

Oblikovanje programske potpore

Ak. god. 2013./2014.

Narudžba pizze

Dokumentacija, Rev. 2

Grupa: SilkRoad

Voditelj: Luka Stričić

Datum predaje: 16. 01. 2014.

Nastavnik: Nikola Bogunović

Asistent: Alan Jović

Popis članova grupe i zaduženja

Luka Strižić - Voditelj projekta, upravljanje ljudskim i projektnim resursima, opis projektnog zadatka, arhitektura, ER model, dijagram komponenti, upute za instalaciju, zaključak, dijagram razmještaja, korisničke uputa

Matej Brkašić - UC dijagrami, sekvensijski dijagrami, dijagram stanja

Belma Gutlić - Zahtjevi i ostali zahtjevi, komunikacijski dijagram, popis korištenih tehnologija

Davor Lovrić - Dodavanje referenci, superskriptiranje ključnih riječi, dijagram stanja

Matija Srbic - Arhitektura, lektoriranje opisa projektnog zadatka, dijagram razreda

Petar Šegina - Arhitektura, objektni dijagrami, implementacija, isječak programskog koda

Nika Tilić - Rječnik pojmove, dijagram aktivnosti

Sadržaj

1. Dnevnik promjena dokumentacije	4
2. Opis projektnog zadatka	5
3. Rječnik pojmove	6
4. Funkcionalni zahtjevi	7
5. Ostali zahtjevi	8
6. Arhitektura i dizajn sustava	9
6.1. Svrha, opći prioriteti i skica sustava	10
6.2. Dijagram razreda s opisom	11
6.3. Dijagram objekata	12
6.4. Ostali UML dijagrami	13
7. Implementacija i korisničko sučelje	14
7.1. Dijagram razmještaja	14
7.2. Korištene tehnologije i alati	15
7.3. Isječak programskog koda vezan za temeljnu funkcionalnost sustava	16
7.4. Ispitivanje programskog rješenja	17
7.5. Upute za instalaciju	18
7.6. Korisničke upute	19
8. Zaključak	20
9. Reference	21
Dodatak A: Dnevnik sastajanja	22

Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodataka	Autor(i)	Datum
0.10	Uređen predložak, inicijalna podjela zadataka	Strižić	19. 10. 2013.
0.11	Opis zadatka	Strižić	27. 10. 2013.
0.20	Opisani dionici i aktori u funkcionalnim zahtjevima	Gutlić	29. 10. 2013.
0.21	Dodani svi opisi obrazaca uporabe	Gutlić	03. 11. 2013.
0.30	Uređivanje rječnika pojmova	Tilić	10. 11. 2013.
0.40	Opis arhitekture sustava	Strižić	19. 11. 2013.
0.41	Dopuna opisa zadatka	Srbić	23. 11. 2013.
0.50	Dijagram razreda	Srbić	23. 11. 2013.
0.51	Izmjena dijagrama razreda	Srbić	25. 11. 2013.
0.60	ER model	Strižić	26. 11. 2013.
0.70	Izmjena i dopuna opisa obrazaca uporabe, dijagrami obrazaca uporabe, sekvencijski dijagrami, opisi sekvencijskih dijagrama	Brkašić	26. 11. 2013.
0.80	Napisani ostali zahtjevi	Gutlić	27. 11. 2013.
0.81	Popis referenci	Lovrić	27. 11. 2013.
0.82	Ispravak ER modela, opisana arhitektura	Strižić	27. 11. 2013.
0.90	Uređivanje rječnika pojmova	Tilić	28. 11. 2013.
1.0	Fork GDocs -> SVN, poliranje	Strižić	28. 11. 2013.
1.1	Dodani i opisani komunikacijski dijagrami	Gutlić	27.12.2013.
1.2	Popisani korišteni alati u projektu	Gutlić	08.01.2014.
1.3	Završna verzija dijagrama razreda	Srbić	10.01.2014.
1.4	Nadopuna opisa, popravak ER dijagrama	Strižić	12.01.2014.
1.5	Izmjena sekvencijskih dijagrama	Brkašić	15.01.2014.
1.6	Dodani dijagrami aktivnosti	Tilić	15.01.2014.
1.7	Dodan dijagram komponenti i razmještaja	Strižić	16.01.2014.
2.0	Fork GDocs -> SVN, poliranje	Strižić	16.01.2014.

Opis projektnog zadatka

Cilj projektnog zadatka jest implementacija web stranice pizzerije "7". Kontraintuitivno, navedena pizzerija u svojoj ponudi neće imati isključivo pizze. Slično dot-com mjeđuriću krajem 90-tih godina, kada bi svakoj tvrtci porasla vrijednost dionica i promet čim bi otvorila .com web stranicu, i Hrvati su shvatili prednosti korištenja interneta u poslovne svrhe. Porast prometa, veća zarada, mnogobrojni korisnici, lakši i brži način kontaktiranja te pristupačnost većem broju stanovništva su samo neke od prednosti omogućavanja poslovanja preko interneta. Također, u 21. stoljeću je naručivanje hrane telefonom stvar prošlosti. Socijalna anksioznost, urbaniji i brži način života koji sve više naginje "oblaku" te sveprisutnost mreže potiču ljudi da naručuju hrani putem interneta. Svaki restoran koji drži do sebe i svoga ugleda će to omogućiti krajnjim korisnicima - korist je obostrana.

Upravo radi već navedenih razloga naručivanje hrane preko interneta mora biti veoma jednostavno, štoviše, jednostavnije nego poziv restoranu! Intuitivno, jednostavno i pametno dizajnirano sučelje i landing page uvelike pomažu na putu do tog cilja.

Na naslovnoj stranici korisnik vidi općenite informacije o restoranu poput imena vlasnika, adrese, prosječne ocjene korisnika, radno vrijeme, očekivano vrijeme dostave, minimalni iznos online narudžbe, cijenu dostave, popuste i akcije te način plaćanja.

Pizzeria 7-ica

Ulica Grada Vukovara 234a, 10000 Zagreb

Vlasnik

Janko Strižić



Radno vrijeme

Pon - Pet 7:00 - 24:00

Subotom 10:00 - 23:00

Nedjeljom ne radimo

Minimalna narudžba

35 HRK

Cijena dostave

10 HRK za narudžbe manje od 100 kuna

Prosječno vrijeme dostave

45 minuta

Načini plaćanja

- Amex
- Gotovinom

O pizzeriji

Pizzeria 7 Vam već generacijama pruža uslugu najbolje pizzerije u gradu. Vrhunski, izvorno talijanski, kuhari spremaju jela koja i bogove zasite. Navratite i iznenadite se vrhuncem kulinarstva ovog stoljeća.



U prosjeku smo ocijenjeni sa ocjenom 5,00

Jela na akciji



Bolognese

Talijanska

38,00



Margherita

Pizze

28,00

Već s jednim klikom miša korisnik ima pristup savršeno organiziranome meniju gdje može pregledati kategoriziran popis jela, istaknutu cijenu i dostupnost. Prilikom odabira jela može specificirati dodatne opcije, primjerice veličinu porcije, dodatne začine, umake, itd. Posebno su istaknuta popularna, često naručivana jela.

Domaća



Sarma i pire krumpir

Cijena: 25,00 HRK

Dodaj u
košaricu



Vuco i janjetina

Cijena: 666,00 HRK

Dodaj u
košaricu



Brodet i žganjci

Cijena: 40,00 HRK

Dodaj u
košaricu

Kineska



Gonbao piletina

Cijena: 30,00 HRK

Dodaj u
košaricu



Osim od menija, jedan ga klik mišem dijeli i od košarice u kojoj prije narudžbe ima pregled odabranih jela, sumu cijena i mogućnost brisanja ili mijenjanja količine pojedine stavke. Dakako, tu se nalazi i tipka za naručivanje, pritiskom na koju potvrđuje trenutni odabir jela i biva odveden do stranice unosa kontakt podataka i odabira načina plaćanja. Nakon naručivanja korisnik dobije potvrdu narudžbe na unesenu e-mail adresu.

Sadržaj košarice



Vuco i janjetina

Cijena: 666,00 HRK

Količina: 2

Dodaj jedan
Makni jedan
Makni sve



Gonbao piletina

Cijena: 30,00 HRK

Količina: 1

Dodaj jedan
Makni jedan
Makni sve

Detalji narudžbe

Cijena artikala: 1362 HRK

Cijena dostave: 0 HRK

Ukupna cijena: 1362 HRK

Adresa dostave:

Kat:

Broj telefona:

e-mail:

Način plaćanja:

Nipošto se ne smije izostaviti mogućnost pregledavanja i unosa komentara na restoran i dobivenu uslugu. Stranica na kojoj se to radi je dostupna sa jednim klikom miša. Korisnik uz uneseni komentar može dati ocjenu restoranu, ili samo ocjenu, bez komentara. Smisao komentara nije tu samo za djelatnike i vlasnika, nego svi korisnici mogu vidjeti sve neobrisane komentare. Povratna informacija je bitna za poboljšanje kvalitete usluge kao i korisnikovog dojma. Ako korisnik primjeti da je savjet poslušan i da je zapravo netko pročitao njegov komentar veći su izgledi da će opet kupovati na tom mjestu.

Morate probati njihovu kinesku kuhinju. Predобра je. Recepti su imao savršeni, kao da kinezi, a ne talijani, rade u kuhinji.

★★★★★
Maja
16. siječnja 2014. 00:25

Ovo je moj restoran! I slažem sa vama ostalima, da je fenomenalan!

★★★★★
Janko
16. siječnja 2014. 00:24

Ovo je jako dobar restoran! Budem češće odavde naručivao!

★★★★★
Luka
16. siječnja 2014. 00:23

Objavi svoje mišljenje!

Ime:

Sadržaj:

Ocjena:

SAVRŠENA pizza rade! Kad sam je probao, rekao sam im da mi odmah naprave još jednu, koliko je dobra bila.

★★★★★
Matija
15. siječnja 2014. 23:24

Za razliku od korisnika koji naručuju hranu, djelatnici restorana i vlasnik će imati dodatne mogućnosti. Te su dodatne mogućnosti dostupne nakon uspješne prijave u sustav sa naslovne stranice.

Djelatnici restorana će na naslovnoj stranici vidjeti ukupni broj narudžbi za taj mjesec. Meni će moći biti uređivan. Djelatnici imaju mogućnost dodavanja ili brisanja kategorija kao i pojedinih jela, promjene dostupnosti i cijene, te vide koliko je puta koje jelo naručeno. Osim pregleda menija, djelatnik može pregledati i sve narudžbe. Svaka narudžba se sastoji od preslike košarice i osobnih podataka korisnika koji je napravio narudžbu. Tada djelatnik ima opciju preuzimanja te narudžbe ako istu nije već preuzeo neki drugi djelatnik. Preuzimanjem narudžbe se zapisuje ime tog djelatnika u narudžbu. Radi izdavanja računa ili prenošenja narudžbe dostavljajuču, djelatnik će moći prenijeti

narudžbu u definirani tekstualni oblik, spremam za ispisivanje ili spremanje na računalo.

Narudžbe

Otvorene narudžbe

16. siječnja 2014. 19:10

Adresa: Stjepana Radića 5

Kat: 3.

Telefon: 0987654321

E-mail: vuco.janjetina@vj.ooo

Način plaćanja: Amex

[Detalji narudžbe](#)

[Preuzmi narudžbu](#)
[Preuzmi kao .txt datoteku](#)

Zatvorene narudžbe

Nema preuzetih narudžbi

Vlasnik restorana ima ulogu administratora cijele web stranice. Uz sve dodatne mogućnosti koje ima djelatnik, iste ima i vlasnik, uz još neke. Na naslovnoj stranici vlasnik ima mogućnost mijenjanja svih podataka osim prosječne ocjene korisnika. Iako može obrisati komentar, ako mu se taj ne sviđa, ocjene nikako ne može obrisati niti promijeniti. Isto tako mu je vidljiv i promet po mjesecima sa podatcima o broju narudžbi, najnaručivanim jelima i ukupnom prometu za taj mjesec. Uz administratorske ovlasti dolazi i pravo pregleda podataka ostalih korisnika, u ovom slučaju djelatnika. Vlasnik može dodati nove djelatnike ili brisati stare te pregledati njihove osobne podatke i broj preuzetih narudžbi. Također, vlasnik jedini ima uvid u promet restorana, kojim se može uvjeriti koliko je dobro dizajniran sustav i koliki je porast prihoda ostvario poslovanjem preko interneta, ako ikada bude sumnja.

Postoji nekoliko web stranica za naručivanje hrane putem interneta i većina njih funkcioniра na sličan način. Razlikuju se po geografskom opsegu i broju restorana iz kojih nude naručivanje. Jedni servisi obuhvaćaju puno raznolikih restorana, dok drugi samo jedan lanac ili skup restorana po tipu kuhinje. Bitan faktor za uzeti u obzir pri isplativosti web stranice jest smještaj restorana i demografija korisnika. U urbanijim naseljima se ulog višestruko isplati zbog načina života, dok u ruralnijim ne toliko.

Najveći hrvatski servis koji nudi slične usluge je Pauza.hr. Njihovim riječima:

"Web i mobilni servis Pauza.hr omogućuje jednostavno i zabavno naručivanje hrane iz velikog broja restorana i cateringa u Zagrebu. Odlična ponuda jela, pregledni jelovnici, ocjene i preporuke korisnika - sve na jednom mjestu!"

Iz ovog kratkog opisa su već neke sličnosti i razlike očite. Najznačajnija razlika je ta da Pauza.hr obuhvaća cijeli Zagreb, odnosno bilo koji restoran koji je voljan sudjelovati, a naša web stranica samo jedan restoran, ili jedan lanac restorana, ako ih se otvoriti više. Također, Pauza.hr je prilagođena pregledu sa mobilnih uređaja (tableta i smartphonea), dok naša stranica preferira računala. Eventualna prednost naše web stranice je jednostavnost - nema zahtjeva za registracijom, dok je na većini drugih servisa registracija nužna za naručivanje i komentiranje.

Obzirom na opseg online usluga i zahtjevanih performansi u ovom projektnom zadatku, za implementaciju je odabran Django framework⁹, pisan u jeziku Python¹⁰. Django omogućuje izrađivanje elegantnih web aplikacija dobrih performansi uz minimalni vremenski ulog. Krivo je reći da je Django najfleksibilnije i najbolje rješenje po pitanju performansi, međutim naglasak je na jednostavnosti, kako korištenja tako i implementacije i administracije. Sa Django instalacijom se pakira i SQLite baza podataka¹². Samo ime sugerira da je rječ o klasičnoj SQL bazi podataka koja je i više nego dovoljna za ovakve potrebe.

Rječnik pojmova

.txt format	Format za pohranjivanje tekstualnih datoteka
Administrator	Osoba odgovorna za vođenje posla, organizaciju i sl.
Aktor	Korisnik
Browser	Web preglednik; sluzi za pretrazivanje sadrzaja na internetu
Controller	U MVC arhitekturi: komponenta koja posreduje između unosa i promjera stanja modela ili viewa
Dionik	Osoba uključena u projekt
Django framework ⁹	Python okruženje visoke razine koje služi za razvoj web aplikacija
Dot-com mjehurić	Abnormalan rast cijena dionica nastao kao posljedica popularizacije interneta krajem 90.-ih godina 20. stoljeća
Feedback	Povratna informacija korisnika
Funkcionalni zahtjev	Izjave o uslugama koje sustav mora pružati, kako će sustav reagirati na određeni ulazni poticaj, te kako bi se sustav trebao ponašati u određenim situacijama ³
Hard disk	Tvrdi disk; uređaj za pohranu digitalnih informacija
ID	identifikacijski broj
Landing page	Web stranica koja se otvara klikom na link u tražilici
Login	Proces identifikacije računalu
Loyalty program	marketinška metoda u kojoj se provodi nagrađivanje klijenata nagrađuje time i poticanje na daljnju suradnju i održavanje kontakata
Model	U MVC arhitekturi: komponenta arhitekture koja čuva podatke web aplikacije, logiku koja čini srž aplikacije i funkcije
MVC arhitektura ¹¹	Uzorak software arhitekture Model-view-controller koji odvaja prikaz informacija od interakcije korisnika sa tim informacijama
Nefunkcionalni zahtjevi	Ograničenja u uslugama i funkcijama, kao što su vremenska ogranicenja (ne)usvojeni standardi, zahtjevi kvalitete, platformski zahtjevi, ograničenja procesu razvoja i oblikovanja itd. ³
Online	Spojen na mrežu
ORM	Object relational mapper, tehnologija koja povezuje domenski sa ER modelom
Python ¹⁰	Viši programski jezik
SQLite ¹²	Relacijska baza podataka
Sudionik	Dio sustava potreban za provođenje zahtjeva
Sučelje	Područje interakcije dva sustava, npr. čovjeka i računala

TortoiseSVN	besplatan program koji se koristi za razvoj i oblikovanje programske potpore
Twitter Bootstrap	besplatan skup alata koji se koriste u web dizajnu
UTF-8	Međunarodni standard kodiranja znakova
View	U MVC arhitekturi: prikaz podataka iz modela (slika, tekst, dijagram, itd.)
Web aplikacija	Aplikacija koja koristi web preglednike kao klijente

Funkcionalni zahtjevi

Dionici:

- Klijenti
- Djelatnici pizzerije
- Vlasnik pizzerije

Aktori i njihovi funkcionalni zahtjevi:

- Klijent, inicijator:
 - Može vidjeti informacije o pizzeriji
 - Može pregledati meni i vidjeti česte narudžbe
 - Može pročitati komentare
 - Može ostaviti komentar
 - Može naručiti sve dostupno u meniju i uređivati košaricu
 - Može ocijeniti pizzeriju
 - Može upisati svoje osobne podatke prilikom narudžbe
- Djelatnik pizzerije, inicijator:
 - Može pregledati popis narudžbi i preuzeti ne preuzete
 - Može snimiti narudžbu u izlaznu datoteku
 - Može uređivati kategorije i jela
 - Vidi ukupan broj narudžbi za trenutni mjesec
- Vlasnik pizzerije, inicijator:
 - Može mijenjati sve podatke o pizzeriji
 - Ima opciju brisanja komentara
 - Vidi popis svih djelatnika i njihovih preuzetih narudžbi
 - Može dodavati/brisati djelatnike
 - Ima pristup izvještaju o prometu pizzerije
 - Sve ostale mogućnosti iste kao i djelatnik
- Baza podataka, sudionik:
 - Sadrži arhivu podataka o kategorijama, jelima
 - Pohranjuje komentare, ocjene i narudžbe
 - Čuva podatke o prometu

Opis obrazaca uporabe:

★ UC1 - *PogledajGlavnuStranicu:*

- **Glavni sudionik:** Korisnik
- **Cilj:** Pogledati sadržaj glavne stranice i primiti informacije o pizzeriji
- **Sudionici:** Baza podataka
- **Rezultat:** Primljene informacije o pizzeriji i naručivanju
- **Željeni scenarij:**
 1. Korisnik šalje upit i pristupa glavnoj stranici.
 2. Sustav vraća opće informacije o pizzeriji, meniju i načinu naručivanja.
 3. Ako je prijavljen djelatnik, daje informaciju o trenutnim narudžbama koje može zaprimiti.

