

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ORGANIZACIJE I INFORMATIKE**  
**V A R A Ź D I N**

**Marko Alerić**  
**Mislav Boras**  
**Andrea Pleše**  
**Lovro Predovan**  
**Josip Primorac**

**driveIT**

**PROJEKTNA DOKUMENTACIJA IZ KOLEGIJA**  
**PROGRAMSKO INŽENJERSTVO**

**Varaždin, 2014.**

**SVEUČILIŠTE U ZAGREBU**  
**FAKULTET ORGANIZACIJE I INFORMATIKE**  
**V A R A Ž D I N**

**Marko Alerić, 39908/11-R, redovni student, Informacijski sustavi**  
**Mislav Boras, 39920/11-R, redovni student, Poslovni sustavi**  
**Andrea Pleše, 40070/11-R, redovni student, Poslovni sustavi**  
**Lovro Predovan, 40075/11-R, redovni student, Informacijski sustavi**  
**Josip Primorac, 40077/11-R, redovni student, Informacijski sustavi**

**driveIT**

**PROJEKTNA DOKUMENTACIJA IZ KOLEGIJA**  
**PROGRAMSKO INŽENJERSTVO**

**Mentor:**

Doc.dr.sc. Zlatko Stapić

**Varaždin, srpanj 2014.**

# Sadržaj

1. Uvod .....	1
2. Specifikacija zahtjeva .....	2
2.1. Uvod .....	2
2.1.1. Svrha dokumenta .....	2
2.1.2. Opseg aplikacije .....	2
2.1.3. Ciljana publika i pregled dokumenta .....	2
2.1.4. Reference .....	2
2.2. Općeniti opis .....	3
2.2.1. Perspektiva proizvoda .....	3
2.2.2. Funkcionalnost proizvoda .....	3
2.2.3. Korisnici i njihove karakteristike .....	4
2.2.4. Radno okruženje .....	4
2.2.5. Ograničenja dizajna i implementacije .....	4
2.2.6. Korisnička dokumentacija .....	4
2.2.7. Pretpostavke i zavisnosti .....	5
2.3. Specifični zahtjevi .....	5
2.3.1. Zahtjevi vanjskog sučelja .....	5
2.3.1.1. Korisnička sučelja .....	5
2.3.1.2. Sučelja hardvera .....	6
2.3.1.3. Sučelja softvera .....	6
2.3.1.4. Sučelja komunikacije .....	6
2.3.2. Zahtjevi ponašanja .....	6
2.4. Drugi ne-funkcionalni zahtjevi .....	6
2.4.1. Zahtjevi s obzirom na performanse .....	6
2.4.2. Zahtjevi sigurnosti .....	7
2.4.3. Atributi kvalitete softvera .....	7
3. Projektna dokumentacija .....	8
3.1. Projektni tim .....	8
3.2. Terminski plan projekta .....	10
3.3. Proračun i budžet projekta .....	13
3.4. Ponuda naručitelju .....	14
4. Tehnička dokumentacija .....	16
4.1. Dijagram slučajeva korištenja .....	16
4.2. Dodavanje novog vozila .....	17
4.3. Prijava u sustav .....	18

4.4. Dodavanje dobavljača .....	18
4.5. Nalog za popravak .....	19
4.6. Paljenje vozila koja dugo nisu paljena .....	20
4.7. Dijagrami slijeda .....	21
4.7.1. Dijagram slijeda – Prijava u aplikaciju .....	21
4.7.2. Dijagram slijeda – Dobavljači .....	22
4.7.3. Dijagram slijeda – Vozila .....	24
4.7.4. Dijagram slijeda – Popravka i kvarova .....	26
4.7.5. Dijagram aktivnosti – Popravka i kvarova .....	27
4.7.6. Dijagram slijeda – Ugovori .....	28
4.8. Dijagram klasa .....	30
4.9. Era model .....	33

# 1. Uvod

Aplikacija „driveIT“ napravljena je kako bi pomogla autosalonima rabljenih vozila oko evidencije stanja vozila u njihovom „dvorištu“. Vozila je potrebno pratiti od početka kupnje od dobavljača pa sve do prodaje i naplate. Vozila mogu biti u različitim stanjima i sa različitom dodatnom opremom stoga je u ovoj aplikaciji moguće za svako vozilo vidjeti njegove specifikacije. Također, vozila mogu biti i neispravna stoga je potrebno pratiti njihove troškove popravka i izračunati minimalnu prodajnu cijenu koja premašuje iznos nabavne cijene i dodatnih troškova. Osim vozila, vodila bi se i evidencija o dobavljačima i kupcima te bi se onda ugovori o kupnji i podaju vozila mogli generirati na osnovu podataka o vozilu u kupcu/dobavljaču za koje se generiraju. Na sučelju aplikacije nalazio bi se prikaz tlocrta određenog autosalona rabljenih vozila za lakši pregled raspoloživih vozila.

## **2. Specifikacija zahtjeva**

### **2.1. Uvod**

#### **2.1.1. Svrha dokumenta**

Svrha ovog dokumenta je da specificira zahtjeve koje cjelokupna aplikacija „driveIT“ treba ispuniti prema svojim korisnicima. Prvenstveno je potrebno definirati namjenu aplikacije. driveIT služi za evidenciju rabljenih vozila u autosalonima te ostalih popratnih podataka poput kupaca i dobavljača. Svrha aplikacije je omogućiti dodavanje i uređivanje podataka o vozilima u dvorištu, evidenciju prodanih vozila, izradu ugovora i njihovo pohranjivanje. Ciljani korisnici su prvenstveno zaposlenici u autosalonima, tj. prodavači i mehaničari te sami vlasnici salona.

#### **2.1.2. Opseg aplikacije**

Najbolji način da definirano opseg aplikacije jest da navedemo prednosti koje ona donosi i ciljeve koje ispunjava. Prednosti se ogledaju u mogućnosti uvida u stanje vozila u svakom trenutku te praćenju promjena koje nastaju nabavkom ili prodajom. Osim evidencije, olakšana je izrada ugovora s kupcima i dobavljačima jer su svi evidentirani u bazi podataka i omogućeno je generiranje ugovora. Mehaničarima ova aplikacija olakšava raspored radnog vremena jer imaju popis svih vozila koje treba pregledati ili popraviti. Ono što aplikacija ne nudi je izradu konačnog računa, za što je zadužen blagajnički sustav.

#### **2.1.3. Ciljana publika i pregled dokumenta**

Ciljana publika ovog dokumenta su budući korisnici sustava koji uvjetuju zahtjeve. Osim njih dokument čitaju i programeri zaslužni za izradu i testiranje, menadžeri projekta te osobe zadužene za izradu dokumentacije. Ostatak specifikacija zahtjeva je organiziran na način da prvo prikazuje perspektivu i funkcionalnost proizvoda, potom korisnike i njihove karakteristike, pa radno okruženje i ograničenja. Na kraju se nalaze specifični zahtjevi korisničkog sučelja, hardverskog i softverskog te komunikacijskog sučelja. Osim njih tu su sigurnosni i zahtjevi performansi.

#### **2.1.4. Reference**

Dokument je pisan po zahtjevima propisanim u „IEEE Std 830-1998, Software Requirements Specifications“ dokumentu.

## **2.2. Općeniti opis**

### **2.2.1. Perspektiva proizvoda**

Aplikacija driveIT se izrađuje kao novi samostalni sustav bez integracije s nekim drugim sustavima. Jedino postoji mogućnost integracije s blagajničkim sustavom poduzeća, ali to neće biti omogućeno u ovoj verziji sustava. Aplikacija jedino zahtjeva MS SQL Server bazu podataka na kojoj će pohranjivati sve podatke.

### **2.2.2. Funkcionalnost proizvoda**

U ovom dijelu specifikacije naveden je popis funkcionalnosti koje bi aplikacija trebala omogućiti korisniku tj. naručitelju:

- Login (prijava) u sustav
- Početni zaslon s izbornikom za odabir pojedinih kategorija: Dobavljači, Kupci i Interesenti, Ugovori, Vozila i prikaz tlocrta dvorišta s vozilima
- Popis dobavljača s mogućnosti detaljnog prikaza pojedinog dobavljača, dodavanja novih, ažuriranje postojećih, te pretraga dobavljača.
- Popis kupaca i interesenata s mogućnosti detaljnog prikaza pojedinog kupca (npr. povijest kupnje) i pojedinog interesenta, dodavanja novih, ažuriranje postojećih, te pretraga kupaca i interesenata.
- Popis svih ugovora sa pretragom i filterima, te mogućnost odabira ručnog i automatskog kreiranja ugovora.
- Popis vozila (u skladištu i u dolasku) gdje imamo mogućnost unosa novog vozila, ažuriranje postojećeg, pretraga, te detaljni pregled svakog vozila
- Kreiranje tlocrta, koji je moguće ažurirati, te dodati i pregledati detalje vozila na parkirnim mjestima.
- Korisničko sučelje za mehaničara u kojem ima uvid u tlocrt, vozila i potrebne popravke na njima.
- Checkbox lista popravka i unos iznosa troška tj. radno vrijeme provedeno na popravku i troškove materijala.

### **2.2.3. Korisnici i njihove karakteristike**

Najbitniji i najčešći korisnici ovog sustava su prodavači i mehaničar u autosalonima. Oni svakodnevno više puta pristupaju sustavu i provjeravaju ili mijenjaju podatke. Trebamo uzeti u obzir njihov stupanj obrazovanja i informatičku pismenost. Stupanj obrazovanja ciljanih korisnika ovog sustava vrlo je raznolik, neki imaju završenu samo osnovnu školu, a neki su čak magistri struke. Naša pretpostavka je da je većina zaposlenika u današnjim poduzećima dovoljno stručna da zna koristiti osnovne aplikacije s obrascima na računalima.

### **2.2.4. Radno okruženje**

driveIT je zamišljena kao Windows Forms aplikacija za osobna računala koja koriste Windows operacijski sustav. Verzija Windows operacijskog sustava nije posebno specificirana ali preporuka su verzije od XP-a pa na dalje. Uz našu aplikaciju, preporučeno je imati instalirano Microsoft Office programski paket namijenjen uredskoj obradi, točnije njegov program Microsoft Word. U ovom alatu biti će omogućeno dodatno uređivanje generiranih ugovora s kupcima i dobavljačima. Ukoliko korisnici žele samo pregled i ispis ugovora, preporučena je instalacija nekog od programa koji omogućuju čitanje PDF-datoteka u čijem formatu će generirani ugovori također biti pohranjeni. Potrebe hardverske platforme nisu posebno specificirane, pa se očekuje da aplikacija radi na svim uređajima s navedenim Windows operacijskih sustavima. Osim aplikacije na lokalnom računalu, potreban je MS SQL Server na kojem je pohranjena baza podataka na koju aplikacija pohranjuje sve podatke.

### **2.2.5. Ograničenja dizajna i implementacije**

Što se tiče ograničenja prilikom implementacije programskog rješenja, ograničili smo se na korisničko sučelje samo na hrvatskom jeziku. Aplikacija zahtjeva pristup mreži na kojoj je spojen i MS SQL Server s kojim komunicira putem TCP/IP protokola. Rad s drugim tipovima baza podataka nismo pokušali implementirati i orijentirali smo se samo na MS SQL Server. Nakon izrade zadovoljavajućeg konačnog proizvoda, održavanje sustava se prebacuje na organizacije u koje će biti implementirano.

### **2.2.6. Korisnička dokumentacija**

Uz ovaj sustav isporučiti će se korisnička dokumentacija koja vizualizira opis načina korištenja aplikacije. Sve forme unutar korisničkog sučelja biti će objašnjene na način da se na slici s formom objasne svi njeni elementi. Uz opise pojedinih forme, definirati ćemo



sljedove korištenja aplikacije, odnosno na koji način doći do određenog elementa aplikacije. Ista ta korisnička dokumentacije će, osim kao zaseban dokument uz ostale dokumentacije, biti dodana u samu aplikaciju. Moći će joj se pristupiti preko gumba pomoć unutar formi.

## **2.2.7. Pretpostavke i zavisnosti**

Od poduzeća koje planira koristiti naš sustav očekujemo da posjeduje barem dva osobna računala (jedno za prodavača u autosalonu i drugo za mehaničara) te poslužitelj s SUBP-om MS SQL Server. Rješenje bi eventualno funkcioniralo i sa samo jednim osobnim računalom, ali smatramo da bi onda razina zadovoljstva korištenja sustava bila smanjena zbog redovite promjene korisničkog sučelja. Očekuje se također stabilna mrežna infrastruktura unutar poduzeća.

