

Programsko inženjerstvo

Ak. god. 2021./2022.

SkillEtCooking

Dokumentacija, Rev. 1.

Grupa: *Progimeri*

Voditelj: *Karlo Frankola*

Datum predaje: *15. studenog 2021.*

Nastavnik: *Eugen Vušak, mag. ing. comp*

Sadržaj

| | |
|--|-----------|
| 1 Dnevnik promjena dokumentacije | 3 |
| 2 Opis projektnog zadatka | 5 |
| 3 Specifikacija programske potpore | 10 |
| 3.1 Funkcionalni zahtjevi | 10 |
| 3.1.1 Obrasci uporabe | 12 |
| 3.1.2 Sekvencijski dijagrami | 21 |
| 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 | 32 |
| 4.2 Dijagram razreda | 33 |
| 4.3 Dijagram stanja | 36 |
| 4.4 Dijagram aktivnosti | 37 |
| 4.5 Dijagram komponenti | 38 |
| 5 Implementacija i korisničko sučelje | 39 |
| 5.1 Korištene tehnologije i alati | 39 |
| 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

| Rev. | Opis promjene/dodatka | Autori | Datum |
|--------|--|------------------|-------------|
| 0.1 | Napravljen predložak | Karlo Frankola | 15.10.2021. |
| 0.2 | Opis projektnog zadatka | Dario Dugonjevac | 28.10.2021. |
| 0.3 | Dodani funkcionalni zahtjevi | Borna Colarić | 31.10.2021. |
| 0.4 | Dodani obrasci uporabe | Toni Serezlija | 01.11.2021. |
| 0.5 | Napravljeni dijagrami obrazaca uporabe | Marko Tunjić | 02.11.2021. |
| 0.6 | Sekvencijski dijagrami | Dario Dugonjevac | 12.11.2021. |
| 0.7 | Definirani nefunkcionalni zahtjevi | Toni Serezlija | 13.11.2021. |
| 0.8 | Opis tablica baze podataka | Karlo Frankola | 14.11.2021. |
| 0.9 | Dijagram baze podataka | Marko Tunjić | 15.11.2021. |
| 0.10 | Dijagram razreda | Jan Brkić | 16.11.2021. |
| 0.10.1 | Dopunjeni dijagrami razreda | Karlo Frankola | 19.11.2021. |

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

| Rev. | Opis promjene/dodatka | Autori | Datum |
|------|---|----------------|-------------|
| 1.0 | Dopuna dnevnika aktivnosti i prva predaja | Karlo Frankola | 19.11.2021. |

2. Opis projektnog zadatka

Cilj ovog projekta je razviti programsku potporu za stvaranje web aplikacije "SkillEtCooking" koja će korisniku olakšati snalaženje u kuhinji i optimizaciju iskorištavanja sastojaka koje osoba već posjeduje. Na taj način korisnik neće morati provesti cijeli dan razmišljajući što i kako napraviti nego to aplikacija radi umjesto njega.

Sustav će podržavati više vrsta korisnika:

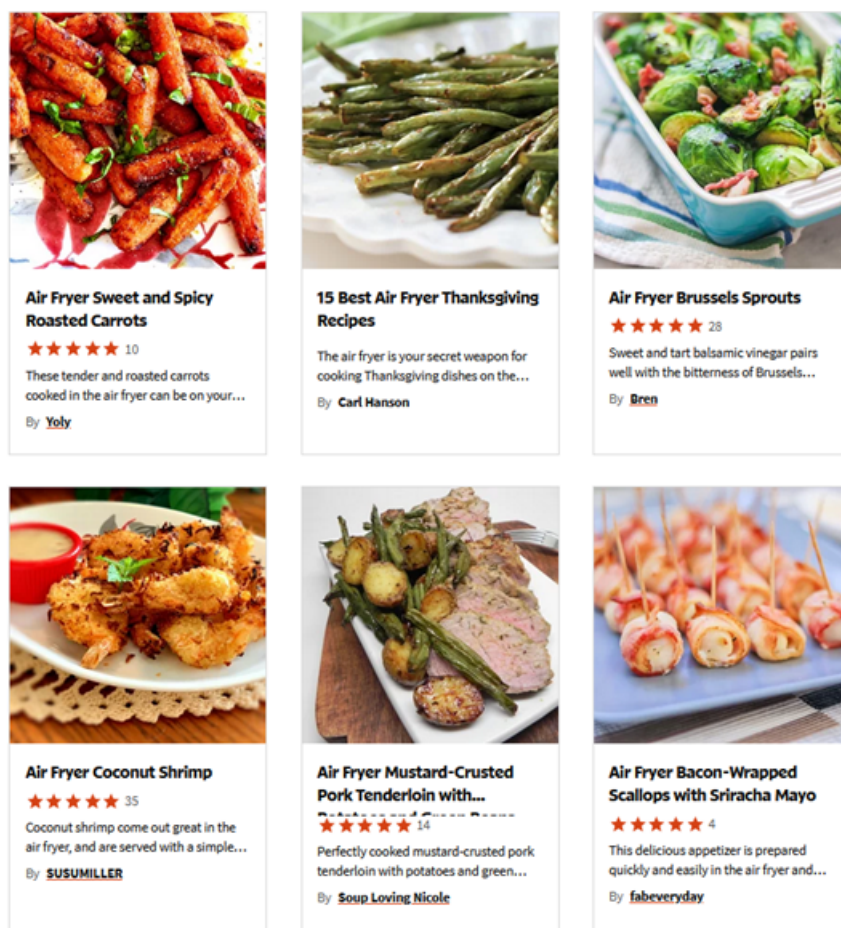
- registrirani
- neregistrirani
- moderatori.

Neregistriranim korisnicima se prilikom otvaranja aplikacije otvori početna stranica na kojoj se nalaze svi recepti (4.5). Recepti se mogu sortirati na razne načine:

- popularnost
- prosječna ocjena recepta
- oznaka "Preporučeno".

Popularnost se određuje na osnovu broja otvaranja pojedinog recepta dok se sortiranje po oznaci "Preporučeno" određuje na osnovu prosjeka popularnosti i prosječne ocjene koristeći funkciju:

- $ocjena/prosjecnaOcjena + brojOtvaranja/prosjecniBrojOtvaranja$



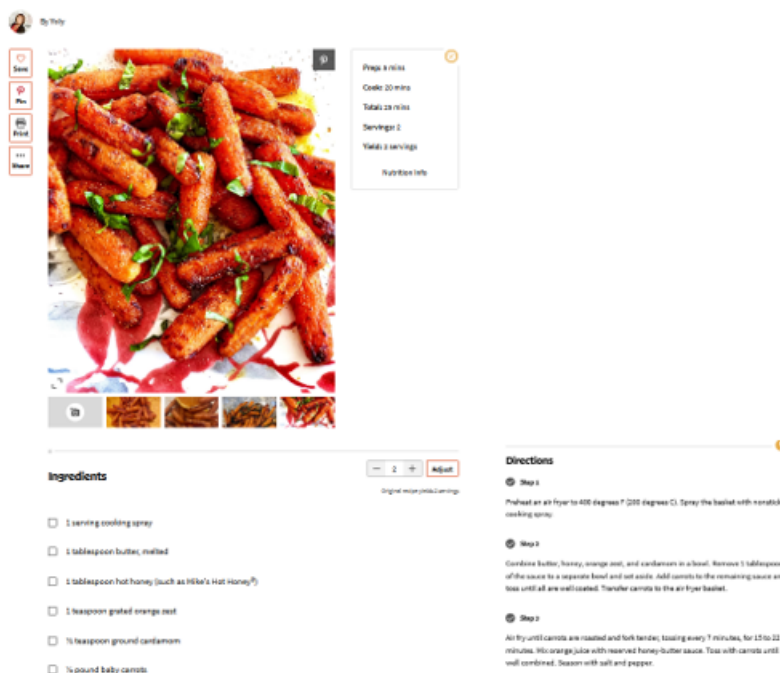
Slika 2.1: Prikaz svih recepata

Neregistrirani korisnici imaju mogućnost filtriranja recepata upisivanjem imena recepta ili sastojka/sastojaka. Nakon filtriranja skupa recepata po sastojcima, recepti se sortiraju po vrijednosti Jaccardovog indeksa sličnosti s time da najbližnji dolazi prvi te se izbacuju svi recepti čija je vrijednost indeksa sličnosti manja od praga. Za to vrijeme, ostali izbori sortiranja su onemogućeni, ali pretraživanje po naslovu je i dalje u funkciji.

Sljedeća mogućnost neregistriranih korisnika je da im se pritiskom na naziv recepta prikazuje stranica s detaljima recepta. Na toj stranici vidljivi su svi podatci vezani za recept:

- slika
- naziv
- procijenjeno vrijeme kuhanja
- sastojci i količina svakog sastojka
- koraci pripreme s kratkim opisom

- ocjena (2.2).