★ UC2 - *PogledajMeni:*

- **Glavni sudionik:** Korisnik
- **Cilj:** Pregledati ponudu jela pizzerije i potaknuti na naručivanje
- **Sudionici:** Baza podataka
- **Rezultat:** Primljene informacije o trenutnim jelima s mogućnošću narudžbe
- **Željeni scenarij:**
 1. Korisnik otvara meni da pregleda jela koja su trenutno moguća za naručivanje
 2. Sustav vraća potrebne informacije o jelima s cijenom, označava često naručivane i daje opciju naručivanja svakog jela.

★ UC3 - *PregledajKošaru:*

- **Glavni sudionik:** Klijent
- **Cilj:** Pregledati sadržaj košarice po stawkama s istaknutim cijenama stavki
- **Sudionici:** Baza podataka

- **Rezultat:** Primljene informacije o sadržaju košarice s mogućnošću brisanja ili izmjene
- **Željeni scenarij:**
 1. Klijent otvara košaricu da pregleda njen sadržaj i cijene
 2. Sustav vraća potrebne informacije o stavkama i cijenama te daje opcije brisanja i izmjene

★ UC4 - *DodajUKošaricu:*

- **Glavni sudionik:** Klijent
- **Cilj:** Dodati u košaricu željena jela
- **Sudionici:** Baza podataka
- **Preduvjeti:** Odabir nekog od ponuđenih jela u meniju
- **Rezultat:** Dodana jela u košarici
- **Željeni scenarij:**
 1. Klijent odabire jelo (ili više njih) i zatraži stavljanje u košaricu.
 2. Sustav prima zahtjev, stavlja izabrana jela u košaricu i računa sveukupni iznos za platiti.

★ UC5 - *Naruči:*

- **Glavni sudionik:** Klijent
- **Cilj:** Naručiti željena jela
- **Sudionici:** Baza podataka
- **Preduvjeti:** Jedno ili više jela u košarici
- **Rezultat:** Narudžba poslana djelatnicima i spremljena u bazu podataka.
- **Željeni scenarij:**
 1. Nakon što je dodao jela u košaricu, klijent upisuje osobne podatke i potvrđuje svoju narudžbu.
 2. Sustav prima zahtjev, sprema narudžbu u bazu podataka zajedno sa upisanim osobnim podacima klijenta i šalje kopiju na klijentov e-mail.
 3. Klijent prima povratnu informaciju da je uspješno obavio narudžbu.

★ UC6 - *OstaviKomentar:*

- **Glavni sudionik:** Klijent
- **Cilj:** Ostaviti komentar o pizzeriji
- **Sudionici:** Baza podataka
- **Preduvjeti:** Minimalni broj slova u komentaru
- **Rezultat:** Ostavljeni komentar zapisuje se u bazu podataka
- **Željeni scenarij:**
 1. Klijent upisuje i pošalje komentar.
 2. Sustav provjerava ispunjava li komentar uvjet o minimalnom broju slova te sprema komentar u bazu podataka.

★ UC7 - *PregledajKomentare*

- **Glavni sudionik:** Korisnik
- **Cilj:** Dobiti ispis svih komentara i ocjena pizzerije
- **Sudionici:** Baza podataka
- **Preduvjeti:** Postojanje komentara
- **Rezultat:** Primljeni sadržaji svih ostavljenih komentara
- **Željeni scenarij:**
 1. Korisnik pristupa stranici sa komentarima i time šalje zahtjev sustavu za pregled komentara.
 2. Sustav pretražuje bazu podataka i vraća sadržaj svih postojećih komentara.
- **Mogući drugi scenariji:**
 2. Ne postoji trenutno niti jedan komentar u bazi podataka.
 1. Sustav obavještava korisnika o nepostojanju komentara.

★ UC8 - *OcjeniRestoran:*

- **Glavni sudionik:** Klijent
- **Cilj:** Ocijeniti restoran sa ocjenom 1-5
- **Sudionici:** Baza podataka
- **Rezultat:** Ostavljena izabrana ocjena korisnika
- **Željeni scenarij:**
 1. Klijent otvara stranicu komentara i bira koju ocjenu želi ostaviti za

pizzeriju

2. Ocjena se zapisuje u bazi podataka.

★ UC9 - *ZaprimiNarudžbu*:

- **Glavni sudionik:** Djelatnik
- **Cilj:** Odabrat i primiti narudžbu
- **Sudionici:** Baza podataka
- **Preduvjeti:** Postojanje nezaprimljene narudžbe i prijava u sustav
- **Rezultat:** Dobiveni podaci o zaprimljenoj narudžbi
- **Željeni scenarij:**
 1. Djelatnik šalje zahtjev sustavu za prikaz liste narudžbi
 2. Sustav vraća listu narudžbi
 3. Djelatnik iz prikazane liste narudžbi odabire nezaprimljenu koju želi preuzeti.
 4. Sustav označava narudžbu sa imenom djelatnika i vraća presliku košarice kljienta sa njegovim dodatnim podacima.
- **Mogući drugi scenariji:**
 2. Ne postoji trenutno niti jedna narudžba.
 1. Sustav vraća informaciju o tome da trenutno ne postoji niti jedna narudžba u bazi podataka pa nema mogućnosti odabira.

★ UC10 - *SnimiNarudžbu*:

- **Glavni sudionik:** Djelatnik
- **Cilj:** Preuzeti narudžbu u izlaznu datoteku
- **Sudionici:** Baza podataka
- **Preduvjeti:** Prijava u sustav
- **Rezultat:** Spremljena narudžba u tekstualnu datoteku na fizički disk.
- **Željeni scenarij:**
 1. Djelatnik šalje zahtjev za snimanjem narudžbe.
 2. Sustav prima zahtjev za snimanje narudžbe te šalje djelatniku izlaznu datoteku koju će spremiti na disk u predodređenom tekstualnom obliku (.pdf).

★ UC11 - *PrijavaSe*:

- **Glavni sudionik:** Djelatnik
- **Cilj:** Prijaviti se u sustav
- **Sudionici:** Baza podataka
- **Preduvjeti:** Već nije prijavljen
- **Rezultat:** Djelatnik je prijavljen u sustav i može koristiti sve funkcije sustava u svojoj ovlasti.
- **Željeni scenarij:**
 1. Djelatnik upisuje i šalje sustavu svoje podatke (korisničko ime i lozinku).
 2. Sustav provjera je li upisani podaci odgovaraju podacima u bazi podataka i daje odgovarajuće ovlasti.
- **Mogući drugi scenariji:**
 2. Podaci ne odgovaraju onima u bazi podataka, nisu pronađeni.
 1. Sustav dojavljuje djelatniku grešku o neispravnim podacima.

★ UC12 - *UrediPopisJela*:

- **Glavni sudionik:** Djelatnik
- **Cilj:** Izvršiti promjene na kategorijama ili jelima
- **Sudionici:** Baza podataka
- **Preduvjeti:** Prijava u sustav
- **Rezultat:** Izvršene izmjene na postojećim kategorijama/jelima ili dodavanje novih.
- **Željeni scenarij:**
 1. Prijavljeni djelatnik šalje sustavu zahtjev za promjenu određene kategorije/određenog jela ili da doda potpuno nove.
 2. Sustav prima i obrađuje zahtjev tako što mijenja ili dodaje nove podatke u bazi podataka te obaviještava djelatnika o uspješnosti.
- **Mogući drugi scenariji:**
 2. Već postoji kategorija ili jelo u bazi koje je djelatnik htio dodati.
 1. Sustav ne dodaje novu kategoriju ili novo jelo u bazu podataka.

2. Sustav obaviještava djelatnika o nemogućnosti dodavanja.

★ UC13 - *UrediDjelatnika*:

- **Glavni sudionik:** Vlasnik
- **Cilj:** Dodati ili obrisati djelatnika
- **Sudionici:** Baza podataka
- **Preduvjeti:** Prijava u sustav
- **Rezultat:** Dodavanje novog računa djelatnika ili brisanje postojećeg
- **Željeni scenarij:**
 1. Prijavljeni vlasnik upisuje osobne podatke o novom djelatniku i šalje zahtjev za dodavanjem.
 2. Sustav dodaje novog djelatnika u bazu podataka i obaviještava vlasnika o uspješnosti radnje.
- **Drugi mogući scenariji:**
 2. Djelatnik s navedenim korisničkim imenom već postoji u bazi.
 1. Sustav ne dodaje novog djelatnika u bazu podataka.
 2. Sustav obaviještava vlasnika o nemogućnosti dodavanja.

ili:

1. Prijavljeni vlasnik šalje zahtjev za brisanje postojećeg djelatnika te unosi njegovo korisničko ime.
2. Sustav briše odabranog djelatnika iz baze podataka i obaviještava vlasnika o uspješnosti radnje.

★ UC14 - *PogledajIzvještaj*:

- **Glavni sudionik:** Vlasnik
- **Cilj:** Pogledati izvještaj o prometu pizzerije
- **Sudionici:** Baza podataka
- **Preduvjeti:** Prijava u sustav
- **Rezultat:** Dobiven mjesecni izvještaj o prometu
- **Željeni scenarij:**
 1. Prijavljeni vlasnik šalje zahtjev za pregledom mjesecnog izvještaja.
 2. Sustav prima zahtjev i vraća za svaki mjesec: broj narudžbi, proječnu

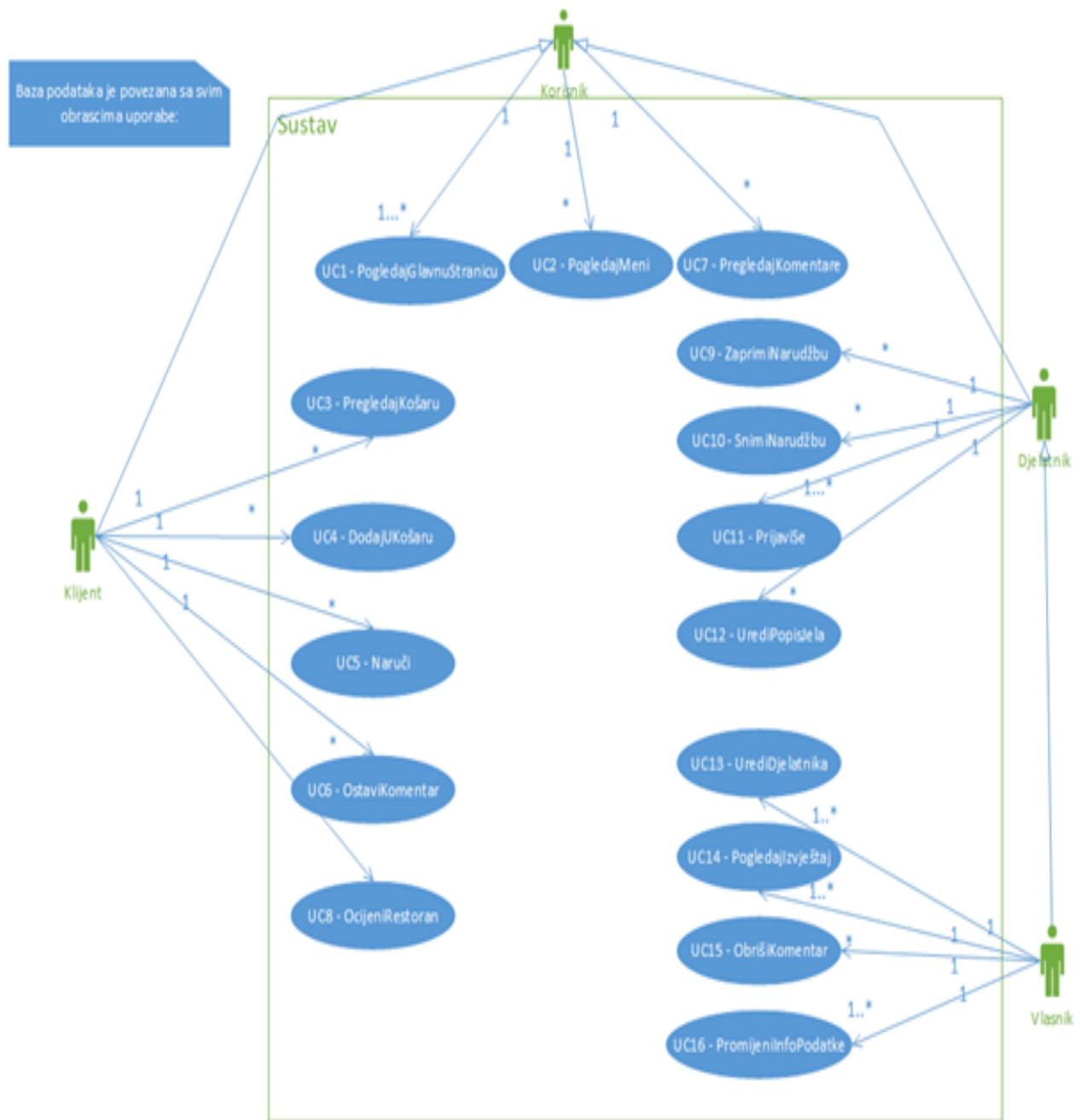
cijenu narudžbe, tri najčešće naručivana jela i ukupni promet pizzerije.

★ UC15 - *ObrišiKomentar:*

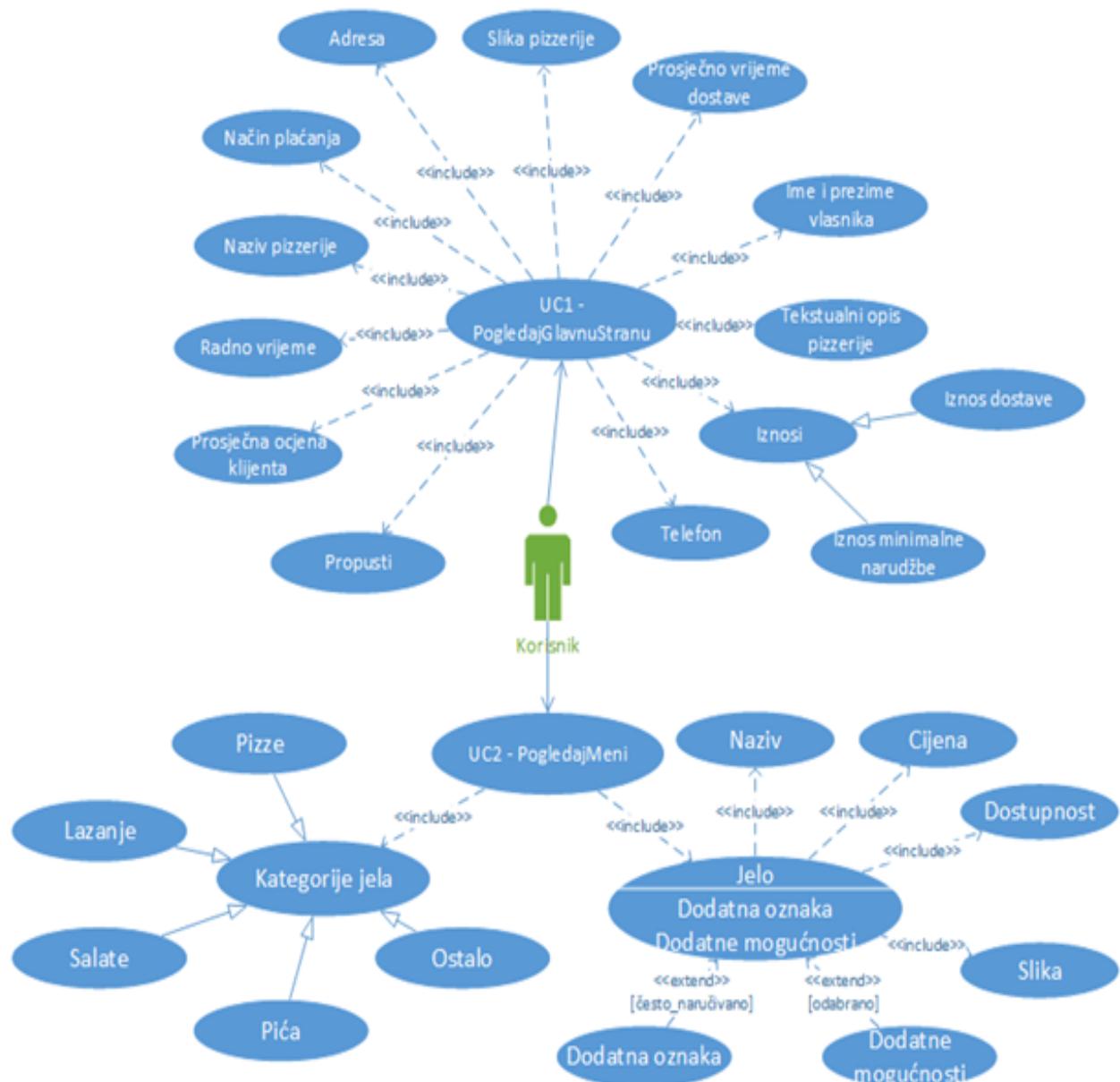
- **Glavni sudionik:** Vlasnik
- **Cilj:** Obrisati komentar o pizzeriji
- **Sudionici:** Baza podataka
- **Preduvjeti:** Prijava u sustav i postojanje komentara
- **Rezultat:** Obrisani komentar koji se ne može vratiti
- **Željeni scenarij:**
 1. Prijavljeni vlasnik šalje zahtjev za brisanjem određenog komentara iz liste.
 2. Sustav prima zahtjev i potpuno briše željeni komentar iz baze podataka.

