

Programsko inženjerstvo

Ak. god. 2020./2021.

Maketa Shop

Dokumentacija, Rev. 1

Grupa: *FERocious*

Voditelj: *Maksimilijan Marošević*

Datum predaje: *14.01.2021*

Nastavnik: *Vlado Sruk*

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	5
3 Specifikacija programske potpore	8
3.1 Funkcionalni zahtjevi	8
3.1.1 Obrasci uporabe	10
3.1.2 Sekvencijski dijagrami	22
3.2 Ostali zahtjevi	25
4 Arhitektura i dizajn sustava	26
4.1 Baza podataka	27
4.1.1 Opis tablica	28
4.1.2 Dijagram baze podataka	33
4.2 Dijagram razreda	34
4.3 Dijagram stanja	35
4.4 Dijagram aktivnosti	37
4.5 Dijagram komponenti	39
5 Implementacija i korisničko sučelje	40
5.1 Korištene tehnologije i alati	40
5.2 Ispitivanje programskog rješenja	41
5.2.1 Ispitivanje komponenti	41
5.2.2 Ispitivanje sustava	41
5.3 Dijagram razmještaja	47
5.4 Upute za puštanje u pogon	48
6 Zaključak i budući rad	54
Popis literature	55
Indeks slika i dijagrama	57

Dodatak: Prikaz aktivnosti grupe

58

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Dodani dionici i aktori i opisano prvih 8 <i>Use Case</i> dijagrama	Jukanović	09.11.2020.
0.2	Dodan dijagram obrasca uporabe (funkcionalnost prijavljenog korisnika)	Jukanović	10.11.2020
0.3	Dodano još 7 (09-15) <i>Use Case</i> dijagrama	Brstilo	10.11.2020.
0.4	Modificirani zahtjevi aktora i dodani (16-25) <i>Use case</i> dijagrama	Antunović	10.11.2020.
0.5	Dodan dijagram obrasca uporabe (funkcionalnost administratora)	Antunović	11.11.2020.
0.6	Dodan dijagram razreda generičke funkcionalnosti	Cigula	12.11.2020.
0.7	Dodani sekvencijski dijagrami	Marošević	13.11.2020.
0.8	Dodana arhitektura baze podataka	Piškur, Marošević	13.11.2020.
0.9	Dodani arhitektura sustava, opis projektnog zadatka i ostali zahtjevi	Rončević	13.11.2020.
1.0	Korigiranje teksta i provjera dokumentacije	Marošević, Cigula, Jukanović	13.11.2020
1.1	Dodan dijagram razmještaja	Jukanović	11.01.2021
1.2	Dodani dijagrami aktivnosti	Antunović	13.01.2021.
1.3	Dodan dijagram stanja	Jukanović	13.01.2021.
1.4	Modificirani modeli baze podataka	Antunović	14.01.2021
1.5	Dodan dijagram razreda	Cigula	14.01.2021
1.6	Dodane slike koda pod ispitivanje komponenti	Cigula, Brstilo	14.01.2021
1.7	Ažurirani sati rada i dnevnik sastajanja	Cigula	14.01.2021
1.8	Dodana uputstva za puštanje u pogon	Marošević	14.01.2021.

Rev.	Opis promjene/dodatka	Autori	Datum
1.9	Dodani zaključak, korištene tehnologije i ispitivanje sustava	Rončević	14.01.2021.
1.10	Dodan dijagram komponenti	Piškur	14.01.2021.
2.0	Grafovi kontribucije i manje korekcije	Marošević	14.01.2021.

2. Opis projektnog zadatka

Cilj ovog projekta je razviti programsku podršku za stvaranje web aplikacije "Maketa shop" koja se sastoji od dva glavna dijela. Prvi dio omogućuje objavljivanje sadržaja vezanih uz izradu maketa. Sadržaj je u obliku priča koje se sastoje od slika, videa, teksta ili kombinacija svega navedenog koji prate proces izrade maketa. Same priče objavljuje administrator dok registrirani korisnici iste mogu predlagati. Postoji mogućnost dodavanja komentara na priče, koje mogu ostavljati registrirani i neregistrirani korisnici.

Drugi dio je *webshop* na kojemu korisnici mogu naručivati makete. Sam *shop* je standardna web-trgovina sa svim funkcionalnostima koje korisnik očekuje od takve stranica. Razlika između "Maketa shopa" i drugih web-trgovina je u mogućnosti naručivanja maketa po vlastitim zamislima. Korisnik može maketaru (administratoru) poslati nacрте i opise makete kakvu želi, a maketar taj zahtjev može prihvatiti ili odbiti. Kada se obje strane slože oko cijene, proces se nastavlja na standardan način.

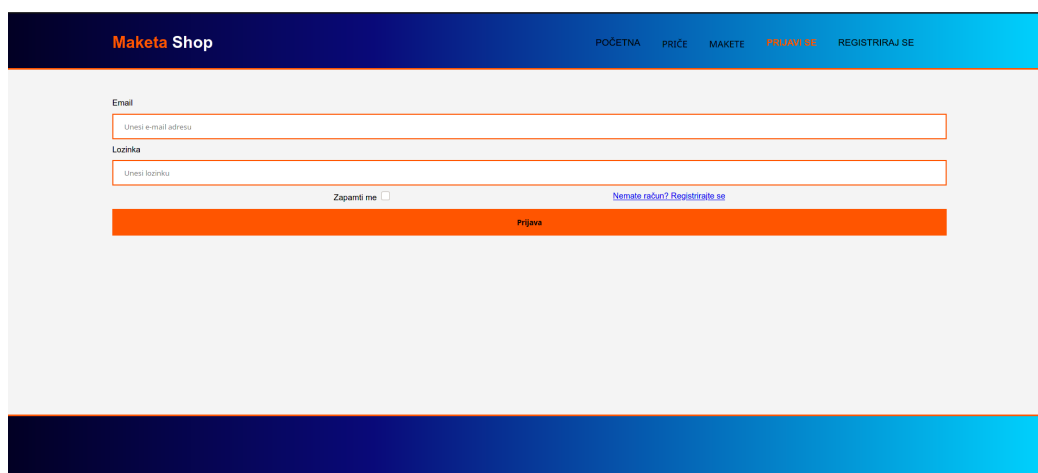


Slika 2.1: Početna stranica

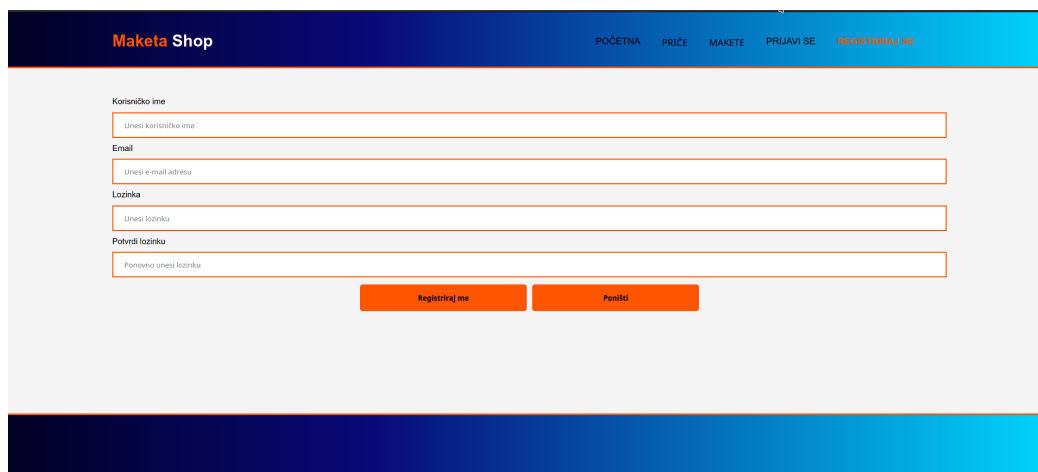
Aplikaciji mogu pristupiti registrirani i neregistrirani korisnici. Pri prvom ulasku u aplikaciju, svaki korisnik ima opciju Registracije, te prijavljivanja s postojećim računom. Aplikacija registriranim korisnicima pruža mogućnost spremanja informacija kako se ne bi trebali svaki put prijavljivati i unositi osobne podatke

prilikom plaćanja narudžbe. Podaci potrebni za kreiranje korisničkog računa su:

- korisničko ime (mora biti jedinstveno)
- e-mail adresa (mora biti jedinstvena)
- lozinka



Slika 2.2: Prijava u sustav



Slika 2.3: Registracija

Korisničko ime je javno svima dok za druge podatke korisnik određuje hoće li biti privatni ili javni.

Aplikacija pruža opciju kupnju standardne makete. Svaka maketa je detaljno opisana (dimenzija, materijal, boja) te se korisniku pruža mogućnost odabira nekoliko različitih materijala koji bi se koristili u izradi makete. Cijene će se razlikovati ovisno o biranim materijalima, te su one unaprijed određene od strane administratora. Također korisniku je omogućeno naručivanje makete prema vlastitim skicama. Atributi makete će se unositi u formular gdje se definira opis makete, dimenzija i materijal za izradu. Nakon što se formular ispuni administratoru se šalje obavijest, te ona ima mogućnost pregleda zahtjeva te određuje cijenu za tu maketu, koju korisnik može prihvatiti ili odbiti.

Plaćanje se 'odvija' pomoću formulara čijim se ispunjavanjem simulira online plaćanje. Podaci uneseni u formular se trebaju spremati, te se registriranim korisnicima nakon prvog plaćanja formular automatski puni. Povijest svih transakcija se treba spremati i pamtit i te mora biti dostupna administratoru.

Postoje 3 vrste korisnika u aplikaciji:

- Neregistrirani korisnik je osnovna uloga u web aplikaciji. Neregistrirani korisnik ima mogućnost kupovine maketa, međutim njegovi podaci se neće automatski popunjavati prilikom svake sljedeće kupovine, omogućeno mu je komentiranje na postojeće priče međutim nema mogućnosti predlaganja novih priča.
- Registrirani korisnik je uloga koja se ostvaruje prilikom registracije i prijave. Registrirani korisnik ima sve mogućnosti kao i neregistrirani korisnik međutim za njega postoji mogućnost automatskog ispunjavanja podataka prilikom plaćanja, te mogućnost predlaganja priča administratoru, također ima mogućnost naručivanja maketa prema vlastitim specifikacijama.
- Administrator je uloga koja se dodjeljuje jednoj osobi te on ima najveće ovlasti. Administrator je jedina osoba koja može objavljivati priče, samostalno ili na prijedlog registriranih korisnika. Također ima mogućnost pregleda proteklih transakcija. Administratoru je omogućena zabrana pristupa registriranim korisnicima.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Korisnik sustava
 - (a) Maketar
 - (b) Kupac
2. Administrator sustava
3. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (inicijator) može:
 - (a) registrirati se u sustav prilaganjem svoje e-mail adrese, lozinke i korisničkog imena
 - (b) pregledati priče i ostavljati komentare na njima
 - (c) unijeti podatke za kupovinu i kupiti proizvod
 - (d) pregledati makete i izabrati materijal makete
2. Korisnik (kupac/maketar) može:
 - (a) pregledati priče i ostavljati komentare na njima
 - (b) pregledati makete i izabrati materijal makete
 - (c) slati administratoru sustava prijedloge za nove priče
 - (d) slati administratoru sustava zahtjev za maketu po narudžbi
 - (e) prihvaćati ili odbijati ponudu cijene za maketu po narudžbi
 - (f) vidjeti podatke o svom korisničkom računu
 - (g) postavljati vidljivost svojih podataka
 - (h) kupiti proizvod
 - (i) odjaviti se
3. Administrator (inicijator) može:

- (a) prihvaćati ili odbijati prijedloge priča
- (b) prihvaćanjem objavljivati predložene priče
- (c) ponuditi cijenu za maketu po narudžbi
- (d) vidjeti podatke o izvršenim transakcijama
- (e) zabraniti pristup registriranim korisnicima
- (f) postavljati makete standardne ponude

4. Baza podataka (sudionik) može:

- (a) pohranjivati podatke o svim korisničkim računima sustava i njihovim ovlastima
- (b) pohranjivati priče i njihove dane opise
- (c) pohranjivati makete
- (d) pohranjivati podatke za plaćanje
- (e) pohranjivati zahtjeve za maketama
- (f) pohranjivati ponude cijena za makete
- (g) pohranjivati stanje vidljivosti podataka korisnika
- (h) pohranjivati podatke o izvršenim transakcijama
- (i) pohranjivati podatke o korisnicima sa zabranjenim pristupom
- (j) pohranjivati podatke maketa standardne ponude

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 – Registracija

- **Glavni sudionik:** Korisnik
- **Cilj:** Napraviti korisnički račun za korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Neregistrirani korisnik odabire opciju za registraciju
 2. Unosi potrebne podatke za registraciju
 3. Korisnikovi podaci se upisuju u bazu te je obaviješten o uspješnoj registraciji
- **Opis mogućih odstupanja:**
 - 2.a Odabir već zauzetog korisničkog imena i/ili e-maila, unos podataka u nedozvoljenom formatu (npr. lozinka) ili pružanje neispravnog e-maila
 1. Korisnik je obaviješten o nastaloj grešci i mjestu na kojem se ona nalazi
 2. Korisnik potom ispravlja grešku i ponovo šalje podatke ili odustaje od registracije

UC2 – Pregled priča

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisniku se pruža stranica sa kratkim prikazom svih priča u sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik sa početne stranice poveznicom dolazi do stranice sa pričama
 2. Aplikacija korisniku prikazuje sve priče u sustavu

UC3 – Pregled pojedine priče

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisniku se pruža detaljan prikaz određene priče
- **Sudionici:** Baza podataka
- **Preduvjet:** -

- **Opis osnovnog tijeka:**

1. Korisnik iz kratkog prikaza priča odabire priču koja ga zanima
2. Aplikacija korisniku pruža detaljan prikaz priče (slika, video, tekst) i mogućnost komentiranja na priču

UC4 – Komentiranje priče

- **Glavni sudionik:** Korisnik

- **Cilj:** Korisnik sustava ostavlja komentar na priču

- **Sudionici:** Baza podataka

- **Preduvjet:** -

- **Opis osnovnog tijeka:**

1. Korisnik otvara priču na kojoj želi ostaviti komentar
2. U za to namijenjeno mjesto upisuje komentar te ga objavljuje
3. Komentar se objavljuje te je uz njega navedeno ime korisnika koji ga je objavio

- **Opis mogućih odstupanja:**

- 3.a Korisnik nije prijavljen u sustav

1. Umjesto korisničkog imena na to mjesto se upisuje "Anoniman korisnik"

UC5 – Predlaganje priče

- **Glavni sudionik:** Korisnik

- **Cilj:** Korisnik šalje prijedlog priče za objavu na stranici administratoru

- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik je prijavljen u sustav

- **Opis osnovnog tijeka:**

1. Korisnik iz izbornika odabire opciju predlaganja priče
2. Aplikacija mu pruža sučelje za dodavanje opisa priče (slika, video, tekst)
3. Korisnik popunjava podatke i šalje ih administratoru na pregled

