

VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE TEHNOLOGIJE



PROJEKTOVANJE INFORMACIONIH SISTEMA

Projektni zadatak: Rent a car

Predmetni nastavnik:
mg. Suzana Marković

Student:
Milan Savić 03/06

Datum predaje

UVOD

Pri radu na izradi dokumentacije za projekat rent-a-car aplikacije, koristio sam osnovne koncepte objektno-orjentisanog pristupa, preko faze analize i projektovanja, korišćenjem UML jezika modelovanja, i njegovih koncepata kao što su dijagrami klasa, dijagrami sekvenci, dijagrami kolaboracije, use-case dijagrami, dijagrami aktivnosti. Pri dekomponovanju sistema, da bi se stiglo do slučajeva korišćenja upotrebljavana je SSA – strukturna sistemska analiza. Pokazaću vam tri modela SSA, kako se jedan proces deli na više procesa. Sve dok se ne stigne do procesa koji više ne može da se deli. Da bi jedan sistem normalno funkcionisao mora da ima i svoju bazu podataka. Za projektovanje baze podataka rent-a-car koristio sam paralelno prošireni model objekti-veze, tj. PMOV, relacioni model, kao i vladajuća IDEF1X notacija. Koristio sam jos i dijagramne klasa, za tačno opisivanje svih procesa, objekata i atributa.

Za izradu same aplikacije koristio sam alat Visual studio 2005, sa pratećim alatima, kao i Telelerik RadControls for ASP.Net koje sam skinuo sa sajta www.telerik.com. Uz pomoc tih kontrola dobio sam bolju vizuelnu sliku same aplikacije.

Posle projektovanja baze podataka, uz pomoć alata SQL Servet Menagement Studio 2005 napravio sam bazu, sa kolonama i poljima.

Projektovanje svih dijagrama radio sam u Visio 2003, koji se nalazi u paketu MS Office 2003. Postoji još jedan alat koji se zove Rational Rose, pomoću koga se takođe projektuju dijagrami.

RENT-A-CAR SCENARIO

VERBALNI OPIS SISTEMA

Preduzeće rent-a-car se bavi iznajmljivanjem automobila klijentima. Svoj vozni park mora da vremenom osveži. Na osnovu toga firma šalje svim dobavljačima poruku da je spremna uložiti novac u vozni park. Dobavljači šalju svoje kataloge rent a car-u, koja odabira najbolje ponude. Posle odabira, šalje se zahtev za isporuku novih automobila. Dobavljač kada primi zahtev za isporuku, priprema ugovor o isporuci i plaćanju automobila. Ugovor se šalje rent a car-u, koji mora da vrati informaciju da se slaže sa ugovorom. Posle toga sledi isporuka automobila na dogovorenu adresu. Posle primanja automobila, službenik potpisuje prijemnicu automobila, i način plaćanja. Primljene automobile unosi u bazu podataka za vozni park.

Svaka firma rent a car mora da ima bazu podataka u koju će da upisuje klijente. Na osnovu toga firma stalnim mušterijama daje određene pogodnosti i popuste. To se radi tako što službenik svaku mušteriju upiše u bazu podataka.

Pored toga svaka nova mušterija se upisuje u bazu podataka da bi se imala evidencija o iznajmljenim automobilima. Kada mušterija dođe u firmu rent a car i zatraži automobil, službenik da upoznaje sa voznim parkom firme, dajući mu katalog automobila. Posle odabira, obe strane se dogovore o dodatnim troškovima i načinu plaćanja za iznajmljeni automobil. Otvara se novi ugovor, popunjava se sa svim zahtevima mušterije i načinom plaćanja. Posle čega se ugovor štampa i obe strane dobijaju po jedan primerak.

Prilikom vraćanja automobila službenik pregleda stanje automobila posle korišćenja. Ako je sve u redu, službenik naplaćuje samo cenu korišćenja automobila, eventualno se naplaćuje ako je kilometraža veća od ugovorene. Isto tako se naplaćuje posebno ako je mušterija vratila automobila posle dogovorenog roka. U slučaju da je automobil oštećen prilikom korišćenja, službenik od mušterije naplaćuje troškove za popravku. Kada službenik naplati uslugu i proveri sve pojedinosti na kolima, mora da poništi ugovor u računar, nakon čega mušteriji daje račun za naplaćenu uslugu.

Ako mušterija želi da produži rentu za automobil mora da se najavi. On tada zove telefonom rent a car kuću i službeniku daje broj ugovora. Na osnovu broja ugovora službenik pronalazi ugovor u bazi i pamera datum za vraćanje i kilometražu. Tada se mušterija obaveštava o promeni ugovora.

Svakom klijentu službenik je dužan uzeti lične podatke i upisati ih u bazu podataka.

STRUKTURNA SISTEMSKA ANALIZA - SSA

OSNOVNI PODATCI

Strukturalna sistemska analiza (SSA) je jedna potpuna metodologija za specifikaciju informacionog sistema, odnosno softvera. Ona se na različite načine može povezati sa metodama drugih faza u neku specifičnu metodologiju celokupnog razvoja IS. Tako na primer, ona može biti polazna osnova za metodu Strukturalnog projektovanja programa, ili projektovanja logičke strukture baze podataka metodom normalizacije, ili se može tretirati kao metodološki postupak dekompozicije nekog sistema na podsisteme sa ciljem da se, nalaženjem modela podataka podsistema i njihovom integracijom, dođe do potpunog modela podataka posmatranog sistema.

Potpuna, tačna, formalna i jasna specifikacija IS, ili kako se to obično kaže, specifikacija zahteva korisnika, zahteva koje budući sistem treba da zadovolji, predstavlja bitan preduslov za uspešno dalje projektovanje i implementaciju sistema. Očigledno je zbog čega specifikacija IS treba da bude potpuna i tačna. Zahtev da specifikacija bude formalna iskazuje se zbog toga što je formalna specifikacija osnov za "transformaciono" projektovanje i implementaciju, za automatizovano generisanje baze podataka i programa iz nje, odnosno za korišćenje CASE sistema. Zahtev da specifikacija bude jasna iskazuje se zbog toga što u specifikaciji IS u velikoj meri učestvuju korisnici sistema, neinformatičari, pa jezik specifikacije mora biti i njima prihvatljiv. Originalna SSA čiji su tvorci Yourdon i njegovi saradnici (DeMarco i drugi) poseduje veoma jednostavne, grafičke, pa samim tim i jasne koncepte. Ovde su svi ovi koncepti zadržani, a strožija formalizacija je dodata samo za opis strukture tokova i skladišta podataka, da bi se obezbedio specifičan transformacioni razvoj IS koji Standardna metodologija zagovara.

Kao što je već ranije rečeno, specifikacija IS treba da prikaže (potpuno, tačno, formalno i jasno) šta budući informacioni sistem treba da radi. Veoma je bitno odmah istaći da specifikacija IS prikazuje šta IS treba da da, a ne i kako to treba da ostvari. Očigledno je da prerano definisanje "kako", odnosno davanje nekih projektantskih rešenja u okviru specifikacije, ograničava kasniji mogući izbor (optimizaciju) načina implementacije sistema. Odgovor na pitanje kako daje se za konkretno okruženje, za definisanu tehnologiju i organizaciju u kojoj se sistem implementira. Da specifikacija ne bi sadržala tehnološki i organizaciono ograničena rešenja, obično se kaže da ona treba da opiše funkcionisanje IS u "idealnoj tehnologiji", gde praktično nikakva ograničenja ne postoje.

SSA posmatra informacioni sistem kao funkciju (proces obrade) koja, na bazi ulaznih, generiše izlazne podatke. Ulazni podaci se dovode u proces obrade, a izlazni iz njega odvođe preko tokova podataka. Tok podataka se tretira kao vod ili kao pokretna traka kroz koji stalno teku ili koja stalno nosi podatke na najrazličitijim nosiocima – papirni dokumenti, niz poruka koje čovek unosi preko tastature terminala, "paket" informacija dobijen preko neke telekomunikacione linije ili slično. Imajući u vidu zahtev da specifikacija treba da se oslobodi svih implementacionih detalja od interesa su samo sadržaj i struktura ulaznog toka, a ne i medijum nosilac toka.

Izvori ulaznih, odnosno ponori izlaznih tokova podataka mogu biti objekti van IS koji sa IS komuniciraju i koji se u SSA nazivaju interfejsi, drugi procesi u sistemu, ili tzv skladišta. Skladišta podataka se posmatraju kao "tokovi u mirovanju", odnosno odloženi, akumulirani tokovi, različite vrste evidencija, arhiva, kartoteka i datoteka. I za skladišta kao i za tokove od interesa su isključivo njihov sadržaj i struktura.

Osnovni koncepti za specifikaciju IS u SSA su, znači, funkcije, odnosno procesi obrade podataka, tokovi podataka, skladišta podataka i interfejsi. Njihov međusobni odnos se prikazuje preko dijagrama toka podataka koji prikazuje vezu interfejsa, odnosno skladišta kao

izvora odnosno ponora podataka, sa odgovarajućim procesima, kao i međusobnu vezu procesa.

U ovom primeru koristiću sledeće grafičke simbole:



krug ili elipsa pretstavlja funkciju ili proces obrade podataka,



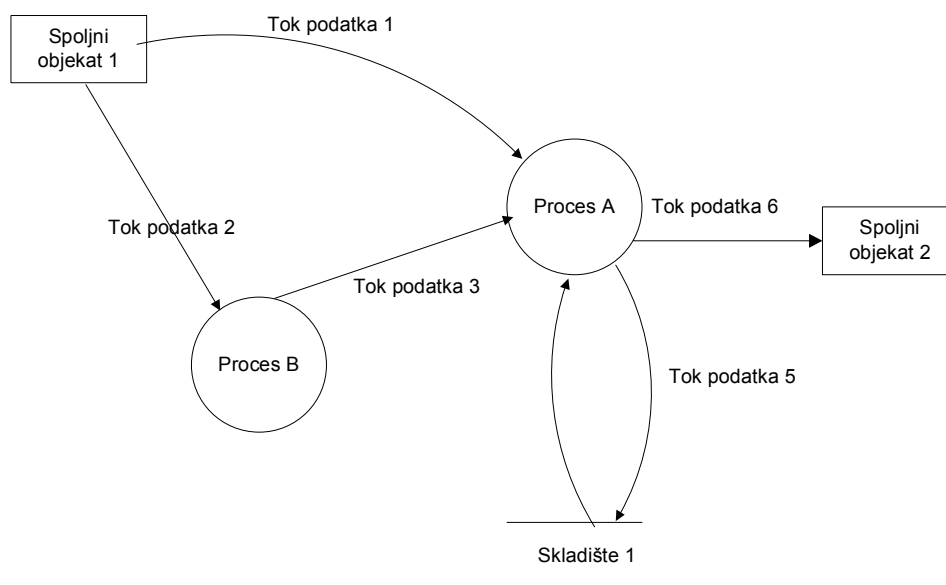
pravougaonik predstavlja interfejs,



usmerena linija predstavlja tok podataka,

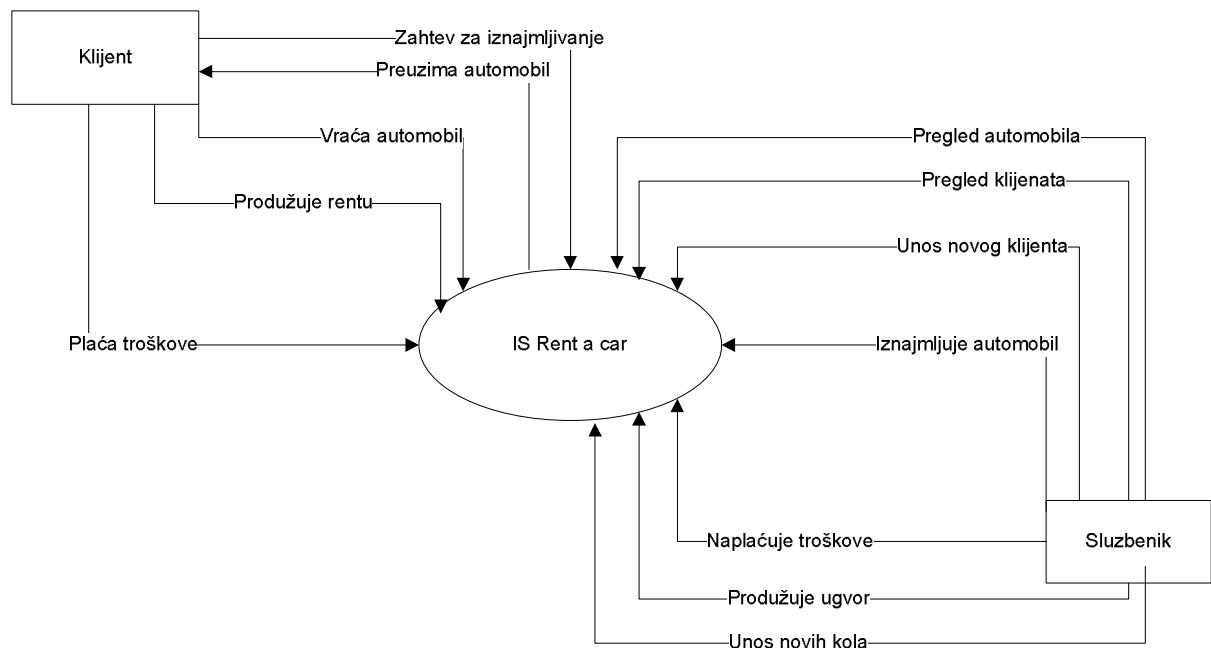


dve paralelne linije ("otvoreni" pravougaonik) predstavlja skladište podataka

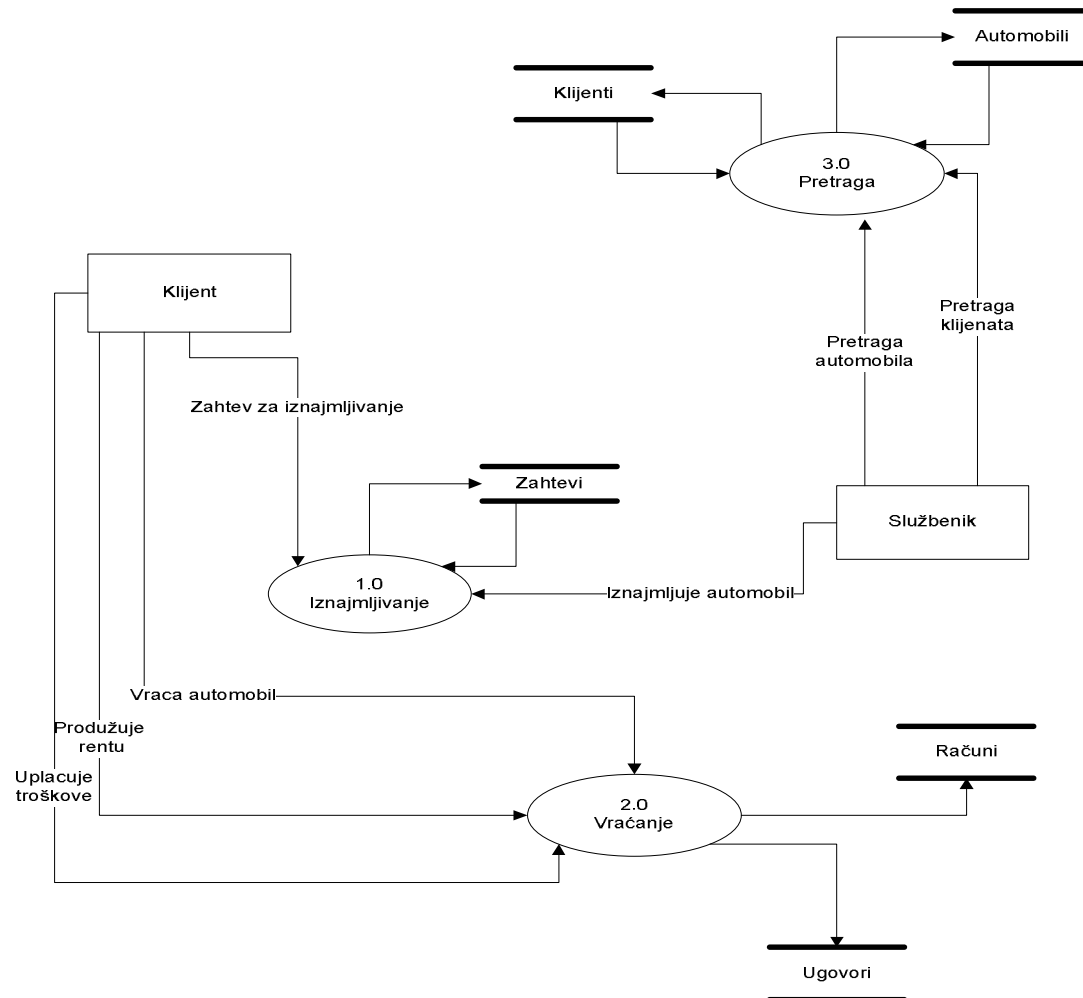


Slika 1. Dijagram toka podataka

Očigledno je da se jedan IS sastoji iz mnoštva procesa, interfejsa, tokova i skladišta podataka. Specifikacija IS treba da bude potpuna (detaljna) i jasna. Kada bi se jedan sistem detaljno opisao i prikazao jednim dijagramom toka podataka, dobio bi se veoma nejasan opis sistema, paukova mreža procesa, tokova, skladišta i interfejsa. Istovremeno detaljan i jasan opis sistema zahteva opis na "različitim nivoima apstrakcije", odnosno hijerarhijski opis u kome se na višim nivoima sistem opisuje opštije, a na nižim, postepenim i organizovanim uvođenjem detalja, potpuno i detaljno. Hijerarhijski opis sistema u tehnici dijagrama tokova podataka se svodi na to da se na višim nivoima definišu globalniji procesi, a da se zatim svaki takav globalni proces, na sledećem nižem nivou, pretstavi novim dijagramom toka podataka. Dijagram toka podataka na vrhu ovakve hijerarhije naziva se **dijagram konteksta**, a procesi na najnižem nivou (procesi koji se dalje ne dekomponuju) nazivaju se primitivni procesi.

DIJAGRAM KONTEKSTA RENT A CAR**Slika 2. Dijagram konteksta Rent A Car****PROCES IS RENT A CAR**

IS Rent a car je predviđen da radi sa klijentima i automobilima. Vidimo da na dijagramu postoje dva objekta, a to su klijent i službenik. Svi zahtevi i akcije su regulisani tokovima podataka, koji su pokazatelji kako objekti komuniciraju sa sistemom.

DIJAGRAM DEKOMPOZICIJE PRVOG NIVOA**Slika 3. Dekompozicija prvog nivoa**

Posle dekompozicije IS firme Rent a car dobijamo nova tri procesa, koji prate sva desavanja u firmi. Sva tri procesa nastala su iz tokova podataka, koji su naznačeni u prvom dijagramu.

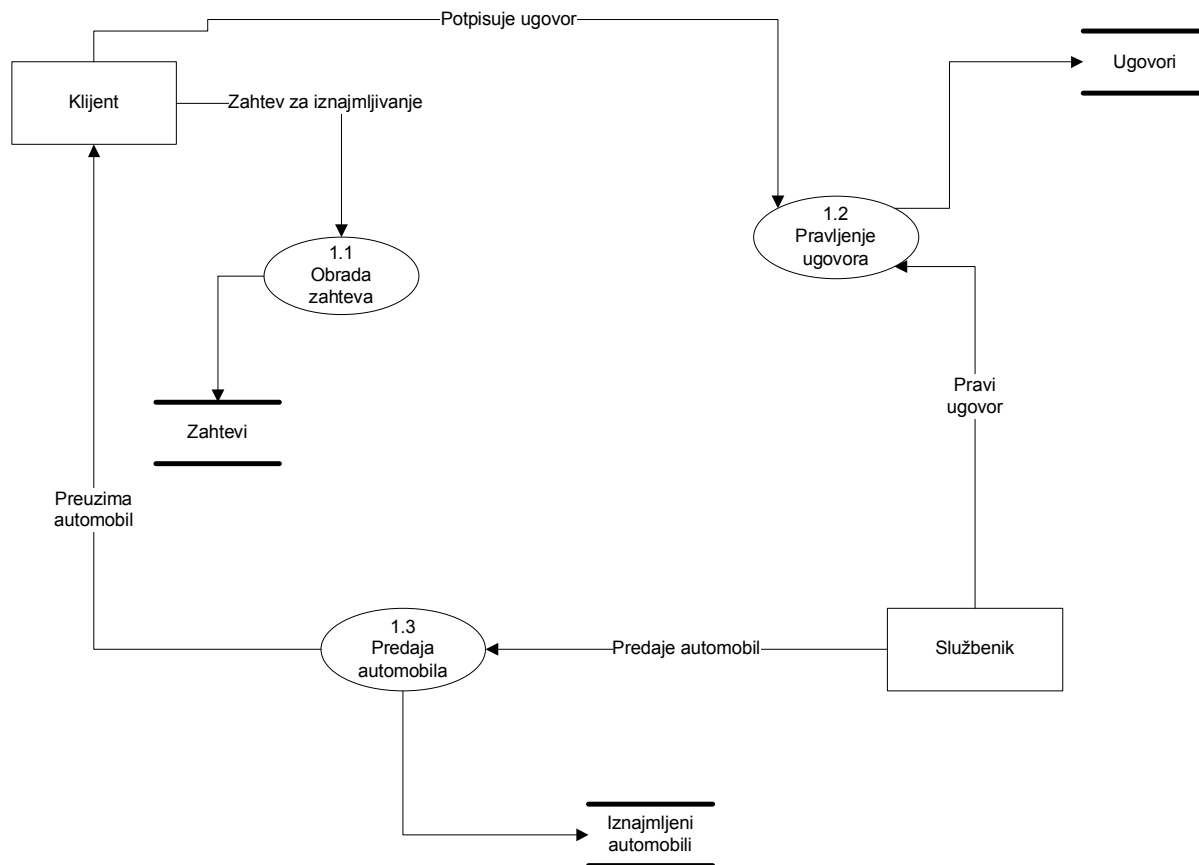
Dobijeni procesi posle dekompozicije :

- Iznajmljivanje
- Vraćanje
- Pretraga

Dobijena skladišta podataka :

- Ugovori
- Računi
- Zahtevi
- Klijenti
- Automobilu

DRUGI NIVO DEKOMPOZICIJE – 1. IZNAJMLJIVANJE



Slika 4. Drugi nivo dekompozicije - Iznajmljivanje

PROCES 1.1 – OBRADA ZAHTEVA

Posle uzimanja zahteva za iznajmljivanje automobila od klijenta, službenik pristupa pretraživanju. Zahtev smešta u bazu radi arhive.

