**SVEUČILIŠTE U ZAGREBU**

**FAKULTET ORGANIZACIJE I INFORMATIKE**

**VARAŽDIN**

**Luka Banožić**

**Karlo Batrla**

**Matija Domjan**

**Igor Kežman**

**Deni Slunjski**

**„PROJEKT IZ KOLEGIJA PROGRAMSKI INŽENJERSTVO“**

**-projektna dokumentacija-**

**Varaždin,2013.**

**SVEUČILIŠTE U ZAGREBU**

**FAKULTET ORGANIZACIJE I INFORMATIKE**

**VARAŽDIN**

**Luka Banožić 38234/09-R**

**Karlo Batrla 38192/09-I**

**Matija Domjan 38218/09-R**

**Igor Kežman 38063/09-R**

**Deni Slunjski 38170/09-R**

**„PROJEKT IZ KOLEGIJA PROGRAMSKI INŽENJERSTVO“**

**-projektna dokumentacija-**

**Mentor:**

**Prof. Dr. sc. Vjeran Strahonja**

**Varaždin, lipanj,2013.**

Sadržaj:

[**1.** **Uvod** 2](#_Toc359942954)

[**2.** **Projektni plan i metoda izrade projekta** 3](#_Toc359942955)

[2.1 Metoda izrade projekta 3](#_Toc359942956)

[2.2 Izrada plana u Microsoft Projectu 4](#_Toc359942957)

[2.3 Proračun projekta i ponuda kupcu 8](#_Toc359942958)

[**3** **Uml dijagrami** 10](#_Toc359942959)

[3.3 Uml dijagram slučajeva 10](#_Toc359942960)

[3.2 Uml dijagrami slijeda 12](#_Toc359942961)

[a) Dijagram slijeda za logiranje u sustav 12](#_Toc359942962)

[b) Dijagram slijeda za rezervaciju,pregled,ažuriranje i brisanje termina 13](#_Toc359942963)

[c) Dijagram slijeda za pregled,unos, brisanje i ažuriranje članova 14](#_Toc359942964)

[d) Dijagram slijeda za račun 17](#_Toc359942965)

[3.3 Dijagrami aktivnosti 19](#_Toc359942966)

[a) Dijagram aktivnosti za logiranje u sustav 19](#_Toc359942967)

[b) Dijagram aktivnosti za rezervaciju termina 20](#_Toc359942968)

[c) Dijagram aktivnosti za pregled članova 21](#_Toc359942969)

[e) Dijagram aktivnosti za ispis i pregled računa 22](#_Toc359942970)

[3.4 Dijagram klasa 25](#_Toc359942971)

[4. Era model 27](#_Toc359942972)

1. **Uvod**

U ovom dokumentu nalazi se sva potrebna dokumentacija koja prikazuje način izrade projekta, što se sve radilo u projektu te koji član tima je radio koji dio te koliko se zalagao.

Za početak članovi tima su: Luka Banožić, Karlo Batrla, Matija Domjan, Igor Kežman te Deni Slunjski. Naziv tima je AppGrade. AppGrade je također naziv reporzitorija na kojem se nalazi sve što se radilo tijekom projekta te tko je što radio. Ovaj tim se odlučio da će raditi projekt prema srednjoj razini te će i prema tome biti napravljan. Nakon ovog uvoda prvo će biti izrađen plan projekta u programu Microsoft Project te nakon toga još će biti i dijagram te na kraju pojašnjenje aplikacije te tko je što radio.

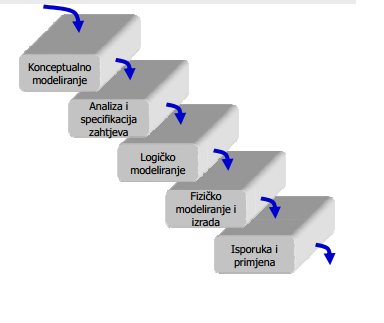
1. **Projektni plan i metoda izrade projekta**

## Metoda izrade projekta

Tema projekta odnosi se na izradu aplikacije za sportski centar. Taj sportski centar ima više terena te razne druge sportske ponude. Mi bismo našom aplikacijom obuhvatili sve te terene,teretanu te ostale popratne sadržaje koje sadrži objekt. Točnije radimo aplikaciju za sportski centar TTS koji sadrži sportske terene, teretanu, vanjske terene te još neke dodatke. Ono što bismo mi omogućili našom aplikacijom je trenutno stanje zauzetosti terena te trenutno stanje oko članarina za teretanu te koliko se korisnika nalazi kod nas te što svaki korisnik ima pravo.

Model koji smo odabrali za izradu aplikacije jest vodopadni model. Koristimo ovaj model jer mislimo da je najprikladniji za naš projekt odnosno da prvo izradimo svu dokumentaciju pa zatim dijagrame prema kojima ćemo razvijati aplikaciju. Nakon što to napravimo idemo na implementaciju aplikacije i na samome kraju njeno testiranje da vidimo što smo dobili.

Po ovim naputcima gore možemo i vidjeti kako slijedno ide vodopadni model a u nastavku možemo vidjeti i sliku.



Slika 1. Vodopadni model

U nastavku još možemo vidjeti i razvojni tim koji je naveden u tabeli te za što je koji član zadužen.

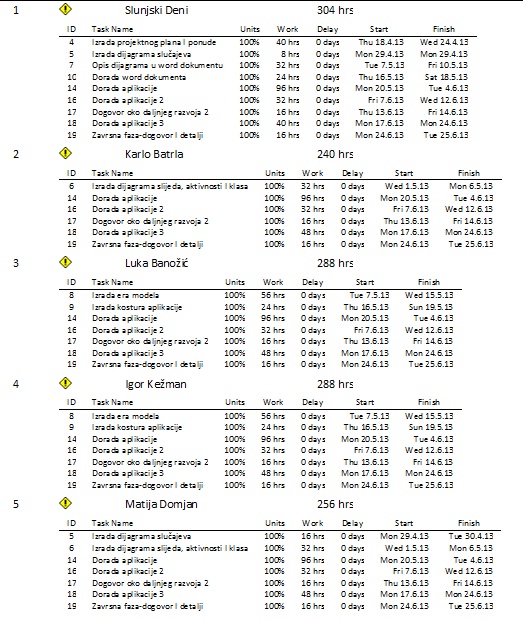
|  |  |
| --- | --- |
| ***Članovi tima*** | ***Opis radnog mjesta*** |
| Luka Banožić | Analiza, izrada i testiranje aplikacije |
| Deni Slunjski | Analiza, izrada i testiranje aplikacije |
| Karlo Batrla | Analiza, izrada i testiranje aplikacije |
| Matija Domjan | Analiza, izrada i testiranje aplikacije |
| Igor Kežman | Analiza, izrada i testiranje aplikacije |

Tablica 1. Prikaz članova projektnog tima

## 2.2 Izrada plana u Microsoft Projectu

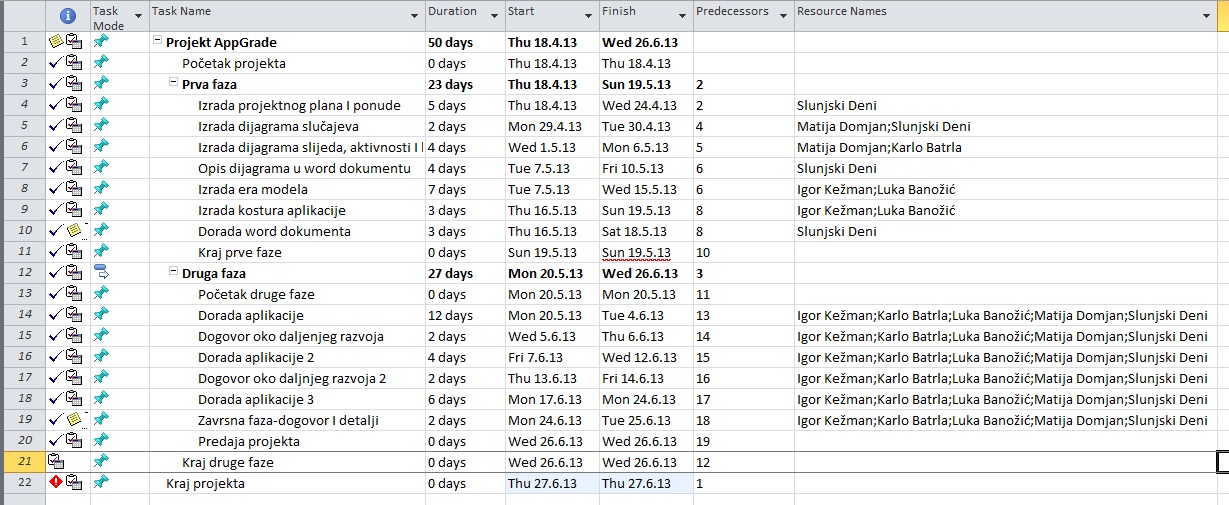
U ovom dijelu projekta prikazan je plan izrade projekta. Odnosno prikazano je tko je zadužen za koji dio projekta te tko je mora što na napraviti te koliko to vremenski traje. U prvoj slici je prikazano do predaje prve faze a kasnije ćemo još prikazati do kraja dokumenta.

