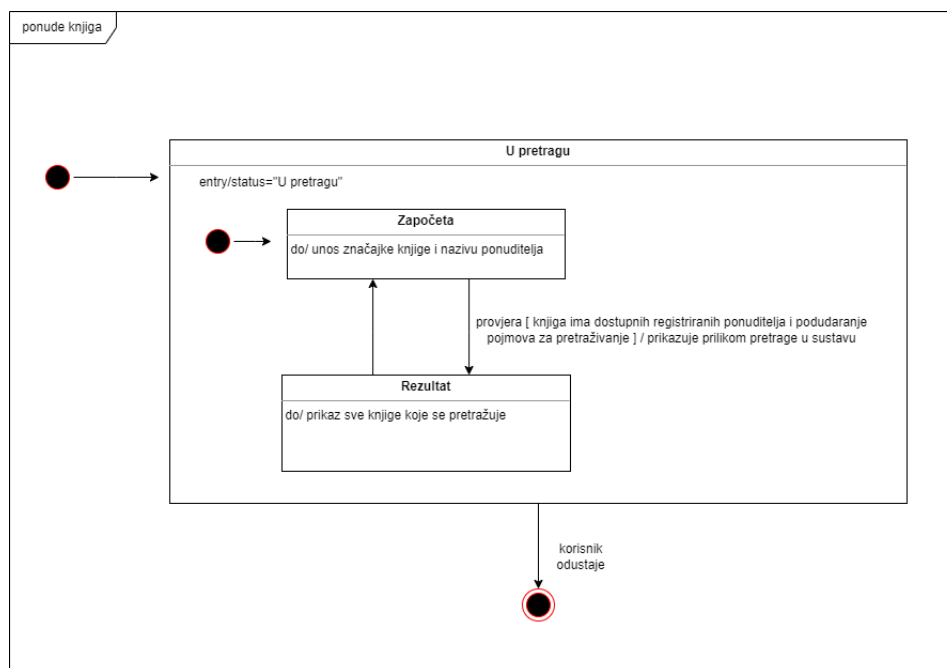
Slika 4.4 prikazuje razrede koji predstavljaju modele preslikane relacije baze podataka. Razred Account predstavlja registriranog korisnika. Razred Seller predstavlja vrstu registriranog korisnika koji ima 3 tipa: PUBLISHER, ANTIQUARIAN i RESELLER. Razred Publisher je tip SELLERA koji isto ima 2 tipa: LOCAL i FOREIGN. Razred Offer predstavlja objavljeni oglas. Razred Book predstavlja knjigu koja se objavljuje u oglasu te Razred Edition koji sadržava ograničenja za Razred Book i bitne kriterije za Offer.