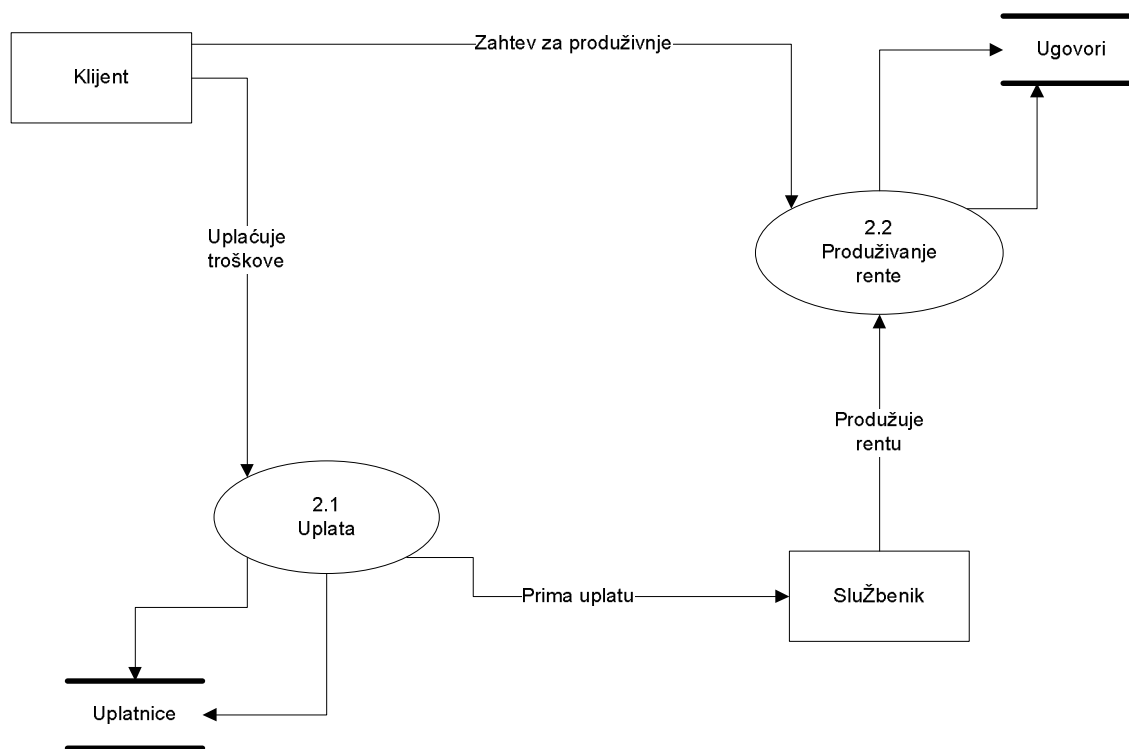
PROCES 1.2 – PRAVLJENJE UGOVORA

Kada se obe strane slože oko izbora automobila, cene i svih dodatnih troškova, službenik pristupa pravljenju ugovora. Jedan primerak predaje klijentu, drugi smešta u bazu ugovora.

PROCES 1.3 – PREDAJA AUTOMOBILA KLIJENTU

Nakon potpisivanja ugovora službenik predaje automobil klijentu.

DRUGI NIVO DEKOMOICIJE – 2. VRAĆANJE



Slika 5. Drugi nivo dekompozicije - Vraćanje

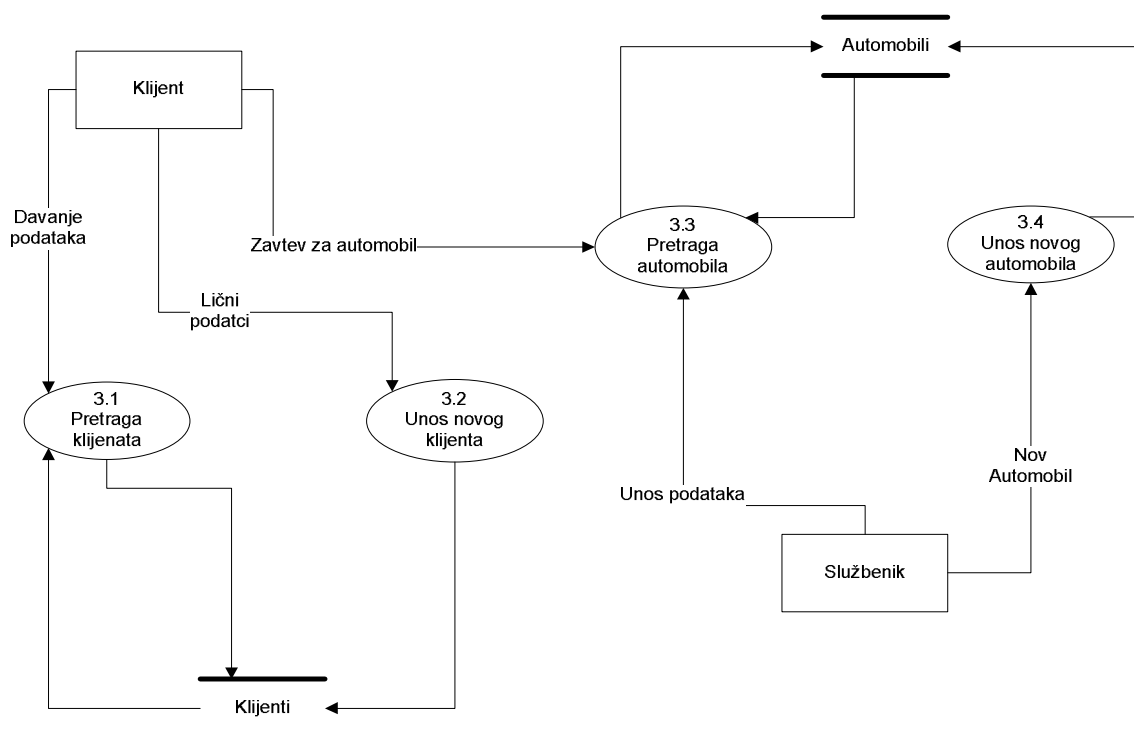
PROCES 2.1 – UPLATA

Ovim procesom vidimo da klijent nakon što je završio rentiranje automobila pristupa uplati za uslugu. Službenik prima uplatu od klijenta i zavodi je u bazu uplatnica.

PROCES 2.2 – PRODUŽIVANJE RENTE

Ovaj proces objašnjava tok podataka kada klijent želi da produži rentu na iznajmljeni automobil. Klijent službenika obaveštava o produživanju i tada službenik u bazi traži ugovor koji firma ima sa klijentom. Poništava stari datum vraćanja i stavlja novi dogovoreni datum. Klijent je saglasan da je i cena zbog produživanja mnogo veća. Nakon obavljenog razgovora službenik novi ugovor smešta u bazu ugovora.

DRUGI NIVO DEKOMPOZICIJE – 3. PRETRAGA



Slika 6. Drugi nivo dekompozicije - Vraćanje

PROCES 3.1 PRETRAGA KLIJENATA

Proces obuhvata pretragu baze podataka u sistemu. Znamo da se tu nalaze svi klijenti od ranije, i koji su upisani. Pretraga se vrši isključivo unošenjem ličnih podataka klijenta, na koje sistem vraća poruku da li se klijent nalazi, ili se ne nalazi u bazi podataka. Ako se klijent nalazi u bazi, u njegov profil se upisuje renta koju uzima, radi evidencije i kasnije nekih popusta za stalne mušterije. Ako se klijent ne nalazi u bazi pristupa se sledećem procesu.

PROCES 3.1 PRETRAGA KLIJENATA

Ovim procesom u sistem se upisuju podatci klijenta koji je toga datuma izvršio rentiranje automobila. Sistem mora da potvrdi da je unos uspešan.

PROCES 3.1 PRETRAGA AUTOMOBILA

Proces obuhvata pretraživanje baze automobila na zahtev klijenta. Sistem vraća poruku o uspešnosti pretrage. Ako je automobil pronađen, sistem na ekranu ispisuje podatke o automobilu, cenu i status automobila. Ako automobil nije pronađen sistem vraća poruku o neuspešnosti pretrage.

PROCES 3.1 PRETRAGA KLIJENATA

Isto kao i sa klijentima. Posle kupovine novog automobila, sve karakteristike se upisuju u bazu podataka. Sistem vraća poruku o uspešnosti unosa novog automobila

REČNIK PODATAKA

Rečnik podataka, kao što je ranije rečeno, daje opis strukture i sadržaja svih tokova i skladišta podataka. Bez obzira šta tok ili skladište podataka predstavljaju, papirni dokumenat, niz karaktera kao ulaz sa terminala, "paket" informacija dobijen telekomunikacionom linijom, kartoteku ili datoteku, kao logička struktura podataka oni predstavljaju neku kompoziciju polja. Da bi precizno definisali logičku strukturu skladišta i tokova i definisali sintaksu rečnika neophodno je da uvedemo definicije svi koncepata rečnika:

1) *Polje* je elementarna (atomska) struktura koja se dalje ne dekomponuje i koja ima svoju vrednost.

2) Polja svoje vrednosti uzimaju iz skupova vrednosti koji se nazivaju *domenima*. Domeni mogu biti:

- "predefinisani", odnosno standardni programsko-jezički domen, kao što su INTEGER, CHARACTER, REAL, LOGICAL i DATE.
- "semantički", kada se definišu posebno, preko svoga imena, predefinisane domene i, eventualno, ograničenja na mogući skup vrednosti predefinisane domena.

3) Pored ograničenja na vrednosti polja, odnosno vrednosti domena koja su data u primerima definišu se i druga. *Ograničenja* mogu biti prosta i složena. Lista dozvoljenih prostih ograničenja je:

(a) konstanta, gde je bilo koji operator poređenja koji se na datom domenu može definisati (na primer, <, >, =, <=, >= za brojne domene), a konstanta je neka definisana vrednost iz datog domena.

(b) BETWEEN konstanta , konstanta, gde su konstante vrednosti iz datog domena.

(c) IN (lista vrednosti), gde se lista formira od konstanti iz odgovarajućeg domena.

(d) NOT NULL, kada dato polje ne može da dobije "nulla vrednost", odnosno mora uvek da ima vrednost.

REČNIK PODATAKA ZA IS RENT A CAR

TABELA AUTOMOBILI

NAZIV PODATKA	DRUGI NAZIV PODATKA	DOMEN	OGRANICENJE	NAPOMENA
Ime proizvođača automobila	Proizvodjac	Varchar (150)	Not null	
Model automobila	Model	Varchar (150)	Not null	
Kategorija Automobila	Kategorija	Varchar (150)	Not null	
Broj vrata	BR_vrata	Varchar (150)	Not null	

Vrsta goriva	Vrsta_goriva	Varchar (150)	Not null	
Menjač na Automobilu	Menjac	Varchar (150)		
Godište Automobila	Godiste	Varchar (150)	Null	
Broj šasiije	BR_sasiije	Varchar (150)	Not null	Primary key
Broj motora	BR_motora	Varchar (150)	Not null	
Boja Automobila	Boja	Varchar (150)	Not null	
Kupovna cena Automobila	Kupovna_cena	Varchar (150)	Null	
Cena iznajmljivanja	Cena_iznajmljivanja	Varchar (150)	Null	
Kubikaža Automobila	CC	Varchar (150)	Null	
Broj registracije	BR_registracije	Varchar (150)	Not null	
Prva registracija	Prva_registracija	Varchar (150)	Null	
Pređena Kilometraža	Kilometraza	Varchar (150)	Null	
Dodatna Oprema	Dodatna_oprema	Varchar (150)	Null	
Standard Automobila	Standard	Varhar (150)	Null	

TABELA IZNAJMI

NAZIV PODATKA	DRUGI NAZIV PODATKA	DOMEN	OGRANICENJE	NAPOMENA
Broj ugovora	[Broj ugovora]	Int	Null	Primary Key
Broj šasiije	BR_Sasiije	Varchar (150)	Null	
Matični broj	Mat_Br	Varchar (150)	Null	
Datum iznajmljivanja Automobila	Datum_iznajmljivanja	Varchar (150)	Not null	
Datum vraćanja Automobila	Datum_Vracanja	Varchar (150)	Not null	

TABELA KLIJENT

NAZIV PODATKA	DRUGI NAZIV PODATKA	DOMEN	OGRANICENJE	NAPOMENA
Ime klijenta	Ime	Varchar (150)	Not null	
Prezime klijenta	Prezime	Varchar (150)	Nut null	
Adresa klijenta	Adresa	Varchar (150)	Not null	
Kontakt telefon	Telefon	Varcab (150)	Not null	
Matični broj	Mat_br	Varchar (150)	Not null	Primary key
Broj lične karte	Broj_LK	Varchar (150)	Not null	
Broj pasosa	Br_pasosa	Varchar (150)	Null	
Naziv firme Klijenta	Naziv_firme	Varcab (150)	Null	
Adresa firme	Adresa_firme	Varcab (150)	Null	
Telefon firme	Tel_firme	Varchar (150)	Null	
PIB firme	PIB	Varchar (150)	Null	
Matični broj Firme	MatBR_Firme	Varchar (150)	Null	
Informacije	Info	Varchar (150)	Null	

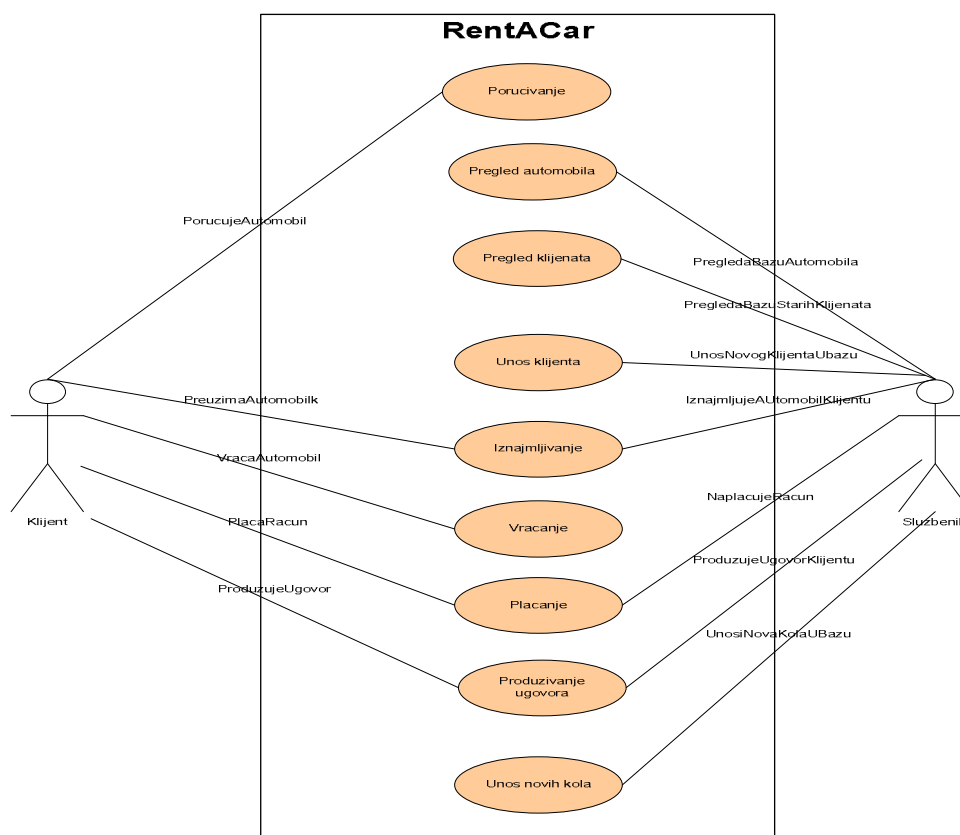
USE CASE – RENT A CAR

Slučajevi korišćenja modeluju dijalog između izvođača i sistema. One predstavljaju funkcionalnost koju obezbeđuje sistem. Skupina slučajeva korišćenja za neki sistem ustanovljava sve definisane načine na koje taj sistem može biti korišćen. Formalna definicija slučajeva korišćenja glasi: slučaj korišćenja je niz transakcija koje izvodi sistem, koji daje merljive rezultate od vrednosti za pojedinačnog izvođača. Između izvođača i slučaja korišćenja može postojati relacija asocijacije, koja se još naziva i komunikacionom asocijacijom. Mogu postojati dva tipa relacija između slučajeva korišćenja: include (uključi) i extend (proširi). Relacije include se formiraju između novog slučajeva korišćenja i svakog drugog slučaja korišćenja koji koristi njegovu funkcionalnost.

Relacija *extend* koristi se da prikaže :

- Opciono ponašanje
- Ponašanje koje se pokreće samo pod određenim uslovima
- Nekoliko različitih tokova koji mogu biti pokrenuti na osnovu izbora izvođača

Dijagram slucajeva koriscenja je grafički prikaz pojedinih ili svih izvođača, slučajeva korišćenja i njihovih interakcija.



Slika 7. Use Case - Rent a Car

Use Case 1. Naručivanje

Primarni akter : Klijent

Opis : Klijent naručuje automobil kod službenika, nakon što je izabrao automobil koji je želeo.

Preduslovi : Sistem priključen na napajanje, ekran uključen u sistem. Službenik koji je na radnom mestu. Radno vreme još u toku. Sistem prima ulazne zahteve klijenta

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Klijent ulazi u firmu, sa zahtevima za kola
3. Službenik pretražuje bazu automobila i nalazi zahtevani automobil
4. Klijent potvrđuje porudžbinu

Alternativni tok događaja : Službenik ne nalazi automobil koji odgovara zahtevu klijenta. Cena iznajmljivanja je visoka za klijenta.

Posledice : Porudžbina je završena

Pretpostavke : Pretpostavimo da klijent tačno zna koja kola mu trebaju

Use Case 2. Pregled automobila

Primarni akter : Službenik, sistem

Opis : Službenik na zahtev klijenta pregleda bazu automobila. On unosi tačno određene zahteve sistem. Službenik prenosi klijentu rezultate pretrage.

Preduslovi : Sistem priključen na napajanje, ekran uključen u sistem. Službenik koji je na radnom mestu. Radno vreme još u toku. Sistem prima ulazne zahteve klijenta. Sistem povezan sa bazom podataka.

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Klijent ulazi u firmu, sa zahtevima za kola
3. Službenik unosi kriterijume za pretraživanje baze
4. Službenik pretražuje bazu automobila i nalazi zahtevani automobil
5. Službenik prenosi klijentu rezultate pretrage

Alternativni tok događaja : Službenik ne nalazi automobil koji odgovara zahtevu klijenta. Službenik greši u kriterijumima i sistem izbacuje pogrešne rezultate.

Posledice : Traženi automobil je pronađen u bazi.

Pretpostavke : Pretpostavimo da klijent tačno zna koja kola mu trebaju. Samim tim i službenik unosi tačne kriterijume. Ako je automobil u bazi sistem će ga izbaciti na ekran.

Use Case 3. Pregled klijenata

Primarni akter : Službenik, sistem

Opis : Po završetku porudžbine službenik pregleda bazu klijenata. Sa cilje da proveri da li je klijent novi, pa je potrebno da se unese u bazu, ili je stari pri čemu dobija neki popust i pogodnosti

Preduslovi : Sistem priključen na napajanje, ekran uključen u sistem. Službenik koji je na radnom mestu. Radno vreme još u toku. Sistem prima ulazne zahteve klijenta. Sistem povezan sa bazom podataka.

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Porudžbina za kola je uspešno završena
3. Službenik pretražuje bazu sa klijentima po imenu i prezimenu
4. Službenik pronalazi klijenta u bazi, što znači da je on već stara mušterija. Dobija popust i pogodnosti za iznajmljivanje
5. Službenik ne nalazi klijenta u bazi

Alternativni tok događaja : Službenik ne nalazi klijenta u bazi, jer ne postoji. Traži se unos istog u bazi. Službenik je pogresio prilikom pretrage klijenata

Posledice : Traženi klijent je pronađen u bazi.

Pretpostavke : Pretpostavimo da službenik ne zna ko je stara, a ko nova mušterija.

Use Case 4. Unos klijenta

Primarni akter : Službenik, sistem

Opis : Po završetku pretrage klijenata, službenik uviđa da klijenta nema u bazi. Potrebno je uneti istog zbog evidencije

Preduslovi : Sistem priključen na napajanje, ekran uključen u sistem. Službenik koji je na radnom mestu. Radno vreme još u toku. Sistem prima ulazne zahteve klijenta. Sistem povezan sa bazom podataka.

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Službenik pretražuje bazu sa klijentima po imenu i prezimenu
3. Službenik ne nalazi klijenta u bazi
4. Službenik unosi novog klijenta u bazu

Alternativni tok događaja : Službenik je pogrešio prilikom pretrage klijenata. Uneo novog, ustvari starog ako već postoji. Sada u bazi imaju dva klijenta sa istim podacima.

Posledice : Klijent je uspešno unešen u bazu klijenata

Pretpostavke : nema

Use Case 5. Iznajmljivanje

Primarni akter : Službenik, klijent

Opis : Po uspešnom odabiru automobila, službenik iznajmljuje automobil klijentu

Preduslovi : Traženi automobil postoji u bazi, unešen novi klijent, stari klijent ažuriran u bazi, ugovor zaključen.

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Službenik zaključuje ugovor sa klijentom
3. Službenik odlazi da pokaže automobil klijentu
4. Službenik predaje klijentu iznajmljen automobil

Alternativni tok događaja : Klijent odustaje od vozila

Posledice : Klijent je uspešno iznajmio automobil

Pretpostavke : nema

Use Case 6. Vraćanje automobila

Primarni akter : Službenik, klijent

Opis : Vraćanje automobila

Preduslovi : Iznajmljen automobil, zaključen ugovor.

Tok događaja : 1. Klijent vraća automobil u propisanom roku
2. Službenik proverava stanje auta
3. Poništava ugovor između rentAcar-a i klijenta
4. Službenik pravi račun na osnovu ugovora

Alternativni tok događaja : Klijent nije vratio kola u predviđenom roku, kola su tokom korišćenja oštećena,

Posledice : Klijent je uspešno vratio automobil

Pretpostavke : Službenik je ovlašćen da preuzme vozilo

Use Case 7. Plaćanje

Primarni akter : Službenik, klijent

Opis : Pre isteka roka klijent vraća kola. Službenik pravi račun i klijent isti račun plaća, posle dogovora o načinu plaćanja

Preduslovi : Automobil je vraćen, ugovor poništen, račun napravljen, dogovoren način plaćanja

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Službenik pronalazi broj ugovora u bazi, proverava validnost i poništava ga
3. Službenik se sa klijentom dogovara o načinu plaćanja
4. Službenik predaje račun klijentu
5. Klijent račun plaća

Alternativni tok događaja : Nevažeća kreditna kartica, prazan bankovni račun, dodatni troškovi,

Posledice : Klijent je uspešno platio korišćenje automobila

Pretpostavke : Službenik ima pravo da naplati na licu mesta

Use Case 8. Produživanje ugovora

Primarni akter : Službenik, klijent

Opis : Pre isteka roka klijent traži da mu se ugovor produži.

Preduslovi : Automobil je iznajmljen, stari ugovor važeći

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Klijent traži da mu se produži ugovor o iznajmljivanju auta
3. Službenik na osnovu starog ugovora, pronalazi klijenta u bazi
4. Produžava mu ugovor
5. Unosi dodatne troškove

Alternativni tok događaja : Nemogućnost produženja ugovora

Posledice : Klijent je uspešno produžio ugovor o iznajmljivanju auta

Pretpostavke : Službenik je ovlašćen da produži ugovor sa klijentom

Use Case 9. Unos novih kola

Primarni akter : Službenik

Opis : Pošto je firma kupila nova kola, i podmladila vozni park. Ista kola treba uneti u bazu. Službenik, pošto je primio kola, unosi specifikacije sa svako vozilo ponaosob

Preduslovi : Zaključen ugovor sa autokućom o isporuci kola. Kola isporučena, kola primljena

Tok događaja : 1. Službenik je logovan sa svojim nalogom na sistem firme
2. Unosi podatke za svako vozilo
3. Azurira bazu voznog parka

Alternativni tok događaja : Vozila nisu isporučena

Posledice : Službenik je uspešno uneo i ažurirao bazu za vozni park

Pretpostavke : Službenik ima pravo pristupa bazi, za unošenje novih kola

DIJAGRAMI AKTIVNOSTI

Šta je dijagram aktivnosti?

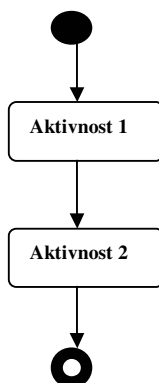
Dijagram aktivnosti je pojednostavljen prikaz onoga što se događa za vreme operacije ili procesa. Pokazuje korake u operaciji ili procesu, koji su nazvani aktivnosti, ali i tačke odluke (decision points) i grane (ogranke). Koristi se npr. za prikazivanje poslovnog procesa ili operacije.

