UNIVERZITET U BEOGRADU

FAKULTET ORGANIZACIONIH NAUKA

Seminarski rad iz predmeta

**INTERNET TEHNOLOGIJE**

Naziv teme:

**Web aplikacija za RentaCar agenciju**

Profesor: dr Marijana Despotović

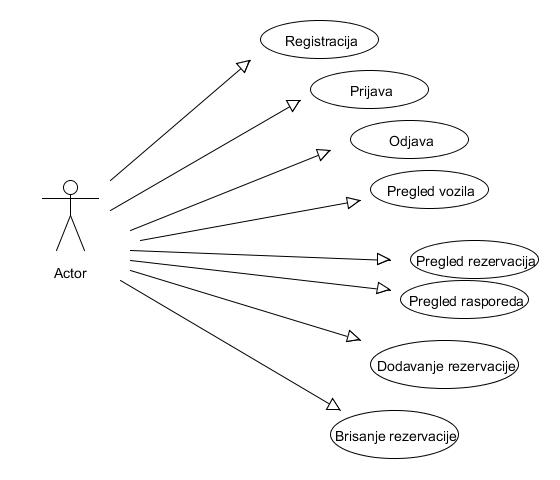
Student: Seke Nenad , 203/04

2010.

# Korisnicki zahtev

Realizvoati aplikacuju za vodjenje online RentaCar agencije. Neophodno je da aplikacija ima podeljen administratorski i korisnicki deo. Korisnicki deo treba da omoguci sledece mogucnosti korisnicima, a to su: da korisnik moze da pogleda raspolozive automobile koje agencija nudi za rentiranje, da vidi njihove slike i osobine, da moze da pogleda zauzece automobila i da rezervise isti automobil. Takodje korinik treba da ima omogucenu opciju da otkaze rezervaciju automobila. Administrator pored svih mogucnosti koje ima i obican korisnik, on moze imati i uvid u sve rezervacije ostalih korisnika a takodje moze i obrisati rezervacije ostalih korisnika. Pored toga administrator treba da moze da doda novi automobil na sistem.

# Dijagram slucajeva koriscenja - korisnicki deo



**SK1: Slučaj korišćenja – Registracija**

**Naziv:** Registracija

**Svrha:** Služi za registraciju novih korisnika

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Osnovni scenario:**

1. Korisnik na glavnoj strani pritiska na link “Registruj se”

2. Sistem prikazuje novu stranicu na kojoj korisnik unosi treba da unese svoje podatke

3. Korisnik unosi podatke : ime , prezime i korisničko ime,password, datum rodjenja i email

4. Sistem ga obaveštava preko AJAX-a o mogućnosti korišćenja odabranog korisničkog imena

5. Korisnik pritiska dugme “Registruj se”

6. Sistem ga ubacuje u bazu podataka.

**Alternativni scenario:**

3.1. Ako je korisnik uneo Username koji se već nalazi u bazi, sistem pomoću AJAX-a odmah obaveštava o tome

5.1 U koliko korisnik nije uneo sve podatke sistem ga obaveštava o tome

6.1 Ako sistem ne može da se poveže sa bazom, izbacuje poruku o tome

**SK2: Slučaj korišćenja – Prijava**

**Naziv:** Prijava na sajt

**Svrha:** Služi za prijavu registrovanog korisnika na sistem

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Preduslov:** Korisnik je registrovan

**Osnovni scenario:**

1. Korisnik na početnoj strani u formu za logovanje unosi korisničko ime i lozinku

2. Sistem proverava podatke u bazi, loguje korisnika na sajt i otvara stranicu u kojoj obaveštava o uspešnoj prijavi

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu, izbacuje poruku o tome

2.2 U koliko korisnuik nije uneo sve podatke uodgovarajuća polja sistem ga obaveštava o tome

2.3 Ukoliko u bazi ne postoji registrovani korisnik sa datim korisničkim imenom i lozinkom sistem obaveštava o pogrešnom unosu podataka

**SK3: Slučaj korišćenja – Odjava**

**Naziv:** Odjava sa sajta

**Svrha:** Služi za odjavu korisnika sa sistema

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Preduslov:** Korisnik je prijavljen na sistem

**Osnovni scenario:**

1. Korisnik u gornjem desnom uglu bira opciju “izloguj se”

2.Sistem odjavljuje korisnika, briše cockie i postavlja korsnika na početnu stranu

**SK4: Slučaj korišćenja – Pregled vozila**

**Naziv:** Pregled vozila

**Svrha:** Služi za pregled vozila koji se nalaze agenciji

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Preduslov:** Korisnik je prijavljen na sistem

**Osnovni scenario:**

1. Korisnik na početnoj strani bira link „vozila“ koje se nalazi u gornjem meniju

2. Sistem prikazuje vozila u tabeli sa svim dedaljima

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

**SK5: Slučaj korišćenja – Pregled rezervacija**

**Naziv:** Pregled rezervacije

**Svrha:** Služi za prikaz rezervacija korisnika

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Preduslov:** Korisnik je prijavljen na sistem

**Osnovni scenario:**

1. Korisnik odlazi na stranu „moje rezervacije“ odabirom linka iz gornjeg menija.
2. Sistem ucitava sve rezervacije za logovanog korisnika i prikazuje ih na strani.

**Alternativni scenario:**

2.1 U koliko sistem ne može da se poveže na bazu i obaveštava korisnika o tome

**SK6: Slučaj korišćenja – Pregled raporeda**

**Naziv:** Pregled knjiga

**Svrha:** Služi za pregled vozila koji se agenciji

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Preduslov:** Korisnik je prijavljen na sistem

**Osnovni scenario:**

1. Korisnik na početnoj strani bira link „reaspored“ koje se nalazi u gornjem meniju

2. Korisnik odabira mesec i godinu iz kombo boksa za koje zeli da vidi raspored.

3. Korisnik poziva sistem da prikaze raspored za unete podatke.

4. Sistem prikazuje korisniku raspored.

**Alternativni scenario:**

4.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

4.1 Ukoliko sistem ne može da prikaze raspored izbacuje poruku o tome

**SK7: Slučaj korišćenja – Dodavanje rezervacije**

**Naziv:** Rezervisanje vozila

**Svrha:** Služi za rezervisanje vozila

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Preduslov:** Korisnik je prijavljen na sistem

**Osnovni scenario:**

1. Korisnik odabira vozilo iz padajuceg menija, unosi datum rezervacije.
2. Korisnik poziva sistem da zapamti rezervaciju.
3. Sistem pamti rezervaciju i obavestava korisnika o tome.

**Alternativni scenario:**

2.1 U koliko sistem ne može da se poveže na bazu i obaveštava korisnika o tome

**SK8: Slučaj korišćenja – Brisanje rezervacije**

**Naziv:** Brisanje ezervisanje

**Svrha:** Služi za brisnje rezervisanje vozila

**Akteri:** Korisnik

**Učesnici:** Korisnik i sistem

**Preduslov:** Korisnik je prijavljen na sistem

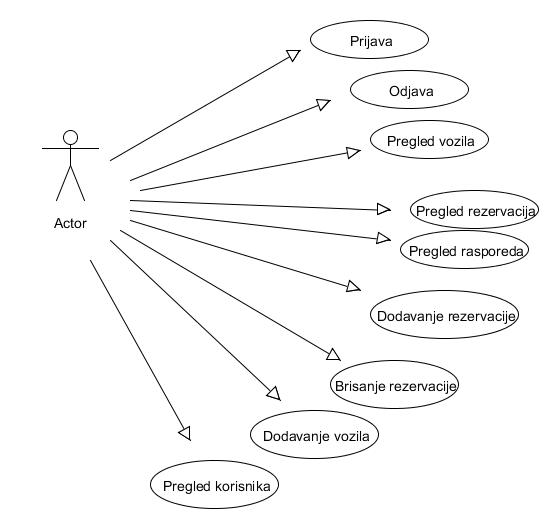
**Osnovni scenario:**

1. Korisnik odabira rezervaciju koju zeli da obrise i poziva sistem da je obrise.
2. Sistem brise rezervaciju.

**Alternativni scenario:**

2.1 U koliko sistem ne može da obrise rezervaciju obaveštava korisnika o tome.

# Dijagram slucajeva koriscenja - administratorski deo



**SK1: Slučaj korišćenja – Prijava**

**Naziv:** Prijava na sajt

**Svrha:** Služi za prijavu registrovanog korisnika na sistem

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je registrovan

**Osnovni scenario:**

1. Administrator na početnoj strani u formu za logovanje unosi korisničko ime i lozinku

2. Sistem proverava podatke u bazi, loguje administratora na sajt i otvara stranicu u kojoj obaveštava o uspešnoj prijavi

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu, izbacuje poruku o tome

2.2 U koliko administrator nije uneo sve podatke uodgovarajuća polja sistem ga obaveštava o tome

2.3 Ukoliko u bazi ne postoji registrovani administrator sa datim korisničkim imenom i lozinkom sistem obaveštava o pogrešnom unosu podataka

**SK2: Slučaj korišćenja – Odjava**

**Naziv:** Odjava sa sajta

**Svrha:** Služi za odjavu administratoa sa sistema

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator u gornjem desnom uglu bira opciju “izloguj se”

2.Sistem odjavljuje korisnika, briše cockie i postavlja administratora na početnu stranu

**SK3: Slučaj korišćenja – Pregled vozila**

**Naziv:** Pregled vozila

**Svrha:** Služi za pregled vozila koji se nalaze agenciji

**Akteri:** administrator

**Učesnici:** Administrator i sistem

**Preduslov:** administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator na početnoj strani bira link „vozila“ koje se nalazi u gornjem meniju

2. Sistem prikazuje vozila u tabeli sa svim dedaljima

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

**SK4: Slučaj korišćenja – Pregled rezervacija**

**Naziv:** Pregled rezervacije

**Svrha:** Služi za prikaz rezervacija korisnika

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator odlazi na stranu „moje rezervacije“ odabirom linka iz gornjeg menija.
2. Sistem ucitava sve rezervacije za logovanog administratora i prikazuje ih na strani.

**Alternativni scenario:**

2.1 U koliko sistem ne može da se poveže na bazu i administratora korisnika o tome

**SK5: Slučaj korišćenja – Pregled rasporeda**

**Naziv:** Pregled knjiga

**Svrha:** Služi za pregled vozila koji se agenciji

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator na početnoj strani bira link „reaspored“ koje se nalazi u gornjem meniju

2. Administrator odabira mesec i godinu iz kombo boksa za koje zeli da vidi raspored.

3. Administrator poziva sistem da prikaze raspored za unete podatke.

4. Sistem prikazuje administratou raspored.

**Alternativni scenario:**

4.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

4.1 Ukoliko sistem ne može da prikaze raspored izbacuje poruku o tome

**SK6: Slučaj korišćenja – Dodavanje rezervacije**

**Naziv:** Rezervisanje vozila

**Svrha:** Služi za rezervisanje vozila

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator odabira vozilo iz padajuceg menija, unosi datum rezervacije.
2. Administrator poziva sistem da zapamti rezervaciju.
3. Sistem pamti rezervaciju i obavestava administratora o tome.

**Alternativni scenario:**

2.1 U koliko sistem ne može da se poveže na bazu i obaveštava administratora o tome

**SK7: Slučaj korišćenja – Brisanje rezervacije**

**Naziv:** Brisanje ezervisanje

**Svrha:** Služi za brisnje rezervisanje vozila

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator odabira rezervaciju koju zeli da obrise i poziva sistem da je obrise.
2. Sistem brise rezervaciju.

**Alternativni scenario:**

2.1 U koliko sistem ne može da obrise rezervaciju obaveštava korisnika o tome.

**SK8: Slučaj korišćenja – Dodavanje vozila**

**Naziv:** Dodavanje vozila

**Svrha:** Služi za dodavanje vozila

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator odlazi na stranu dodaj vozilo.
2. Administrator unosi podatke o vozilu.
3. Administrator poziva sistem da zapamti vozilo.
4. Sistem pamti vozilo i obavestava administratora o tome

**Alternativni scenario:**

* 1. U koliko sistem ne može da zapamti vozilo obavestava administratora o tome.

**SK9: Slučaj korišćenja – Pregled korisnika**

**Naziv:** Pregled korisnika

**Svrha:** Služi za Prikaz svih registrovanih korisnika

**Akteri:** Administrator

**Učesnici:** Administrator i sistem

**Preduslov:** Administrator je prijavljen na sistem

**Osnovni scenario:**

1. Administrator odlazi na stranu korisnici.
2. Sistem prikazuje administratoru sve registrovane korsnike

**Alternativni scenario:**

* 1. U koliko sistem ne može da zapamti vozilo obavestava administratora o tome.

# Opis proces za par karakteristicnih slucajeva koriscenja

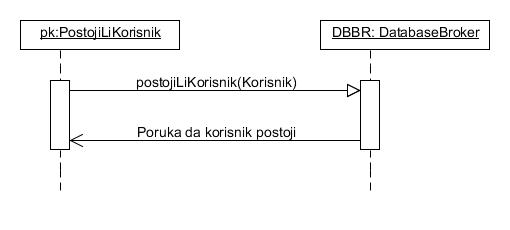
# (dijagrama sekvenci)

**SK1: Slučaj korišćenja – Prijava**

**Osnovni scenario:**

1. Administrator na početnoj strani u formu za logovanje unosi korisničko ime i lozinku

2. Sistem proverava podatke u bazi, loguje administratora na sajt i otvara stranicu u kojoj obaveštava o uspešnoj prijavi

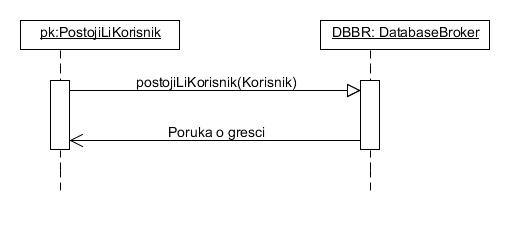
****

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu, izbacuje poruku o tome

2.2 U koliko administrator nije uneo sve podatke uodgovarajuća polja sistem ga obaveštava o tome

2.3 Ukoliko u bazi ne postoji registrovani administrator sa datim korisničkim imenom i lozinkom sistem obaveštava o pogrešnom unosu podataka

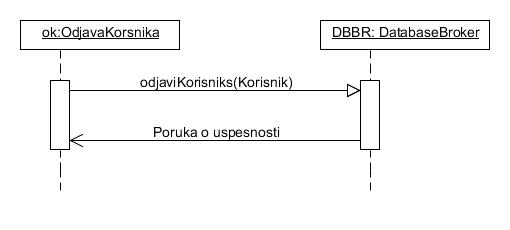


**SK2: Slučaj korišćenja – Odjava**

**Osnovni scenario:**

1. Administrator u gornjem desnom uglu bira opciju “izloguj se”

2.Sistem odjavljuje korisnika, briše cockie i postavlja administratora na početnu stranu

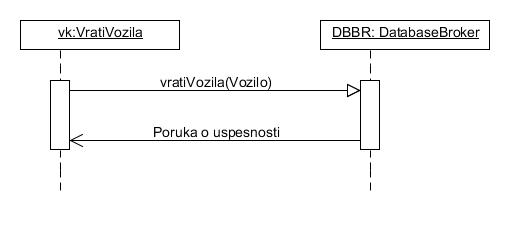


**SK3: Slučaj korišćenja – Pregled vozila**

**Osnovni scenario:**

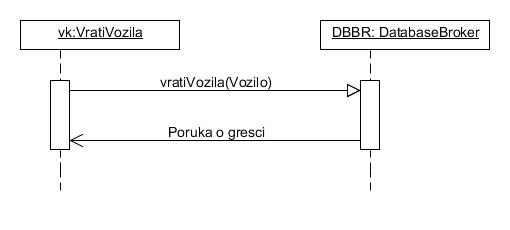
1. Administrator na početnoj strani bira link „vozila“ koje se nalazi u gornjem meniju

2. Sistem prikazuje vozila u tabeli sa svim dedaljima

****

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome



**SK4: Slučaj korišćenja – Pregled rezervacija**

**Osnovni scenario:**

1. Administrator odlazi na stranu „moje rezervacije“ odabirom linka iz gornjeg menija.
2. Sistem ucitava sve rezervacije za logovanog administratora i prikazuje ih na strani.

**Alternativni scenario:**

2.1 U koliko sistem ne može da se poveže na bazu i administratora korisnika o tome

**SK5: Slučaj korišćenja – Pregled raporeda**

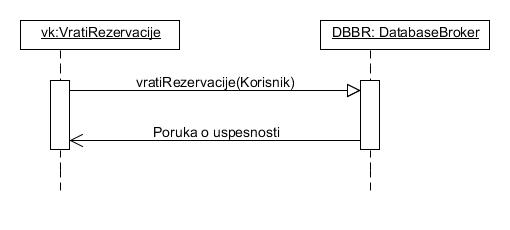
**Osnovni scenario:**

1. Administrator na početnoj strani bira link „reaspored“ koje se nalazi u gornjem meniju

2. Administrator odabira mesec i godinu iz kombo boksa za koje zeli da vidi raspored.

3. Administrator poziva sistem da prikaze raspored za unete podatke.

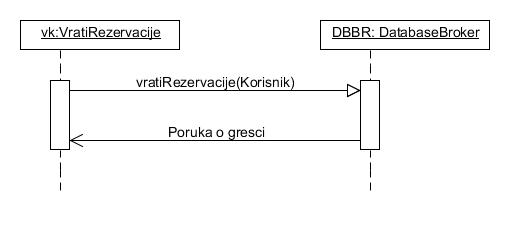
4. Sistem prikazuje administratou raspored.