dio 2. revizije

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

4.3 Dijagram stanja

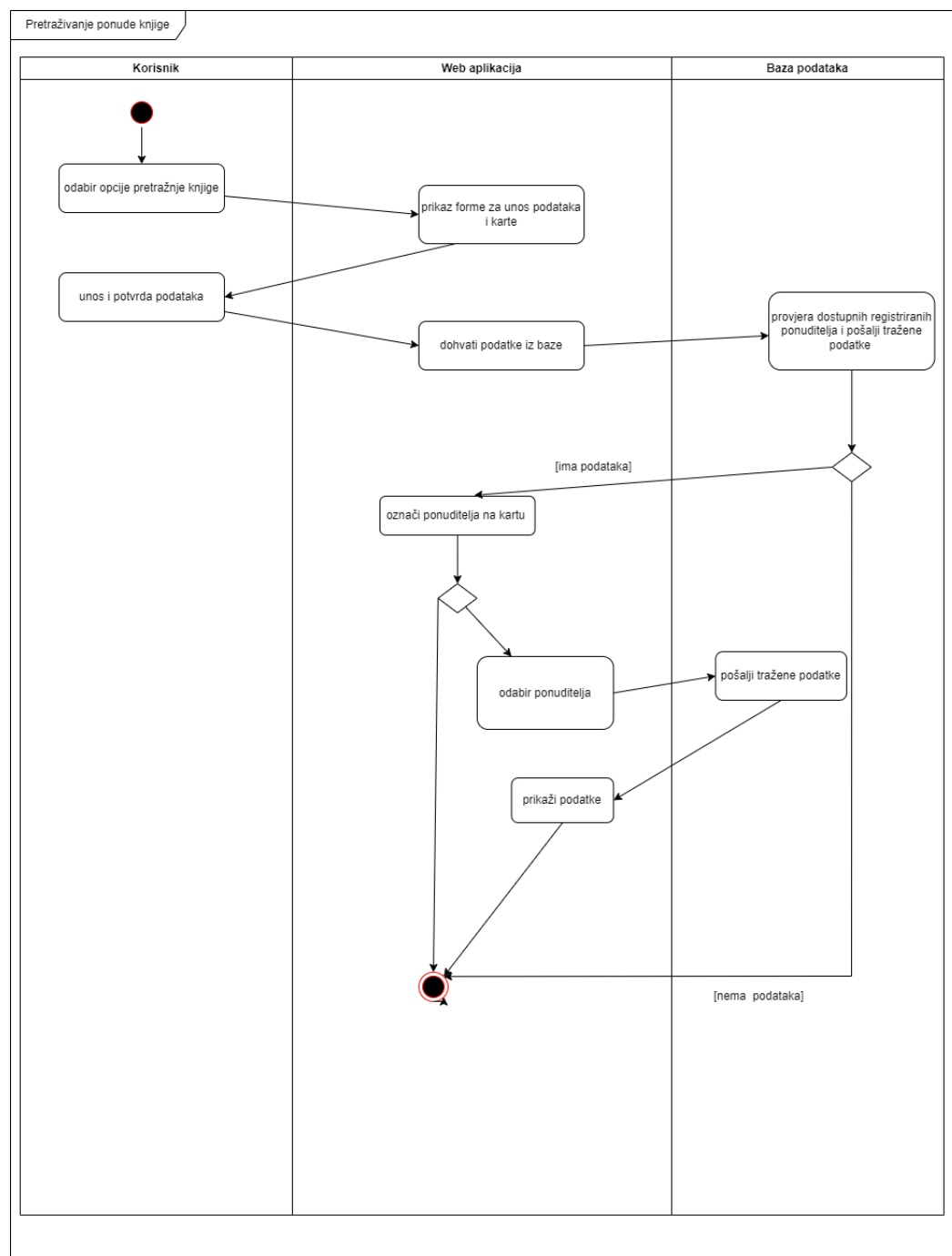
Web aplikacija omogućuje korisnicima pretraživanje ponude knjiga. Kad korisnik dođe na stranicu pretrage tada ima status 'U pretragu'. Korisnik tada može unijeti podatke za pretraživanje. Pri dobivanju rezultata, mora provjeriti je li knjiga ima dostupnih registriranih ponuditelja. Ako nema, onda knjigu se ne smije prikazivati prilikom pretrage u sustavu. Obrnuto, može se. Nakon što završe provjera, prikazuje sve knjige koje se pretražuje. Korisnik može ponoviti ovaj postupak dok ne odustaje od pretraživanja ili izlazak iz stranice pretrage.