Prvo ćemo prikazati tablicu zadataka svakog korisnika i u toj tablici može se vidjeti koliko je ko radio koji dio projekta do sada i koliko bi još trebalo raditi da projekt bude u potpunosti ostvaren. Pa pogledajmo tablicu.



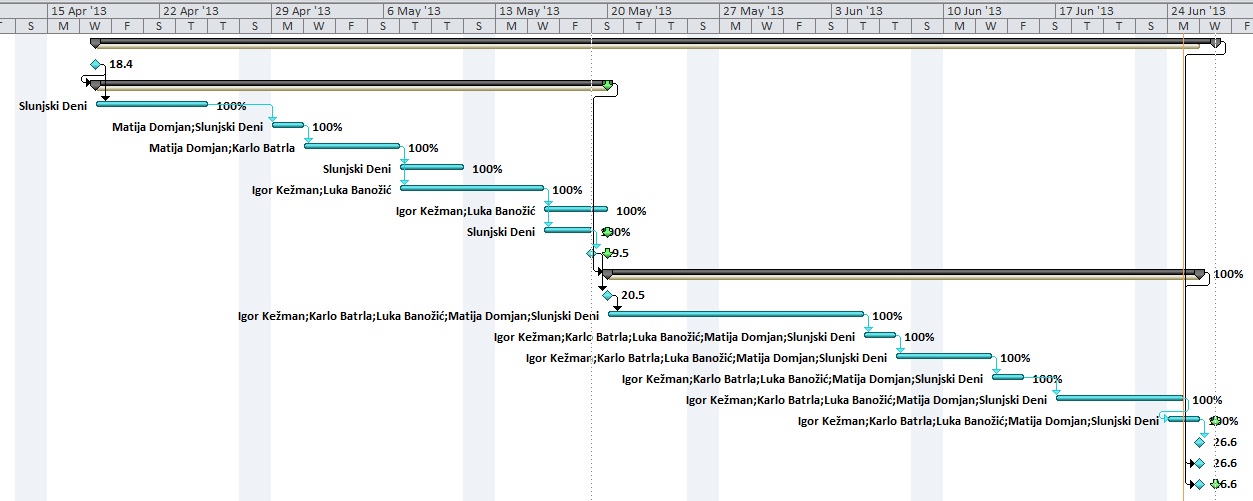
Tablica 2. Prikaz satnice članova

Sljedeći dijagram radio je Slunjski Deni. U nastavku smo napravili još dva dijagrama. Jedan je prikaz zadataka koje smo koristili u projektu. U nastavku ga možemo vidjeti.



Tablica 3. Prikaz zadataka koji su se radili

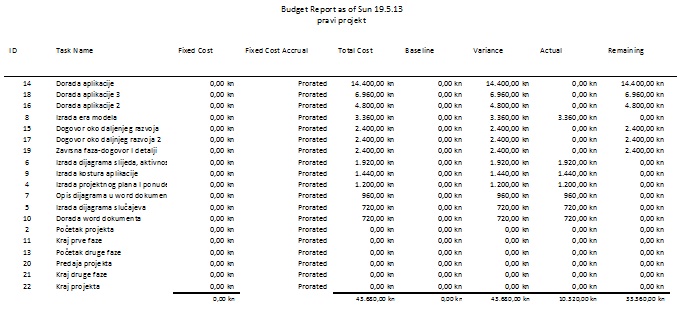
U nastavku možemo vidjeti i gantogram zadataka koji su se radili te kako je tekao projekt do sada te kako bi treba teći.



Gantogram 1. Prikaz aktivnosti na gantogramu

## Proračun projekta i ponuda kupcu

Ova dio projekta te isto tako i sve što je rađeno u Microsoft Projectu radio je Deni Slunjski. U ovom dijelu prikazati ćemo procjenu proračuna projekta pomoću programa Microsoft Projecta. Kad smo unijeli satnicu rada koju smo mogli vidjeti gore program je izbacio sljedeće rezultate.



Tablica 4. Prikaz procjene projekta

 AppGrade d.o.o. Zagrebačaka 170,

42000 Varaždin

Ponuda broj: 1.

Naručitelj: **SPORTSKI CENTAR - centar za sport, rekreaciju i rehabilitaciju**

Red. Br. Naziv Cijena

1. Izrada dokumentacije 10.320,00 kn

2. Izrada aplikacije 33.600,00 kn

Ukupno: 43.680,00 kn

Potpis

1. **Uml dijagrami**
   1. Uml dijagram slučajeva

U ovom poglavlju smo napravi dijagram slučajeva. Ovaj dijagram izradili su Matija Domjan te Slunjski Deni.

## Prvi dijagram1.jpg

Slika 2. Dijagram slučajeva

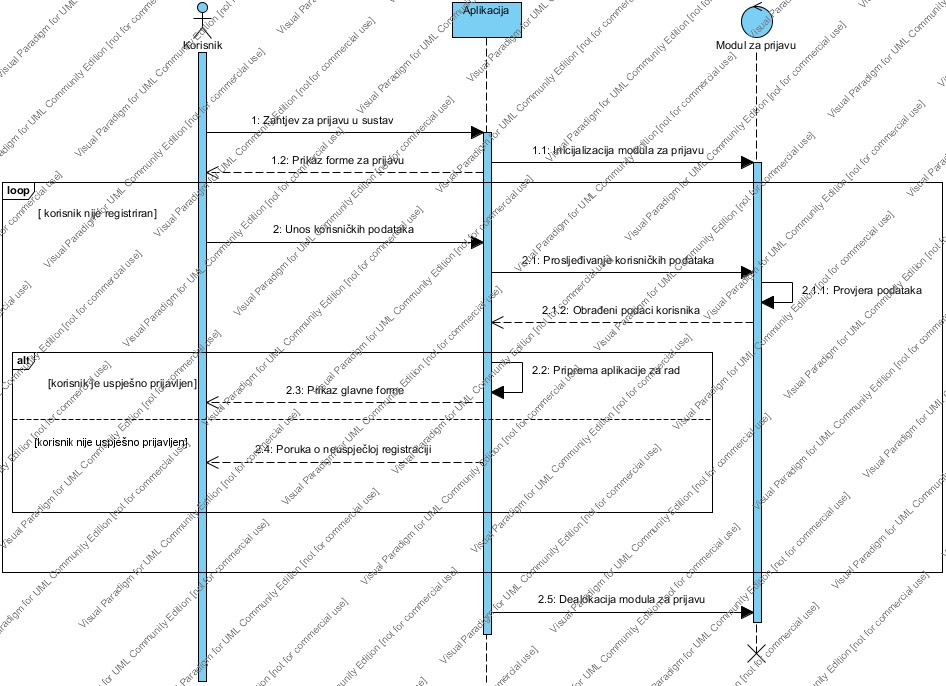
U dijagramu slučajeva možemo vidjeti kakve sve zadatke mora posjedovati aplikacija te se vidi koji sve članovi sudjeluju u aplikaciji te što sve oni mogu napraviti.