**Alternativni scenario:**

4.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

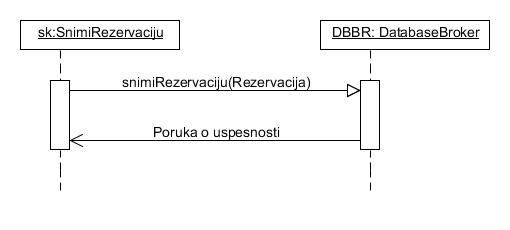
4.1 Ukoliko sistem ne može da prikaze raspored izbacuje poruku o tome



**SK6: Slučaj korišćenja – Dodavanje rezervacije**

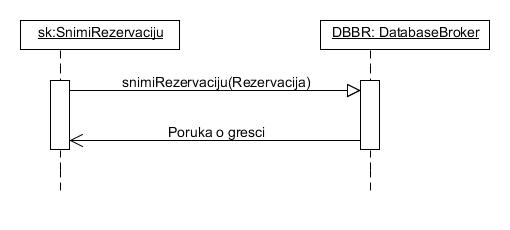
**Osnovni scenario:**

1. Administrator odabira vozilo iz padajuceg menija, unosi datum rezervacije.
2. Administrator poziva sistem da zapamti rezervaciju.
3. Sistem pamti rezervaciju i obavestava administratora o tome.



**Alternativni scenario:**

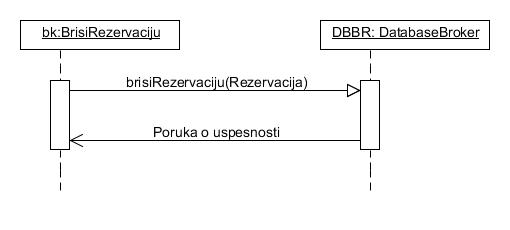
2.1 U koliko sistem ne može da se poveže na bazu i obaveštava administratora o tome



**SK7: Slučaj korišćenja – Brisanje rezervacije**

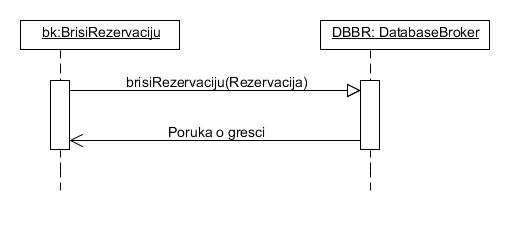
**Osnovni scenario:**

1. Administrator odabira rezervaciju koju zeli da obrise i poziva sistem da je obrise.
2. Sistem brise rezervaciju.



**Alternativni scenario:**

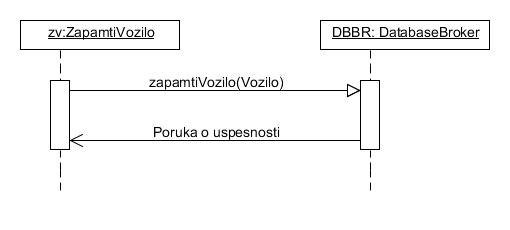
2.1 U koliko sistem ne može da obrise rezervaciju obaveštava korisnika o tome.



**SK8: Slučaj korišćenja – Dodavanje vozila**

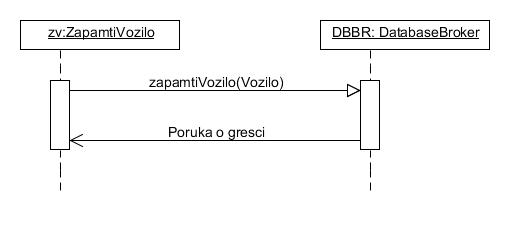
**Osnovni scenario:**

1. Administrator odlazi na stranu dodaj vozilo.
2. Administrator unosi podatke o vozilu.
3. Administrator poziva sistem da zapamti vozilo.
4. Sistem pamti vozilo i obavestava administratora o tome



**Alternativni scenario:**

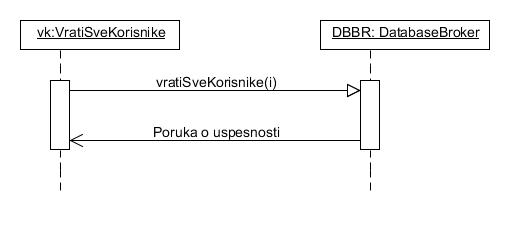
* 1. U koliko sistem ne može da zapamti vozilo obavestava administratora o tome.



**SK9: Slučaj korišćenja – Pregled korisnika**

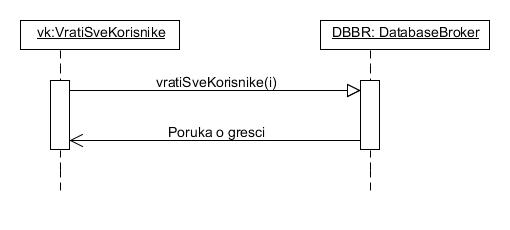
**Osnovni scenario:**

1. Administrator odlazi na stranu korisnici.
2. Sistem prikazuje administratoru sve registrovane korsnike



**Alternativni scenario:**

4.1 U koliko sistem ne može da zapamti vozilo obavestava administratora o tome.



# Model podataka – MOV



# Prevodjenje PMOV u relacioni model

Korisnik(**username#,**password,ime,prezime,datumrodj,mail,mail,tip)

Vozilo(**sifra#**,model,klasa,potrosnja,1dan,2dana,34dana,56dana,7dana,14dana)

Rezervacije(***username#,sifra#****,*datumOd,datumDo)

CREATE TABLE `korisnik` (

`username` varchar(20) NOT NULL DEFAULT '',

`password` varchar(20) DEFAULT NULL,

`ime` varchar(20) DEFAULT NULL,

`prezime` varchar(20) DEFAULT NULL,

`datumrodj` varchar(20) DEFAULT NULL,

`email` varchar(20) DEFAULT NULL,

`tip` varchar(1) DEFAULT NULL,

PRIMARY KEY (`username`)

)

CREATE TABLE `vozilo` (

`sifra` double NOT NULL AUTO\_INCREMENT,

`model` varchar(20) DEFAULT NULL,

`klasa` varchar(20) DEFAULT NULL,

`dan1` double DEFAULT NULL,

`dan2` double DEFAULT NULL,

`dan34` double DEFAULT NULL,

`dan56` double DEFAULT NULL,

`dan7` double DEFAULT NULL,

`dan14` double DEFAULT NULL,

`potrosnja` double DEFAULT NULL,

PRIMARY KEY (`sifra`)

)

CREATE TABLE IF NOT EXISTS `rezervacija` (

`username` varchar(20) NOT NULL DEFAULT '',

`sifra` double NOT NULL DEFAULT '0',

`datumod` date DEFAULT NULL,

`datumdo` date DEFAULT NULL,

PRIMARY KEY (`username`,`sifra`),

KEY `rezervacija\_fk1` (`sifra`),

CONSTRAINT `rezervacija\_fk` FOREIGN KEY (`username`) REFERENCES `korisnik` (`username`),

CONSTRAINT `rezervacija\_fk1` FOREIGN KEY (`sifra`) REFERENCES `vozilo` (`sifra`)

)

# Opis tehnologija korišćenih u aplikaciji

## HTML

HTML, HyperText Markup Language, je standarizovani jezik koji se koristi pri strukturiranju tekstova, medija i ugrađenih objekata u web stranice i elektronsku poštu. Kao modifikovanu i pojednostavljenu verziju SGML jezika, HTML standarizuje i održava World Wide Web Consortium (W3C). Originalnu verziju HTML-a kreirao je Tim Berners-Lee, a prva zvanična verzija je izašla juna 1993 godine.

HTMl jezik je sačinjen od običnog teksta i tagova. Iako se HTML znakovi (tags) često nazivaju kodom, tehnički HTML nije kod jer kompjuterski kod su instrukcije koje od računara traže da izvrši određenu operaciju. Svrha HTML znakova je da se se struktura dokumenta "označi" tako da bi korisnikov agent (user agent), tj. internet pretraživač mogao da prepozna strukturu dokumenta i ispravno je prikaže u prozoru internet pretraževača kojeg korisnik koristi.

Za dodatno modifikovanje web stranica, do sada su izašli neki dodaci:

* CSS ili Cascading Style Sheets za izgled i položaj prezentacije
* Skriptni jezici (Javascript, VBScript) za omogućavanje dinamičnosti i interaktivnosti na web prezentacijama,
* DOM ili Document Object Model, koji označava vezu između skripte i elementa na stranici

Gornji dodaci, u paketu sa HTML jezikom, ponekad se nazivaju DHTML ili Dinamični HTML.

#### CSS

*CSS* je u osnovi skup instrukcija koje definišu kako neki HTML dokument treba da se prikaže. Najbolja stvar kod CSS-a je da se on može deflnisati u spoljašnjoj datoteci, ili listu (sheet). Pretraživač onda može da pročita ovaj spoljašnji list i da primeni sadržaj na određeni sadržaj datoteke. Na taj način se stil i formatiranje strane odvajaju od samog sadržaja. Ovo je vrlo bitan koncept kod dizajna web strana. Ovo Vam pruža moćan i detaljan način da ažurirate svoj web sajt.

## JAVASCRIPT

Java Script pripada grupi jezika za skriptovanje, pre svega klijentske strane, mada se može izvršavati i na serveru (runat=“server“). Java Script je najpopularniji jezik na Internetu, koji je dizajniran da poveća interaktivnost HTML strana.

JavaScript dodaje interaktivnost veb prezentaciji na strani klijenta. Na primer, korišćenjem JavaScripta je moguće odgovarati na akcije korisnika u samom veb čitaču. Bitno je naglasiti da JavaScript program može da se izvrašava samo u okviru veb čitača i nigde drugde. Pri kreiranju JavaScript koda može se koristiti bilo koji ASCII editor.

## PHP

PHP je open source jezik koji se koristi za razvoj server-side aplikacija, kao i dinamičkog Web sadržaja. PHP dozvoljava interakciju sa velikim brojem relacionih baza podataka kao što su MySQL, Oracle, IBM D2, Microsoft SQL Server, PostgreSQL i SQLite. PHP radi na većini operativnih sistema današnjice, kao što su UNIX, Linux, Windows i Mac OS i može da interaguje sa većinom Web servera.

PHP je stekao popularnost zbog svoje jednostavnosti i sintakse nasleđene iz programskog jezika C. Tokom vremena jezik se proširivao i sticao mogućnosti za objektno orijentisano programiranje, naročito od verzije 5.0. Nalikuje jeziku C++ u smislu da dozvoljava i čisto-proceduralno programiranje ali omogućava i korišćenje klasa i drugih koncepata objektno orijentisanog programiranja (nasleđivanje, apstraktne metode, interfejsi itd.).

Za PHP fajlove, bitne su sledeće činjenice:

* mogu da sadrže tekst, HTML tagove i skripte
* vraćaju se klijentskom browser – u kao HTML tekst (*plain HTML*)
* imaju ekstenziju: “.*php*”, “.*php3*” ili “.*phtml*”

## AJAX

AJAX predstavlja web razvojnu tehniku za kreiranje interaktivnih web sadržaja. AJAX tehnologija se zasniva na razmeni malih količina podataka sa serverom, kako korisnik ne bi stalno morao da osvežava stranicu. Na taj način se omogućava potpuna interaktivnost, efikasnost i poboljšava funkcionisanje stranice uopšte. AJAX nije tehnologija sam po sebi, već termin koji se odnosi na korišćenje grupe tehnologija.

AJAX obuhvata:

* prezentaciju baziranu na standardima koristeći XHTML i CSS;
* dinamički prikaz i interakciju preko Document Object Model-a;
* razmenu i manipulaciju nad podacima koristeći XML i XSLT;
* asinhrono prikupljanje podataka uz pomoć XMLHttpRequest-a koji čini jezgro;
* i JavaScript koji povezuje sve u celinu.

Ideja koja se krije iza AJAX-a jeste da se stranica na kojoj se odvija web aplikacija učita samo jednom, a da se svaka dalja komunikacija sa serverom sakrije od očiju korisnika i obavlja bez ponovnog učitavanja čitave stranice. Svaki prenos podataka između servera i klijenta (u slučaju AJAX-a to je browser) vrši se u pozadini. Jasno je da je ovo nemoguće izvesti statičkim HTML-om, pa tu na scenu stupa JavaScript. JavaScript je zadužen za komuniciranje sa serverom – slanje HTTP zahteva, prijem podataka sa servera i njihov prikaz na stranici, a i interakcija sa korisnikom postiže se korišćenjem ovog jezika.

## SQL

SQL je standardni programski jezik za pristup i manipulaciju podacima iz relacionih baza podataka i podržan je od strane svih servera relacionih baza podataka. U relacionim bazama podaci su skladišteni preko niza tabela. Svaka tabela sadrži kolone koje opsiuju atribute podataka, a svaki red je instanca podataka.

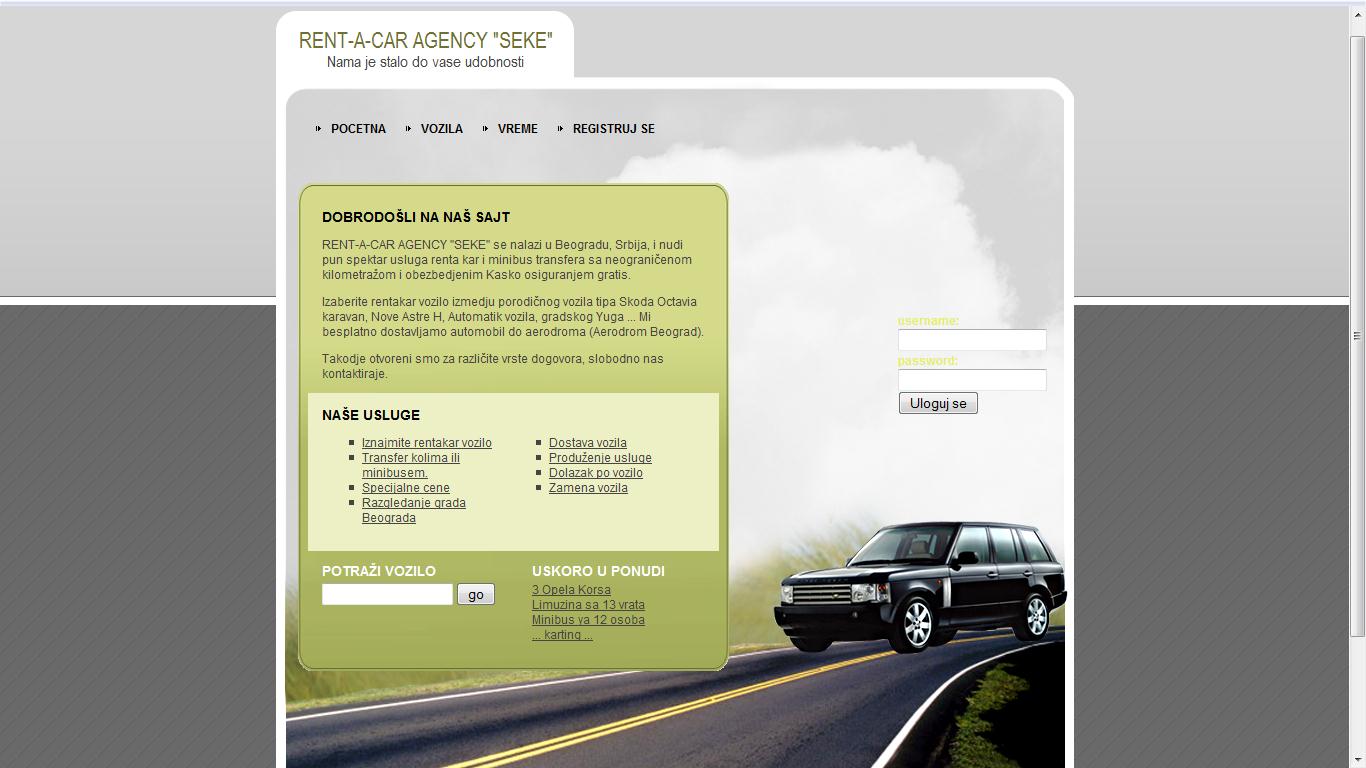
Relational Data Base Management Systems (RDBMS), odnosno sistemi za upravljanje relacionim bazama podataka su se u praksi pokazali kao najbolji način upravljanja podacima.