Dijagram aktivnosti je ekstenzija (nastavak, dodatak) state diagrama - dijagrama stanja. Dijagram stanja pokazuje stanje objekta i aktivnosti prikazuje kao strelice koje povezuju stanje, dok dijagram aktivnosti ističe baš aktivnosti.

Svaka aktivnost je predstavljena zaobljenim pravougaonikom. Kad se završi odvijanje jedne aktivnosti, automatski se prelazi na odvijanje druge aktivnosti. Strelice predstavljaju kretanje od jedne aktivnosti do sledeće. Kao i dijagram stanja, dijagram aktivnosti ima početnu tačku prikazanu kao bikovo oko (bull's eye, kruga s manjim ispunjenim kružićem unutar njega)

Na slici vidimo :

1. početnu tačku
2. završnu tačku
3. dve aktivnosti
4. kretanje procesa

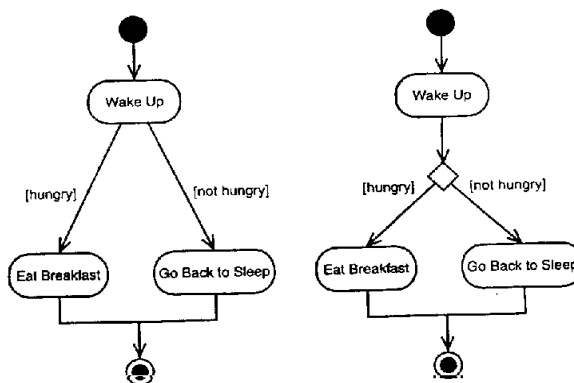


Slika 8. Dijagram aktivnosti

Odluke

Kad usled odvijanja aktivnosti dođe do tačke u kojoj jedan skup uslova vodi na jednu stazu, a drugi skup uslova na drugu stazu treba prikazati grananje.

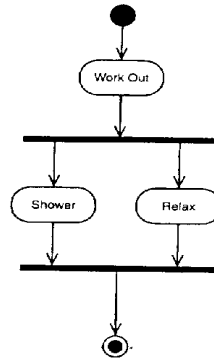
Tačka u kojoj dolazi do grananja se naziva **decision point** (tačka odluke) i može se prikazati na dva načina. Jedan način je da moguće staze izlaze (dolaze) direktno iz aktivnosti. Drugi način je da imamo tranziciju (prelaz) aktivnosti do "malog dijamanta" (◇ - simbol odluke iz flowcharta) i onda moguće staze iz tog simbola. U oba slučaja uslov koji vodi za određenu stazu je u uglatim zagradama blizu odgovarajuće staze.



Slika 9. Odluke

Uporedne staze

Usporedne staze su staze koje kreću u isto vreme, a kasnije se opet spajaju u jednu stazu. Odvijaju se uporedno. Separacija (odvajanje staza) se prikazuje podebljanim linijama okomitim na tranziciju (smer u kojem se prikazuje odvijanje operacije ili procesa). Spajanje usporednih staza prikazuje se podebljanom poprečnom linijom.

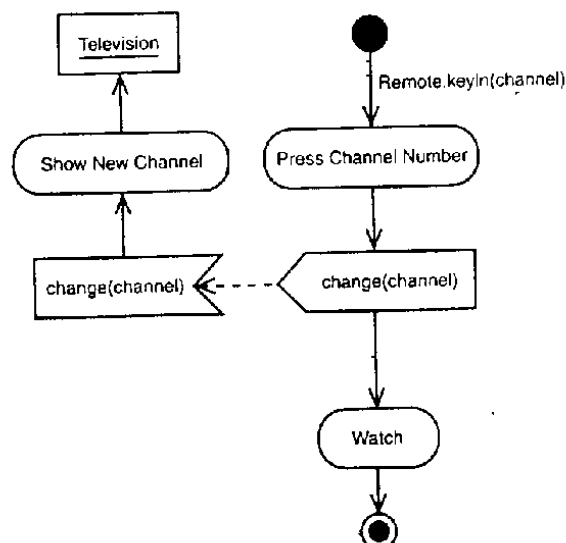


Slika 10. Uporedne staze

Signali

Za vrijeme sekvence aktivnosti moguće je poslati signal. Kad je primljen signal uzrokuje da aktivnost zauzme svoje mjesto.

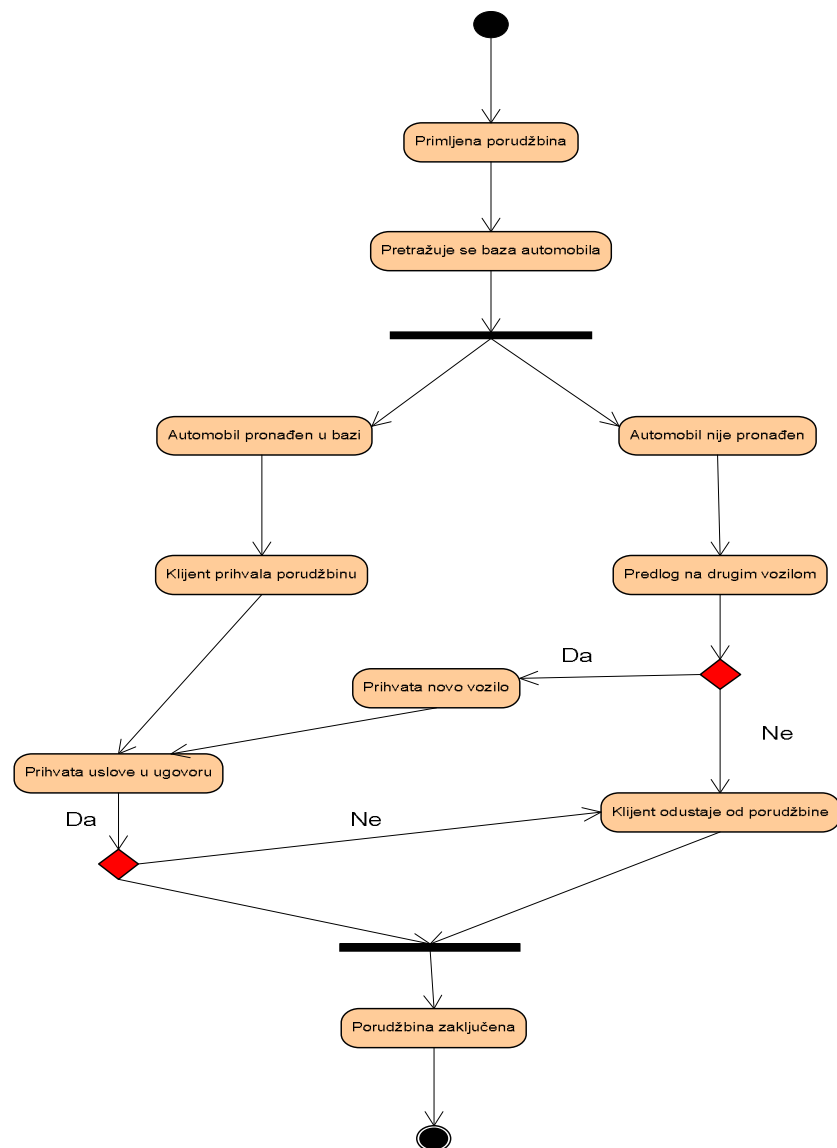
Simbol za slanje signala je konveksni peterokut.
 Simbol za primanje signala je konkavni peterokut.
 Konveksni poligon simbolizira output - događaj.
 Konkavni poligon simbolizira input - događaj.



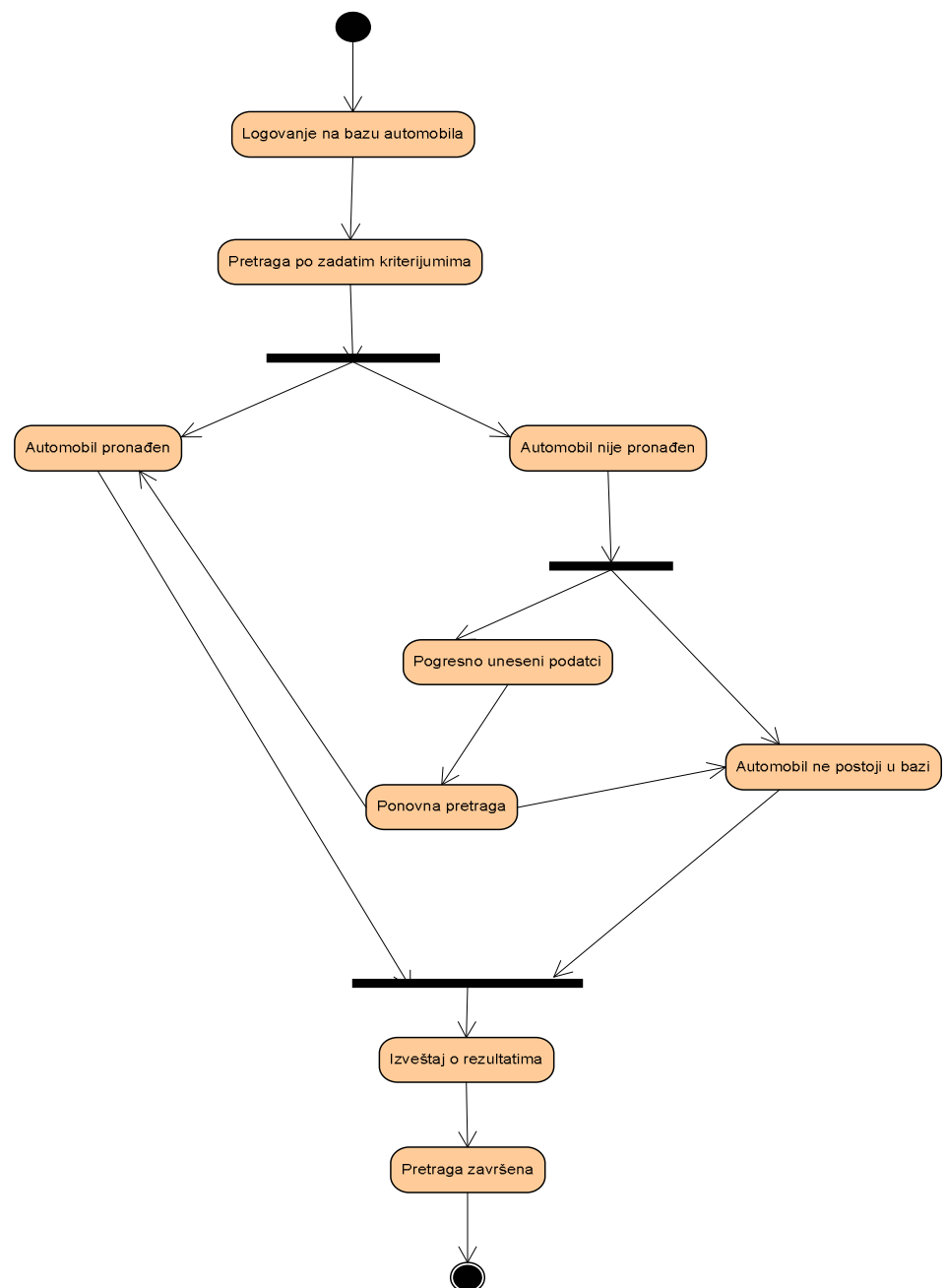
Slika 11. Signali

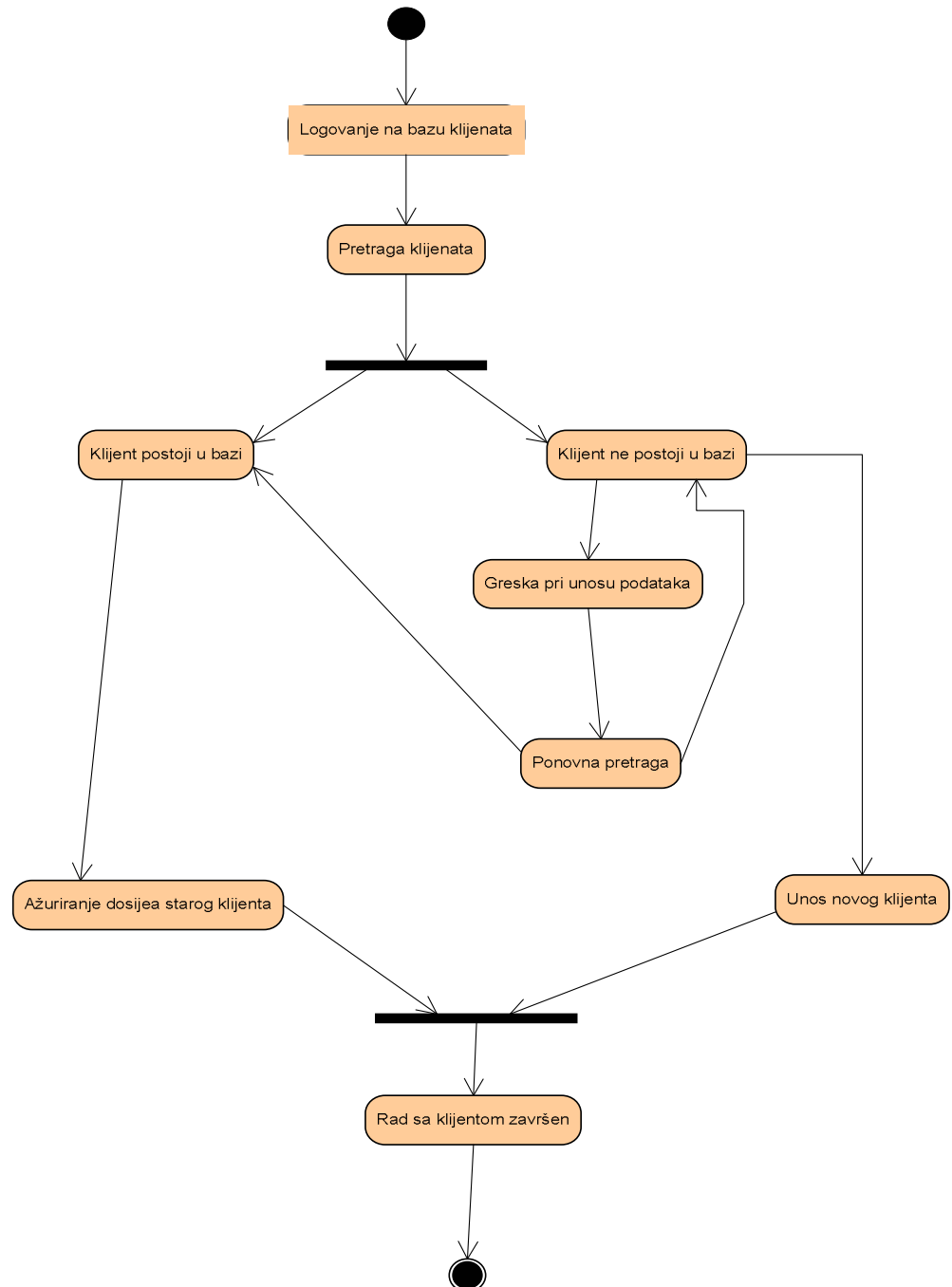
DIJAGRAMI AKTIVNOSTI IS RENT A CAR

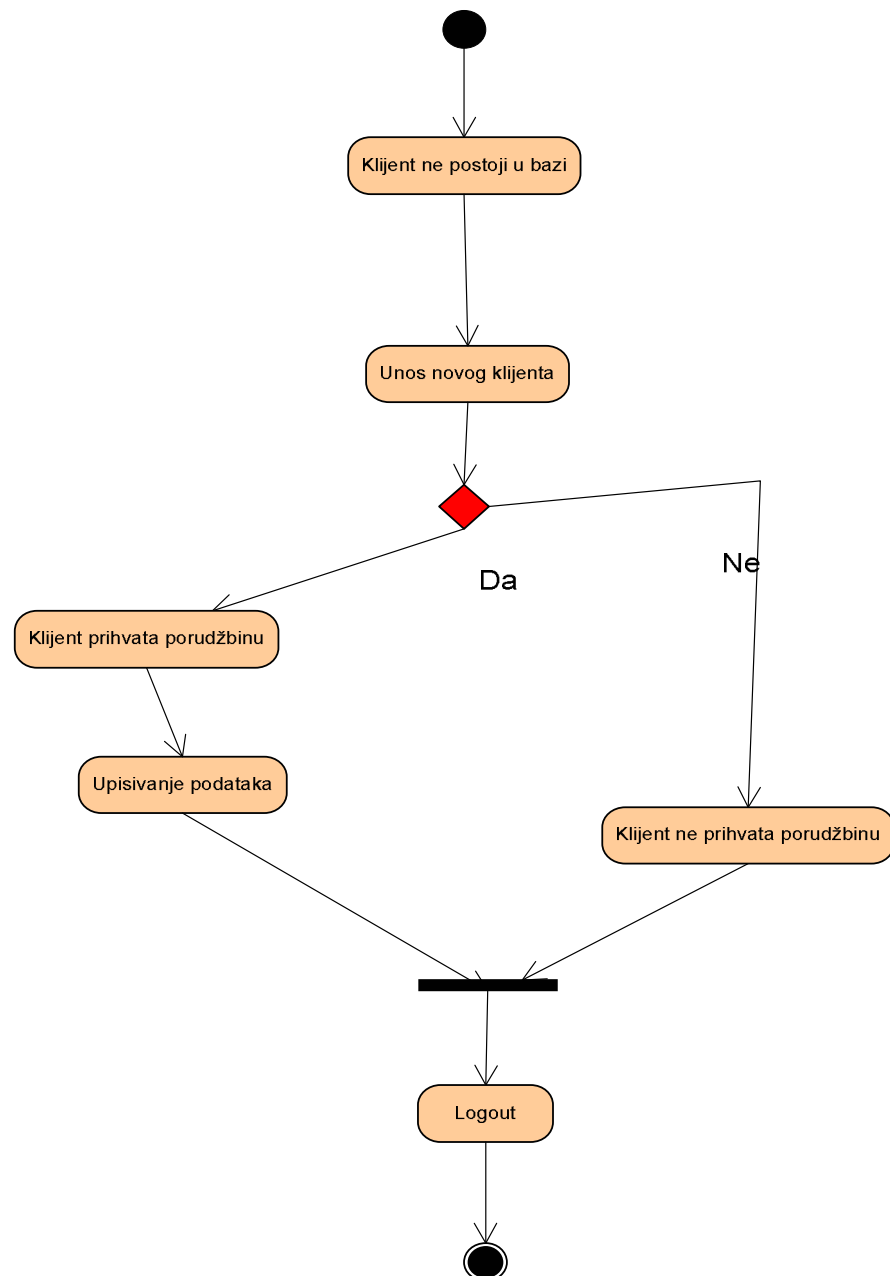
Dijagram Aktivnosti 1. Porudžbina

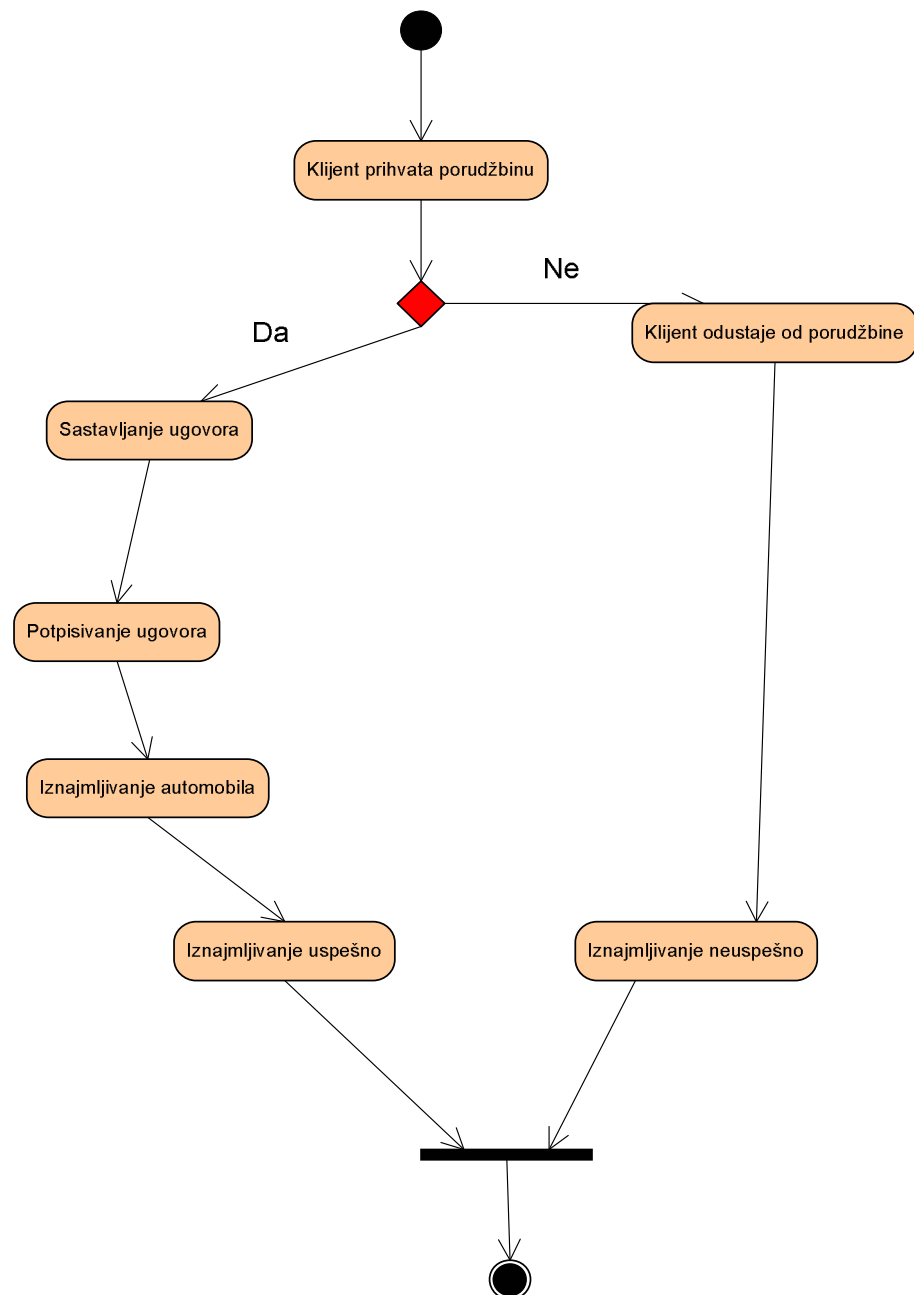


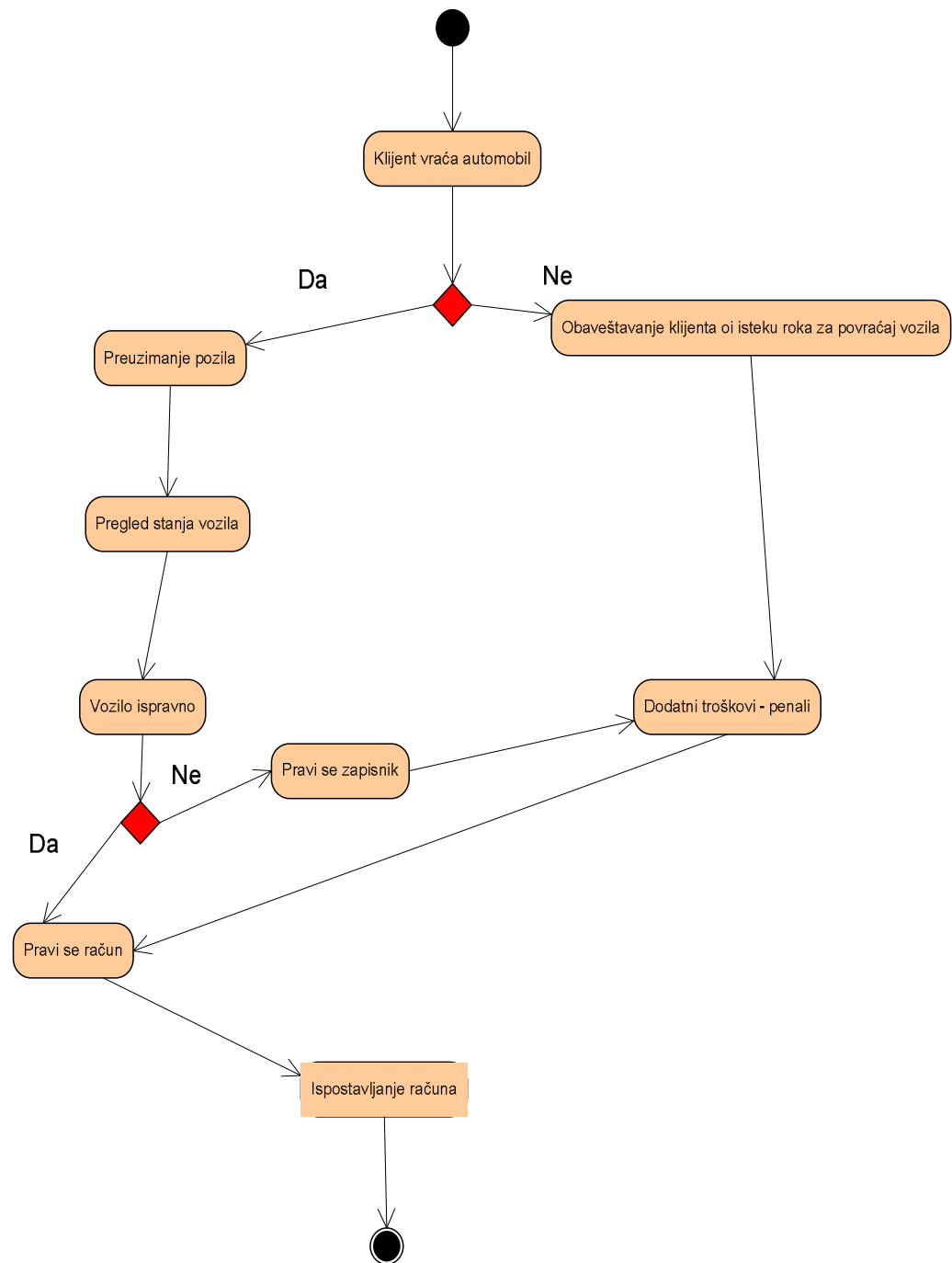
Slika 12. Dijagram aktivnosti - Porudžbina