- **Opis mogućih odstupanja:**

- 2.a Korisnik nije predao dovoljno podataka (nema ni slike, ni teksta, ni videa i/ili nema naslova)

1. Korisnik je obaviješten o grešci i mjestu na kojem je nastala
2. Korisnik ispravlja grešku i šalje podatke administratoru na pregled ili odustaje od slanja prijedloga priče

UC6 – Prilaganje slike/teksta/videoa priči

- **Glavni sudionik:** Korisnik
- **Cilj:** Opisivanje priče koja se predlaže administratoru tekstom, slikom i videom
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisniku je pružen upravljač kojim dodaje proizvoljan broj elemenata teksta, slike i videa
 2. Korisnik u elemente upisuje tekst ili prilaže sliku ili video ili ih uklanja
 3. Nakon što je popunio podatke, može ih slati administratoru
- **Opis mogućih odstupanja:**
 - 2.a Korisnik prilaže sliku ili video nedozvoljenog formata
 1. Korisnik je obaviješten o grešci i dano mu je pojašnjenje te greške
 2. Korisnik odustaje od prilaganja slike/video uz priču ili prilaže sliku/video u drugom formatu

UC7 – Odobrenje/odbijanje priče

- **Glavni sudionik:** Administrator
- **Cilj:** Predložena priča se objavljuje na za to predviđenoj stranici ili se odbacuje
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen i dodana su mu prava administratora
- **Opis osnovnog tijeka:**
 1. Aplikacija administratoru pruža listu priča danih na prijedlog i njihov opis
 2. Administratoru je ispod svake priče ponuđena opcija odbacivanja ili prihvaćanja priče
 3. Priča se ovisno o administratorovom izboru uklanja iz sustava ili se objavljuje

UC8 – Objava priče

- **Glavni sudionik:** Administrator
- **Cilj:** Priča se nakon administratorovog odobravanja prikazuje svim korisnicima sustava na za to određenom mjestu
- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik je prijavljen u sustav i dodana su mu prava administratora
- **Opis osnovnog tijeka:**
 1. Administrator prihvaća priču
 2. Priča se pohranjuje u bazu podataka te se prikazuje na određenoj stranici zajedno sa ostalim pričama

UC9 – Prijava

- **Glavni sudionik:** Korisnik
- **Cilj:** Prijaviti korisnika u sustav
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je registriran u bazi podataka
- **Opis osnovnog tijeka:**
 1. Neprijavljen korisnik odabire opciju za prijavu
 2. Unosi potrebne podatke za prijavu
 3. Podaci se provjeravaju u bazi podataka te se obavještava o uspješnoj prijavi
- **Opis mogućih odstupanja:**
 - 2.a Unos nepostojećeg korisničkog imena ili e-maila, unos pogrešne lozinke
 1. Korisnik je obaviješten o nastaloj grešci i mjestu na kojem se ona nalazi
 2. Korisnik potom ispravlja grešku i ponovo šalje podatke ili odustaje od prijave

UC10 – Odjava

- **Glavni sudionik:** Korisnik
- **Cilj:** Odjaviti se iz sustava
- **Sudionici:** -
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Prijavljen korisnik odabire opciju za odjavu
 2. Obavještava se korisnika o uspješnoj odjavi

UC11 – Prikaz korisničkog računa

- **Glavni sudionik:** Korisnik
- **Cilj:** Prikazati podatke o korisničkom računu

- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Prijavljen korisnik odabire opciju za prikaz svog korisničkog računa
 2. Prikazuju se podaci o korisničkom računu

UC12 – Unos podataka za kupovinu za prijavljenog korisnika

- **Glavni sudionik:** Korisnik
- **Cilj:** Popuniti formular za kupovinu makete
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u bazi podataka
- **Opis osnovnog tijeka:**
 1. Prijavljen korisnik odabire opciju za kupnju makete
 2. Formular s podacima za kupovinu se automatski ispuni osobnim podacima
 3. Odabire se opcija za izvršavanje transakcije/kupnje
 4. Obavještava se korisnika o uspješnoj transakciji/kupnji

UC13 – Unos podataka za kupovinu za neregistriranog korisnika

- **Glavni sudionik:** Korisnik
- **Cilj:** Popuniti formular za kupovinu makete
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Neregistrirani korisnik odabire opciju za kupnju makete
 2. Ispunjava formular za kupovinu s osobnim podacima
 3. Odabire se opcija za izvršavanje transakcije/kupnje
 4. Obavještava se korisnika o uspješnoj transakciji/kupnji
- **Opis mogućih odstupanja:**
 - 2.a Unos pogrešnih osobnih podataka ili podataka potrebnih za transakciju
 1. Korisnik je obaviješten o nastaloj grešci i mjestu na kojem se ona nalazi
 2. Korisnik potom ispravlja grešku i ponovo šalje podatke ili odustaje od kupnje

UC14 – Spremanje izvršenih transakcija

- **Glavni sudionik:** Baza podataka
- **Cilj:** Spremiti izvršene transakcije u bazu podataka
- **Sudionici:** -
- **Preduvjet:** Transakcija je uspješno obavljena
- **Opis osnovnog tijeka:**
 1. Sustav javlja korisniku da je transakcija uspješna
 2. Baza podataka sprema podatke o transakciji

UC15 – Pregled izvršenih transakcija

- **Glavni sudionik:** Administrator
- **Cilj:** Pregled podataka o izvršenim transakcijama
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodana su mu prava administratora
- **Opis osnovnog tijeka:**
 1. Administrator odabire opciju za pregled prošlih transakcija
 2. Prikazuju se podaci iz baze podataka o transakcijama

UC16 – Pregled maketa

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisniku se pruža stranica s kratkim prikazom svih maketa u sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik s početne stranice poveznicom dolazi do stranice s maketama
 2. Aplikacija korisniku prikazuje sve makete u sustavu

UC17 – Pregled pojedine makete

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisniku se pruža detaljan prikaz pojedine makete
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik sa stranice s pregledom svih maketa odabire maketu koja ga zanima

2. Aplikacija korisniku prikazuje detaljan opis tražene makete koji uključuje tekst, slike, izbornike te mogućnost kupovine makete

UC18 – Odabir materijala za maketu

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisniku se pruža mogućnost odabira materijala od kojeg želi da se izradi maketa
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti na stranici pojedinačne makete
- **Opis osnovnog tijeka:**
 1. Korisnik na stranici makete otvara padajući izbornik s opcijama za materijale te odabire željeni materijal
 2. Aplikacija na osnovu odabranog materijala izračunava cijenu makete te prikazuje novu cijenu na stranici

UC19 – Spremanje podataka formulara za plaćanje

- **Glavni sudionik:** Korisnik
- **Cilj:** Formular koji je ispunio registrirani korisnik pri plaćanju se sprema za ponovno korištenje
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti registriran te ispravno popuniti formular
- **Opis osnovnog tijeka:**
 1. Korisnik na stranici za plaćanje ispunjava formular
 2. Aplikacija provjerava ispravnost unesenih podataka
 3. Nakon provjere podatci se spremaju u bazu podataka za ponovnu upotrebu pri sljedećoj kupovini
- **Opis mogućih odstupanja:**
 - 2.a Korisnik neispravno popuni jedno ili više polja ili ostavi prazno polje
 1. Korisnik je obaviješten o grešci i dano mu je pojašnjenje te greške
 2. Korisnik prepravlja ili nadopunjava podatke te se oni ponovno šalju na validaciju ili odustaje od ispunjavanja formulara

UC20 – Popunjavanje formulara za maketu po narudžbi

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisnik predaje formular sa specifikacijama željene makete
- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik mora biti registriran te ispravno popuniti formular
- **Opis osnovnog tijeka:**
 1. Korisnik na stranici za ispunjavanje formulara popunjava formular sa željenim dimenzijama, materijalom i opisom
 2. Aplikacija provjerava ispravnost unesenih podataka
 3. Nakon provjere podatci se spremaju u bazu podataka
- **Opis mogućih odstupanja:**
 - 2.a Korisnik neispravno popuni jedno ili više polja ili ostavi prazno polje
 1. Korisnik je obaviješten o grešci i dano mu je pojašnjenje te greške
 2. Korisnik prepravlja ili nadopunjava podatke te se oni ponovno šalju na validaciju ili odustaje od ispunjavanja formulara

UC21 – Ponuda makete na zahtjev

- **Glavni sudionik:** Administrator
- **Cilj:** Administrator pregledava zahtjev te šalje ponudu na osnovu traženih specifikacija
- **Sudionici:** Baza podataka, korisnik
- **Preduvjet:** Administrator mora imati predan zahtjev od strane korisnika
- **Opis osnovnog tijeka:**
 1. Administrator dobija obavijest o podnesenom zahtjevu te otvara zahtjev
 2. Administrator na osnovu zahtjeva određuje cijenu te šalje ponudu korisniku
 3. Korisnik dobija obavijest o ponudi

UC22 – Prihvat ili odbijanje ponude makete

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisnik prihvća ili odbija ponuđenu cijenu makete
- **Sudionici:** Baza podataka, administrator
- **Preduvjet:** Korisnik mora biti registriran
- **Opis osnovnog tijeka:**
 1. Korisnik dobija obavijest o ponudi cijene za maketu
 2. Korisnik odbija ili prihvća ponudu
 3. Šalje se povratna informacija administratoru

UC23 – Postavljanje postavki vidljivosti korisničkog računa

- **Glavni sudionik:** Korisnik

- **Cilj:** Korisnik postavlja željenu vidljivost svojih podataka
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik mora biti registriran
- **Opis osnovnog tijeka:**
 1. Korisnik odabire stranicu s postavkama korisničkog računa
 2. Korisnik odabire opciju postavljanja vidljivosti
 3. Korisnik odabire stavke koje želi da budu privatne ili javne (korisničko ime mora biti javno)
 4. Korisnik potvrđuje odabir te se on šalje sustavu
 5. Stranica osvježava podatke
- **Opis mogućih odstupanja:**
 - 4.a Korisnik ne potvrdi svoj odabir prije napuštanja stranice
 1. Korisnik je obaviješten o grešci te se nudi opcija povratka na potvrdu odabira
 2. Korisnik potvrđuje ili odustaje od izmjena

UC24 – Zabrana pristupa korisniku

- **Glavni sudionik:** Administrator
- **Cilj:** Administrator zabranjuje pristup određenom registriranom korisniku
- **Sudionici:** Baza podataka, korisnik
- **Preduvjet:** Korisnik mora biti registriran
- **Opis osnovnog tijeka:**
 1. Administrator odabire korisnika kojem želi zabraniti pristup
 2. Potvrđuje odabir te se sprema u bazu podataka

UC25 – Kreiranje standardne ponude maketa

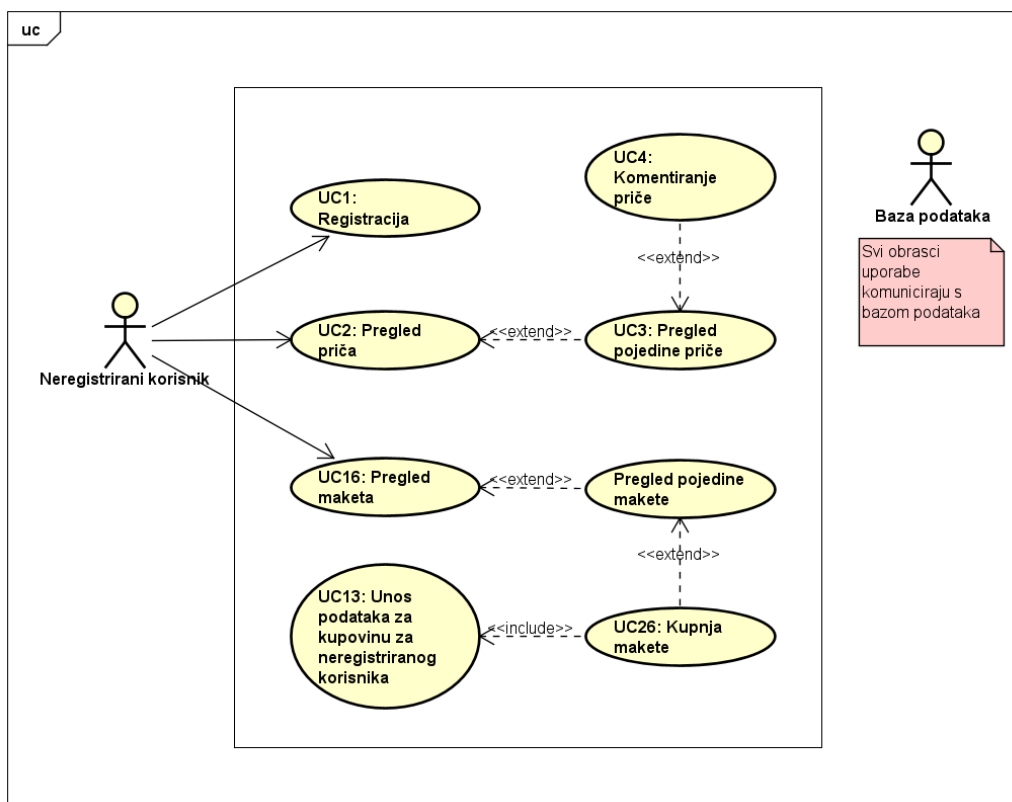
- **Glavni sudionik:** Administrator
- **Cilj:** Administrator ispunjava i objavljuje standardnu ponudu na stranici
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Administrator ispunjava podatke za novu ponudu makete
 2. Administrator potvrđuje formular te se šalje u bazu podataka
 3. Nova maketa se sprema te se počinje prikazivati na stranici
- **Opis mogućih odstupanja:**
 - 2.a Administrator krivo ispuni ili ostavi prazno polje u formularu

1. Administrator je obaviješten o grešci te se nudi opcija povratka na ispunjavanje formulara
2. Korisnik izmjenjuje ili nadopunjava podatke te ponovno potvrđuje odabir