Tako u našoj aplikaciji postoje dvije strane, korisnik aplikacije te sustav.

Korisnik je onaj koji se služi aplikacijom odnosno on kontrolira rezervacije, unosi rezervacije, naplaćuje račune, ima pregled zauzetih terena, može pregledati postojeće članove, može dodavati nove članove i također vidi obavijesti aplikacije odnosno to znači da sustav sam prilikom neke greške obavijesti korisnika te također ako nekom članu ističe rezervacija sustav podsjeća korisnika da se to događa, također isto tako i za rezervacije terena. Sustav obavještava tko nije platio neku stavku ili kome ističe rezervacija terena. Također dodatak aplikaciji jest ako sustav dvoje ljudi se pokušava unositi podatke u sustav u isto vrijeme aplikacija upozorava drugog korisnika da on to ne može odnosno da ne može unijeti pod tim brojem jer neko drugi trenutno to rezervira. O tim detaljima ćemo kasnije. Također korisnik ima opciju izdavanja računa te naplate.

## 3.2 Uml dijagrami slijeda

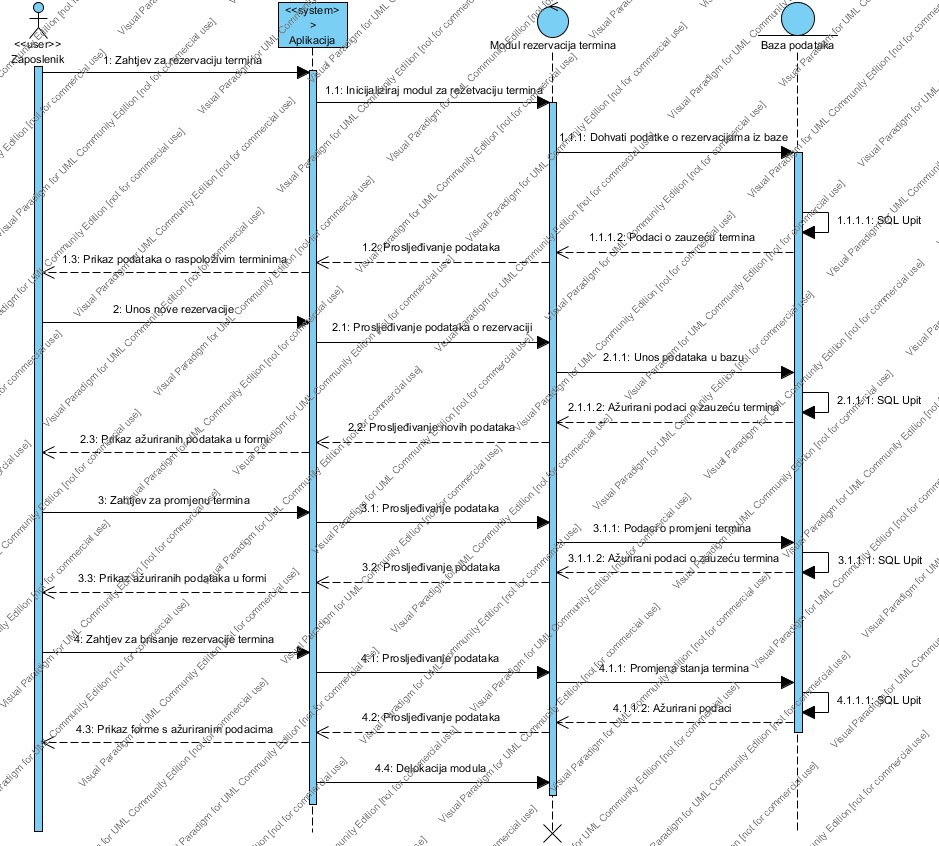
1. Dijagram slijeda za logiranje u sustav



Slika 2. Dijagram slijeda za logiranje u sustav

Dijagram slijeda za logiranje u sustav napravili su Karlo Batrla te Matija Domjan a Deni Slunjski ga je opisao. U ovom dijagramu vidimo na jednoj strani korisnika a na drugoj aplikaciju. Korisnik šalje zahtjev za prijavu u sustav. Aplikacija mu prvo prikazuje formu te inicijalizira modul za prijavu. Korisnik unosi podatke u formu. Aplikacija zatim šalje podatke modulu za logiranje, podaci se provjeravaju te se obrađeni tako šalju aplikaciji. Aplikacija radi formu za rad i prikazuje aplikaciju korisniku. Ukoliko se korisnik nije uspješno prijavio vraća mu se poruka o neuspješnoj prijavi.

1. Dijagram slijeda za rezervaciju,pregled,ažuriranje i brisanje termina



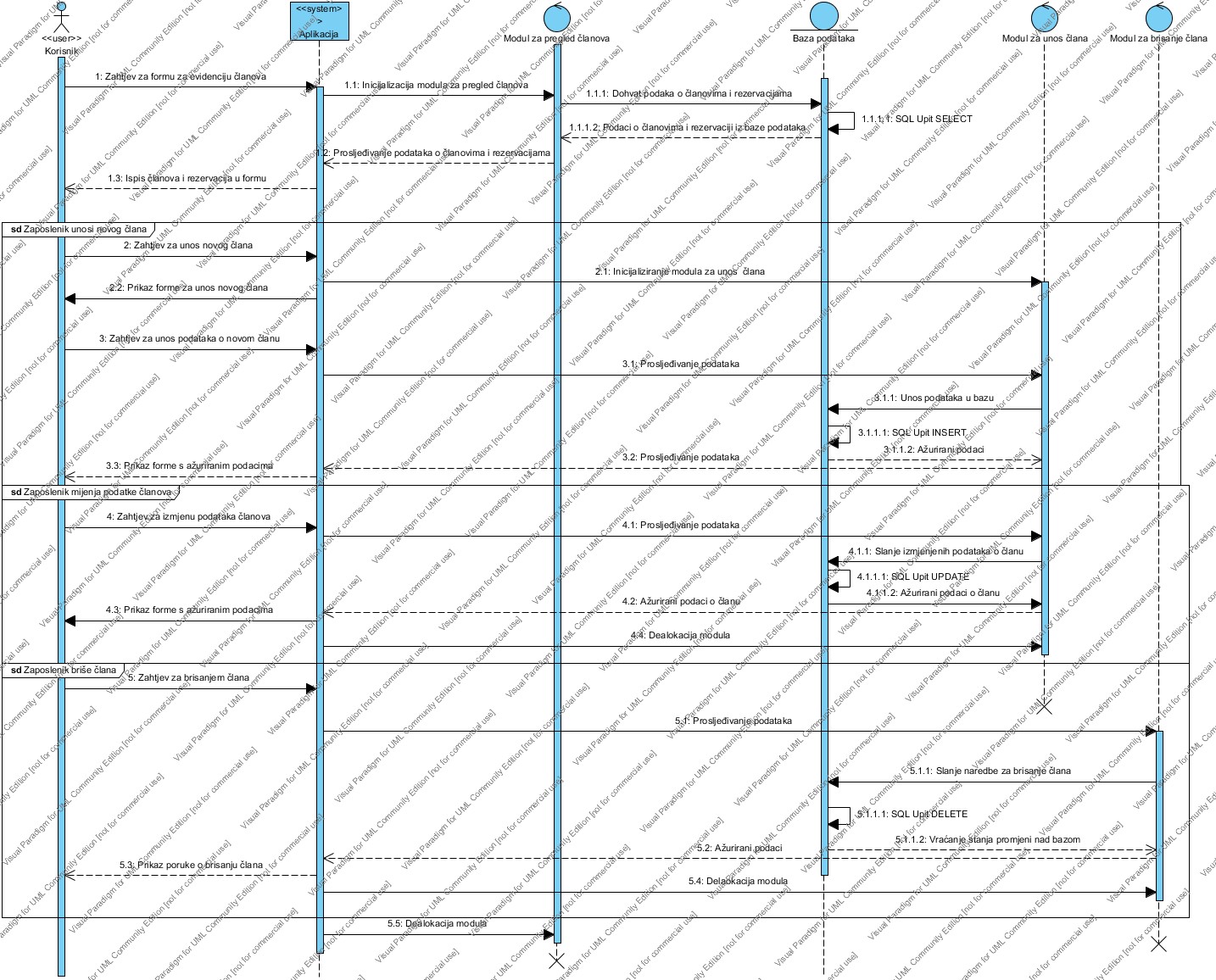
Slika 3. Dijagram slijeda za rezervaciju termina

Ovaj dijagram su napravili Karlo Batrla te Matija Domjan a Slunjski Deni je opisao.