## **2.3. Specifični zahtjevi**

### **2.3.1. Zahtjevi vanjskog sučelja**

#### **2.3.1.1. Korisnička sučelja**

Planirana su dva glavna korisnička sučelja sustava namijenjena različitim vrstama korisnika. Prvo je namijenjeno prodavačima u autosalonima i vlasnicima salona. U ovom sučelju moguć je uvid u sve podatke sustava. Sučelje se sastoji od osnovne forme u kojoj je moguć odabir između uređivanja različitih vrsta podataka: o dobavljačima, o kupcima/interesentima, o vozilima, o ugovorima. Odabirom nekih od opcija otvara se nova forma gdje se prikazuje popis svih objekata tog tipa. Klikom na nekog od njih otvara se njegov detaljan opis. Unutar tog opisa moguće je odabrati brisanje objekta ili uređivanje njegovih podataka. Svaku od formi osim početne moguće je zatvoriti i na zaslonu te se onda u fokusu prikaže forma s koje smo pristupili ugašenoj.

Korisničko sučelje za mehaničare ima mogućnost prikaza podataka relevantnih samo za vozila koja zahtijevaju neke popravke ili inspekciju. Prijavom u aplikaciju korisničkim računom mehaničara prikazuje se popis vozila koja zahtijevaju tehnički pregled. Klikom na određeno vozilo otvara se njegov detaljni opis unutar kojeg se nalazi i checkbox lista s potrebnim radnjama na vozilu. Ukoliko mehaničar označi neki element te liste, otvara mu se nova forma u kojoj treba unijeti sve troškove te radnje (promijenjene dijelove i njihov trošak,

te provedene sate). Kada dovrši unos klikom na potvrdu vraća se na prethodnu formu koja je ostala u pozadini.

#### **2.3.1.2. Sučelja hardvera**

Po pitanju sučelja hardvera, nema nekih posebnih sučelja koja aplikacija zahtjeva. Od potrebnih, možemo spomenuti mrežno sučelje koje je potrebno za komunikaciju s poslužiteljem na kojem se nalazi baza podataka. Za sada ne možemo navesti sve potrebne biblioteke koje ćemo koristiti za tu komunikaciju.

#### **2.3.1.3. Sučelja softvera**

Kako je već spomenuto, driveIT aplikaciju je moguće izvršavati samo na računalima s operacijskim sustavom Windows. Za komunikaciju s operacijskim sustavom za sada znamo da nam trebaju iduće biblioteke: Microsoft.CSharp, System, System.Core, System.Data, System.Data.DataSetExtensions, System.Deployment, System.Drawing, System.Windows.Forms, System.Xml, System.Xml.Linq. Ostale ćemo navesti nakon realizacije cjelokupnog programskog rješenja kada budemo znali sve potrebne biblioteke.

#### **2.3.1.4. Sučelja komunikacije**

Što se komunikacije tiče, aplikacija komunicira jedino s poslužiteljem na kojem je baza podataka. Komunikacija se odvija preko TCP/IP protokola, a za razmjenu podataka i izvršavanje upita koristi se Entity Framework koji je ugrađen u .NET Framework.

### **2.3.2. Zahtjevi ponašanja**

Dijagrami korištenja navedeni malo niže u dokumentaciji detaljnije opisuju zahtjeve ponašanja sustava pa ih nije potrebno ovdje opisivati.

## **2.4. Drugi ne-funkcionalni zahtjevi**

### **2.4.1. Zahtjevi s obzirom na performanse**

Pošto se radi o aplikaciji koja se nalazi na jednom osobnom računalu i ne zahtjeva njegove velike kapacitete hardvera tokom izvođenja, ne nalazimo neka ograničenja koja bi mogla ugroziti performanse aplikacije. Očekujemo da će aplikacija nakon pokretanja biti spremna za rad unutar nekoliko sekundi. Pojedine forme trebale bi se također otvarati maksimalno unutar nekoliko sekundi, dok bi se na jačim konfiguracijama te operacije odvijale za treptaj oka. Promjene se evidentiraju na bazi podataka i smatramo da bi transakcije oduzimale najviše vremena u radu aplikacije. Pretpostavljamo da se poslužitelj s bazom

podataka nalazi u istoj lokalnoj mreži čije su brzine oko 100 Mb/s pa ni ove transakcije ne bi trebale trajati duže od par sekundi.

### **2.4.2. Zahtjevi sigurnosti**

Da bi aplikacija funkcionirala ispravno ne smiju se dogoditi promjene nad podacima koje nije počinio netko od korisnika. Na sreću, aplikacija se nalazi u lokalnoj mreži i ne komunicira s „vanjskim svijetom“ pa ju je lakše zaštititi od vanjskih utjecaja. Drugi problem predstavlja gubitak podataka usred kvara na poslužitelju. Na njega mi ne može utjecati i preporučamo našim klijentima da si naprave repliku poslužitelja baze podataka i redovito ju osvježavaju. Kvar na lokalnom računalu je manji problem jer se s novom instalacijom aplikacije povuku podaci s poslužitelja.

Osim zaštite izvana, sustav je potrebno zaštititi i od samih korisnika i njihovih nenamjernih krivih unosa koje ćemo ostaviti tako što će aplikacija provjeravati svaki unos ili promjenu podataka da li su valjanih oblika. Sustav je u stanju provjeriti da li je pravilno napisano, ali ne i da li je točno napisano.

### **2.4.3. Atributi kvalitete softvera**

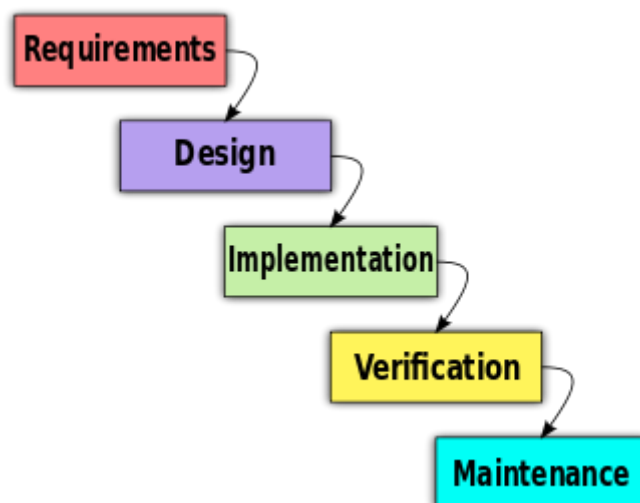
Korisnici bi se na aplikaciju driveIT trebali brzo naviknuti jer su se često susreli s mnogo sličnih sustava za evidenciju. Raspored upravljanja aplikacijom je logički posložen i lako pamtljiv. Aplikaciji je moguće pristupiti u svako doba dana jer se radi o nekom web-sustavu koji zahtjeva vezu na Internet. Moguće ju je instalirati na više računala i koristiti s više njih istovremeno. Broj instanci aplikacije nije striktno definirani i može varirati ovisno u broju računala u poduzeću. Instancu aplikacije nije moguće prenositi s jednog računala na drugo već je potrebno drugu instalirati na drugom računalu. Ali to ne predstavlja neko ograničenje jer se podaci povuku s poslužitelja u lokalnoj mreži pa je jedini zahtjev da računalo ima pristup toj mreži.

## 3. Projektni plan

### 3.1. Projektni tim

Na ovom projektu radi tim od pet studenata među kojima su raspoređene aktivnosti koje svatko mora napraviti u određenom vremenskom razdoblju kako je definiramo projektnim planom. Potrebno je dobro definirati te aktivnosti kako bi se mogao pratiti napredak projekta i njegov uspjeh u poštivanju vremenskih rokova. Za realizaciju ovog projekta odabrali smo vodopadni model (slika modela 2.) koji se sastoji od pet faza: Specifikacije zahtjeva, Dizajna, Implementacije, Testiranja i Održavanja. Specifikacija zahtjeva odnosi se na dogovaranje oko funkcionalnosti, specifikacije korisnika i na razmatranje ograničenja dizajna implementacije. Faza Dizajn se odnosi na UML modeliranje problema i izrade ERA modela. Implementacija se odnosi na izradu aplikacije, a faza Testiranje uključuje testiranje izrađenog programskog rješenja. Zadnji dio faze testiranja uključuje izradu korisničke dokumentacije. Zadnja faza Održavanje se odnosi na održavanje aplikacije.

Svi članovi tima sudjeluju u svim dijelovima projekta podjednako stoga nisu striktno definirane uloge među članovima projektnog tima. Aktivnosti su se podijelile među članovima, s ciljem da svi jednako utroše vremena na izradu ovog projekta. Na slici 2. se može vidjeti tko šta radi na izradi projekta. Radno vrijeme tima je svaki dan od 16.00 do 19.00.



*Slika 1. Vodopadni model*

ID	Resource Name	Work				
1	Marko	68,5 hrs				
ID	Task Name	Units	Work	Delay	Start	Finish
15	Izrada Use case	100%	3 hrs	0 days	Wed 16.4.14	Wed 16.4.14
7	Dogovaranje funkcionalnosti	100%	2 hrs	0 days	Tue 1.4.14	Tue 1.4.14
8	Specifikacija korisnika	100%	1 hr	0 days	Tue 1.4.14	Tue 1.4.14
9	Razmatranje ograničenja dizajna implementacije	100%	1,5 hrs	0 days	Wed 2.4.14	Wed 2.4.14
28	Izrada modula za prijavu	100%	18 hrs	0 days	Thu 22.5.14	Tue 27.5.14
25	Izrada formi i dizajna	100%	12 hrs	0 days	Fri 2.5.14	Mon 5.5.14
33	Izrada interaktivnog parkirališta	100%	18 hrs	0 days	Tue 10.6.14	Sun 15.6.14
40	Testiranje dijela za mehaničare	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
41	Dorada dijela za mehaničare	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
42	Testiranje cijele aplikacije	100%	1 hr	0 days	Fri 20.6.14	Fri 20.6.14
17	Izrada dijagrama aktivnosti	100%	3 hrs	0 days	Sat 19.4.14	Sat 19.4.14
30	Izrada sučelja za mehaničara	100%	7 hrs	0 days	Thu 29.5.14	Sat 31.5.14
2	Mislav	69,5 hrs				
ID	Task Name	Units	Work	Delay	Start	Finish
7	Dogovaranje funkcionalnosti	100%	2 hrs	0 days	Tue 1.4.14	Tue 1.4.14
8	Specifikacija korisnika	100%	1 hr	0 days	Tue 1.4.14	Tue 1.4.14
9	Razmatranje ograničenja dizajna implementacije	100%	1,5 hrs	0 days	Wed 2.4.14	Wed 2.4.14
15	Izrada Use case	100%	3 hrs	0 days	Wed 16.4.14	Wed 16.4.14
25	Izrada formi i dizajna	100%	12 hrs	0 days	Fri 2.5.14	Mon 5.5.14
27	Izrada modula za prikaz detalja	100%	14 hrs	0 days	Sat 17.5.14	Wed 21.5.14
28	Izrada modula za prijavu	100%	18 hrs	0 days	Thu 22.5.14	Tue 27.5.14
33	Izrada interaktivnog parkirališta	100%	18 hrs	0 days	Tue 10.6.14	Sun 15.6.14
3	Andrea	67,5 hrs				
ID	Task Name	Units	Work	Delay	Start	Finish
7	Dogovaranje funkcionalnosti	100%	2 hrs	0 days	Tue 1.4.14	Tue 1.4.14
8	Specifikacija korisnika	100%	1 hr	0 days	Tue 1.4.14	Tue 1.4.14
9	Razmatranje ograničenja dizajna implementacije	100%	1,5 hrs	0 days	Wed 2.4.14	Wed 2.4.14
16	Izrada dijagrama slijeda	100%	6 hrs	0 days	Sat 19.4.14	Sun 20.4.14
33	Izrada interaktivnog parkirališta	100%	18 hrs	0 days	Tue 10.6.14	Sun 15.6.14
29	Izrada modula za osvježavanje	100%	9 hrs	0 days	Tue 27.5.14	Thu 29.5.14
26	Izrada kontrolera do kraja	100%	13 hrs	0 days	Mon 12.5.14	Fri 16.5.14
43	Izrada korisničke dokumentacije	100%	6 hrs	0 days	Sat 21.6.14	Sun 22.6.14
42	Testiranje cijele aplikacije	100%	1 hr	0 days	Fri 20.6.14	Fri 20.6.14
38	Testiranje dijela za prodavače	100%	1 hr	0 days	Fri 20.6.14	Fri 20.6.14
39	Dorada dijela za prodavače	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
14	Izrada ERA modela	50%	3 hrs	0 days	Wed 16.4.14	Thu 17.4.14
32	Izrada Help dokumenta	100%	3 hrs	0 days	Tue 3.6.14	Tue 3.6.14
40	Testiranje dijela za mehaničare	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
41	Dorada dijela za mehaničare	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
4	Lovro	70,1 hrs				
ID	Task Name	Units	Work	Delay	Start	Finish
7	Dogovaranje funkcionalnosti	100%	2 hrs	0 days	Tue 1.4.14	Tue 1.4.14
8	Specifikacija korisnika	100%	1 hr	0 days	Tue 1.4.14	Tue 1.4.14
9	Razmatranje ograničenja dizajna implementacije	100%	1,5 hrs	0 days	Wed 2.4.14	Wed 2.4.14
16	Izrada dijagrama slijeda	100%	6 hrs	0 days	Sat 19.4.14	Sun 20.4.14
14	Izrada ERA modela	100%	6 hrs	0 days	Wed 16.4.14	Thu 17.4.14
18	Izrada dijagrama klasa	100%	6 hrs	0 days	Sat 26.4.14	Sun 27.4.14
23	Izrada DataBaseAdapter-a i MsSQLDataLink-a	100%	10 hrs	0 days	Fri 2.5.14	Mon 5.5.14
24	Izrada CRUD kontrola	100%	12 hrs	0 days	Wed 7.5.14	Sat 10.5.14
29	Izrada modula za osvježavanje	100%	9 hrs	0 days	Tue 27.5.14	Thu 29.5.14
33	Izrada interaktivnog parkirališta	70%	12,6 hrs	0 days	Tue 10.6.14	Sun 15.6.14
38	Testiranje dijela za prodavače	100%	1 hr	0 days	Fri 20.6.14	Fri 20.6.14
39	Dorada dijela za prodavače	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
40	Testiranje dijela za mehaničare	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
41	Dorada dijela za mehaničare	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14
5	Josip	70,3 hrs				
ID	Task Name	Units	Work	Delay	Start	Finish
18	Izrada dijagrama klasa	50%	3 hrs	0 days	Sat 26.4.14	Sun 27.4.14
7	Dogovaranje funkcionalnosti	100%	2 hrs	0 days	Tue 1.4.14	Tue 1.4.14
8	Specifikacija korisnika	100%	1 hr	0 days	Tue 1.4.14	Tue 1.4.14
9	Razmatranje ograničenja dizajna implementacije	100%	1,5 hrs	0 days	Wed 2.4.14	Wed 2.4.14
14	Izrada ERA modela	80%	4,8 hrs	0 days	Wed 16.4.14	Thu 17.4.14
16	Izrada dijagrama slijeda	50%	3 hrs	0 days	Sat 19.4.14	Sun 20.4.14
26	Izrada kontrolera do kraja	100%	13 hrs	0 days	Mon 12.5.14	Fri 16.5.14
31	Izrada modula za pdf dokumente	100%	13 hrs	0 days	Fri 30.5.14	Tue 3.6.14
27	Izrada modula za prikaz detalja	100%	14 hrs	0 days	Sat 17.5.14	Wed 21.5.14
24	Izrada CRUD kontrola	100%	12 hrs	0 days	Wed 7.5.14	Sat 10.5.14
38	Testiranje dijela za prodavače	100%	1 hr	0 days	Fri 20.6.14	Fri 20.6.14
42	Testiranje cijele aplikacije	100%	1 hr	0 days	Fri 20.6.14	Fri 20.6.14
41	Dorada dijela za mehaničare	100%	1 hr	0 days	Thu 19.6.14	Thu 19.6.14