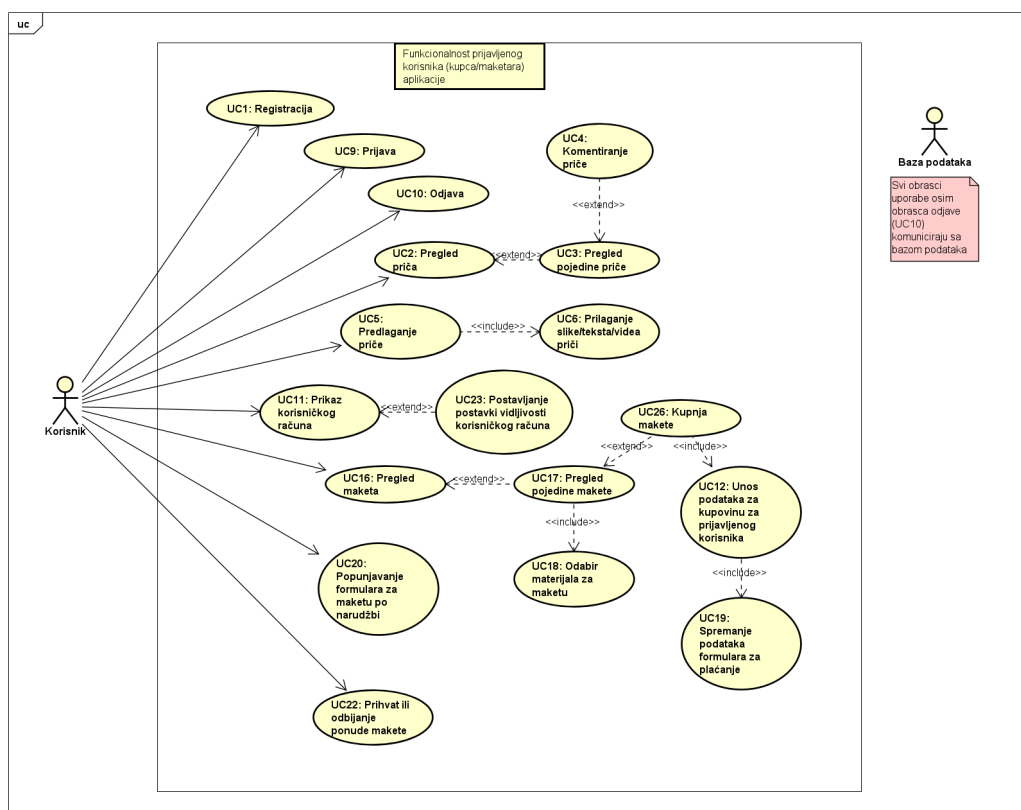
UC26 – Kupnja makete

- **Glavni sudionik:** Korisnik
- **Cilj:** Korisniku se nudi obrazac koji služi za popunjavanje podataka potrebnih za kupnju
- **Sudionici:** Baza podataka
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik sa stranice makete odabire opciju kupnje nje
 2. Korisniku se prikazuje obrazac u kojem popunjava ili provjerava podatke za kupnju

Dijagrami obrazaca uporabe



Slika 3.1: Dijagram obrasca uporabe, funkcionalnost neregistriranog korisnika



Slika 3.2: Dijagram obrasca uporabe, funkcionalnost prijavljenog korisnika



Slika 3.3: Dijagram obrasca uporabe, funkcionalnost administratora

3.1.2 Sekvencijski dijagrami

Obrazac upotrebe UC5 - Predlaganje priče i UC6 - Prilaganje slike/teksta/videoa priči

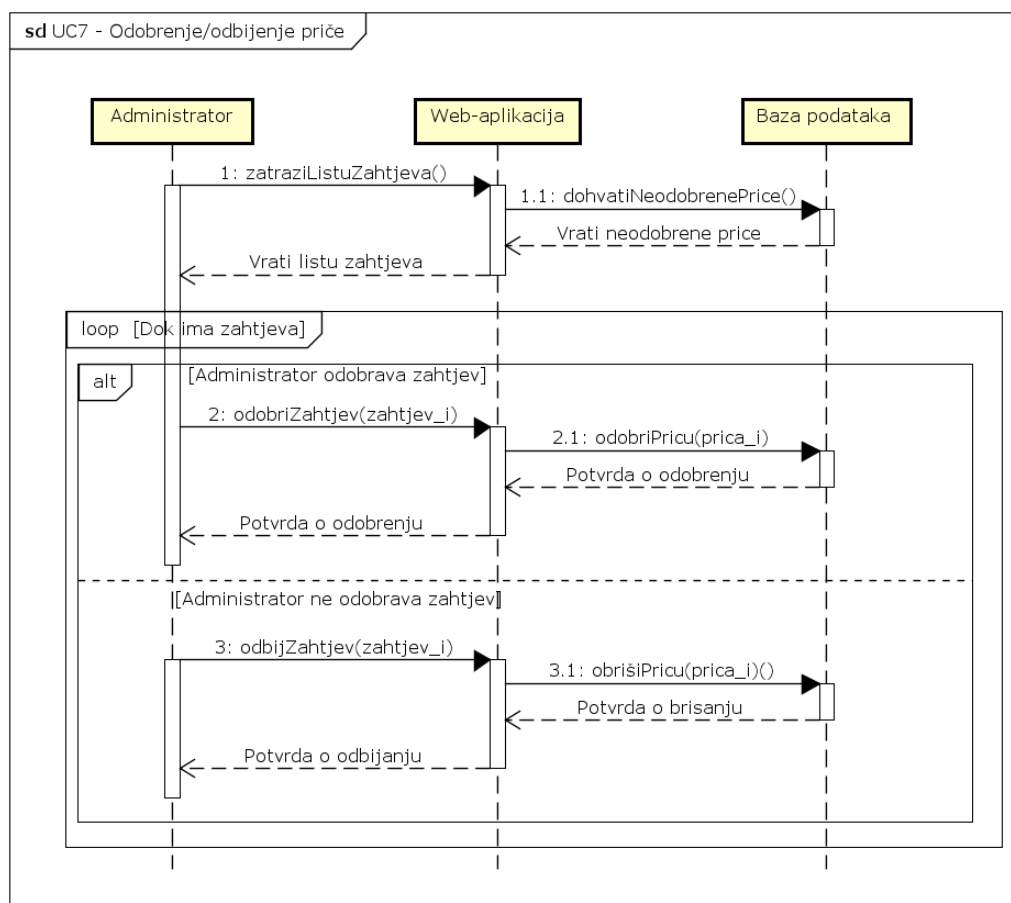
Korisnik u formular unosi željeni naslov priče, a zatim i prozvoljan broj komada sadržaja tipa tekst, slika i video. Priča se na poslužitelj prenosi u jednom komadu pa se do slanja sprema u lokalnoj memoriji web-preglednika. Po prijenosu, aplikacija nad pristiglim podacima provodi *sanity check*, to jest provjerava ispravnost podataka. Ako su podaci ispravni, šalju se bazi podataka. Baza stvara novu n-torku u relaciji Story i pohranjuje opće podatke o novonastaloj priči te ju označava neodobrenom. Potom stvara po jednu n-torku u relaciji StoryContent za svaki komad sadržaja priče. Ako je korisnik poslao neispravne podatke, ili praznu priču, aplikacija mu šalje poruku o grešci i daje mu da ponovno unese potrebne podatke.



Slika 3.4: Sekvencijski dijagram, UC5 - Predlaganje priče i UC6 - Prilaganje slike / teksta / videa priči

Obrazac upotrebe UC7 - Odobrenje/odbijanje priče

Administrator od aplikacije traži sve neodobrene zahtjeve. Aplikacija od baze dohvaća sve neodobrene priče i šalje ih nazad administratoru. Za svaku neodobrenu priču administrator ima opciju odobriti ju ili odbiti. Odobrene priče aplikacija u bazi označava odobrenima. Odbijene priče aplikacija briše iz baze podataka.



Slika 3.5: Sekvencijski dijagram, UC7 - Odobrenje/odbijanje priče

Obrazac upotrebe UC20 - Popunjavanje formulara za maketu po narudžbi

Korisnik ispunjava formular u kojem opisuje vlastitu maketu te ju šalje aplikaciji. Aplikaciji provodi *sanity check* pristiglih podataka. Ako su podaci ispravni, šalju se bazi podataka. Baza stvara novu n-torku u relaciji Models, pohranjuje korisnikove opise u nju i označava ju neodobrenom. Aplikacija potom korisniku šalje potvrdu o zaprimtku zahtjeva. Ako su podaci nepotpuni ili neispravni na neki drugi način, aplikacija korisniku šalje poruku u grešci i daje mu da ponovno unese svoj opis.



Slika 3.6: Sekvencijski dijagram, UC20 - Popunjavanje formulara za maketu po narudžbi

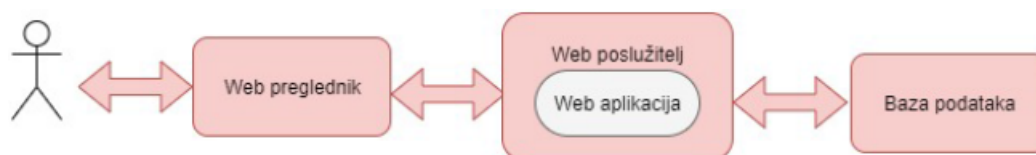
3.2 Ostali zahtjevi

- Sustav treba biti implementiran u jeziku objektno-orijentirane paradigme
- Sustav treba omogućiti nesmetan rad više korisnika u isto vrijeme
- Sustav mora podržavati hrvatsku abecedu
- Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
- Korisničko sučelje treba biti jednostavno i intuitivno za korištenje
- Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
- Veza s bazom podataka mora biti sigurna, brza i otporna na vanjske greške
- Sustav kao valutu koristi HRK
- Veza sa sustavom mora biti zaštićena HTTPS-om

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 (internetski preglednik) program je koji omogućuje korisnicima pristup web stranicama i multimedijalnim sadržajima vezanih uz njih. Svaki internetski preglednik je prevoditelj što znači da je stranica pisana u kodu koji se potom interpretira kao nešto svakome razumljivo. Putem web preglednika korisnik šalje zahtjeve web poslužitelju.

Web poslužitelj je računalni program koji služi kao medij između korisnika i drugih programa ili uređaja. Poslužitelj je osnova rada web aplikacije budući da on pokreće web aplikaciju te joj preusmjerava zahtjeve klijenata. Njegova primarna funkcija je pohranjivanje, obrada i isporuka web stranica klijentima. Komunikacija između klijenta i poslužitelja odvija se pomoću protokola HTTP (engl. *Hyper Text Transfer Protocol*).

Web aplikacija je računalni program kojemu klijent pristupa iz web preglednika za obradu željenih zahtjeva. Ovisno o tipu zahtjeva, aplikacija pristupa bazi podataka zatim preko poslužitelja vraća korisniku odgovor vidljiv u web pregledniku kao HTML dokument.

Programski jezik koji smo koristili za izradu web aplikacije "Maketa Shop" je Python s Flask radnim okvirom.

MVC (Model-View-Controller) je obrazac koji se obično koristi za razvoj korisničkih sučelja zato jer dijeli programsku logiku na tri međusobno povezana elementa što pojednostavnjuje razvoj web aplikacija. Flask nije struktuiran kao čisti MVC okvir, ali po funkcionalnosti je vrlo sličan. Stoga možemo definirati sva tri elementa:

- Model — Središnja komponenta sustava. To je dinamička struktura podataka aplikacije neovisna o korisničkom sučelju. Izravno upravlja podacima, logikom i pravilima aplikacije i prima podatke od Controllera.
- View — Svako predstavljanje informacija poput grafova, dijagrama ili tablica. Moguće je višestruko prikazivanje istih podataka, poput trake grafikona za upravljanje i tabličnog prikaza.
- Controller — Prihvaća unos i pretvara ga u naredbe za Model ili View. Upravlja korisničkim zahtjevima te izvodi daljnju interakciju.

4.1 Baza podataka

Sustav koristi relacijsku bazu podataka. Baza se sastoji od relacija, tablica koje svojim atributima modeliraju pojave iz stvarnog svijeta. Uloga baze je struktuirana pohrana podataka radi njihova lakšeg dohvaćanja i izmjenjivanja. Baza ovog sustava se sadrži od sljedećih entiteta:

- User (korisnik)
- Banned (blokirani korisnici)
- Profile (osobni podaci korisnika)
- BillingInfo (podaci o plaćanju)
- Story (priča)
- StoryContent (dio sadržaja priče)
- Model (maketa)
- ModelPhoto (fotografija koja se prilaže maketi)
- ModelPrice (cijena modela ovisno o materijalu)

- ModelNotification (obavijest)
- Comment (komentar)
- Order (narudžba)
- OrderModel (makete po narudžbi)
- Cart (košarica)
- CartModel (model u košarici)

4.1.1 Opis tablica

User Podaci potrebni za prijavu u sustav. Sadrži attribute: identifikacijski broj, korisničko ime, e-mail adresu i lozinku. U vezi je *One-to-One* s entitetom BillingInfo preko atributa id, u vezi *One-to-One* s entitetom Profile preko atributa id, u vezi *One-to-Many* s entitetom Story preko atributa id, u vezi *One-to-Many* s entitetom Order preko atributa id, u vezi *One-to-Many* s entitetom Model preko atributa id, u vezi *One-to-Many* s entitetom Cart preko atributa id, u vezi *One-to-Many* s entitetom Model preko atributa id, u vezi *One-to-Many* s entitetom Comment preko atributa id, u vezi *One-to-Many* s entitetom ModelNotification preko atributa id.

User		
id	INT	identifikacijski broj korisnika; samoinkrementirajući broj jedinstven za svakog korisnika
username	VARCHAR	ime koje se prikazuje drugim korisnicima; jedinstveno
email	VARCHAR	E-mail adresa kojom se prijavljuje u sustav; jedinstvena
password	CHAR	lozinka kojom se prijavljuje u sustav, kriptirana algoritmom Bcrypt na 60 bajtova

Profile Podaci koje korisnik može prikazati na svom profilu. Sadrži attribute: ime, prezime, datum rođenja, kratak životopis, te postavke privatnosti za svaku od stavki. U vezi *One-to-One* s entitetom User.

Profile		
user_id	INT	identifikacijski broj korisnika kojemu profil pripada
first_name	VARCHAR	korisnikovo ime; opcionalno
private_name	BOOLEAN	TRUE ako je javno, FALSE ako nije
last_name	VARCHAR	korisnikovo prezime; opcionalno
private_lname	BOOLEAN	TRUE ako je javno, FALSE ako nije
DOB	DATETIME	korisnikov datum rođenja; opcionalno
private_DOB	BOOLEAN	TRUE ako je javno, FALSE ako nije
bio	TEXT	kratak životopis; opcionalno
private_bio	BOOLEAN	TRUE ako je javno, FALSE ako nije

BillingInfo Sadrži korisnikove podatke o plaćanju. Sadrži attribute: puno ime, adresa za naplatu, broj kreditne kartice, datum isteka kartice, CVC kartice, državu, puno ime kartice, zip kod i email adresu. U vezi *One-to-One* s entitetom User.

BillingInfo		
user_id	INT	identifikacijski broj korisnika kojemu podaci pripadaju
full_name	VARCHAR	korisnikovo ime i prezime
email	VARCHAR	email adresa korisnika
address	VARCHAR	adresa
country	VARCHAR	država
city	VARCHAR	grad
zip_code	VARCHAR	poštanski broj
card_full_name	VARCHAR	puno ime kreditne kartice
card_number	VARCHAR	broj kreditne kartice
card_expiry	DATETIME	datum isteka kartice
card_CVC	INT	CVC kartice

Story Osnovne informacije o priči koju korisnik postavlja na aplikaciju. Sadrži attribute: identifikacijski broj, naslov priče, vrijeme nastanka, stanje (prihvaćena / neprihvaćena) i identifikacijski broj autora. U vezi je *One-to-Many* s entitetom StoryContent te u vezi *Many-to-One* s entitetom User.

Story		
id	INT	identifikacijski broj priče; samoinkrementirajući broj jedinstven za svaku priču
title	TEXT	naslov priče
time_created	DATETIME	vrijeme nastanka priče
validated	BOOLEAN	TRUE ako je priča prihvaćena, FALSE ako nije
author_id	INT	identifikacijski broj autora

StoryContent Tekst, slika ili video koji se prilaže priči. Sadrži attribute: identifikacijski broj priče, redni broj dijela sadržaja, tekst priče, ime slike i ime videa. U vezi je *Many-to-One* s entitetom Story.