Drugi dijagram slijeda koji ćemo opisati je dijagram za rezervaciju termina. Sa jedne strane imamo korisnika aplikacije a sa druge strane nam je aplikacija. Prva radnja koju radi korisnik jest zahtjev za rezervacijom termina koja se šalje od korisnika prema aplikaciji. To inicira u aplikaciji modul za rezervaciju termina . Unutar aplikacije model za rezervaciju termina inicira dohvat podataka iz baze. Baza obrađuje podatke i vraća o zauzeću termina u model. Model prosljeđuje podatke do aplikacije koja šalje prikaz podataka o raspoloživosti termina. Kad korisnik vidi da je raspoloživo šalje zahtjev za unos nove rezervacije. Aplikacija prosljeđuje podatke do modela koji unosi podatke u bazu. Podaci se u bazi obrade i šalju povratnu poruku o ažuriranosti sve do korisnika. Korisnik također može poslati zahtjev za promjenu termina. Aplikacija prosljeđuje podatke i podaci preko modela dolaze do baze podataka gdje se opet obrađuju. Podaci o zauzeću ponovo se prosljeđuju i aplikacija ih prikazuje korisniku. Zadnja radnja koju korisnik može napraviti jest brisanje rezervacije termina. Podaci se prosljeđuju na model koji šalje podatke u bazu. U bazi se rješava upit i kad su podaci izbrisani baza ponovo šalje podatke u ažuriranosti baze. Na kraju korisnik vidi da su podaci izbrisani.

1. Dijagram slijeda za pregled,unos, brisanje i ažuriranje članova

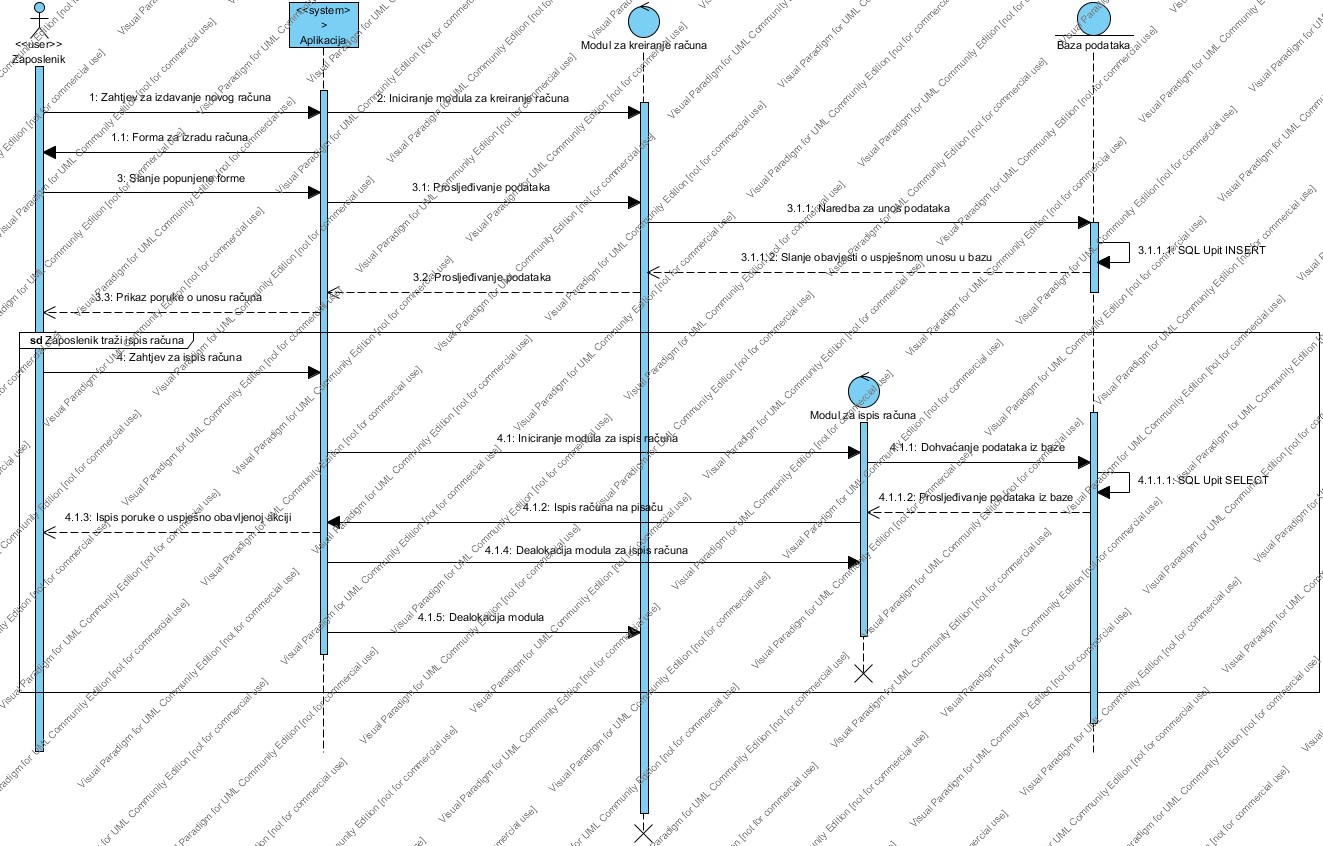
Sljedeći dijagram je za pregled svih članova. U ovom dijagramu ćemo vidjeti kako korisnik aplikacije može pregledati sve članove koji se nalaze u bazi podataka.



Slika 4. Dijagram slijeda za pregled članova

Možemo vidjeti da je na ovaj dijagramu stavljen i pregled ,unos,brisanje te ažuriranje članova. U početku korisnik šalje zahtjev za evidenciju članova. Aplikacija inicijalizira modul za pregled članova. Modul za pregled dohvaća iz baze podatke o članovima te ih ponovo vraća aplikaciji na pregled korisniku. Kad korisnik šalje zahtjev za unos članova, aplikacija inicijalizira modul za unos članova koji unose podatke u bazu. Provodi se sql upit insert i ažuriraju se podaci. Tada baza podataka vraća ažurirane podatke modulu za unos i prikazuje ih korisniku. Ako korisnik želi promijeniti podatke o članu ponovo pokreće aplikaciju modul za unos članova koja komunicira za bazom i mijenja podatke koje korisnik kasnije dobiva natrag u izmijenjenom stanju. Ako korisnik želi obrisati člana šalje modulu za brisanje člana naredbu da se izbriše član. Tada modul stupa u kontakt sa bazom podataka i briše člana u bazi podataka. Poruka o izbrisanom članu se vraća modulu koji šalje poruku aplikaciji. I na kraju korisnik može vidjeti poruku da su podaci izbrisani. Na kraju aplikacija radi dealokaciju modula za pregled i modula za brisanje članova.