Slika 2. Izvješće "Tko što radi"

U tablici 1. prikazane su uloge svakog člana tima te opisane ukratko za što je bio točno zadužen pojedini član tima.

Član tima	Uloga	Opis
Marko Alerić	User Interface Designer, Programmer	Zadatak je dizajniranje korisničkog sučelja aplikacije i njegova implementacija u alatu Visual Studio.
Mislav Boras	User Interface Designer, Programmer	Zadatak je dizajniranje korisničkog sučelja aplikacije i njegova implementacija u alatu Visual Studio.
Lovro Predovan	Database Administrator, Software Arhitekt	Zadatak je oblikovati bazu podataka te omogućiti spajanje aplikacije s njom, te izrada UML dijagrama
Andrea Pleše	Project Manager, Tehnical Writer	Zadatak je izrada plana, proračuna, ponude, prati napredak projekta u skladu sa rokovima te izrađuje korisničku dokumentaciju
Josip Primorac	Programmer, Software Arhitekt	Zadatak je pisanje programskog koda i izrada UML dijagrama

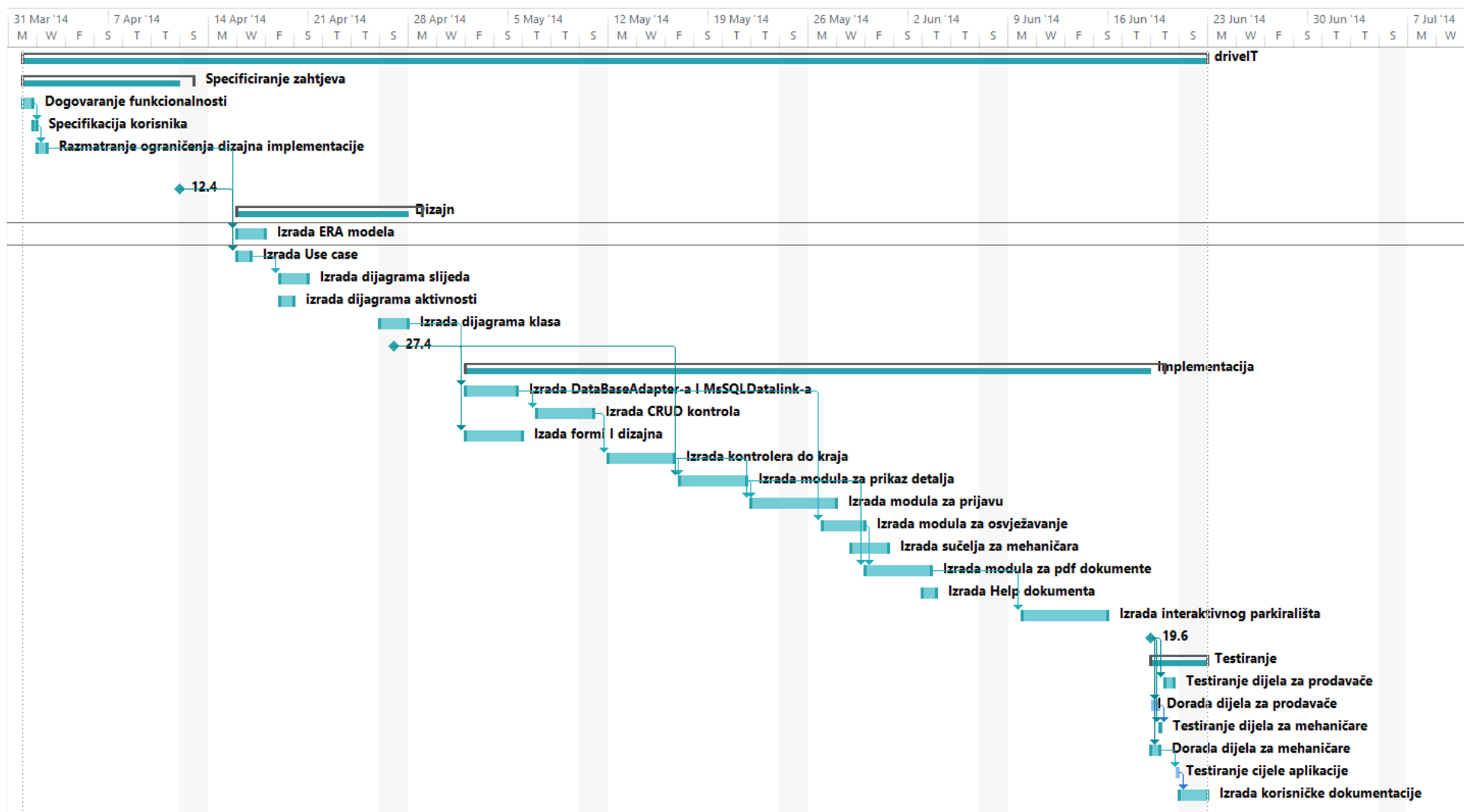
*Tablica 1. Prikaz uloga u timu*

### 3.2. Terminski plan projekta

Za izradu projektnog plana koristili smo alat MS Project 2013. Na slici 3. je prikazan terminski plan aktivnosti na kojem se vidi kad je koja aktivnost započela, završila te redoslijed izvršavanja aktivnosti. Također, vidi se koliko je svaka aktivnost trajala i njeni potrebni resursi da bi se aktivnost izvršila. Slika 4. predstavlja ganttogram, odnosno grafičko prikazivanje informacija koje se koristi za utvrđivanje rasporeda aktivnosti. Na njemu se jasno vidi redoslijed kojim se aktivnosti izvršavaju i njihova međuovisnost.

Task Name	Duration	Start	Finish	Pre	Resource Names
➤ <b>driveIT</b>	<b>83 days</b>	<b>Tue 1.4.14</b>	<b>Sun 22.6.14</b>		
➤ <b>Specificiranje zahtjeva</b>	<b>12 days</b>	<b>Tue 1.4.14</b>	<b>Sat 12.4.14</b>		
Dogovaranje funkcionalnosti	2 hrs	Tue 1.4.14	Tue 1.4.14		Andrea ;Josip;Lovro ;Marko;Mislav ;Laptop1[1];Laptop2[1]
Specifikacija korisnika	1 hr	Tue 1.4.14	Tue 1.4.14	3	Andrea ;Josip;Lovro ;Marko;Mislav ;Laptop1[1];Laptop2[1]
Razmatranje ograničenja dizajna implementacije	1,5 hrs	Wed 2.4.14	Wed 2.4.14	4	Andrea ;Josip;Lovro ;Marko;Mislav ;Laptop1[1];Laptop2[1]
Kraj Specificiranja zahtjeva	0 days	Sat 12.4.14	Sat 12.4.14		
➤ <b>Dizajn</b>	<b>13 days</b>	<b>Wed 16.4.14</b>	<b>Mon 28.4.14</b>		
Izrada ERA modela	2 days	Wed 16.4.14	Thu 17.4.14	6;5	Josip[80%];Lovro ;Andrea [50%];Laptop1[1];Laptop2[1]
Izrada Use case	1 day	Wed 16.4.14	Wed 16.4.14	6	Marko;Mislav ;Laptop1[1]
Izrada dijagrama slijeda	2 days	Sat 19.4.14	Sun 20.4.14	9	Lovro ;Josip[50%];Andrea ;Laptop1[1]
izrada dijagrama aktivnosti	1 day	Sat 19.4.14	Sat 19.4.14		Marko;Laptop2[1]
Izrada dijagrama klasa	2 days	Sat 26.4.14	Sun 27.4.14		Josip[50%];Lovro ;Laptop1[1]
Kraj modeliranja	0 days	Sun 27.4.14	Sun 27.4.14		
➤ <b>Implementacija</b>	<b>49 days</b>	<b>Fri 2.5.14</b>	<b>Thu 19.6.14</b>		
Izrada DataBaseAdapter-a i MsSQLDatalink-a	10 hrs	Fri 2.5.14	Mon 5.5.14	13;12	Lovro ;Laptop1[1]
Izrada CRUD kontrola	12 hrs	Wed 7.5.14	Sat 10.5.14	15	Josip;Lovro ;Laptop1[1];Laptop2[1]
Izrada formi i dizajna	12 hrs	Fri 2.5.14	Mon 5.5.14	13	Mislav ;Marko
Izrada kontrolera do kraja	13 hrs	Mon 12.5.14	Fri 16.5.14	16	Josip;Andrea ;Laptop3[1];Laptop4[1]
Izrada modula za prikaz detalja	14 hrs	Sat 17.5.14	Wed 21.5.14	18;13	Josip;Mislav ;Laptop2[1];Laptop4[1]
Izrada modula za prijavu	18 hrs	Thu 22.5.14	Tue 27.5.14	19;18	Marko;Mislav ;Laptop2[1];Laptop5[1]
Izrada modula za osvježavanje	9 hrs	Tue 27.5.14	Thu 29.5.14	15	Andrea ;Lovro ;Laptop1[1];Laptop3[1]
Izrada sučelja za mehaničara	7 hrs	Thu 29.5.14	Sat 31.5.14		Laptop2[1];Marko
Izrada modula za pdf dokumente	13 hrs	Fri 30.5.14	Tue 3.6.14	19;21	Josip;Laptop4[1]
Izrada Help dokumenta	3 hrs	Tue 3.6.14	Tue 3.6.14		Andrea ;Laptop3[1]
Izrada interaktivnog parkirališta	18 hrs	Tue 10.6.14	Sun 15.6.14	23	Andrea ;Lovro [70%];Marko;Mislav ;Laptop1[1];Laptop2[1]
Kraj realizacije	0 days	Thu 19.6.14	Thu 19.6.14		
➤ <b>Testiranje</b>	<b>4 days</b>	<b>Thu 19.6.14</b>	<b>Sun 22.6.14</b>		
Testiranje dijela za prodavače	1 hr	Fri 20.6.14	Fri 20.6.14	26	Josip;Lovro ;Andrea ;Laptop1[1]
Dorada dijela za prodavače	1 hr	Thu 19.6.14	Thu 19.6.14	26	Lovro ;Andrea ;Laptop2[1]
Testiranje dijela za mehaničare	1 hr	Thu 19.6.14	Thu 19.6.14	26;29	Marko;Lovro ;Laptop3[1];Andrea
Dorada dijela za mehaničare	1 hr	Thu 19.6.14	Thu 19.6.14	26	Marko;Lovro ;Laptop4[1];Andrea ;Josip
Testiranje cijele aplikacije	1 hr	Fri 20.6.14	Fri 20.6.14	31	Andrea ;Marko;Laptop1[1];Laptop2[1];Josip
Izrada korisničke dokumentacije	6 hrs	Sat 21.6.14	Sun 22.6.14	32	Andrea ;Laptop3[1]
Održavanje					

Slika 3. Terminski plan aktivnosti (MS Project)



Slika 4. Ganttov dijagram (MS Project)



### 3.3. Proračun i budžet projekta

Proračun tj. budžet predstavlja troškove projekta. Troškovi ovog projekta su trošak ljudskog resursa. Plaća ljudskog resursa iznosi 35 kn/h, a trošak materijalnog resursa 0,01kn/h. Na slici 5. je prikazan proračun projekta gdje se jasno vidi da cijena same izrade aplikacije driveIT, bez PDV-a i marže iznosi 12.106,85 kn. Cijena aplikacije zajedno sa maržom i PDV-om iznosi 20.430,31kn.