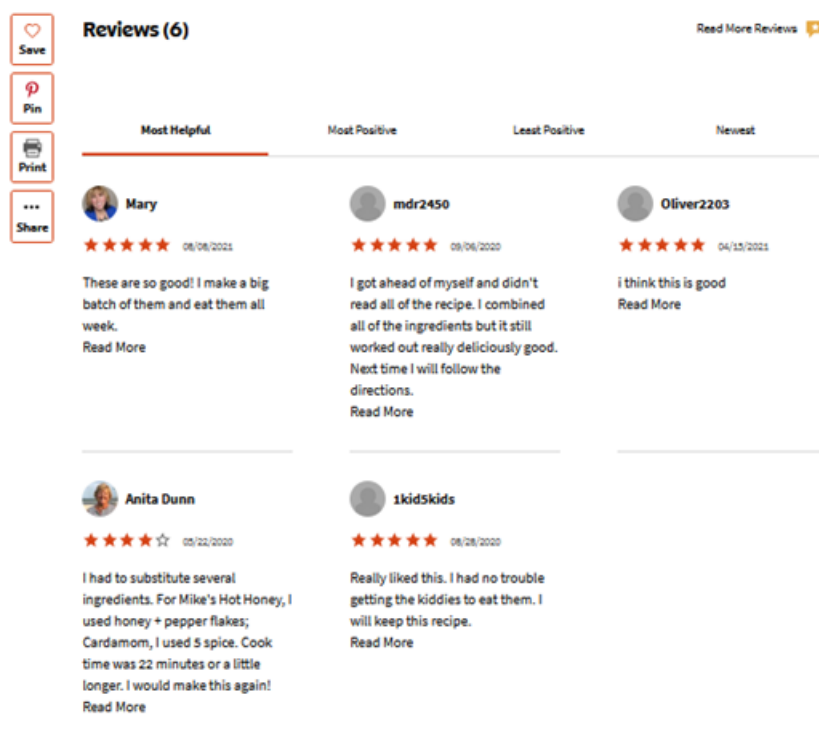


Slika 2.2: Primjer otvorenog recepta

Stranica sadrži prostor sa komentarima i ocjenama gdje je moguće vidjeti iskustva drugih korisnika što će biti izvedeno korištenjem Disqus usluge (2.3). Tu je vidljiva prva razlika između registriranog i neregistriranog korisnika, a to je da komentare i ocjene mogu ostaviti samo registrirani korisnici. Također ukoliko je autor ostavio komentar na svoj recept, komentar će biti dodatno naglašen i bolje uočljiv ostalim korisnicima.

Osim toga, stranica sadrži i mogućnost da klikom na gumb korisnik može pregledati sve ostale recepte s istim autorom kao i odabrani recept.

Registrirani korisnik se može prijaviti, a neregistrirani će imati mogućnost registracije u sustav (2.4). Prijavljeni korisnik može koristiti aplikaciju na jednak način kao i neprijavljeni korisnik, ali prijavljeni korisnik također može dodavati i uređivati vlastite recepte, uređivati vlastite podatke te koristiti prethodno objašnjenu funkcionalnost ostavljanja komentara. Pravila prilikom unosa recepta su da korisnik mora unijeti sliku jela (maksimalno 5), barem jedan sastojak i količinu te barem jedan korak pripreme.



Slika 2.3: Komentari na recept

The screenshot shows a user interface for logging in or signing up. It features two tabs: 'LOG IN' and 'SIGN UP'. The 'LOG IN' tab is currently selected, displaying a form with an 'Email Address' field, a 'Password' field, a 'Remember me' checkbox, and a blue 'LOG IN' button. The 'SIGN UP' tab is also visible, showing a form with 'First Name*', 'Last Name*', 'Email Address*', 'Username*', 'Password*', and 'Retype Password*' fields, and a blue 'SUBMIT' button.

Slika 2.4: Prijava i registracija u sustav

Sustav koriste i moderatori koji su interni zaposlenici projekta, a njihovi računi se dodaju direktno u bazu podataka. Moderatori se prijavljuju u sustav jednako kao i ostali korisnici. Oni mogu dodati komentar na nekome receptu koji će kao i kod autora biti dodatno vizualno označen, ali ne mogu ocjenjivati recepte.

Dodatne mogućnosti moderatora su:

- brisanje komentara
- brisanje recepata
- pregled korisnika sustava.

Također sustav mora biti funkcionalan ukoliko je više korisnika prijavljeno u isto vrijeme neovisno o tome jesu li moderatori ili obični registrirani korisnici.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Korisnici
 - (a) Registrirani/prijavljeni
 - (b) Neregistrirani
2. Moderator
3. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (inicijator) može:
 - (a) Pregledati sve recepte
 - (b) Sortirati i filtrirati recepte po raznim argumentima
 - (c) Filtrirati recepte po sastojcima
 - (d) Pretražiti recepte po imenu/autoru
 - (e) Registrirati se
2. Registrirani/prijavljeni korisnik (inicijator) može:
 - (a) Raditi sve kao i neregistrirani/neprijavljeni korisnik
 - (b) Dodati vlastiti recept (Dodani recept mora sadržavati maksimalno 5 slika, te po jedan sastojak i korak pripreme)
 - (c) Urediti vlastiti recept (obrisati/dodati sastojke, urediti i dodati korake i opis)
 - (d) Dodavati komentare na recepte
 - (e) Brisati vlastite komentare

3. Moderator (inicijator) ima mogućnost:

- (a) Brisanja i dodavanja komentara, ali ne i ocjenjivanja recepata
- (b) Brisanja recepata
- (c) Brisanja korisničkih računa
- (d) Pregleda svih korisnika

4. Baza podataka (sudionik) mora moći:

- (a) Spremati sve podatke o receptima
- (b) Spremati sve podatke o korisnicima
- (c) Spremati sve podatke o moderatorima
- (d) Izvršiti zadani upit i vratiti rezultat

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 - Sortiranje recepata po popularnosti

- **Glavni sudionik:** Korisnik
- **Cilj:** Filtrirati recepte po popularnosti
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Prilikom učitavanja je prikazana početna stranica na kojoj se nalazi lista recepata
 2. Korisnik na karti odabire način sortiranja: prema popularnosti
 3. Prikazuju se recepti sortirani silazno prema broju otvaranja

UC2 - Sortiranje recepata prema prosječnoj ocjeni

- **Glavni sudionik:** Korisnik
- **Cilj:** Filtrirati recepte prema prosječnoj ocjeni
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Prilikom učitavanja je prikazana početna stranica na kojoj se nalazi lista recepata
 2. Korisnik na karti odabire način sortiranja: prema prosječnoj ocjeni
 3. Prikazuju se recepti sortirani silazno prema prosječnoj ocjeni korisnika

UC3 - Sortiranje recepata prema oznaci "Preporučeno"

- **Glavni sudionik:** Korisnik
- **Cilj:** Filtrirati recepte prema oznaci "Preporučeno"
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Prilikom učitavanja je prikazana početna stranica na kojoj se nalazi lista recepata
 2. Korisnik na karti odabire način sortiranja: prema prosječnoj ocjeni

3. Prikazuju se recepti sortirani silazno prema oznaci “Preporučeno” koja se određuje kao funkcija popularnosti i ocjene koja je prepuštena projektantima

UC4 – Pretraživanje recepata po naslovu

- **Glavni sudionik:** Korisnik
- **Cilj:** Prikazati sve recepte koje sadrže upisani termin
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Prilikom učitavanja je prikazana početna stranica na kojoj se nalazi lista recepata
 2. Korisnik pretražuje određeni termin tako da upiše naslov recepta kojeg želi otvoriti
 3. Prikazuju se svi recepti koji sadrže upisani termin

UC5 – Pretraživanje recepata po sastojcima

- **Glavni sudionik:** Korisnik
- **Cilj:** Prikazati sve recepte čiji sastojci su najsličniji unesenim sastojcima
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Prilikom učitavanja je prikazana početna stranica na kojoj se nalazi lista recepata
 2. Korisnik upisuje sastojke koje ima pri ruci
 3. Prikazuju se recepti sortirani po vrijednosti indeksa sličnosti

UC6 – Registracija

- **Glavni sudionik:** Korisnik
- **Cilj:** Stvoriti korisnički račun za pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za registraciju
 2. Korisnik unosi potrebne korisničke podatke

3. Korisnik prima obavijest o uspješnoj registraciji te se podaci spremaju u bazu podataka
- **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
 1. Sustav obavještava korisnika o neuspjelom upisu i vraća ga na stranicu za registraciju
 2. Korisnik mijenja potrebne podatke te završava unos ili odustaje od registracije

UC7 – Prijava u sustav

- **Glavni sudionik:** Korisnik
- **Cilj:** Dobiti pristup dodatnim funkcionalnostima poput unosa, pregledavanja i uređivanja
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija
- **Opis osnovnog tijeka:**
 1. Unos korisničkog imena i lozinke
 2. Potvrda o ispravnosti unesenih podataka
 3. Pristup korisničkim funkcijama
- **Opis mogućih odstupanja:**
 - 1.a Neispravno korisničko ime ili lozinka
 1. Sustav obavještava korisnika o neuspjelom upisu i omogućuje mu ponovno prijavu u sustav

UC8 – Pregled pojedinog recepta

- **Glavni sudionik:** Korisnik
- **Cilj:** Prikazati sve podatke vezane za recept
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik odabere jedan od ponuđenih recepata
 2. Otvori se stranica sa svim informacijama o receptu