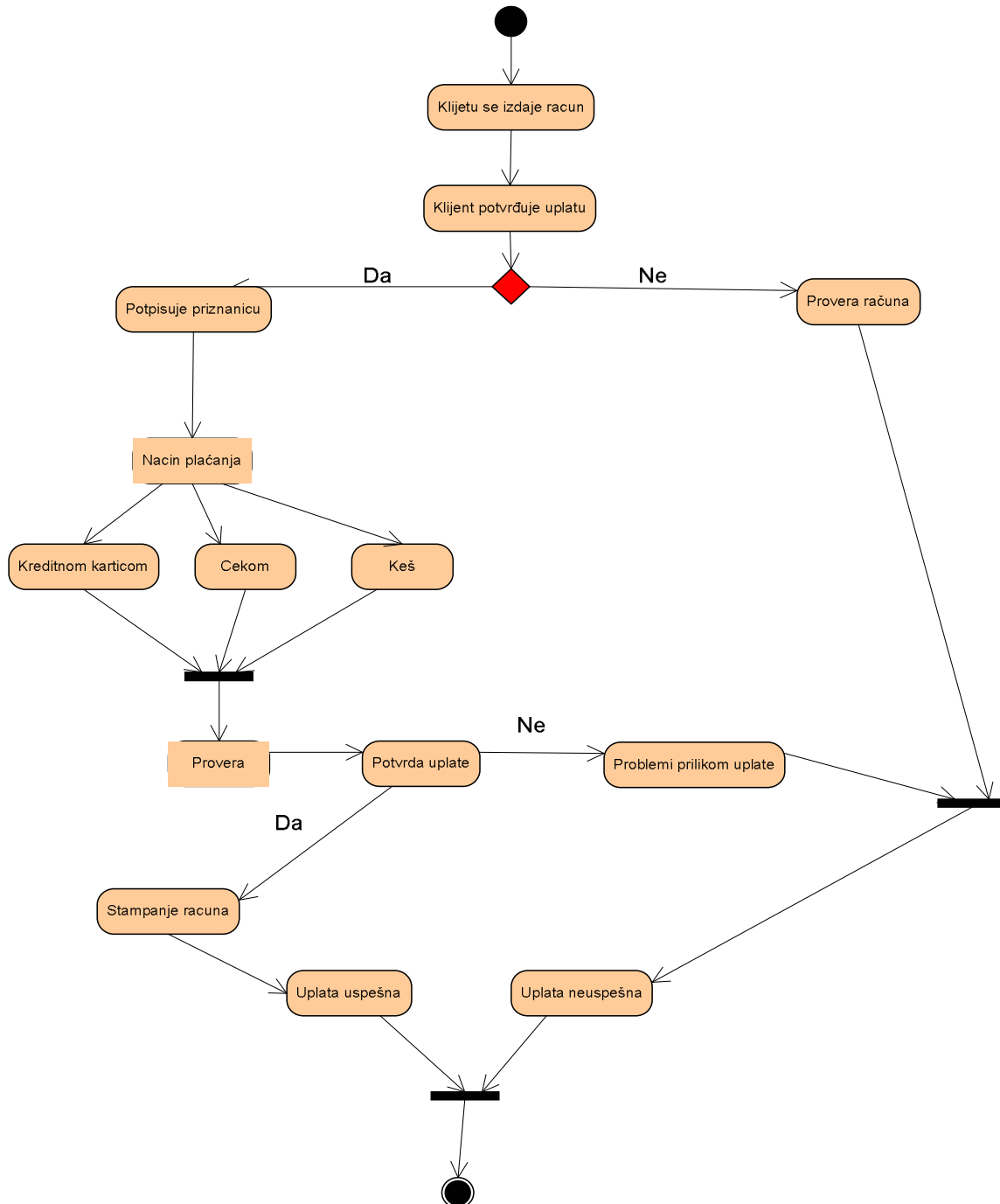
Dijagram Aktivnosti 2. Pregled automobila**Slika 13. Dijagram aktivnosti - pregled automobila**

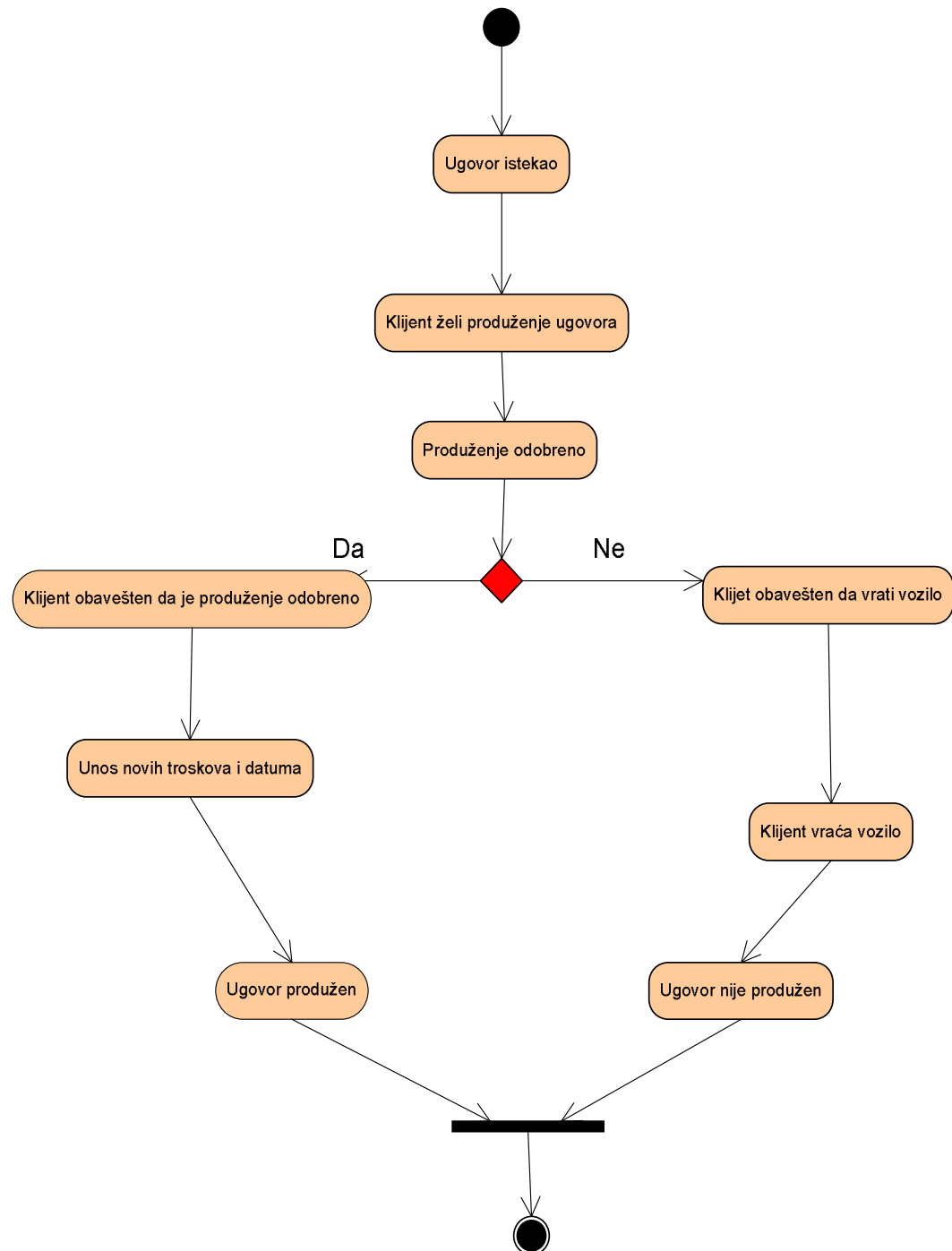
Dijagram Aktivnosti 3. Pregled klijenata**Slika 14. Dijagram aktivnosti - Pregled klijenata**

Dijagram Aktivnosti 4. Unos novog klijenta**Slika 15. Dijagram aktivnosti - Unos novog klijenta**

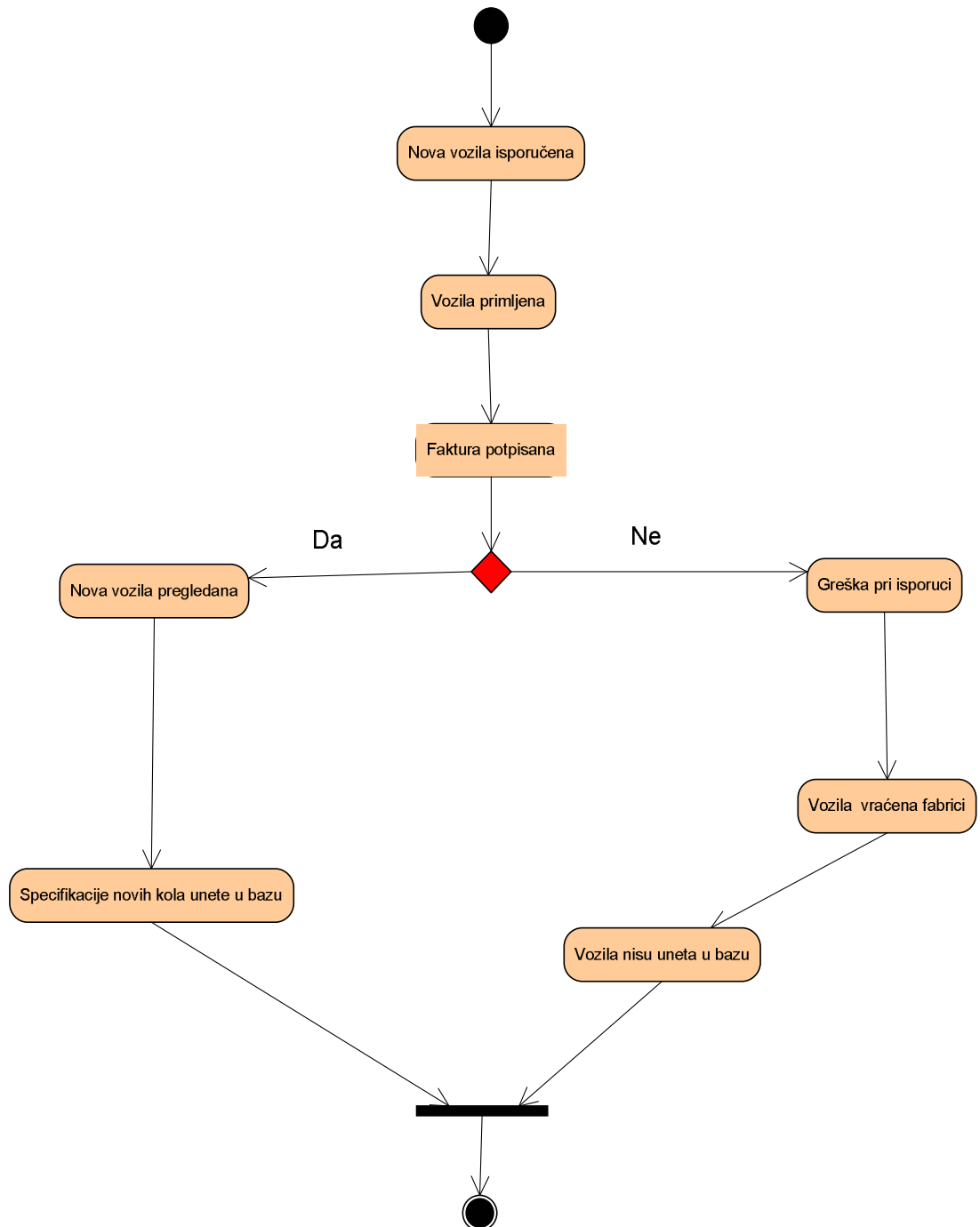
Dijagram Aktivnosti 5. Iznajmljivanje**Slika 16. Dijagram aktivnosti - Iznajmljivane**

Dijagram Aktivnosti 6. Vraćanje automobila**Slika 17. Dijagram aktivnosti - Vraćanje automobila**

Dijagram Aktivnosti 7. Plaćanje**Slika 18. Dijagram aktivnosti - Plaćanje**

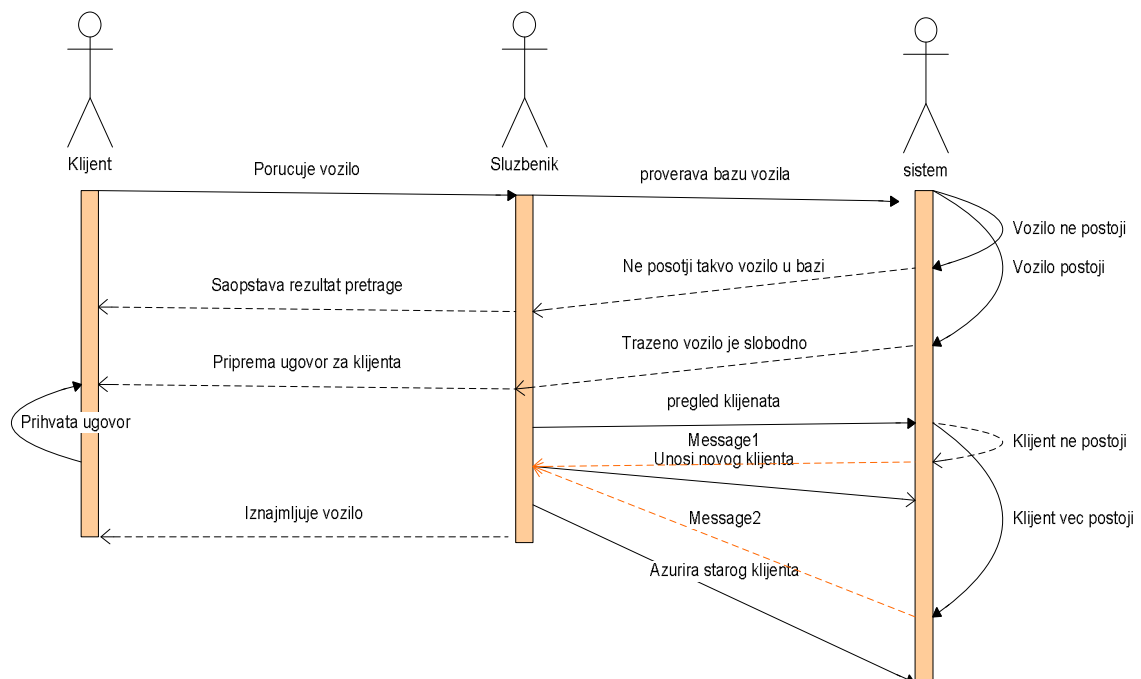
Dijagram Aktivnosti 8. Produživanje ugovora

Slika 19. Dijagram aktivnosti - Produživanje ugovora

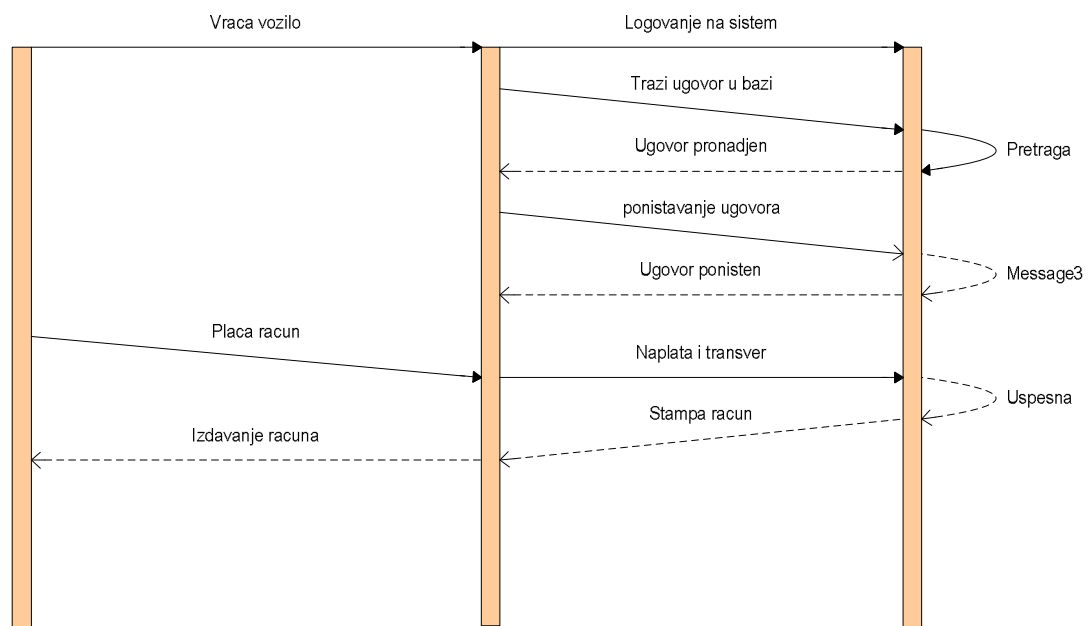
Dijagram Aktivnosti 9. Unos novih vozila

Slika 20. Dijagram aktivnosti - Unos novog vozila

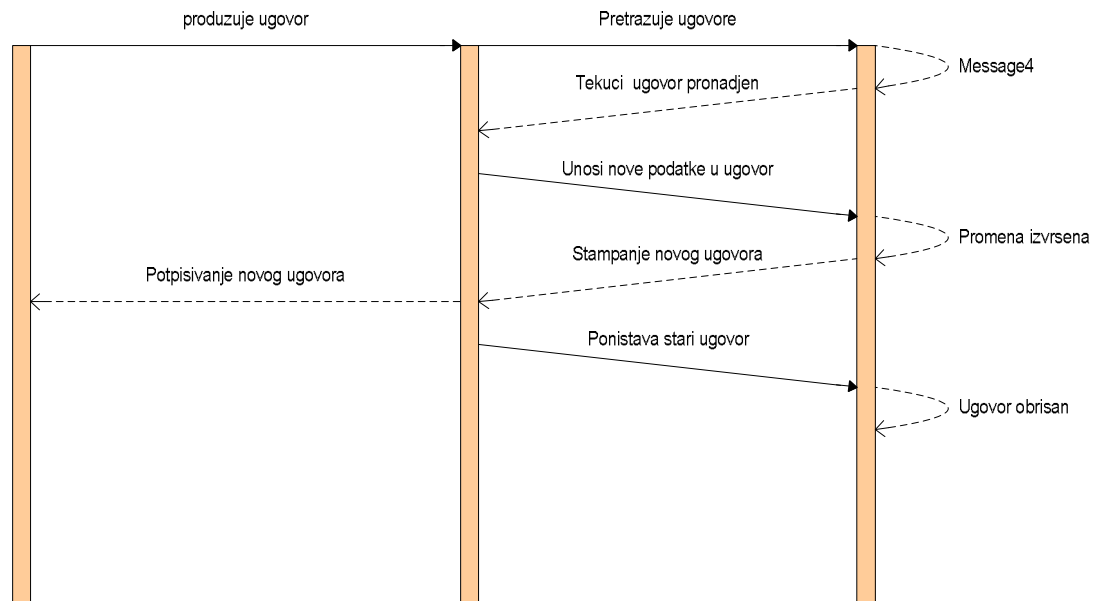
SEKVENCIJALNI DIJAGRAM AKTIVNOSTI



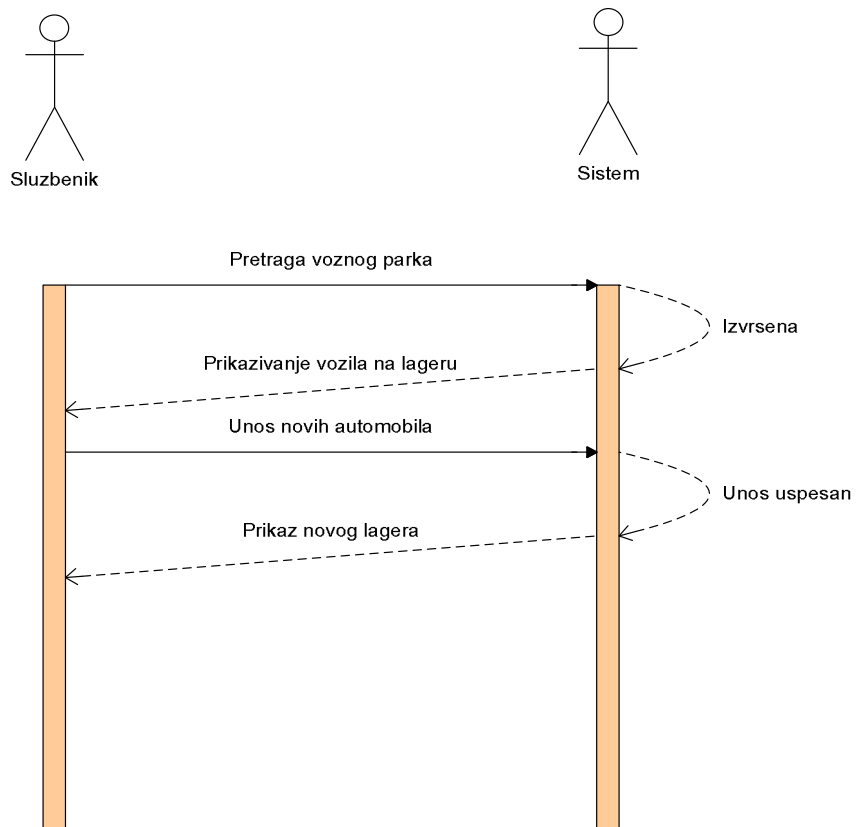
Slika 21. Sekvencijalni dijagram - Iznajmljivanje automobila