Slika 4.5: Dijagram stanja

4.4 Dijagram aktivnosti

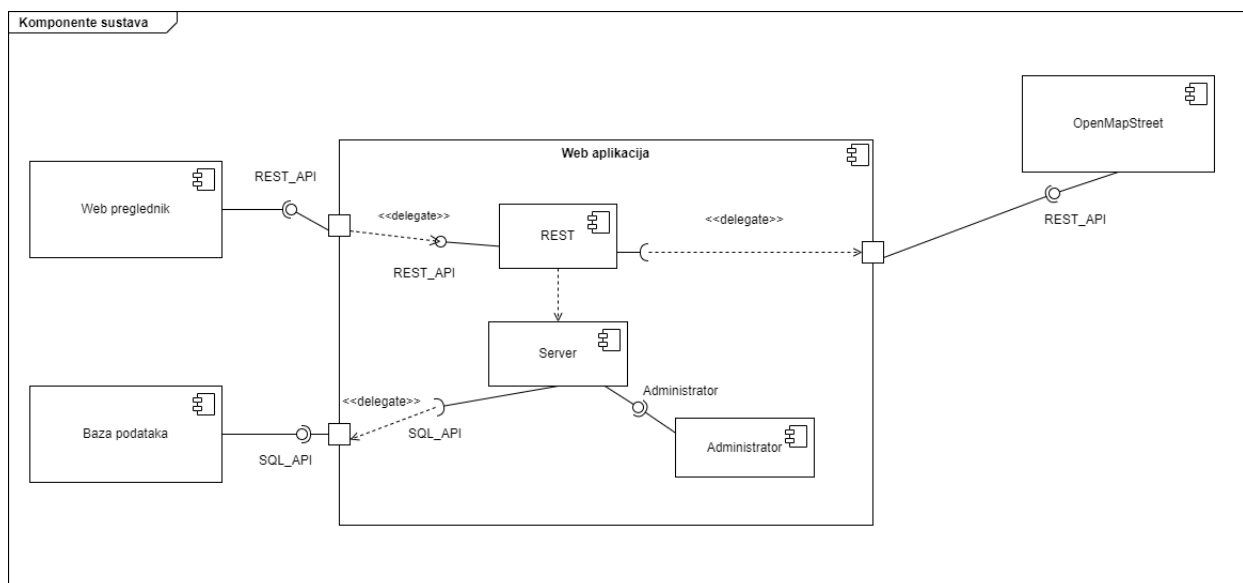
Proces pretraživanje započinje tako što korisnik odabere opciju pretraživanje knjige i zatim mu se otvara forma za unos podataka. Nakon potvrde upisanih podataka, web aplikacija se spaja na bazu podataka i pokušava dohvatiti podatke iz baze. Prije šaljenje, mora se provjeriti je li knjiga ima dostupnih registriranih ponuditelja. Nakon provjere ako web aplikacija dobije podaci, onda se mora označiti te podatke(ponuditelj) na kartu. Ako korisnik ne stisne te oznake na kartu ili ne dobije podaci, pretraživanje završava. Ako korisnik stisne bilo koje oznake na kartu, web aplikacija pokušava dobiti sve knjige dotičnog ponuditelja iz baze podataka. Nakon uspješno dobivanje podataka, treba prikazati na stranicu.



Slika 4.6: Dijagram aktivnosti

4.5 Dijagram komponenti

Korisnik pristupa web aplikaciji koristeći web preglednik i sučelje REST API koje su podržane operacije GET i POST protokola HTTP. Usluga za kartu OpenStreet-Map komunicira s web aplikacijom isto preko sučelja REST API. Web aplikacija je organizirana modularno. Modul REST odgovara na upite korisnika, komunicira s uslugom za kartu i modulom Server. Modul Server je zadužen za dohvat i obradu podatka iz baze preko sučelja SQL API te komunikaciju s modulom Administrator.



Slika 4.7: Dijagram komponenti

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

Komunikacija unutar tima ostvaruje se korištenjem popularnih aplikacija, kao što su **WhatsApp**^[1] i **Discord**^[2]. Za modeliranje UML dijagrama koristili smo **Astah UML**^[3].

Git^[4] je centralni sustav za upravljanje izvornim kodom, a projektni udaljeni repozitorij dostupan je na web platformi **GitHub**^[5].

Razvoj frontend-a izveden je u **JavaScriptu**^[6] koristeći biblioteku **React**^[7] koja je održavana od strane Facebooka. Za uređivanje koda koristili smo Microsoftov uređivač koda **Visual Studio Code (VSCode)**^[8], koji pruža izvrsnu podršku za razvoj modernih web aplikacija.

Za backend, koristili smo **IntelliJ IDEA**^[9], integrirano razvojno okruženje (IDE) tvrtke JetBrains. IntelliJ pruža snažnu potporu za razvoj Java Spring aplikacija.

Baza podataka koju koristimo je **PostgreSQL**^[10]. PostgreSQL je snažan open-source sustav za upravljanje bazama podataka.

Pogon aplikacije vršimo putem **Render**^[11] platforme, koja omogućava jednostavan i efikasan proces. Render pruža usluge poput web hostinga, skaliranja aplikacija i upravljanja resursima u oblaku.

Za dokumentaciju koristimo **TeXstudio**^[12] koji podržava LaTeX. LaTeX je sistem za pripremu dokumenata koji se koristi za visokokvalitetno formatiranje teksta.