UC9 – Komentiranje i ocjenjivanje recepata

- **Glavni sudionik:** Korisnik
- **Cilj:** Ostaviti komentar i ocjenu za određeni recept
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik napiše recenziju i ocijeni recept
 2. Ocjena i osvrt se pohranjuju u bazu podataka i ažurira se prosječna ocjena recepta

UC10 – Pregled svih recepata određenog autora

- **Glavni sudionik:** Korisnik
- **Cilj:** Prikazati sve recepte određenog autora
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti na stranici recepta
- **Opis osnovnog tijeka:**
 1. Korisnik se nalazi na stranici recepta
 2. Korisnik odabire opciju “Prikaži sve recepte autora“
 3. Prikazuju se svi recepti autora uključujući i odabrani recept

UC11 – Dodavanje recepta

- **Glavni sudionik:** Korisnik
- **Cilj:** Dodati novi recept
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za dodavanje novog recepta
 2. Korisnik prilaže maksimalno 5 slika jela, sastojke (makar 1) i količinu te korake pripreme (makar 1)
 3. Ukoliko je korisnik zadovoljio uvjete, podaci se spremaju u bazu podataka te su vidljivi u aplikaciji
- **Opis mogućih odstupanja:**
 - 2.a Korisnik nije priložio niti jedan sastojak ili jedan korak pripreme
 1. Sustav obavještava korisnika o uvjetima koji nisu zadovoljeni
 2. Korisnik dodaje potrebne podatke te završava unos ili odustaje od dodavanja recepta

UC12 – pregled vlastitih recepata

- **Glavni sudionik:** Korisnik
- **Cilj:** Prikazati sve vlastite recepte
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju “Pregledaj recepte”
 2. Korisniku se prikazuju svi recepti kojima je on autor

UC13 – Uređivanje recepta

- **Glavni sudionik:** Korisnik
- **Cilj:** Urediti željeni recept
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je autor recepta
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju uređivanja recepta
 2. Korisnik unosi željene podatke
 3. Ukoliko su uvjeti zadovoljeni, korisnik prima obavijest o uspješnom uređivanju recepta
- **Opis mogućih odstupanja:**
 - 2.a Korisnik je prilikom uređivanja uklonio sve sastojke ili korake pripreme
 1. Sustav obavještava korisnika o uvjetima koji nisu zadovoljeni
 2. Korisnik dodaje potrebne podatke te završava izmjene ili odustaje od uređivanja recepta te recept ostaje nepromijenjen

UC14 – Dodavanje komentara moderatora

- **Glavni sudionik:** Korisnik
- **Cilj:** Dodati komentar na određeni recept
- **Sudionici:** Baza podataka
- **Preduvjet:** Moderator mora biti prijavljen
- **Opis osnovnog tijeka:**
 1. Moderator ostavlja komentar koji je u konačnici vizualno dodatno naglašen
 2. Komentar se pohranjuje u bazu podataka

UC15 – Brisanje komentara

- **Glavni sudionik:** Korisnik
- **Cilj:** Obrisati komentare koji nisu primijenjeni
- **Sudionici:** Baza podataka
- **Preduvjet:** Moderator mora biti prijavljen
- **Opis osnovnog tijeka:**
 1. Moderator odabire željeni komentar za određeni recept
 2. Klikom na komentar prikaže mu se opcija “Izbriši”
 3. Komentar se uklanja iz baze podataka

UC16 – Brisanje vlastitih komentara

- **Glavni sudionik:** Korisnik
- **Cilj:** Autor želi izbrisati svoj komentar
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik na stranici recepta odabire vlastiti komentar
 2. Klikom na komentar prikaže mu se opcija “Izbriši”
 3. Komentar se uklanja iz baze podataka

UC17 - Brisanje recepata

- **Glavni sudionik:** Moderator
- **Cilj:** Obrisati željene recepte
- **Sudionici:** Baza podataka
- **Preduvjet:** Moderator mora biti prijavljen
- **Opis osnovnog tijeka:**
 1. Moderator odabire željeni recept
 2. Klikom na opciju “Izbriši” recept se uklanja iz baze podataka

UC18 - Brisanje vlastitih recepata

- **Glavni sudionik:** Korisnik
- **Cilj:** Autor želi obrisati svoj recept
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti prijavljen
- **Opis osnovnog tijeka:**

1. Korisnik odabire recept kojemu je on autor
2. Klikom na opciju "Izbriši" recept se uklanja iz baze podataka

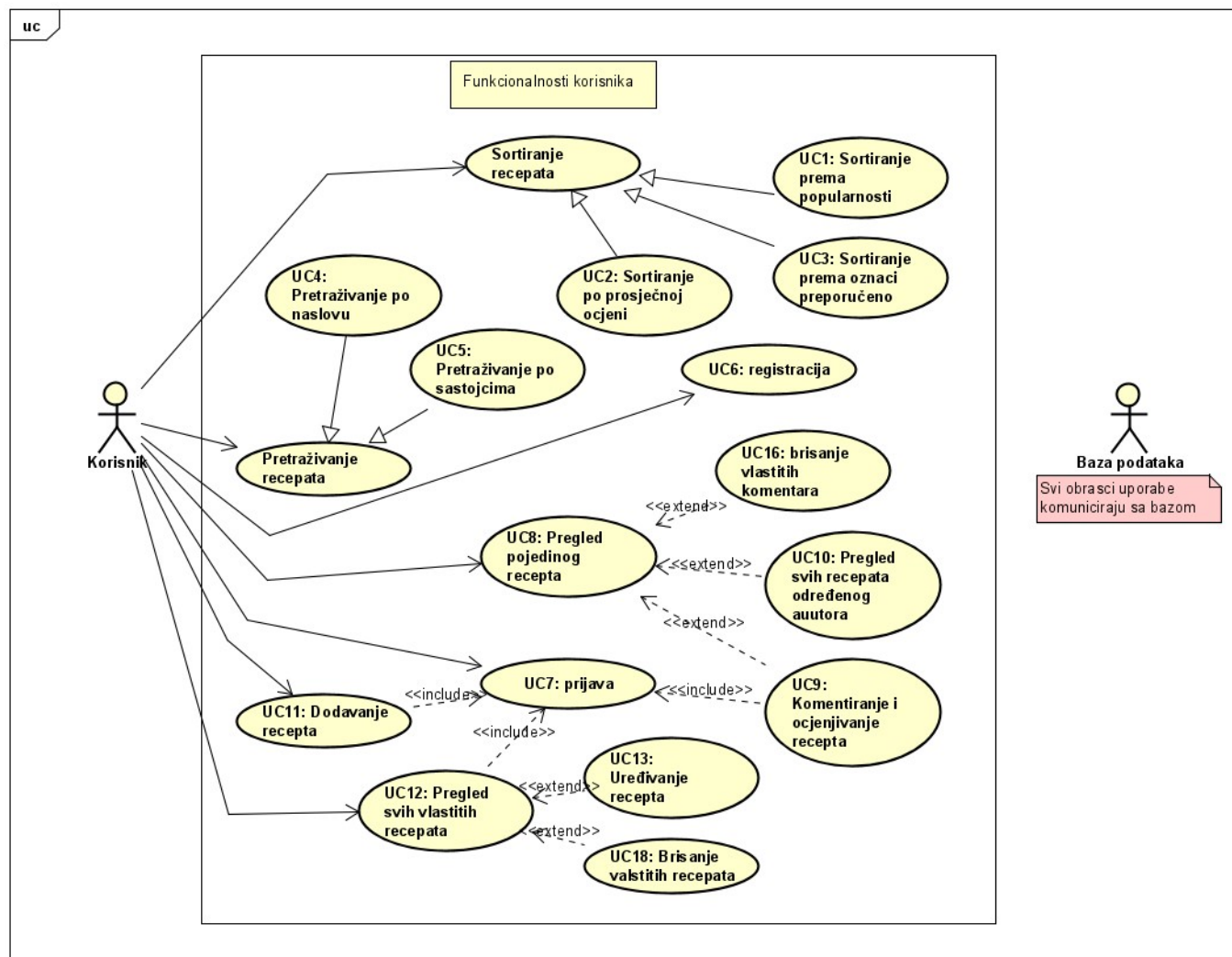
UC19 - Pregled korisnika sustava

- **Glavni sudionik:** Moderator
- **Cilj:** Pregledati registrirane korisnike
- **Sudionici:** Baza podataka
- **Preduvjet:** Moderator mora biti prijavljen
- **Opis osnovnog tijeka:**
 1. Moderator odabire opciju pregledavanja korisnika
 2. Prikaže se lista svih ispravno registriranih korisnika s osobnim podacima