Slika 22. Sekvencijalni dijagram - Vraćanje automobila

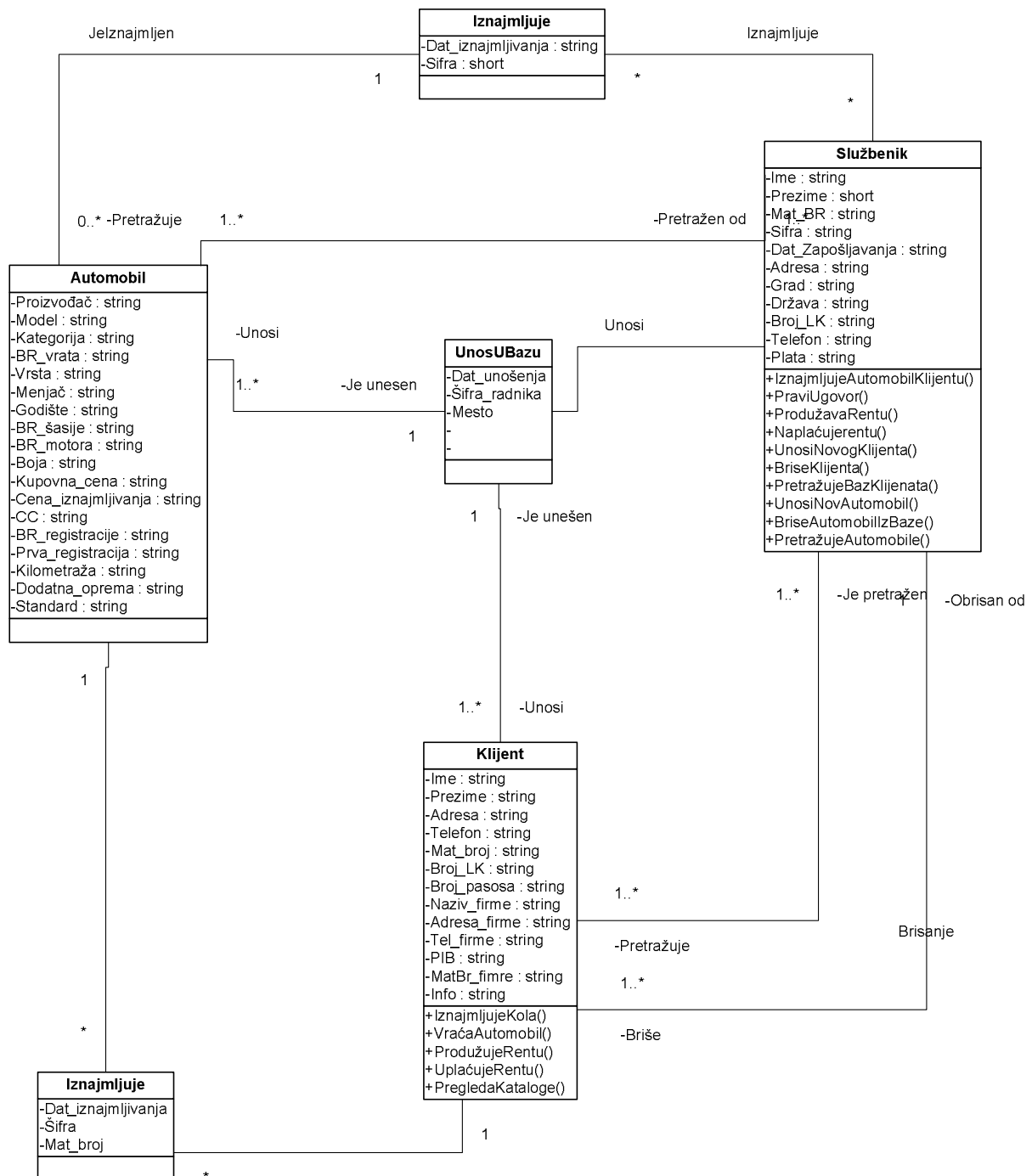


Slika 23. Sekvencijani dijagram - Produživanje ugovora



Slika 24. Sekvencijalni dijagram - Pretraga

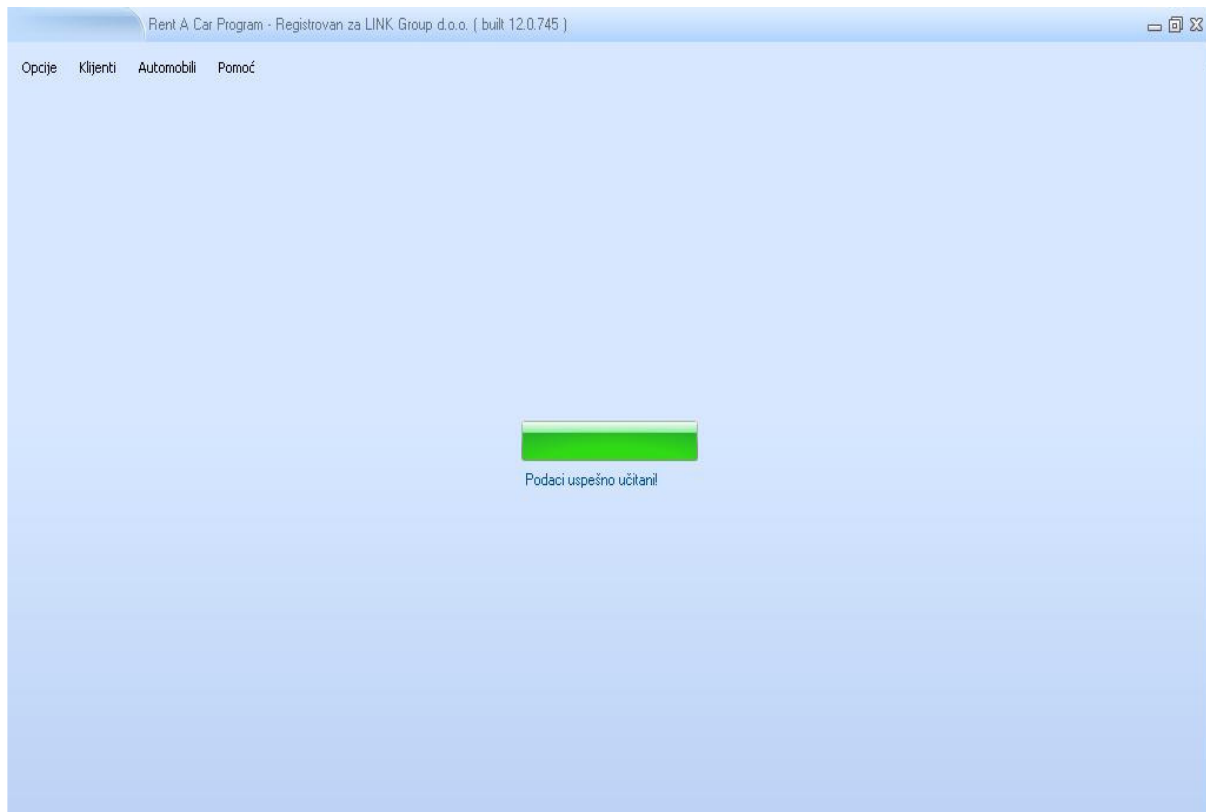
DIJAGRAM KLASA



Slika 25. Dijagram klasa - Rent a Car

GRAFIČKI PRIKAZ SISTEMA

GLAVNA FORMA



Slika 26. Glavna forma

Na ovoj slici vidimo jednu formu, koju smo nazvali glavna forma. Na njoj se nalaze određeni podatci koji se učitavaju. Na vrhu imamo meni koji sadrži polja Opcije, Klijenti, Automobili, Pomoć. Vidimo jednu zelenu liniju (ProgresBar) koja se popunjava, i sve dok se popunjava ne možemo da koristimo meni. Kada se završi popunjavanje ProgresBar-a, meni postaje crne boje i spreman je za korišćenje.

Kako to izgleda u kodu :

```
//metoda za učitavanje zelene linije na formi
private void ucitajPodatke_DoWork(object sender, DoWorkEventArgs e)
{
    //menuStrip1.Metoda za prikazivanje meni-a na formi
    //u našem slučaju prikazuje nam Opcije, Klijenti, Automobili, Pomoć
    //Enabled = false znači da je opcija korišćenja meni-a ukinuta
    menuStrip1.Enabled = false;

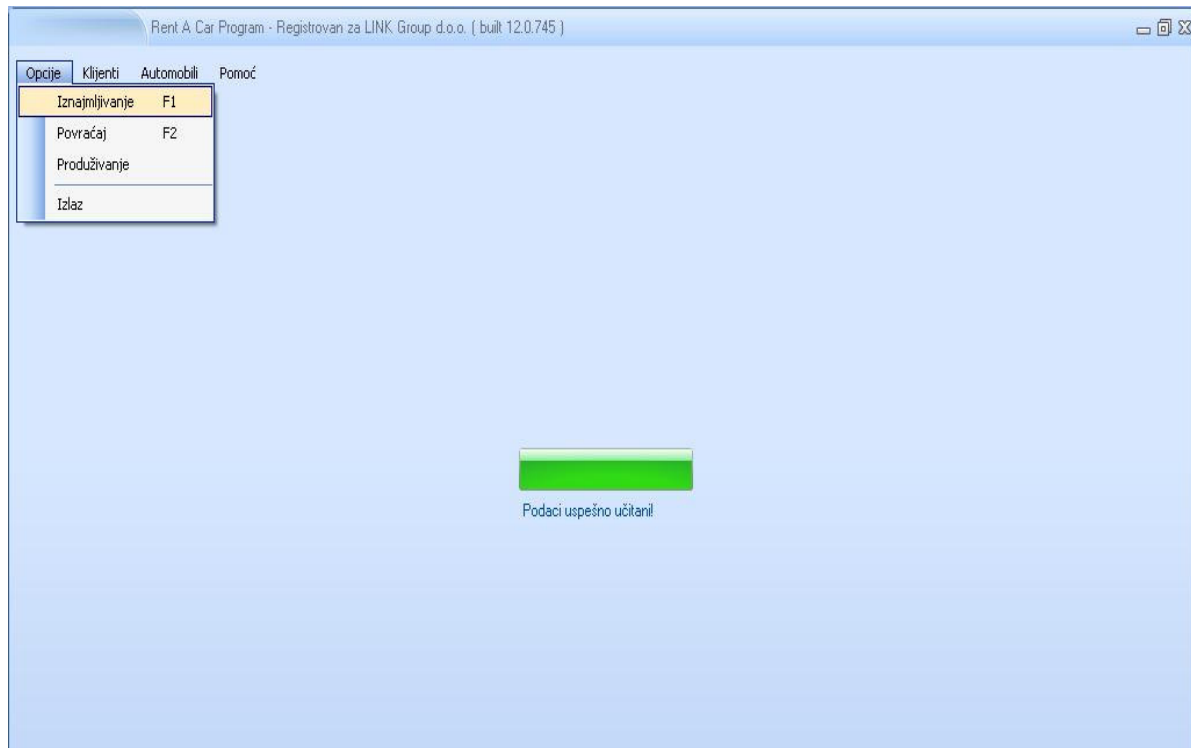
    //dajemo vrednost 10
    radProgressBar1.Value1 = 10;

    //dok got je vrednost manja od 100 sistem se učitava
    while (radProgressBar1.Value1 != 100)
    {
```

```
        System.Threading.Thread.Sleep(300);

        //progres bar se popunjava za 10 vrednost
        radProgressBar1.Value1 += 10;
    }

    //metoda koja dolazi posle učitavanja podataka
    //ova metoda služi da nam omogući korišćenje meni-a
    private void ucitajPodatke_RunWorkerCompleted(object sender,
    RunWorkerCompletedEventArgs e)
    {
        //posle popunjavanja zelene linije ispisuje se poruka da su podatci uspešno učitani
        radLabel1.Text = "Podaci uspešno učitani!";
        //meni je spreman za korišćenje
        menuStrip1.Enabled = true;
    }
```

GLAVNA FORMA – OPCIJE**Slika 27. Pregled opcija**

Forma nam omogućava da iz meni-a izaberemo neke stavke koje smo prethodno uneli. U meni možemo staviti beskonačno aplikacija, ali se obično stavlja onaj broj koji nam koristi. Pošto nada aplikacija radi na iznajmljivanju automobila logično je da u pod Opcije stavimo polja koja odgovaraju toj radnji.

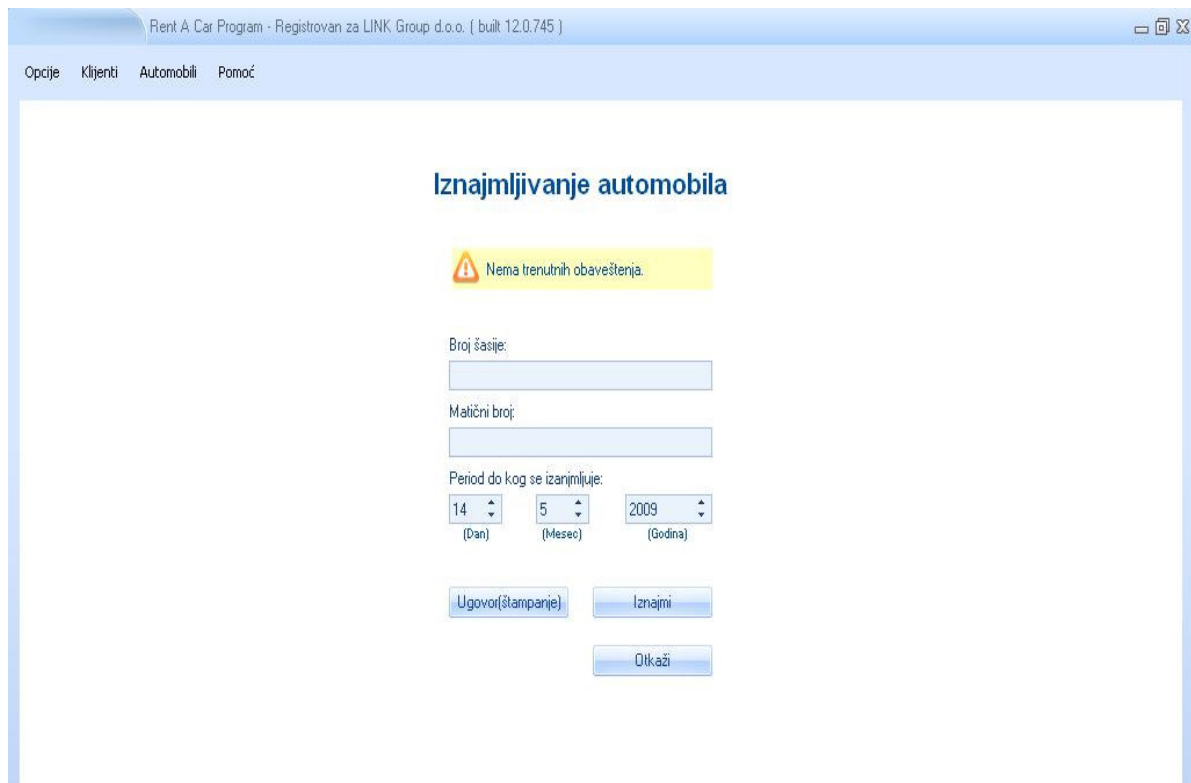
Opcije sadrže:

- Iznajmljivanje
- Povraćaj
- Produživanje
- Izlaz

Pritiskom na opciju Iznajmljivanje otvara nam se nov panel koji nam služi da klijentu iznajmimo automobil i podatke o iznajmljivanju unesemo u bazu podataka.

Evo kako to izgleda u C# kodu :

```
//Metoda koja nam služi da otvorimo panel za iznajmljivanje automobila
//radi samo kada se stisne ova opcija
private void iznajmljivanjeToolStripMenuItem_Click(object sender, EventArgs e)
{
    //Brisemo sve što se nalazilo na glavnom panelu
    panel1.Controls.Clear();
    //Na mesto glavnog panela dodajemo panel za iznajmljivanje
    panel1.Controls.Add(new iznajmiAutomobil());
}
```

FORMA IZNAJMLJIVANJE AUTOMOBILA**Slika 28. Forma za iznajmljivanje automobila**

Dobrodošli na panel za iznajmljivanje automobila.

Kao što vidimo on se sastoji od jednog naslova koji nam govori o kom panelu se radi. Ispod naslova se nalaze dva TextBox-a, u koja upisujemo broj sasijske automobila i maticni broj klijenta. Naravno sve to moramo da označimo nekim datumom. U našem slučaju mi upisujemo datum do kada klijent iznajmljuje automobil.

Ispod datuma nalaze se tri dugmeta preko kojih štampano ugovor, iznajmljujemo automobil i podatke o iznajmljivanju upisujemo u bazu podataka, i otkazujemo iznajmljivanje pri čemu se upisani podatci automatski brišu.

Dugme Otkazi prikazano u C# kodu :

```
//metoda za otkazivanje iznajmljivanje
private void button2_Click(object sender, EventArgs e)
{
    //služi da se sakriju sva polja na panelu
    this.Hide();
}
```

Dugme Iznajmi prikazano u C# kodu :

```
//Metoda za iznajmljivanje automobila i podatke upisuje u bazu
private void button3_Click(object sender, EventArgs e)
{
```

```
//hvatamo izuzetak
try
{

    //SQL konekcija koja nosi tačnu putanju do baze
    SqlConnection conn = new SqlConnection(Properties.Settings.Default.ConnStr);

    //Komanda koja uzima iz baze tačno ono što joj zadamo.
    SqlCommand met = new SqlCommand("Select count(*) from Klijent where Mat_Br
= "" + textBox2.Text + """, conn);
    //Otvaramo konekciju
    conn.Open();

    SqlDataReader reader = met.ExecuteReader();
    reader.Read();
    string metrez = reader.GetInt32(0).ToString();
    conn.Close();

    SqlCommand brs = new SqlCommand("Select count(*) from Automobili where
Br_Sasije = "" + textBox1.Text + """, conn);
    conn.Open();

    SqlDataReader reader3 = brs.ExecuteReader();
    reader3.Read();
    string brsres = reader3.GetInt32(0).ToString();
    //Zatvaramo konekciju
    conn.Close();

    //Ako su polja za unos prazna
    if (textBox1.Text.Replace(" ", "") != "" && textBox2.Text.Replace(" ", "") != "")
    {

        if (metrez != "1")
        {
            warningBox1.Message("Niste uneli validan matican broj");
        }
        else if (brsres != "1")
        {
            warningBox1.Message("Niste uneli validan Broj sasije");
        }
        else
        {

            SqlCommand cmd = new SqlCommand("Insert into Iznajmi (Br_Sasije,
Mat_Br, Datum_Iznajmljivanja, Datum_vracanja) Values("" + textBox1.Text + """," +
textBox2.Text + """," + dtPicker1.MinDate + """," + dtPicker1.GetDate() + """)", conn);
            conn.Open();
            cmd.ExecuteNonQuery();

            warningBox1.Message("Uspesno ste iznajmili auto.");
        }
    }
    else
        warningBox1.Message("Niste uneli sve podatke!");
}
catch (Exception ex) { MessageBox.Show(ex.Message); }
```

}

FORMA POVRAĆAJ AUTOMOBILA**Slika 29. Forma za vraćanje automobila**

Nakon korištenja klijent vraća automobil. U TextBox službenik upisuje broj ugovora koji se napravio kada je klijent iznajmio automobil. Imamo jedan ComboBox u kojem se nalaze načini plaćanja(keš, visa, master card, ček,). Službenik treba samo da pritisne na polje i pojaviće se lista sa svim unešenim načinima plaćanja. Nakon izbora plaćanja upisuje se kilometreža posle rentiranja. Posle unošenja podataka klijent je obavešten o iznosu koji treba da plati. Ako se slaže službenik stiskom na dugme Račun štampa račun za klijenta. Ili pritiskom na dugme otkazi ponistava sve, ako se klijent ne slaže na iznosom rentiranja.

FORMA POVRAĆAJ AUTOMOBILA – DODATNI TROŠKOVI
Slika 30. Forma sa dodatnim troškovima

Nekada je potrebno posle rentiranja automobila uključiti i dodatne troškove koje klijent mora da plati. Tu spadaju razna oštećenja, prevelika kilometraža, itd. U TextBox naziv službenik upisuje ime troška i u TextBox Cena upisuje cenu tog troška. Na dugme dodaj oba upisana polja se opet upisuju u tabelu, nakon čega se cena za uplatu povećava za upisani iznos.

Dodatne troškove smo uključili pritiskom na checkBox "Uključi dodatne troškove".
To u kodu izgleda ovako :

```
private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
    if (radPanel1.Visible == true)
    {
        radPanel1.Visible = false;
        Cena -= CenaTroskova;
        CenaTroskova = 0;
        listView1.Items.Clear();
        label8.Text = String.Format("{0:Ukupno: 0.00 Eura}", Cena);
    }
    else
    {
        radPanel1.Visible = true;
    }
}
```

FORMA PRODUŽIVANJE UGOVORA**Slika 31. Forma za produžavanje ugovora**

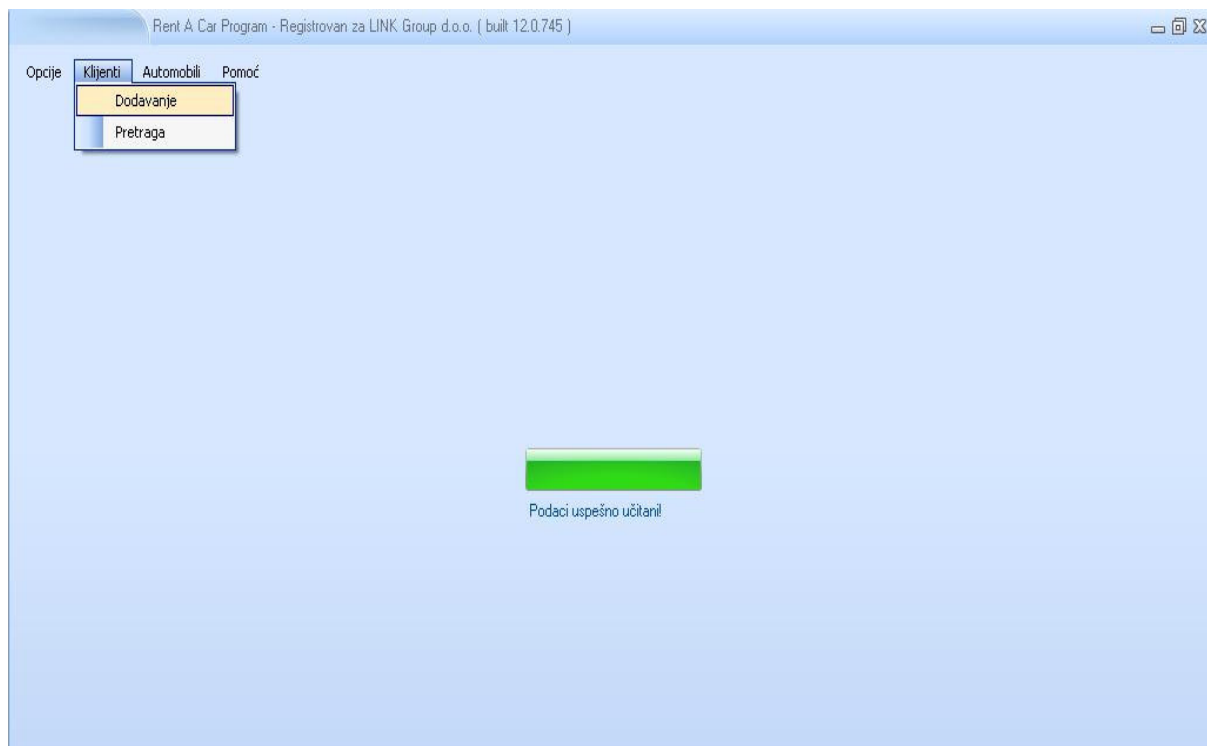
Na ovom panelu samo produžujemo datum za trajanje ugovora. Službenik u polje upisuje broj ugovora nakon cega menja datum. Na dugme produži podatci u bazi se ažuriraju.