StoryContent		
story_id	INT	identifikacijski broj priče kojemu se sadržaj prilaže
ordinal_number	INT	redni broj dijela sadržaja u priči
story_text	TEXT	tekst koji se prilaže priči; opcionalno
image_name	VARCHAR	ime slike koja se prilaže priči; opcionalno
video_name	VARCHAR	ime videa koji se prilaže priči, opcionalno

Model Podaci o maketi koju predlaže korisnik. Sadrži attribute: identifikacijski broj, ime, opis, identifikacijski broj korisnika koji ju predlaže, dimenzije, boje i odobrenost. U vezi je *Many-to-One* s entitetom User, u vezi *One-to-Many* s entitetom ModelPrice, u vezi *One-to-Many* s entitetom ModelPhoto, u vezi *Many-to-Many* s entitetom Order-Model te u vezi *One-to-Many* s entitetom CartModel.

Model		
id	INT	identifikacijski broj makete; samoinkrementirajući broj jedinstven za svaku maketu
name	VARCHAR	ime makete; jedinstveno
description	TEXT	opis makete
dimension	VARCHAR	dimenzije; odvojene znakom ", "
colors	VARCHAR	dostupne boje; odvojene znakom ", "
approved	BOOLEAN	TRUE ako je model prihvaćen, FALSE ako nije

Model		
creator_id	INT	identifikacijski broj korisnika koji predlaže maketu

ModelPhoto Fotografija koja se prilaže maketi. Sadrži attribute: ime slike i identifikacijski broj makete. U vezi *Many-to-One* s entitetom Model.

ModelPhoto		
image_name	VARCHAR	ime fotografije
model_id	INT	identifikacijski broj makete kojemu se fotografija prilaže

ModelPrice Cijena makete ovisno o materijalu od kojeg je izrađena. Sadrži attribute: identifikacijski broj modela, materijal od kojeg je izrađena, cijenu makete te samoinkrementirajući id makete. U vezi *Many-to-One* s entitetom Model.

ModelPrice		
model_id	INT	identifikacijski broj makete
material	VARCHAR	ime materijala
price	NUMERIC	cijena makete za zadani materijal
id	INT	identifikacijski broj; samoinkrementirajući broj jedinstven za svaku instancu ModelPrice

Order Podaci o narudžbi koju korisnik šalje trgovini. Sadrži attribute: identifikacijski broj narudžbe, vrijeme nastanka, identifikacijski broj korisnika koji šalje narudžbu te ime, adresu i email kupca. U vezi je *Many-to-One* s entitetom User i u vezi *One-to-Many* s entitetom OrderModel.

Order		
id	INT	identifikacijski broj narudžbe; samoinkrementirajući broj jedinstven za svaku narudžbu
time_created	DATETIME	vrijeme nastanka
buyer_name	VARCHAR	ime kupca
buyer_email	VARCHAR	email adresa kupca
buyer_address	VARCHAR	adresa kupca

Order		
user_id	INT	identifikacijski broj korisnika koji šalje narudžbu

OrderModel Lista maketa u narudžbi. Sadrži attribute: identifikacijski broj narudžbe, identifikacijski broj makete, materijal od kojeg je maketa izrađena količinu i cijena makete u trenutku narudžbe.

OrderModel		
order_id	INT	identifikacijski broj narudžbe
model_id	INT	identifikacijski broj makete
material	VARCHAR	ime materijala od kojeg je maketa izrađena
price	NUMERIC	cijena makete za zadani materijal u trenutku narudžbe
quantity	INT	količina makete

Comment Komentari ispod pojedine priče, u vezi *Many-to-One* s entitetom User te u vezi *Many-to-One* s entitetom Story.

Comment		
id	INT	identifikacijski broj komentara
id	INT	identifikacijski broj priče
author_id	INT	identifikacijski broj autora komentara
text	TEXT	tekst komentara
timestamp	DATETIME	vrijeme nastanka komentara

ModelNotification Obavijest o prihvaćanju naručene makete, u vezi *Many-to-One* s entitetom Model te u vezi *Many-to-One* s entitetom User.

ModelNotification		
id	INT	identifikacijski broj obavijesti
receiver_id	INT	identifikacijski broj primatelja obavijesti
model_id	INT	identifikacijski broj naručenog modela
seen	BOOLEAN	pregledanost obavijesti
time_create	DATETIME	vrijeme nastanka obavijesti

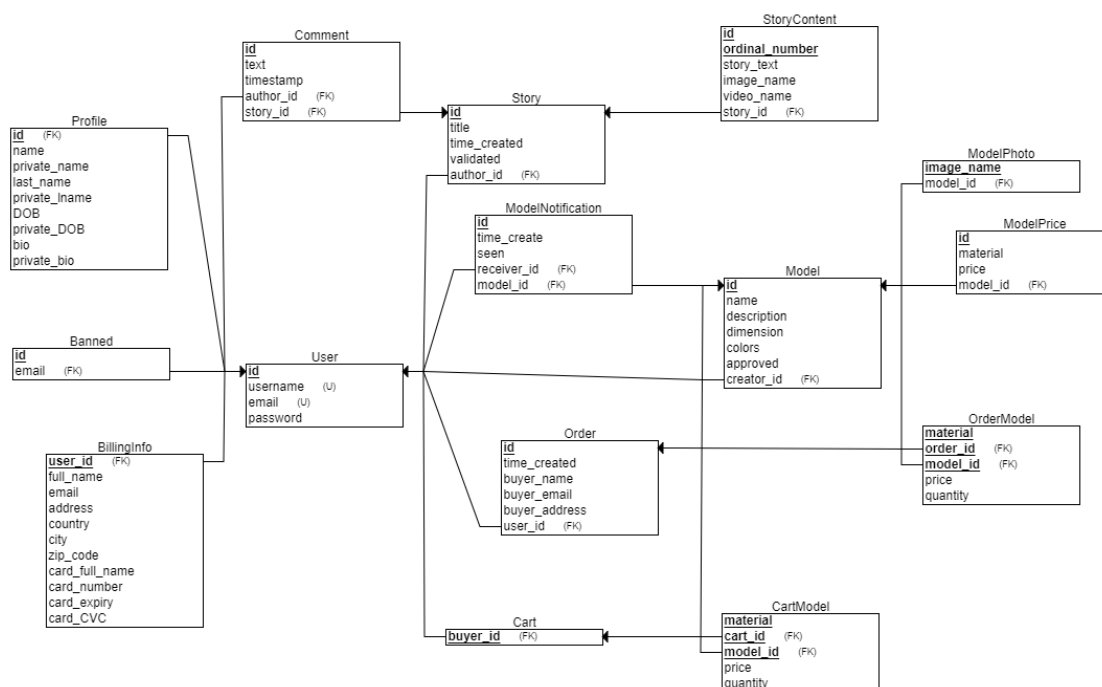
Cart Entitet košarice pridružen entitetu User vezom *One-to-One*.

Cart		
buyer_id	INT	identifikacijski broj kupca

CartModel Lista maketa u košarici. Sadrži attribute: identifikacijski broj košarice, identifikacijski broj makete, materijal od kojeg je maketa izrađena te količinu i cijenu makete u košarici, u vezi *Many-to-One* s entitetom Cart.

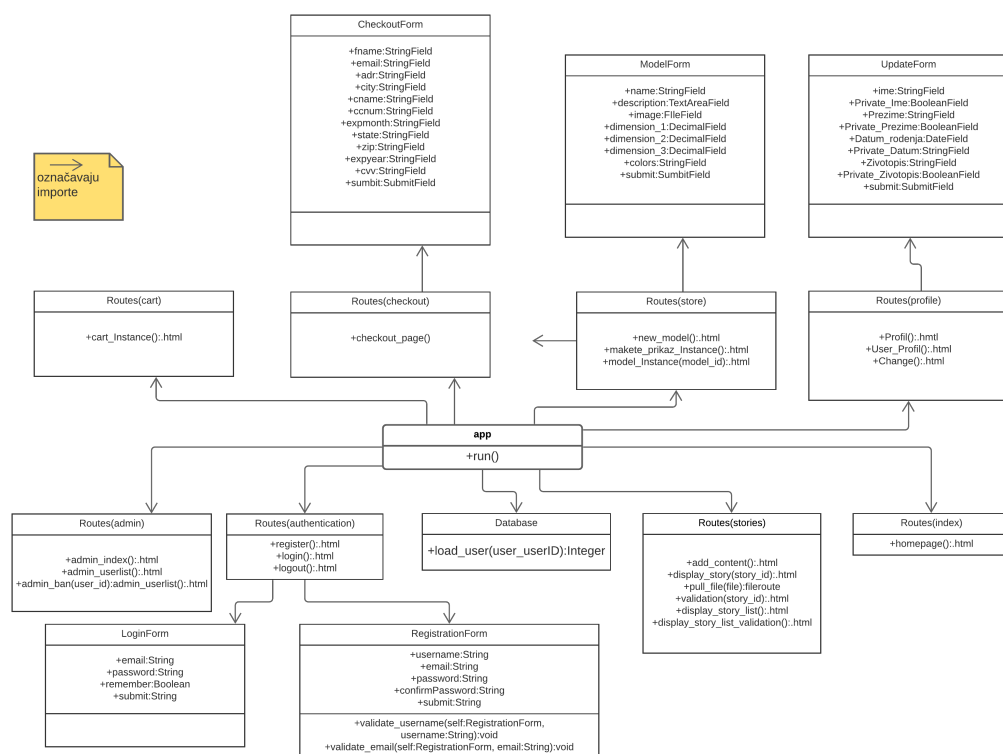
CartModel		
cart_id	INT	identifikacijski broj košarice
model_id	INT	identifikacijski broj makete
material	VARCHAR	ime materijala od kojeg je maketa izrađena
price	NUMERIC	cijena makete za zadani materijal u košarici
quantity	INT	količina maketa

4.1.2 Dijagram baze podataka



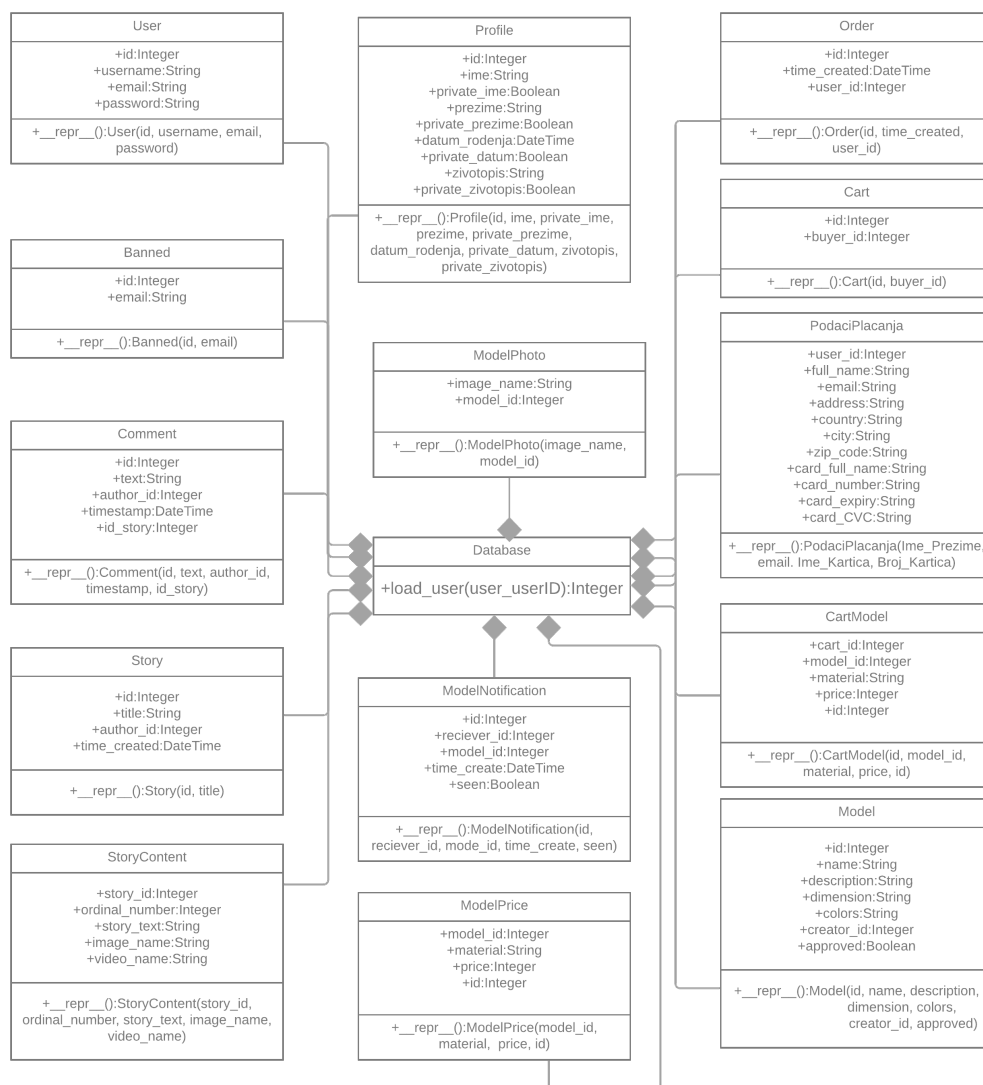
Slika 4.2: E-R dijagram baze podataka

4.2 Dijagram razreda



Slika 4.3: Dijagram razreda

Razredi vezani uz bazu podataka su smješteni na zasebnoj slici zbog lakše preglednosti.