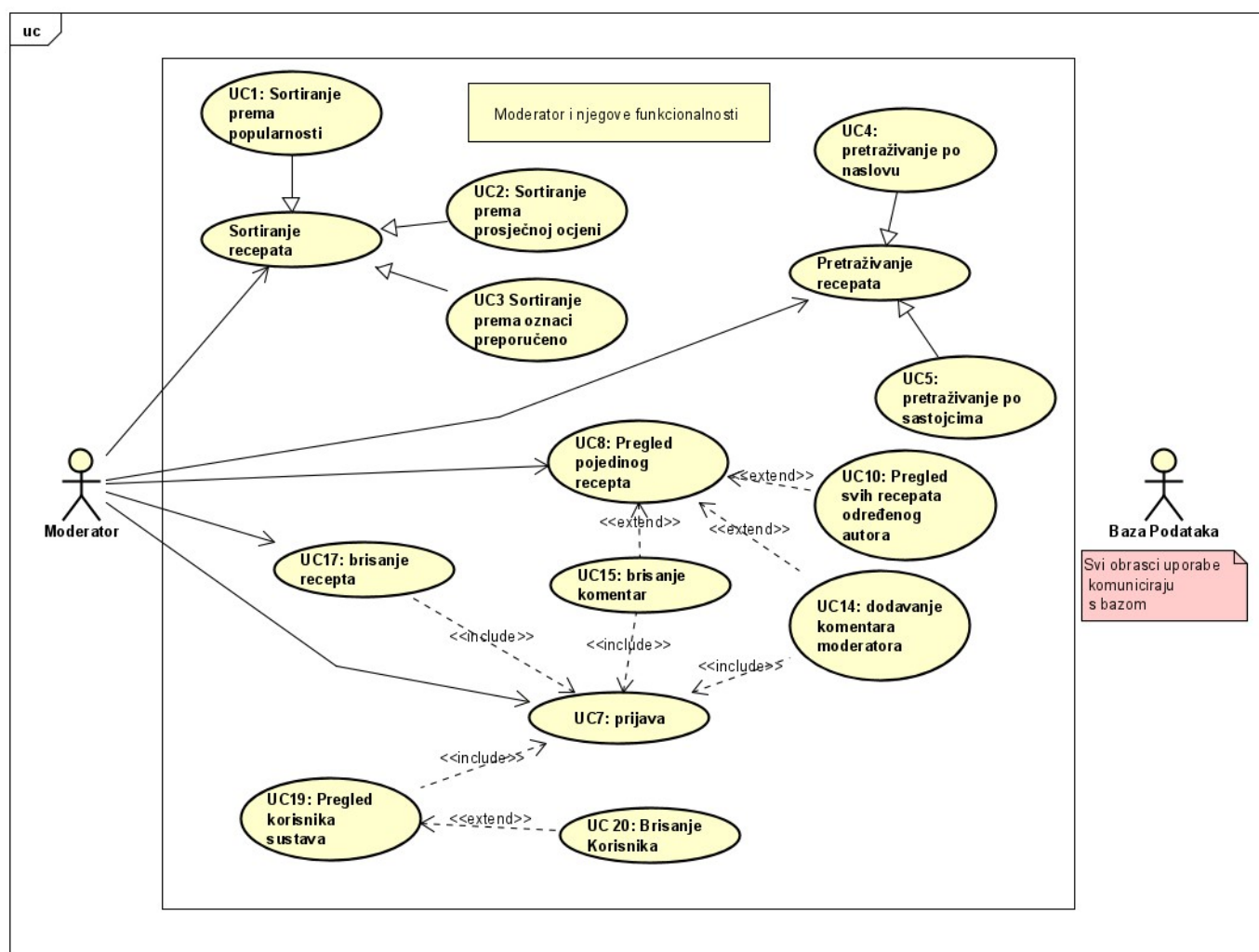
UC20 - Brisanje korisnika

- **Glavni sudionik:** Moderator
- **Cilj:** Obrisati korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Moderator mora biti prijavljen
- **Opis osnovnog tijeka:**
 1. Moderator odabire opciju uklanjanja korisnika
 2. Moderator pronalazi željenog korisnika
 3. Moderator uklanja željenog korisnika i njegove podatke iz baze podataka

Dijagrami obrazaca uporabe



Slika 3.1: Dijagram obrasca uporabe, funkcionalnost korisnika

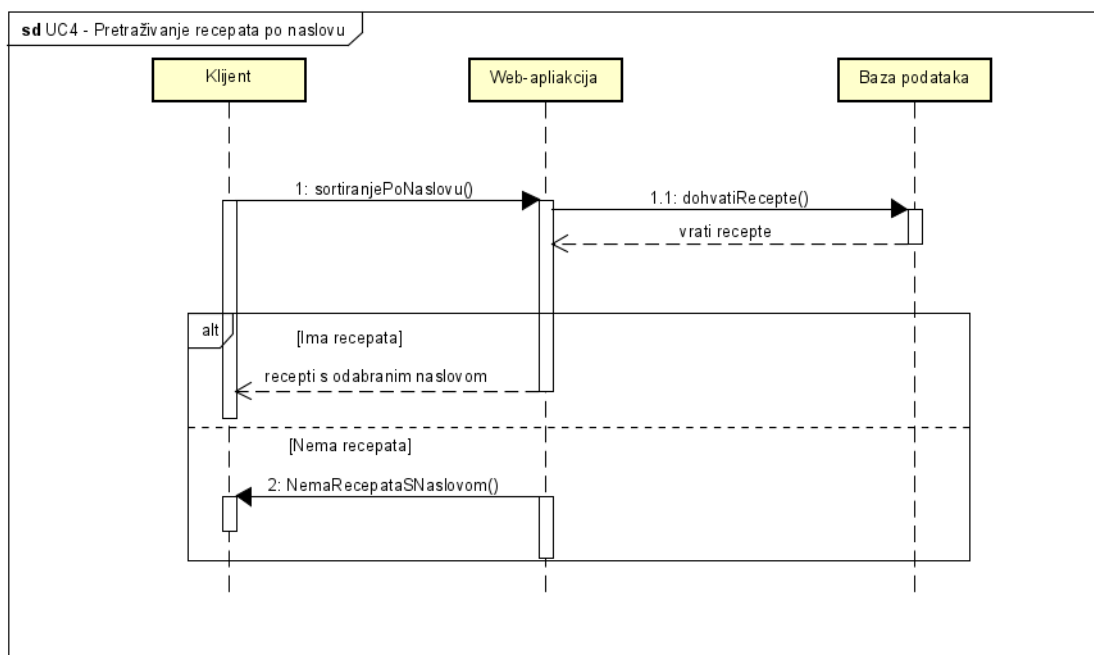


Slika 3.2: Dijagram obrasca uporabe, funkcionalnost moderatora

3.1.2 Sekvencijski dijagrami

Obrazac uporabe UC4 - Pretraživanje recepata po naslovu

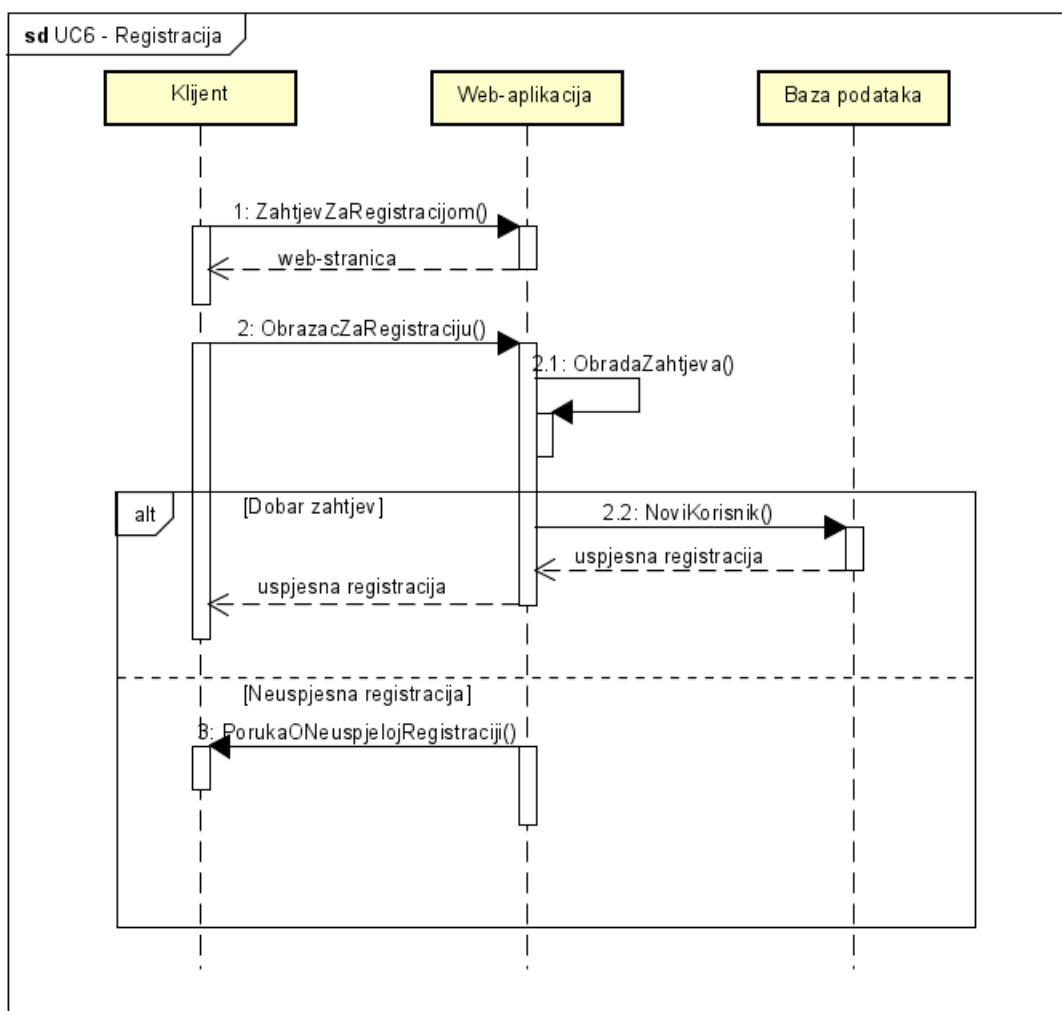
Klijent na početnoj stranici ima uvid u sve recepte, ali ima i mogućnost filtriranja tih recepata. Jedno od mogućnosti filtriranja jest filtriranje recepata po naslovu. Klijent upisuje naslov ili dio naslova kojim želi naći određeni recept te pritisće tipku pretraži. Pritiskom na tipku, web-apliakcija šalje upit bazi podataka. Baza podataka pretražuje sve recepte i traži one recepte koji imaju sve riječi ili samo jednu riječ upisanu u tražilicu. Ukoliko baza podataka ne pronade niti jedan recept nakon pretraživanja, klijent će biti obaviješten o tome. U suprotnome, baza podataka vraća web-aplikaciji sve recepte koje je pronašla nakon pretraživanja. Web-apliakcija potom ispisuje sve recepte klijentu te se time završava pretraživanje.