Budget Report as of Sat 12.7.14  
driveIT ms project 2007

ID	Task Name	Fixed Cost	Fixed Cost Accrual	Total Cost	Baseline	Variance	Actual
33	Izrada interaktivnog parkirališta	0,00 kn	Prorated	2.331,02 kn	0,00 kn	2.331,02 kn	0,00 kn
28	Izrada modula za prijavu	0,00 kn	Prorated	1.260,02 kn	0,00 kn	1.260,02 kn	0,00 kn
27	Izrada modula za prikaz detalja	0,00 kn	Prorated	980,02 kn	0,00 kn	980,02 kn	0,00 kn
26	Izrada kontrolera do kraja	0,00 kn	Prorated	910,02 kn	0,00 kn	910,02 kn	0,00 kn
24	Izrada CRUD kontrola	0,00 kn	Prorated	840,02 kn	0,00 kn	840,02 kn	0,00 kn
25	Izrada formi I dizajna	0,00 kn	Prorated	840,00 kn	0,00 kn	840,00 kn	0,00 kn
29	Izrada modula za osvježavanje	0,00 kn	Prorated	630,02 kn	0,00 kn	630,02 kn	0,00 kn
16	Izrada dijagrama slijeda	0,00 kn	Prorated	525,01 kn	0,00 kn	525,01 kn	0,00 kn
14	Izrada ERA modela	0,00 kn	Prorated	483,02 kn	0,00 kn	483,02 kn	0,00 kn
31	Izrada modula za pdf dokumen	0,00 kn	Prorated	455,01 kn	0,00 kn	455,01 kn	0,00 kn
7	Dogovaranje funkcionalnosti	0,00 kn	Prorated	350,02 kn	0,00 kn	350,02 kn	0,00 kn
23	Izrada DataBaseAdapter-a I Ms	0,00 kn	Prorated	350,01 kn	0,00 kn	350,01 kn	0,00 kn
18	Izrada dijagrama klasa	0,00 kn	Prorated	315,01 kn	0,00 kn	315,01 kn	0,00 kn
9	Razmatranje ograničenja dizajn	0,00 kn	Prorated	262,52 kn	0,00 kn	262,52 kn	0,00 kn
30	Izrada sučelja za mehaničara	0,00 kn	Prorated	245,01 kn	0,00 kn	245,01 kn	0,00 kn
15	Izrada Use case	0,00 kn	Prorated	210,01 kn	0,00 kn	210,01 kn	0,00 kn
43	Izrada korisničke dokumentacij	0,00 kn	Prorated	210,01 kn	0,00 kn	210,01 kn	0,00 kn
8	Specifikacija korisnika	0,00 kn	Prorated	175,02 kn	0,00 kn	175,02 kn	0,00 kn
41	Dorada dijela za mehaničare	0,00 kn	Prorated	140,01 kn	0,00 kn	140,01 kn	0,00 kn
42	Testiranje cijele aplikacije	0,00 kn	Prorated	105,02 kn	0,00 kn	105,02 kn	0,00 kn
17	izrada dijagrama aktivnosti	0,00 kn	Prorated	105,01 kn	0,00 kn	105,01 kn	0,00 kn
32	Izrada Help dokumenta	0,00 kn	Prorated	105,01 kn	0,00 kn	105,01 kn	0,00 kn
38	Testiranje dijela za prodavače	0,00 kn	Prorated	105,01 kn	0,00 kn	105,01 kn	0,00 kn
40	Testiranje dijela za mehaničare	0,00 kn	Prorated	105,01 kn	0,00 kn	105,01 kn	0,00 kn
39	Dorada dijela za prodavače	0,00 kn	Prorated	70,01 kn	0,00 kn	70,01 kn	0,00 kn
44	Održavanje	0,00 kn	Prorated	0,00 kn	0,00 kn	0,00 kn	0,00 kn
		<b>0,00 kn</b>		<b>12.106,85 kn</b>	<b>0,00 kn</b>	<b>12.106,85 kn</b>	<b>0,00 kn</b>

Slika 5. Izvještaj o budžetu projekta (MS Project)

U tablici 2. prikazano je kolika je cijena aplikacije ako se računa samo ljudski i materijalni rad, kolika je cijena uz maržu od 35% te cijena za kupca uz maržu i PDV.

	Cijena (u kunama)
Cijena driveIT aplikacije (bez marže i PDV-a)	12.106,85
Cijena driveIT aplikacije sa maržom od 35% (4.237,4 kn)	16.344,25
Cijena driveIT aplikacije sa maržom(35%) PDV-om(25%)	20.430,31

Tablica 2. Prikaz cijena aplikacije

### **3.4. Ponuda naručitelju**

Nakon što je napravljen plan projekta potrebno je napraviti ponudu za buduće korisnike. U ponudi je potrebno navesti okvirnu cijenu i troškove cijelog projekta. Ukoliko, netko od budućih korisnika prihvati našu ponudu, potrebno je prezentirati plan programskog rješenja i nastaviti dalje sa radom na modeliranju i realizaciji planirane aplikacije.

team.NOT  
Pavlinska 2,  
42 000 Varaždin



## PONUDA

### driveIT

Poštovani,  
tvrtka team.NOT d.o.o. razvila je aplikaciju koja pomaže autosalonima rabljenih vozila oko evidencije stanja vozila u njihovom „dvorištu“. Smatramo da bi ova aplikacija driveIT bila idealno rješenje za Vaš autosalon.

Pomoću ove aplikacije imati ćete sve informacije o Vašim vozilima, dobavljačima, kupcima na jednom mjestu. Moći ćete vidjeti specifikacije svakog vozila i sve potrebne informacije od svakog dobavljača i kupca. Također, moguće je dodavati nova vozila, dobavljače i kupce. Ova aplikaciju nudi i generiranje ugovora o kupnji i prodaji Vaših vozila. Velika prednost ove aplikacije je što nudi kalkulaciju cijena sučelje na kojem bi se nalazio tlocrt Vašeg „dvorišta“ kako bi te mogli jednostavnije pregledavati raspoloživa vozila.

Cijena aplikacije driveIT (bez PDV-a) je: 16.344,25 kn

Cijena aplikacije driveIT (s PDV-om) je: 20.430,31 kn

Za dodatna pitanja možete se nam obratiti na mail ili možemo dogovoriti sastanak na kojem ćemo dogovoriti sve pojedinosti.

Veselimo se budućoj suradnji!

S poštovanjem,

Marko Alerić ([maleric@foi.hr](mailto:maleric@foi.hr))

Mislav Boras ([mboras@foi.hr](mailto:mboras@foi.hr))

Andrea Pleše ([aplese@foi.hr](mailto:aplese@foi.hr))

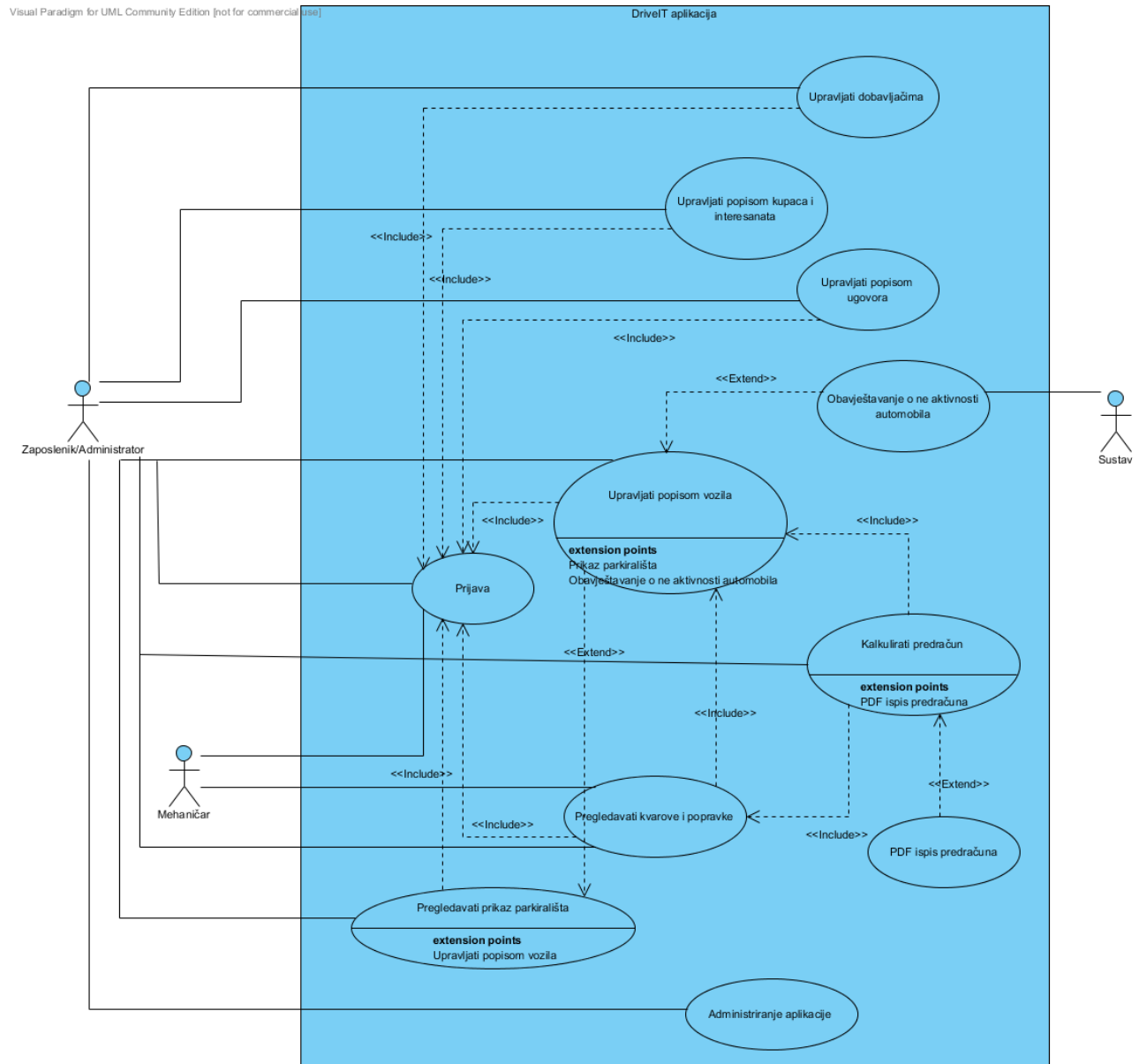
Lovro Predovan ([lpredova@foi.hr](mailto:lpredova@foi.hr))

Josip Primorac ([jprimora@foi.hr](mailto:jprimora@foi.hr))

## 4. Tehnička dokumentacija

### 4.1. Dijagram slučajeja korištenja

S obzirom na zahtjeve aplikacije, na slici ispod se nalazi dijagram slučajeja korištenja.



Prilikom korištenja aplikacije, korisnik se prvo mora prijaviti te s obzirom na njegovu ulogu, na raspolaganju ima određenu funkcionalnost. U sustavu imamo 2 učesnika, **mehaničara**, **administratora** te sustav koji šalje automatski obavijesti o neaktivnosti vozila. Kao što je navedeno svaki od navedenih učesnika osim sustava, prvo se mora prijaviti u sustav.