★ UC16 - *PromjeniInfoPodatke:*

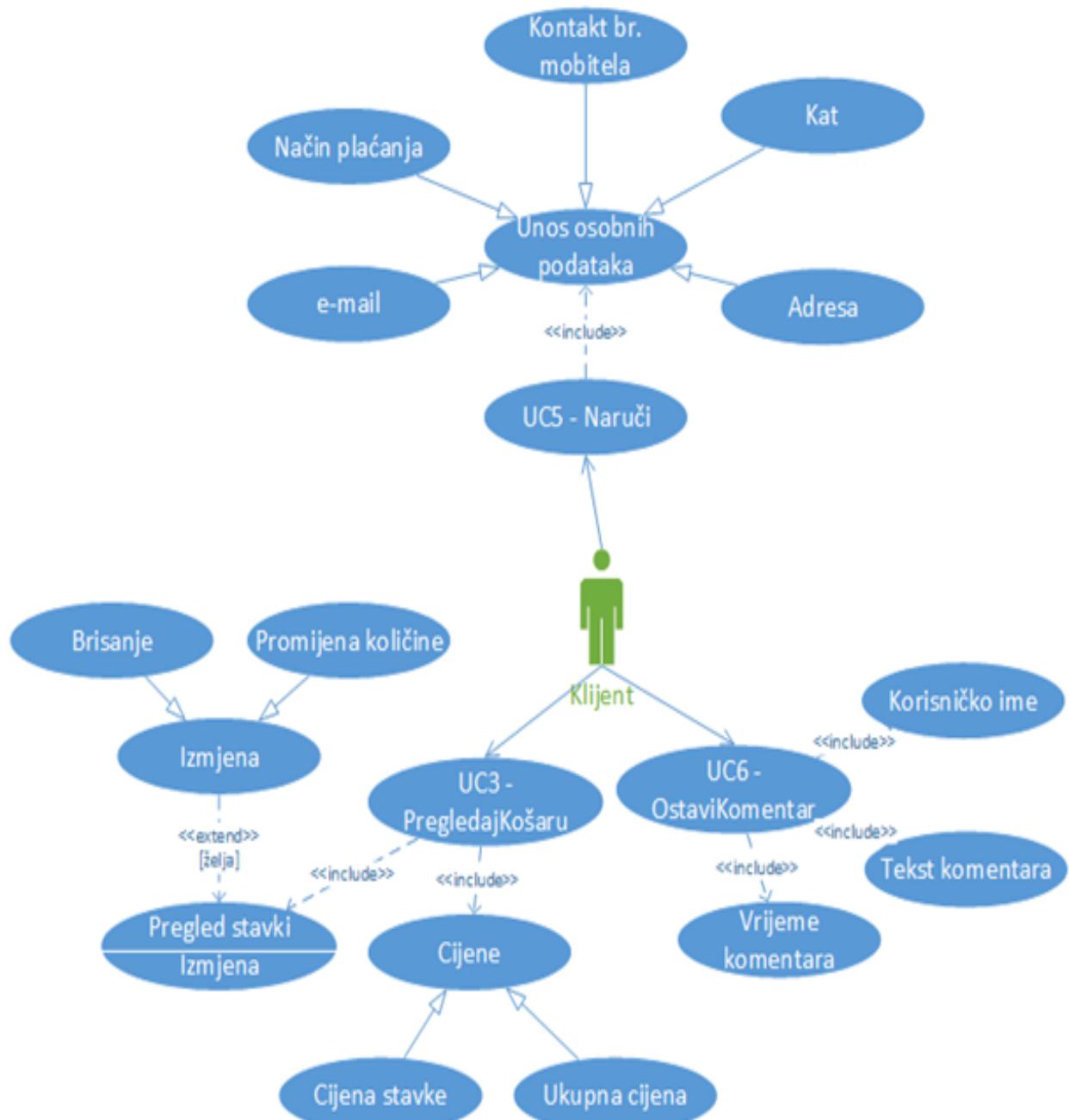
- **Glavni sudionik:** Vlasnik
- **Cilj:** Promijeniti informacije o pizzeriji na glavnoj stranici
- **Sudionici:** Baza podataka
- **Preduvjeti:** Prijava u sustav
- **Rezultat:** Promijenjeni podaci o pizzeriji
- **Željeni scenarij:**
 1. Prijavljeni vlasnik šalje zahtjev za promjenu podataka o pizzeriji na glavnoj stranici.
 2. Sustav prima zahtjev o promjeni podataka i obrađuje ga.



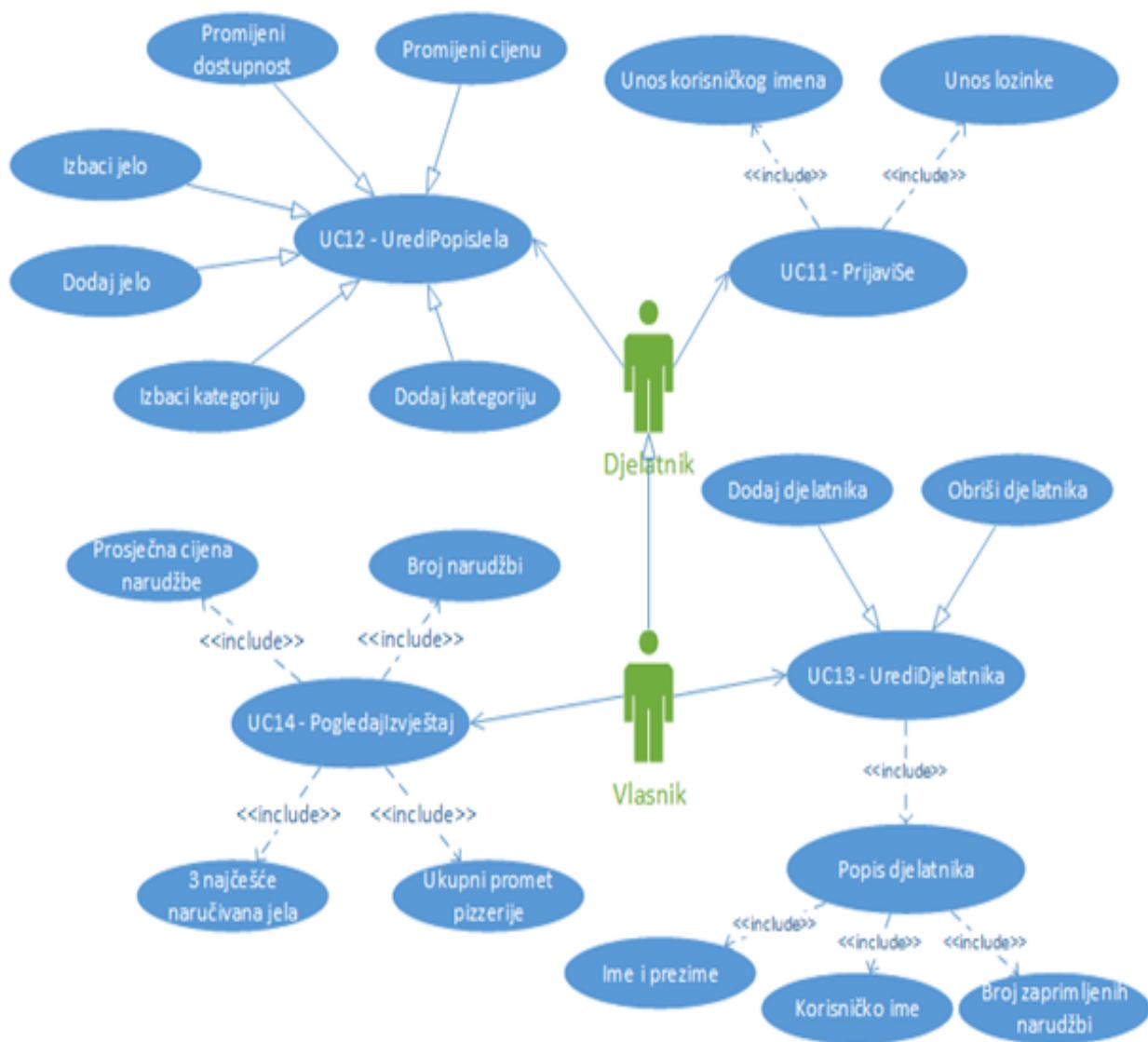
Slika 4.1. Dijagram obrazaca uporabe za UC1 do UC16



Slika 4.2. Detaljniji prikaz obrazaca UC1 i UC2



Slika 4.3. Detaljniji prikaz obrazaca UC3, UC5 i UC6

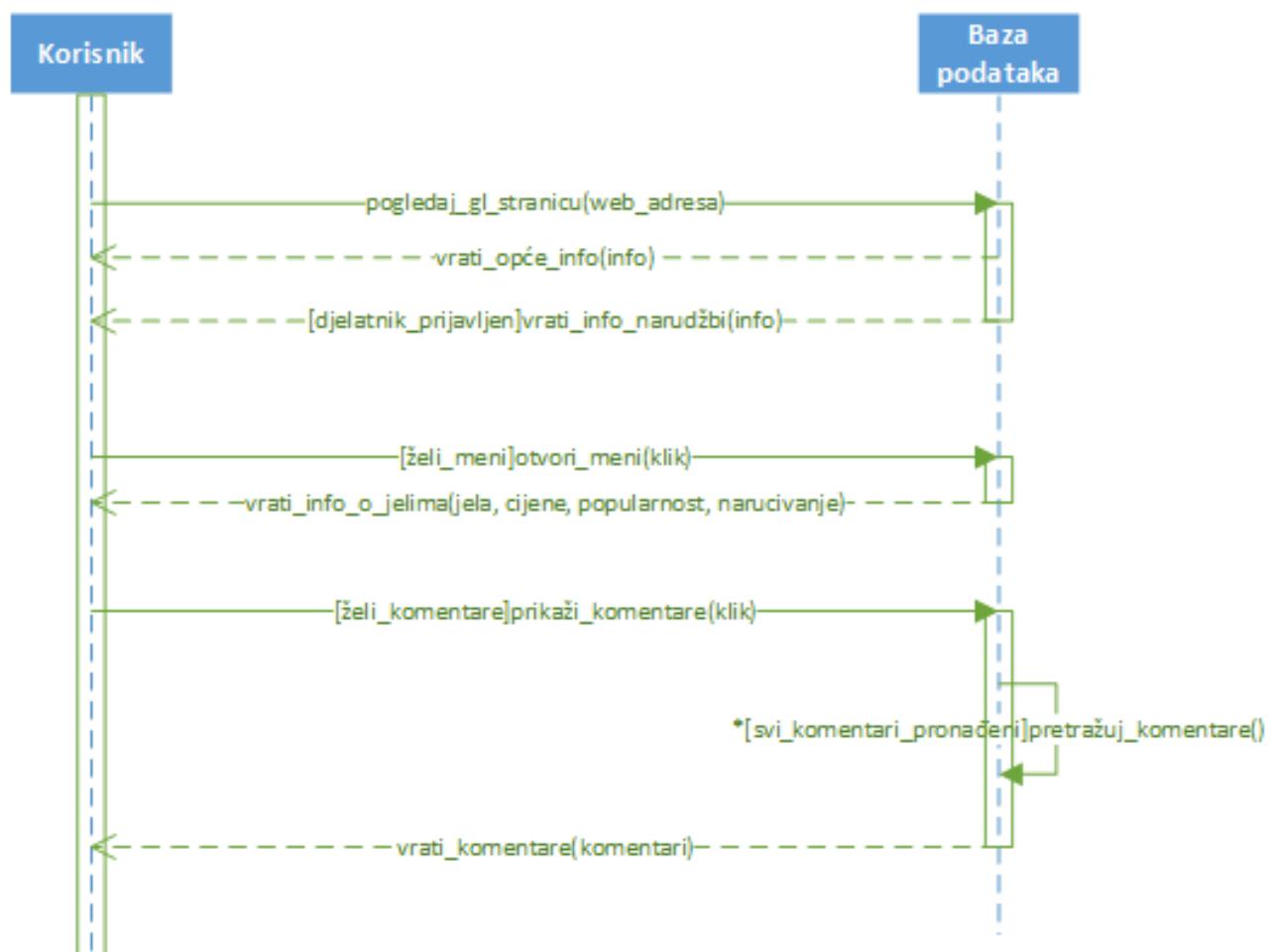


Slika 4.4. Detaljniji prikaz obrazaca UC11, UC12, UC13 i UC14

Sekvencijski dijagrami:

Obrasci uporabe UC1, UC2 i UC7:

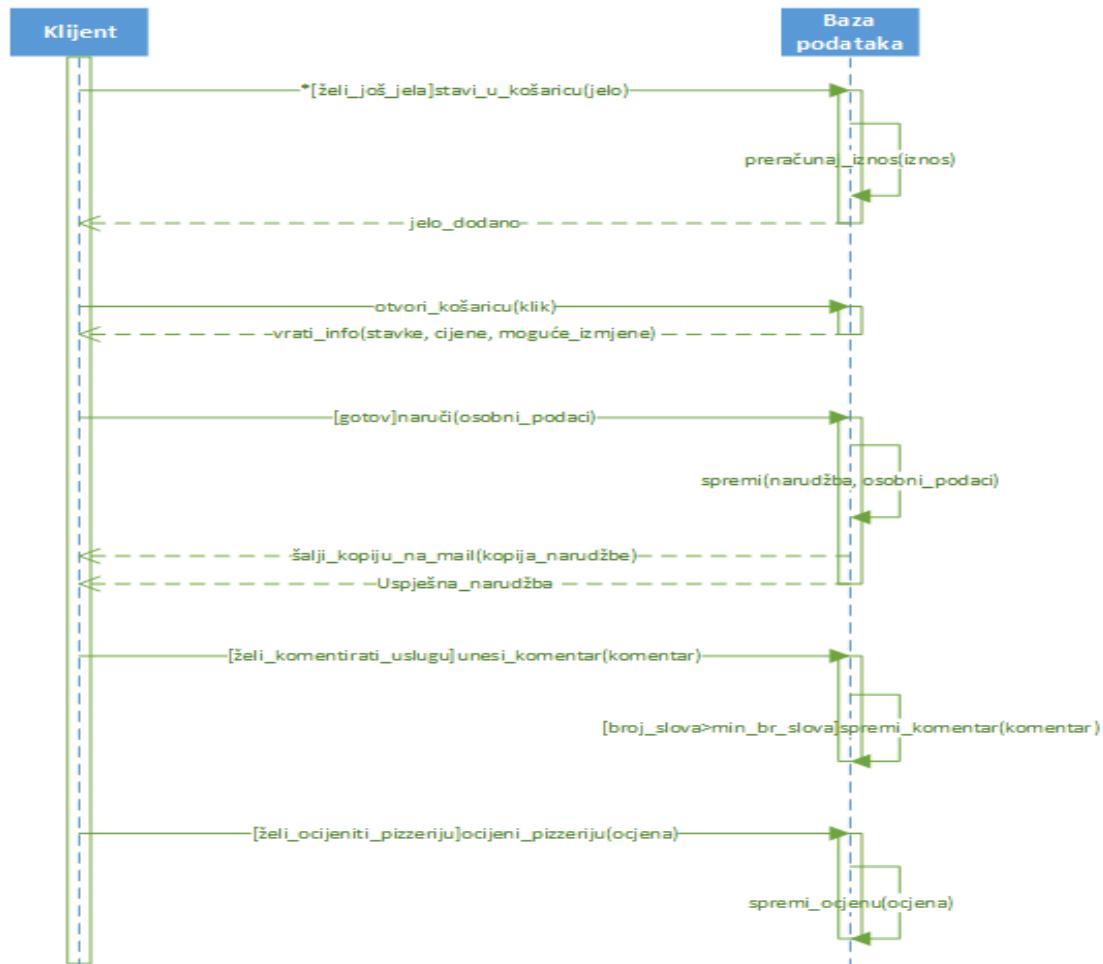
Korisnik započinje pregled glavne stranice unosom njene web adrese. Baza podataka vraća opće informacije o glavnoj stranici, a ako je prijavljen djelatnik vraća i informacije o narudžbama. Ukoliko korisnik želi pogledati meni, jednim klikom ga otvara. Baza podataka vraća informacije o jelima te opcije naručivanja. Ukoliko korisnik želi pogledati komentare, opet jednim klikom zatražuje prikaz komentara. Baza podataka pretražuje komentare dok svi nisu pronađeni te vraća komentare korisniku.