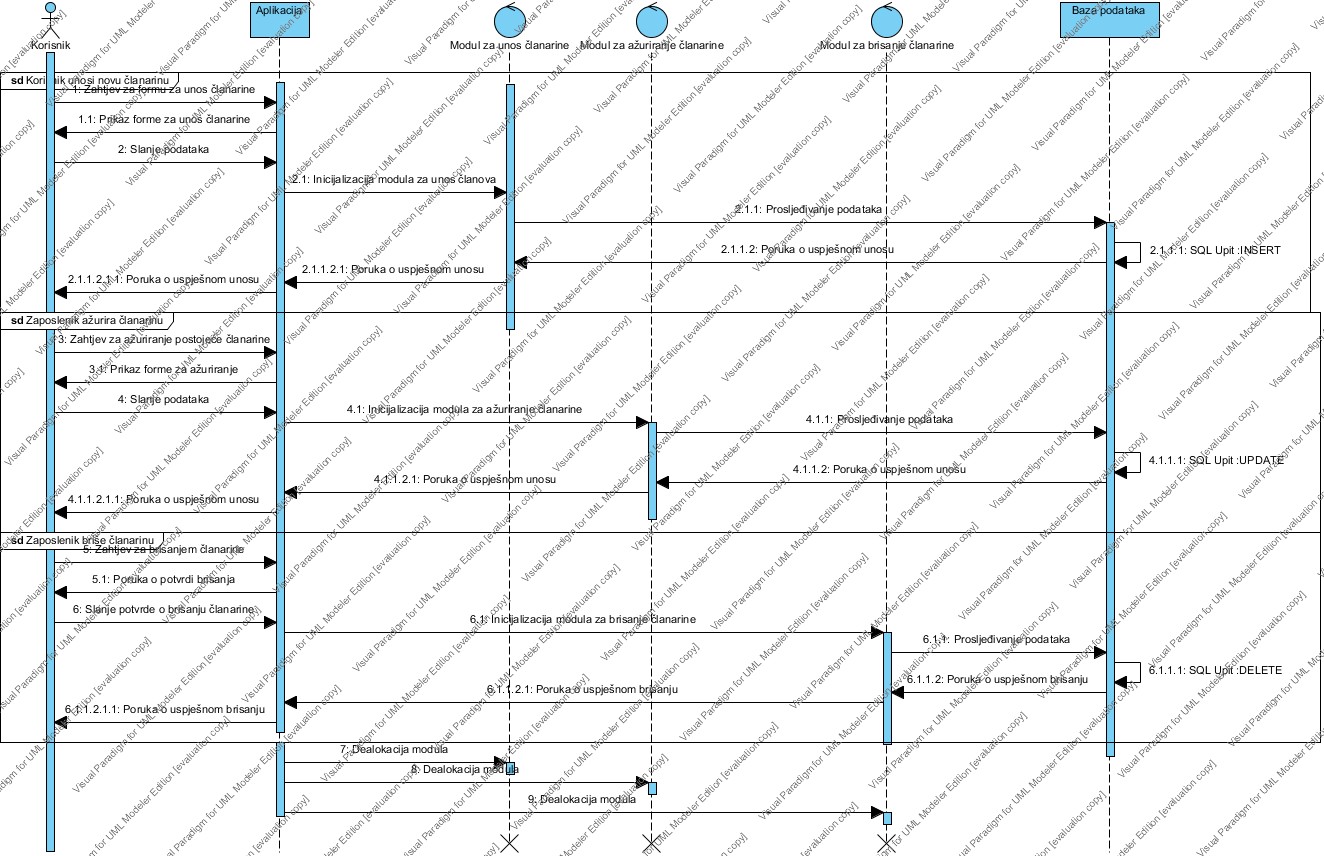
1. Dijagram slijeda za račun



Slika 6. Dijagram slijeda za račun

Dijagram slijeda za račun nam prikazuje na koji način sustav izrađuje račun. Prvo korisnik šalje zahtjev za izdavanje novog računa prema aplikaciji. Aplikacija inicira novi modul za kreiranje računa. Forma se vraća korisniku te korisnik popunjava formu. Kad je forma popunjena šalju se podaci prema modulu. Modul šalje podatke bazi gdje ih baza obradi i spremi i šalje natrag modulu poruku o spremljenom računu. Tada aplikacija pokazuje korisniku ove podatke. Kad je sve to obavljeno korisnik šalje zahtjev za ispisom računa. Tada aplikacija inicijalizira modul za ispis računa koji onda šalje zahtjev bazi podataka. Baza podataka obrađuje upit te vraća podatke modulu za ispis računa. Račun se ispisuje te se šalje povratna poruka korisniku da je račun uspješno ispisan. Na kraju aplikacija gasi oba dva modula koji su na bili potrebni.

### e) Dijagram slijeda za članarine

****

Slika 7. Dijagram slijeda za članarine

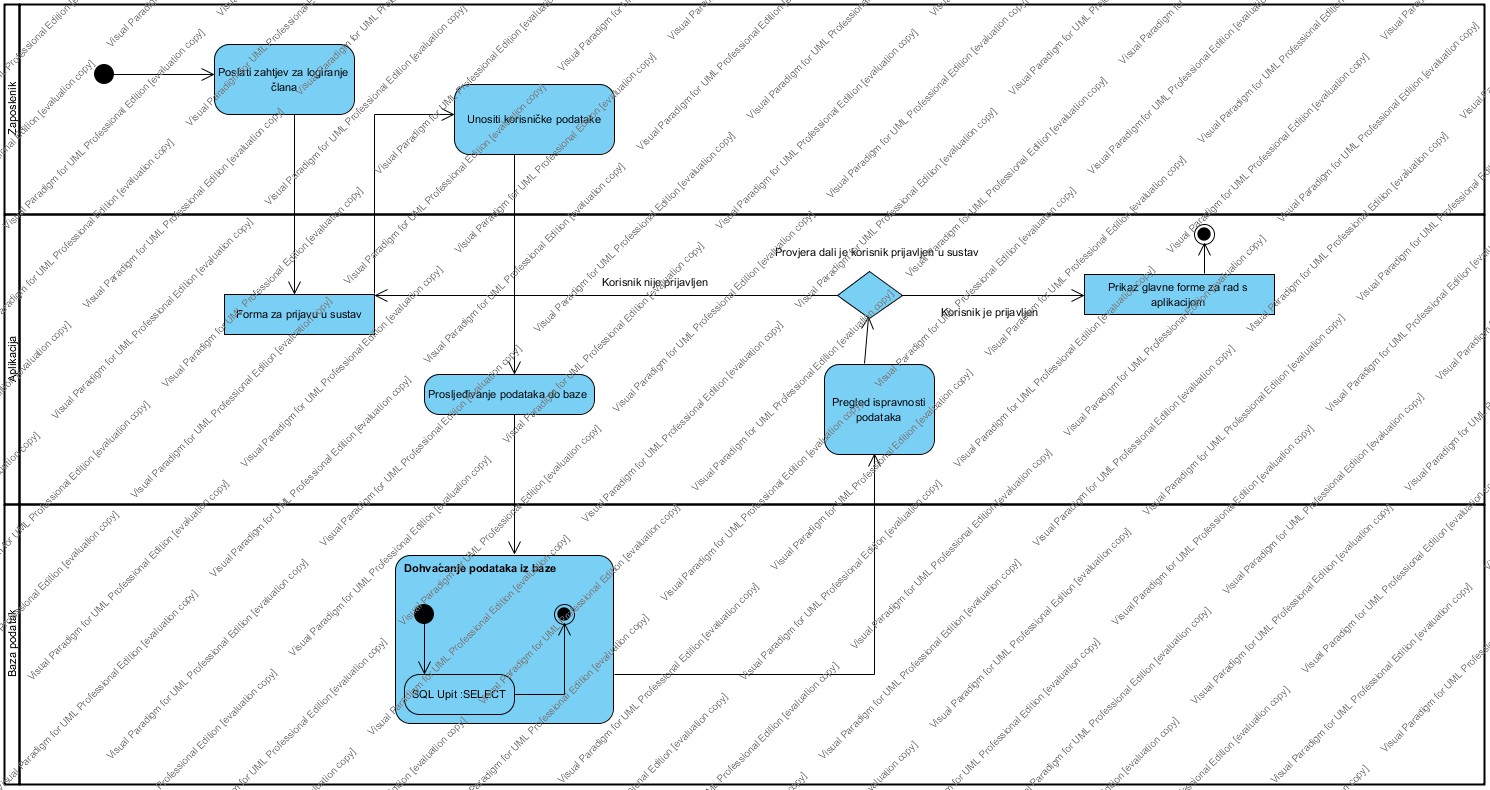
Zaposlenik šalje zahtjev našoj aplikaciji za unos novog člana,te aplikacija prikazuje formu te korisnik popunjava formu s podacima novog člana. Dobivene podatke,modul za unos članarine prosljeđuje podatke do baze podataka. Proslijeđeni podaci se SQL naredbom : “insert“. Nakon što modul za bazu podatak izvrši ovaj upit prema korisniku se prosljeđuje poruka za uspješan unos novog člana.