U kodu to izgleda ovako :

```
private void button3_Click(object sender, EventArgs e)
{
    try
    {
        SqlConnection conn = new SqlConnection(Properties.Settings.Default.ConnStr);
        SqlCommand cmd = new SqlCommand("Update Iznajmi Set Datum_Vracanja = " +
dtPicker1.GetDate() + " where [Broj Ugovora] = " + comboBox2.SelectedItem + "", conn);

        conn.Open();
        cmd.ExecuteNonQuery();

        warningBox1.Message("Uspesno ste produzili ugovor");
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
```

GLAVNA FORMA – KLIJENTI**Slika 32. Pregled klijenata**

Pritiskom na polje Klijenti/Dodavanje otvara se panel za Dodavanje novih klijenata u bazu podataka.

U kodu to izgleda ovako :

```
private void dodajToolStripMenuItem_Click(object sender, EventArgs e)
{
    panel1.Controls.Clear();
    panel1.Controls.Add(new klijentDodaj());
}
```

FORMA DODAVANJE KLIJENATA
Slika 33. Forma za dodavanje klijenata

Ovaj panel nam služi da u bazu podataka unesemo nove klijente. Na njemu se nalaze brojni TextBox-ovi koji sam služe da upisujemo podatke o klijentu. Oni moraju da odgovaraju poljima u bazi. Posle popunjavanja polja na dugme Dodaj svi podatci će biti upisani u bazu. Na dugme otkazi sve će da se obriše.

Dugme Dodaj U C# kodu :

```
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        if (textBox1.Text.Replace(" ", "") != "" && textBox2.Text.Replace(" ", "") != "" &&
            textBox3.Text.Replace(" ", "") != "" && textBox4.Text.Replace(" ", "") != "" &&
            (textBox5.Text.Replace(" ", "") != "" || textBox6.Text.Replace(" ", "") != ""))
        {
            using (SqlConnection conn = new
            SqlConnection(Properties.Settings.Default.ConnStr))
            {
                SqlCommand cmd = new SqlCommand("Insert Into Klijent(Ime, Prezime,
                Adresa, Telefon, Mat_Br, Broj_LK, Br_Pasosa, Naziv_Firme, Adresa_Firme, Tel_Firme, PIB,
                MatBR_Firme, Info) values ('" + textBox1.Text + "','" + textBox2.Text + "','" + textBox3.Text
                + "','" + textBox4.Text + "','" + textBox5.Text + "','" + textBox6.Text + "','" + textBox7.Text
```

```
+ "','" + textBox8.Text + "','" + textBox9.Text + "','" + textBox10.Text + "','" +  
textBox11.Text + "','" + textBox12.Text + "','" + radTextBox1.Text + "')", conn);  
    conn.Open();  
    cmd.ExecuteNonQuery();  
}  
    MessageBox.Show("Test");  
}  
    else  
        MessageBox.Show("Niste uneli sva obvezna polja!");  
} catch (Exception){}
```

Dugme Otkazi u C# kodu :

```
private void button2_Click(object sender, EventArgs e)  
{  
    this.Hide();  
}
```

FORMA PRETRAGA KLIJENATA

Pretraga klijenata

Nema trenutnih obaveštenja.

Ime: Traži

Prezime: Otkazi

Rezultat:

Trenutno nema rezultata

Slika 34. Forma za pretragu klijenata

Panel nam služi da pretražujemo podatke o klijentima. Službenik popunjava polja Ime i Prezime i pritiskom na dugme Traži u panelu Rezultat se pojavljuju podatci o klijentu. Ako je pretraga neuspela pojavljuje se poruka da nema rezultata. Dugme otkazi nam služi da prekinemo pretragu.

Dugme Traži u C# kodu :

```
private void button1_Click(object sender, EventArgs e)
{
    radPanel1.Visible = false;
    listView1.Items.Clear();
    listView1.Visible = true;
    pictureBox1.Visible = true;

    try
    {
        using (SqlConnection conn = new
SqlConnection(Properties.Settings.Default.ConnStr))
        {
            conn.Open();
```

```
SqlCommand cmd = new SqlCommand("SELECT Mat_Br, Ime, Prezime FROM  
Klijent WHERE Ime LIKE '%" + textBox1.Text + "%' AND Prezime LIKE '%" + textBox2.Text +  
"%'", conn);
```

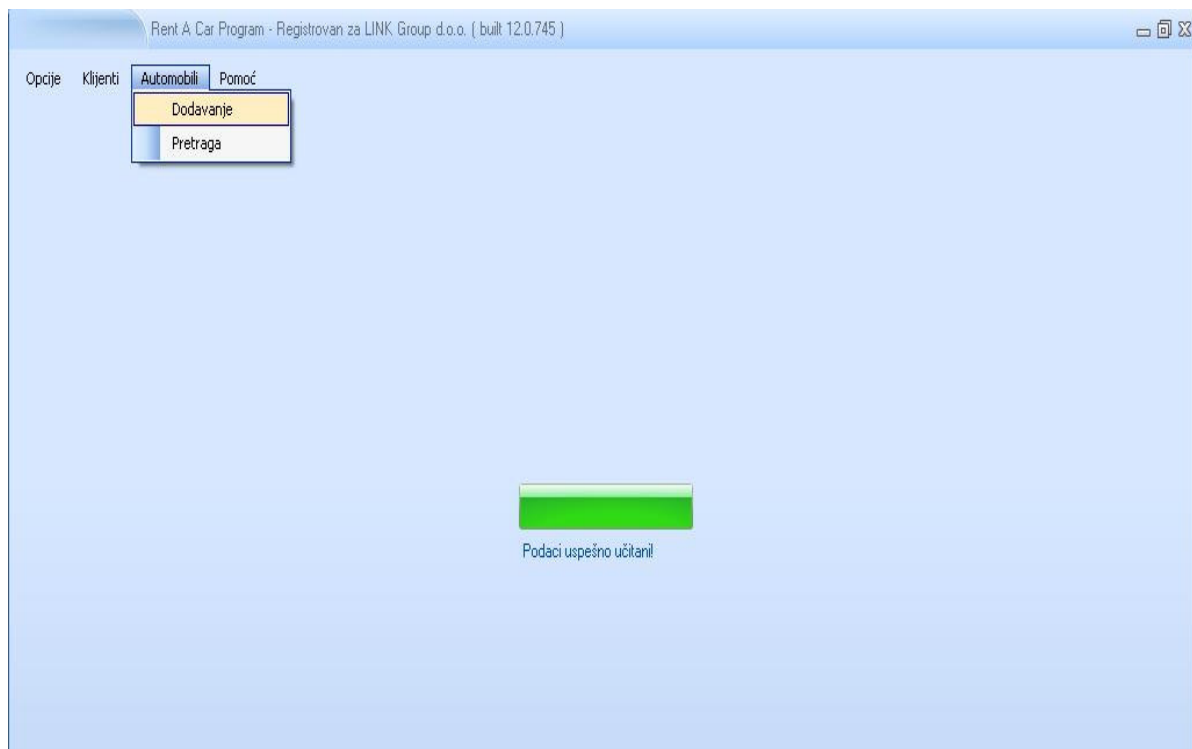
```
SqlDataReader dr = cmd.ExecuteReader();
```

```
while (dr.Read())  
{  
    ListViewItem lvi = new ListViewItem();  
    lvi.Text = dr.GetString(0);  
    lvi.SubItems.Add(dr.GetString(1));  
    lvi.SubItems.Add(dr.GetString(2));  
    listView1.Items.Add(lvi);  
}
```

```
    }  
}  
catch (Exception) { }
```

```
if (listView1.Items.Count == 0)  
{  
    radPanel1.Visible = true;  
    listView1.Visible = false;  
    pictureBox1.Visible = false;  
}
```

```
}
```

GLAVNA FORMA – AUTOMOBILI**Slika 35. Opcija automobili**

Pritiskom na dugme Klijenti/Dodavanje otvaramo panel u+z adodavanje klijenata u bazu podataka.

U C# kodu to izgleda ovako :

```
private void dodavanjeNovogToolStripMenuItem_Click(object sender, EventArgs e)
{
    panel1.Controls.Clear();
    panel1.Controls.Add(new automobilDodaj());
}
```

FORMA DODAVANJE AUTOMOBILA

Rent A Car Program - Registrovan za LINK Group d.o.o. (built 12.0.745)

Opcije Klijenti Automobili Pomoć

Dodavanje automobila

Nema trenutnih obaveštenja.

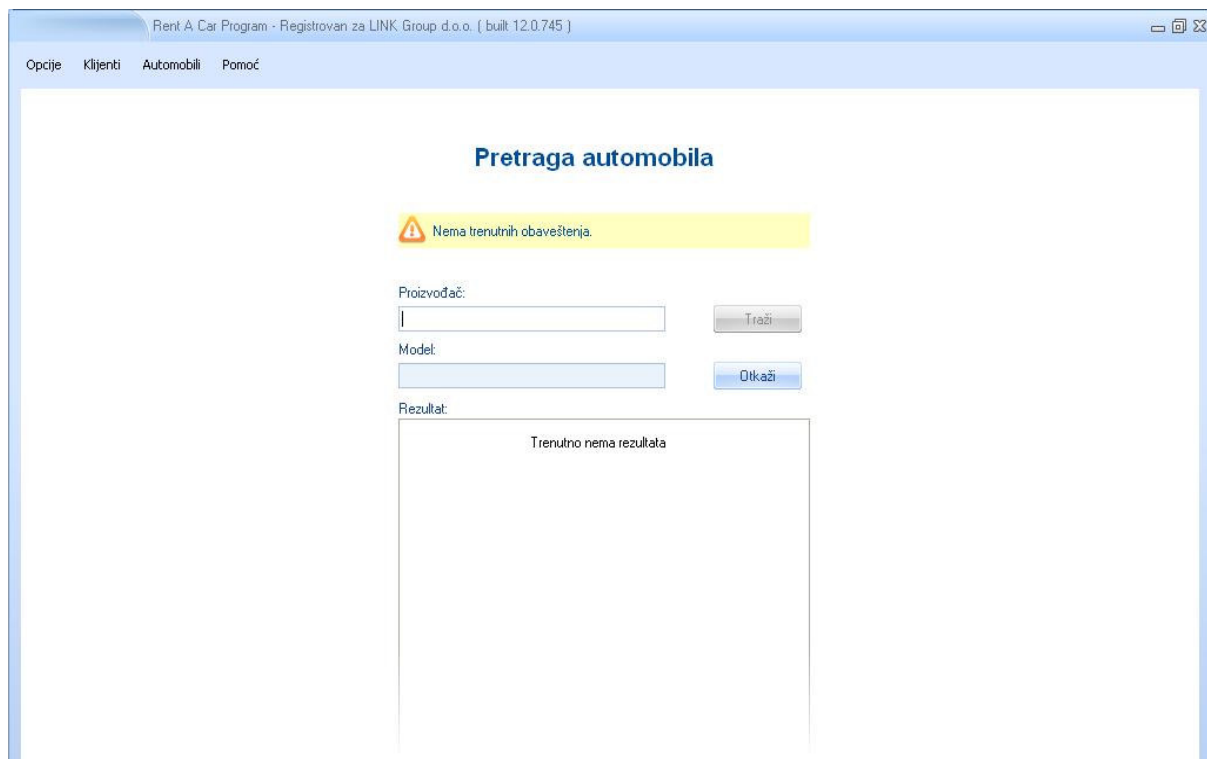
Proizvođač:	Kupovna cena (Euro):
Model:	Cena iznajmljivanja(Euro):
Kategorija:	Kubikaža:
Broj vrata:	Broj registracije:
Vrsta goriva:	Standard
Menjač:	☒ Euro 3 ☐ Euro 4
Godište:	Prva registracija:
Broj šasije:	14/5/2009
Broj motora:	Kilometraža:
Boja:	Dodatna oprema:

Dodaj Otkazi

Slika 36. Forma za dodavanje automobila

Kao i kod klijenata ovaj panel nam služi da podatke o novom automobilu unesemo u bazu podataka. Panel se sastoji od mnoštvo polja koja službenik popunjava i pritiskom na dugme Dodaj unešene podatke upisuje u bazu. Dugme otkazi služi da se sve poništi.

Dugme Dodaj u C# kodu izgleda isto kao kod klijenata, razlika je u imenima i broju polja. Svako polje mora da odgovara onom u bazi podataka.

FORMA PRETRAGA AUTOMOBILA

Rent A Car Program - Registrovan za LINK Group d.o.o. (built 12.0.745)

Opcije Klijenti Automobili Pomoć

Pretraga automobila

 Nema trenutnih obaveštenja.

Proizvođač:

Model:

Rezultat:

Trenutno nema rezultata

Slika 37. Forma za pretragu automobila

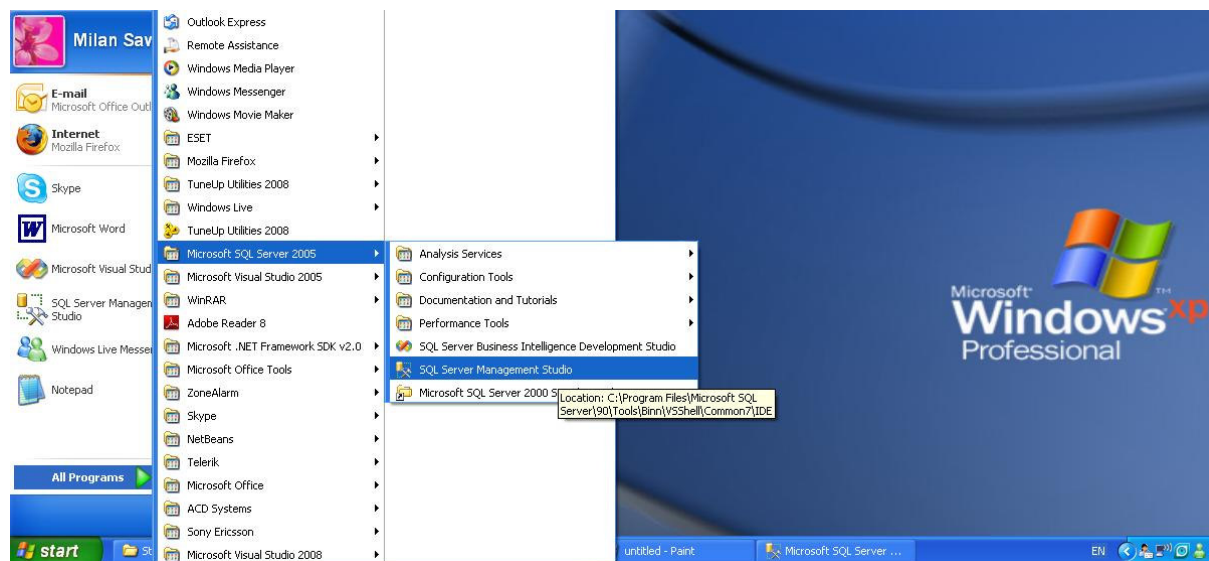
Isto kao kod klijenata službenik upisuje podatke u polja i na osnovu njih traži u bazi podataka.

POVEZIVANJE NA BAZU PODATAKA

RAD SA BAZOM RENT A CAR

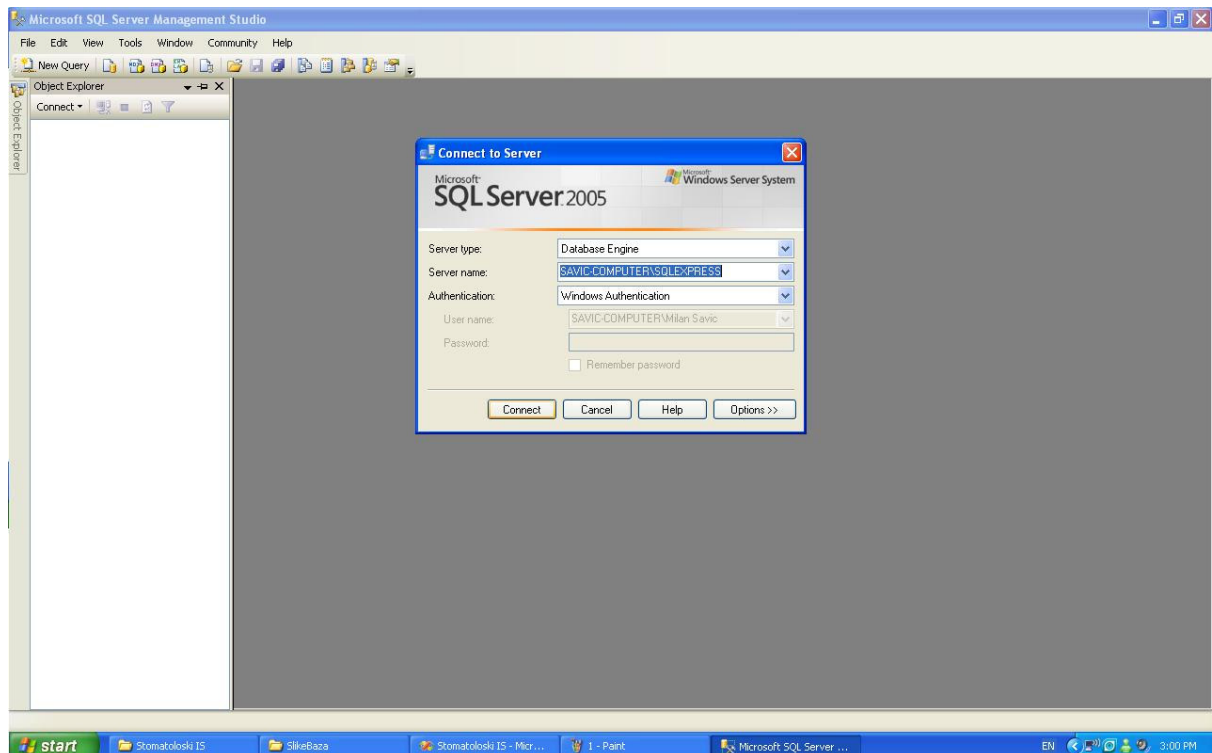
Bazu podataka sam radio u Sql Management Studio 2003. Uz ovaj program se dobija i Sql Server 2003 koji je potreban ba bi napravljena baza radila. Posle projetovanja aplikacije i recnika podataka pristupamo samoj izradi baze.

PROCES IZRADE BAZE PODATAKA



Slika 38. Startovanje Sql Management aplikacije

Posle instalacije SQL Servera 2005, vaš program možete naći na Start/Programs/Microsoft SQL Server 2005/ SQL Server Management Studio. Pokretanjem ove aplikacije spremni smo da projektujemo i pravimo bazu podataka.

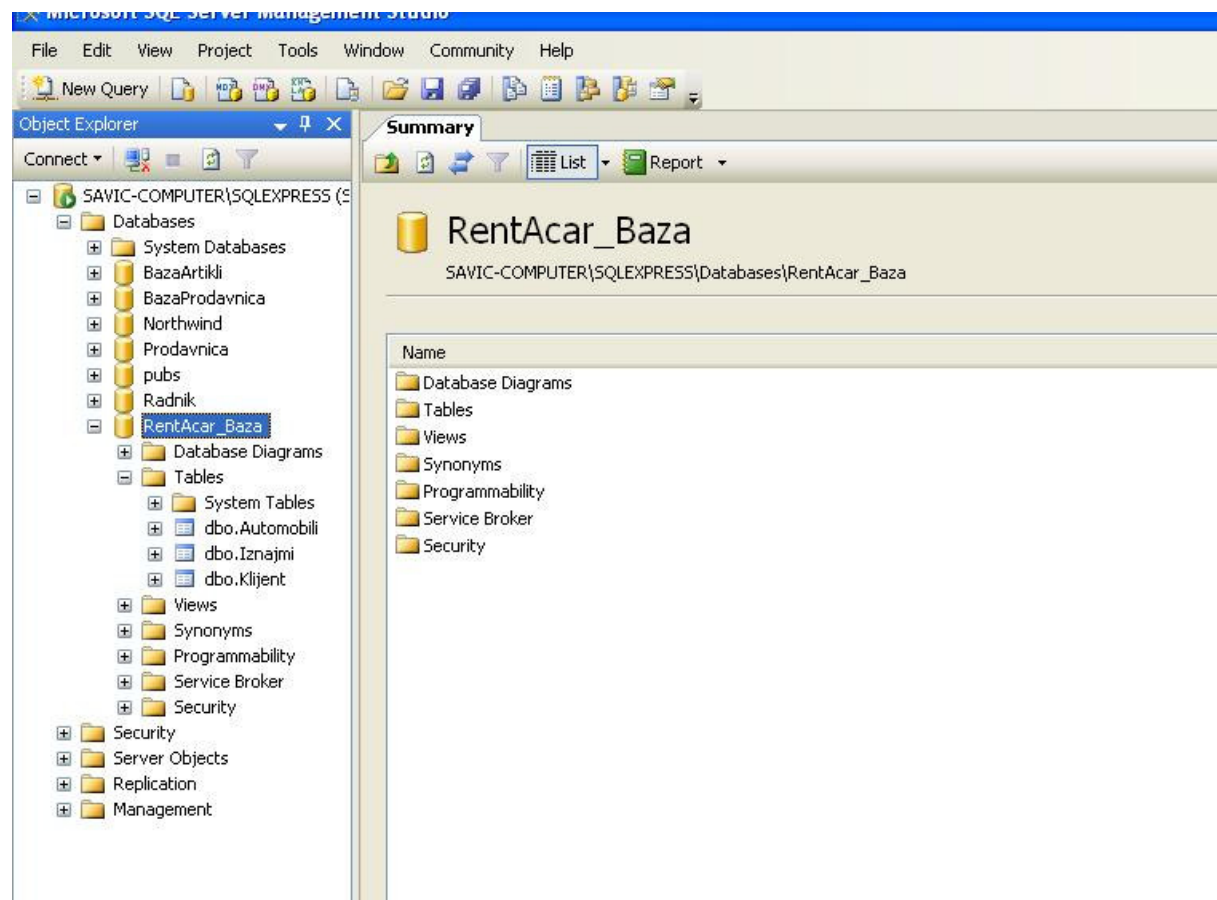


Slika 39. Logovanje na SQL Server

Nakon pokretanja aplikacije pojaviće vam se prozor kao na slici. Ispratite korake koje vam budem napisao da bi se logovali na server.