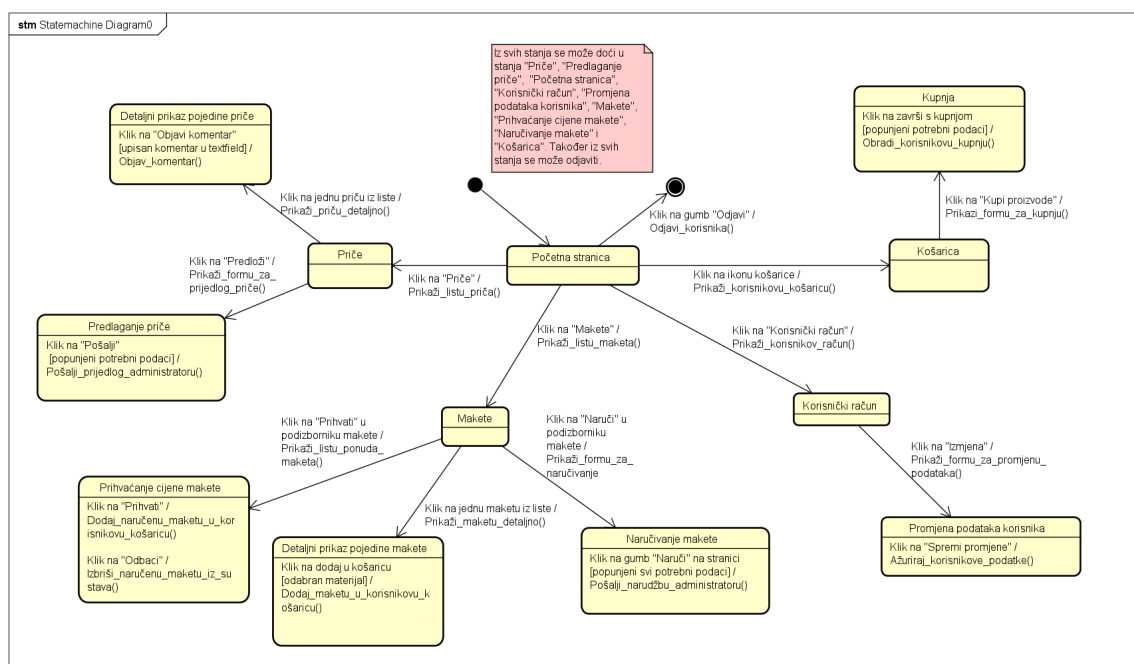


Slika 4.4: Dijagram razreda - Baza podataka

4.3 Dijagram stanja

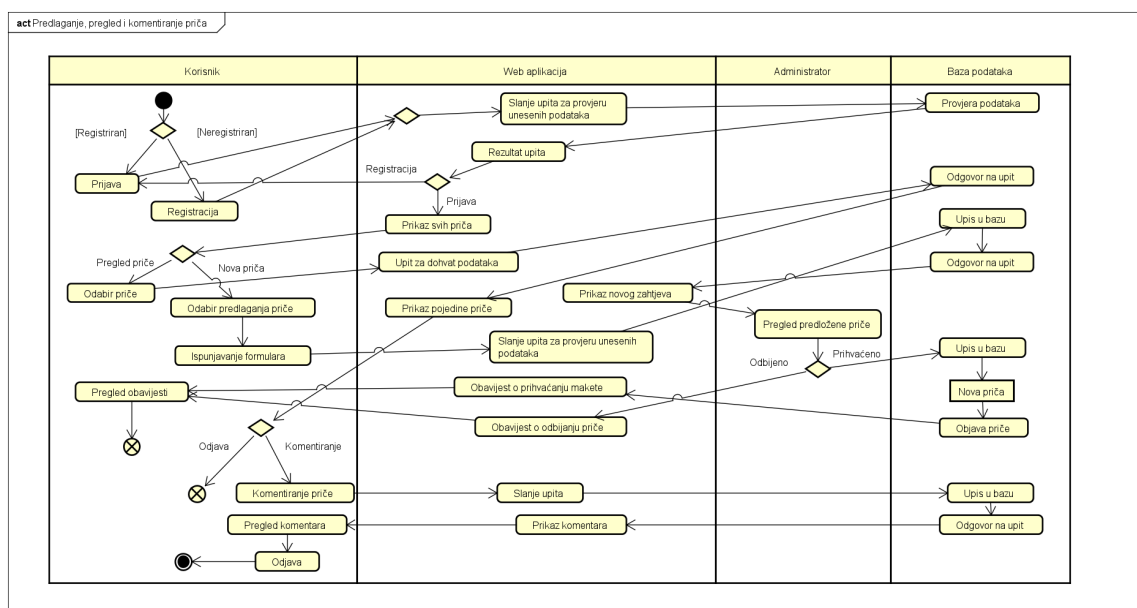
Dijagram stanja opisuje dinamičko ponašanje dijela sustava u vremenu. Njim se prikazuju stanje objekta te prijelazi iz jednog stanja u drugo temeljeni na događajima. Na slici ispod prikazan je dijagram stanja za prijavljenog korisnika u sustav. Korisniku se pri otvaranju stranice prikazuje početna stranica te dalje pomoću izbornika u zaglavlju pristupa funkcijama koje mu se nude. Prijavljeni korisnik može pregledavati, komentirati i predlagati priče. Također, može pregledavati standardne makete u sustavu i kupovati ih ili naručiti maketu po svojim zahtjevima. Klikom na "Korisnički račun" u izborniku korisniku se daje pregled njegovog računa sa

moгуćnošću izmjene podataka i vidljivosti svakog podatka. Na kraju, korisnik također može pristupiti svojoj košarici i preko nje dovršiti kupnju.

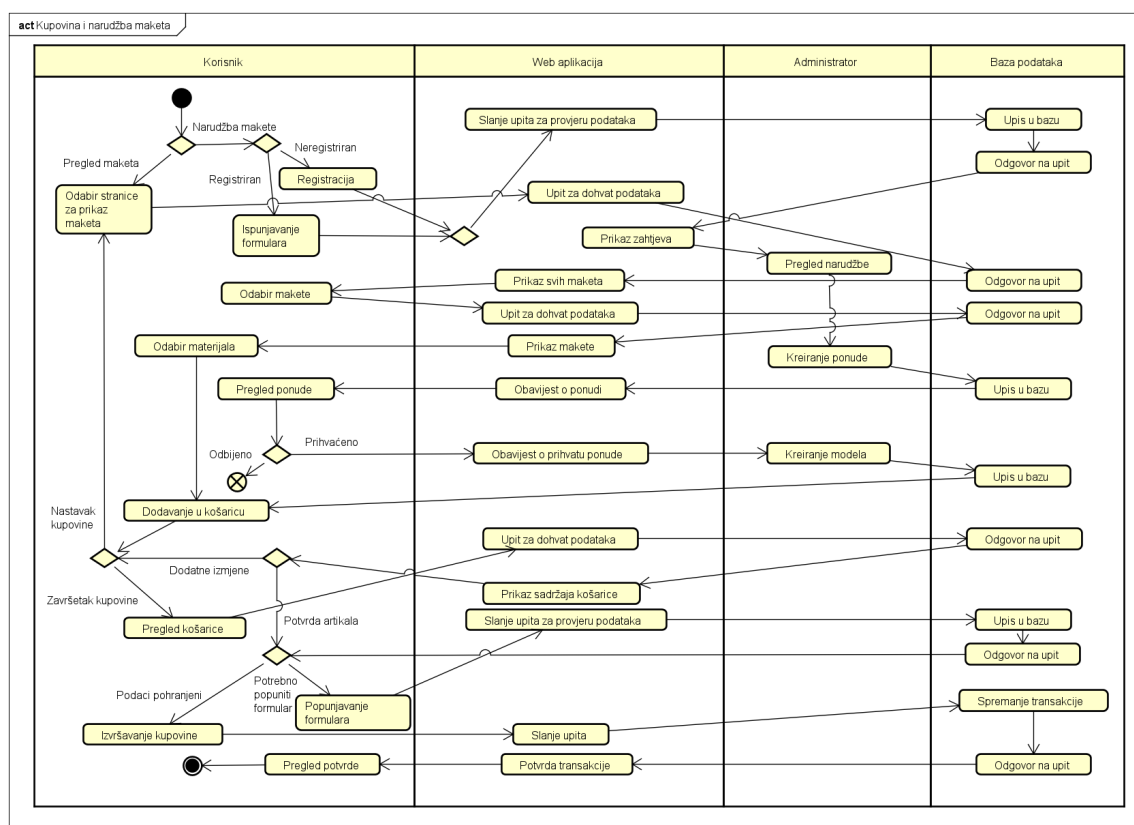


Slika 4.5: Dijagram stanja

4.4 Dijagram aktivnosti



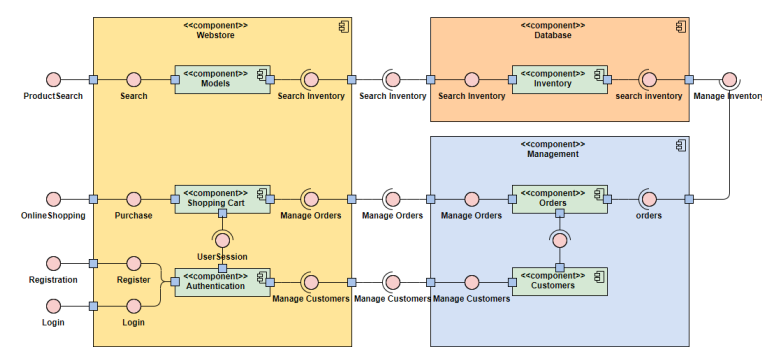
Slika 4.6: Dijagram aktivnosti za predlaganje, pregled i komentiranje priča



Slika 4.7: Dijagram aktivnosti za kupovinu i narudžbu maketa

4.5 Dijagram komponenti

Dijagram komponenti služi za prikaz organizacije i interakcije između komponenta implementacije te odnos programske potpore prema okolini. Oblikovanje sustava mora biti dovršeno prije nego se počnu modelirati komponente. Autentifikacija korisnika podrazumijeva mogućnost prijave i registracije novog korisnika. Korisnik može pregledavati ponuđene makete i nakon toga ih dodavati po potrebi u košaricu. Također postoji mogućnost narudžbe makete prema korisnikovim preferencijama i vlastitim skicama vezano uz veličinu, boju, materijal izrade itd. Opcija naručivanja maketa predviđena je samo za registrirane korisnike.



Slika 4.8: Dijagram komponenti

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

Za komunikaciju među članovima tima korišten je Whatsapp (<https://www.whatsapp.com/>).

Za praćenje napretka među članovima tima korišten je Miro (<https://miro.com/>).

Kao sustav za upravljanje izvornim kodom korišten je Git (<https://git-scm.com/>), a za udaljeni repozitorij projekta korišten je GitLab (<https://gitlab.com/>).

Za izradu UML dijagrama u dokumentaciji korišten je Astah UML (<https://astah.net/>).

Za izradu backend dijela aplikacije korišten je programski jezik python (<https://www.python.org/>) verzija 3.8, te framework Flask (<https://flask.palletsprojects.com/en/1.1.x/>). Kao aplikacijski server korišten je Heroku (<https://heroku.com/>). Korištena je baza podataka PostgreSQL (<https://www.postgresql.org/>).

5.2 Ispitivanje programskog rješenja

5.2.1 Ispitivanje komponenti

```
#1. provjera indexa
def test_index(self):
    tester = application.test_client(self)
    response = tester.get('/', content_type='html/text')
    self.assertEqual(response.status_code, 200)

#2. provjera trazi li prijedlog price da je korisnik ulogiran
def test_story_requires_login(self):
    tester = application.test_client(self)
    response = tester.get('/storyrequest', follow_redirects=True)
    self.assertIn(b'Unesi lozinku', response.data)

#3. provjera vraca li login upozorenje prilikom passworda s manje od 8 znakova
def test_user_login(self):
    tester=application.test_client(self)
    response = tester.post('/login', data=dict(email='bezveze@email.com', password='bla'), follow_redirects=True)
    self.assertIn(b"Lozinka mora biti barem 8 znakova duga.", response.data)

#4. provjera moze li maketu pregledati ne ulogirani korisnik
def test_anon_user_view_model(self):
    tester=application.test_client(self)
    response = tester.get('/makete_prikaz', follow_redirects=True)
    self.assertIn(b'Maketa grad', response.data)

#5. provjera nemogucnosti usera da pristupi admin stranici
def test_admin_page(self):
    tester=application.test_client(self)
    response = tester.get('/admin', follow_redirects=True)
    self.assertEqual(response.status_code, 404)

#6. provjera je li password dobar poslije unhasha
def test_pass_hash(self):
    user = User.query.filter_by(email='test1@mail.com').first()
    self.assertTrue(bcrypt.check_password_hash(user.password, '123123123'))
```

Slika 5.1: Ispitivanje jedinica

```
(ferocious) ~/Desktop/oldprojects/ferocious$ python3 test.py
/home/ferocious/.local/share/virtualenvs/ferocious-akn9mV/lib/python3.8/site-packages/flask_sqlalchemy/_init_.py:83: FSADeprecationWarning: SQLAlchemy_TRACK_MODIFICATIONS adds significant overhead and will be disabled by default in the future. Set it to True or False to suppress this warning.
  warnings.warn(FSADeprecationWarning(
.....
Ran 6 tests in 0.242s
OK
```

Slika 5.2: Rezultat ispitivanja jedinica

5.2.2 Ispitivanje sustava

Provedeno je testiranje rada sustava pomoću Selenium testova(Selenium IDE) . Testirane su neke od osnovnih funkcionalnosti sustava. Svi testovi izvršeni su ručno. U dokumentaciji su prikazani slučajevi UC1, UC9, UC10, UC11, UC23.

Ispitni slučaj 1 : Registracija

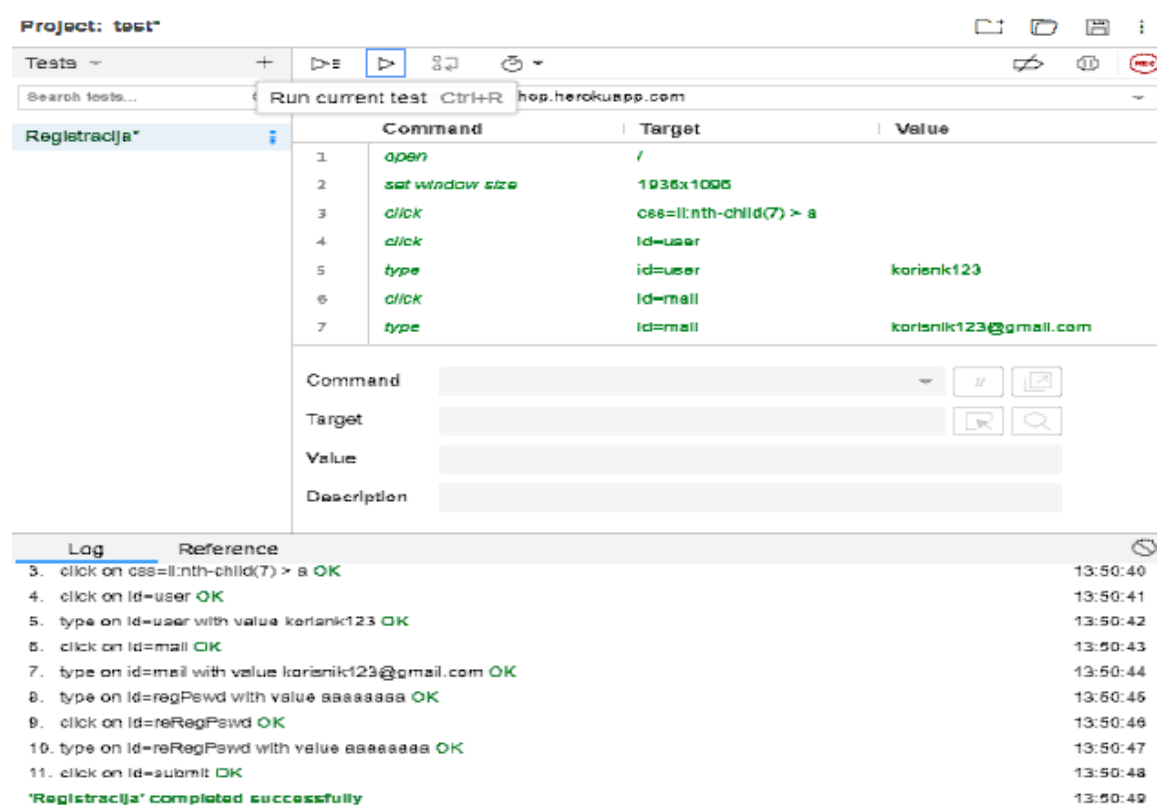
Ulaz :

- : 1. Otvaranje pocetne stranice u web pregledniku.
- 2. Odabir na gumb registracije
- 3. Unos potrebnih podataka

Ocekivani rezultat :

- : 1.a Prikazao se ekran za unos podataka
- 1.b Ako su uneseni valjani podaci korisnik ce biti preusmjeren na login screen

Rezultati : Očekivani rezultat 1.b je zadovoljen. Aplikacija je prošla test.