Slika 3.3: Sekvencijski dijagram za UC4

Obrazac uporabe UC6 - Registracija

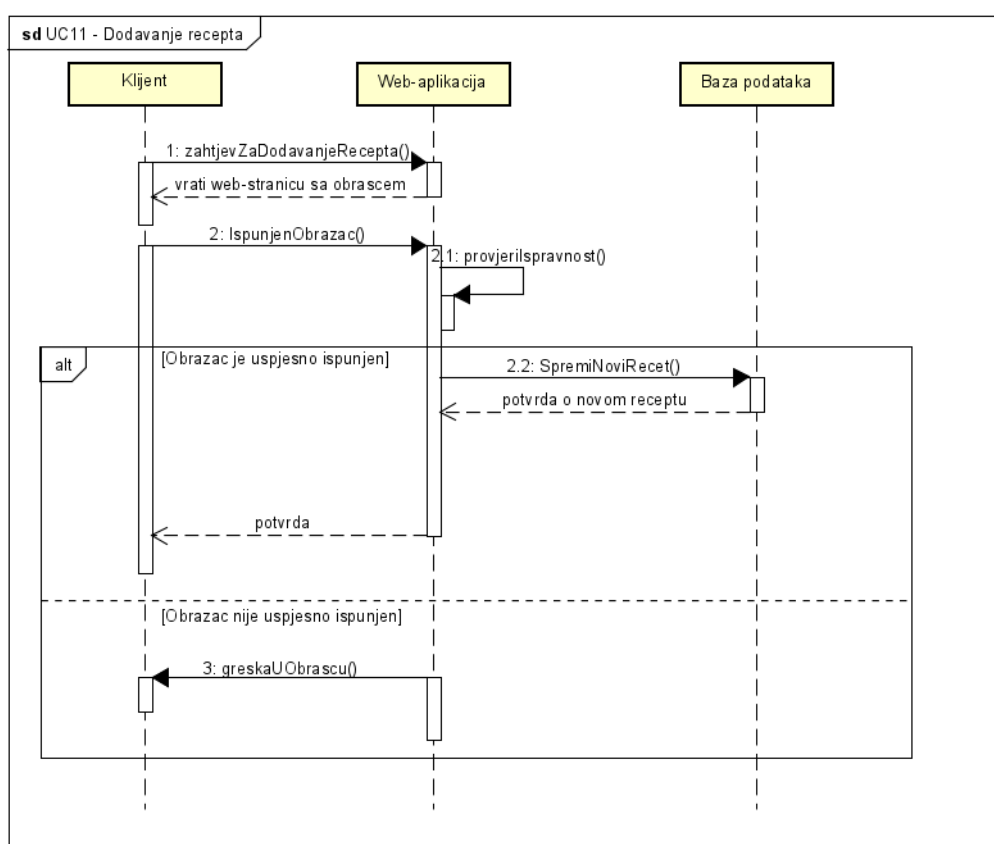
Klijent ima mogućnost registracije. Registracija je postupak stvaranja računa po prvi puta u web-aplikaciji. Klijent pritiskom na gumb "prijava" otvara novu web-stranicu gdje mu je omogućena prijava. Osim prijave, klijentu je omogućena i registracija, ukoliko nema već postojeći račun. Klikom na registraciju, pojavljuje se obrazac za stvaranje računa gdje je potrebno unijeti podatke koji će se potom spremati u bazu podataka. Kako bi obrazac bio uspješno napisan potrebno je: ispuniti sva polja, upisati nepostojeće korisničko ime i lozinku koja se pridržava propisanih pravila. Ako je klijent uspješno ispunio zahtjev, njegov račun će se spremiti u bazu podataka i dobit će obavijest o uspješnoj registraciji. Ako klijent nije uspješno ispunio zahtjev, dobit će poruku o odgovarajućoj pogrešci.



Slika 3.4: Sekvencijski dijagram za UC6

Obrazac uporabe UC11 - Dodavanje recepta

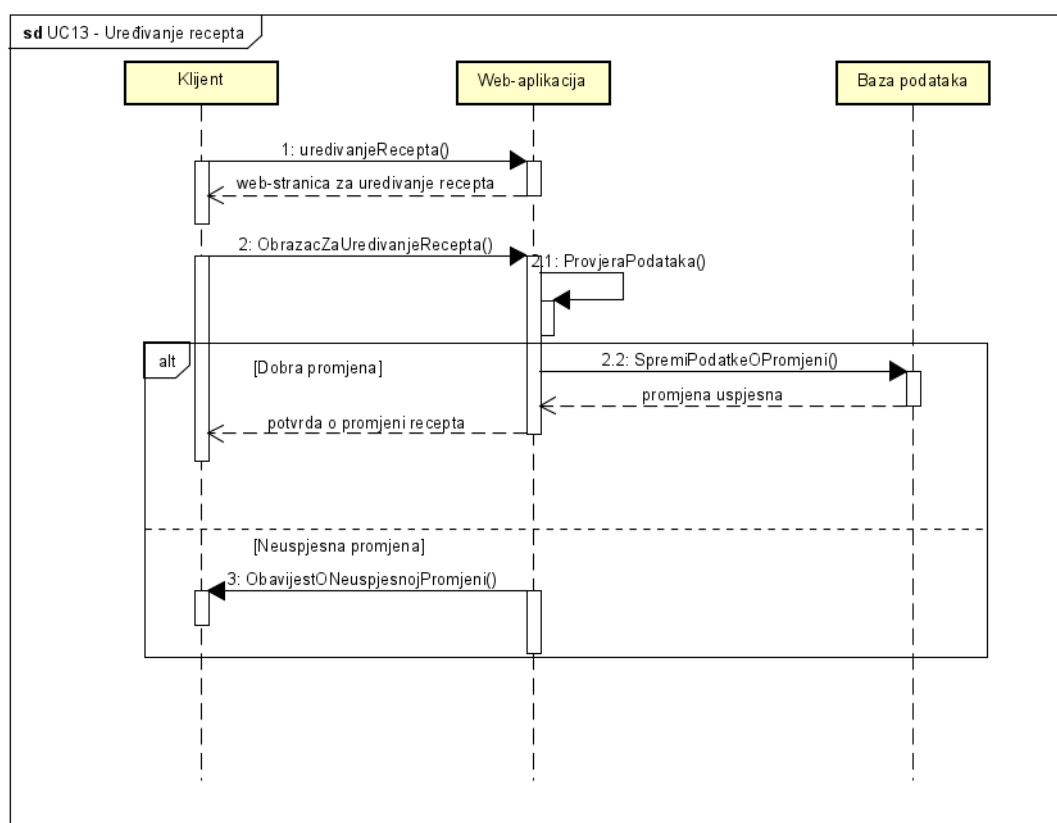
Registrirani korisnik (nadalje klijent) ima mogućnost dodavanja recepta. Dodavanje recepta je vidljivo na profilu klijenta. Odabirom usluge dodavanja recepta, otvara se web-stranica sa obrascem za dodavanje recepta. U obrascu je potrebno popuniti označena polja, te priložiti do maksimalno 5 slika. Ako je klijent uspješno popunio obrazac, recept se dodaje u bazu podataka i klijent dobiva obavijest o uspješnome dodavanju recepta. Ako klijent nije uspješno popunio obrazac, ispisuje mu se odgovarajuća pogreška, ali se postojeći podaci ne brišu te klijent ima mogućnost ispravljanja pogreške.