Slika 4.5. Sekvencijski dijagram obrazaca uporabe UC1, UC2 i UC7

Obrasci uporabe UC3, UC4, UC5, UC6 i UC8:

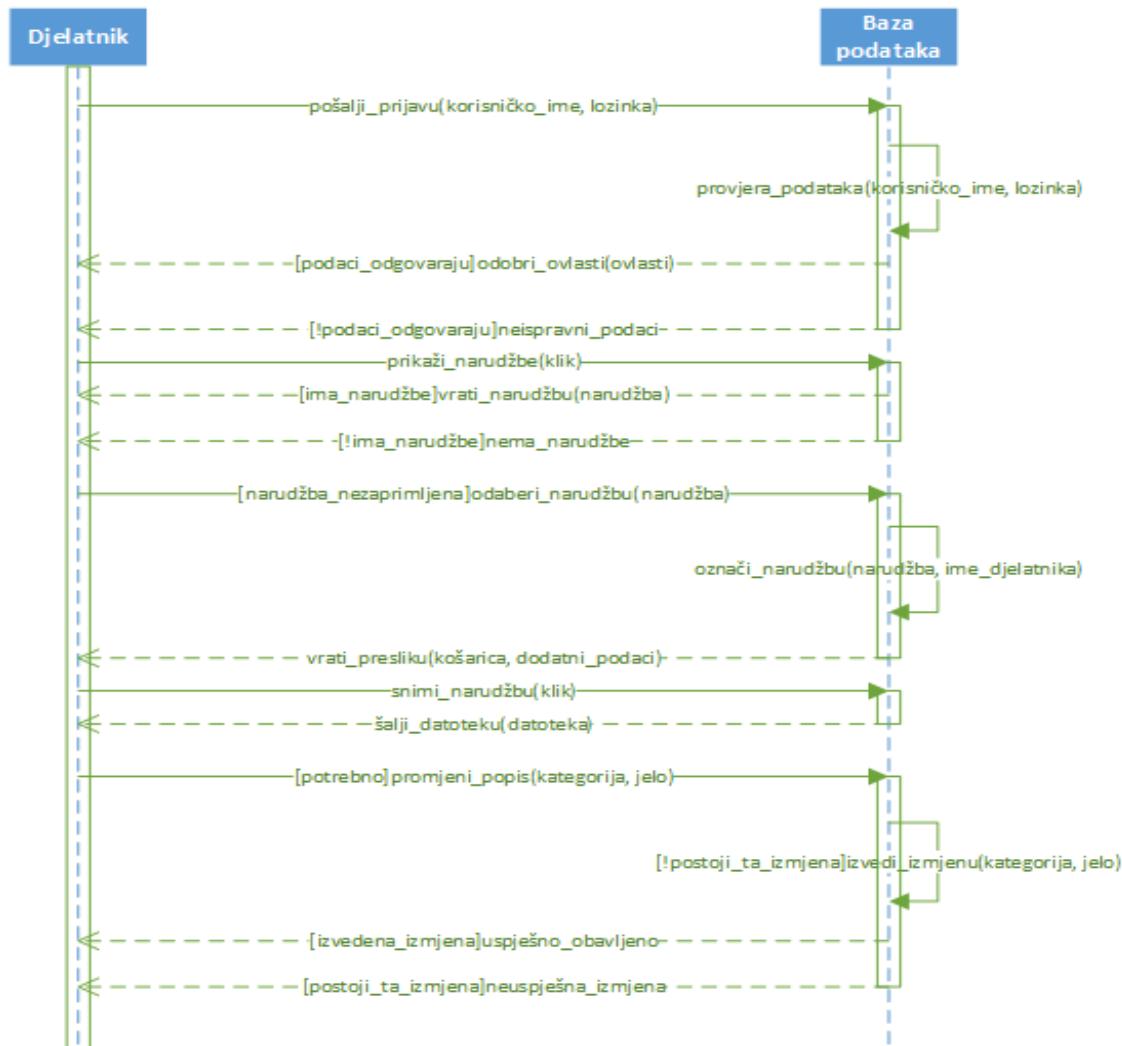
Klijent stavlja koliko god želi jela u svoju košaricu. Za svako jelo, baza podataka, preračunava ukupan iznos te obavještava klijenta da je jelo dodano. Kada klijent želi pregledati svoj odabir jela, otvara košaricu. Baza podataka vraća sve informacije o odabranim jelima s mogućnostima izmjene. Kada je klijent gotov s odabirom jela, odlučuje se za naručivanje te šalje osobne podatke. Baza podataka sprema klijentovu narudžbu i osobne podatke, šalje kopiju narudžbe na klijentovu e-mail adresu te ga obavještava o uspješnosti narudžbe. Nakon naručivanja, ako želi, klijent može komentirati uslugu unosom komentara. Baza podataka provjerava da li komentar zadovoljava uvjet minimalnog broja slova te, ako da, sprema komentar. Nakon toga klijent, ako želi, ocijenjuje pizzeriju unosom ocjene koja se sprema u bazu podataka.



Slika 4.6. Sekvencijski dijagram obrazaca uporabe UC3, UC4, UC5, UC6 i UC8

Obrasci uporabe UC9, UC10, UC11 i UC12:

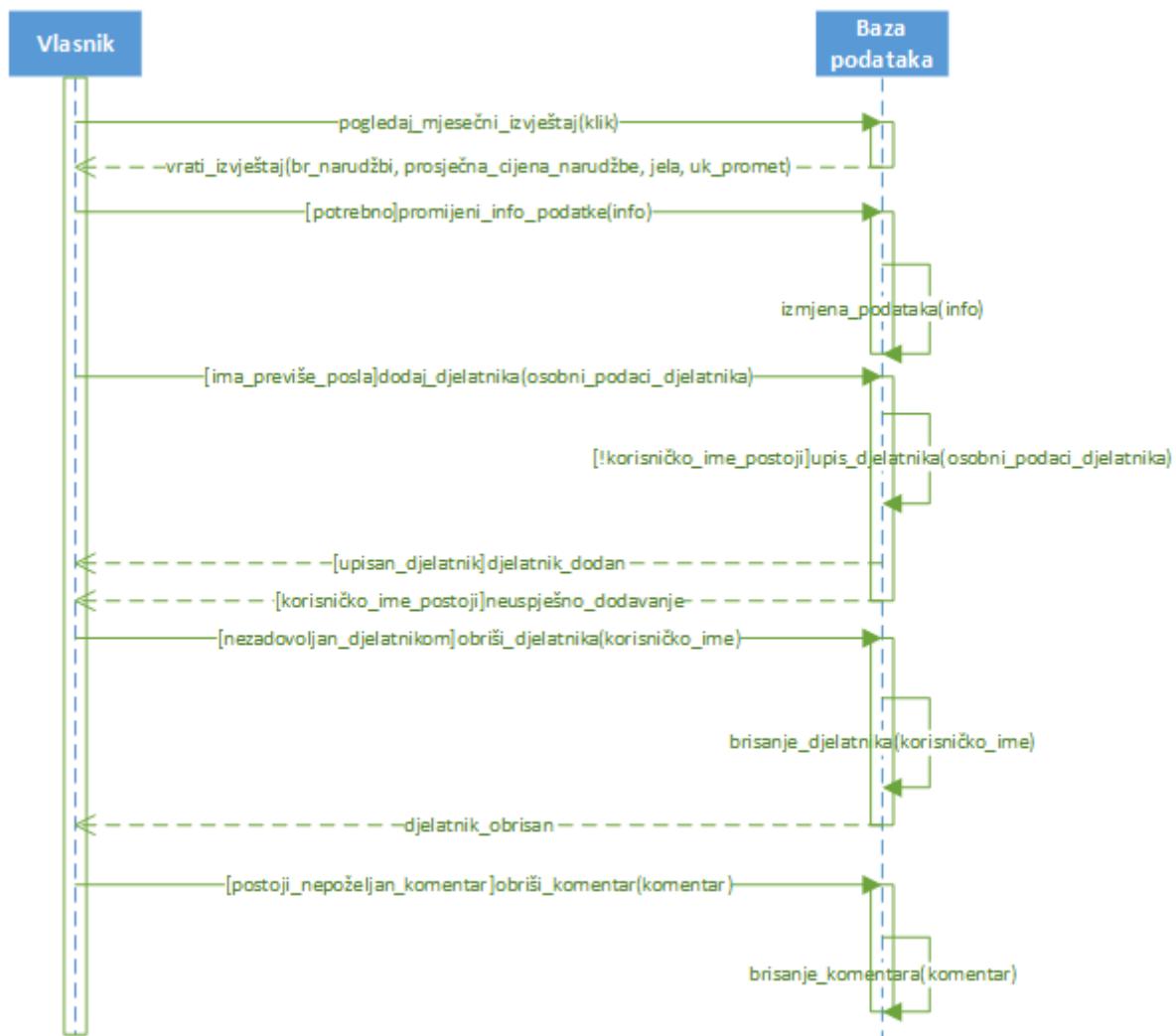
Djelatnik šalje prijavu u sustav unosom korisničkog imena i lozinke. Baza podataka provjerava te podatke te, ako podaci odgovaraju, odobrava ovlasti djelatniku. Ako podaci ne odgovaraju, javlja da su uneseni neispravni podaci. Nakon prijave, djelatnik zahtjeva prikaz narudžbi te mu baza podataka vraća narudžbe ako ih ima. Ako ih nema, javlja da nema narudžbi. Djelatnik odabire nezaprimljenu narudžbu. Baza podataka označava tu narudžbu imenom djelatnika te mu vraća presliku narudžbe sa sadržajem košarice i osobnim podacima. Djelatnik zahtjeva snimanje narudžbe te od baze dobiva datoteku narudžbe. Kada je završio narudžbe, ako je potrebno, djelatnik mijenja popis kategorija i jela. Baza izvršava izmjenu ako ona već ne postoji te javlja djelatniku da je uspješno obavljena. Ako je postojala ta izmjena, djelatniku se javlja neuspješnost.



Slika 4.7. Sekvencijski dijagram obrazaca uporabe UC9, UC10, UC11 i UC12

Obrasci uporabe UC13, UC14, UC15 i UC16:

Vlasnik odabire pregled mjesecnog izvještaja. Baza podataka mu vraća izvještaj sa svim podacima. Ako je potrebno vlasnik unosi drugačije informacije o pizzeriji. Baza podataka vrši izmjenu podataka. Ako je bilo previše posla vlasnik želi dodati novog djelatnika unosom njegovih osobnih podataka. Ako uneseno korisničko ime već ne postoji baza podataka upisuje djelatnika te vlasnik dobiva poruku da je djelatnik dodan. Ako je postojalo uneseno korisničko ime, vlasnik dobiva poruku da je dodavanje neuspješno. Ako je vlasnik nezadovoljan djelatnikom želi ga obrisati unosom njegovog korisničkog imena. U bazi podataka se briše djelatnik te vlasnik dobiva poruku da je djelatnik obrisan. Na kraju vlasnik pregledava komentare te, ako postoji nepoželjan komentar, zahtjeva njegovo brisanje. Baza podataka briše odabrani komentar.



Slika 4.8. Sekvencijski dijagram obrazaca uporabe UC13, UC14, UC15 i UC16

Ostali zahtjevi

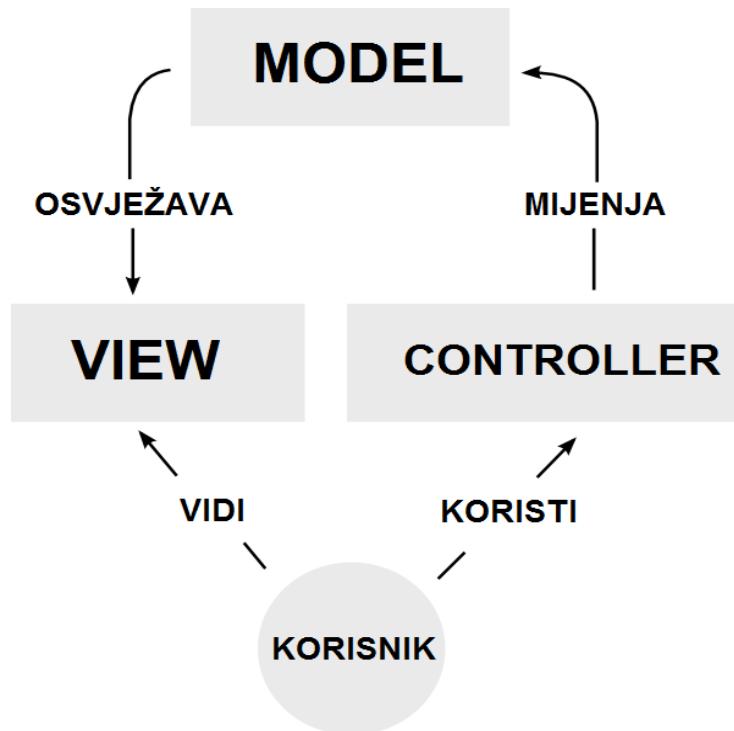
- Sustav mora podržavati istovremeni rad najmanje tri djelatnika, pet klijenata i jednog vlasnika.
- Sustav mora biti jednostavan za korištenje, bez potrebnih dodatnih uputa za korisnike.
- Sustav bi trebao davati pouzdane informacije o pizzeriji te ispravan izračun ukupnog iznosa.
- Tekstualni oblik izlazne datoteke i cijelo sučelje treba podržavati unos hrvatskih dijakritičkih znakova (UTF-8).
- Korisničko sučelje mora omogućiti izbor dodatnih mogućnosti koje dolaze sa dodijeljenim korisničkim dozvolama.
- Sustav mora svako jelo koje je naručeno barem dva puta češće od prosjeka naručivanja za sva jela označiti kao “često naručivano”.
- Sustav ne smije dozvoliti manju narudžbu od određene minimalne cijene.
- Sustav se ispituje sa najmanje četiri kategorije jela i najmanje 30 pojedinačnih jela.
- Sustav podržava nadogradnju izvornog koda uz korištenje Django 1.5 verzije.

Arhitektura i dizajn sustava

Svrha, opći prioriteti i skica sustava

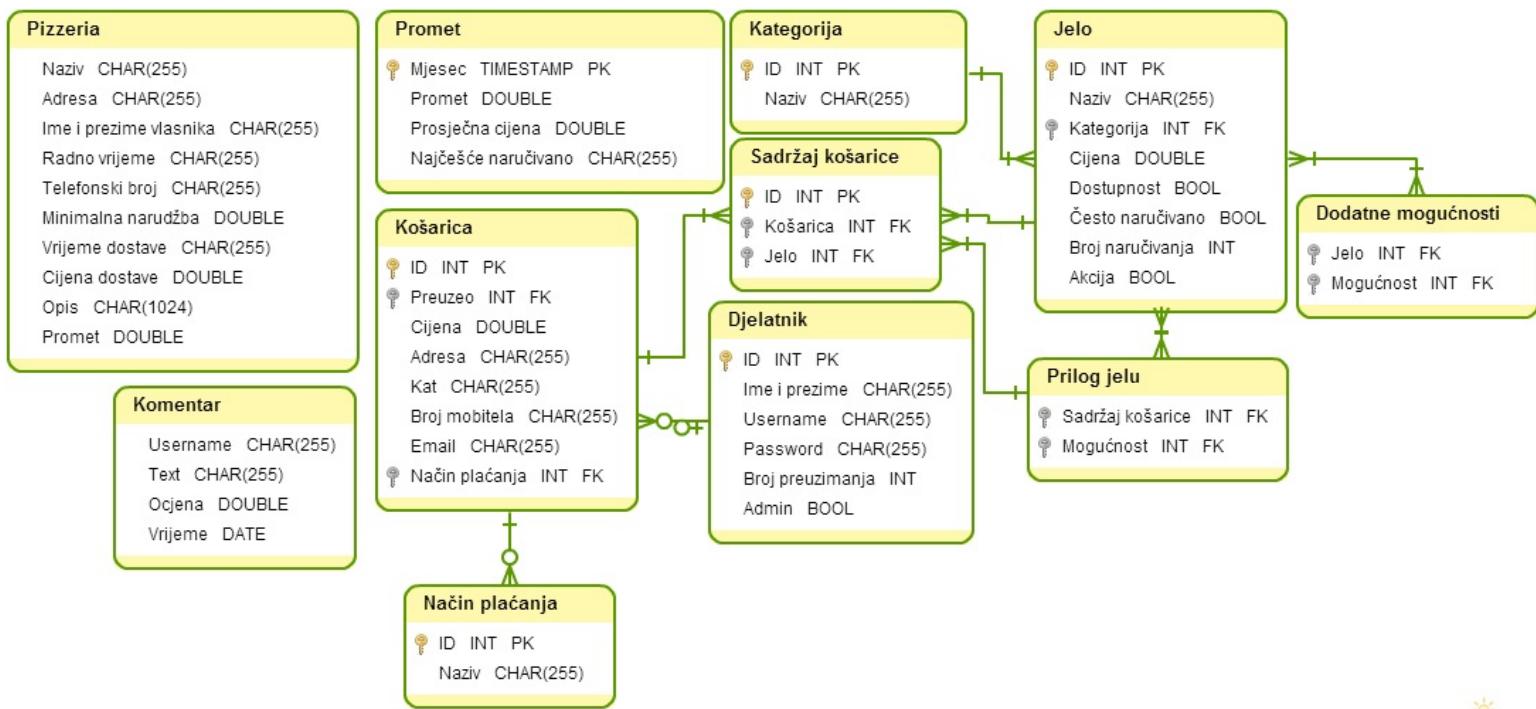
Sustav implementira online naručivanje hrane. Obzirom da vjerojatnost da korisnik uistinu nešto i naruči ovisi o lakoći korištenja sučelja, ono mora biti što jednostavnije i intuitivnije. Korisnik vrlo vjerojatno neće biti voljan instalirati dodatnu programsku podršku nego će se oslanjati na ono što već ima, a to je standardni browser. Zbog navedenih razloga je odluka pala na arhitekturu web aplikacije, konkretno često korištenu MVC arhitekturu¹¹.

MVC arhitektura se sastoji od tri dijela: **model**, **view** i **controller**. Svaki dio aplikacije pripada jednom od tih dijelova, ovisno o njegovoj ulozi u sustavu. Zbog takve apstraktne podjele olakšava se implementacija i ideja korištenja, kao i održavanje sustava te dodavanja novih mogućnosti. To je razlog zašto baš ova arhitektura prevladava internet aplikacijama. **MVC** arhitektura uključuje i interakciju između komponenti. Korisnik koristi **controller** te vidi **view**. **Controller** mijenja model, a model osvježava **view**. Dakako, na slici nije ilustrirano da je korisnik zapravo iza preglednika.



Model je komponenta arhitekture koja čuva podatke web aplikacije (u obliku baze podataka), logiku koja čini srž aplikacije i funkcije. Prilikom promjene stanja

modela svi asocirani **viewovi** i **controlleri** mogu biti obaviješteni o događaju. **View** tada osvježi svoj izlaz, a **controller** osvježi dostupne naredbe. **Model** će u našoj web aplikaciji sadržavati sučelje za SQLite bazu podataka, logiku za brojanje narudžbi, prometa te izračunavanja najčešće naručivanih jela. Baza podataka će držati informacije o jelima i kategorijama, narudžbama, kao i popis radnika i komentara, te konačno, osnovne informacije o samoj pizzeriji.