# Korisničko uputstvo – korisnički deo

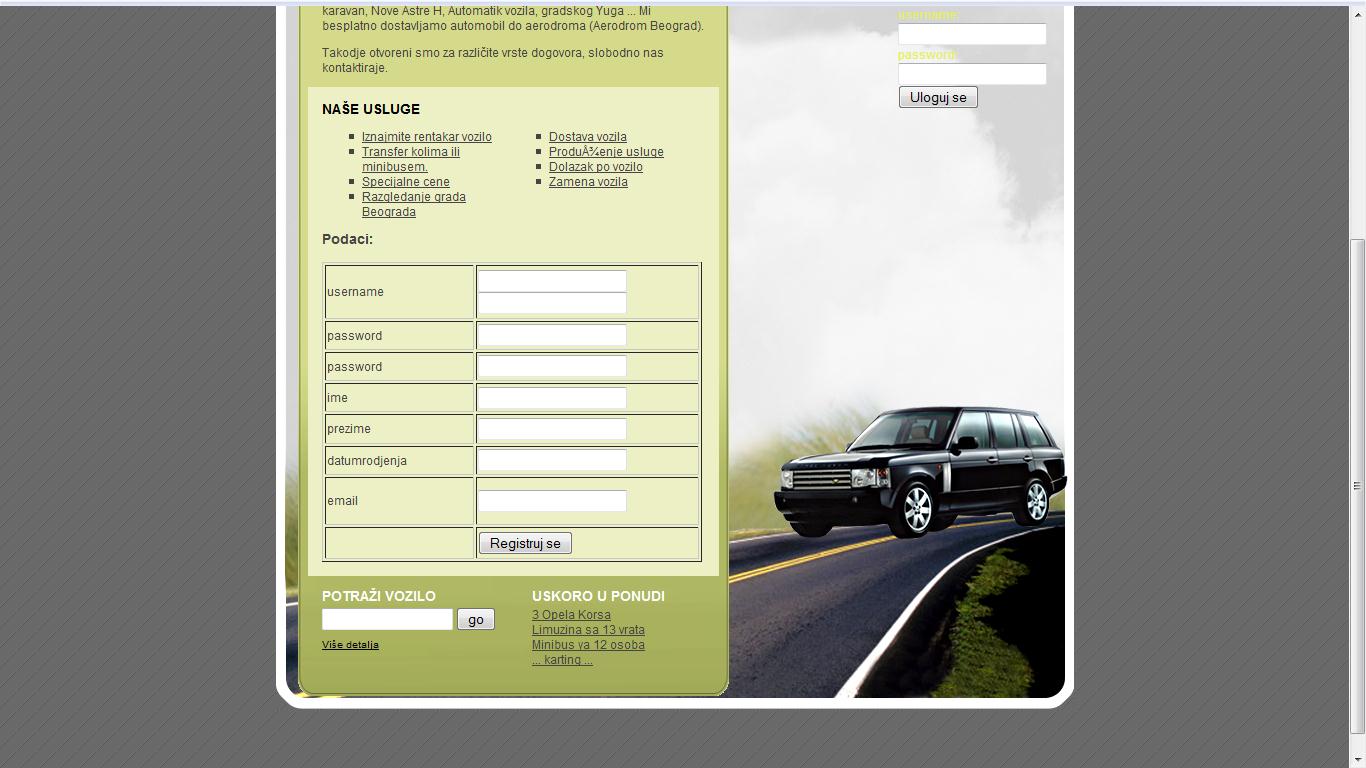
**SK1: Slučaj korišćenja – Registracija**

**Osnovni scenario:**

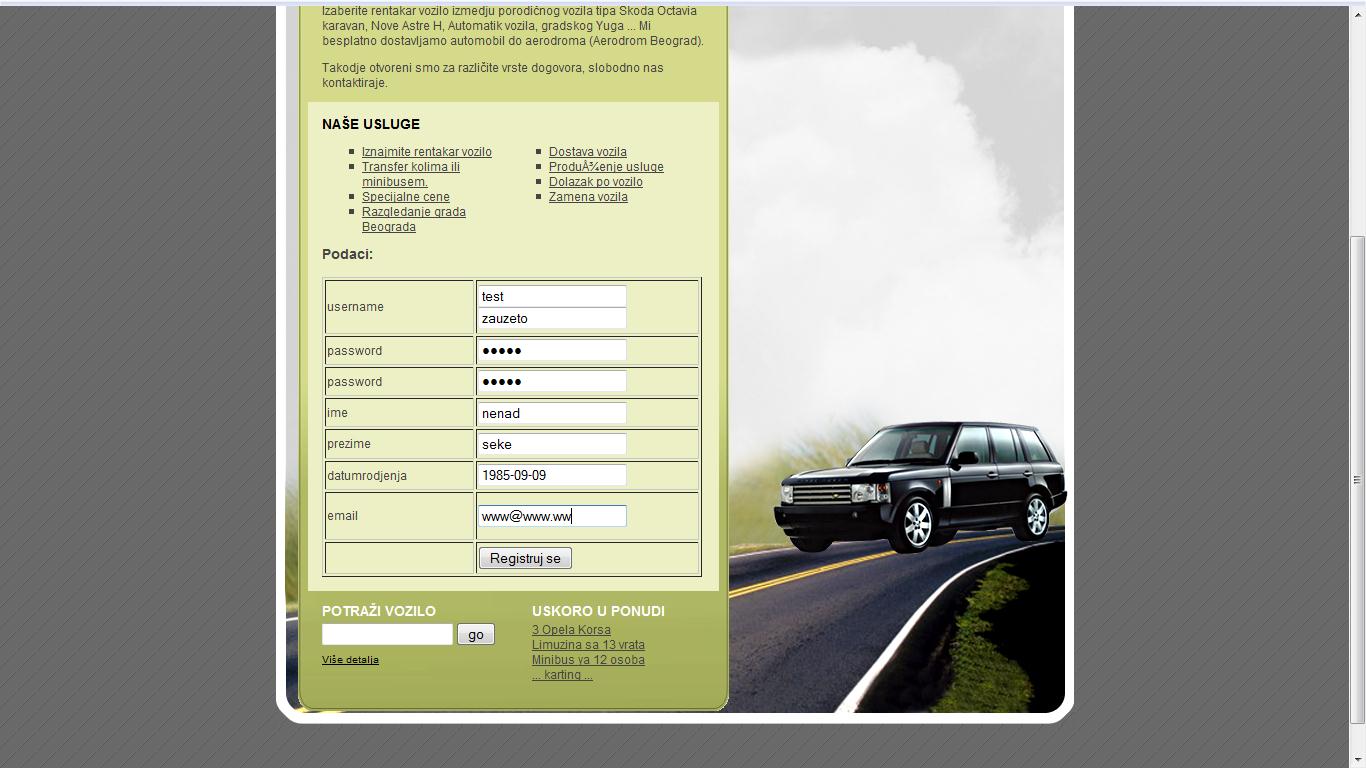
1. Korisnik na glavnoj strani pritiska na link “Registruj se”



2. Sistem prikazuje novu stranicu na kojoj korisnik treba da unese svoje podatke



3. Korisnik unosi podatke : ime , prezime i korisničko ime,password, datum rodjenja i email



4. Sistem ga obaveštava preko AJAX-a o mogućnosti korišćenja odabranog korisničkog imena

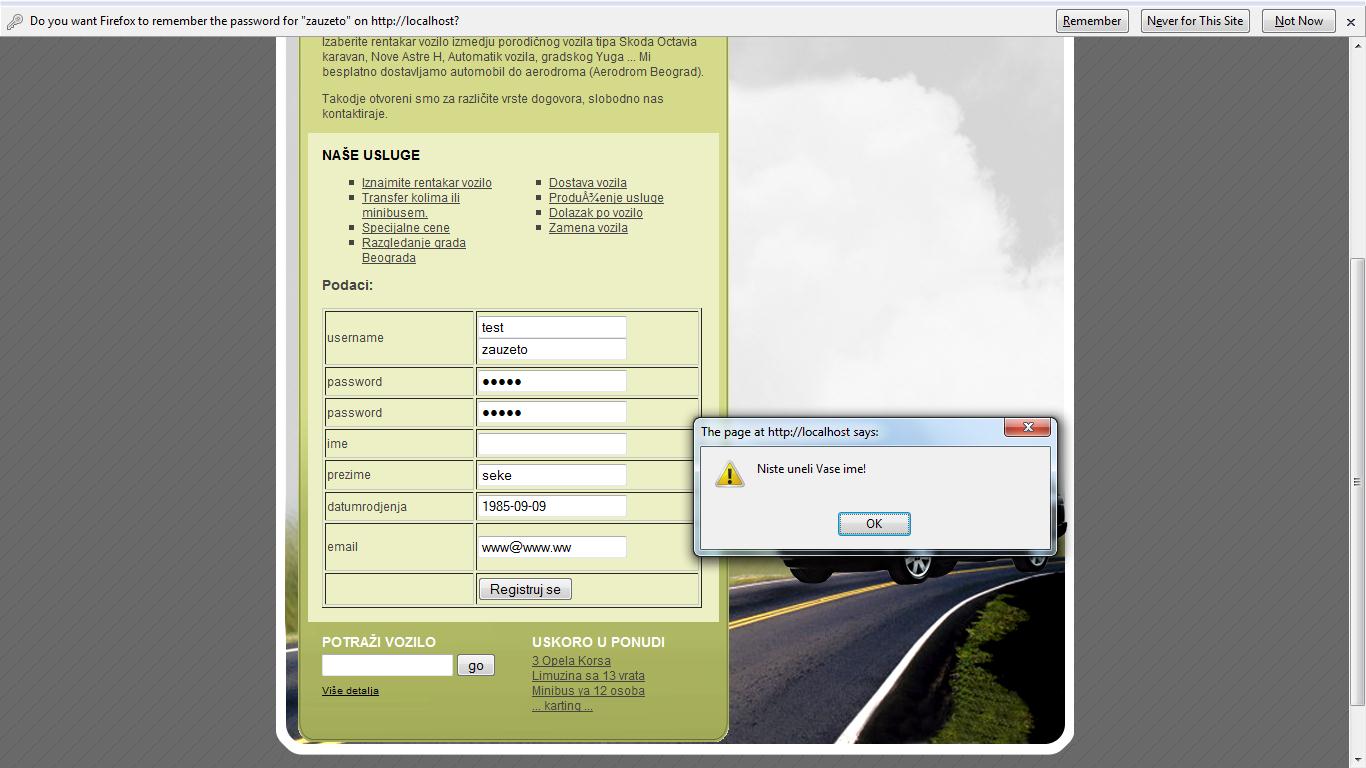
5. Korisnik pritiska dugme “Registruj se”