Slika 3.5: Sekvencijski dijagram za UC11

Obrazac uporabe UC13 - Uređivanje recepta

Registrirani korisnik (nadalje klijent) ima mogućnost uređivanja vlastitog recepta. Uređivanje recepta je vidljivo na stranici klijentovog recepta. Odabirom usluge uređivanja recepta, otvara se web-stranica s već postojećim obrascem. Podaci u već postojećem obrascu izvučeni su iz baze podataka. Klijent može dodavati ili brisati podatke u obrascu. Klijent mora poštovati pravila za uređivanje obrasca kao i kod dodavanja recepta. Ako je klijent napravio prihvatljivu promjenu, promjena će se spremiti u bazu podataka i klijent dobiva obavijest o uspješnome uređivanju recepta. Ako je klijent napravio neprihvatljivu promjenu, promjena neće biti spremljena i klijent će dobiti obavijest o pogrešci.



Slika 3.6: Sekvencijski dijagram za UC13

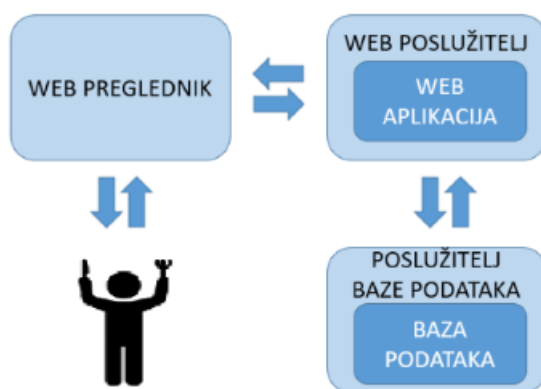
3.2 Ostali zahtjevi

- *Sustav treba omogućiti rad više korisnika u stvarnom vremenu*
- *Korisničko sučelje i sustav moraju podržavati hrvatsku abecedu (dijakritičke znakove) pri unosu i prikazu tekstualnog sadržaja*
- *Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi*
- *Sustav treba biti implementiran kao web aplikacija koristeći objektno-orijentirane jezike*
- *Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava*
- *Sustav treba biti jednostavan za korištenje, korisnici se moraju znati koristiti sučeljem bez opširnih uputa*
- *Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava*
- *Veza s bazom podataka mora biti kvalitetno zaštićena, brza i otporna na vanjske greške*
- *Pristup sustavu mora biti omogućen iz javne mreže pomoću HTTPS.*

4. Arhitektura i dizajn sustava

Arhitektura se može podijeliti na tri podsustava:

- Web poslužitelj
- Web aplikacija
- Baza podataka



Slika 4.1: Arhitektura sustava

Web preglednik je program koji korisniku omogućuje pregled web-stranica i multimedijalnih sadržaja vezanih uz njih. Svaki internetski preglednik je prevođitelj. Dakle, stranica je pisana u kodu koji preglednik nakon toga interpretira kao nešto svakome razumljivo. Korisnik putem web preglednika šalje zahtjev web poslužitelju.

Web poslužitelj osnova je rada web aplikacije. Njegova primarna zadaća je komunikacija klijenta s aplikacijom. Komunikacija se odvija preko HTTP (engl. Hyper Text Transfer Protocol) protokola, što je protokol u prijenosu informacija na webu. Poslužitelj je onaj koji pokreće web aplikaciju te joj prosljeđuje zahtjev.

Korisnik koristi web aplikaciju za obrađivanje željenih zahtjeva. Web aplikacija obrađuje zahtjev te ovisno o zahtjevu, pristupa bazi podataka nakon čega preko poslužitelja vraća korisniku odgovor u obliku HTML dokumenta vidljivog u web pregledniku.

Programski jezik kojeg smo odabrali za izradu naše web aplikacije je Java zajedno sa Spring Boot radnim okvirom. Odabrano razvojno okruženje je Visual Studio Code. Arhitektura sustava temeljiti će se na MVC (Model-View-Controller) konceptu.

Karakteristika MVC koncepta je nezavisan razvoj pojedinih dijelova aplikacije što za posljedicu ima jednostavnije ispitivanje kao i jednostavno razvijanje i dodavanje novih svojstava u sustav.

MVC koncept sastoji se od:

- **Model** - Središnja komponenta sustava. Predstavlja dinamičke strukture podataka, neovisne o korisničkom sučelju. Izravno upravlja podacima, logikom i pravilima aplikacije. Također prima ulazne podatke od Controllera.
- **View** - Bilo kakav prikaz podataka, poput grafa. Mogući su različiti prikazi iste informacije poput grafičkog ili tabličnog prikaza podataka.
- **Controller** - Prima ulaze i prilagođava ih za prosljeđivanje Modelu ili Viewu. Upravlja korisničkim zahtjevima i temeljem njih izvodi daljnju interakciju s ostalim elementima sustava.

4.1 Baza podataka

Za potrebe našeg sustava koristit ćemo relacijsku bazu podataka koja svojom strukturom olakšava modeliranje stvarnog svijeta. Gradivna jedinka baze je relacija, odnosno tablica koja je definirana svojim imenom i skupom atributa. Zadaća baze podataka je brza i jednostavna pohrana, izmjena i dohvat podataka za daljnju obradu. Baza podataka ove aplikacije sastoji se od sljedećih entiteta:

- Users
- Roles
- Images
- Ingredients
- Recipes
- Ratings
- Recipe_steps

4.1.1 Opis tablica

Users Ovaj entitet sadržava sve važne informacije o korisniku aplikacije. Sadrži attribute: `first_name`, `last_name`, `date_of_birth`, `email`, `id`, `password.hash`, `username` `role_id`. Ovaj entitet u vezi je *Many-to-One* s entitetom Roles preko atributa `role_id`, u vezi *One-to-Many* s entitetom Recipes preko atributa `created_by` te je u vezi *One-to-Many* s entitetom Ratings preko atributa `user_id`.

| Users | | |
|----------------------------|---------|-------------------------------------|
| <code>id</code> | SERIAL | jedinstveni identifikator korisnika |
| <code>first_name</code> | VARCHAR | ime korisnika |
| <code>last_name</code> | VARCHAR | prezime korisnika |
| <code>date_of_birth</code> | DATE | datum rođenja korisnika |
| <code>email</code> | VARCHAR | korisnikov email |
| <code>password.hash</code> | VARCHAR | korisnikova lozinka |
| <code>username</code> | VARCHAR | korisničko ime |
| <code>role_id</code> | INT | strani ključ na entitet Roles |

Recipes Ovaj entitet sadržava sve važne informacije o receptu unutar aplikacije. Sadrži attribute: `id`, `popularity`, `title`, `created_at`, `last_updated_at`, `recipe_description`, `estimated_time` i `created_by`. Ovaj entitet u vezi je *One-to-Many* s entitetom `Ingredients` preko atributa `recipe_id`, u vezi *One-to-Many* s entitetom `Images` preko atributa `recipe_id`, u vezi *Many-to-One* s entitetom `Users` preko atributa `created_by`, u vezi *One-to-Many* s entitetom `Ratings` preko atributa `recipe_id` te je u vezi *One-to-Many* s entitetom `Recipe_steps` preko atributa `recipe_id`.

| Recipes | | |
|---------------------------------|-----------|--|
| <code>id</code> | SERIAL | jedinstveni identifikator recepta |
| <code>popularity</code> | INT | popularnost recepta |
| <code>title</code> | VARCHAR | naslov recepta |
| <code>created_at</code> | TIMESTAMP | datum nastanka recepta |
| <code>last_updated_at</code> | TIMESTAMP | datum zadnje izmjene recepta |
| <code>recipe_description</code> | VARCHAR | kratki opis recepta |
| <code>estimated_time</code> | INT | procijenjeno vrijeme za pripremu |
| <code>created_by</code> | INT | strani ključ na entitet <code>Users</code> |

Ingredients Ovaj entitet predstavlja jedan sastojak unutar nekog recepta. Sadrži attribute: `id`, `ingredient_name`, `ingredient_measure`, `ingredient_quantity` i `ingredient_order`. Ovaj entitet u vezi je *Many-to-One* s entitetom `Recipes` preko atributa `recipe_id`.

| Ingredients | | |
|----------------------------------|---------|------------------------------------|
| <code>id</code> | SERIAL | jedinstveni identifikator sastojka |
| <code>ingredient_name</code> | VARCHAR | ime sastojka |
| <code>ingredient_quantity</code> | INT | količina sastojka |
| <code>ingredient_measure</code> | VARCHAR | ime mjere sastojka(ml, g, kg) |
| <code>ingredient_order</code> | INT | redni broj sastojka unutar recepta |

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

| Ingredients | | |
|-------------|-----|---------------------------------|
| recipe_id | INT | strani ključ na entitet Recipes |

Ratings Ovaj entitet predstavlja ocjenu na određeni recept. Sadrži attribute: id, rating_value, user_id i recipe_id. Ovaj entitet u vezi je *Many-to-One* s entitetom Recipes preko atributa recipe_id te je u vezi *Many-to-One* s entitetom Users preko atributa user_id

| Ratings | | |
|--------------|--------|----------------------------------|
| id | SERIAL | jedinstveni identifikator ocjene |
| rating_value | INT | vrijednost ocjene |
| user_id | INT | strani ključ na entitet Users |
| recipe_id | INT | strani ključ na entitet Recipes |