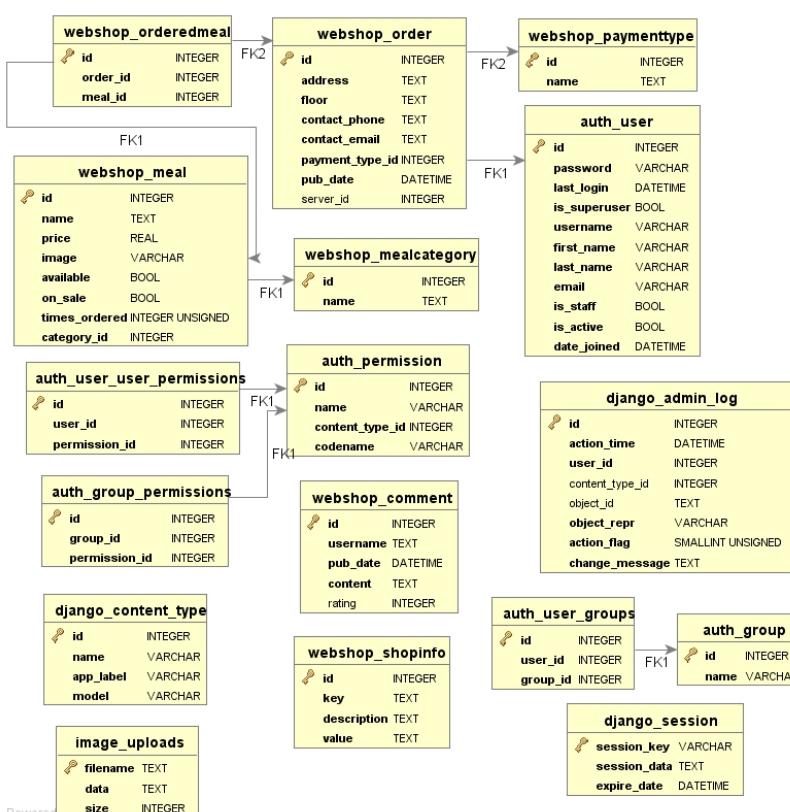


Tablica **Pizzeria** sadrži osnovne informacije od kojih će se većina prikazivati na naslovnoj stranici svakom korisniku. Tablica **Promet** služi kao log iznosa prometa po mjesecima. Aktualni mjesec je u tablici **Pizzeria**, te se na kraju mjeseca automatski prebacuje u tablicu **Promet**. Tablica **Kategorija** čuva nazive kategorija. Tablica **Jelo** drži sve informacije o pojedinom jelu, uključujući kategoriju kojoj to jelo pripada. Bitno je reći da ona drži i popis dodatnih mogućnosti i priloga. Mogućnost odabira neke dodatne mogućnosti ovisi o jelu, pa se zato tablica **Jelo** refleksivno veže sama sa sobom kroz tablicu **Dodatne mogućnosti**. Tamo su povezana jela sa svim dodatnim mogućnostima koje se mogu birati za to pojedino jelo. **Košarica** je log svih podataka o svim dostavama. Očigledno, tablica **Sadržaj košarice** skupa sa tablicom **Prilog jelu** je veza između jela, košarica i odabranih priloga, odnosno utjelovljenje je popisa jela i pripadajućih dodatnih mogućnosti koji su unutar jedne košarice. Tablica **Komentar**

služi kao zapisnik svih komentara i ocjena, dok tablica **Djelatnik** služi kao popis svih aktualnih djelatnika i administratora (vlasnika). Nakon što djelatnik preuzme neku košaricu, njegovo se ime zapiše u odgovarajući stupac tablice **Košarica**. U toj je tablici također i zapisan odabrani način plaćanja, sa popisa svih dostupnih načina plaćanja iz tablice **Način plaćanja**.

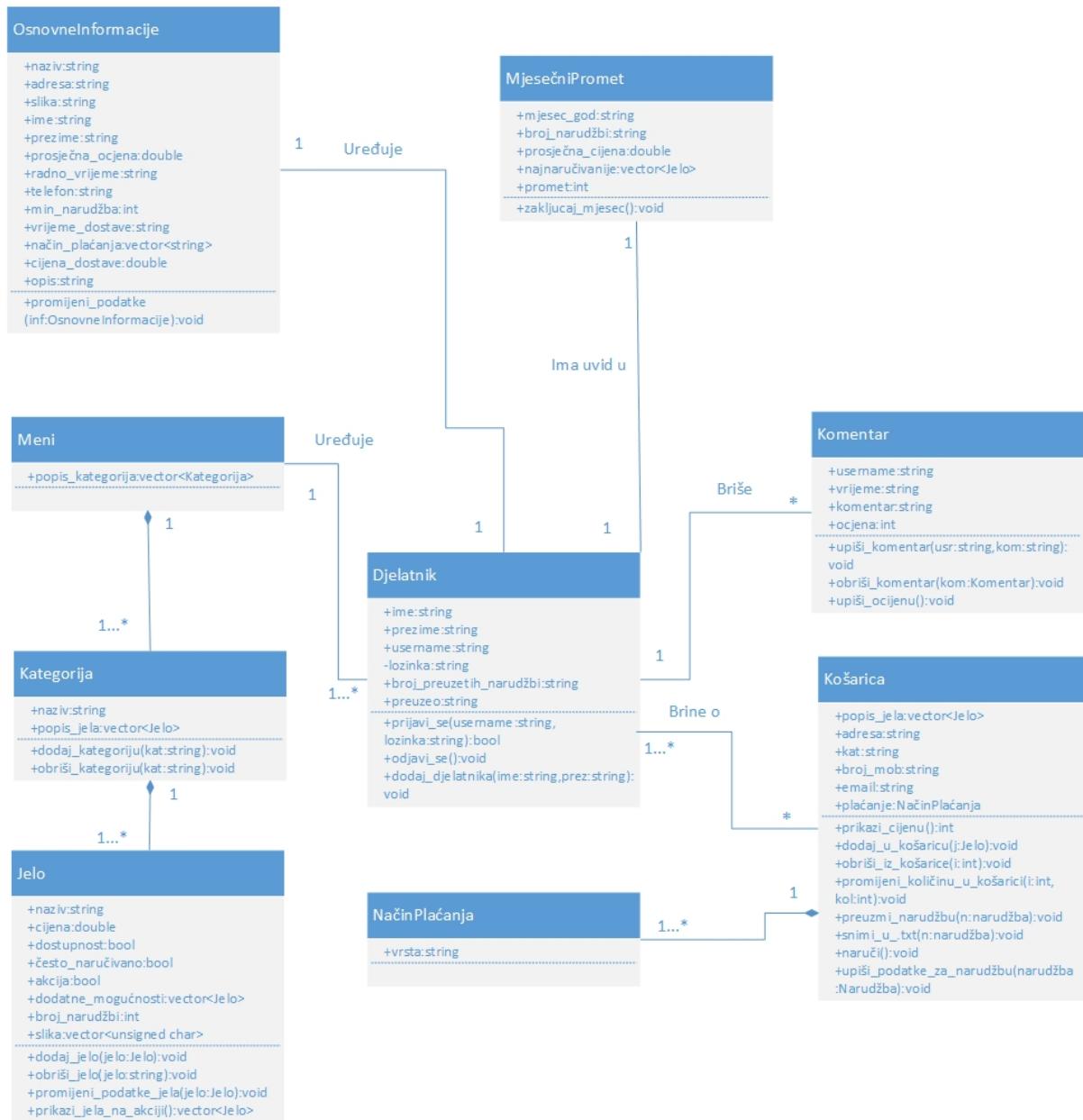
View je bilo koji prikaz podataka iz **modela**, poput slike, teksta, dijagrama, grafa itd. Različiti **viewovi** mogu prikazivati iste podatke u različitim oblicima, dakle reprezentacija podataka ovisi isključivo o implementaciji **viewa**. U tom smislu je **view** pasivna komponenta **MVC** arhitekture - ne upravlja ničime, niti što mijenja. U našoj aplikaciji će **viewovi** biti pogledi na naslovnu stranicu, meni, košaricu, narudžbu, popis narudžbi za preuzimanje, popis komentara i popis radnika.

Controller je komponenta koja posreduje između unosa i promjene stanja **modela** ili **viewa**. **Controller** oblikuje korisnikov unos, pretvarajući ga u naredbe za interpretaciju **modelu** i/ili **viewu**. Na svakom **viewu** ćemo imati kontrole koje odgovaraju korisnikovoj razini prava. Administrator i djelatnik će moći uređivati neke podatke iz **modela**, koje će klijent samo moći gledati. Klijent će upravljati narudžbom koristeći određene kontrole, dok djelatnik neće.



Dosad je bila objašnjavana apstraktna ideja arhitekture sustava. Django framework puno stvari napravi automatski, bez programerovog utjecaja, stoga ER dijagram izgleda znatno drugačije.

Dijagram razreda s opisom



Razred **OsnovneInformacije** za članove ima sve podatke koji će se prikazivati na naslovnoj stranici. Sukladno specifikacijama, postoje tri razine prava pristupa, te ovisno o njima neke od tih informacija će biti izostavljene iz prikaza.

Razred **Vlasnik** predstavlja vlasnika restorana koji jedini ima ulogu administratora. On jedini može uređivati i mijenjati **OsnovneInformacije** i jedini ima pristup objektima razreda **MjesečniPromet**.

Očigledno, **MjesečniPromet** se sastoji od podataka o prometu kroz mjesecu i ostalih interesantnih podataka, poput najnaručivanih jela i ukupnog prometa. Na kraju

svakog mjeseca će se automatski pozvati funkcija `zakljucaj_mjesec()` koja će spriječiti daljnje izmjene u proteklom mjesecu.

Osim što **Vlasnik** može mijenjati **OsnovneInformacije**, također može mijenjati i **PopisDjelatnika**. Vlasnik je osoba koja odlučuje tko se zapošljava u restoranu, stoga je prirodno da razred **Vlasnik** jedini može dodavati nove zaposlenike.

Razred **Meni** je hijerarhijski iznad razreda **Kategorija** i **Jelo**, te će kao takav sadržavati popis **Kategorija**. Prateći isti princip, svaki razred **Kategorija**, osim svog imena, sadrži popis jela koje pripadaju toj kategoriji. Objekti razreda **Jelo** će predstavljati pojedino jelo koje se može ubaciti u košaricu i naručiti.

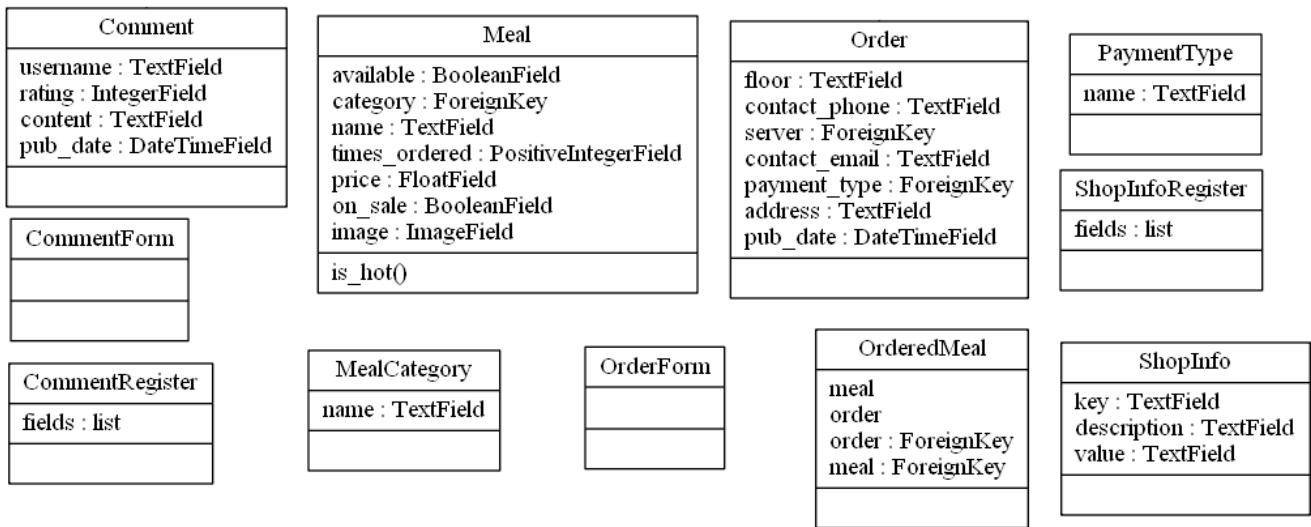
Jedini razred koji može uređivati **Meni** je razred **Djelatnik**, međutim, prema specifikacijama vlasnik ima sva prava kao i djelatnici i neka dodatna, stoga razred **Vlasnik** nasljeđuje razred **Djelatnik**. Djelatnik nad menjem vrši operacije dodavanja, brisanja i uređivanja jela i kategorija. To mu je pravo eksplicitno dodjeljeno samo nad **Menijem**, no zato jer **Meni** u sebi kao članove ima **Kategorije i Jela**, ista prava vrijedi i nad njima.

Osim što može uređivati **Meni**, djelatnik ima i opciju preuzimanja **Narudžbe**. Razred **Narudžba** nasljeđuje razred **Košarica**, zato jer sadrži sve iste članove kao i **Košarica**, pa se na to nadoveže sa osobnim podatcima kupca koji radi narudžbu. Preuzimanjem narudžbe djelatnik vidi jela koja su naručena i adresu na koju treba napraviti dostavu. Kao dodatnu mogućnosti tu se ističe spremanje te narudžbe u .txt format na hard disk.

Razredi **Djelatnik** i **Kupac** oboje nasljeđuju apstraktni razred **Korisnik**, zato jer su ultimativno i **Djelatnik** i **Vlasnik** i **Kupac** korisnici sustava. Ono što im je svima zajedničko je to da imaju korisničko ime, kao što стоји u **Korisniku**.

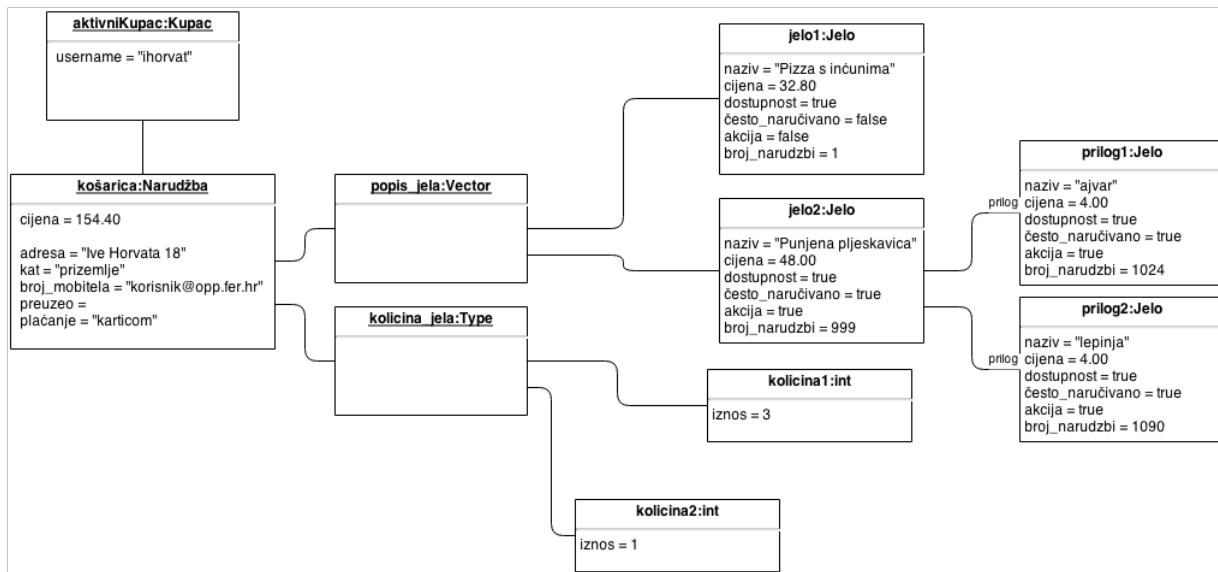
Kupac je razred koji stvara **Košarice** i bira što će sa **Menija** ići u nju. Kada je zadovoljan **Košaricom** koju je napravio, stvara **Narudžbu**. Osim naručivanja hrane, **Kupac** može ostaviti **Komentar** na uslugu, kao i ocjenu. I dok je stina da **Vlasnik** može brisati **Komentare** koji mu se ne sviđaju, ocjene ne može obrisati te su one trajno zapamćene u sustavu.

Kao i kod opisa modela, opisana je apstraktna ideja koja se od implementacije razlikuje.

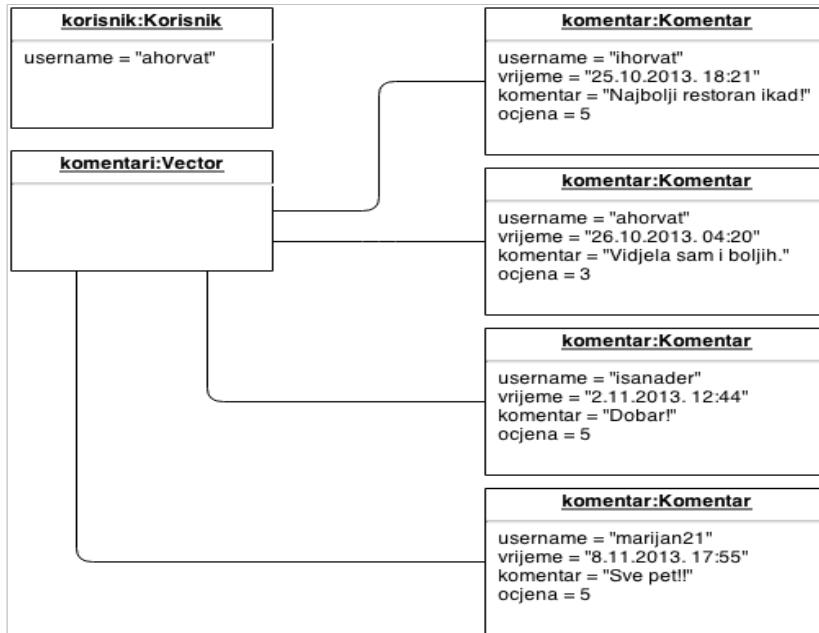


Važno je napomenuti da veze među razredima nisu eksplisitno istaknute već su izražene preko posebnog razreda ForeignKey koji pripada Djangu i služi za povezivanje modela pomoću Djangovog ORM-a.

Dijagram objekata



Slika 6.4 prikazuje objektni dijagram svih aktivnih objekata u trenutku narudžbe.