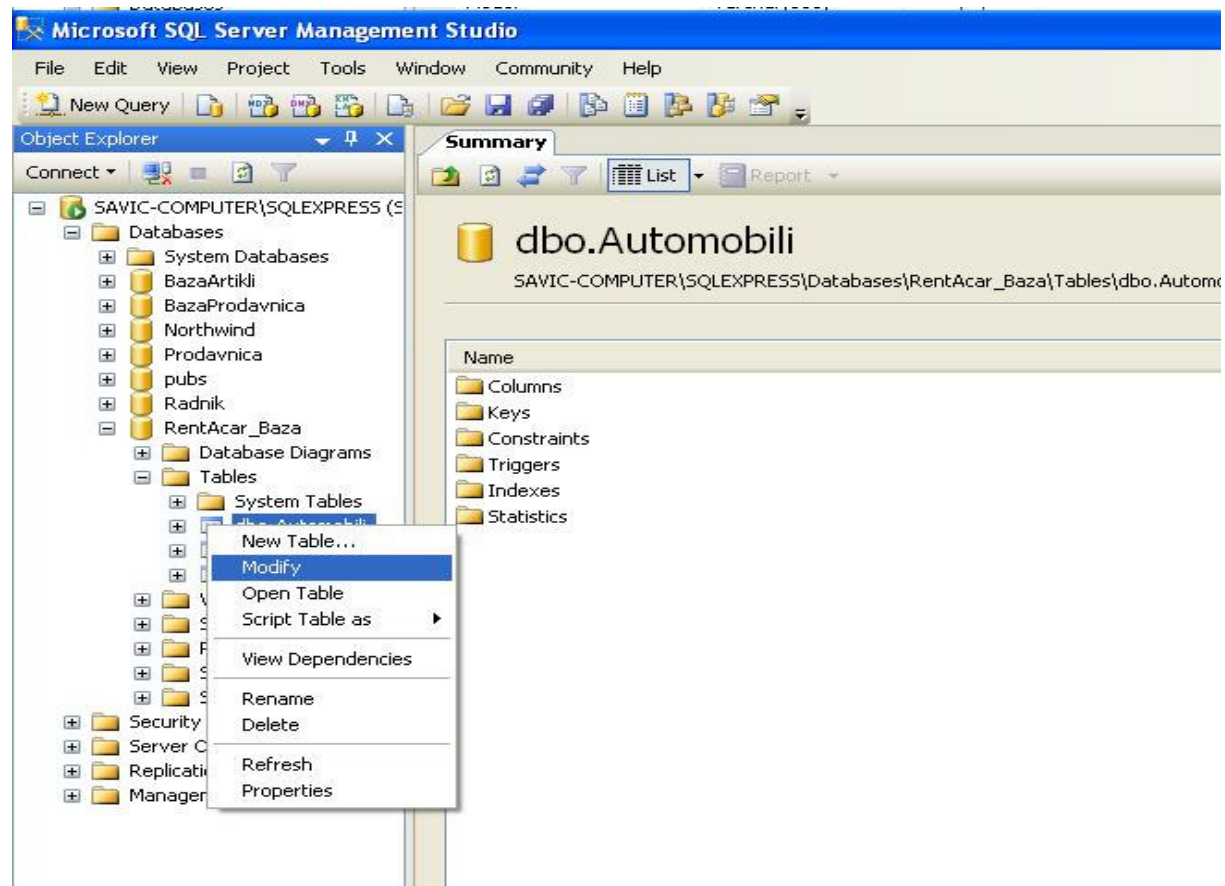
- Server type – Database Engine
- Server name – vi pišete ime vašeg računara\\sql express
- Authentication – Windows authentication

Posle toga pritisakmo dugme Connect.....



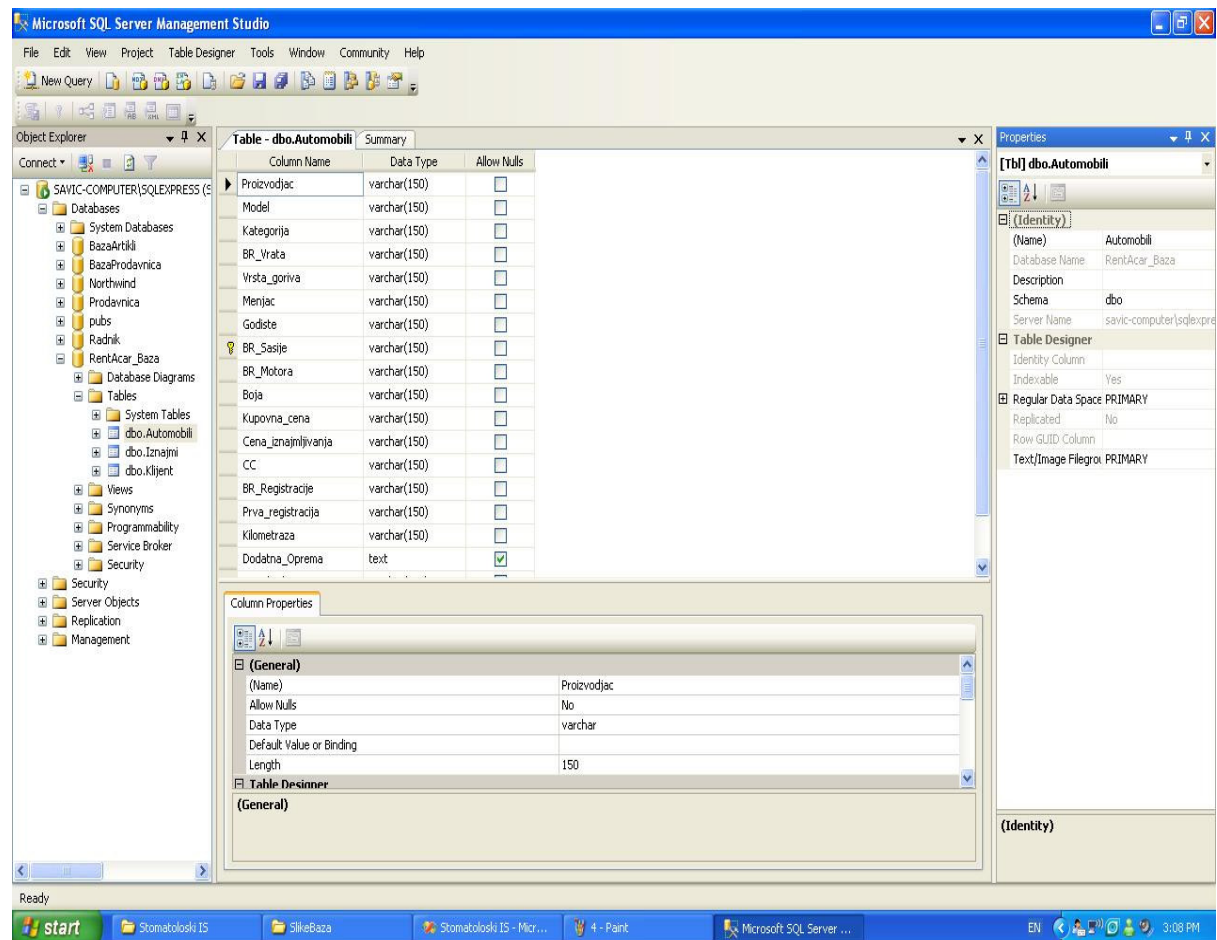
Slika 40. Pregled svih baza

Posle konektovanja na server dobitete ovakvu sliku. Sa leve strane imate polje Object Explorer, i tu su vam podatci o vašim bazama koje pravite ili koje vam je sistem dodelio. Pritiskom na krstic Databases, otvara se na dole još polja. Videćete imena nekih baza. Ja sam već napravio bazu podataka RentACar_Baza. Njih naziv mi piše. Pritiskom na krstić moje baze otvaraju mi se polja koja se tiču samo baze. Na tables gledamo tabele koja se nalaze u mojoj bazi podataka. Te tabele sadrže polja iz Rečnika podataka, koja ćemo mi da koristimo za našu aplikaciju.



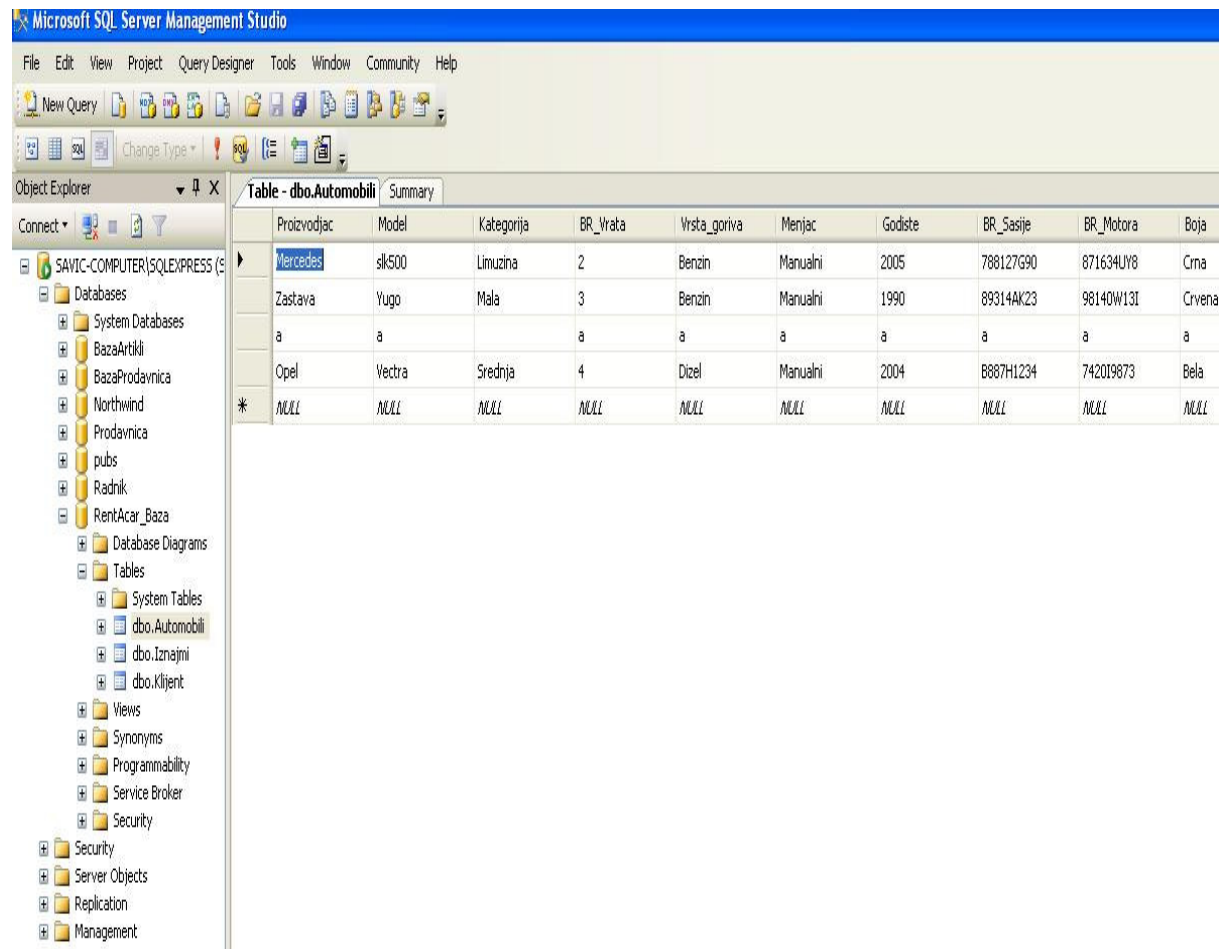
Slika 41. Opcija Modify

U našoj bazi podataka, izaberemo jednu tabelu i desnim klikom označimo na Modify. To nam otvara tabelu sa svim poljima koje smo napravili, i sada ih možemo modifikovati, brisati, dodavati nova polja, menjati ograničenja, menjati tip polja. Na sledećoj slici imate detaljan prikaz šta se dešava kada se pritisne Modify...



Slika 42. Prikaz polja tabelle Automobili za izmenu

Vidite sami da nam je sistem ponudio sva polja koja želimo da modifikujemo. U zavisnost od potražnje mi ćemo da menjamo polja.



The screenshot shows the Microsoft SQL Server Management Studio interface. On the left, the Object Explorer displays the database structure for 'SAVIC-COMPUTER\SQLEXPRESS'. The 'RentAcar_Baza' database is expanded, showing 'Tables' which includes 'dbo.Automobili'. The main pane displays the 'Table - dbo.Automobili' with a 'Summary' tab. The table contains the following data:

Proizvodjac	Model	Kategorija	BR_Yrata	Vrsta_goriva	Menjac	Godiste	BR_Sasije	BR_Motora	Boja
Mercedes	slk500	Limuzina	2	Benzin	Manualni	2005	786127G90	871634UY8	Crna
Zastava	Yugo	Mala	3	Benzin	Manualni	1990	89314AK23	98140W13I	Crvena
a	a		a	a	a	a	a	a	a
Opel	Vectra	Srednja	4	Dizel	Manualni	2004	B887H1234	7420I9873	Bela
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

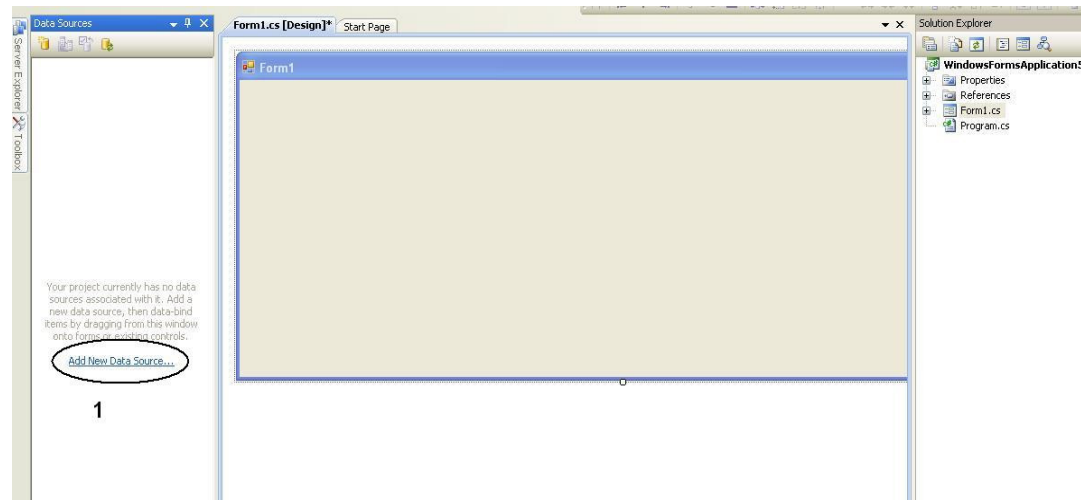
Slika 43. Prikaz polja za popunjavanje

Kada smo na tabelu Automobili sa desnim klikom pritisnuli opciju Open, dobili smo ovaj prozor. Tu su nam ponudjena polja koja sadrže neke vrednosti. Bazu možemo popunjavati ovako, direktno iz programa. Dok je možemo popunjavati i iz aplikacije.

POVEZIVANJE NOVE APLIKACIJE I BAZE PODATAKA

Kada smo u Visual Studio 2003 napravili nov windows projekat i sačuvali ga, moramo da povežemo projekat sa odgovarajućom bazom podataka. Jer naša aplikacija i radi sa bazom podataka. Znaci mi sada samo povezujemo aplikaciju i bazu, da bi dobili **"ConnectionString"**, Koji ćemo kasnije da koristimo u kodu.

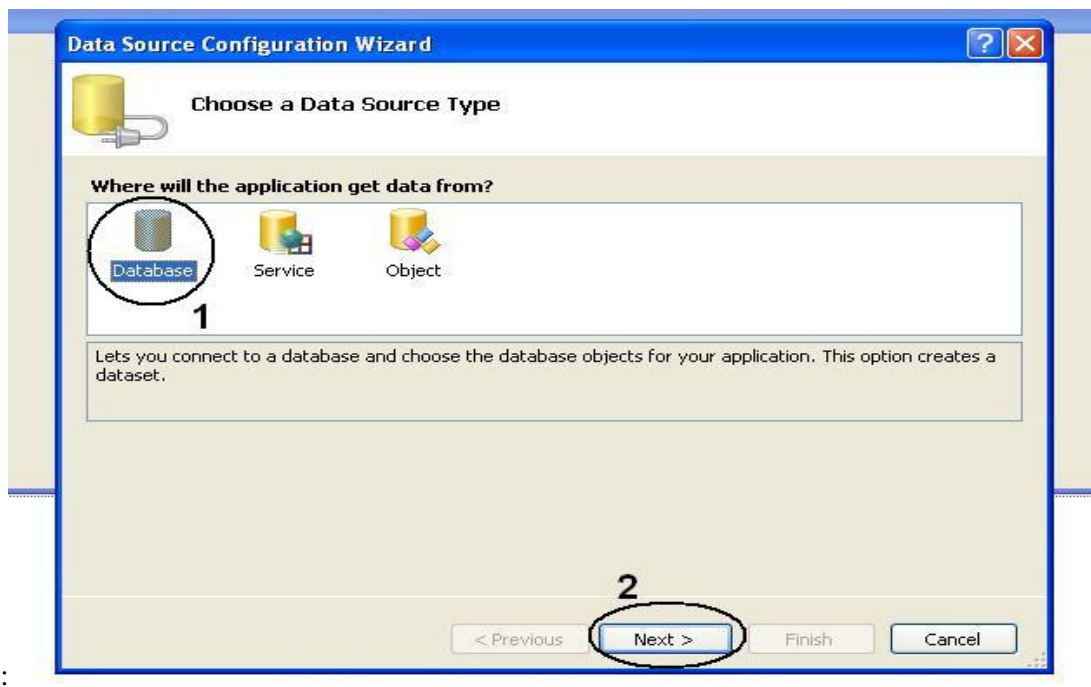
Korak 1 :



Slika 44. Izbor novog Data Source-a

Ovdje imamo prikaz ekrana Visual Studio 2003 sa opcijom da se povežemo na bazu. Ta opcija se zove Data Sources. Ako slučajno ne vidite ovu opciju potražite je gore u listi na Data/Show Data Sources. Kada je postavimo na ekran izaberite opciju Add new Data Sources..

Korak 2 :



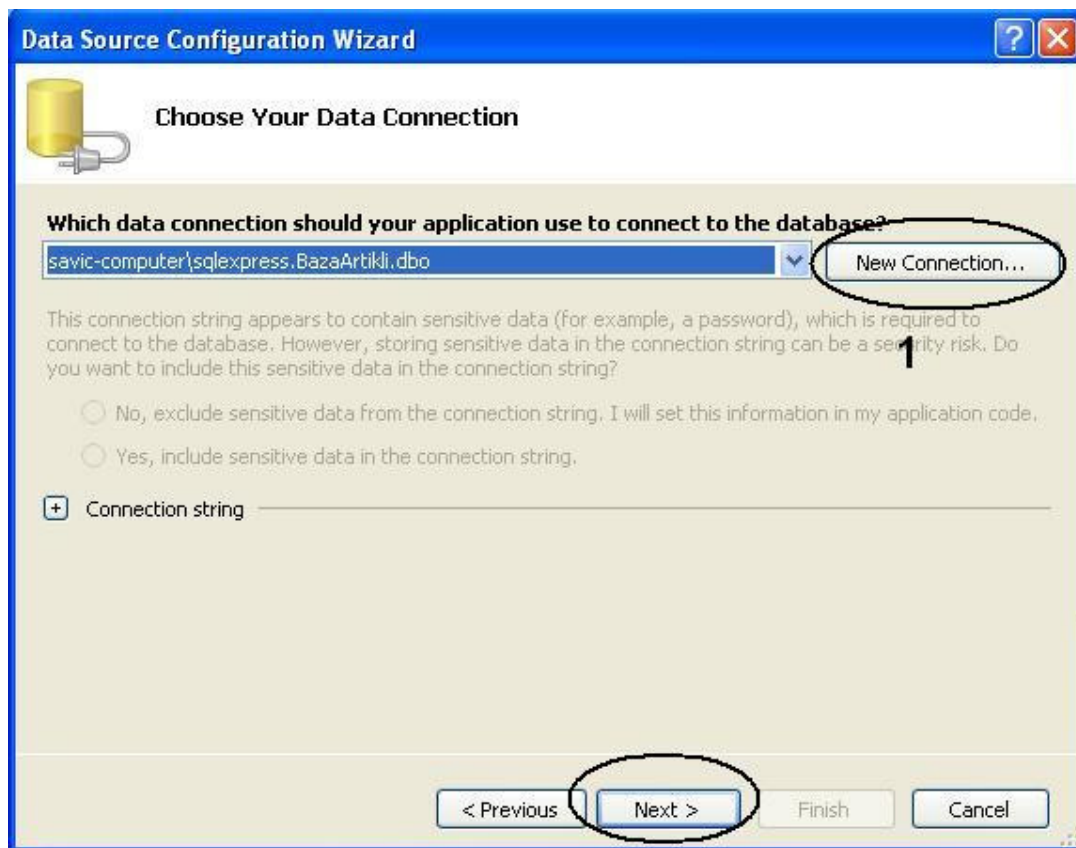
Slika 45. Opcija Database

Sistem će da vas pita Odaklen želite da vučete podatke, i daje vam tri opcije :

- DataBases
- Services
- Object

Mi naravno uzimamo Databases i idemo na Next....

Korak 3 :



Slika 46. Nova konekcija

Sada dolazimo do opcije kada biramo bazu podataka. Naravno sistem je već ponudio jednu bazu, koja u ovom slučaju nama ne treba. Da bi došli do naše baze moramo da idemo na New Connection, sa čime biramo novu konekciju sa drugom bazom.

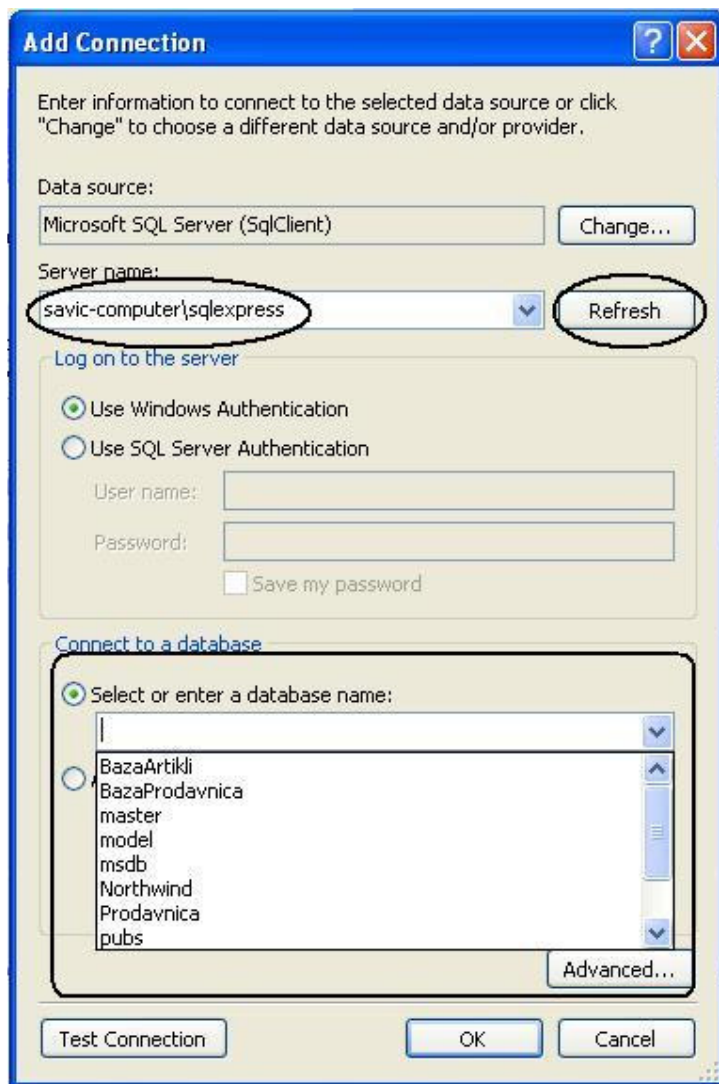
Korak 4 :

The screenshot shows the 'Add Connection' dialog box. At the top, it says 'Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.' Below this, the 'Data source' is set to 'Microsoft SQL Server (SqlClient)' with a 'Change...' button. The 'Server name' field is empty and circled in red. To its right is a 'Refresh' button. Below the server name, there's a section 'Log on to the server' with two radio buttons: 'Use Windows Authentication' (selected) and 'Use SQL Server Authentication'. Below these are fields for 'User name:' and 'Password:', and a checkbox for 'Save my password'. The next section is 'Connect to a database' with two radio buttons: 'Select or enter a database name:' (selected) and 'Attach a database file:'. Below these are a dropdown menu, a 'Browse...' button, and a 'Logical name:' field. At the bottom right is an 'Advanced...' button. At the very bottom are 'Test Connection', 'OK', and 'Cancel' buttons.