6. Sistem ga ubacuje u bazu podataka.

**Alternativni scenario:**

3.1. Ako je korisnik uneo Username koji se već nalazi u bazi, sistem pomoću AJAX-a odmah obaveštava o tome

5.1 U koliko korisnik nije uneo sve podatke sistem ga obaveštava o tome

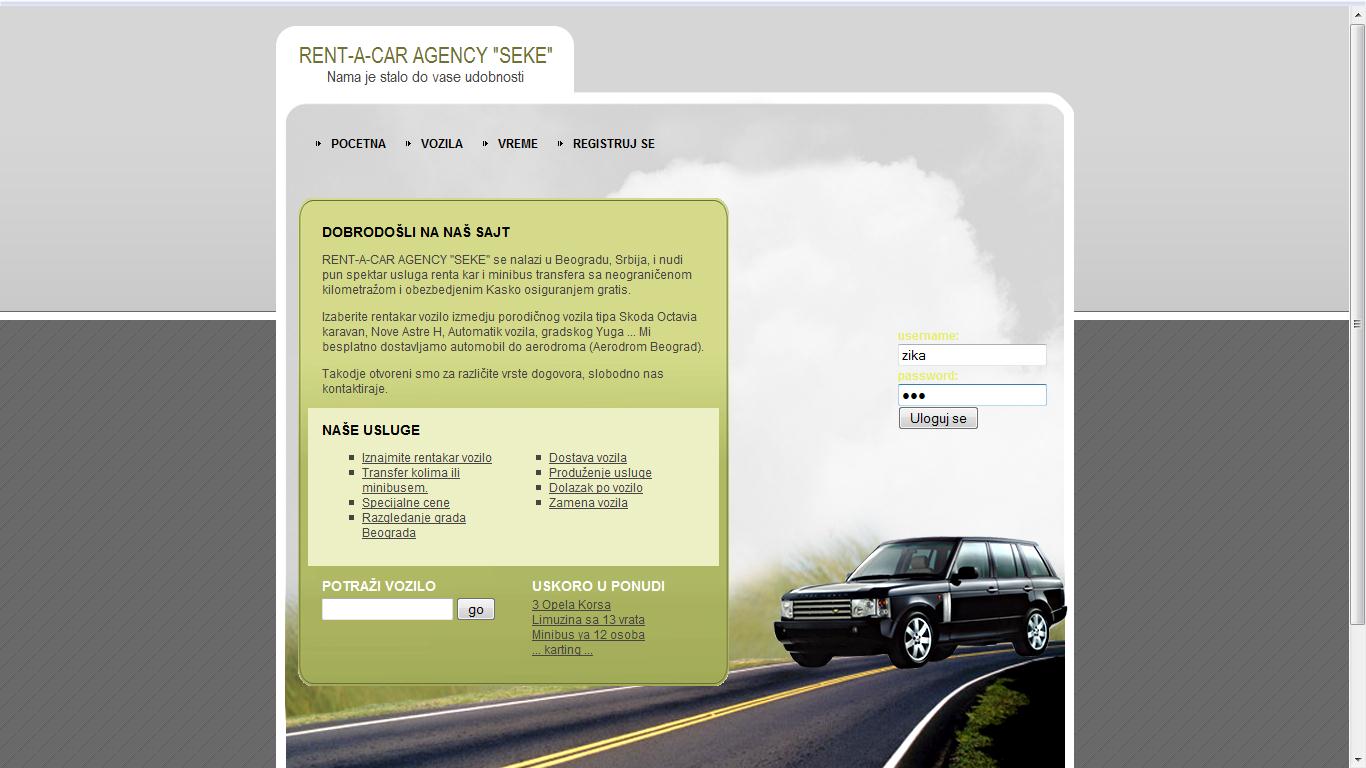


6.1 Ako sistem ne može da se poveže sa bazom, izbacuje poruku o tome

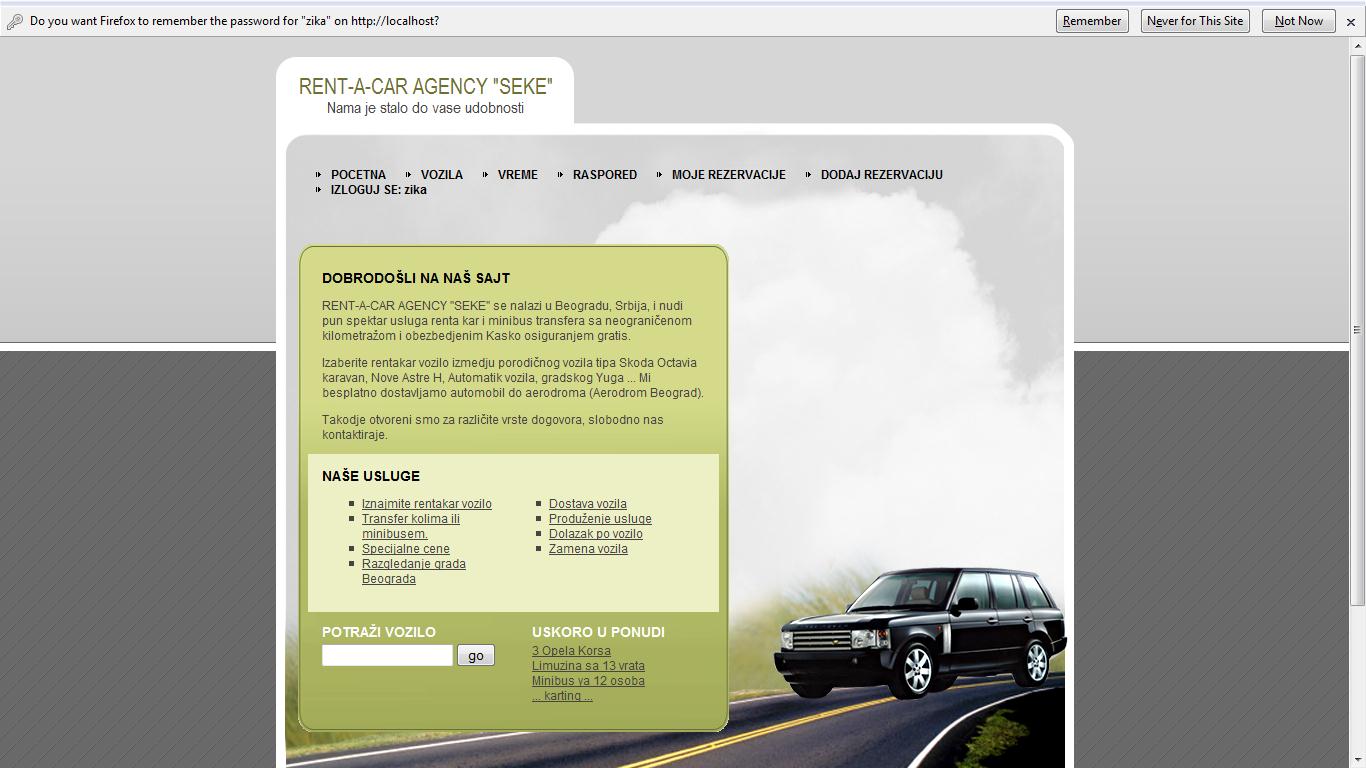
**SK2: Slučaj korišćenja – Prijava**

**Osnovni scenario:**

1. Korisnik na početnoj strani u formu za logovanje unosi korisničko ime i lozinku

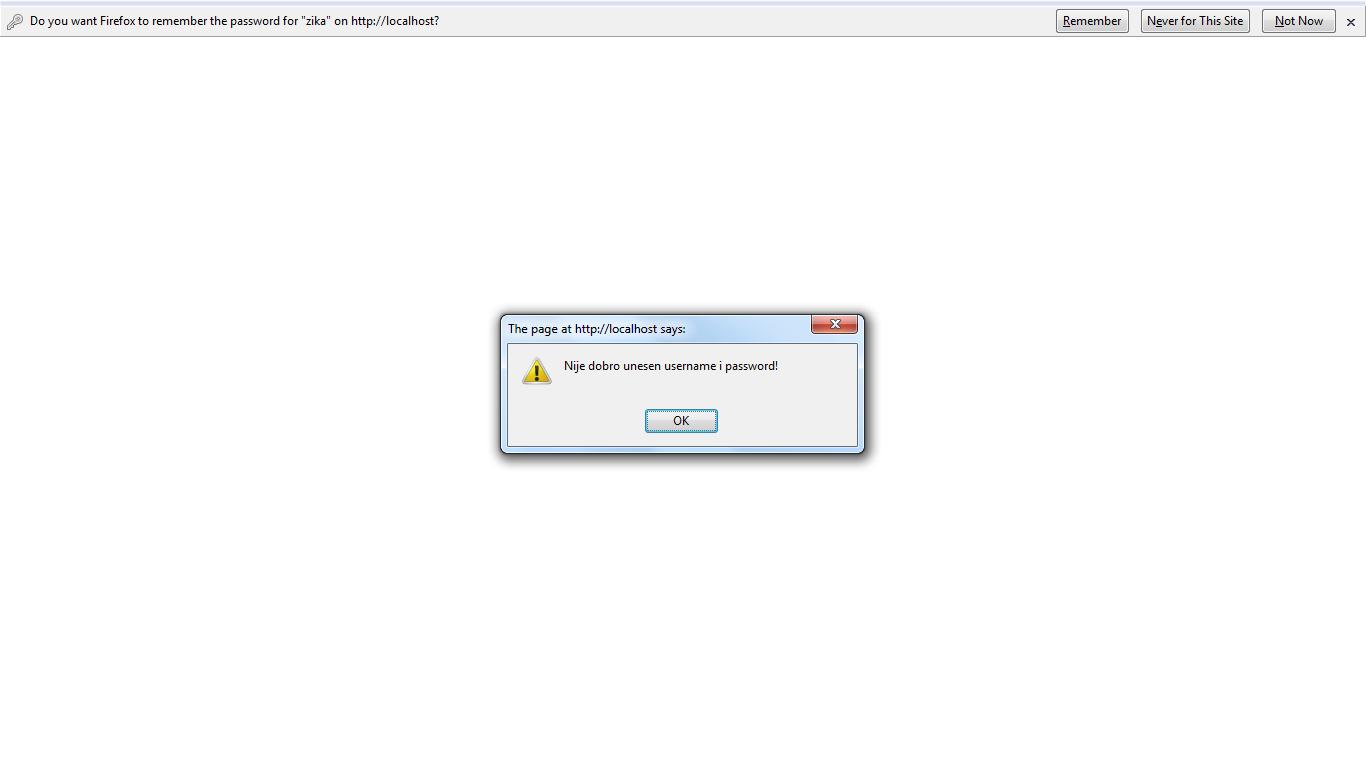


2. Sistem proverava podatke u bazi, loguje korisnika na sajt i otvara stranicu u kojoj obaveštava o uspešnoj prijavi

****

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu, izbacuje poruku o tome



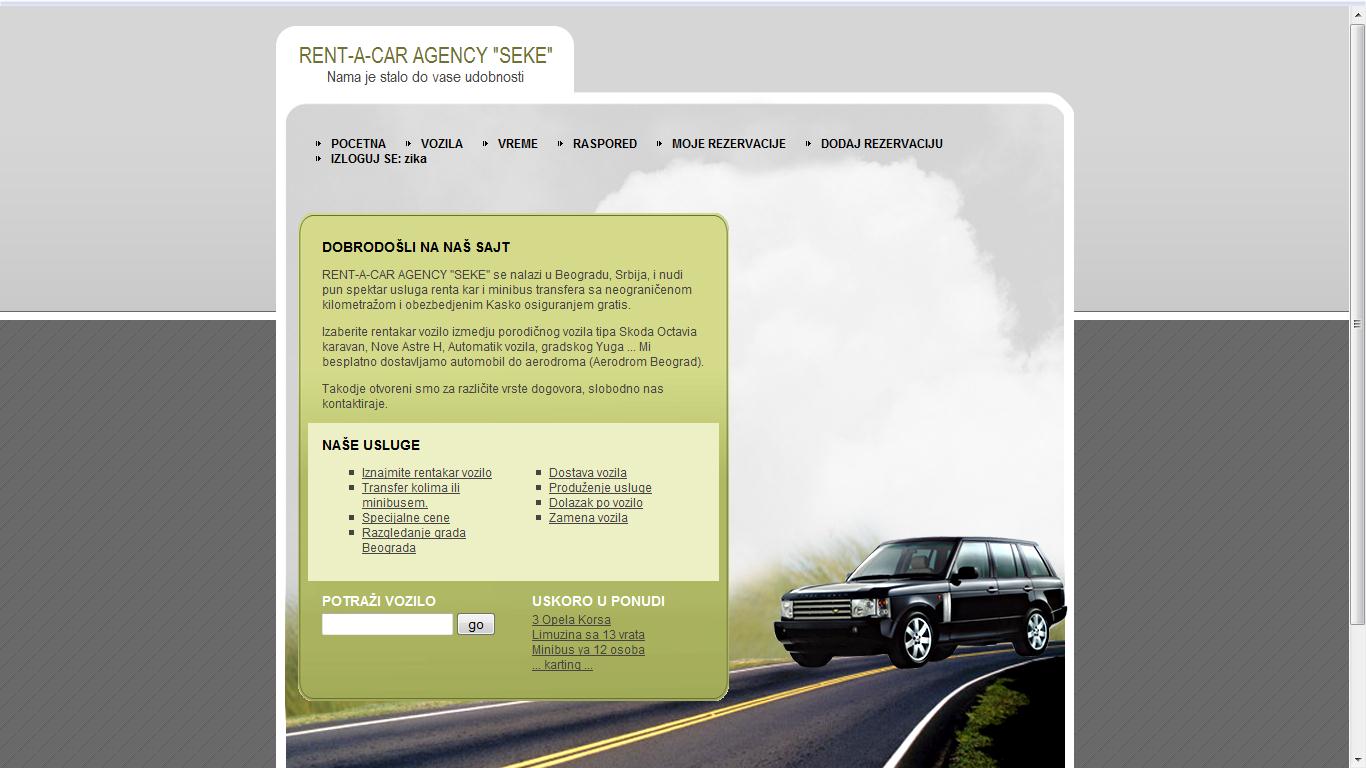
2.2 U koliko korisnuik nije uneo sve podatke uodgovarajuća polja sistem ga obaveštava o tome

2.3 Ukoliko u bazi ne postoji registrovani korisnik sa datim korisničkim imenom i lozinkom sistem obaveštava o pogrešnom unosu podataka

**SK3: Slučaj korišćenja – Odjava**

**Osnovni scenario:**

1. Korisnik u gornjem desnom uglu bira opciju “izloguj se”



2.Sistem odjavljuje korisnika, briše cockie i postavlja korsnika na početnu stranu

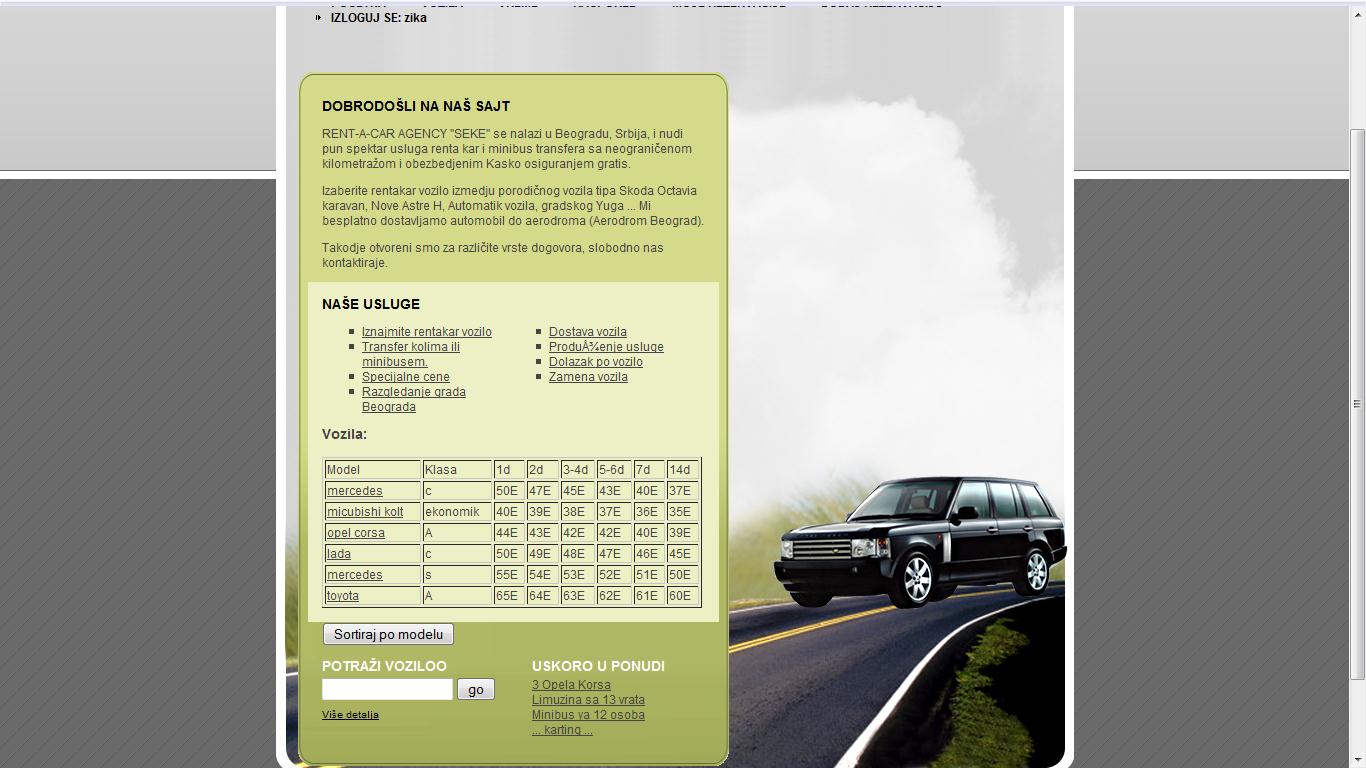


**SK4: Slučaj korišćenja – Pregled vozila**

**Osnovni scenario:**

1. Korisnik na početnoj strani bira link „vozila“ koje se nalazi u gornjem meniju

2. Sistem prikazuje vozila u tabeli sa svim dedaljima

****

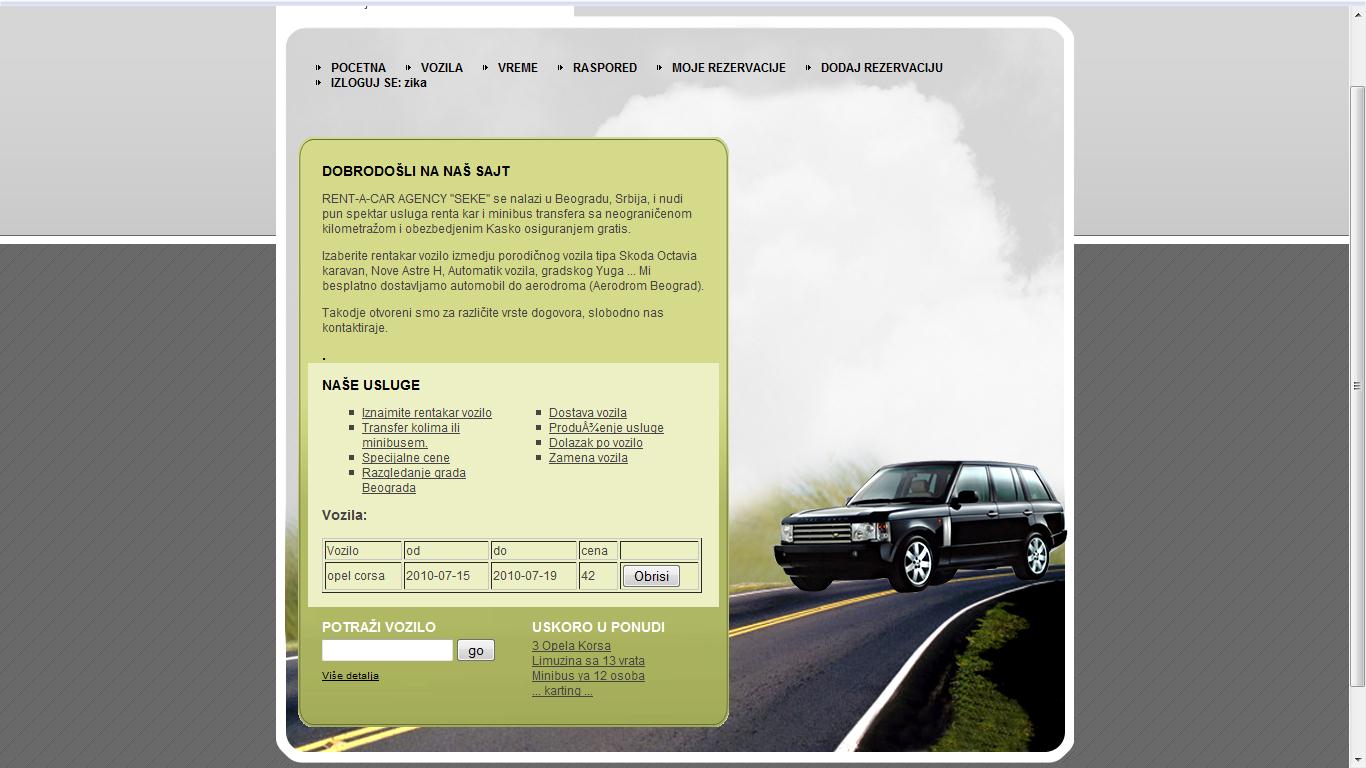
**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

**SK5: Slučaj korišćenja – Pregled rezervacija**

**Osnovni scenario:**

1. Korisnik odlazi na stranu „moje rezervacije“ odabirom linka iz gornjeg menija.
2. Sistem ucitava sve rezervacije za logovanog korisnika i prikazuje ih na strani.