Images Ovaj entitet predstavlja jednu sliku koja se dodaje prilikom dodavanja recepta. Sadrži attribute: id, image_data, image_order i recipe_id. Ovaj entitet u vezi je *Many-to-One* s entitetom Recipes preko atributa recipe_id.

| Images | | |
|-------------|--------|---------------------------------------|
| id | SERIAL | jedinstveni identifikator slike |
| image_data | BYTEA | prostor koji zauzima slika u memoriji |
| image_order | INT | poredak slike prilikom dodavanja |
| recipe_id | INT | strani ključ na entitet Recipes |

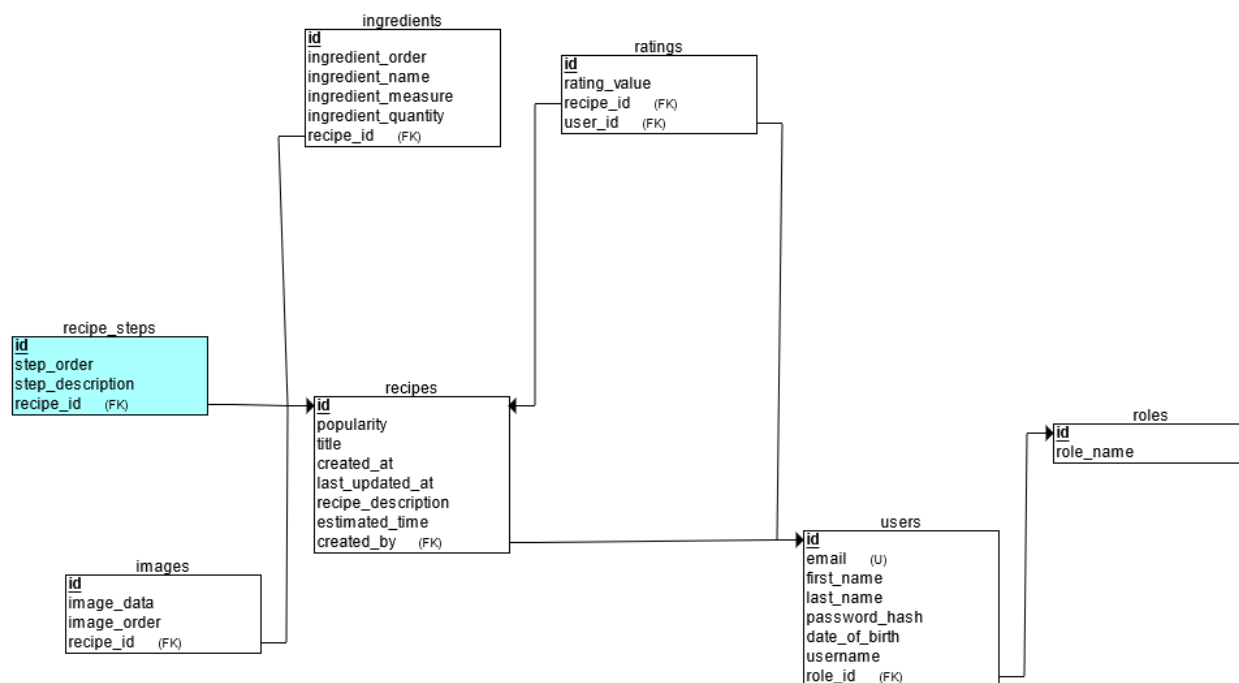
Recipe_steps Ovaj entitet predstavlja jedan korak pripreme u receptu. Sadrži attribute: id, step_order, step_description i recipe_id. Ovaj entitet u vezi je *Many-to-One* s entitetom Recipes preko atributa recipe_id.

| Recipe_steps | | |
|------------------|---------|---|
| id | SERIAL | jedinstveni identifikator koraka pripreme |
| step_order | INT | redni broj koraka pripreme |
| step_description | VARCHAR | kratki opis koraka pripreme |
| recipe_id | INT | strani ključ na entitet Recipes |

Roles Ovaj entitet predstavlja ulogu korisnika u sustavu. Sadrži attribute: id i role_name. Ovaj entitet u vezi je *One-to-Many* s entitetom Users preko atributa role_id.

| Roles | | |
|-----------|---------|---------------------------------|
| id | SERIAL | jedinstveni identifikator uloge |
| role_name | VARCHAR | ime uloge (korisnik, moderator) |

4.1.2 Dijagram baze podataka

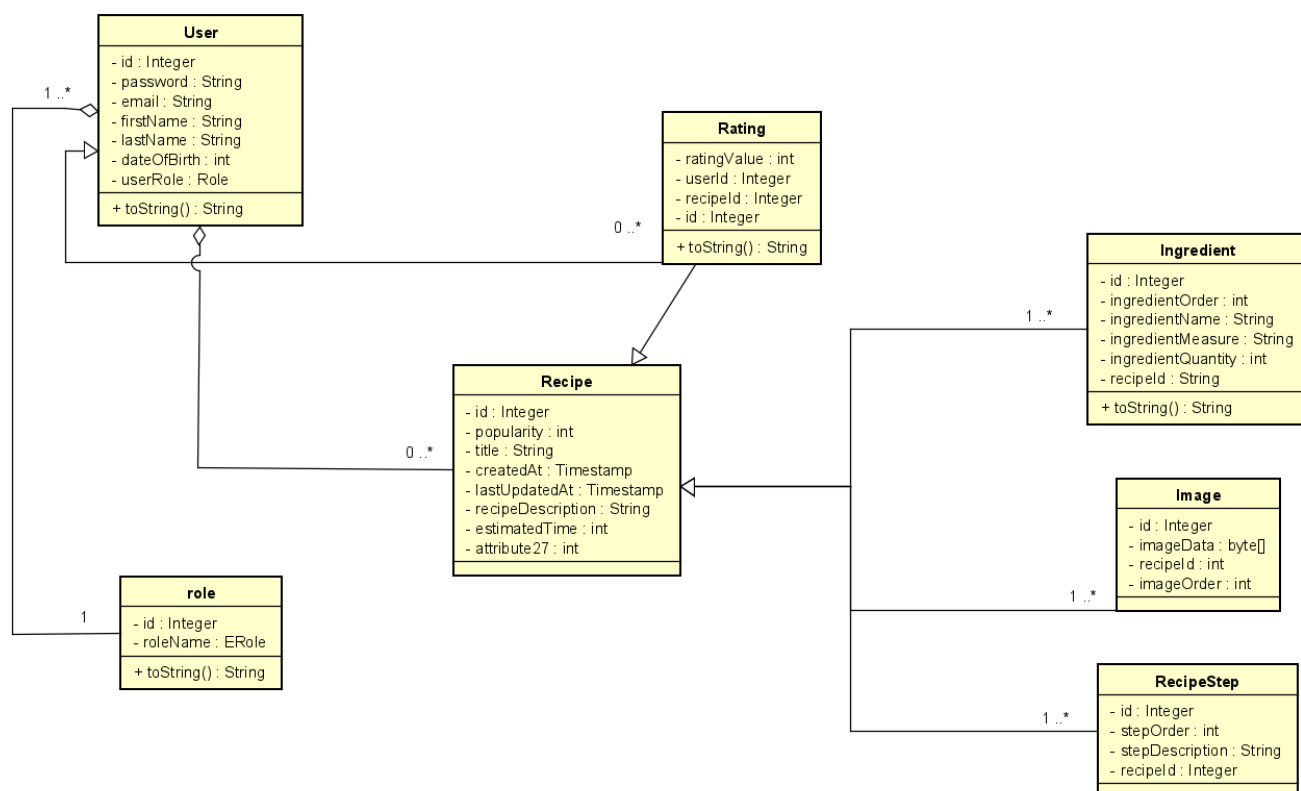


Slika 4.2: E-R dijagram baze podataka

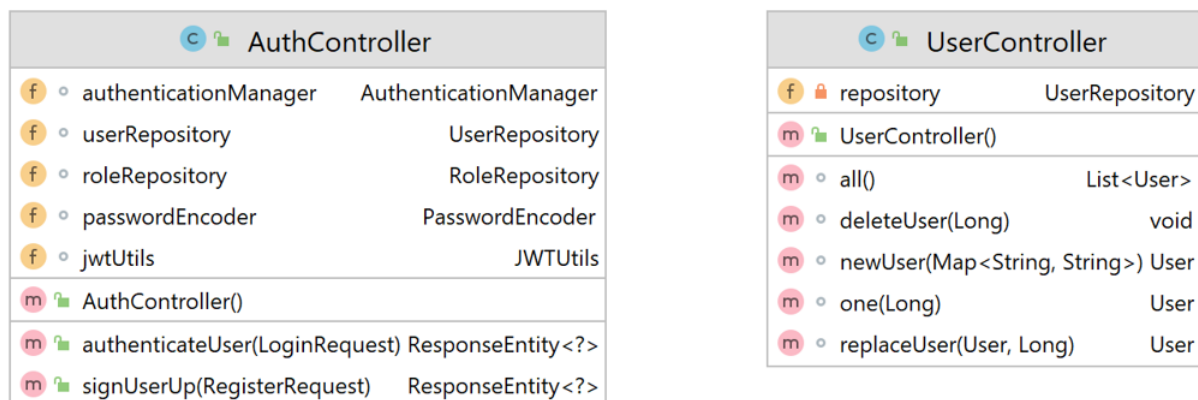
4.2 Dijagram razreda

Na slikama 4.3, 4.4 i 4.5 prikazani su razredi koji pripadaju backend dijelu MVC arhitekture. Razredi prikazani na 4.3 su Model razredi koji su implementacije entiteta baze. Razredi prikazani na slici 4.3 su Controller razredi te oni implementiraju funkcionalnosti u ovisnosti na HTTP upit i vraćaju valjani odgovor u JSON formatu. Bazi se pristupa preko Interfacea koji nasljeđuju JpaRepository.