Slika 47. Polje za upis ime servera

U polje Server name upisujemo ime našeg računara i posle kose crte pišemo sqlexpress. Čekiramo opciju Use Windows Authentication, jer ne želimo user name i password. Na Refresh ažuriramo unešene podatke. Ako je sve dobro unešeno rezultate ćete videti na sledećoj slici.

Korak 5 :



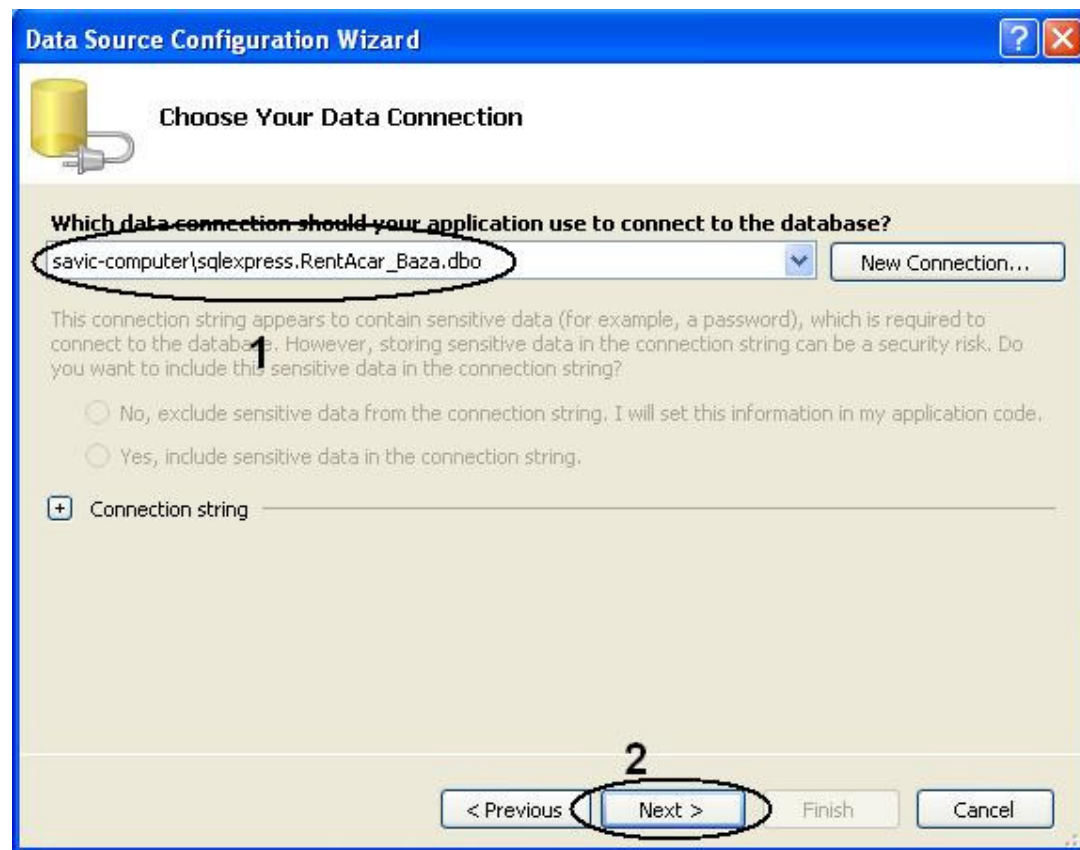
Slika 48. Izbor baze

Ako čekiramo Select or enter a database name i pritisnemo na polje dobićemo spisak svih baza u Serveru. Naravno mi biramo nasu RentACar_Base i pritisnemo Test Connection. Ako je sve u redu dobićemo poruku da je konekcija uspešna, kao na sledećoj slici.



Slika 49. Povezivanje uspešno

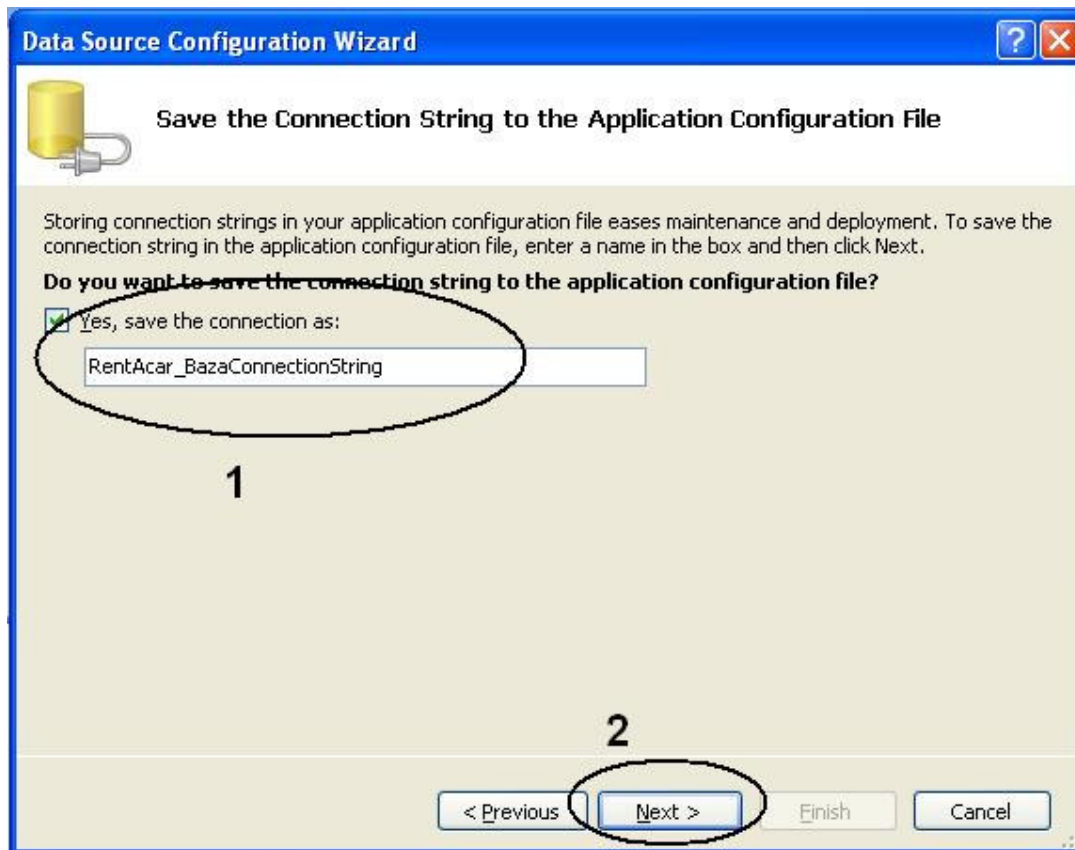
Korak 6 :



Slika 50. Nov naziv za konekciju

Kada se vratimo na formu za konekciju vidimo da je promenjena konekcija. Sada u nazivu konekcije piše ime naše baze. Idemo na Next.

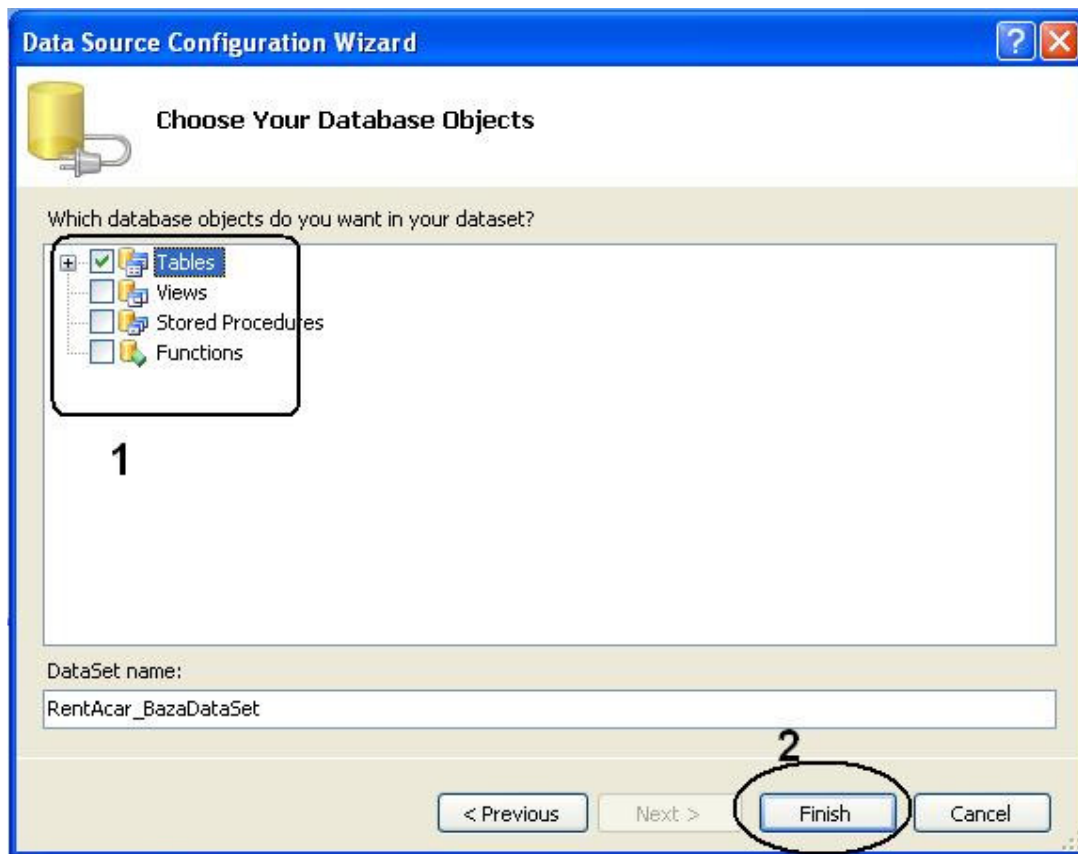
Korak 7 :



Slika 51. Konekcioni string

Sistem nam nudi naziv za ConnectionString. To je string koji nosi sve podatke o bazi. Taj string mi možemo da promenimo, da stavimo naziv koji se lako pamti, jer ćemo kasnije da ga koristimo u kodu. Ja lično ne menjam ono što mi je sistem ponudio. Idemo na Next.

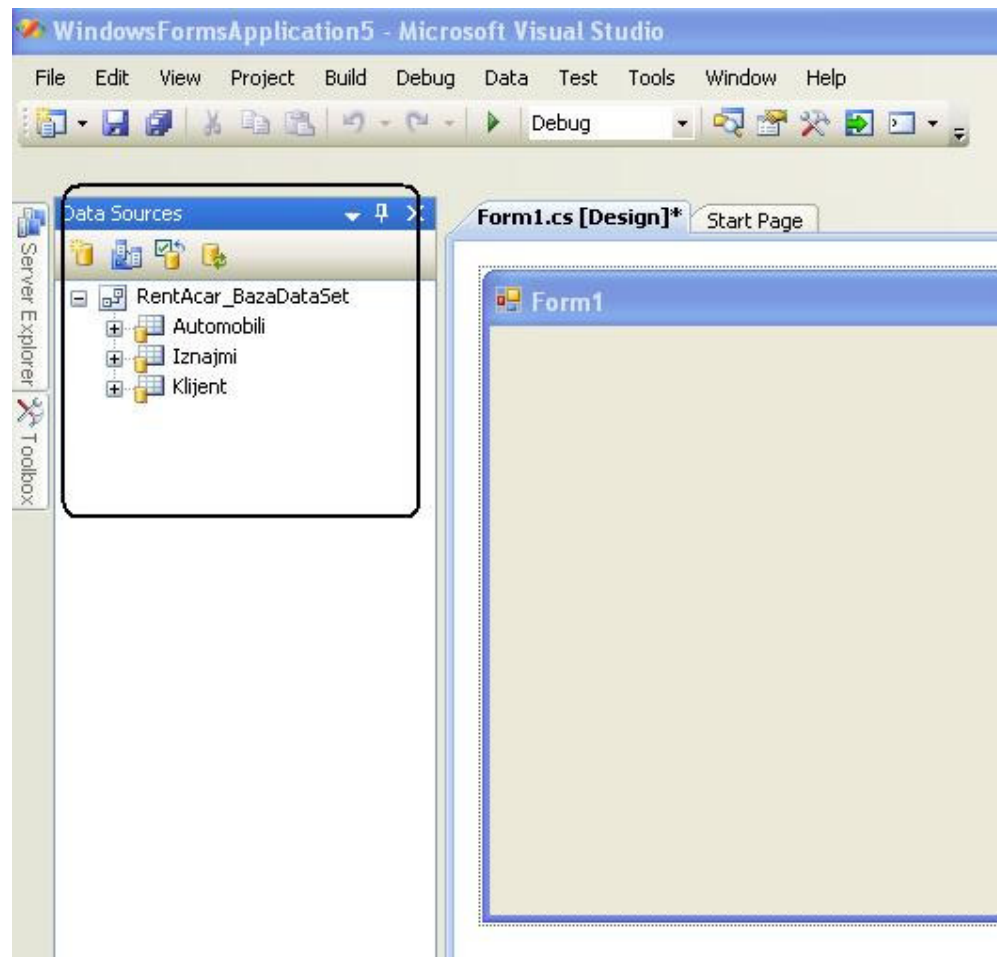
Korak 8 :



Slika 52. Izbor tabela baze

Od opcija koje nam sistem nudi mi biramo Tables. Tako ćemo da prenesemo sve tabele iz baze u nas Visual Studio 2003. Dugme Finish označava da se povezivanje završeno.

Korak 9 :



Slika 53. Data Source

Kada se vratimo u Visual Studio 2003 i pogledamo Data Sources, vidimo da se tu nalaze podaci o našoj bazi.

ZAKLJUČAK

Videli smo put razvoja jedne aplikacije sa bazom podataka. Da bi se jedna aplikacija napravila mora se proći kroz set dijagrama, i projektovanja same aplikacije. Naravno na početku je potrebno koristiti samo činjenice i zahteve korisnika. Na osnovu toga se radi SSA, koja nam pomaže da dobijemo neke okvire puteve za aplikaciju. SSA nam pomaže da pratimo tokove podataka i da opisemo sta ta aplikacija ustvari treba da radi. Ona se sastoji od tokova podataka, procesa i skladišta.

Tokovi podataka su putevi kojima će naši zahtevi i informacije putovati. Skladišta su usko povezana sa bazama podataka, i ona ustvari predstavljaju tabele u bazi podataka. Proces i u SSA su metode u aplikaciji. Posle toga pravimo rečnik podataka koji nam verno predstavlja tabele u našoj bazi. Rečnik podataka sadrži imena koja ćemo da koristimo za polja u tabelama, tip promjenjive, ograničenja.

Nakon kreiranja baze podataka dolazimo do pravljenja Use Case dijagrama, sa kojima opisujemo aplikaciju. Sve procese mi verno prikazujemo i njihove tokove. Sa ovim dijagramima već imamo neke smernice u pravljenju informacionog sistema. Use case se može reci da je scenario našeg IS.

Sekvencijalni dijagram nam prikazuje procese sa tačnim vremenom korištenja. Svakom procesu je dato neko vreme koje je potrebno da se ono završi.

Da bi dokumentacija bila kompletna moralo sam napraviti grafičke prikaze same aplikacije i jasnije objasniti povezivanje na bazu.

Znamo da se aplikacija sastoji od formi koje su medjusobno povezane i koje komuniciraju. Svaka forma na sebi ima neka polja, dugmad, checkbox-ove. Svaki od ovih komponenti ima svoj C# kod koji krajnji korisnik ne vidi. Njemu nije ni važno kako to radi.

Za bolje razumevanje aplikacije za svaku formu sam prikazao neki najvažniji dio koda. Svaki red koda je prokomentarisan i komentari su označeni zelenom bojom.

Povezivanje same aplikacije sa bazom podataka ispratio sam u nekoliko slika, kao i pravljenje nove baze.

TABELA SLIKA

Slika 1. Dijagram toka podataka	5
Slika 2. Dijagram konteksta Rent A Car	6
Slika 3. Dekompozicija prvog nivoa	7
Slika 4. Drugi nivo dekompozicije - Iznajmljivanje	8
Slika 5. Drugi nivo dekompozicije - Vraćanje	9
Slika 6. Drugi nivo dekompozicije - Vraćanje	10
Slika 7. Use Case - Rent a Car	14
Slika 9. Odluke	20
Slika 10. Uporedne staze	21
Slika 11. Signali	21
Slika 12. Dijagram aktivnosti - Porudžbina	22
Slika 13. Dijagram aktivnosti - pregled automobila	23
Slika 14. Dijagram aktivnosti - Pregled klijenata	24
Slika 15. Dijagram aktivnosti - Unos novog klijenta	25
Slika 16. Dijagram aktivnosti - Iznajmljivane	26
Slika 17. Dijagram aktivnosti - Vraćanje automobila	27
Slika 18. Dijagram aktivnosti - Plaćanje	28
Slika 19. Dijagram aktivnosti - Produživanje ugovora	29
Slika 20. Dijagram aktivnosti - Unos novog vozila	30
Slika 21. Sekvencijalni dijagram - Iznajmljivanje automobila	31
Slika 22. Sekvencijalni dijagram - Vraćanje automobila	31
Slika 23. Sekvencijalni dijagram - Produživanje ugovora	32
Slika 24. Sekvencijalni dijagram - Pretraga	32
Slika 25. Dijagram klase - Rent a Car	33
Slika 26. Glavna forma	34
Slika 27. Pregled opcija	36
Slika 28. Forma za iznajmljivanje automobila	37
Slika 29. Forma za vraćanje automobila	39
Slika 30. Forma sa dodatnim troškovima	40
Slika 31. Forma za produžavanje ugovora	41
Slika 32. Pregled klijenata	42
Slika 33. Forma za dodavanje klijenata	43
Slika 34. Forma za pretragu klijenata	45
Slika 35. Opcija automobili	47
Slika 36. Forma za dodavanje automobila	48
Slika 37. Forma za pretragu automobila	49
Slika 38. Startovanje Sql Managenent aplikacije	50
Slika 39. Logovanje na SQL Server	51
Slika 40. Pregled svih baza	52
Slika 41. Opcija Modify	53
Slika 42. Prikaz polja tabele Automobili za izmenu	54
Slika 43. Prikaz polja za popunjavanje	55
Slika 44. Izbor novog Data Source-a	56
Slika 45. Opcija Database	57
Slika 46. Nova konekcija	58
Slika 47. Polje za upis ime servera	59
Slika 48. Izbor baze	60

Slika 49. Povezivanje uspešno	60
Slika 50. Nov naziv za konekciju	61
Slika 51. Konekcioni string.....	62
Slika 52. Izbor tabela baze.....	63
Slika 53. Data Source	64

SADRŽAJ

UVOD	2
RENT-A-CAR SCENARIO	3
VERBALNI OPIS SISTEMA	3
STRUKTURNA SISTEMSKA ANALIZA - SSA.....	4
OSNOVNI PODATCI	4
DIJAGRAM KONTEKSTA RENT A CAR.....	6
PROCES IS RENT A CAR.....	6
DIJAGRAM DEKOMPOZICIJE PRVOG NIVO.....	7
DRUGI NIVO DEKOMPOZICIJE – 1. IZNAJMLJIVANJE	8
PROCES 1.1 – OBRADA ZAHTEVA.....	8
PROCES 1.2 – PRAVLJENJE UGOVORA.....	8
PROCES 1.3 – PREDAJA AUTOMOBILA KLIJENTU	8
DRUGI NIVO DEKOMPOZICIJE – 2. VRAĆANJE	9
PROCES 2.1 – UPLATA.....	9
PROCES 2.2 – PRODUŽIVANJE RENTE	9
DRUGI NIVO DEKOMPOZICIJE – 3. PRETRAGA	10
PROCES 3.1 PRETRAGA KLIJENATA	10
PROCES 3.1 PRETRAGA KLIJENATA	10
PROCES 3.1 PRETRAGA AUTOMOBILA	10
PROCES 3.1 PRETRAGA KLIJENATA	10
REČNIK PODATAKA	11
REČNIK PODATAKA ZA IS RENT A CAR	11
TABELA AUTOMOBILI.....	11
TABELA IZNAJMI.....	12
TABELA KLIJENT.....	13
USE CASE – RENT A CAR	14
USE CASE 1. NARUČIVANJE	15
USE CASE 2. PREGLED AUTOMOBILA	15
USE CASE 3. PREGLED KLIJENATA	16
USE CASE 4. UNOS KLIJENTA.....	16
USE CASE 5. IZNAJMLJIVANJE.....	17
USE CASE 6. VRAĆANJE AUTOMOBILA	17
USE CASE 7. PLAĆANJE	18
USE CASE 8. PRODUŽIVANJE UGOVORA.....	18
USE CASE 9. UNOS NOVIH KOLA.....	19
DIJAGRAMI AKTIVNOSTI	20
ŠTA JE DIJAGRAM AKTIVNOSTI?	20

<i>Odluke</i>	20
<i>Uporedne staze</i>	21
<i>Signali</i>	21
DIJAGRAM AKTIVNOSTI 1. PORUDŽBINA	22
DIJAGRAM AKTIVNOSTI 2. PREGLED AUTOMOBILA	23
DIJAGRAM AKTIVNOSTI 3. PREGLED KLIJENATA	24
DIJAGRAM AKTIVNOSTI 4. UNOS NOVOG KLIJENTA	25
DIJAGRAM AKTIVNOSTI 5. IZNAJMLJIVANJE	26
DIJAGRAM AKTIVNOSTI 6. VRAĆANJE AUTOMOBILA	27
DIJAGRAM AKTIVNOSTI 7. PLAĆANJE	28
DIJAGRAM AKTIVNOSTI 8. PRODUŽIVANJE UGOVORA	29
DIJAGRAM AKTIVNOSTI 9. UNOS NOVIH VOZILA	30
SEKVENCIJALNI DIJAGRAM AKTIVNOSTI	31
DIJAGRAM KLASA	33
GRAFIČKI PRIKAZ SISTEMA	34
GLAVNA FORMA	34
GLAVNA FORMA – OPCIJE	36
FORMA IZNAJMLJIVANJE AUTOMOBILA	37
FORMA POVRAĆAJ AUTOMOBILA	39
FORMA POVRAĆAJ AUTOMOBILA – DODATNI TROŠKOVI	40
FORMA PRODUŽIVANJE UGOVORA	41
GLAVNA FORMA – KLIJENTI	42
FORMA DODAVANJE KLIJENATA	43
FORMA PRETRAGA KLIJENATA	45
GLAVNA FORMA – AUTOMOBILI	47
FORMA DODAVANJE AUTOMOBILA	48
FORMA PRETRAGA AUTOMOBILA	49
POVEZIVANJE NA BAZU PODATAKA	50
RAD SA BAZOM RENT A CAR	50
<i>PROCES IZRADE BAZE PODATAKA</i>	50
<i>POVEZIVANJE NOVE APLIKACIJE I BAZE PODATAKA</i>	56
<i>ZAKLJUČAK</i>	65
TABELA SLIKA	66
SADRŽAJ	67