Ažuriranje člana započinje korisničkim zahtjevnom za ažuriranjem,također se prikazuje forma kja omogućava ažuriranje, te korisnik unaša nove,ažurirane podatke. Podaci se prosljeđuju modulu aplikacije koji omogućava ažuriranje,te te iste podatke prosljeđuje u bazu podataka. Baza izvršava SQL naredbu : “update“,te korisniku prosljeđuje poruku o uspješnom ažuriranju članova.

Na isti način djeluje i slučaj prilikom brisanja člana, ali jednine razlike koje se koriste su upotreba modula za brisanje člaana, te SQL naredba: „delete“, također, nakon izvršavanja SQL naredbe aplikacija prikazuje poruku o uspješnom brisanju člana.

## Dijagrami aktivnosti

1. Dijagram aktivnosti za logiranje u sustav

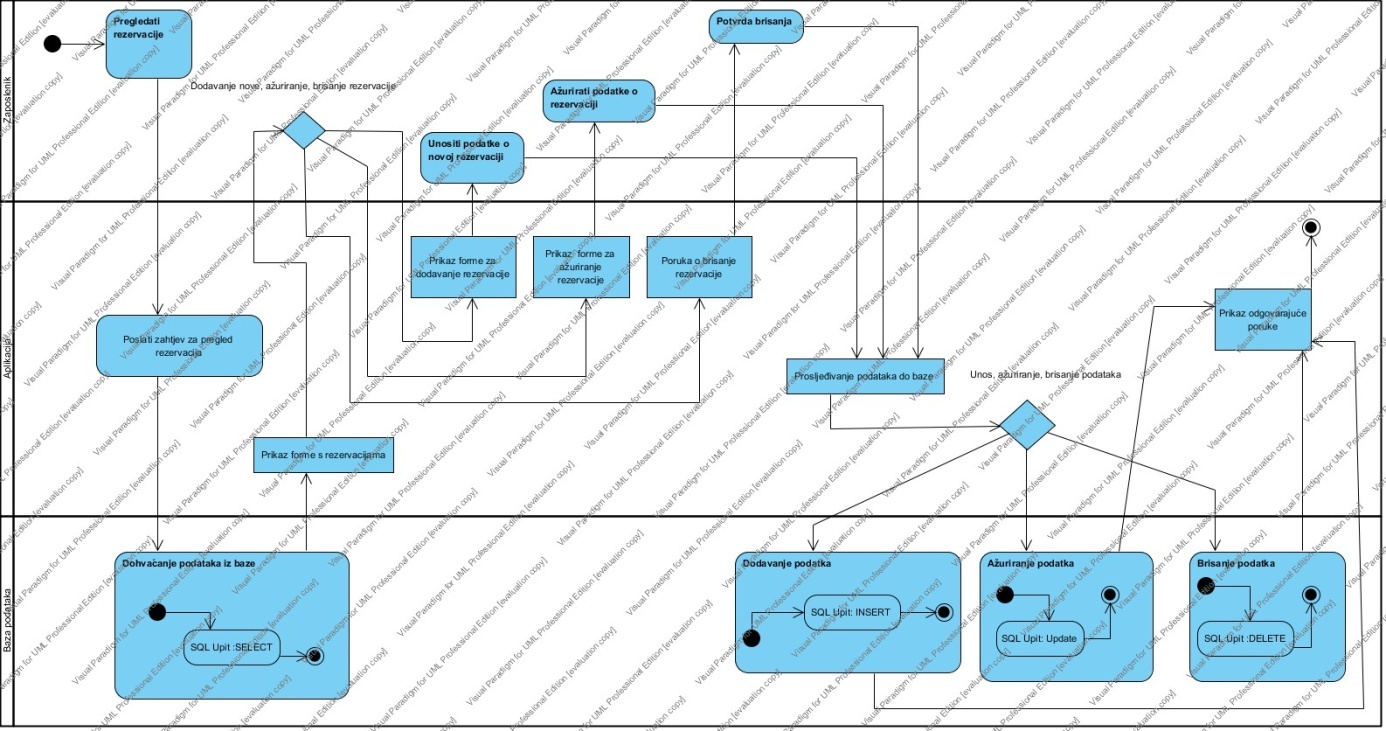


Slika 8. Dijagram aktivnosti za logiranje u sustav

Dijagram aktivnosti za logiranje u sustav napravili su Karlo Batrla i Matija Domjan a opisao je Deni Slunjski.

Iz dijagrama možemo vidjeti da korisnik šalje zahtjev za logiranje aplikaciji, aplikacija zatim pokreće formu za unos podataka. Nakon toga korisnik unosi podatke u formu i ti podaci se šalju bazi. Baza zatim provjerava točnost podataka i ako su oni točni pokreće se aplikacija a ako nisu točni aplikacija ispisuje da podaci nisu točni te ponovo vraća formu za unos.

1. Dijagram aktivnosti za rezervaciju termina

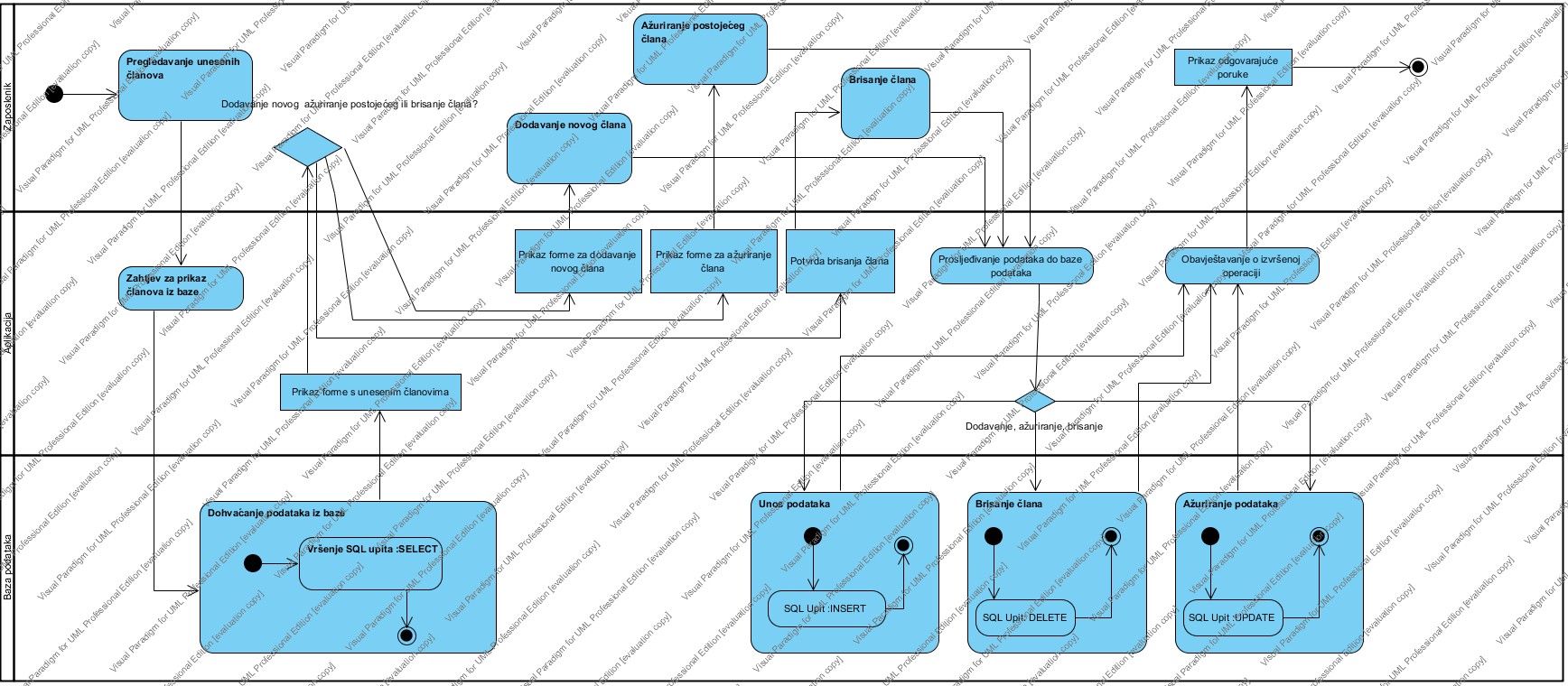


Slika 9. Dijagram aktivnosti za rezervaciju termina

Dijagram aktivnosti nam prikazuje koje sve radnje se rade prilikom određenog rada sa aplikacijom.