Zaposlenik sustava može vidjeti popis dobavljača, te bez obzira koliko je ljudi na tom popisu ima mogućnost osnovnih CRUD operacija, gdje čita sve dobavljače te može dodati novog, ažurirati postojećeg ili pak obrisati dobavljača iz sustava. Svaki dobavljač iz liste dobavljača, ima mogućnost pretrage te određene informacije i detalje koje zaposlenik također može vidjeti ukoliko to želi.

Zaposlenik ima uvid na **popis kupaca i interesenata**, gdje kao i kod slučaja dobavljača može dodati nove kupce i/ili interesente, ažurirati ih i brisati, pretraživati te vidjeti detalje za svakog pojedinog kupca i/ili interesenta.

**Popis vozila** može vidjeti i administrator i mehaničar, također imaju osnovne CRUD operacije, pretraga vozila te je moguće vidjeti detalje o pojedinom vozilu, galeriju slika te popis dodane opreme na vozilu. Kod detalja vozila može kreirati nalog za servis.

**Prikaz parkirališta** može vidjeti administrator koji na njima može obavljati osnovne CRUD operacije.

Zaposlenik ima na uvid **popis ugovora**, gdje može dodati novi ugovor, ažurirati ili obrisati postojeći, vršiti pretragu, te vidjeti detalje o određenom ugovoru.

**Pregled popravaka i kvarova** mogu vidjeti zaposlenik i mehaničar, gdje mogu unijeti, ažurirati i brisati popravke i kvarove, te vidjeti detalje o svakom od njih.

## Specifikacije slučaja korištenja:

### 4.2. Dodavanje novog vozila

<b>Brief</b>	Dodavanje		
<b>Description</b>			
<b>Preconditions</b>	Prijava u sustav		
<b>Post-conditions</b>	Uspješno ili neuspješno dodan dobavljač		
<b>Importance</b>			
<b>Flow of Events</b>		<b>Actor Input</b>	<b>System Response</b>
	1	Odabir vozila iz glavne forme	
	2		Prikaz forme vozila
	3	Odabir dodavanje novog vozila	
	4		Prikaz forme za dodavanje
	5	Popunjavanje forme podacima	
	6	Odabir dodavanja vozila	
	7		Spremanje vozila u bazu i zatvaranje forme
	8		

### 4.3. Prijava u sustav

**Brief Description** Prijava korisnika u sustav obavezna je za sve korisnike koji rade sa aplikacijom.

**Preconditions** Korisnik mora imati username i password za login.

**Post-conditions** Korisnik je prijavljen ili odbijen.

#### Importance

Flow of Events		Actor Input	System Response
	1	Pokreće aplikaciju	
	2		Prikazuje ekran za prijavu
	3	Unosi korisničke podatke	
	4		Prikazuje ekran aplikacije
	5		Ispisuje se poruka o grešci

### 4.4. Dodavanje dobavljača

**Brief Description** Dodavanje novog dobavljača

**Preconditions** Prijava u sustav.

**Post-conditions** Uspješno ili neuspješno dodan dobavljač

#### Importance

Flow of Events		Actor Input	System Response
	1	Odabir dobavljača iz glavne forme	
	2		Prikaz forme dobavljača
	3	Odabire dodavanje dobavljača	
	4		Prikazuje formu za dodavanje
	5	Popunjavanje forme podacima	
	6	Odabir dodavanja dobavljača	
	7		Ispis poruke o uspješnom ili neuspješnom dodavanju dobavljača
			Spremanje dobavljača u bazu i zatvaranje forme za dodavanje

## 4.5. Nalog za popravak

Super Use Case	Izdavanje naloga za popravak automobila		
Brief Description	Izdavanje naloga za popravak automobila. Nalog je vidljiv mehaničaru koji može taj nalog obraditi		
Preconditions	Dodan automobil		
Post-conditions	Uspješno ili neuspješno kreiran nalog		
Importance			
Flow of Events		Actor Input	System Response
	1	Odabire opciju vozila iz glavne forme	
	2		Otvara se forma vozila
	3	Klikće na željeno vozilo iz liste	
	4		Otvara se forma s detaljima o odabranom vozilu
	5	Odabire nalog za popravak	
	6	Zatvara obrazac	Otvara se forma za ispunjavanje naloga
	7	Unosi podatke	
	8	Odabire kreiranje naloga	
	9		Ispis poruke o uspješnom ili neuspješnom kreiranju naloga
	10		Zatvara se forma za kreiranje naloga

## 4.6. Paljenje vozila koja dugo nisu paljena

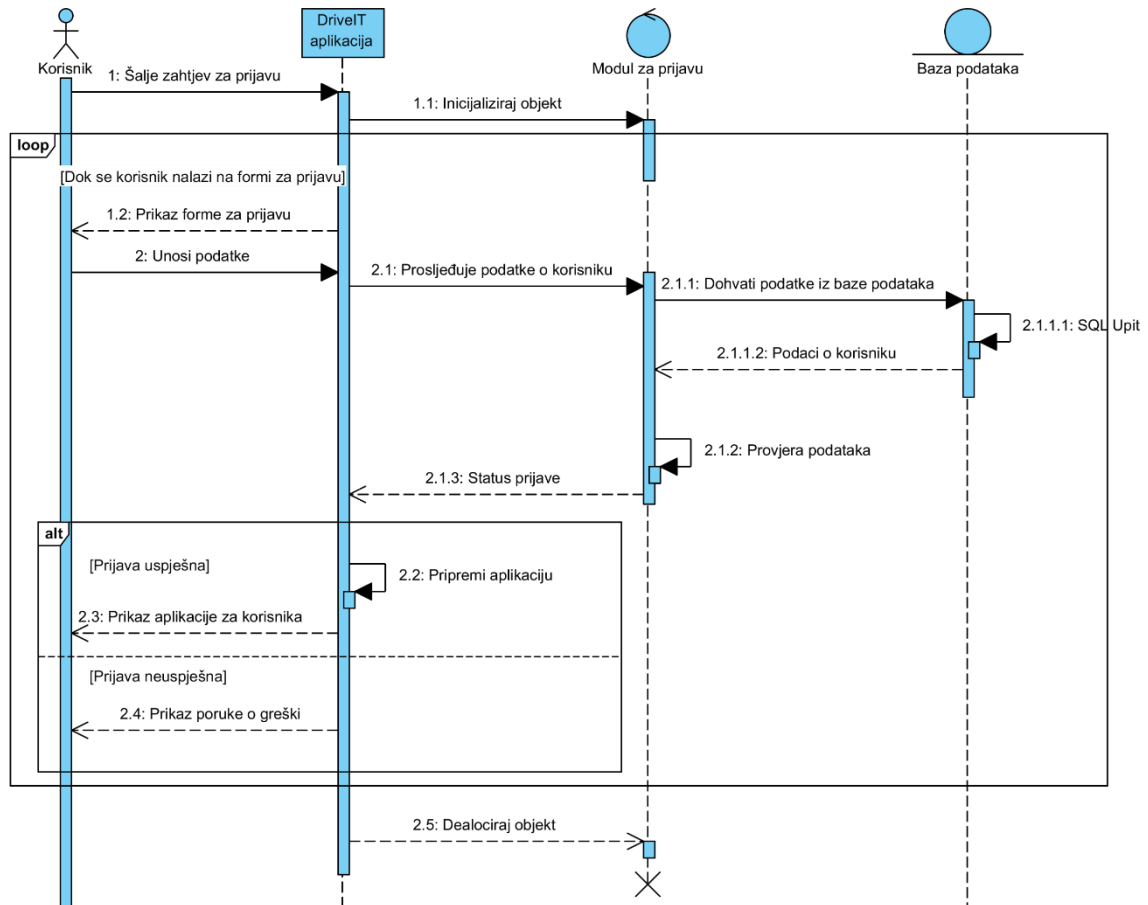
<b>Brief Description</b>	Paljenje vozila koja dugo nisu paljena. Ovu funkciju obavlja mehaničar		
<b>Preconditions</b>	Logiranje u sustav kao mehaničar		
<b>Post-conditions</b>	Upaljeno vozilo		
<b>Importance</b>			
<b>Flow of Events</b>		<b>Actor Input</b>	<b>System Response</b>
	1	Prijava u sustav kao mehaničara	
	2		Otvaranje forme za mehaničara
	3	Iz liste vozila koja dugo nisu paljena odabire vozilo	
	4	Odabir gumba upali za paljenje vozila	
	5		Poruka da je vozilo upaljeno
	6		Ažuriranje liste vozila koja dugo nisu paljena



## 4.7. Dijagrami slijeda

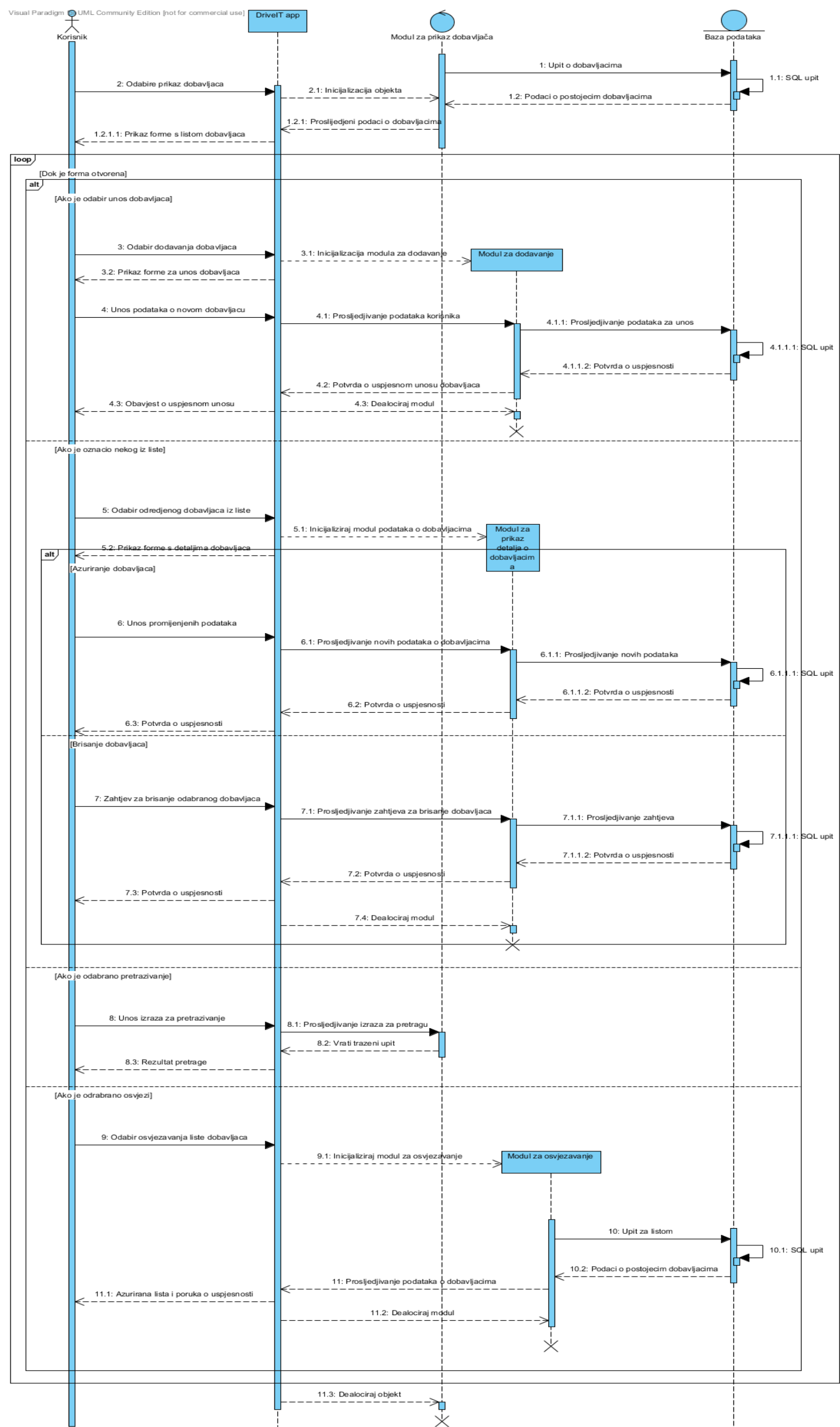
### 4.7.1. Dijagram slijeda – Prijava u aplikaciju

UML Paradigm for UML Community Edition [not for commercial use]



Jedan od osnovnih dijagrama slijeda je prijava u aplikaciju. Prilikom pokretanja aplikacije inicijalizira se modul za prijavu te se prikazuje korisniku. Korisnik unosi podatke koji se preko modula prosljeđuju u bazu podataka gdje se čitaju svi postojeći korisnici. Podaci iz baze se vraćaju modulu za prijavu gdje se vrši kontrola između unesenih podataka i podataka iz baze. Ukoliko je status prijave bio uspješan, korisniku se pripremi i prikaže aplikacija, dok u suprotnom slučaju se ispiše poruka o greški. Postupak se vrši dok se korisnik nalazi na formi za prijavu, nakon čega se vrši dealokacija modula za prijavu.