**Alternativni scenario:**

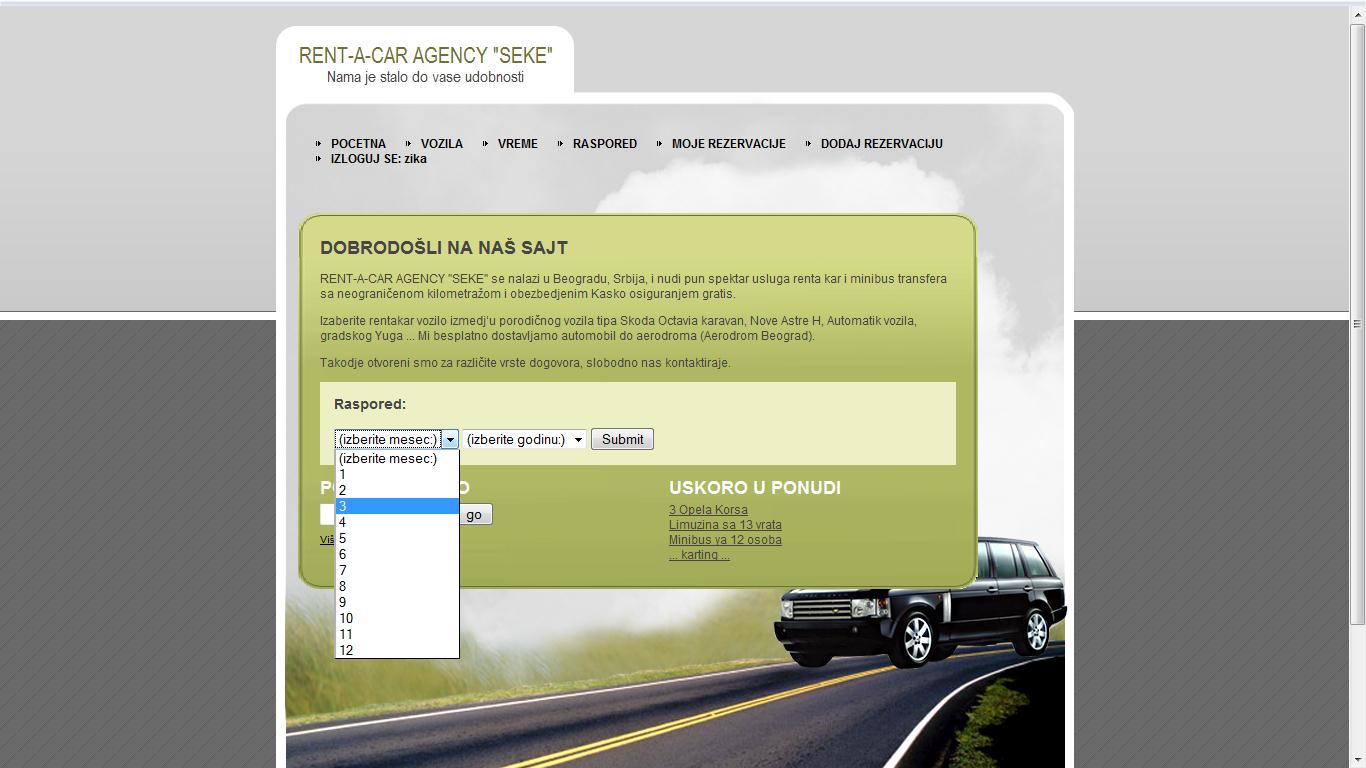
2.1 U koliko sistem ne može da se poveže na bazu i obaveštava korisnika o tome

**SK6: Slučaj korišćenja – Pregled raporeda**

**Osnovni scenario:**

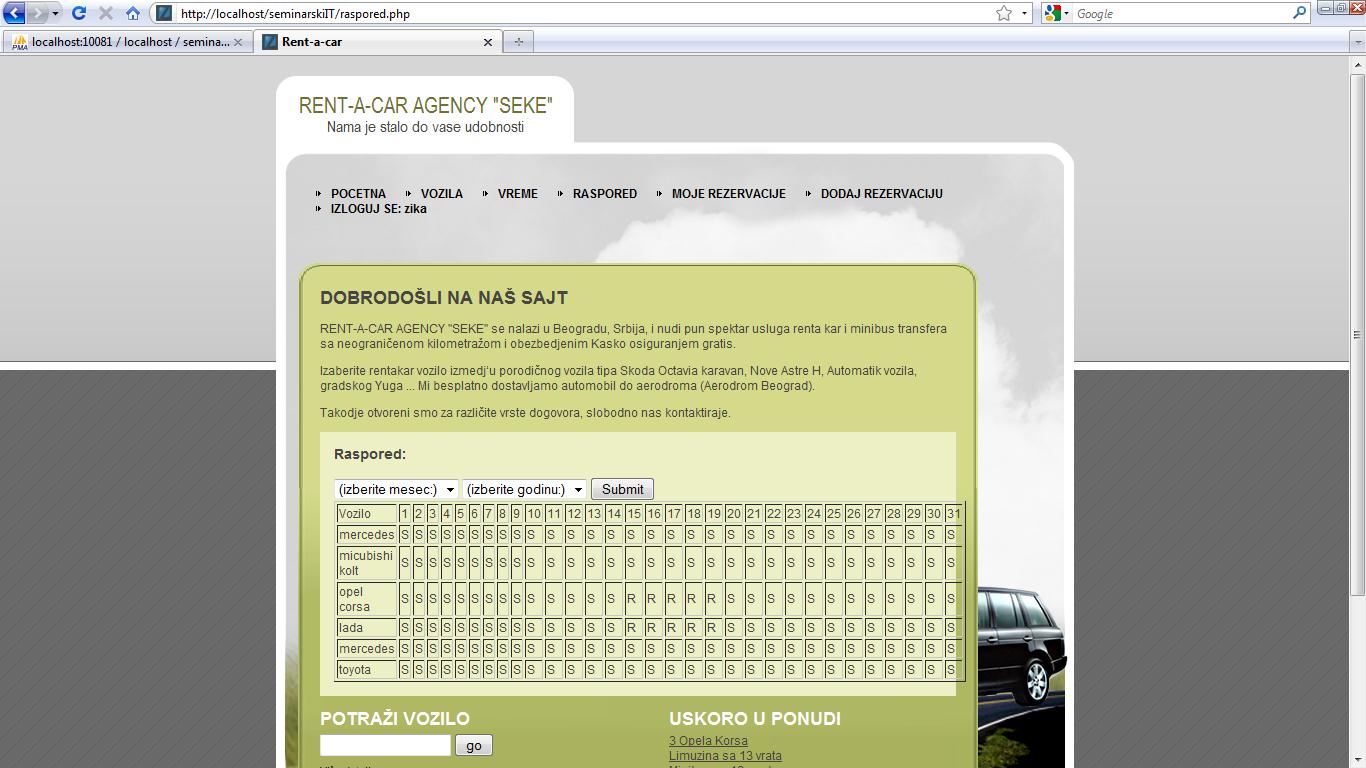
1. Korisnik na početnoj strani bira link „reaspored“ koje se nalazi u gornjem meniju

2. Korisnik odabira mesec i godinu iz kombo boksa za koje zeli da vidi raspored.



3. Korisnik poziva sistem da prikaze raspored za unete podatke.

4. Sistem prikazuje korisniku raspored.

****

**Alternativni scenario:**

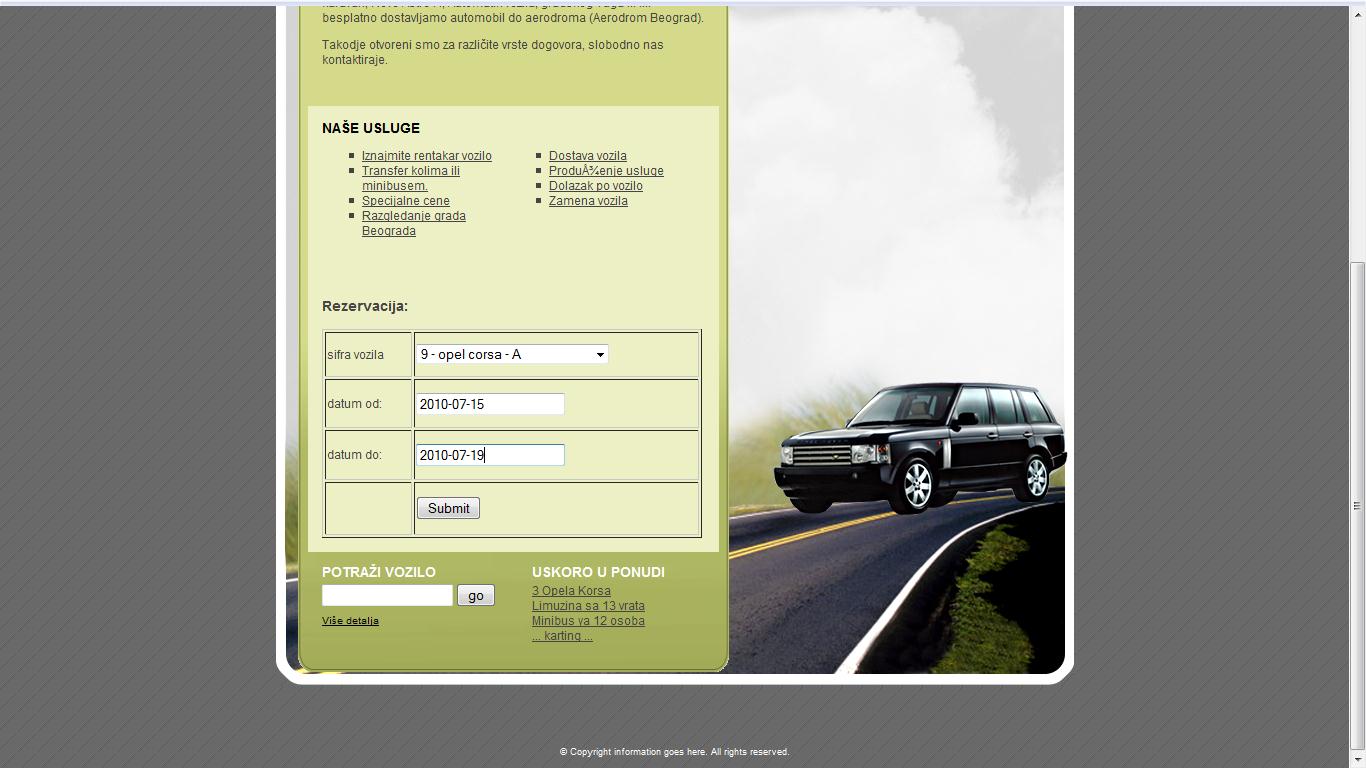
4.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

4.1 Ukoliko sistem ne može da prikaze raspored izbacuje poruku o tome

**SK7: Slučaj korišćenja – Dodavanje rezervacije**

**Osnovni scenario:**

1. Korisnik odabira vozilo iz padajuceg menija, unosi datum rezervacije.



1. Korisnik poziva sistem da zapamti rezervaciju.
2. Sistem pamti rezervaciju i obavestava korisnika o tome.

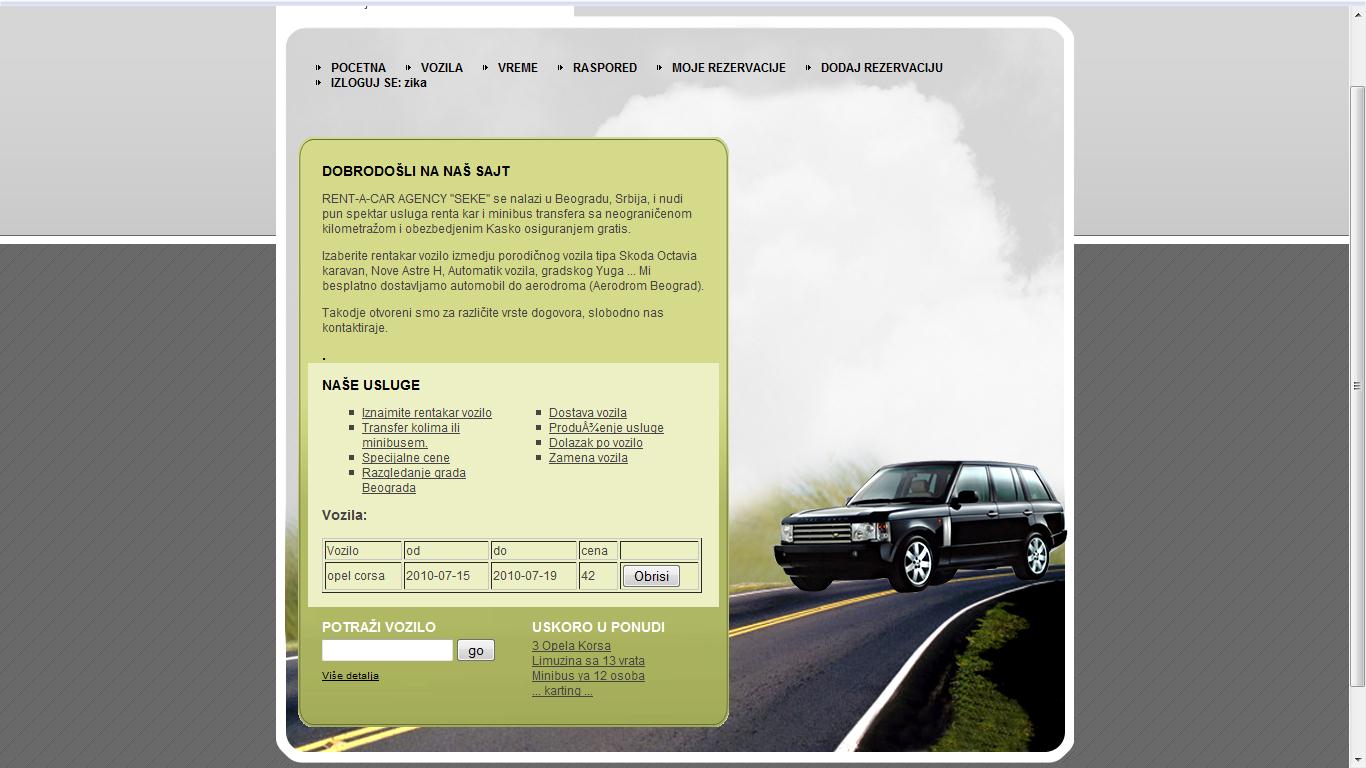
**Alternativni scenario:**

2.1 U koliko sistem ne može da se poveže na bazu i obaveštava korisnika o tome

**SK8: Slučaj korišćenja – Brisanje rezervacije**

**Osnovni scenario:**

1. Korisnik odabira rezervaciju koju zeli da obrise i poziva sistem da je obrise.



1. Sistem brise rezervaciju.

**Alternativni scenario:**

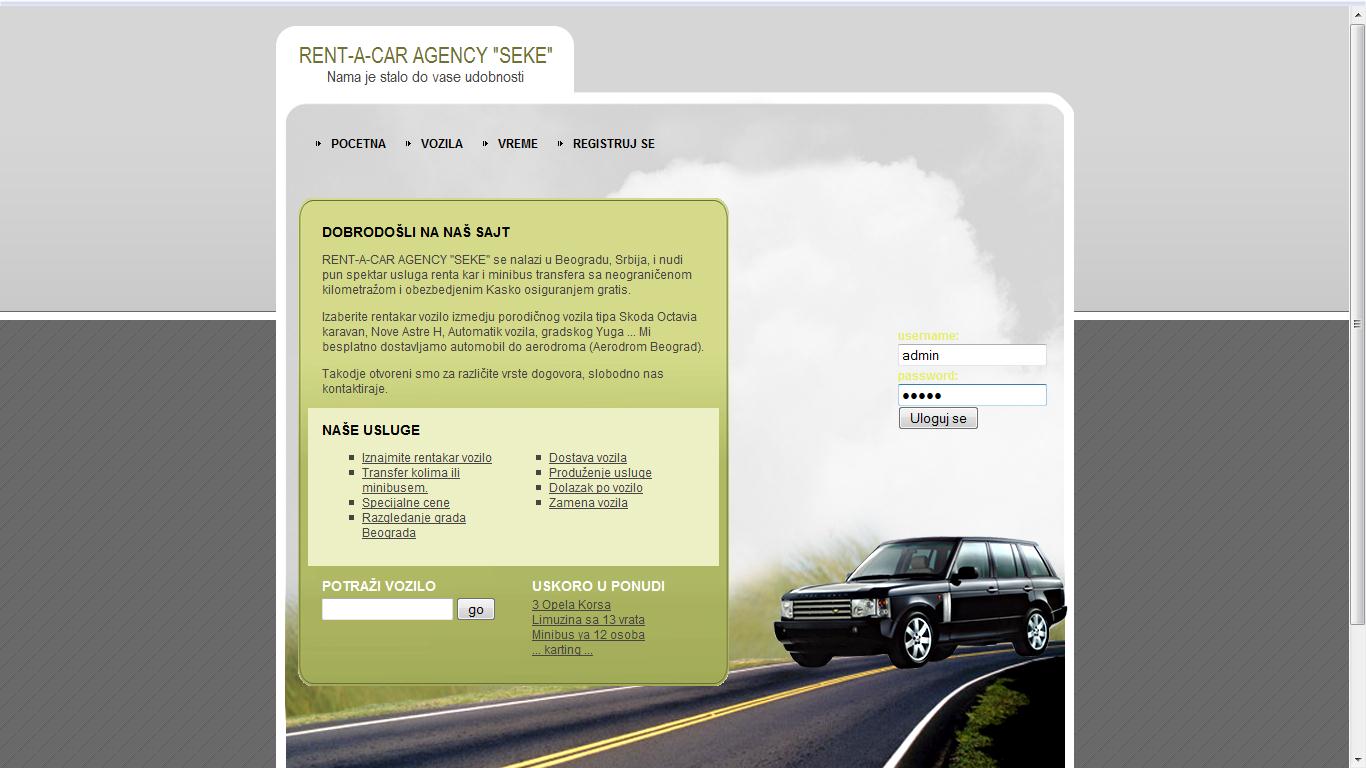
2.1 U koliko sistem ne može da obrise rezervaciju obaveštava korisnika o tome.