Zaposlenik može pregledati sve rezervacije koja su do sad napravljene. Aplikacija mu prikazuje formu sa svim rezerviranim terminima. Ako korisnik želi unijeti novu rezervaciju prvo šalje zahtjev aplikaciji. Zatim aplikacija pokreće formu za unos nove rezervacije. Podaci koji su uneseni u formu zatim idu u bazu gdje se oni ubacuju u samu bazu i spremaju pomoću sql upita. Zatim baza obavještava aplikaciju o izvršenoj radnji te aplikacija na kraju prikazuje poruku korisniku da je radnja uspješno obavljena.

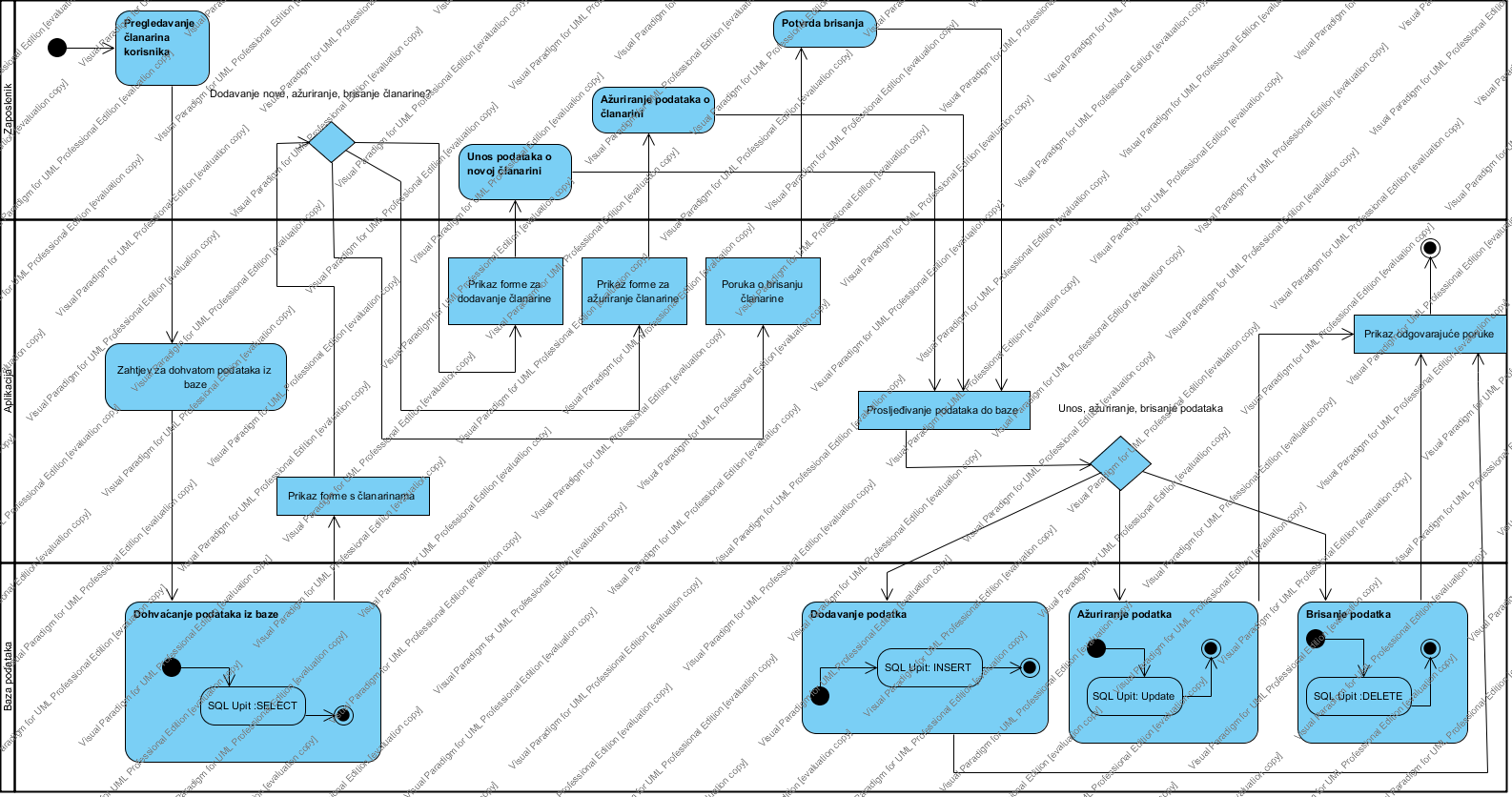
1. Dijagram aktivnosti za pregled,unos i brisanje članova



Slika 10. Dijagram aktivnosti za pregled, unos i brisanje članova

Dijagram aktivnosti za pregled članova nam opisuje radnje za pregled članova. Prva radnja je kad korisnik šalje zahtjev za pregled članova. Od te točke šalje se zahtjev aplikaciji koja šalje sql upit bazi podataka. Baza podataka obrađuje sql upit te šalje podatke aplikaciji. Aplikacija prima podatke i obrađuje da li postoji traženi član, ako ne postoji šalje se poruka korisniku koji može ponovo poslati zahtjev a ako postoji onda se šalje podaci prema korisniku odnosno ispisuju mu se na ekranu da on može vidjeti. Aplikacija zatim prikazuje formu sa unesenim članovima. Nakon toga korisnik može odabrati želi li dodati novog člana, obrisati ili ažurirati postojećeg. Ako želi dodati novoga aplikacija prikazuje formu za dodavanje novog člana. Podaci se tada preko aplikacije prosljeđuju u bazu gdje se oni spremaju i vrača se obavijest o izvršenim radnjama. Također ako želi ažurirati isti je postupak. Kod brisanja člana korisnik ga označi i šalje se informacija od aplikacije prema bazi gdje baza briše korisnika u bazi.