Slika 5.3: Ispitni test 1

Ispitni slučaj 2 : Prijava

Ulaz :

- : 1. Otvaranje pocetne stranice u web pregledniku.
- 2. Odabir na gumb prijava

- 3. Unos potrebnih podataka

Očekivani rezultat :

- : 1.a Prikazao se ekran za unos podataka
- 1.b Ako su uneseni valjani podaci korisnik ce biti preusmjeren na pocetnu stranicu

Rezultati : Očekivani rezultat 1.b je zadovoljen. Aplikacija je prošla test.

The screenshot displays the Selenium IDE interface. At the top, the status bar indicates 'Executing' and 'Run current test Ctrl+R'. The test name is 'Prijava*' and the URL is 'hop.herokuapp.com'. Below this, a table lists the test steps:

	Command	Target	Value
7	click	id=loginMail	
8	click	id=loginMail	
9	type	id=loginMail	korisnik123@gmail.com
10	click	id=loginLoz	
11	click	id=loginLoz	
12	type	id=loginLoz	aaaaaaaa
13	click	id=submit	

Below the table, there are input fields for 'Command', 'Target', 'Value', and 'Description'. At the bottom left, it shows 'Runs: 1 Failures: 0'. The bottom section is a log table:

Log	Reference	
5. click on id=loginMail	OK	14:01:18
6. doubleClick on id=loginMail	OK	14:01:19
7. click on id=loginMail	OK	14:01:20
8. click on id=loginMail	OK	14:01:21
9. type on id=loginMail with value korisnik123@gmail.com	OK	14:01:22
10. click on id=loginLoz	OK	14:01:23
11. click on id=loginLoz	OK	14:01:24
12. type on id=loginLoz with value aaaaaaaaa	OK	14:01:25
13. click on id=submit	OK	14:01:26
'Prijava' completed successfully		14:01:27

Slika 5.4: Ispitni test 2

Ispitni slučaj 3 : Odjava

Ulaz :

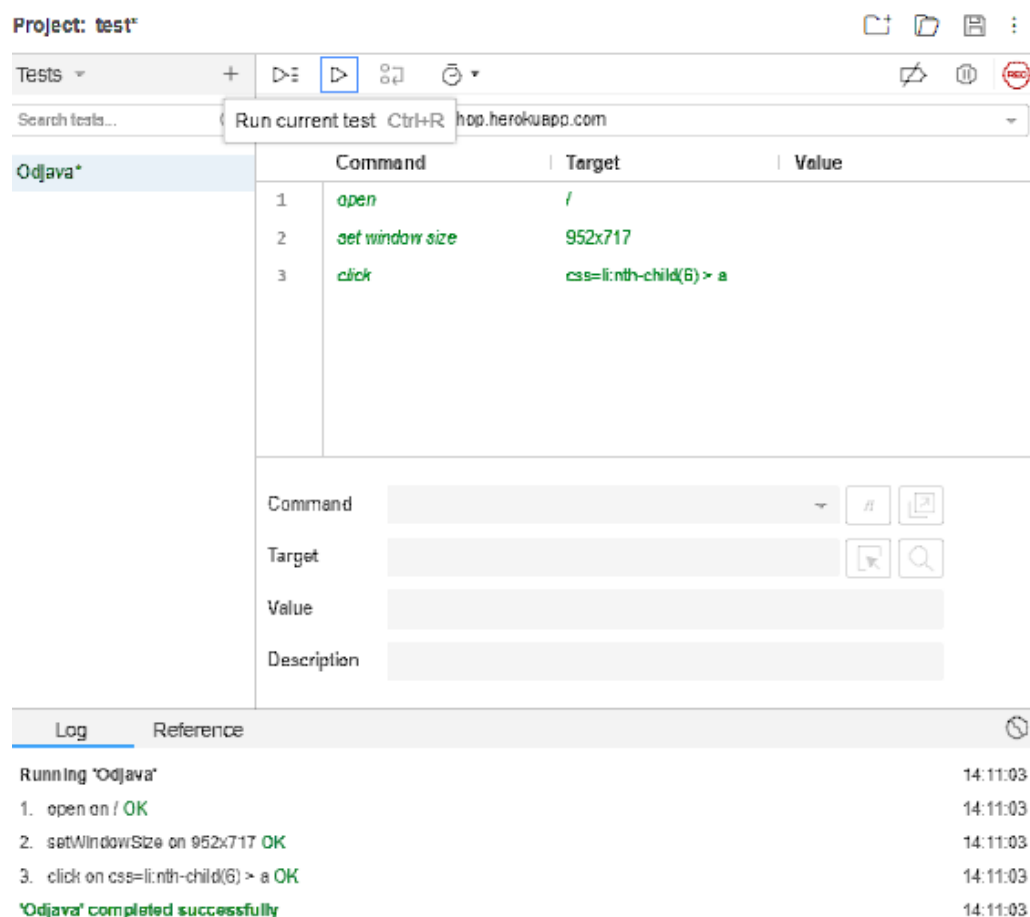
- : 1. Korisnik treba biti prijavljen

- 2. Odabir na gumb odjavi se

Očekivani rezultat :

- : 1. Korisnik će biti odjavljen i preusmjeren na pocetnu stranicu

Rezultati : Očekivani rezultat 1. je zadovoljen. Aplikacija je prošla test.



Slika 5.5: Ispitni test 3

Ispitni slučaj 4 : Odjava

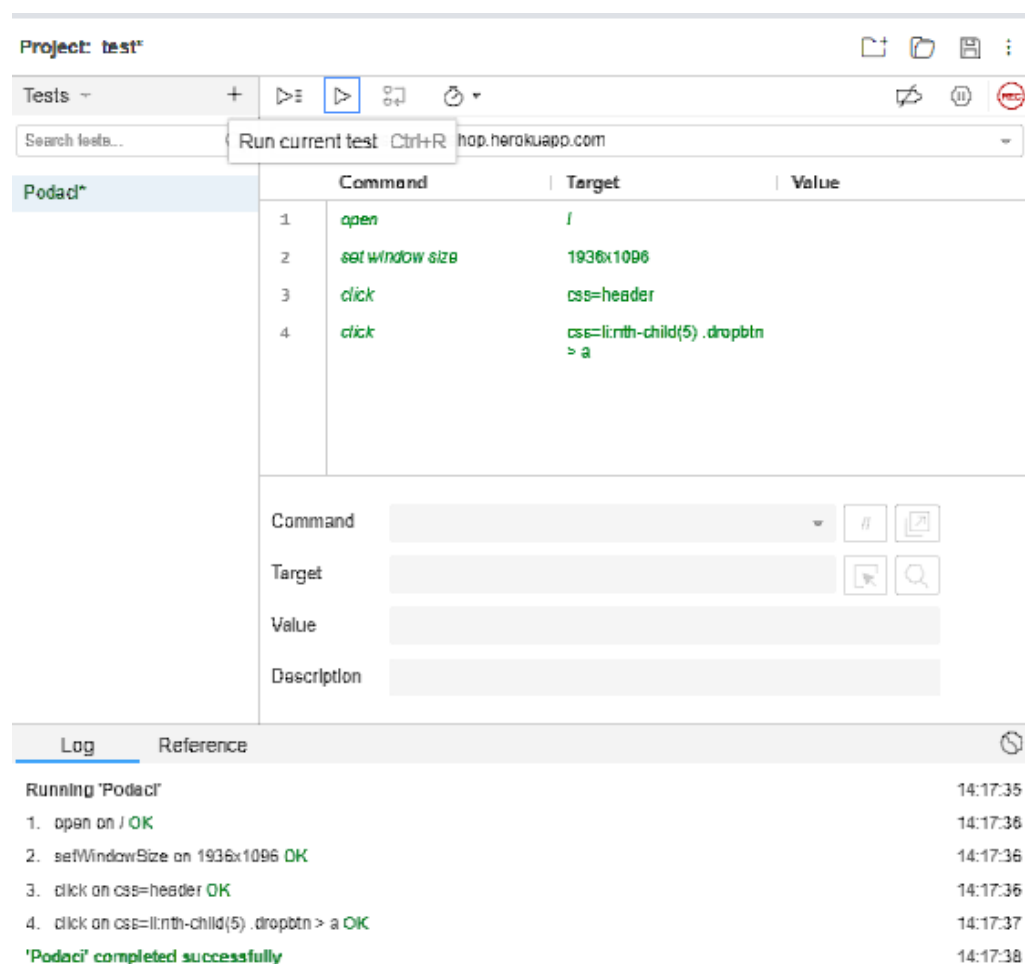
Ulaz :

- : 1. Korisnik treba biti prijavljen
- 2. Odabir na gumb moj profil

Očekivani rezultat :

- : 1. Korisnik će biti preusmjeren na stranicu sa svojim podacima

Rezultati : Očekivani rezultat 1. je zadovoljen. Aplikacija je prošla test.



Slika 5.6: Ispitni test 4

Ispitni slučaj 5 : Postavljanje postavki vidljivosti korisničkog računa

Ulaz :

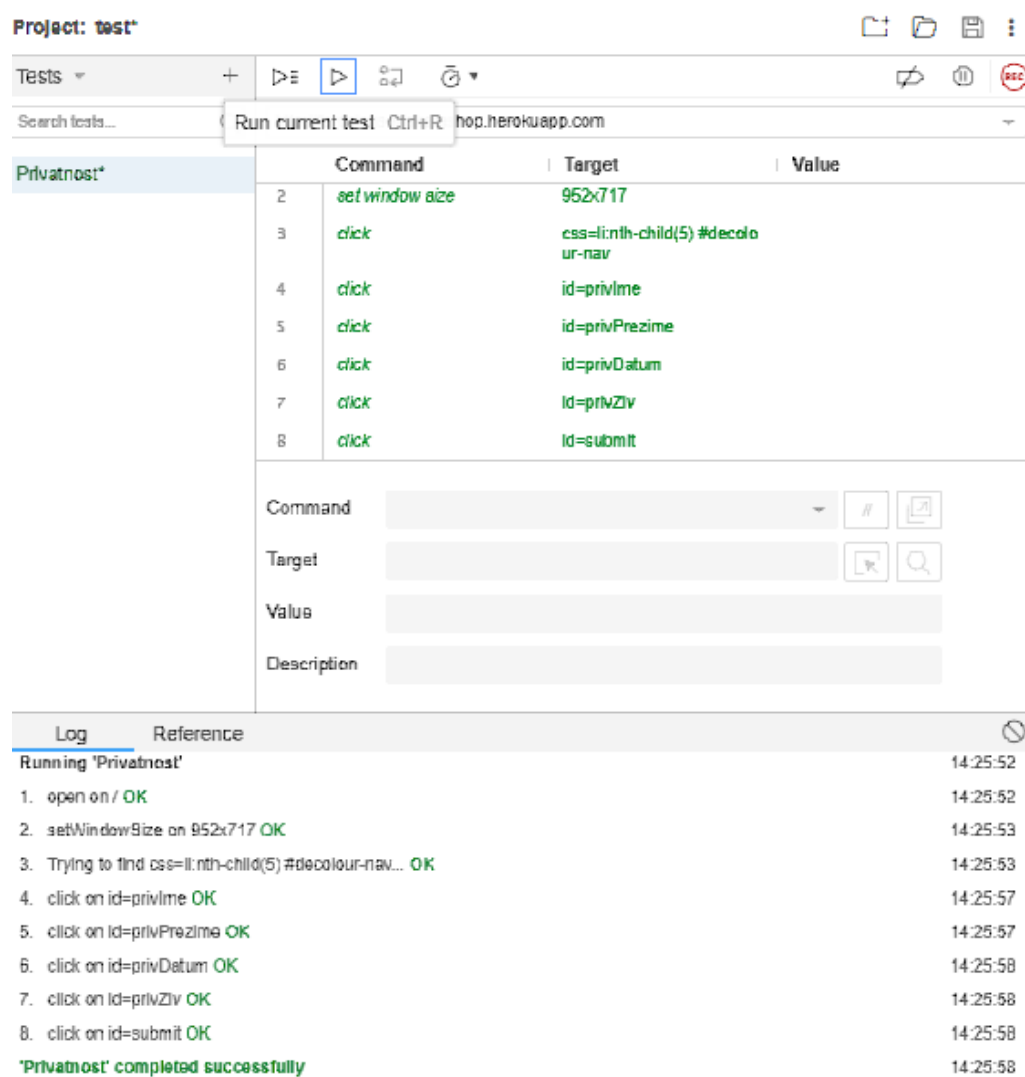
- : 1. Korisnik treba biti prijavljen
- 2. Odabir na gumb uredi
- 3. Odabir postavki vidljivosti

- 4. Pritisak na gumb spremi promjene

Ocekivani rezultat :

- : 1.a Korisnik će biti preusmjeren na stranicu postavka profila
- 1.b Nakon odabira postavka privatnosti i pristika na gumb korisnik će biti preusmjeren na stranicu sa svojim podacima.

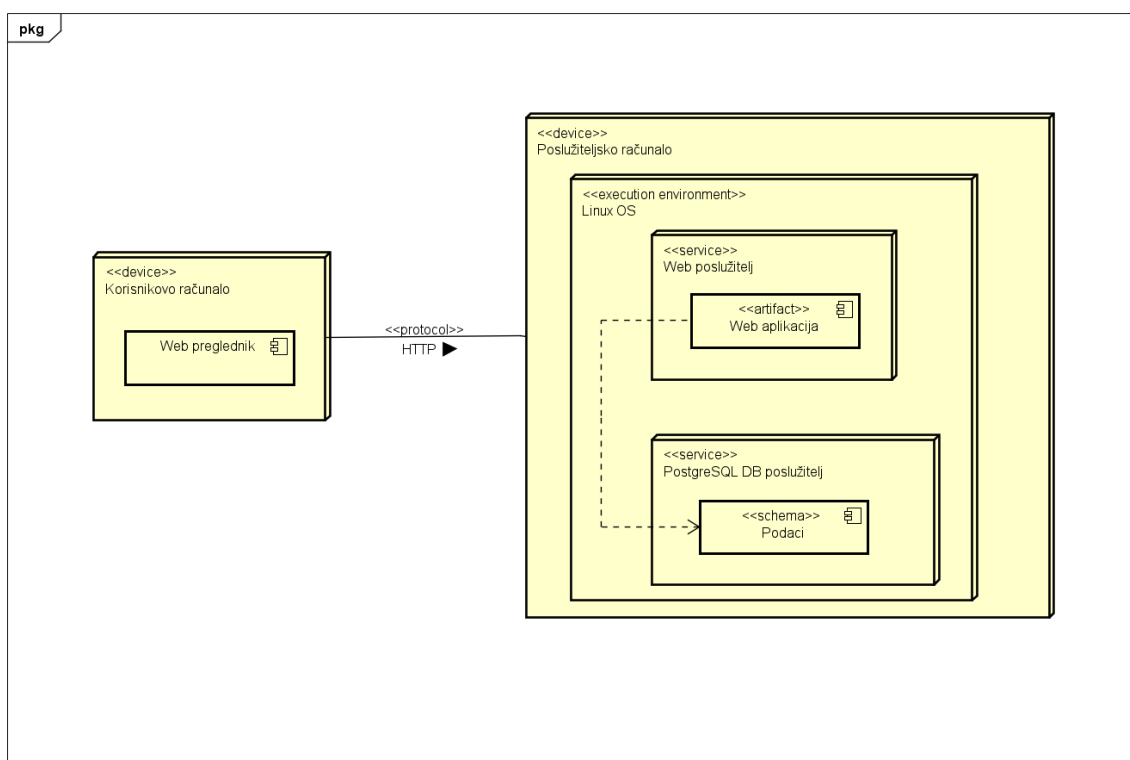
Rezultati : Očekivani rezultat 1.b je zadovoljen. Aplikacija je prošla test.