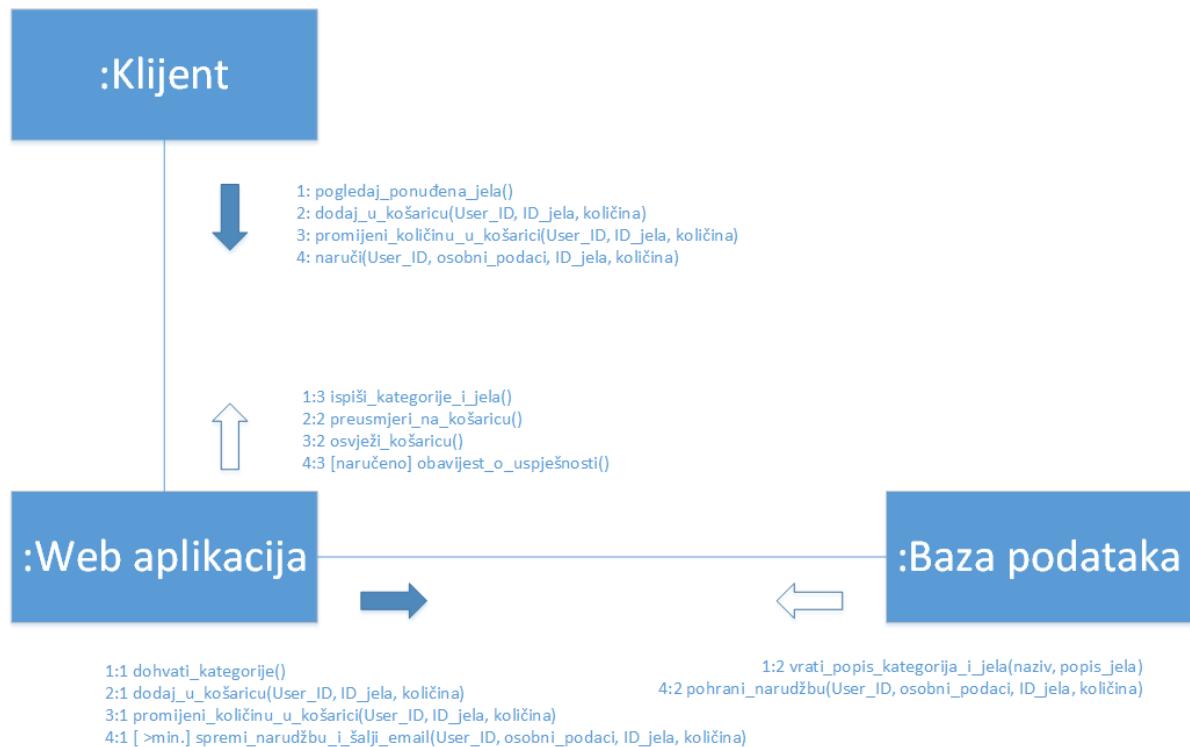
Slika 6.5. prikazuje objektni dijagram pri pregledu komentara

Dijagrami objekata su sukladni prvoj verziji dijagrama razreda. Da bi ovaj dijagram prikazao stvarno stanje sustava, trebali bismo istraživati Djangov izvorni kod, što je veoma opsežan zadatak.

Ostali UML dijagrami

Komunikacijski dijagram

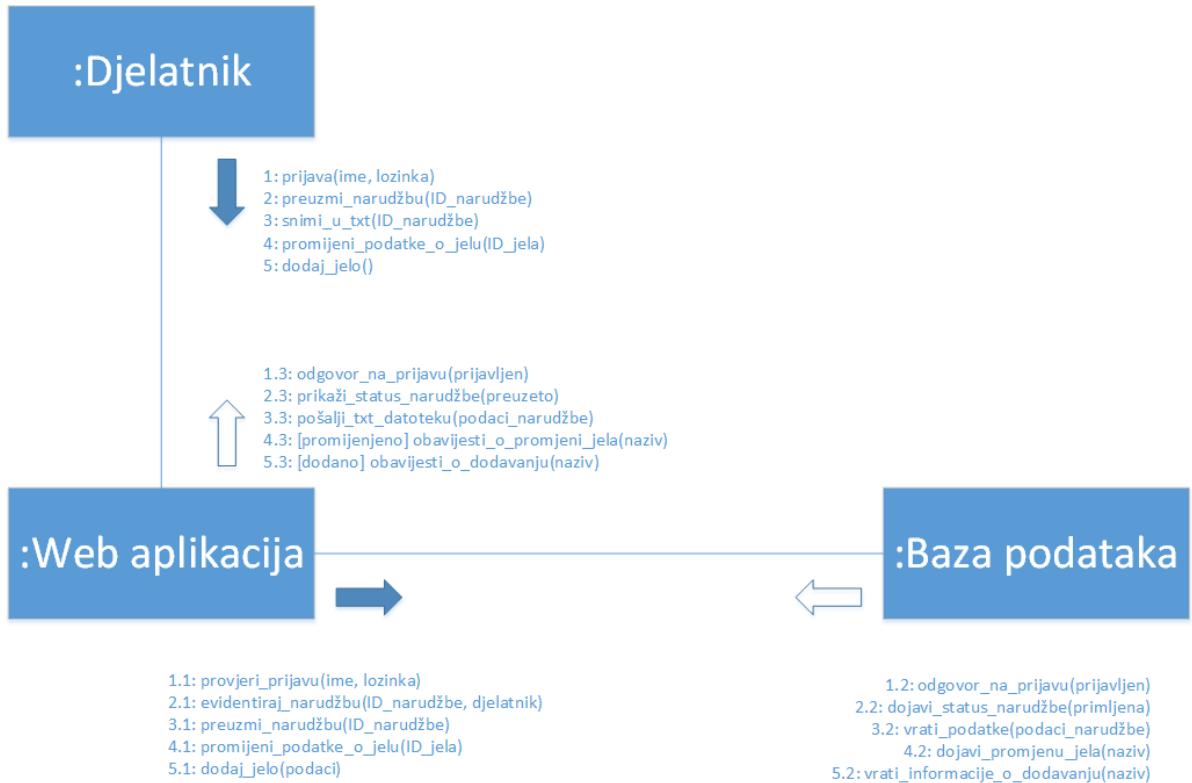
Klijent dolazi na stranicu pizzerije i želi pregledati ponuđeni meni. Nakon što odabere i zatraži stavljanje jela u košaricu, sustav prima zahtjev, preusmjerava korisnika na košaricu koja sad sadrži ta jela i prikazuje njeno trenutno stanje. Korisnik može manipulirati jelima u košarici kako želi te isto tako dodavati nova. Nakon što je gotov sa odabirom, korisnik konačno naručuje tako što upiše osobne podatke i potvrdi narudžbu. Sustav tu narudžbu sprema u bazu podataka, šalje kopiju na klijentovu e-mail adresu i daju klijentu povratnu informaciju o uspješnosti narudžbe.



Komunikacijski dijagram (UC2, UC3, UC4, UC5)

Djelatnik otvara web aplikaciju i traži prijavu u administracijski sustav pomoću svog korisničkog imena i lozinke. Nakon uspješne prijave, djelatnik bira narudžbu koju želi preuzeti i šalje zahtjev za primanjem narudžbe pa web aplikacija u komunikaciji sa

bazom podataka evidentira njegov zahtjev. Djelatnik može snimiti (zatražiti ispis) narudžbu. Tada, web aplikacija šalje upit bazi podataka sa ID-om te narudžbe da bi dobila sve podatke o njoj koje šalje korisniku u tekstualnoj datoteci. Djelatnik ima dozvolu da mijenja i popis jela ako želi. Prilikom dodavanja novog jela baza podataka uspješno izvršava dodavanje ako takvo jelo već ne postoji na meniju.

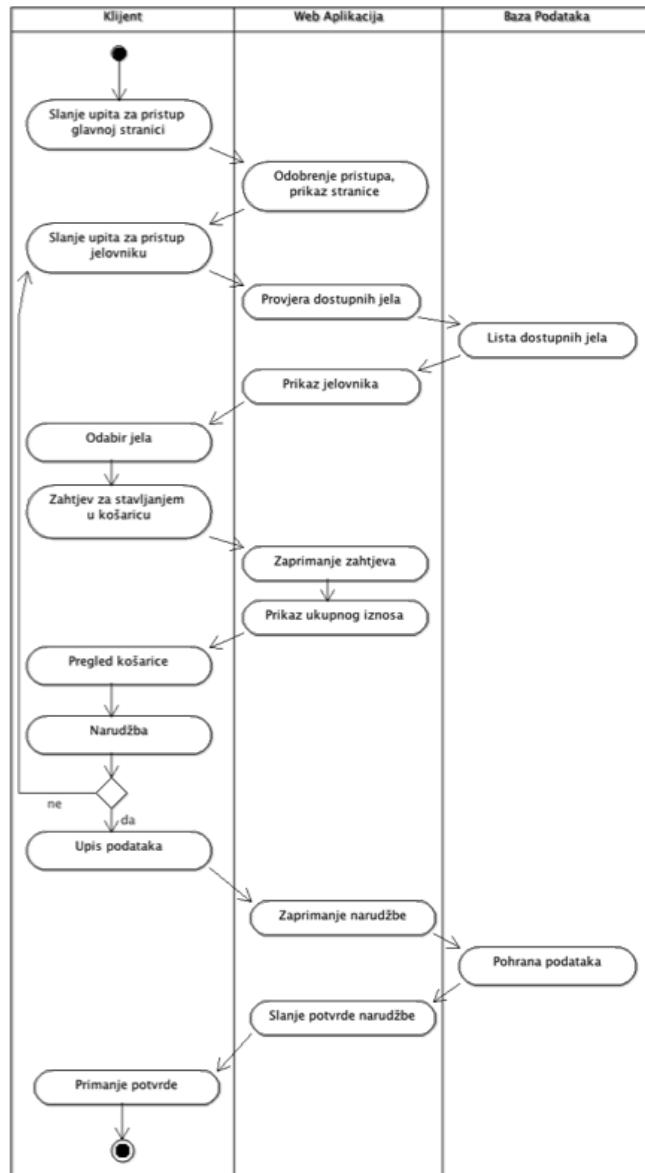


Komunikacijski dijagram (UC9, UC10, UC11, UC12)

Dijagram aktivnosti

Klijent šalje upit i pristupa glavnoj stranici pri čemu aplikacija vraća podatke o pizzeriji, meniju i načinu naručivanja. Nakon toga klijent zahtijeva pristup meniju, a aplikacija mu iz baze podataka vraća sva jela dostupna za naručivanje. Klijent pohranjuje odabrana jela u košaricu. Aplikacija računa trenutni iznos. Klijent izabire želi

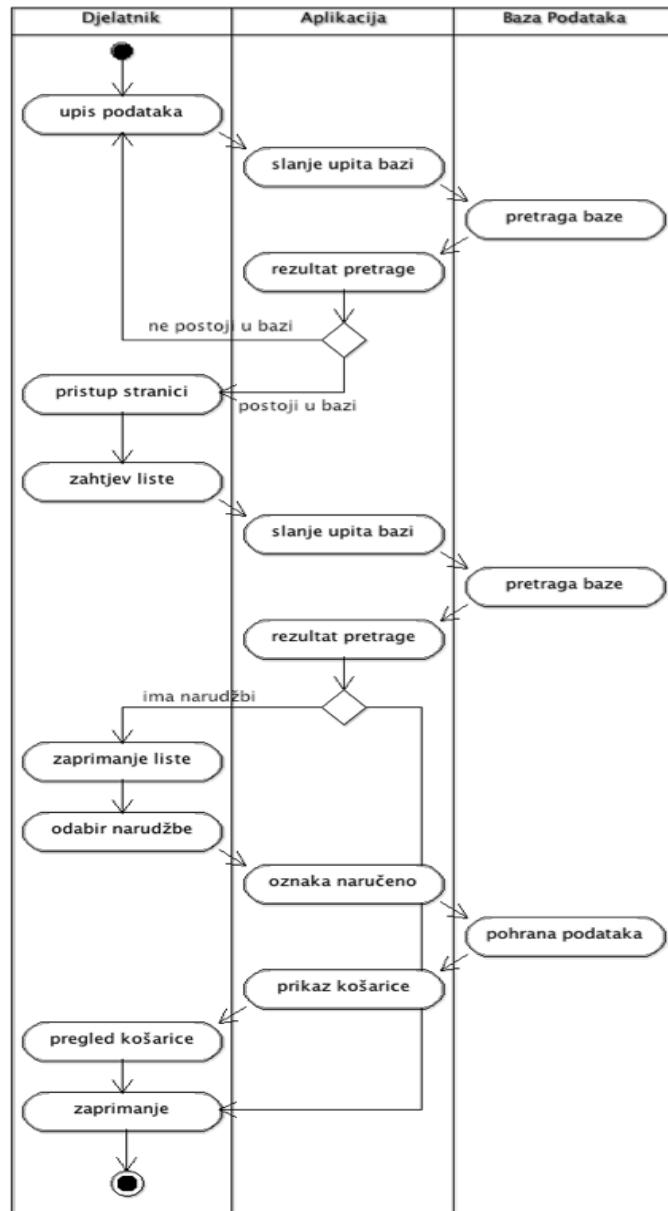
li naručiti iduće jelo ili potvrditi narudžbu. Ukoliko se odluči za izbor jela otvara se jelovnik, a ukoliko se odluči za potvrdu narudžbe upisuje podatke. Aplikacija šalje podatke na pohranu u bazu podataka i šalje klijentu potvrdu o provedenoj narudžbi.



Dijagram aktivnosti (UC1, UC2, UC4, UC5)

Djelatnik upisivanjem podataka pristupa aplikaciji. Ako su podaci u bazi podataka, njegov identitet je potvrđen te mu se odobrava pristup. Ukoliko podaci nisu pronađeni, sustav od djelatnika zahtjeva ponovni upis podataka. Kada jednom pristupi stranici, djelatnik zahtjeva listu narudžbi koje su na raspolaganju. Ukoliko je lista prazna,

djelatnik završava s radom. Ako nije, djelatnik odabire narudžbu, aplikacija sprema promjene u bazu podataka te mu daje prikaz košarice.



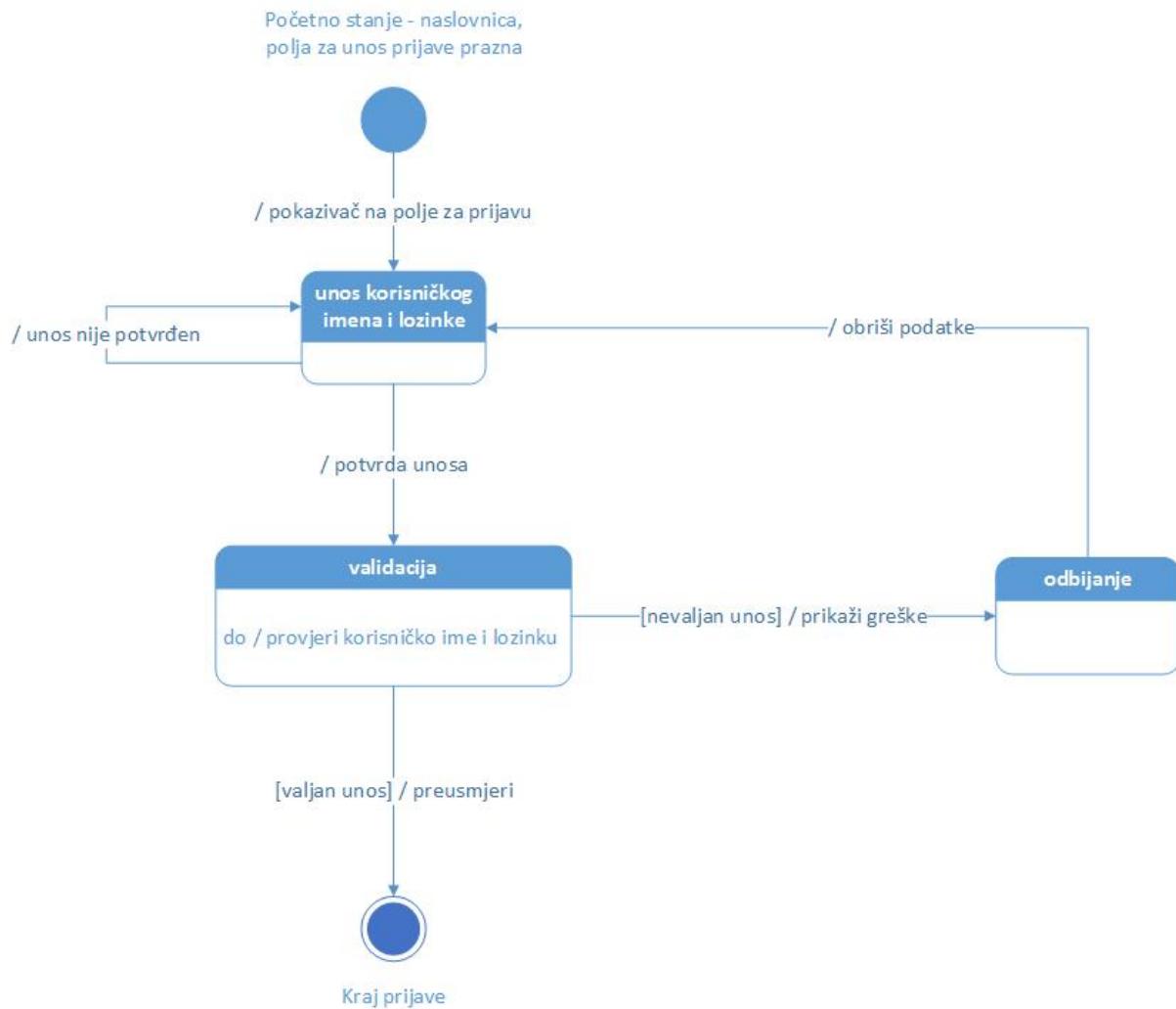
Dijagram aktivnosti (UC9, UC11)

Dijagrami stanja

Prijava u sustav - korisnik se prijavljuje u sustav s početne stranice odabirom administracijskog sučelja te unosom podataka u polja za korisničko ime i lozinku. Nakon

potvrde unosa traženih podataka pokreće se validacija koja provjerava postoji li korisnik te je li unio ispravnu lozinku.