¹<https://www.whatsapp.com/>

²<https://discord.com/>

³<https://astah.net/products/astah-uml/>

⁴<https://git-scm.com/>

⁵<https://github.com/>

⁶<https://www.javascript.com/>

⁷<https://reactjs.org/>

⁸<https://code.visualstudio.com/>

⁹<https://www.jetbrains.com/idea/>

¹⁰<https://www.postgresql.org/>

¹¹<https://render.com/>

¹²<https://www.texstudio.org/>

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 Selenium¹. 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:

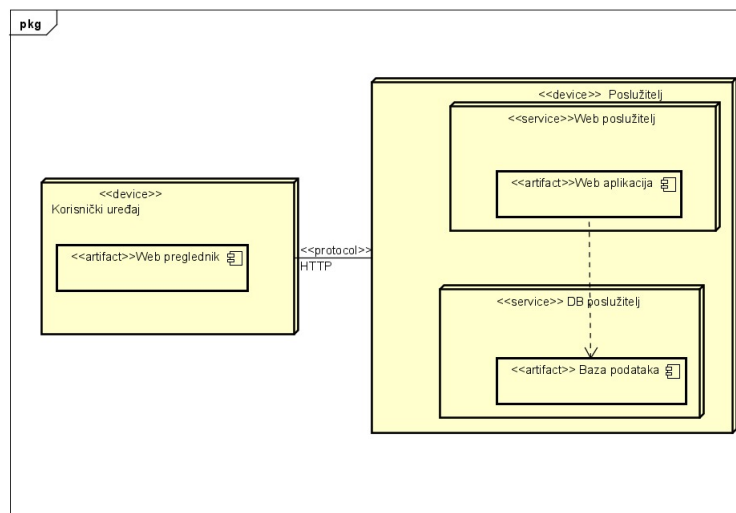
- *dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita*
- ***Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsko sučelje.*

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

¹<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

Slika 5.1 prikazuje odnos komponenti u sustavu. Klijent preko svog Korisničkog uređaja putem Web preglednika pristupa Poslužitelju putem HTTP protokola. Poslužitelj se sastoji od Web aplikacije koja izvlači podatke iz baze podataka te zato ovisi o njoj.



Slika 5.1: Dijagram Razmještaja

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

Zadatak naše grupe je bio razvoj web aplikacije za povezivanje čitatelja knjiga sa ponuditeljem, ali i za povezivanje ponuditelja međusobno. Aplikacija čitatelju ostvaruje mogućnost pretrage knjiga po njenim značajkama, po nazivu ponuditelja te nudi i pregled karte s označenim adresama ponuditelja. Isto tako aplikacija čitatelju nudi mogućnost slanja zahtjeva za prijevod knjige ako je ponuditelj vrste Izdavač. Ponuditelj kroz aplikaciju ima isto mogućnost pretrage, ali nakon registracije ima mogućnost unosa svoje ponude knjiga ovisno o vrsti ponuditelja koja se odabire prilikom registracije. Za ovaj zadatak smo imali 11 tjedana koji su podijeljeni na 2 faze.

Prva faza uključivala je okupljanje tima, dodjelu projektnog zadatka i većinom dokumentiranje zahtjeva. Naglasak je bio na izradi obrazaca i dijagrama koji su kasnije služili podtimovima backend i frontend u izradi aplikacije u drugoj fazi. U prvoj fazi nije bilo problema s dokumentacijom što se tiče dijagrama i obrazaca, već malih poteškoća s puštanjem aplikacije u pogon zbog manjka iskustva članova u alatima koji su korišteni u izradi aplikacije. No uz trud članova prošlo je uspješno.

Druga faza imala je veći fokus na poboljšanju funkcionalnosti i samostalnom radu na aplikaciji. Isto tako je bilo par poteškoća zbog manjka iskustva što je navelo članove tima na istraživanje i učenje korištenja njima dotad stranih alata i programa, no na kraju je i to savladano i projekt je uspješno napravljen.

Komunikacija u timu se vodila putem WhatsAppa i Discorda radi lakšeg informiranja članova o napretku te međusobnog pomaganja. Za budućnost moguće je usavršavanje web aplikacije s ostvarivanjem veće funkcionalnosti te moguće izrade mobilne aplikacije koja bi bolje ostvarila cilj povezivanja čitatelja i ponuditelja.