Slika 5.7: Ispitni test 5

5.3 Dijagram razmještaja

Dijagram razmještaja je strukturni statički UML dijagram koji opisuje topologiju sustava i usredotočen je na odnos sklopovskih i programskih dijelova. Postoji više vrsta dijagrama razmještaja, a ovdje se koristi specifikacijski dijagram razmještaja kako bi se prije navedeni odnos prikazao. Za ostvarenje ove aplikacije se koristi arhitektura "klijent - poslužitelj", a komunikacija se odvija protokolom HTTP. Na klijentskoj strani se nalazi web preglednik pomoću kojeg korisnik pristupa poslužitelju, odnosno šalje HTTP zahtjeve i prima HTTP odgovore. Poslužiteljska strana je nešto složenija te se ona sastoji od web poslužitelja na koji pristižu HTTP zahtjevi i poslužitelja baze podataka kojem web poslužitelj pristupa i iz kojeg uzima, uređuje ili sprema podatke.



Slika 5.8: Dijagram razmještaja

5.4 Upute za puštanje u pogon

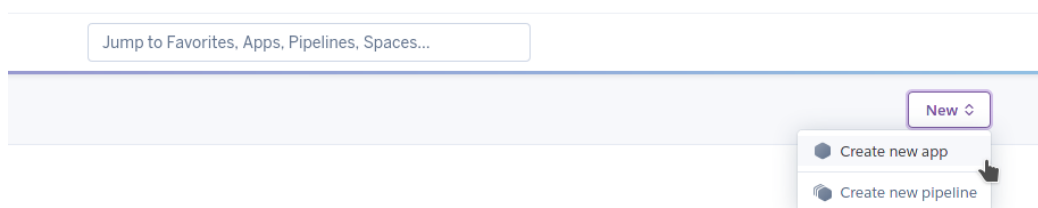
Heroku

Aplikacija je namjenjena puštanju u pogon preko platforme Heroku. Heroku je servis koji nudi usluge posluživanja aplikacija u više programskih jezika, među kojima su Ruby, Java, Node.js, Scala, Clojure, Python, PHP i Go, te usluge posluživanja baza podataka. Heroku je izabran za ovaj projekt zbog niske cijene (besplatan za stranicu ovog opsega) i jednostavnosti korištenja. Pretpostavit ćemo da korisnik na računalu ima instaliran Git te da je aplikacija preuzeta na računalo.

Stvaranje projekta

Potrebno je stvoriti račun na www.heroku.hr te se prijaviti. Stranica će nas preusmjeriti na *dashboard*, odakle stvaramo novi projekt klikom na gumb "New" - "Create new app" u gornjem desnom kutu grafičkog sučelja (Slika 5.9). Pokazat će nam se forma za unos imena aplikacije i regije na kojoj će biti puštena u pogon (Slika 5.10). Nakon toga smo preusmjereni na stranicu novokreiranog projekta.

Napomena: Ime aplikacije mora biti jedinstveno na razini čitave platforme.



Slika 5.9: Nova aplikacija

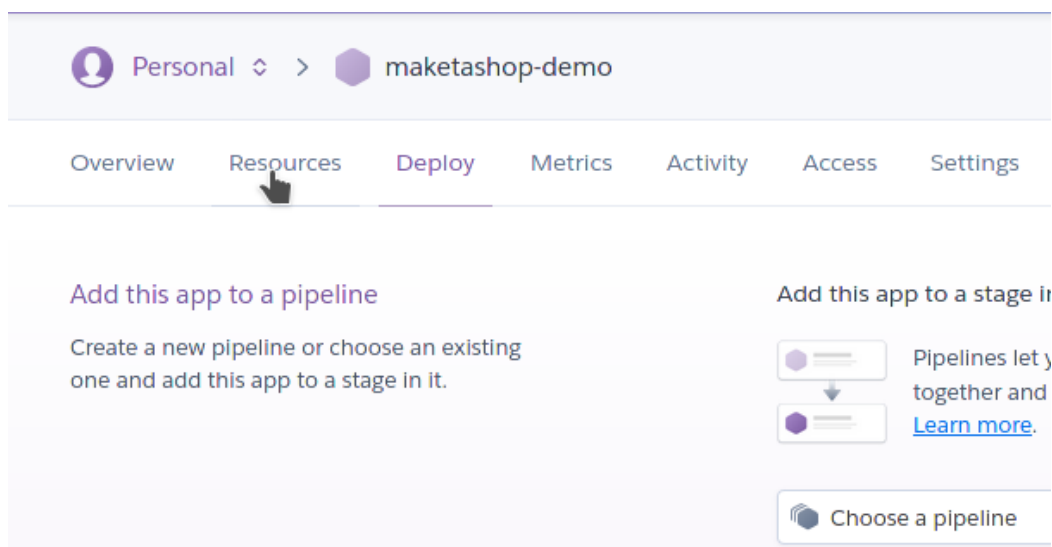
A screenshot of the Heroku "Create new app" form. The form has two main sections. The first section is titled "App name" and contains a text input field with the value "maketashop-demo". Below the input field, there is a green message that says "maketashop-demo is available". The second section is titled "Choose a region" and contains a dropdown menu with the value "Europe" selected. The dropdown menu shows a blue square icon with a white 'G' and the text "Europe".

Slika 5.10: Ime aplikacije

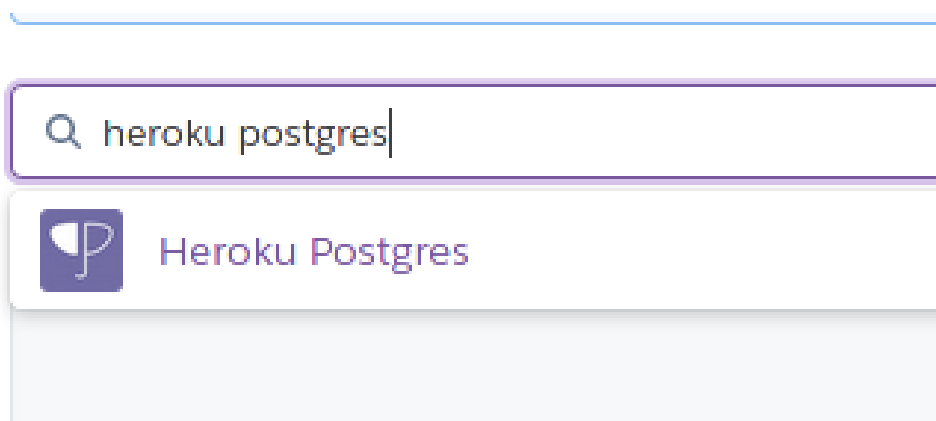
Stvaranje baze podataka

Heroku pruža uslugu posluživanje PostgreSQL baze podataka. Kako bismo ju omogućili, na stranici projekta kliknemo na "Resources" karticu (Slika 5.11). U traci za pretraživanje poglavlja "Add-ons" upišemo pojam "Heroku Postgres" i odaberemo prvi rezultat (Slika 5.12). Odaberemo plan pri vrhu sučelja, te kliknemo "Submit Order Form" (Slika 5.13).

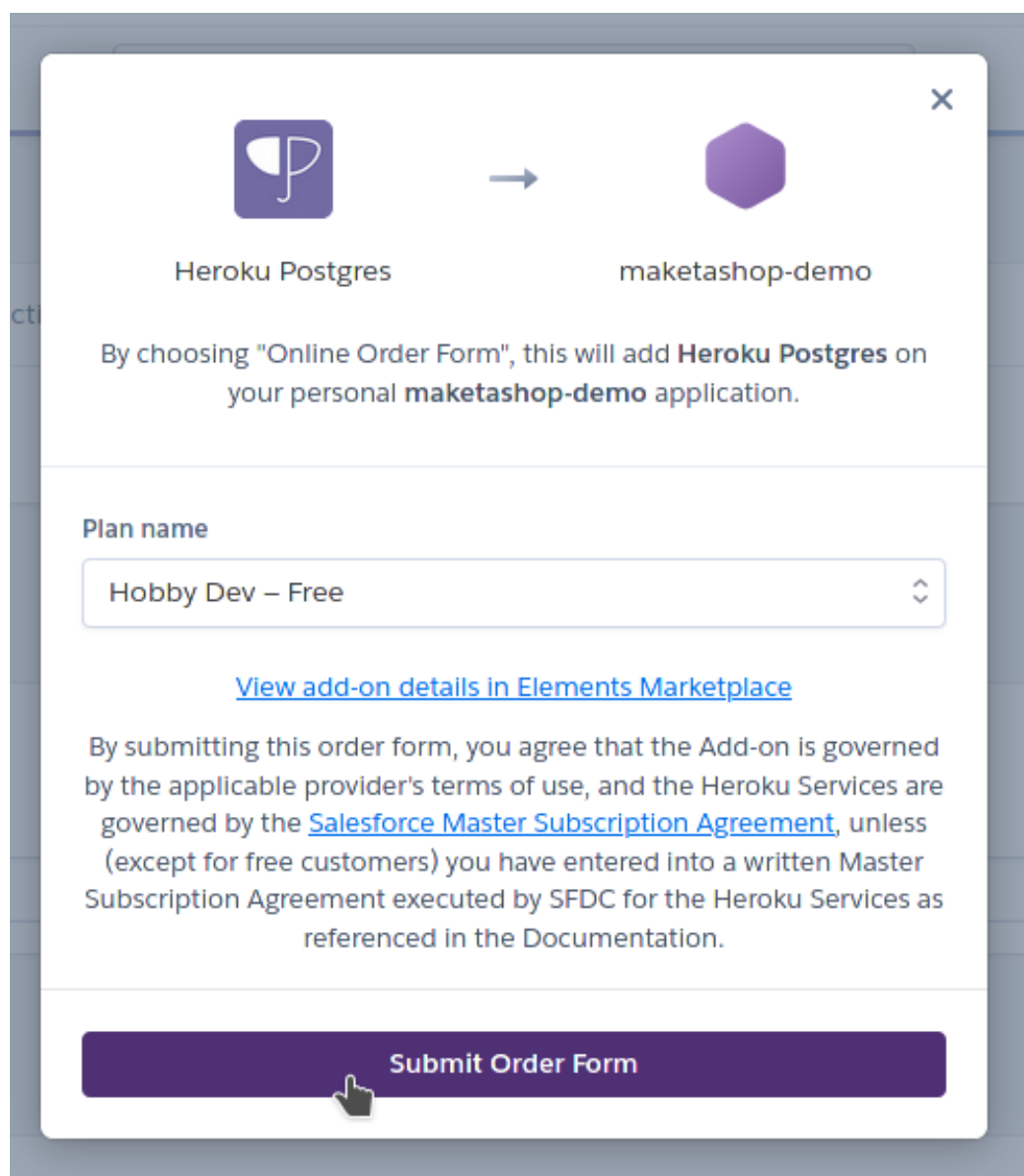
Sada navigiramo na "Settings" karticu (Slika 5.14) i nađemo poglavlje "Config Vars". Kliknemo gumb "Reveal Config Vars" za prikaz varijabli okruženja te radimo dvije stvari. Prvo se uvjerimo da postoji varijabla "DATABASE_URL" koja je nastala kada smo stvorili PostgreSQL bazu. Nakon toga u polje "KEY" upisujemo pojam "SECRET" (bez navodnika) i u polje "VALUE" upisujemo bilo kakav niz znakova. On će služiti za enkripciju poruka koje aplikacija razmjenjuje. Unos potvrdimo klikom na gumb "Add". Sučelje bi trebalo izgledati slično slici ??.



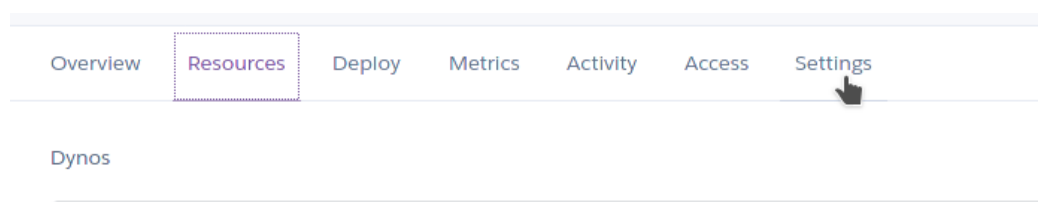
Slika 5.11: Kartica "Resources"



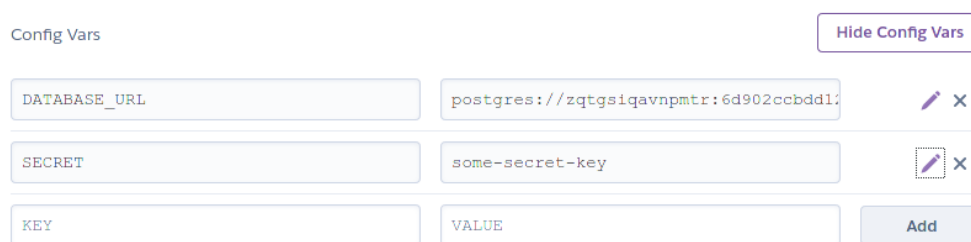
Slika 5.12: Pretraživanje "Heroku Postgres"



Slika 5.13: "Submit Order Form"



Slika 5.14: "Settings" kartica

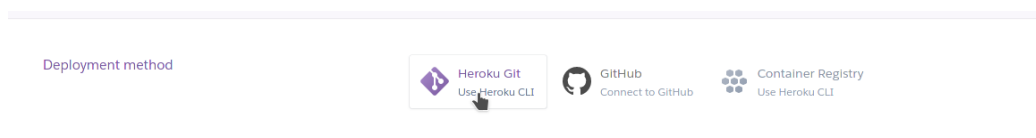


Slika 5.15: Config Vars

Heroku CLI

Heroku CLI (*Command Line Interface*) nam omogućuje pristup platformi Heroku iz komandne linije. Navigiramo nazad na karticu "Deploy". Kliknemo na poveznicu "Deployment method" - "Heroku Git" (Slika ??) te pratimo uputstva za instalaciju ovisno o operacijskom sustavu koji koristimo.