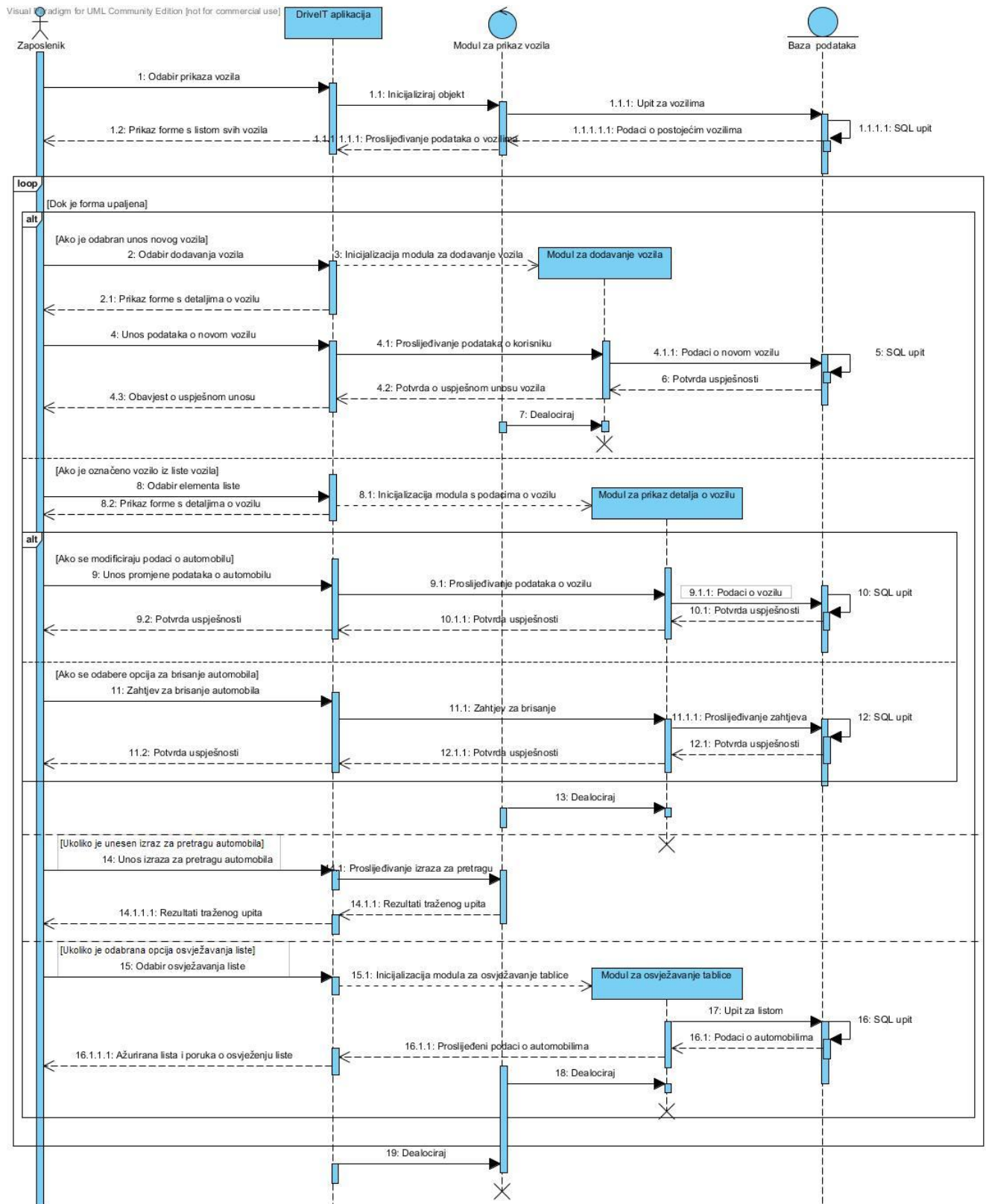
4.7.2. Dijagram slijeda – Dobavljači



Slika 6. Dijagram slijeda - Dobavljači

Na slici je opisan dijagram slijeda aktivnosti ukoliko korisnik odabere opciju “Dobavljači”. Dakle aktivnost počinje kada korisnik iz glavne forme odabere opciju “Dobavljači”, zatim se u pozadini inicijalizira modul za prikaz dobavljača koji iz baze podataka dohvati sve dobavljače te ih prikaže korisniku u odgovarajućoj formi. Zatim korisnik u toj formi ima mogućnost da unese novog dobavljača, osvježi dobavljače ili pretraži sve dobavljači. Klikom na pojedinog dobavljača se otvara forma sa svim detaljima o dobavljaču i s mogućnošću da se taj dobavljač ažurira ili obriše. Npr. ako korisnik odabere unos novog dobavljača. Klikom na gumb “Unesi novog dobavljača” započinje se proces dodavanja novog dobavljača. Istovremeno se inicijalizira modul za dodavanje i korisniku se prikazuje odgovarajuća forma za dodavanje dobavljača u koju unosi određene vrijednosti o svakom dobavljaču. Zatim se ti podatci proslijede na modul za dodavanje koji dalje šalje te podatke na bazu podataka gdje se ti podatci i upisuju. Zatim ako je uspješan korisniku se prikaže obavijest o uspješnom unosu. Ova funkcionalnost je aktivna sve dok korisnik ne izađe iz forme i prilikom izlaska se modul dealocira.

### 4.7.3. Dijagram slijeda – Vozila



Dijagram slijeda pregleda vozila prikazuje mogućnosti koje korisnik može odabrati nakon što na glavnom izborniku u programu odabere opciju pregled vozila. Samim pokretanjem opcije instancira se objekt odnosno forma s elementima za prikaz mogućih

opcija koje se nude korisniku. Karakteristično za ovu klasu formi ( CRUD forme) u našem programu je to da nude listu svih elemenata odnosno u ovom slučaju automobila te opcije poput osvježavanja pogleda , pretraživanja elemenata te naravno osnovne CRUD operacije. Nakon instanciranja forme s pregledom automobila odlazi zahtjev na server koji bi trebao vratiti listu svih automobila te korisnik može započeti s radom. Dok god je forma upaljena korisnik može odabrati opcije poput unosa novog automobila. Tada se instancira novi modul za dodavanje automobila te se instancira modul za dodavanje automobila koji će kreirati objekte , odnosno podatke o novim unosima će slati na server te od njega primiti poruke o statusu našeg upita odnosno o uspješnom izvršavanju ili pak o grešci. Nakon tog se modul za dodavanje vozila dealocira.

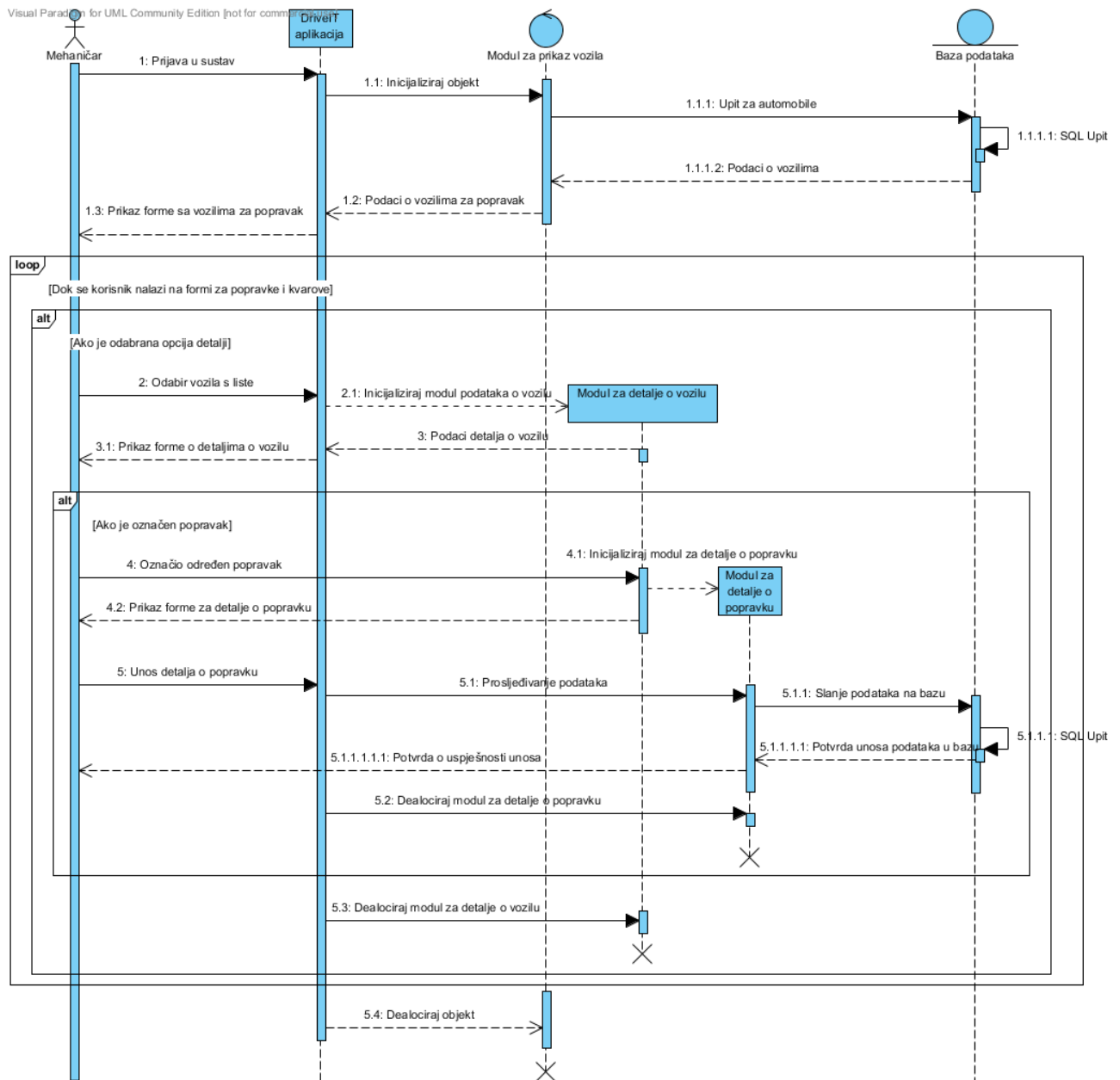
Odabirom bilo kojeg elementa u listi otvara se detaljniji prikaz karakteristika vozila te tu možemo odabrati opcije modifikacije ili brisanja određenog elementa liste ( vozila ) .U oba slučaja procedura je ista jer naš modul će kontaktirati server koji će nam vratiti poruku o statusu našeg upita. Izvršavanjem neke od ovih akcija modul za prikaz detalja o automobilu se automatski dealocira.

Još neke dodatne mogućnosti modula za prikaz vozila bi bile pretraga vozila i osvježavanje liste.

Kod pretrage vozila ne trebamo kontaktirati server jer pretražujemo listu automobila koji su učitani u trenutku kad smo upalili ovu formu u aplikaciji.