# Korisničko uputstvo – administratorski deo

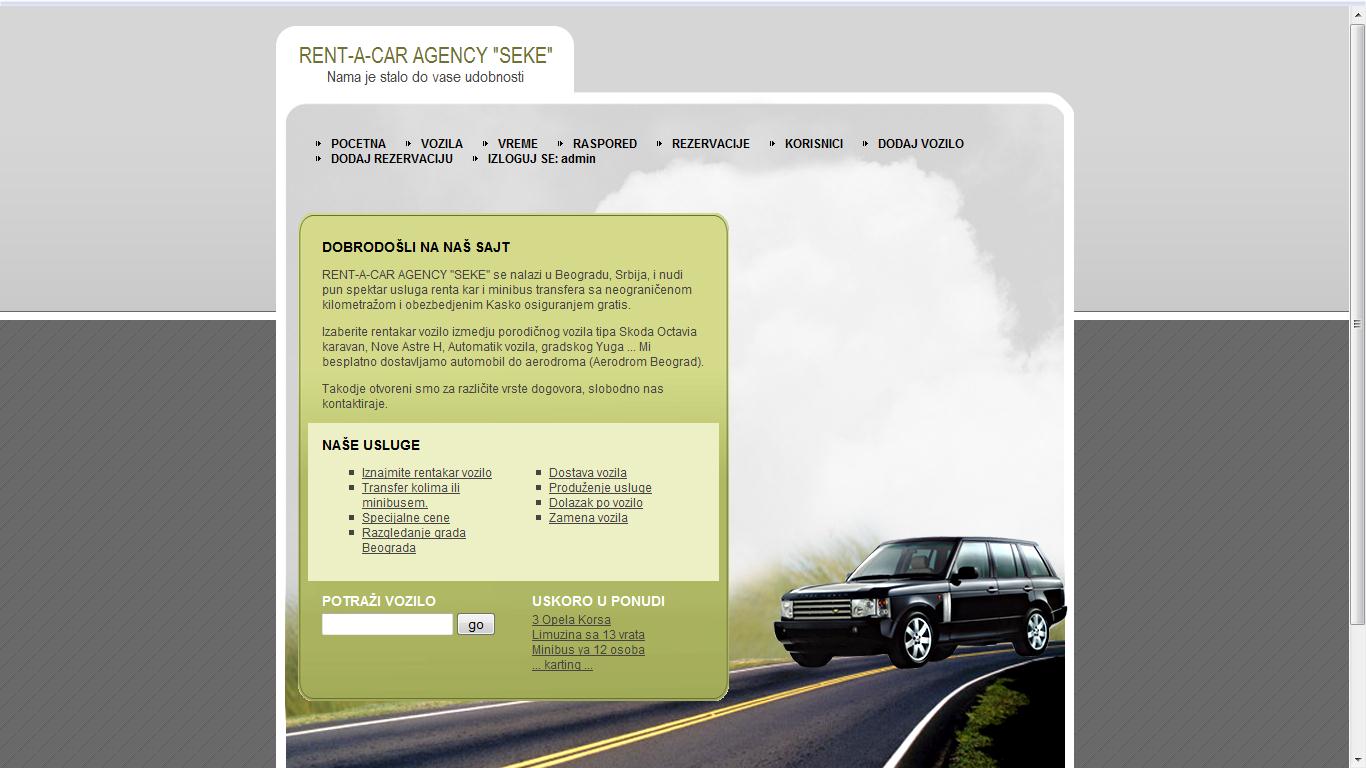
**SK1: Slučaj korišćenja – Prijava**

**Osnovni scenario:**

1. Korisnik na početnoj strani u formu za logovanje unosi korisničko ime i lozinku

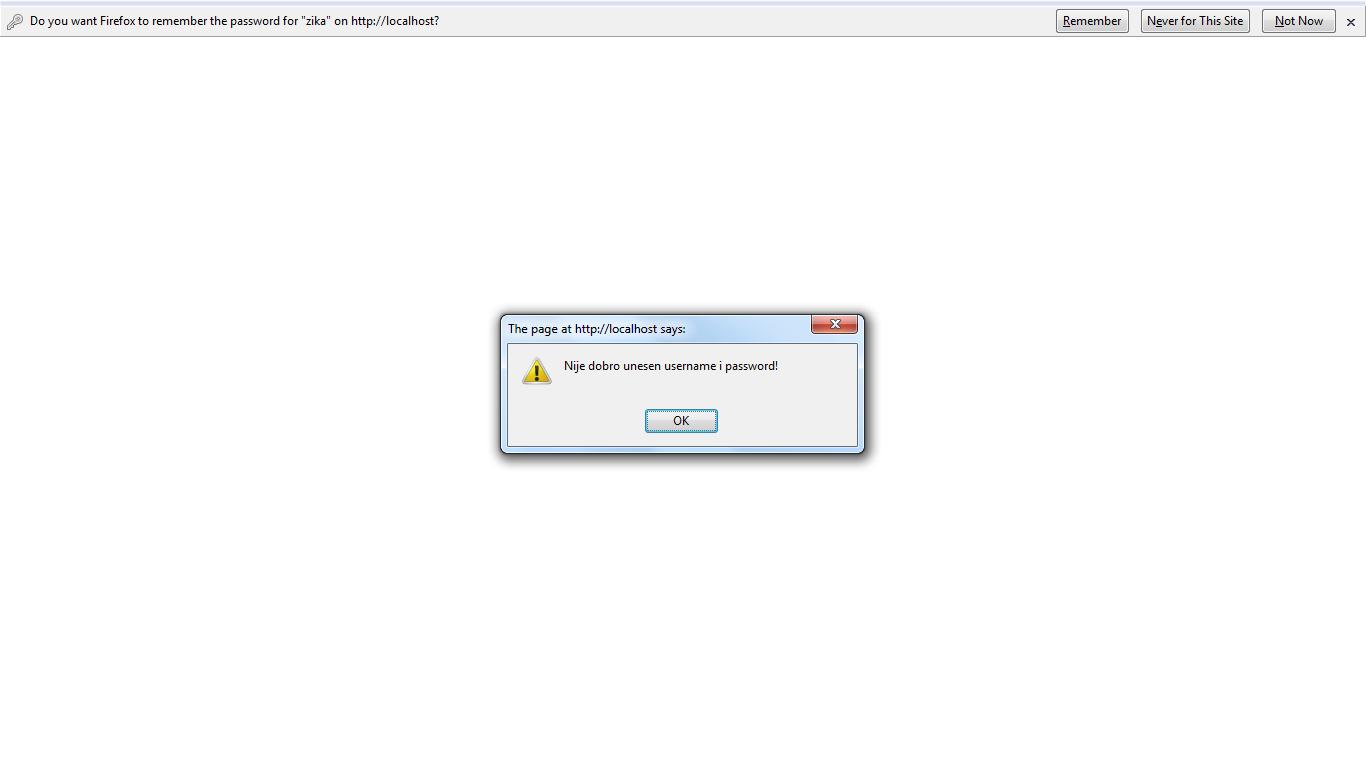


2. Sistem proverava podatke u bazi, loguje korisnika na sajt i otvara stranicu u kojoj obaveštava o uspešnoj prijavi

****

**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu, izbacuje poruku o tome



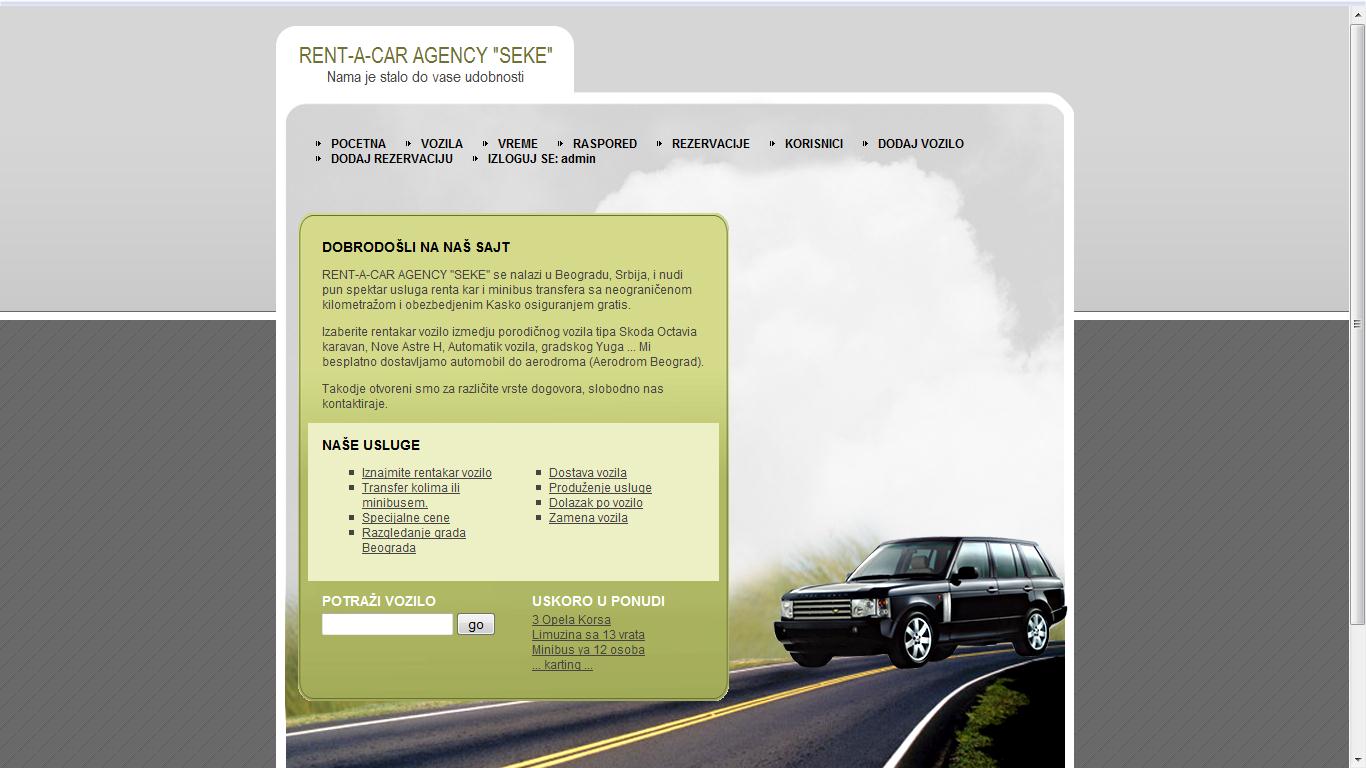
2.2 U koliko korisnuik nije uneo sve podatke uodgovarajuća polja sistem ga obaveštava o tome

2.3 Ukoliko u bazi ne postoji registrovani korisnik sa datim korisničkim imenom i lozinkom sistem obaveštava o pogrešnom unosu podataka

**SK2: Slučaj korišćenja – Odjava**

**Osnovni scenario:**

1. Korisnik u gornjem desnom uglu bira opciju “izloguj se”



2.Sistem odjavljuje korisnika, briše cockie i postavlja korsnika na početnu stranu

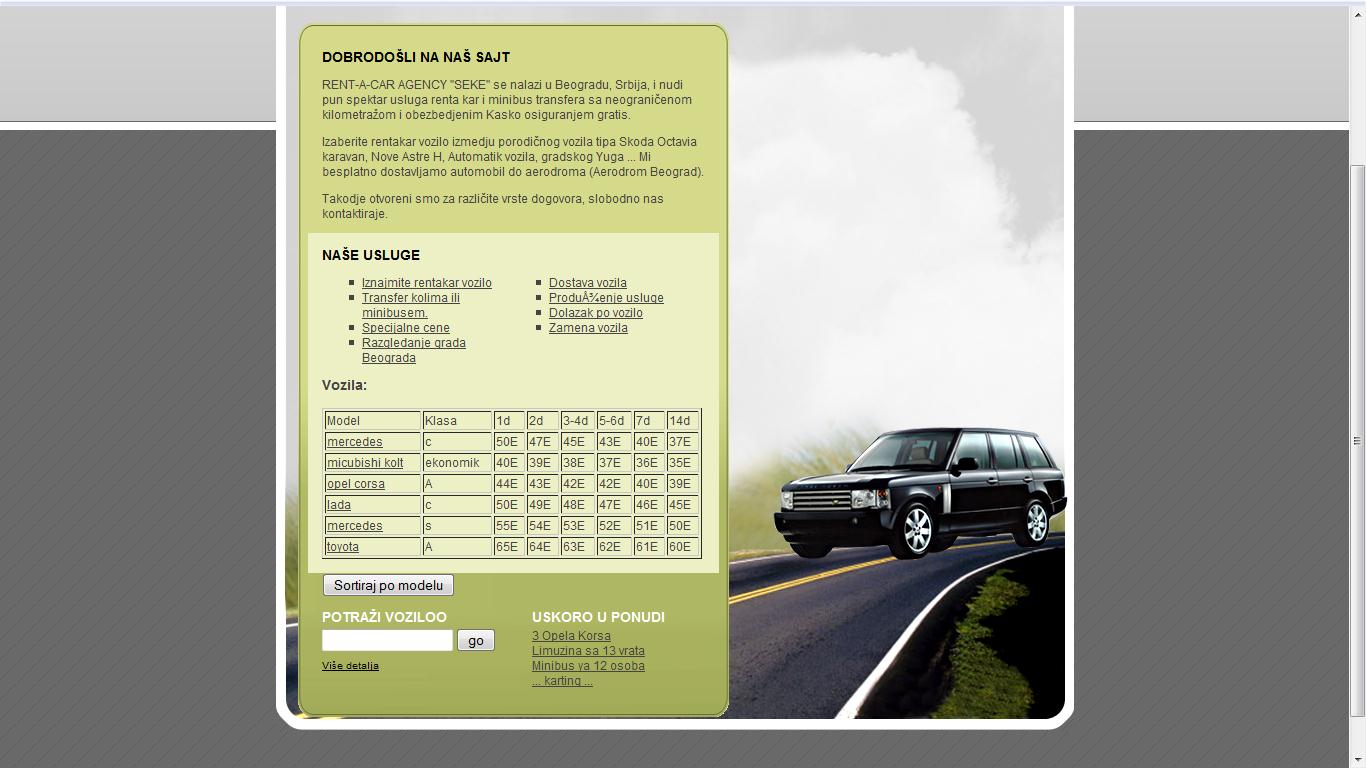


**SK3: Slučaj korišćenja – Pregled vozila**

**Osnovni scenario:**

1. Korisnik na početnoj strani bira link „vozila“ koje se nalazi u gornjem meniju

2. Sistem prikazuje vozila u tabeli sa svim dedaljima

****

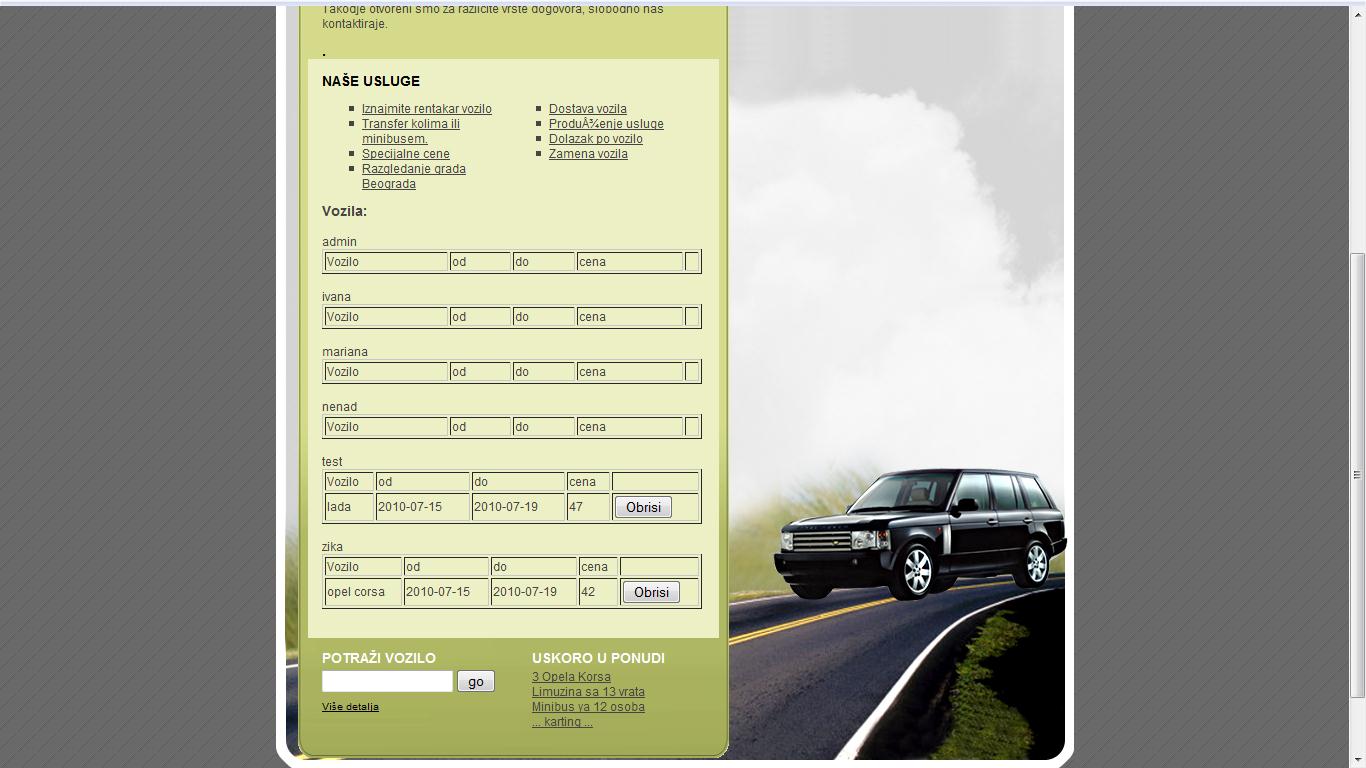
**Alternativni scenario:**

2.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

**SK4: Slučaj korišćenja – Pregled rezervacija**

**Osnovni scenario:**

1. Korisnik odlazi na stranu „moje rezervacije“ odabirom linka iz gornjeg menija.
2. Sistem ucitava sve rezervacije za logovanog korisnika i prikazuje ih na strani.