Po završetku instalacije otvorimo komandnu liniju *cmd*, *terminal* i unesemo naredbu `heroku login`, nakon čega pratimo uputstva za prijavu.



Slika 5.16: Heroku CLI

Spajanje Git repozitorija i Heroku

U komandnoj liniji se pozicioniramo na direktorij aplikacije (*naredba cd*). Za idući korak je bitno da imamo inicijaliziran Git repozitorij u direktoriju. To možemo provjeriti naredbom `git status`. Ukoliko je ispis `fatal: not a git repository...`, inicijaliziramo ga naredbom `git init`. Nakon toga dodajemo Heroku udaljeni repozitorij pomoću `heroku git:remote -a <ime-aplikacije>`, gdje *ime-aplikacije* zamijenimo imenom koji smo odabrali pri kreaciji projekta.

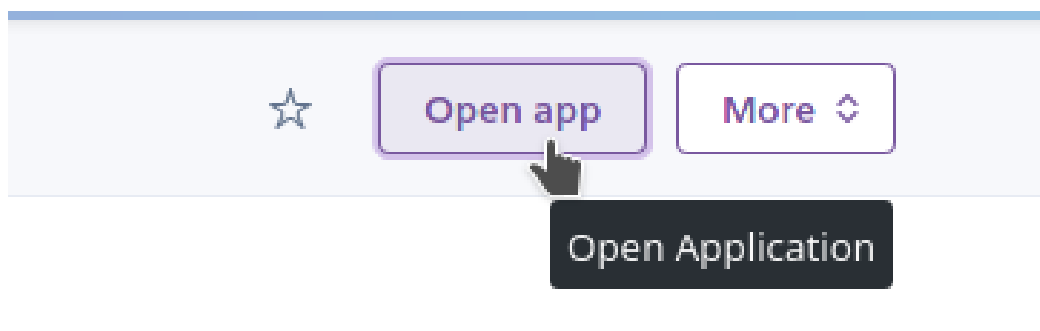
Sada *pushamo* projekt na Heroku udaljeni repozitorij slijedećim nizom naredbi:

```
git add .
git commit -am "Commit"
git push heroku master
```

Ovo može potrajati oko 1 minute. Nakon toga Heroku preuzima stvar i puštanje u pogon se automatizira.

Pristup stranici

Vraćamo se na Heroku web-stranicu. Jedino što preostaje je u gornjem desnom kutu *dashboarda* kliknuti na "Open App" što će nas preusmjeriti na aplikaciju (Slika 5.17).



Slika 5.17: Otvori aplikaciju

6. Zaključak i budući rad

Projektni zadatak naše grupe sastojao se od izrade web aplikacije za omogućavanje prodaje maketa, koja je trebala omogućiti stvaranje profila te zatraživanje izrade prilagođenih maketa uz posebne cijene. Prvi dio izrade projekta sastojao se od formiranja grupe, dobivanja zadatka, dokumentacije i implementacije generičkih funkcionalnosti. U prvom dijelu, bilo je važno izraditi specifikacije zadatka pomoću obrazaca i dijagrama, što je dalo okvir za daljnji razvoj projekta. Također, trebali smo odrediti koje će se tehnologije koristiti i upoznati se s njima. Osnovna podjela posla bila je u podtimovima za backend i frontend, te raspodijeliti dijelove dokumentacije za svakog člana.

U drugom dijelu, kroz drugi ciklus semestra, fokus je bio na implementaciji ostalih funkcionalnosti aplikacije, što smo radili samostalno ili u podtimovima, uz pisanje ostatka dokumentacije i testiranje projekta. Za rad na projektu posebno važna je bila organizacija i komunikacija u timu, te podjela posla i upravljanje vremenom. Komunikacija je ostvarena putem Whatsapp, gdje smo se međusobno obaviještavali o napretku i problemima sa zadacima koje treba obaviti. Sudjelovanje na ovom projektu bilo je vrijedno iskustvo, korisno za razvoj vještina rada u timu, vremenske organiziranosti i koordinacije zadataka, kao i usvajanje novih tehnologija potrebnih za izradu aplikacije. Stečene vještine bit će veoma korisne na ostatku studija i u budućem radu.

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

Popis literature

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	Početna stranica	5
2.2	Prijava u sustav	6
2.3	Registracija	6
3.1	Dijagram obrasca uporabe, funkcionalnost neregistriranog korisnika	19
3.2	Dijagram obrasca uporabe, funkcionalnost prijavljenog korisnika . .	20
3.3	Dijagram obrasca uporabe, funkcionalnost administratora	21
3.4	Sekvencijski dijagram, UC5 - Predlaganje priče i UC6 - Prilaganje slike / teksta / videa priči	22
3.5	Sekvencijski dijagram, UC7 - Odobrenje/odbijanje priče	23
3.6	Sekvencijski dijagram, UC20 - Popunjavanje formulara za maketu po narudžbi	24
4.1	Arhitektura sustava	26
4.2	E-R dijagram baze podataka	33
4.3	Dijagram razreda	34
4.4	Dijagram razreda - Baza podataka	35
4.5	Dijagram stanja	36
4.6	Dijagram aktivnosti za predlaganje, pregled i komentiranje priča . .	37
4.7	Dijagram aktivnosti za kupovinu i narudžbu maketa	38
4.8	Dijagram komponenti	39
5.1	Ispitivanje jedinica	41
5.2	Rezultat ispitivanja jedinica	41
5.3	Ispitni test 1	42
5.4	Ispitni test 2	43
5.5	Ispitni test 3	44
5.6	Ispitni test 4	45
5.7	Ispitni test 5	46
5.8	Dijagram razmještaja	47
5.9	Nova aplikacija	48

5.10 Ime aplikacije	48
5.11 Kartica "Resources"	49
5.12 Pretraživanje "Heroku Postgres"	50
5.13 "Submit Order Form"	51
5.14 "Settings" kartica	51
5.15 Config Vars	52
5.16 Heroku CLI	52
5.17 Otvori aplikaciju	53
6.1 Dijagrami pregleda promjena, 1. dio	63
6.2 Dijagrami pregleda promjena, 2. dio	63
6.3 Dijagrami pregleda promjena, 3. dio	63

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 14. listopada 2020.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević, M.Piskur, T.Rončević
- Teme sastanka:
 - odabir objektno orijentiranog programskog jezika (Python 3)
 - odabir web frameworka (Flask)
 - rasprava vezana uz opće probleme izvođenja zadatka
 - postavljanje GitLab repozitorija
 - dogovor o strukturiranju repozitorija te interna pravila vezana uz commit

2. sastanak

- Datum: 21. listopada 2020.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević, M.Piskur, T.Rončević
- Teme sastanka:
 - odabir platforme za praćenje projekta i kooperaciju (Miro)
 - diskusija vezana uz interna pravila te način praćenja unutar alata
 - gruba podjela poslova na back-end i front-end
 - odabir template engine-a u okviru Flaska (Jinja2)
 - odabir DB toolkit-a u okviru Pythona (SQLAlchemy)
 - odabir register/login funkcionalnosti kao generičke funkcionalnosti koja će biti prezentirana nastavniku

3. sastanak

- Datum: 28. listopada 2020.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević, M.Piskur, T.Rončević
- Teme sastanka:

- konkretna podjela poslova u okviru grube front-end, back-end podjele na 2. sastanku
- podjela izrade dokumentacije među članovima tima
- opća rasprava vezana uz opseg te zahtjevnost pojedinih poslova
- postavljanje internih rokova u svrhu pravovremenog rješavanja kontinuiranih problema
- diskusija vezana uz poteškoće povezivanja .html stranica i aplikacije

4. sastanak

- Datum: 4. studenoga 2020.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević, M.Piskur, T.Rončević
- Teme sastanka:
 - rasprava vezana uz dijagrame razreda u Flasku
 - opća rasprava o pojedinim segmentima dokumentacije
 - dogovor vezan uz back-end implementaciju
 - opća rasprava o index stranici te već implementiranim rješenjima

5. sastanak

- Datum: 2. prosinca 2020.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević, M.Piskur, T.Rončević
- Teme sastanka:
 - rasprava vezana uz prvu demonstraciju projekta
 - dogovor o uspostavljanju internih rokova
 - rasprava vezana uz back-end funkcionalnosti

6. sastanak

- Datum: 23. prosinca 2020.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević, M.Piskur, T.Rončević
- Teme sastanka:
 - dogovor o rasporedu zadataka tijekom praznika
 - pregled trenutno implementiranih funkcionalnosti
 - rasprava o vezi između back i front-enda

7. sastanak

- Datum: 6. siječnja 2021.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević,

M.Piskur, T.Rončević

- Teme sastanka:
 - rasprava o funkcionalnosti admina
 - rasprava o checkout funkcionalnosti
 - provjera trenutno dovršenih funkcionalnosti
 - dogovor o daljnjem izvršavanju individualnih zadataka

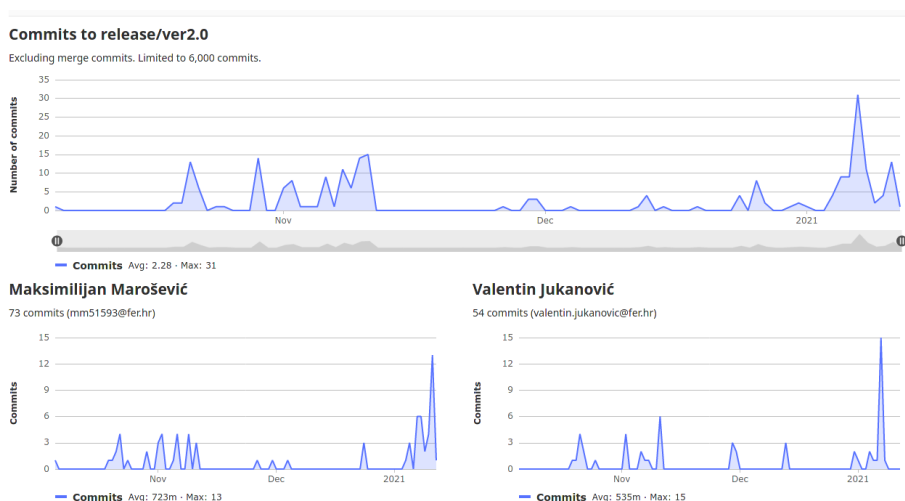
8. sastanak

- Datum: 13. siječnja 2021.
- Prisustvovali: B.Antunović, T.Brstilo, L.Cigula, V.Jukanović, M.Marošević, M.Piskur, T.Rončević
- Teme sastanka:
 - rasprava vezana uz pisanje dokumentacije
 - dodatan diskurs vezan uz implementiranje funkcionalnosti koje nedostaju

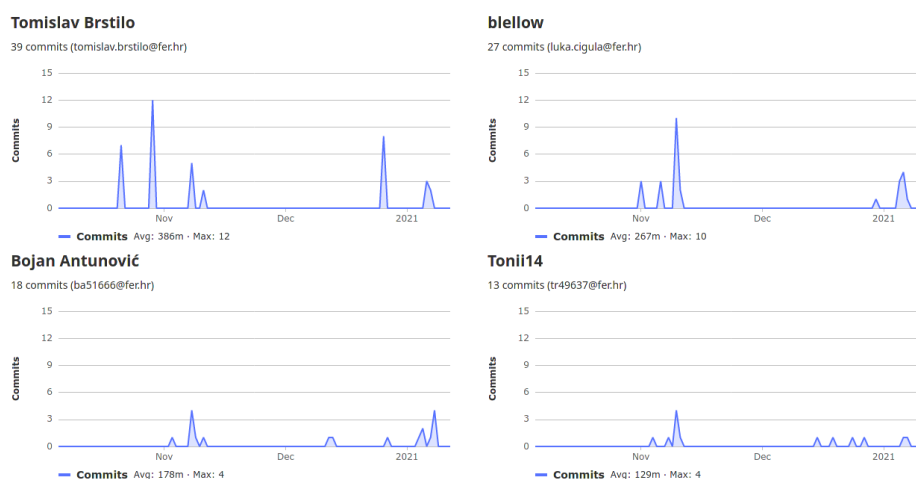
Tablica aktivnosti

	Maksimilijan Marošević	Valentin Jukanović	Bojan Antunović	Tomislav Brstilo	Mijo Piskur	Toni Rončević	Luka Cigula
Upravljanje projektom	8	8	8	8	8	8	8
Opis projektnog zadatka						2	
Funkcionalni zahtjevi		2	2	3			
Opis pojedinih obrazaca							
Dijagram obrazaca							
Sekvencijski dijagrami	3						
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava						3	
Baza podataka			3		4		
Dijagram razreda							4
Dijagram stanja		3					
Dijagram aktivnosti			2				
Dijagram komponenti					4		
Korištene tehnologije i alati						1	
Ispitivanje programskog rješenja			3			2	3
Dijagram razmještaja		2					
Upute za puštanje u pogon	2						
Dnevnik sastajanja							3
Zaključak i budući rad						2	
Popis literature							
<i>index</i>	1						3
<i>autentikacija</i>	7					1	1
<i>postavljanje priča</i>	4						1

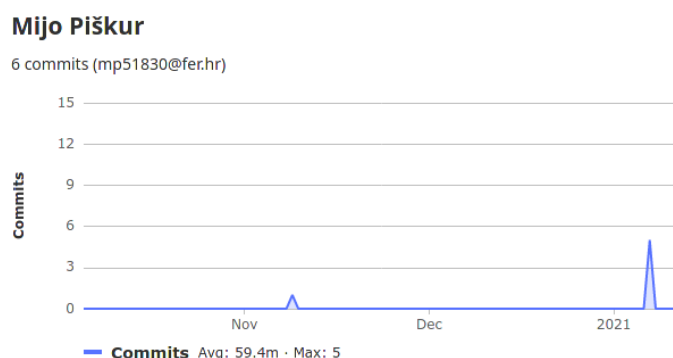
	Maksimilijan Marošević	Valentin Jukanović	Bojan Antunović	Tomislav Brstilo	Mijo Piskur	Toni Rončević	Luka Cigula
<i>shop</i>			9				
<i>korisnički profil</i>						6	
<i>baza podataka</i>			6		5		
<i>zahtjev priče</i>	4						
<i>prihvrat priče</i>							5
<i>stranica proizvoda</i>			4				
<i>prihvrat proizvoda</i>	3						
<i>zahtjev makete</i>	3						
<i>prihvrat cijene</i>	2						
<i>komentari</i>					5		
<i>košarica</i>			5				
<i>admin</i>							6
<i>checkout</i>						5	
<i>front end</i>		17		16			



Slika 6.1: Dijagrami pregleda promjena, 1. dio



Slika 6.2: Dijagrami pregleda promjena, 2. dio



Slika 6.3: Dijagrami pregleda promjena, 3. dio