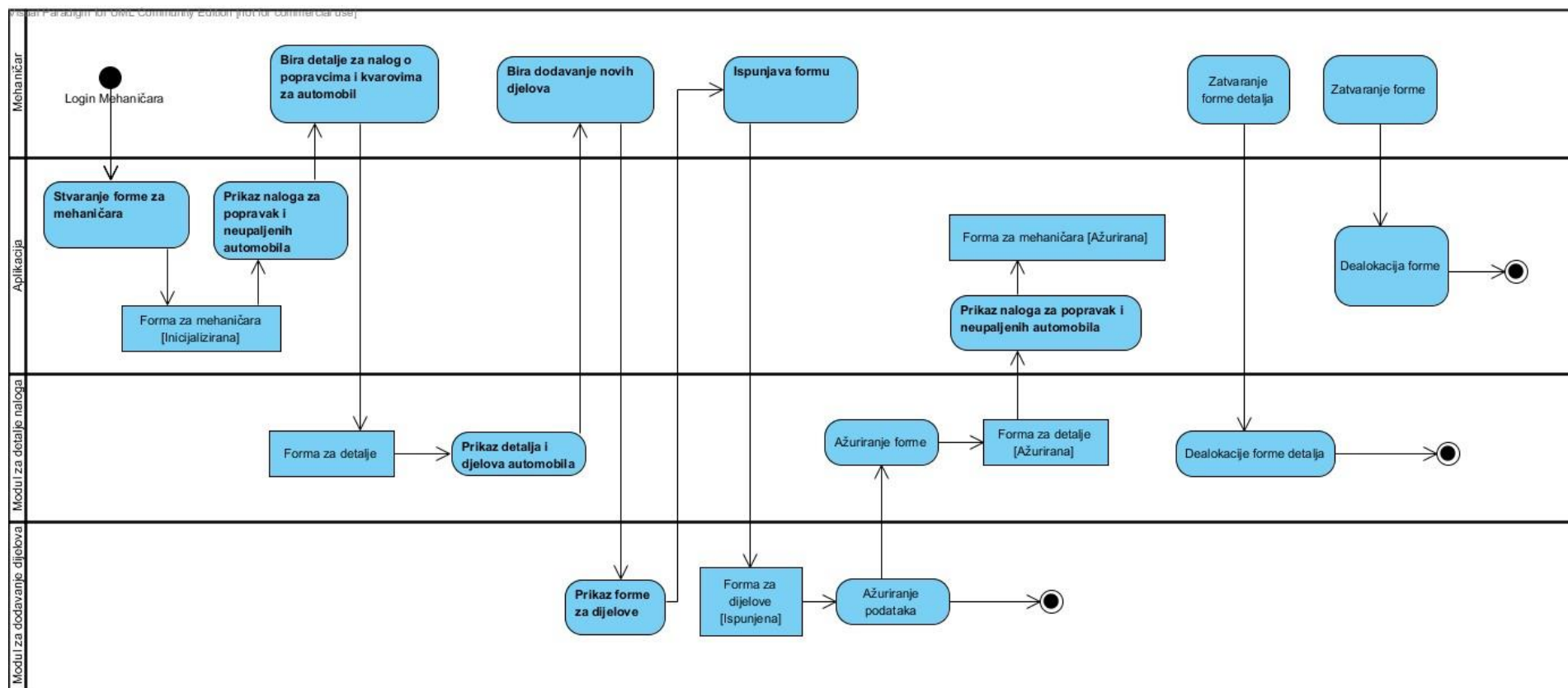
Kod osvježavanja aplikacije zovemo modul za osvježavanje aplikacije koji nam samo povlači podatke sa servera i sprema ih opet u listu automobila. Nakon što se spomenuto obavi, javlja se poruka o uspješnosti operacije i objekt se sam dealocira. Nakon završetka rada objekt se prilikom gašenja forme sam dealocira.

#### 4.7.4. Dijagram slijeda – Popravka i kvarova



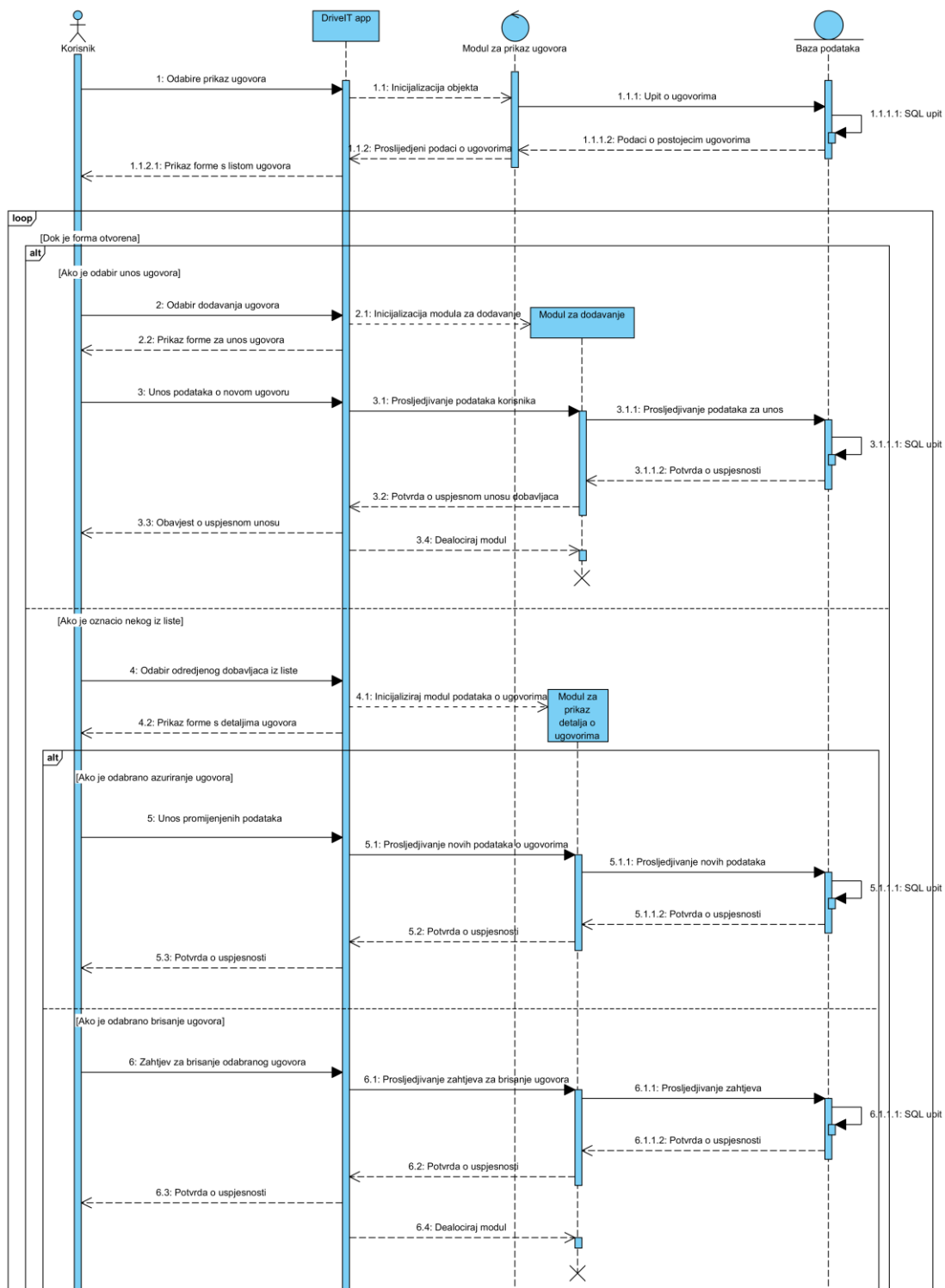
Na ovom dijagramu prikazan je slijed popravaka i kvarova za mehaničara. Prilikom prijave u sustav inicijalizira se modul za prikaz vozila koji dohvaća podatke iz baze, zatim se filtriraju sva vozila za popravak i vraćaju korisniku unutar forme. Sve dok se korisnik nalazi na formi za popravke i kvarove, mogu se vidjeti detalji o označenom automobilu s liste. U tom slučaju se inicijalizira modul za detalje o vozilu nakon čega se korisniku prikaže prikaz forme detalja vozila. Ukoliko korisnik označi popravak vozila, inicijalizira se novi modul za detalje o popravku koji se prikazuje korisniku. Korisnik unosi detalje o popravku, koji se prosljeđuju preko modula u bazu podataka koja vraća potvrdu o uspješnosti koja se ispisuje korisniku. Nakon toga se dealocira modul za detalje o popravku. Također na kraju se dealociraju moduli za detalje o vozilu i modul za prikaz vozila

#### 4.7.5. Dijagram aktivnosti – Popravka i kvarova

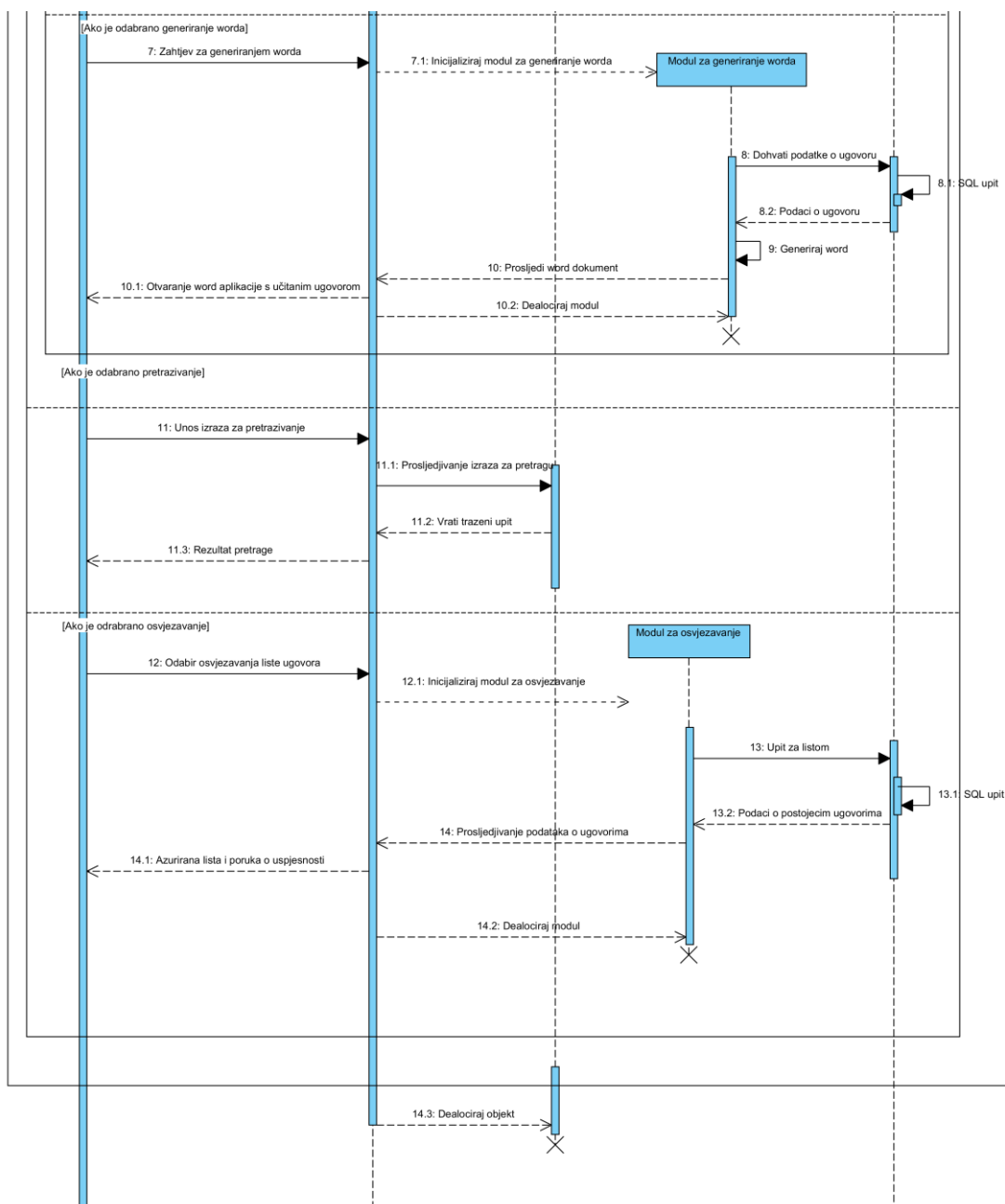


## 4.7.6. Dijagram slijeda – Ugovori

UML Paradigm for UML Community Edition [not for commercial use]