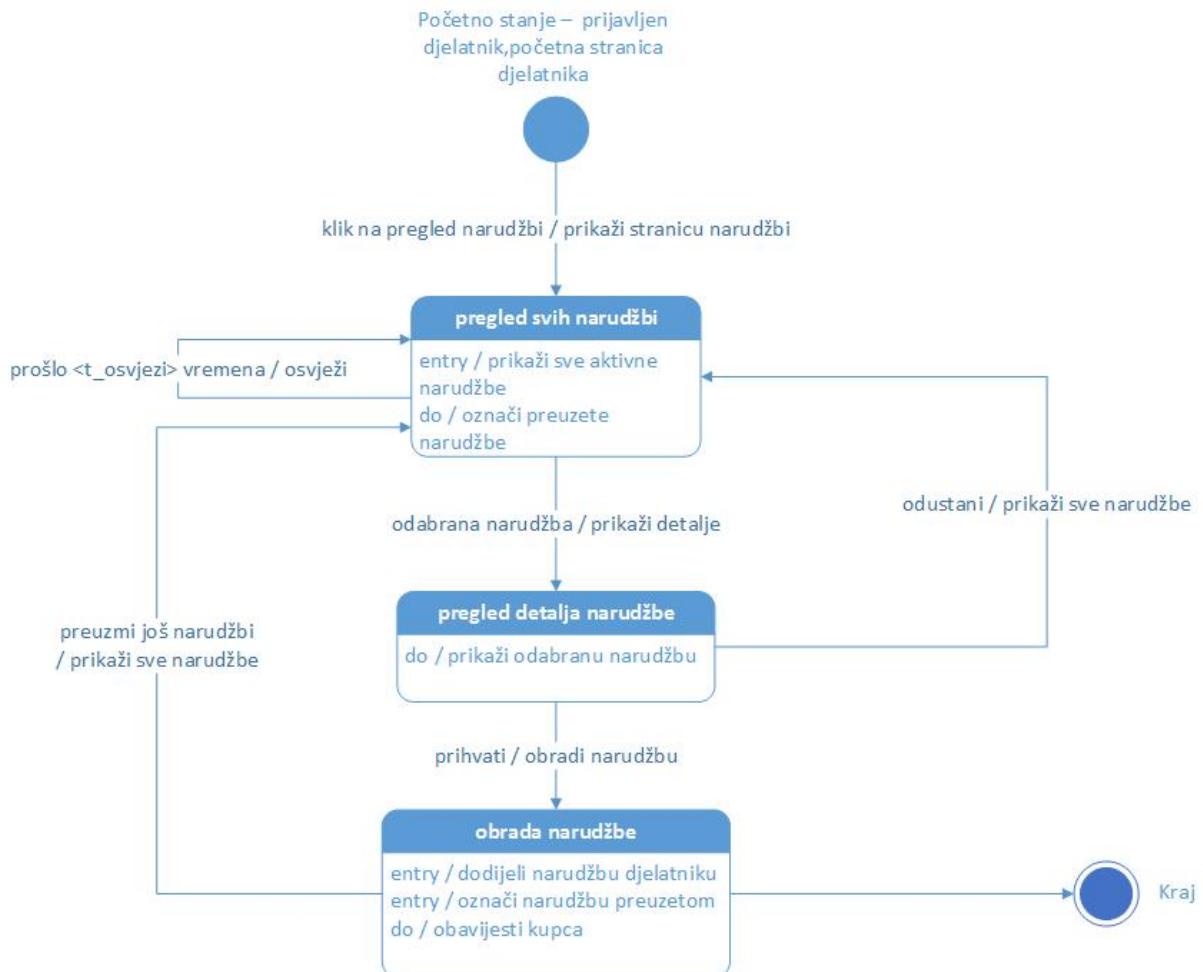
1. U slučaju greške - korisnika se obavještava odgovarajućom porukom, brišu unešeni podaci te ga se preusmjerava na početnu stranicu da ponovi unos.
2. U slučaju ispravnih podataka - prijava se prihvata te se korisnika preusmjerava uzimajući u obzir njegove ovlasti.



Djelatnik preuzima narudžbu - djelatnik je priavljen u sustav i nalazi se na svojoj početnoj stranici. Odabire pregled svih narudžbi te mu se otvara stranica sa prikazom

svih aktualnih narudžbi, stranica se periodički osvježava. Djelatnik može izabrati narudžbu te pogledati njene detalje. Potom narudžbu može:

- odbij - povratak na popis svih aktualnih narudžbi
- prihvati - akcija se obrađuje, narudžba se dodjeljuje djelatniku, označava preuzetom te se obavještava kupca o istome. Djelatnik potom može preuzeti još narudžbi.

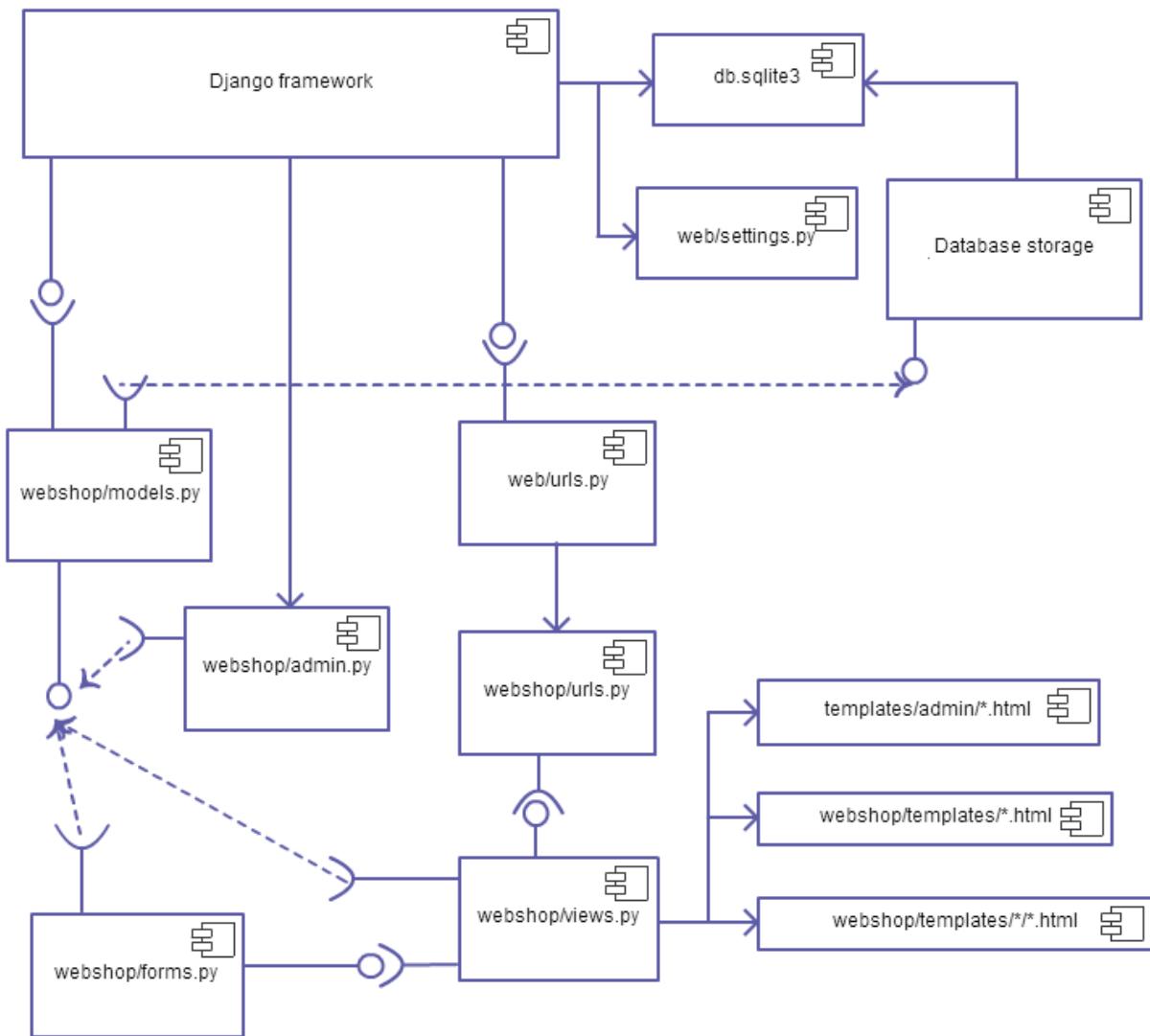


Dijagram stanja - djelatnik prihvata narudžbu/e

Dijagram komponenti

Komponente i njihov odnos odgovara idućem dijagranmu.

Komponenta Django framework predstavlja upravo to - sve ono što je u pozadini,



čemu programer ima malo pristupa i što preuzme puno obaveza na sebe. Django pruža sučelje koje se najviše koristi u *models.py*, i *urls.py*. Programer bazi podataka pristupa posredno kroz razred *Model*, kojeg pruža Django. Većina komunikacije sa bazom podataka se odvija bez programerove intervencije. Kao konfiguracijske datoteke se koriste *admin.py* i *settings.py*.

models.py implementira dio modela MVC arhitekture, koji je dostupan programeru. Svaki implementirani razred nasljeđuje razred *Model* kojeg pruža Django

komponenta. Također se koriste usluge *Database storage* dodatka, za jednostavno spremanje slika u bazu podataka.

forms.py predstavlja forme za upis potrebnih podataka prilikom narudžbe i ostavljanja komentara s ocjenom. Da bi se to ostvarilo, mora se imati pristup odgovarajućim rezredima (Comment, primjerice).

views.py je odgovoran za generiranje view-ova MVC modela. Dakako, mora moći pristupiti formama koje treba prikazati, stoga se spaja na sučelje koje pruža *forms.py*. Sve .html datoteke koje *views.py* pročita imaju dio sintakse specifičan za Django, koji dinamički generira sadržaj.

Temeljem pravila u *urls.py* datotekama Django odlučuje koji će view prikazati korisniku ovisno o zahtjevanom URL-u.

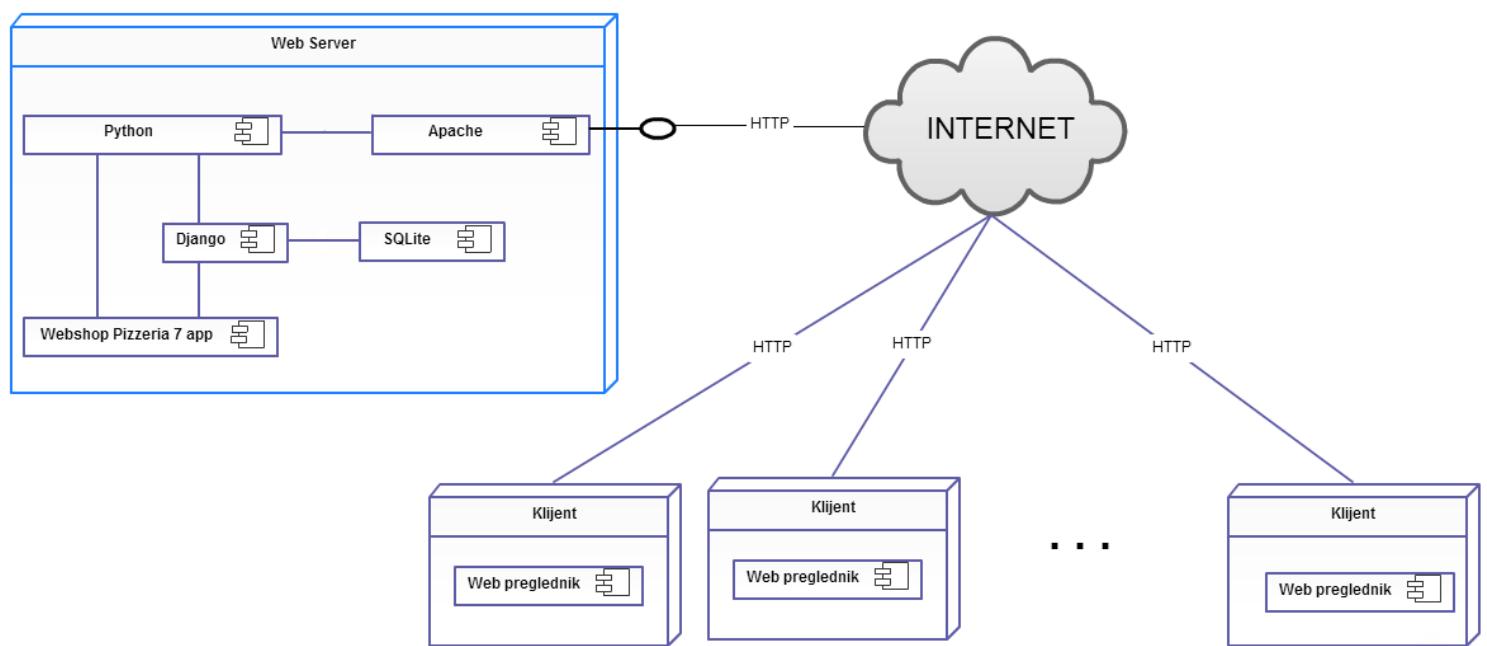
U dijagramu komponenti se mogu primjetiti slični elementi i veze kao na općenitom dijagramu MVC arhitekture.

Radi specifičnosti jezika Python se program ne prevodi u izvršnu datoteku, nego svaka .py datoteka dobije svoj .pyc par koji sadrži prevedeni *bytecode*. Obzirom da .pyc datoteke nisu bitne za prikaz komponenata i njihovih veza, one nisu prikazane u dijagramu.

Implementacija i korisničko sučelje

Dijagram razmještaja

Dijagram razmještaja je veoma jasan. Prikazuje klijent-server odnos. Serveru se pristupa standardnim web preglednikom. Python, Django i aplikacija se na serveru izvode iza Apache web servera.



Korištenе tehnologije i alati

Za razvoj web aplikacije pizzerije korišten je moćni *Vim* tekst editor te napredno *Python* okruženje (framework) *Django 1.6* koji je sve češće rješenje za izradu ovakvih web aplikacija zbog svog čitljivog koda, jednostavne upotrebe i održavanja.

Za kreiranje korisničkog sučelja korišten je *Twitter Bootstrap* koji svojim gotovim elementima olakšava izradu responzivnog dizajna koji će jednako dobro izgledati na stolnim računalima, prijenosnicima i mobitelima.

Kao baza podataka odabran je *SQLite* sustav, koji dolazi upakiran sa *Django* instalacijom. Ukoliko se ukaže potreba, veoma je lako zamijeniti *SQLite* sa nekom ozbiljnijom bazom podržanom u Djangu, poput *PostgreSQL* ili *MySQL*.

Za lakše spremanje slika sa jelovnika u bazu podataka koriste se dodatci *Pillow* i *Django Database Storage*.

Još jedan dodatak pomaže pri baratanju sa vremenskim oznakama. Zove se *pytz*.

Prilikom razvoja projekta, zbog lakše komunikacije i zajedničkog rada između članova tima, korišten je *Google Docs* i za rad s repozitorijom subversion klijent *TortoiseSVN*.

Za virtualne sastanke i sastanke na daljinu korišteni su *Skype* i *Facebook*.

Isječak programskog koda vezan za temeljnu funkcionalnost sustava

```
102 def cart(request):
103     cart = request.session.get('cart', {})
104     meals = [(Meal.objects.get(pk=k), v) for k, v in cart.items()]
105     total = sum(map(lambda (m, c): m.price * c, meals))
106     minimum = float(ShopInfo.objects.get(key='min_order').value)
107     delivery = float(ShopInfo.objects.get(key='delivery_cost').value)
108     full = total + delivery
109
110     if request.method == 'POST':
111         form = OrderForm(request.POST)
112         if form.is_valid():
113             order = form.save()
114
115             for (meal, count) in meals:
116                 meal.times_ordered += 1
117                 meal.save()
118                 for i in range(0, count):
119                     orderedMeal = OrderedMeal()
120                     orderedMeal.meal = meal
121                     orderedMeal.order = order
122                     orderedMeal.save()
123
124             request.session.__setitem__('cart', {})
125
126             mail = render_to_string('order/textual.html', {
127                 'order': order,
128                 'meals': map(lambda (m,c):m, meals),
129                 'total': full,
130                 'delivery': delivery
131             })
132
133             #send_mail(u'Vaša narudžba (#{}).format(order.id),
134
135             return redirect('webshop.views.cart_success')
136     else:
137         form = OrderForm()
138
139     return render(request, 'cart/index.html', {
140         'meals': meals,
141         'total': total,
142         'form': form,
143         'delivery': delivery,
144         'full': full,
145         'minimum': minimum
146     })
```

Slika prikazuje odsječak koda (točnije *view*) zaslužan za prikaz i obradu košarice. Odsječak zaslužan za slanje email obavijesti (linija 133) je zakomentiran zbog poteškoća s e-mail poslužiteljem.

Ispitivanje programskog rješenja

Test 1

Ispitni slučaj:

U meniju bi se ispravno trebala prikazati jela koja su češće naručivana.

Očekivani rezultat:

Jela koja su naručivana dvostruko više od prosjeka bi trebala imati sukladnu oznaku pokraj imena.

Ostvareni rezultat:

Uistinu, uz jela koja zadovolje kriterij pojavi se vruća oznaka često naručivanog jela.

Test 2

Ispitni slučaj:

U košarici bi se trebala prikazati odabrana jela sa menija uz ukupnu cijenu.

Očekivani rezultat:

Svaki put prilikom klika na neko jelo, ono se doda u košaricu, te se pregledom košarice može količina pojedinog jela povećati, smanjiti ili se jelo u potpunosti maknuti iz košarice.

Ostvareni rezultat:

Upravo kako je očekivano, tako i radi. Prikazuju se i forme za upis osobnih podataka i ukupna cijena narudžbe, uključujući dostavu i provjeru je li ispod minimalnog iznosa.

Test 3

Ispitni slučaj:

Osim općih informacija, na naslovnoj stranici bi se trebale prikazivati prosječna ocjena korisnika i jela na akciji.

Očekivani rezultat:

Prosječna se ocjena mijenja sukladno prosjeku ocjena iz komentara korisnika.

Prikazuju se sva jela koje djelatnici označe kao da su na akciji.

Ostvareni rezultat:

Ispravno su prikazani i prosjek ocjena i jela na akciji.

Test 4

Ispitni slučaj:

Samo djelatnici i vlasnik trebaju moći pregledati postavljene narudžbe.

Očekivani rezultat:

Kada se djelatnik ili vlasnik prijave na sustav i pregledaju narudžbe, zlonamjeran korisnik očekuje pritiskom na tipku back doći do osobnih podataka nevinih naručitelja, iako vlasnik ili djelatnik više nisu prijavljeni.

Ostvareni rezultat:

Unatoč očekivanjima, zlonamjerni se korisnik nije mogao vratiti na traženu stranicu. Dinamička priroda Djanga onemogućava ovakve zloporabe.

Upute za instalaciju

Web aplikacija se treba izvoditi u kontekstu web servera, uz Pythonov interpreter. Python je podržan na jako puno operacijskih sustava, od kojih su najznačajniji Windows, UNIX/Linux i Mac OS X. Koristeći službene upute kao referencu, prepostaviti ćemo korištenje Apache servera, iako je sa Djangom uključen *lightweight* server, za testiranje tokom razvoja.

U Django instalaciju je uključena SQLite baza podataka, koja je prepostavljena u uputama radi jednostavnosti. Inače, moguće je koristiti i neku drugu implementaciju baze podataka.

Prva tri koraka nisu glavna tema ove dokumentacije, stoga nisu pokriveni detaljno.