**Alternativni scenario:**

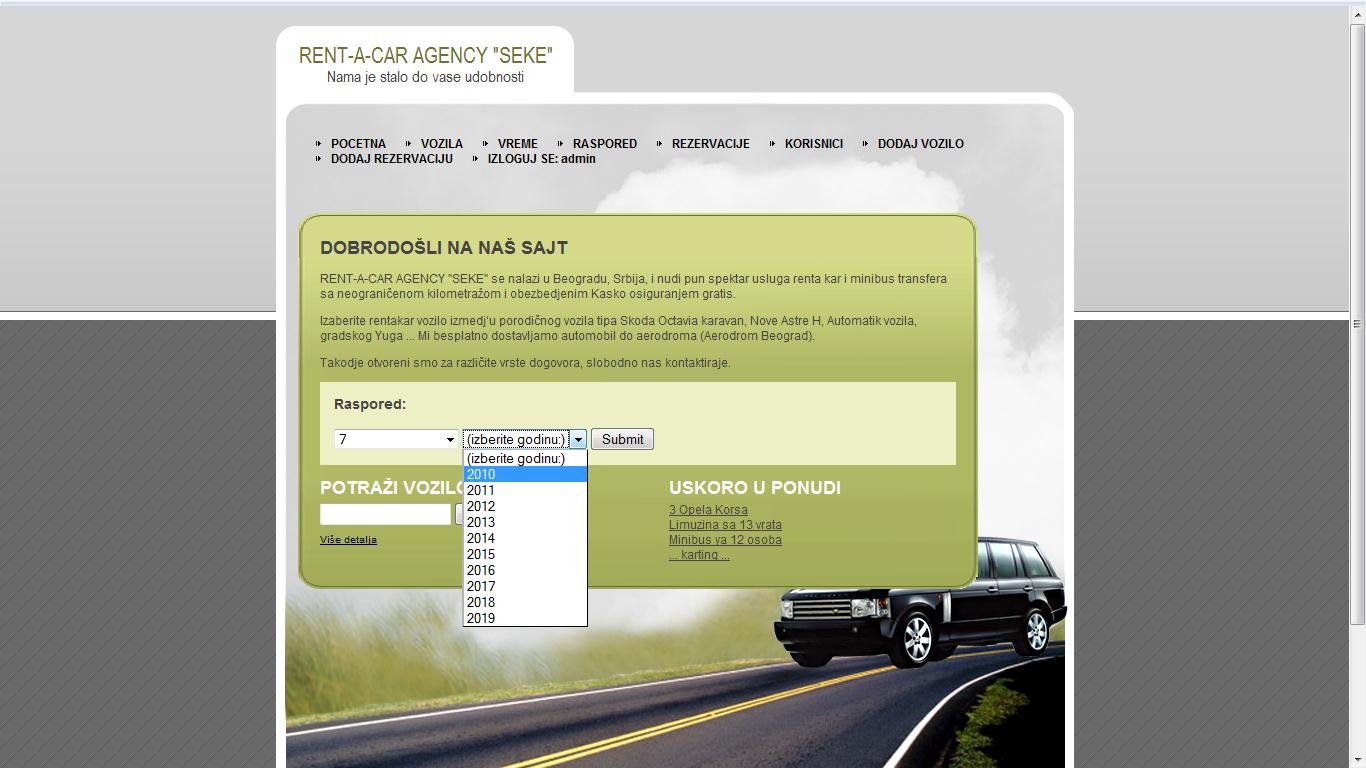
2.1 U koliko sistem ne može da se poveže na bazu i obaveštava korisnika o tome

**SK5: Slučaj korišćenja – Pregled rasporeda**

**Osnovni scenario:**

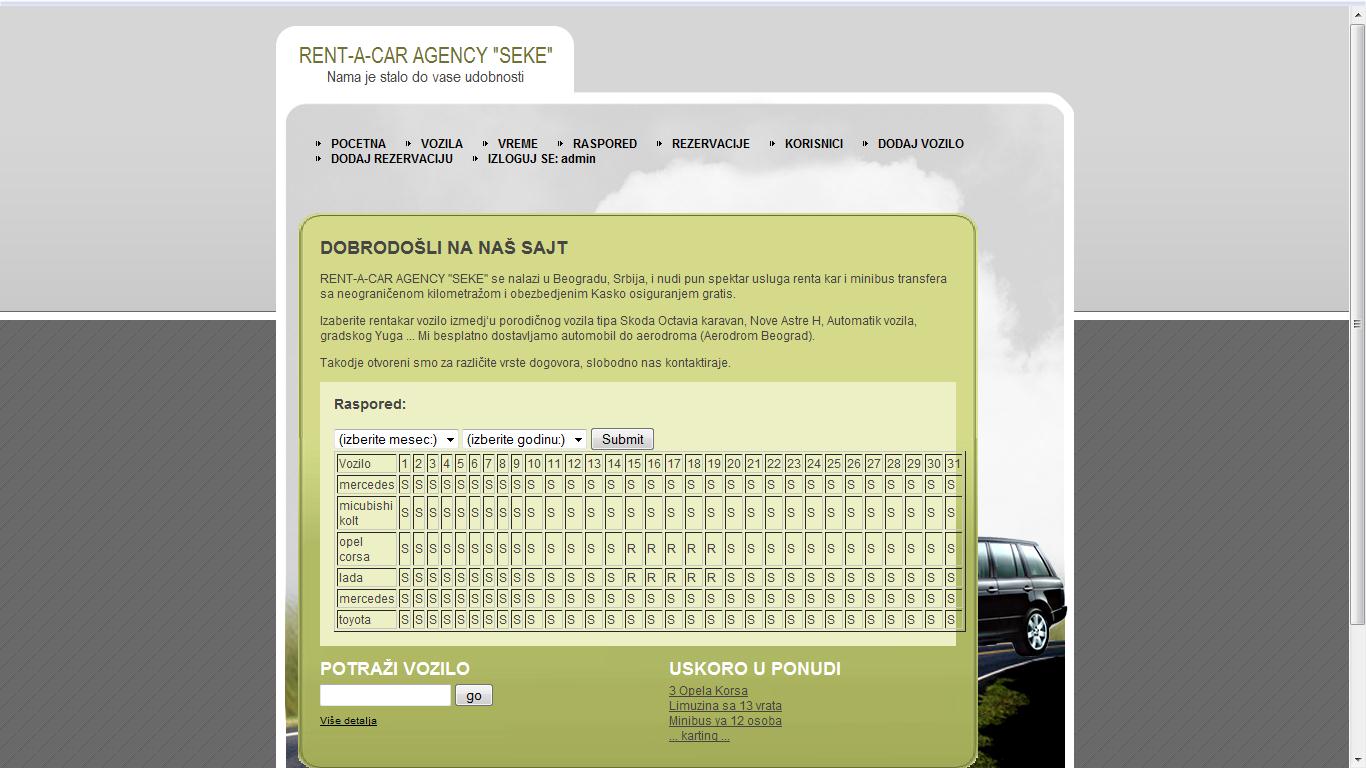
1. Korisnik na početnoj strani bira link „reaspored“ koje se nalazi u gornjem meniju

2. Korisnik odabira mesec i godinu iz kombo boksa za koje zeli da vidi raspored.



3. Korisnik poziva sistem da prikaze raspored za unete podatke.

4. Sistem prikazuje korisniku raspored.

****

**Alternativni scenario:**

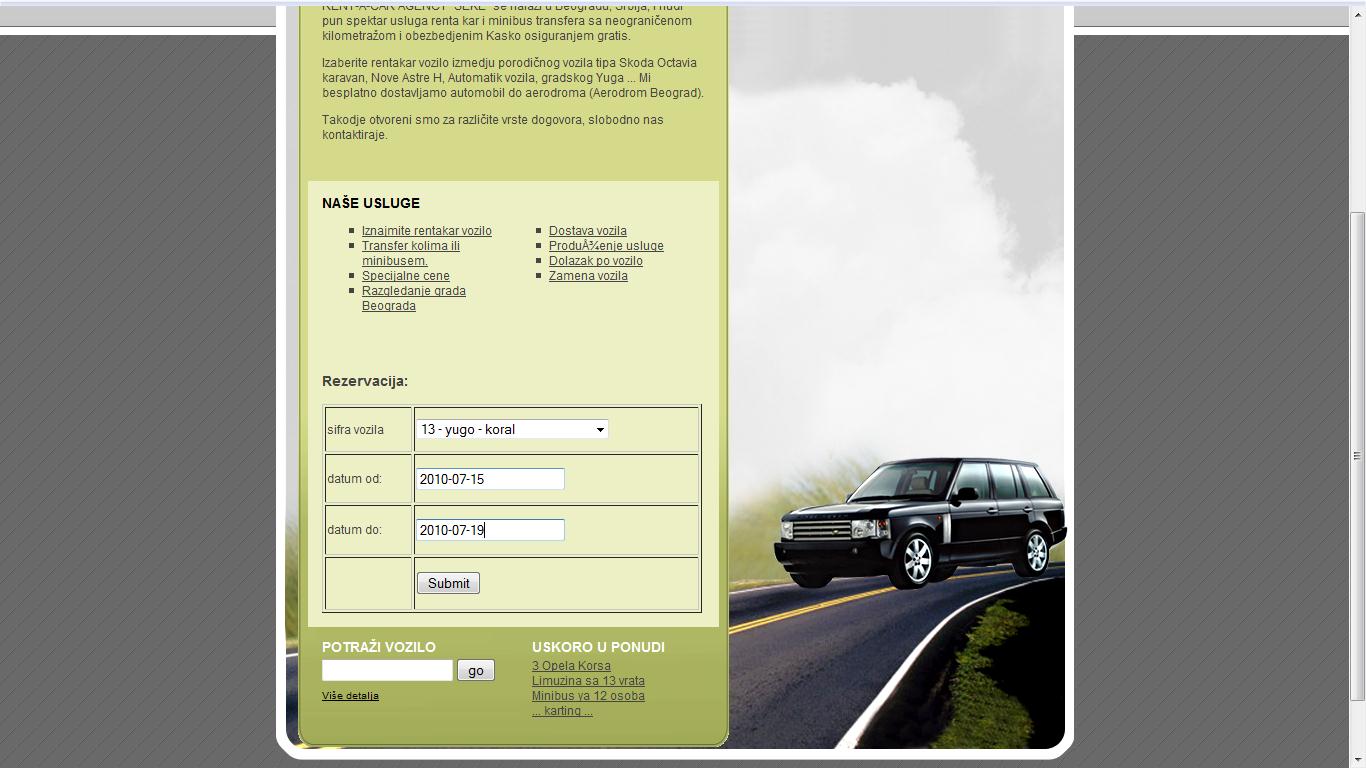
4.1 Ukoliko sistem ne može da se poveže na bazu izbacuje poruku o tome