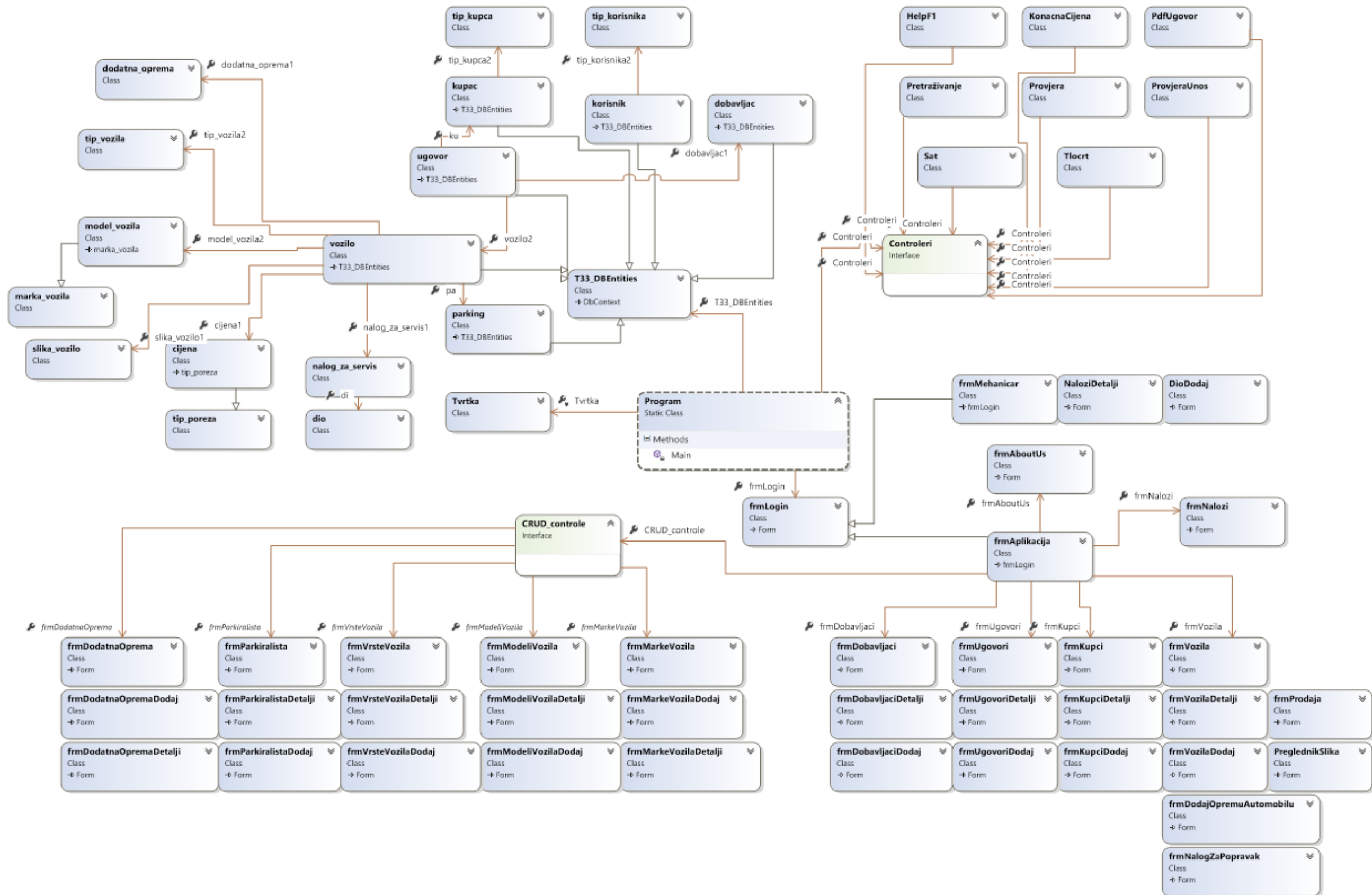




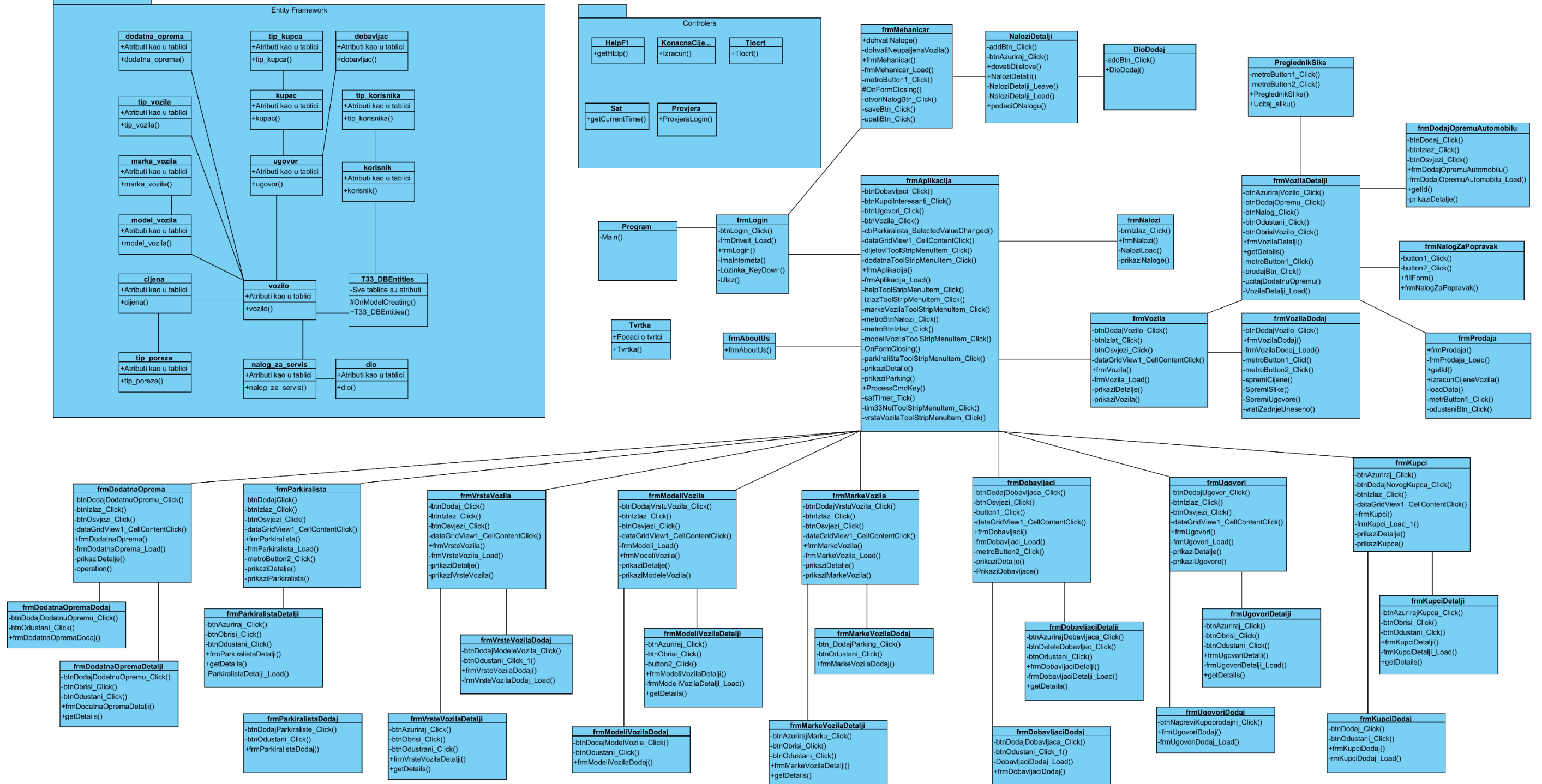


Na slici je opisan dijagram slijeda aktivnosti ukoliko korisnik odabere opciju “Ugovori”. Dakle aktivnost počinje kada korisnik iz glavne forme odabere opciju “Ugovori”, zatim se u pozadini inicijalizira modul za prikaz ugovora i korisniku se prikaže forma s listom svih ugovora koji su do sada uneseni i korisnik može s dodati novi ugovor, pretražiti sve ugovore ili ih osvježiti. Klikom na određeni ugovor se prikažu svi detalji tog ugovora te korisnik ima mogućnost da ažurira taj ugovor, obriše ili da generira word dokument iz njega. Prilikom generiranja word dokumenta otvara se aplikacija u kojoj se otvara generirani word dokument i korisnik ima mogućnost printanja iz samog word dokumenta. Nakon izlaska iz forme dealociraju se moduli.

## 4.8. Dijagram klasa



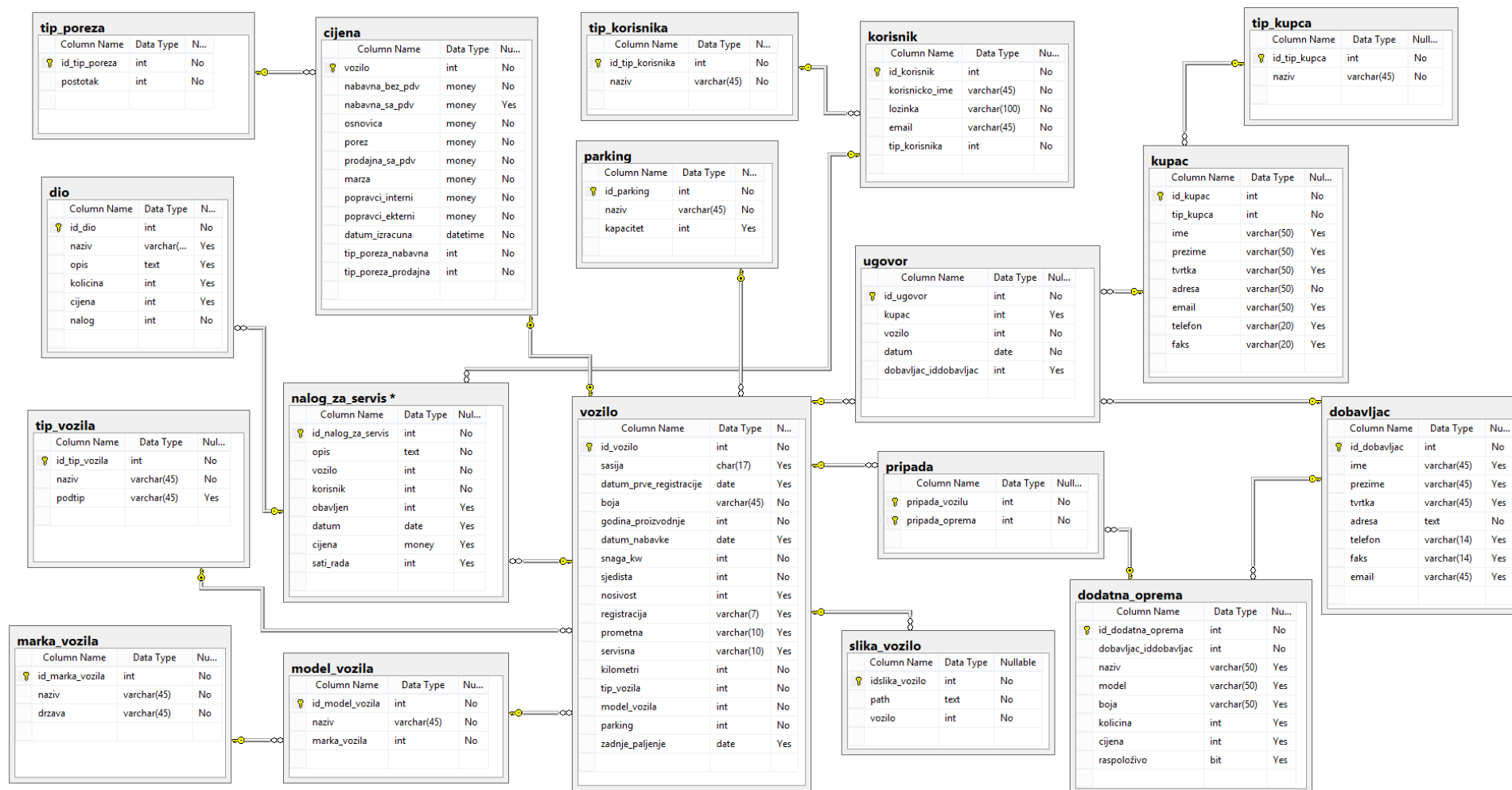
Slika 7. Zamisljeni dijagram klasa



Slika 8. Realizirani dijagram klasa

Dijagram klasa je vrsta strukturnog statičkog UML dijagrama koji opisuje strukturu modeliranog sustava: klase, attribute, operacije i statičke veze među klasama. Klasa predstavlja skup sličnih objekata, odnosno objekata koji posjeduju iste attribute, metode i odnose pa su tako, osim atributa, definirane i operacije kojima se ostvaruje funkcionalnost aplikacije koja se izrađuje. Kod našeg dijagrama klasa nastojali smo što bolje implementirati MVC uzorak dizajna prema kojem imamo odvojene slojeve prezentacije, logike i upravljanja podacima te smo mu nadodali i servise za komunikaciju sa serverom. Servisi se nadovezuju na model u kojem imamo približne klase koje odgovaraju tablicama u era modelu te ćemo njih popunjavati podacima koje ćemo slati kontroleru na obradu i manipulaciju. Svi podaci će ići i biti obrađivani preko i u kontroleru koji će popunjavati klase djecu od apstraktne klase "view". U kontroleru ćemo imati dva dodatna modula za ispis i generiranje dokumenata koji će biti zasebne komponente te ćemo implementirati osluškivač koji će nakon svakog logina u sustav javljati zaposleniku koji automobili se nisu dugo palili. Klase djeca od klase View će biti forme koje smo podijelili u formu za prijavu, CRUD forme i interaktivno parkiralište i svaka od njih poziva odgovarajuće metode u kontroleru koji dalje komunicira s modelom ili sa serverom preko modela.

## 4.9. Era model



Slika 9. Era model

ERA model (EVA - entiteti veze atributi) je konceptualni model podataka koji realni svijet vidi kroz entitete i njihove atribute. On se jednostavno transformira u relacijski. Entiteti su opisani atributima koji se kasnije pretvore u atribute pripadnih tablica. Za prikaz veza više-više potrebno je kreirati dodatni asocijativni entitet koji povezuje ta dva. Glavna tablica na koju se veže većina drugih je „vozilo“. Ona sadrži osnovne podatke o vozilu, a ostali podaci se nalaze u povezanim tablicama. Svako vozilo ima dobavljača od kojeg je nabavljeno, te ima ili će imati kupca koji ga je kupio. S objema strankama sklapaju se ugovori koje je potrebno evidentirati. Za cijenu je bitno definirati stope poreza pri kupnji i prodaji. Pošto aplikaciji pristupa više različitih korisnika potrebno je njihove podatke zabilježiti u tablici korisnik.