1. Dijagram aktivnosti za unos,brisanje i ažuriranje članarina

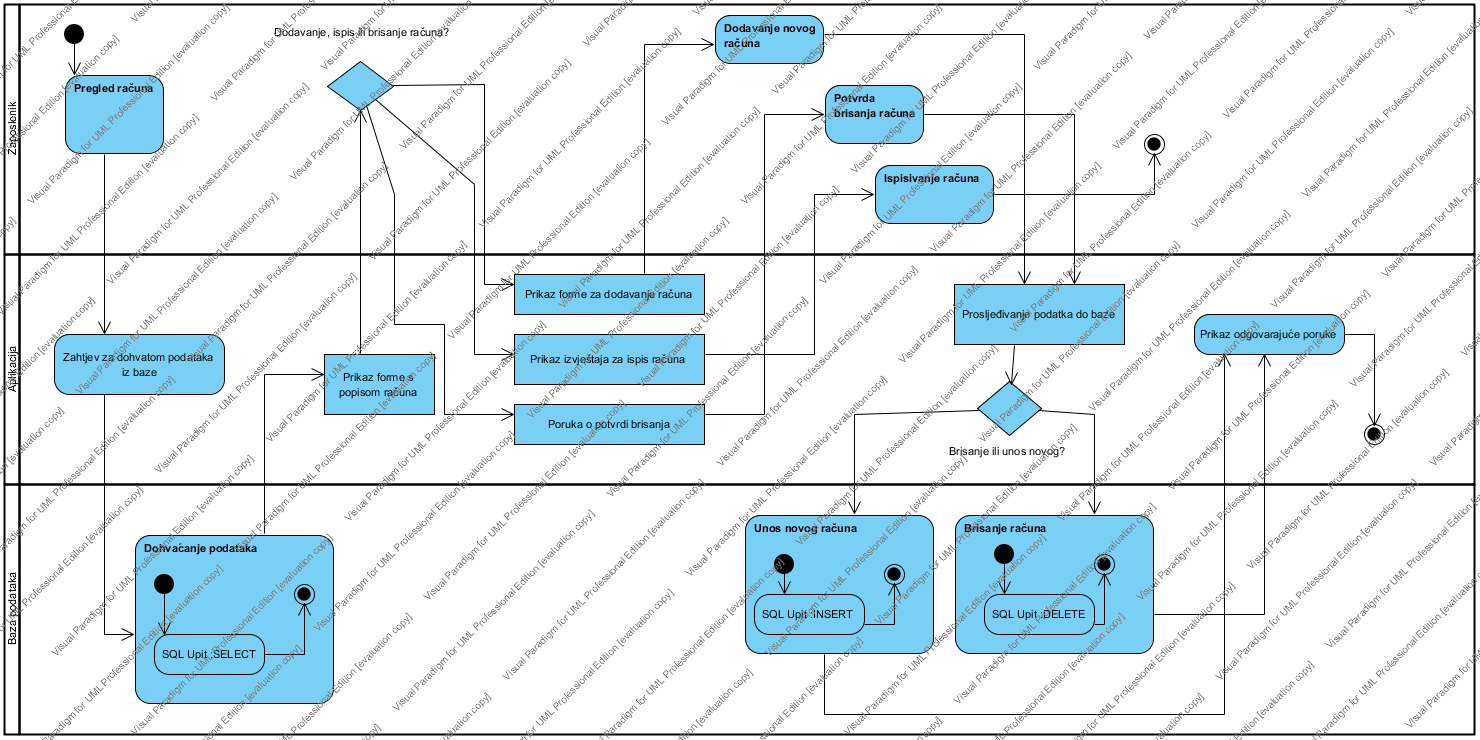


Slika 11. Dijagram aktivnosti za unos, brisanje i ažuriranje članarina

Korisnik prvo šalje zahtjev za otvaranje pregleda članarina. Zahtjev se šalje aplikaciji gdje aplikacija šalje zahtjev bazi. Baza vraća sve članarine i aplikacija ih prikazuje na formi. Tada korisnik donosi odluku koju radnju želi raditi. Ako želi unositi nove članarine aplikacija mu otvara formu za novi unos. Korisnik popunjava formu te šalje zahtjev aplikaciji za unosu bazu. Aplikacija komunicira sa bazom te sprema nove podatke u bazu. Baza vraća potvrdi odgovor da je sve u redu. Također isto tako je i za ažuriranje. Ako korisnik želi ažurirati neku članarinu aplikacija mu otvara novu formu za ažuriranje. Korisnik mijenja podatke te šalje zahtjev aplikaciji. Aplikacija komunicira sa bazom te sprema nove podatke u bazu. Ako korisnik želi obrisati neku članarinu otvara mu se forma za brisanje. Tamo odabire kojega želi obrisati. Ponovo šalje zahtjev aplikaciji a aplikacija komunicira sa bazom. Ako je ve dobro provedeno u bazi, baza vraća odgovor aplikaciji da je sve u redu.

1. Dijagram aktivnosti za ispis i pregled računa

Ovaj dijagram izradili su Karlo Batrla te Matija Domjan. Deni Slunjski je opisao. U nastavu možemo na drugo strani vidjeti dijagram.



Slika 12. Dijagram aktivnosti za izradu računa

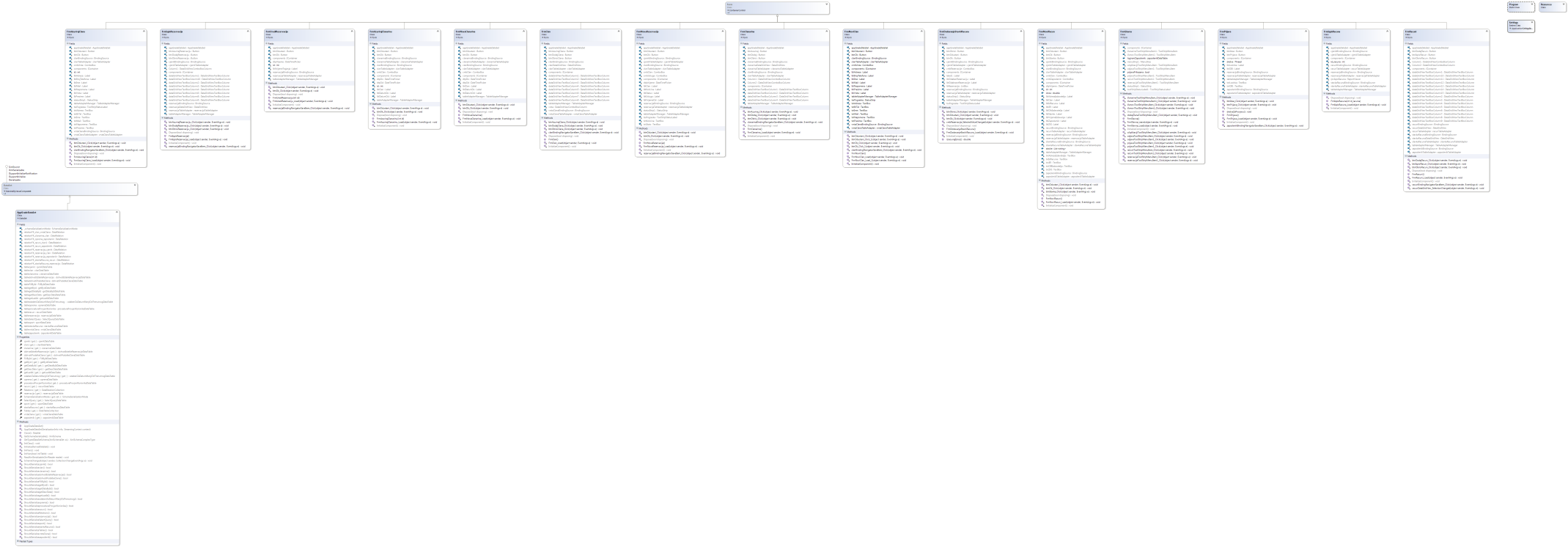
Na gornjem dijagramu aktivnosti možemo vidjeti koje se sve aktivnosti izvode prilikom ispisa računa. Prvo korisnik otvara formu za račun te je zatim šalje zahtjev aplikaciji. Aplikacija šalje bazi upit o stavkama računa te baza obrađuje taj zahtjev pa ponovo šalje aplikaciji koja onda te stavke pri kazuje korisniku. Korisnik bira potrebne stavke te unosi količine koje su mu potrebne. Zatim se sve to šalje aplikaciji koja računa stavke računa. Tako generiran račun se šalje na pregled korisniku. Korisnik zatim traži ispis računa te šalje to aplikaciji. Aplikacija prvo sprema račun u bazu a zatim šalje račun na printer za ispis.

* 1. Dijagram klasa

U nastavku slijedi dijagram klasa koji su napravili Karlo Batrla i Matija Domjan dok su era model napravili Igor Kežman i Luka Banožić. Također oni su napravili i kostur aplikacije. Slunjski Deni je zaslužan za opis era modela.

Dijagram klasa smo prikazali uz pomoć veze generalizacije,odnosno prikazali smo način nasljeđivanja klasa,te njihovu hijerarhiju. Generalizacija je veza između klasa u kojoj klasa dijete nasljeđuje metode i svojstva od klase roditelja. Ovakav prikaz je jedan od najjednostavnijih prikaza dijagrama klasa. Dobar primjer nasljeđivanja u našem slučaju jest da forma *Form* je nadklasa dok ostale forme nasljeđuju svojstva te klase. Također,ovim dijagramom možemo vidjeti metode svake klase, te njihova svojstva

.



# 4. Era model