1. Instalacija i konfiguracija Apache poslužitelja

(moguće je koristiti bilo koji poslužitelj s podrškom za FastCGI protokol)

2. Instalacija i konfiguracija mod_wsgi modula za Apache

3. Instalacija Pythona (2.7, ostale verzije nisu uspjele pravilno izvoditi aplikaciju)

4. Uz Python se distribuira *package manager* "pip", pa se pokretanjem naredbe

pip install Django

iz komandne linije na najjednostavniji način instalira Django

5. Prvi *dependency* je Python Image Library koji se instalira na sličan način:

pip install Pillow

6. Drugi *dependency* je Django-database-storage koji se instalira na sličan način:

pip install django-database-storage

7. Treći *dependency* je pytz koji se isto tako instalira:

pip install pytz

8. Skidanje izvornog koda aplikacije, te pozicioniranje u korjenSKI direktorij na komandnoj liniji, zatim izvođenje

manage.py syncdb

nakon čega će biti ponuđen upis administratorskog korisničkog imena i šifre za pristup bazi podataka

Nakon ovih koraka sve što je potrebno za pokretanje poslužitelja i aplikacije je instalirano. Eventualno, za potrebe testiranja aplikacije, izvođenjem sljedeće naredbe na komandnoj liniji

`manage.py runserver`

pokreće se Djangov jednostavan testni web poslužitelj, koji postavi web stranicu na `localhost` adresu, na port 8000. Naravno, adresa i port mogu se mijenjati.

Korisničke upute

Do aplikacije se dolazi kao do bilo koje druge web stranice, upisivanje odgovarajuće adrese u adresnu traku u preferiranom web pretraživaču. Aplikaciji mogu pristupiti tri različita tipa korisnika:

- neregistrirani korisnik/klijent
- vlasnik restorana
- djelatnik restorana (dijeli puno ovlasti sa vlasnikom)

Klijent

Snalaženje na web stranicama restorana je jednostavno i intuitivno. Na naslovnoj stranici klijent može pročitati korisne informacije, bitne za narudžbu, poput radnog vremena, cijene dostave i prosječnog vremena dostave ili jela na akciji.

The screenshot shows the homepage of the Pizzeria 7-ica website. At the top, there is a navigation bar with links: Pizzeria 7-ica, O nama, Meni, Moja košarica, Komentari drugih, and Administracija. Below the navigation bar, the main content area is divided into several sections:

- Pizzeria 7-ica**: Includes the address Ulica Grada Vukovara 234a, 10000 Zagreb and the owner's name Janko Strižić.
- Radno vrijeme**: Shows opening hours: Pon - Pet 7:00 - 24:00, Subotom 10:00 - 23:00, and Nedjeljom ne radimo. It also mentions Minimalna narudžba (35 HRK) and Cijena dostave (10 HRK).
- O pizzeriji**: Describes the restaurant as Vam već generacijama pruža uslugu najbolje pizzerije u gradu. Vrhunski, izvorno talijanski, kuvari spremaju jela koja i bogove zasite. Navratite i iznenadite se vrhuncem kulinarstva ovog stoljeća.
- Jela na akciji**: Lists two promotional offers:
 - Bolognese Talijanska 38,00
 - Margherita Pizze 28,00
- Načini plaćanja**: Lists payment methods: Amex and Gotovinom.

At the bottom of the page, there is a 5-star rating icon followed by the text "U prosjeku smo ocijenjeni sa ocjenom 5,00".

Pri vrhu ekrana je vidljiv minimalistički izbornik koji klijenta odvodi do njemu bitnih mjesto. Prva poveznica nakon poveznice na naslovnu stranicu ga vodi na pregled

menija, gdje može pogledati sve trenutno dostupna jela, složena po kategorijama. Jela koja su naručivana dvostruko više od prosjeka su drugačije označena. U gornjem desnom kutu se nalazi pregled jela koja su trenutno u košarici.

Domaća



Sarma i pire krumpir

Cijena: 25,00 HRK

Dodaj u
košaricu



Vuco i janjetina

Cijena: 666,00 HRK

Dodaj u
košaricu



Brodet i žganjci

Cijena: 40,00 HRK

Dodaj u
košaricu

U vašoj košarici



Sarma i
pire
krumpir

Kineska



Gonbao piletina

Cijena: 30,00 HRK

Dodaj u
košaricu



Nadalje, treća poveznica sa trake vodi klijenta u detaljniji pregled košarice, gdje može odabrati količinu pojedinih jela i dovršiti narudžbu upisivanjem osobnih podataka te biranjem načina plaćanja.

Sadržaj košarice



Sarma i pire krumpir

Cijena: 25,00 HRK

Količina: 1

Dodaj jedan
Makni jedan
Makni sve

Ukupna vrijednost: 25,00 HRK

Detalji narudžbe

Cijena artikala: 25,00 HRK

Cijena dostave: 20,00 HRK

Ukupna cijena: 45,00 HRK

Adresa dostave:

Kat:

Broj telefona:

e-mail:

Način plaćanja:

Konačno, četvrta poveznica sa trake vodi klijenta u pregled ostavljenih ocjena i komentara gdje takđer ima priliku ostaviti svoje.

Morate probati njihovu kinesku kuhinju. Predобра je. Recepti su imao savršeni, kao da kinezi, a ne talijani, rade u kuhinji.	Ovo je moj restoran! I slažem sa vama ostalima, da je fenomenalan!	Ovo je jako dobar restoran! Budem češće odavde naručivao!
 Maja 16. siječnja 2014. 00:25	 Janko 16. siječnja 2014. 00:24	 Luka 16. siječnja 2014. 00:23
SAVRŠENA pizze rade! Kad sam je probao, rekao sam im da mi odmah naprave još jednu, koliko je dobra bila.		
 Maja 15. siječnja 2014. 23:24		

Objavi svoje mišljenje!

Ime:

Sadržaj:

Ocjena:

Objavi

Vlasnik

Na traci je ostala još jedna nespomenuta poveznica, administracija. Ona nas vodi na prijavu u sustav, gdje se vlasnik i djelatnici mogu prijaviti.

Pizzeria 7-ica admin

Korisničko ime:

Lozinka:

Prijavi se

Nakon prijave se pojavljuje administratorski pregled bitnijih elemenata aplikacije s mogućnošću uređivanja istih.

The screenshot shows the 'Pizzeria 7-ica admin' dashboard. On the left, there's a sidebar with 'Auth' and 'Korisnici' sections. The main area has a 'Nedavne promjene' (Recent changes) box listing 'Moje promjene' (My changes) with items like 'Novi unos' (New entry) for 'Vlasnik' and 'Korisnik'. Below this is a 'Informacija o dućanu' (Shop information) section with tabs for 'Jela', 'Kategorije jela', 'Komentari', 'Narudžbe', and 'Načini plaćanja'.

Svaki podizbornik administrira jednu kategoriju. U "Informacije o dućanu" se mijenjaju općenite informacije koje se prikazuju na naslovnoj stranici. U izborniku "Jela" se, logično, mijenjaju informacije o jelima koja se nude u meniju. U "Kategorije" se mogu mijenjati/dodavati/brisati kategorije kojima jela pripadaju i koje se prikazuju na meniju. Izbornik "Komentari" nudi mogućnost brisanja komentara koji se ne sviđaju vlasniku. Konačno, u zadnjem se izborniku mogu mijenjati dostupni načini plaćanja.

Izbornik "Korisnici" omogućava dodavanje novih djelatnika ili promjenu vlasnika, ako on to želi. Pri dodavanju novog korisnika mogu se upisati njegove informacije te odabrati grupa kojoj pripada.

This screenshot shows the 'Promijeni korisnik' (Change user) form. It includes fields for 'Korisničko ime' (Username) set to 'djelatnik', 'Lozinka' (Password) set to 'algoritam: pbkdf2_sha256 iteracije: 12000 slučajna vrijednost: uGeVJ***** hash: O+Hfwf*****', 'Osobni podaci' (Personal data) with 'Ime' (Name) 'Marin', 'Prezime' (Last name) 'Držić', and 'E-mail adresa' (Email address) 'marin.drzic@md.com'. Under 'Privilegije' (Privileges), 'Aktivan' (Active) is checked. In the 'Grupe:' (Groups) section, 'Dostupne grupe' (Available groups) shows 'Vlasnik', and 'Odabранo grupe' (Selected groups) shows 'Djelatnici'.

Povratkom na naslovnu stranicu, vlasnik može primjetiti dvije nove poveznice. "Osoblje" i "Narudžbe".

Pizzeria 7-ica

Ulica Grada Vukovara 234a, 10000 Zagreb

Vlasnik

Janko Stržić



U "Osoblje" može vidjeti koliko je narudžbi koji djelatnik preuzeo, a u "Narudžbe" može i sam preuzeti narudžbu.

Popis djelatnika

Upravljanje djelatnicima

Korisničko ime	Ime	Prezime	Broj preuzetih narudžbi
admin			1
djelatnik			0
vlasnik			0

Narudžbe

Otvorene narudžbe

16. siječnja 2014. 22:24

Adresa: asd

Telefon: asd

Detalji narudžbe

Kat: asd

E-mail: luka.strzic@hotmail.com

[Preuzmi narudžbu](#)
[Preuzmi kao txt datoteku](#)

Način plaćanja: Gotovinom

Zatvorene narudžbe

16. siječnja 2014. 21:44

Adresa: Stjepana Radića 5

Telefon: 0987654321

Detalji narudžbe

Kat: 3

E-mail: vutojanjetina@vj.***

Preuzeo: admin

Način plaćanja: Amex

[Preuzmi kao txt datoteku](#)

Djelatnik

Prijavom djelatnika, na naslovnoj stranici poveznica "Osoblje" nestaje, i isto tako, zbog smanjenih privilegija se administracijsko sučelje mijenja.

Administracija stranica

Webshop

[Novi unos](#) [Promjeni](#)

Moje promjene

Jela

[Novi unos](#) [Promjeni](#)

Nije dostupno

Kategorije jela

[Novi unos](#) [Promjeni](#)

Narudžbe

Otvorene narudžbe

16. siječnja 2014. 22:24

Adresa: asd

Kat: asd

Telefon: asd

E-mail: luka.strizic@hotmail.com

Način plaćanja: Gotovinom

[Detalji narudžbe](#)

[Preuzmi narudžbu](#)

[Preuzmi kao txt datoteku](#)

Zatvorene narudžbe

16. siječnja 2014. 21:44

Adresa: Stjepana Radića 5

Kat: 3.

Telefon: 0987654321

E-mail: vuco.janjetina@vj.vj

Način plaćanja: Amex

[Detalji narudžbe](#)

[Preuzeo: admin](#)

[Preuzmi kao txt datoteku](#)

Zaključak

Trenutno stanje proizvoda je sukladno zahtjevima klijenta. Sve navedeno u dokumentaciji je implementirano i ispravno radi osim funkcionalnosti priloga jelu koja zbog vremenskih ograničenja i tehničkih zahtjeva nije bila implementirana.

Najteži izazov je bio naučiti baratati Django frameworkom. Pythonov moto je *Batteries included*, što Django doslovno i provodi. Puno koda izgradi i izvršava sam framework bez intervencije programera, zbog čega treba napisati izuzetno malo linija vlastitog koda. Time programer nema uvijek slobodu na kakvu je naviknut, što ponekad stvara poteškoće pri pokušaju pretakanja prvotnih ideja u kod. To je razlog zašto se neki dijagrami prilično razlikuju od prve verzije do druge, pritom prvenstveno misleći na dijagram razreda.

Nadalje, još jedan od izazova je bio i uskladiti sve članove tima za sastanke uživo. Nije nam dugo trebalo da odustanemo od toga i prihvatimo online konferencije i virtualne sastanke kao alternativu.

Također, iako je funkcionalnost slanja e-mail poruka implementirana, u trenutnoj verziji je onemogućena zbog loših iskustava. Naime, probni kreirani račun e-pošte je nakon nekoliko minuta bio blokiran zbog sumnje na SPAM. U stvarnom scenariju najbolje rješenje bilo bi koristiti vlastiti e-mail poslužitelj ili neku od postojećih komercijalnih usluga.

Članovi tima koji su radili na implementaciji stekli su iskustvo rada sa Django frameworkom. Obzirom da je Django jedan od najpopularnijih alata za programiranje web aplikacija, to iskustvo može biti vrlo korisno.

Uz ostala znanja, stečeno je iskustvo rada u timu. Kao posljedica neiskustva voditelja nije primjenjena nijedna formalna metoda upravljanja projektom i timom, što je možda moglo podići kvalitetu proizvoda i ubrzati cijeli projekt.

Pri odabiru tehnologije za implementaciju u obzir su dolazile i one sa kojima su se neki članovi ekipe ranije susreli, poput Ruby on Rails, ali je kolektivna odluka bila odabir nečega novog. Posljedica takve odluke nažalost jest da zasigurno nisu svi dijelovi aplikacije optimalni.

Zadatak je bio specifičan, u tu ruku da je bila konkretno zadana Pizzeria 7,

međutim sve ostvareno može poslužiti kao generička pozadina za bilo koji restoran, lanac restorana, ili čak i više njih, uz minimalne izmjene. Implementacija sustava za jedan restoran nije jako težak zadatak. Ono što ga čini zanimljivijim i zahtjevnim su sve mogućnosti koje se mogu dodati. Neke potencijalne mogućnosti za proširenje koje bi izdvojili su dodavanje *newslettera* za obavijesti o akcijama i popustima, uključivanje *loyalty* programa i daljnja generalizacija u smjeru platforme za naručivanje, poput *Pauze.hr*.

Reference

- ¹ Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
- ² Tigris.org Open Source Software Engineering Tools, <http://readyset.tigris.org/>
- ³ UML 2 Class Diagram Guidelines,
<http://www.agilemodeling.com/style/classDiagram.htm>
- ⁴ Domain Class Diagram Modeling Standards and Guidelines,
<http://www.bced.gov.bc.ca/imb/downloads/classdiagramstandards.pdf>
- ⁵ ArgoUML, <http://argouml.tigris.org/>
- ⁶ UML, http://en.wikipedia.org/wiki/Unified_Modeling_Language
- ⁷ Use case Diagrams, http://en.wikipedia.org/wiki/Use_case
- ⁸ Sequence Diagrams, http://en.wikipedia.org/wiki/Sequence_diagram
- ⁹ Django, <https://www.djangoproject.com/>
- ¹⁰ Python, <https://wiki.python.org/moin/>
- ¹¹ MVC, <http://en.wikipedia.org/wiki/Model-view-controller>
- ¹² SQLite, <http://en.wikipedia.org/wiki/SQLite>
- ¹³ Creately - Diagram Software, <http://creately.com/>
- ¹⁴ Object Diagrams, http://en.wikipedia.org/wiki/Object_diagram

Dodatak A: Dnevnik sastajanja

+-----+
| Dnevnik sastajanja projektne grupe SilkRoad |
+-----+

S1: 06. 10. 2013., 20h, FER, Zgrada E

Prisutni:

Luka Strižić, Belma Gutlić, Petar Šegina, Niko Tilić

Sažetak:

Upoznavanje prisutnih članova tima i razgovor o mogućnostima implementacije (jezici, paketi, biblioteke, frameworkci).

Luka Strižić (voditelj), Belma Gutlić, Matija Srbić, Petar Šegina:

- fokus na implementaciju

Matej Brkašić, Davor Lovrić, Niko Tilić:

- fokus na dokumentiranje

Zaključci:

Luka Strižić deklariran voditeljem. Podjela zadataka na najvišoj razini po vještinama.

S2: 18. 10. 2013., 13h, FER, aula

Prisutni:

Luka Strižić, Matej Brkašić, Belma Gutlić, Davor Lovrić, Matija Srbić, Petar Šegina, Niko Tilić

Sažetak:

Konkretni dogovor o korištenim tehnologijama. Razrada zadataka.

Luka Strižić (voditelj):

- koordiniranje
- opis projektnog zadatka
- arhitektura

Matej Brkašić:

- zahtjevi
- reference

Belma Gutlić:

- zahtjevi
- reference

Davor Lovrić:

- zahtjevi
- rječnik pojmova

Matija Srbić:

- opis projektnog zadatka
- arhitektura

Petar Šegina:

- upoznavanje Django frameworka
- arhitektura

Nika Tilić:

- zahtjevi
- rječnik pojmova

Zaključci:

Započet rad na dokumentaciji. Podjeljeni konkretni zadatci nekim članovim. Inicialne ideje o arhitekturi sustava i baze podataka.

S3: 17. 11. 2013., 18h, Skype

Prisutni:

Luka Strižić, Matej Brkašić, Belma Gutlić, Davor Lovrić, Matija Srbić, Petar Šegina, Nika Tilić

Sažetak:

Rekapitulacija napravljenog i nenapravljenog. Konačna podjela zadataka. Usklađivanje dijagrama i ideja.

Luka Strižić (voditelj):

- koordiniranje
- opis projektnog zadatka
- arhitektura
- ER dijagram

Matej Brkašić:

- use case dijagrami
- sekvencijski dijagrami

Belma Gutlić:

- zahtjevi
- ostali zahtjevi

Davor Lovrić:

- dodavanje referenci
- superskriptiranje riječi sa referencama

Matija Srbić:

- arhitektura
- lektoriranje opisa projektnog zadatka
- dijagram razreda

Petar Šegina

- arhitektura
- objektni dijagrami

Nika Tilić

- rječnik pojmove
- lektoriranje

Zaključci:

Definiranje zadnjih zadataka i završavanje dokumentacije. Ispravljanje typoa, lektoriranje, uređivanje, čišćenje. Sinkronizacija eventualnih neslaganja između pojedinih dijelova.

S4: 26. 12. 2013., Facebook grupa

Prisutni:

Luka Strižić, Matej Brkašić, Belma Gutlić, Davor Lovrić, Matija Srbić, Petar Šegina, Nika Tilić

Sažetak:

Retrospekcija na učinjeno. Planiranje ispravaka i novih zadataka.

Luka Strižić (voditelj):

- koordiniranje
- ispravak ER dijagrama
- nadopuna opisa

Matej Brkašić:

- popravak sekveničkih dijagrama
- dijagram stanja

Belma Gutlić:

- komunikacijski dijagram
- popis korištenih tehnologija

Davor Lovrić:

- dijagram stanja
- dodavanje referenci

Matija Srbić:

- nadopunjavanje dijagrama razreda

Petar Šegina

- implementacija

- nadopuna objektnog dijagrama

Nika Tilić

- rječnik pojmove
- komunikacijski dijagram

Zaključci:

Razrada ideje za dalje. Obzirom da praznici još traju, zadatci vjerojatno neće biti obavljeno do početka 1. mjeseca.