Rad na projektu bio je vrijedno iskustvo svim članovima tima jer smo u kratkom periodu stekli puno znanja, ali i bolje shvatili mogućnosti rada u timu. Unatoč par poteškoća u izradi zbog manjka iskustva zadovoljni smo odrađenim projektom i napravljenom aplikacijom.

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>
7. metLib aplikacija, Point d.o.o., <https://play.google.com/store/apps/details?id=com.metlib.librarymetel&hl=hr&gl=US>

Indeks slika i dijagrama

2.1	Primjer prijave u aplikaciju	8
2.2	Primjer glavnog prozora	9
2.3	Primjer liste knjiga	10
2.4	Primjer pregleda odabrane knjige	11
3.1	Dijagram obrasca uporabe	19
3.2	Sekvencijski dijagram za UC1	20
3.3	Sekvencijski dijagram za UC2-3	22
3.4	Sekvencijski dijagram za UC4-5	24
4.1	Prikaz komunikacije između navedenih dijelova sustava	26
4.2	ER dijagram	31
4.3	Dijagram razreda - odnos komponenti	32
4.4	Dijagram razreda - repozitoriji i servisi	33
4.5	Dijagram stanja	34
4.6	Dijagram aktivnosti	36
4.7	Dijagram komponenti	37
5.1	Dijagram Razmještaja	41

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

1. sastanak

- Datum: 15. listopada 2023.
- Prisustvovali: Sven Winkler, Luka Kitarović, Yu Xing Jin, Božidar Pučar, Andrej Lovei, Vedran Marković, Damjan Šarlija
- Teme sastanka:
 - razrada zadatka
 - podijela članova na backend i frontend
 - podijela dijelova dokumentacije

2. sastanak

- Datum: 29. listopada 2023.
- Prisustvovali: Sven Winkler, Luka Kitarović, Yu Xing Jin, Božidar Pučar, Andrej Lovei, Vedran Marković, Damjan Šarlija
- Teme sastanka:
 - rasprava o stanju dokumentacije
 - početna podijela backend i frontend zadataka

3. sastanak

- Datum: 5. studenog 2023.
- Prisustvovali: Sven Winkler, Luka Kitarović, Yu Xing Jin, Božidar Pučar, Andrej Lovei, Vedran Marković, Damjan Šarlija
- Teme sastanka:
 - razgovor o napretku dokumentacije i implementacije

4. sastanak

- Datum: 12. studenog 2023.
- Prisustvovali: Sven Winkler, Luka Kitarović, Yu Xing Jin, Božidar Pučar, Andrej Lovei, Vedran Marković, Damjan Šarlija

- Teme sastanka:
 - razgovor o napretku dokumentacije i implementacije

5. sastanak

- Datum: 17. listopada 2023.
- Prisustvovali: Sven Winkler, Luka Kitarović, Yu Xing Jin, Božidar Pučar, Andrej Lovei, Vedran Marković, Damjan Šarlija
- Teme sastanka:
 - rad na implementaciji

Tablica aktivnosti

Kontinuirano osvježavanje

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

	Sven Winkler	Luka Kitarović	Yu Xing Jin	Božidar Pučar	Andrej Lovei	Vedran Marković	Damjan Šarlija	
Upravljanje projektom	6							
Opis projektnog zadatka				5				
Funkcionalni zahtjevi					2			
Opis pojedinih obrazaca					2			
Dijagram obrazaca					3			
Sekvencijski dijagrami	4							
Opis ostalih zahtjeva					1			
Arhitektura i dizajn sustava							4	
Baza podataka			5					
Dijagram razreda	2				3			
Dijagram stanja								
Dijagram aktivnosti								
Dijagram komponenti								
Korištene tehnologije i alati								
Ispitivanje programskog rješenja								
Dijagram razmještaja								
Upute za puštanje u pogon	3							

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

	Sven Winkler	Luka Kitarović	Yu Xing Jin	Božidar Pučar	Andrej Lovei	Vedran Marković	Damjan Šarlija	
Dnevnik sastajanja				2				
Zaključak i budući rad								
Popis literature				1				
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>								
<i>frontend</i>			10	10			10	
<i>izrada baze podataka</i>		3						
<i>spajanje s bazom podataka</i>	2	2			2	2		
<i>back end</i>	10	13			5	3		

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 stranice, u izborniku Repository, pritiskom na stavku Contributors.