4.1 Ukoliko sistem ne može da prikaze raspored izbacuje poruku o tome

**SK6: Slučaj korišćenja – Dodavanje rezervacije**

**Osnovni scenario:**

1. Korisnik odabira vozilo iz padajuceg menija, unosi datum rezervacije.



1. Korisnik poziva sistem da zapamti rezervaciju.
2. Sistem pamti rezervaciju i obavestava korisnika o tome.

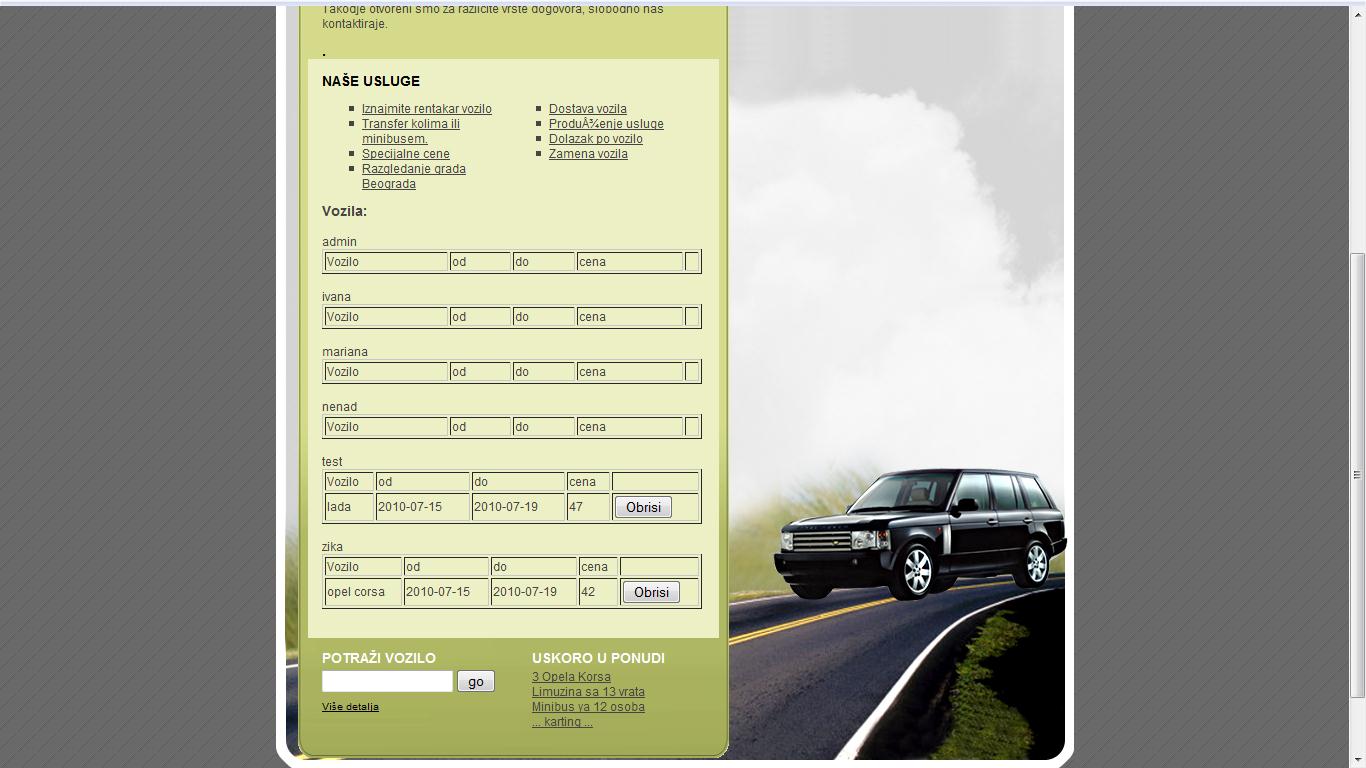
**Alternativni scenario:**

2.1 U koliko sistem ne može da se poveže na bazu i obaveštava korisnika o tome

**SK7: Slučaj korišćenja – Brisanje rezervacije**

**Osnovni scenario:**

1. Korisnik odabira rezervaciju koju zeli da obrise i poziva sistem da je obrise.



1. Sistem brise rezervaciju.

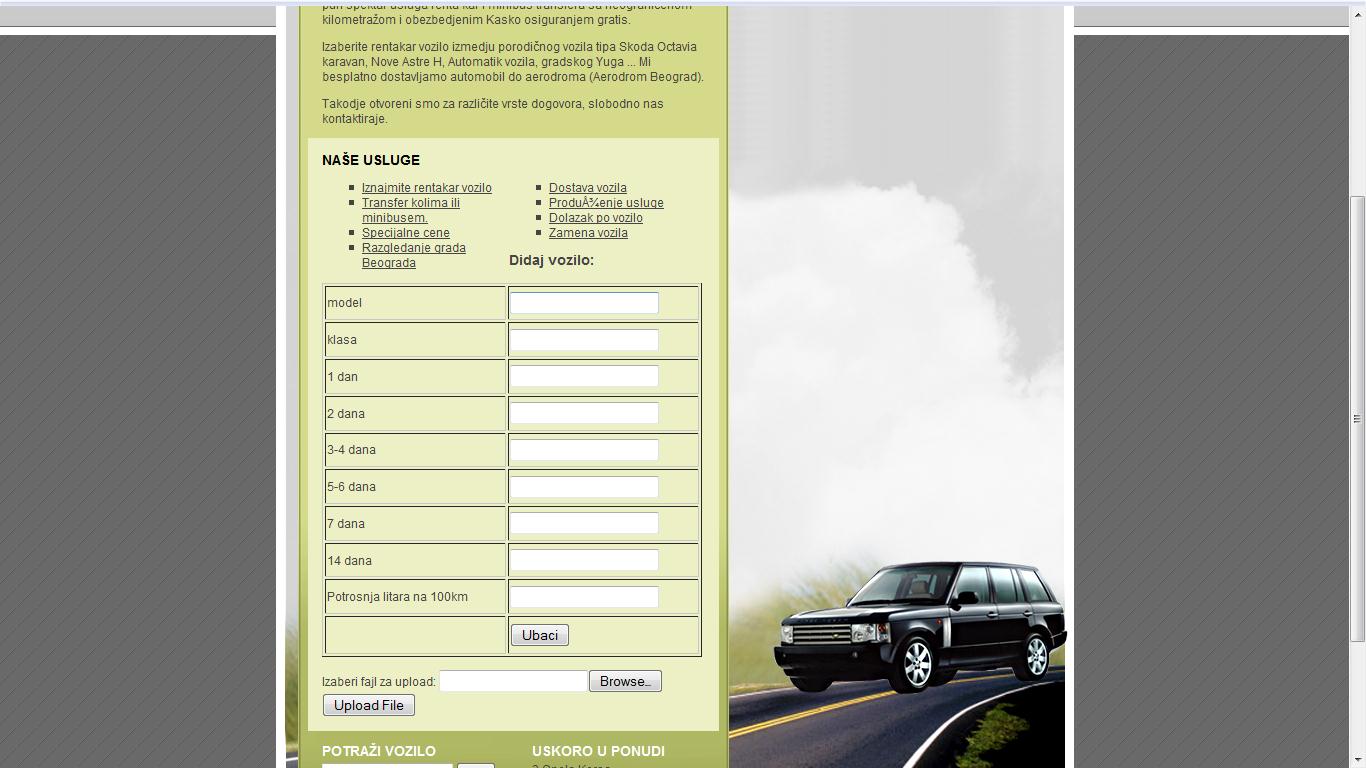
**Alternativni scenario:**

2.1 U koliko sistem ne može da obrise rezervaciju obaveštava korisnika o tome.

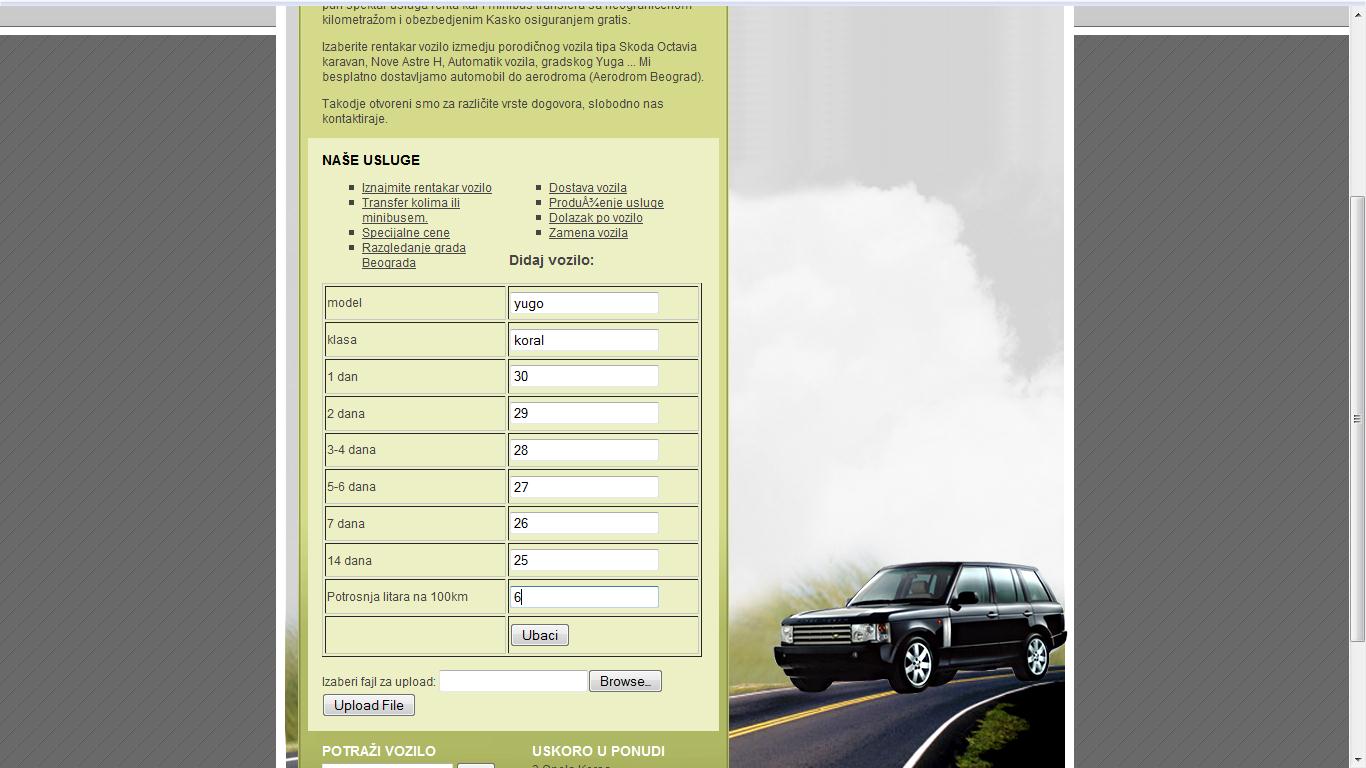
**SK8: Slučaj korišćenja – Dodavanje vozila**

**Osnovni scenario:**

1. Administrator odlazi na stranu dodaj vozilo.



1. Administrator unosi podatke o vozilu.



1. Administrator poziva sistem da zapamti vozilo.
2. Sistem pamti vozilo i obavestava administratora o tome

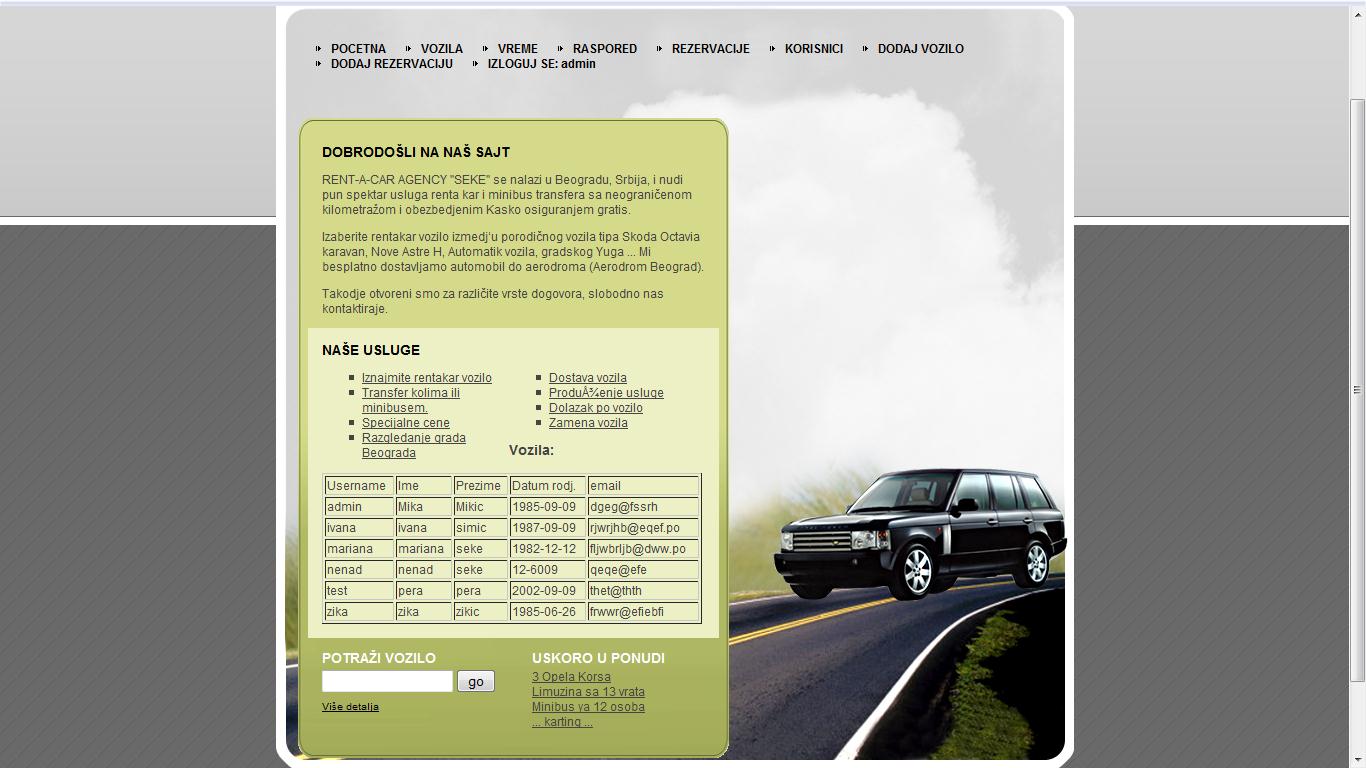
**Alternativni scenario:**

* 1. U koliko sistem ne može da zapamti vozilo obavestava administratora o tome.

**SK9: Slučaj korišćenja – Pregled korisnika**

**Osnovni scenario:**

1. Administrator odlazi na stranu korisnici.

****

1. Sistem prikazuje administratoru sve registrovane korsnike

**Alternativni scenario:**

* 1. U koliko sistem ne može da zapamti vozilo obavestava administratora o tome.

# Prikaz reprezentativnih delova koda

Klasa Databasebroker koja se nalazi u fajlu databasebroker.php sluzi za rad bazom podataka. Deo nje izlgeda:

class DatabaseBroker {

public function DatabaseBroker() {

if (! $db = mysql\_connect ( "localhost", "root", "nenad654321" )) {

die ( "<b>Spajanje na mysql server je bilo neuspesno</b>" );

}

if (! mysql\_select\_db ( "seminarski", $db )) {

die ( "<b>Greska pri odabiru baze</b>" );

}

}

public function zapamtiKorisnika(Korisnik $k) {

$username = $k->getUsername ();

$password = $k->getPassword ();

$ime = $k->getIme ();

$prezime = $k->getPrezime ();

$datumrodj = $k->getDatumrodj ();

$email = $k->getEmail ();

$tip = $k->getTip ();

$upit = "insert into korisnik values ('$username','$password','$ime','$prezime','$datumrodj','$email','$tip');";

if (mysql\_query ( $upit )) {

echo "";

} else {

echo "Korisnik vec pod tim imenom postoji u bazi<br>";

}

}

public function zapamtiRezervaciju(Rezervacija $r) {

$sifra = $r->getSifra ();

$username = $r->getUsername ();

$datumOd = $r->getDatumOd ();

$datumDo = $r->getDatumDo ();

$upit = "insert into rezervacija values ('$username','$sifra','$datumOd','$datumDo');";

echo $upit;

if (mysql\_query ( $upit )) {

echo "";

} else {

echo "Nastala je greska pri upisu rezervacije u bazu<br>" . mysql\_error ();

}

}

…

//ostale metode koje sluze za rad sa bazom

}

Primer jedne sistemske operacije:

<?php

class DodajVozilo {

public static function uradi(Vozilo $v) {

$dbbr = new DatabaseBroker ();

$dbbr->zapamtiVozilo ( $v);

unset ( $dbbr );

}

}

?>

Web servis za racunanje potrosnje goriva:

<?php

require\_once 'lib/nusoap.php';

// Pokretanje soap servera

$server = new soap\_server ();

// Konfigurisanje WSDL-a

$server->configureWSDL ( 'kilometri', 'urn:kilometri' );

$server->configureWSDL ( 'potrosnja', 'urn:potrosnja' );

// Registrovanje promenljivih koriscenih u funkcijama

$server->register ( "izracunaj", array ('kilometri' => 'xsd:double', 'potrosnja' => 'xsd:double' ), array ('return' => 'xsd:double' ), 'urn:potrosnja', 'urn:potrosnja' );

function izracunaj($kilometri, $potrosnja) {

return ($potrosnja \* $kilometri);

}

$HTTP\_RAW\_POST\_DATA = isset ( $HTTP\_RAW\_POST\_DATA ) ? $HTTP\_RAW\_POST\_DATA : '';

$server->service ( $HTTP\_RAW\_POST\_DATA );

?>

Javaskrpte za proveru unosa podataka i ajaxSkripta:

**function** ajaxFunction()

{

**var** xmlHttp;

**try**

{

// Firefox, Opera 8.0+, Safari

xmlHttp=**new** XMLHttpRequest();

}

**catch** (e)

{

// Internet Explorer

**try**

{

xmlHttp=**new** ActiveXObject("Msxml2.XMLHTTP");

}

**catch** (e)

{

**try**

{

xmlHttp=**new** ActiveXObject("Microsoft.XMLHTTP");

}

**catch** (e)

{

alert("Your browser does not support AJAX!");

**return** **false**;

}

}

}

//var zaPronaci = "zika";

**var** zaPronaci = document.getElementById("textID").value;

xmlHttp.open("GET","zauzetost.php?zaPronaci="+zaPronaci,**true**);

xmlHttp.send(**null**);

xmlHttp.onreadystatechange=**function**()

{

**if**(xmlHttp.readyState==4)

{

document.getElementById("vreme").value=xmlHttp.responseText;

}

}

}

**function** proveri() {

**if** (document.forms[0].textID.value == '') {

alert("Niste uneli username!");

document.forms[0].usernameID.focus();

**return** **false**

}

**if** (document.forms[0].passwordID.value == '') {

alert("Niste uneli sifru !");

document.forms[0].passwordID.focus();

**return** **false**;

}

**if** (document.forms[0].passwordID.value != document.forms[0].passwordID2.value) {

alert("Sifra se ne poklapa!");

document.forms[0].passwordID.focus();

**return** **false**;

}

**if** (document.forms[0].imeID.value == '') {

alert("Niste uneli Vase ime!");

document.forms[0].imeID.focus();

**return** **false**;

}

**if** (document.forms[0].prezimeID.value == '') {

alert("Niste uneli Vase prezime!");

document.forms[0].prezimeID.focus();

**return** **false**;

}

**if** (document.forms[0].datumRodjID.value == '') {

alert("Niste uneli Vas datum rodjenja!");

document.forms[0].datumRodjID.focus();

**return** **false**;

}

**if** (document.forms[0].emailID.value == '') {

alert("Niste uneli Vas email!");

document.forms[0].emailID.focus();

**return** **false**;

}

**if** (document.forms[0].emailID.value.indexOf('@') == -1) {

alert("Niste uneli e-mail adresu u odgovarajucem formatu!");

document.forms[0].emailID.focus();

**return** **false**;

}

}

Php skripta pozivana od strane ajax funkcije za proveru zauzetosti ussername-a:

<?php

error\_reporting(0);

include 'dbbr/databasebroker.php';

$username = $\_GET[zaPronaci];

$p = PostojiLiKorisnik::uradi ( $username );

$r = "";

if ($p) {

$r = "zauzeto";

}else {

$r = "slobodno";

}

echo $r;

?>

Php metode u koje se pozivaju prilikom pozivanja odredjenih akcija tokom registracije:

function proveriDugmeRegistrujSeID(){

if(isset($\_POST["registrujSeID"])){

$korisnik = new Korisnik();

if ( ctype\_alnum($\_POST[textID])){

$korisnik->setUsername($\_POST[textID]);

}else{

$poruka = "Nije dobro unesen username!";

echo "<script language=\"javascript\">alert('$poruka');</script>";

return;

}

if ( ctype\_alnum($\_POST[passwordID])){

$korisnik->setPassword($\_POST[passwordID]);

}else{

$poruka = "Nije dobro unesen password!";

echo "<script language=\"javascript\">alert('$poruka');</script>";

return;

}

if ( ctype\_alpha($\_POST[imeID])){

$korisnik->setIme($\_POST[imeID]);

}else{

$poruka = "Nije dobro uneseno ime!";

echo "<script language=\"javascript\">alert('$poruka');</script>";

return;

}

if ( ctype\_alpha($\_POST[prezimeID])){

$korisnik->setPrezime($\_POST[prezimeID]);

}else{

$poruka = "Nije dobro uneseno prezime!";

echo "<script language=\"javascript\">alert('$poruka');</script>";

return;

}

if(proverDatum($\_POST[datumRodjID])){

$korisnik->setDatumrodj($\_POST[datumRodjID]);

}else{

$poruka = "Nije dobro unesen datum!";

echo "<script language=\"javascript\">alert('$poruka');</script>";

return;

}

$email = $\_POST[emailID];

if(filter\_var($email, FILTER\_VALIDATE\_EMAIL)) {

$korisnik->setEmail($\_POST[emailID]);

}

$korisnik->setTip($\_POST[tipID]);

DodajKorisnika::uradi($korisnik);

header ( "Location: index.php" );

}

}

function proverDatum($datum){

if(empty($datum))

return false;

if (($timestamp = strtotime($datum)) === false) {

return false;

} else {

return true;

}

}

?>

# Literatura

Materijali sa myelab.net

http://php.net/index.php