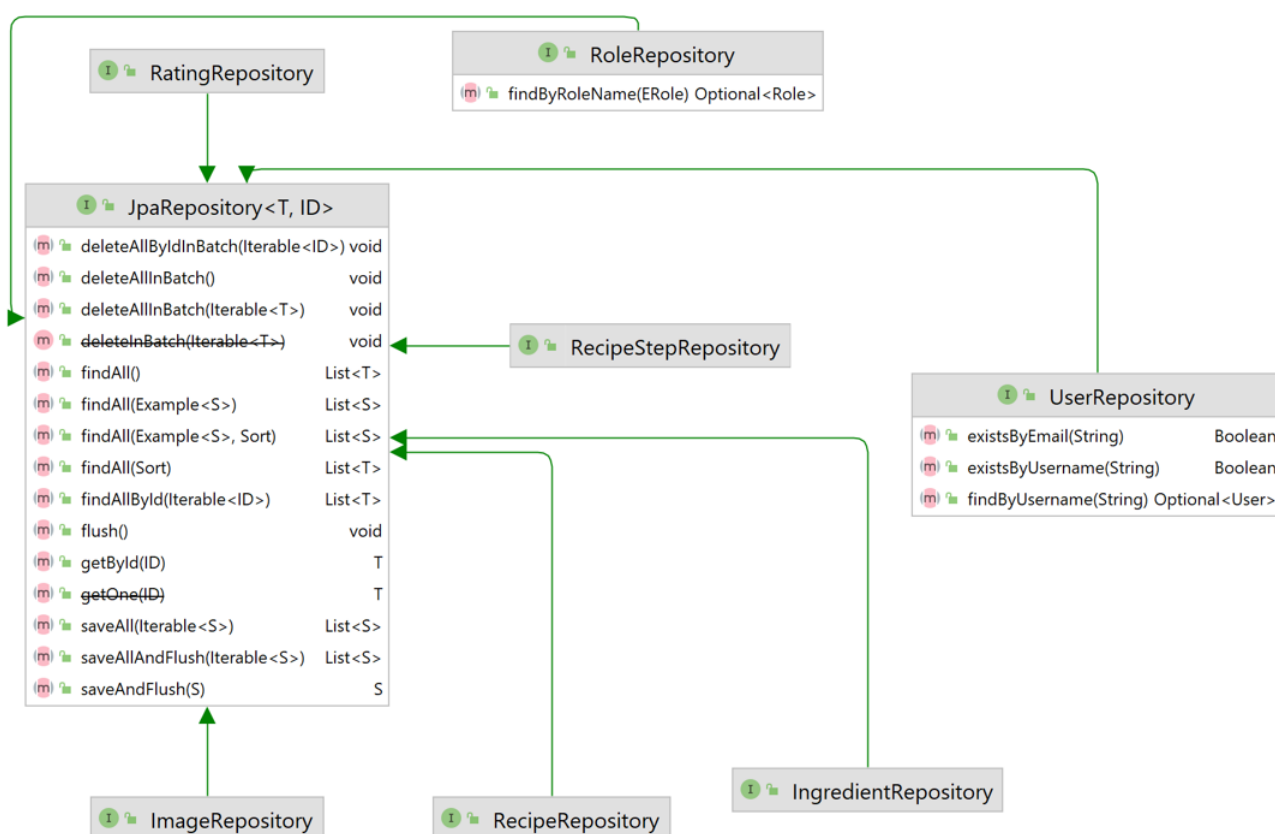
Zbog lakše organizacije razredi su podijeljeni na tri logičke jedinice te su prikazani odvojenim dijagramima. Iz naziva i tipova atributa mogu se zaključiti ostale ovisnosti među razredima.



Slika 4.3: Dijagram razreda - Models



Slika 4.4: Dijagram razreda - Controllers



Slika 4.5: Dijagram razreda - Repositories

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 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

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 | Prikaz svih recepata | 6 |
| 2.2 | Primjer otvorenog recepta | 7 |
| 2.3 | Komentari na recept | 8 |
| 2.4 | Prijava i registracija u sustav | 8 |
| 3.1 | Dijagram obrasca uporabe, funkcionalnost korisnika | 19 |
| 3.2 | Dijagram obrasca uporabe, funkcionalnost moderatora | 20 |
| 3.3 | Sekvencijski dijagram za UC4 | 21 |
| 3.4 | Sekvencijski dijagram za UC6 | 22 |
| 3.5 | Sekvencijski dijagram za UC11 | 23 |
| 3.6 | Sekvencijski dijagram za UC13 | 24 |
| 4.1 | Arhitektura sustava | 26 |
| 4.2 | E-R dijagram baze podataka | 32 |
| 4.3 | Dijagram razreda - Models | 33 |
| 4.4 | Dijagram razreda - Controllers | 34 |
| 4.5 | Dijagram razreda - Repositories | 34 |

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 4. listopada 2021.
- Prisustvovali: K.Frankola, B.Colarić, D.Dugonjevac, T.Serezlija, M.Tunjić
- Teme sastanka:
 - Međusobno upoznavanje i predstavljanje
 - Rasprava o projektu i zajedničkim aktivnostima (u grubo)
 - Prijava na stranicu raspodjele grupa

2. sastanak

- Datum: 11. listopada 2021.
- Prisustvovali: K.Frankola, J.Brkić, B.Colarić, D.Dugonjevac, P.Gavran, T.Serezlija, M.Tunjić
- Teme sastanka:
 - Upoznavanje novih članova
 - Ponovna rasprava o odabranim tehnologijama
 - Podjela aktivnosti po članovima

3. sastanak

- Datum: 27. listopada 2021.
- Prisustvovali: K.Frankola, J.Brkić, B.Colarić, D.Dugonjevac, P.Gavran, T.Serezlija, M.Tunjić
- Teme sastanka:
 - Daljnja podjela aktivnosti (dokumentacija i programiranje)
 - Razmatranje arhitekture i ponašanja sustava

4. sastanak

- Datum: 6. studenoga 2021.
- Prisustvovali: K.Frankola, J.Brkić, B.Colarić, D.Dugonjevac, P.Gavran, T.Serezlija, M.Tunjić
- Teme sastanka:

- Osnovni dijagram baze
- Osnovni dijagram ponašanja sustava
- Aktivnosti koje je potrebno napraviti do predaje

Tablica aktivnosti

Kontinuirano osvježavanje

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

| | Karlo Frankola | Jan Brkić | Borna Colarić | Dario Dugonjevac | Petar Gavran | Toni Serezlija | Marko Tunjić |
|----------------------------------|----------------|-----------|---------------|------------------|--------------|----------------|--------------|
| Upravljanje projektom | 4 | | | | | | |
| Opis projektnog zadatka | | | 1 | 2 | | | 2 |
| Funkcionalni zahtjevi | | | 3 | | | | 2 |
| Opis pojedinih obrazaca | | | | | | 3 | |
| Dijagram obrazaca | | | | | | | 1 |
| Sekvencijski dijagrami | | | | 2 | | | |
| Opis ostalih zahtjeva | | | | | | 1 | |
| Arhitektura i dizajn sustava | | | | | | 2 | |
| Baza podataka | | | | | | 2 | |
| Dijagram razreda | | 1.5 | | | | | |
| Dijagram stanja | | | | | | | |
| Dijagram aktivnosti | | | | | | | |
| Dijagram komponenti | | | | | | | |
| Korištene tehnologije i alati | | | | | | | |
| Ispitivanje programskog rješenja | | | | | | | |
| Dijagram razmještaja | | | | | | | |

Nastavljeno na idućoj stranici

Nastavljeno od prethodne stranice

| | Karlo Frankola | Jan Brkić | Borna Colarić | Dario Dugonjevac | Petar Gavran | Toni Serezlija | Marko Tunjić |
|----------------------------------|----------------|-----------|---------------|------------------|--------------|----------------|--------------|
| Upute za puštanje u pogon | | | | | | | |
| Dnevnik sastajanja | | | | | | | |
| Zaključak i budući rad | | | | | | | |
| Popis literature | | | | | | | |
| <i>Izrada početne stranice</i> | 4 | 1 | | | | | |
| <i>Izrada baze podataka</i> | 1 | | 1 | 1 | | 1 | 1 |
| <i>Spajanje s bazom podataka</i> | 2 | | | | | | |
| <i>Back End</i> | 6 | | 4 | 3 | | 4 | 4 |
| <i>Front End</i> | 2 | 5 | | 2 | | 2 | 3 |
| <i>Puštanje u pogon</i> | 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.