

Projekat primene

e-vodič kroz Kozmetički salon

*Seminarski rad iz predmeta Internet tehnologije*

Studenti:

Katarina Lazić 1/16

Milica Bradić 1043/17

Marija Panić 358/16

Beograd,Septembar 2020

Sadržaj

[1. Korisnički zahtev 3](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652016)

[2. Opis sistema 4](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652017)

[2.1.Opis slučajeva korišćenja 4](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652018)

[2.1.1 Slučajevi korišćenja za korisnika 5](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652019)

[2.1.2 Log in 7](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652020)

[2.1.3 Log out 9](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652021)

[2.1.4Pregled ponuda 10](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652022)

[2.1.5 Zakazivanje masaže 11](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652023)

[2.2. Slučajevi korišćenja za administratora 12](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652024)

[2.2.1 Registracija administratora 12](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652025)

[2.2.1Log in. 14](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652026)

[2.2.3 Log out 15](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652027)

[2.2.4 Dodavanje nove masaže 16](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652028)

[2.2.5 Generisanje pdf-a za pregled rezervacija 17](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652030)

[2.2.6 Pregled statistike 18](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652031)

[2.3 Model podataka 19](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652032)

[2.3.1 Pmov 19](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652033)

[2.3.2 Dijagram klasa 20](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652034)

[2.4. Specifikacija REST API-ja 21](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652035)

[3.Opis tehnologija korišćenih u aplikaciji 26](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652036)

[3.1.PHP 26](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652037)

[3.2.AJAX(Asynchronous JavaScript And XML) 27](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652038)

[3.3.JSON 28](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652039)

[3.4.JavaScript 28](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652040)

[3.5.Jquery 29](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652041)

[4.Korisničko uputstvo 31](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652042)

[4.1.Korisnik 31](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652043)

[4.2.Admin 33](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652044)

[5.Prikaz reprezentativnih delova koda 35](file:///C:\xampp\htdocs\masazaProjekat\Dokumentacija.docx#_Toc506652045)

# Korisnički zahtev

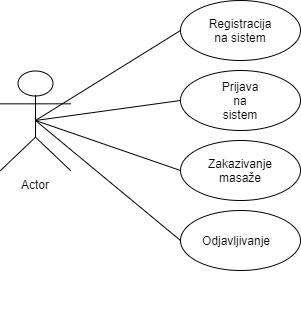
Potrebno je projektovati i implementirati veb aplikaciju za kozmetički salon. Sistem treba da prepoznaje dve vrste korisnika: administratora sistema i korisnika sistema sa običnim privilegijama. Za korisnika sistema je potrebno omogućiti registraciju na sajt, logovanje i zakazivanje termina. Ako korisnik nije registrovan, nije u mogućnosti da zakaže masažu. Administrator sistema ima pristup administratorskom panelu i uvidu u zakazane masaže. Administratorski panel se sastoji iz dela koji se odnosi na brisanje ili izmenu vrsta masaža, zatim je administratoru omogućen pregled svih korisnika koji su zakazali termin iz baze podataka, dalje je omogućena pretraga korisnika sa najvećim brojem uzetih masaža, mapa gde se salon nalazi, kao i kačenje pdf dokumenata na sajt.

# Opis sistema

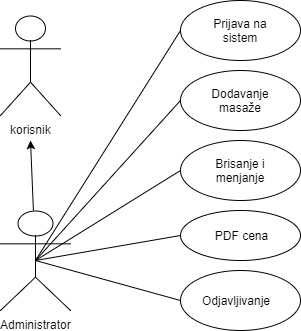
## 2.1.Opis slučajeva korišćenja

Karakteristični slučaji koriščenja:

1. Registracija korisnika
2. Prijava (login) korisnika/admina
3. Izbor masaže
4. Zakazivanje masaže
5. Izmena osnovne cene masaže
6. Brisanje masaže
7. Pretraga korisnika sa najvećim brojem uzetih masaža
8. Pregled pdf-a
9. Odjavljivanje korisnika/administratora



Dijagram slučajeva korišćenja 1 Korisnik



Dijagram slučajeva korišćenja 2 Administrator

## 2.1.1 Slučajevi korišćenja za korisnika

**Naziv SK:** Registracija korisnika

**Svrha:** Registracija korisnika na sajt bioskopa

**Akteri SK:** Korisnik

**Preduslov**:/

***Osnovni scenario:***

1. Korisnik otvara odgovarajuću formu za registraciju.(APSO)
2. Sistem vraća formu za registraciju.(IA)
3. Nakon unosa podataka korisnik pritiska dugme “Registruj se.“(APSO)
4. Sistem prihvata podatke, proverava ih i šalje poruku „Uspešno ste se registrovali“.(IA)

**Dijagram sekvenci:**



**Alternativni scenario:**

* 1. Sistem javlja da uneto korisničko korisničko ili email ime već postoji u bazi porukom “Korisnik sa ovim korisničkim imenom već postoji" ili "Korisnik sa ovim emailom već postoji".

**Dijagram sekvenci:**



## 2.1.2 Log in

**Naziv SK:** Log in

**Svrha:** Logovanje korisnika u cilju pregleda filmova u bioskopu i njihove rezervacije

**Akteri SK:** Korisnik

**Osnovni scenario:**

1. Korisnik poziva sistem da otvori formu za Log in (APSO)
2. Sistem otvara formu za log in. (IA)
3. Korisnik unosi korisnicko ime i sifru i potvrđuje unos.(APSO)
4. Sistem prosleđuje korisnika na odgovarajucu stranicu – stranica dobrodošlice.(IA)

**Dijagram sekvenci:**



**Alternativni scenario:**

4.1. Pogrešno uneti korisničko ime ili šifra, sistem obaveštava korisnika o grešci porukom :“Pogresno korisničiko ime ili šifru. “(IA)

4.2. Ukoliko sistem ne može da se poveže na bazu obaveštava korisnika o tome.(IA)

**Dijagram sekvenci:**



### 2.1.3 Log out

**Naziv SK:** Log out

**Svrha:** Odjava korisnika sa sistema

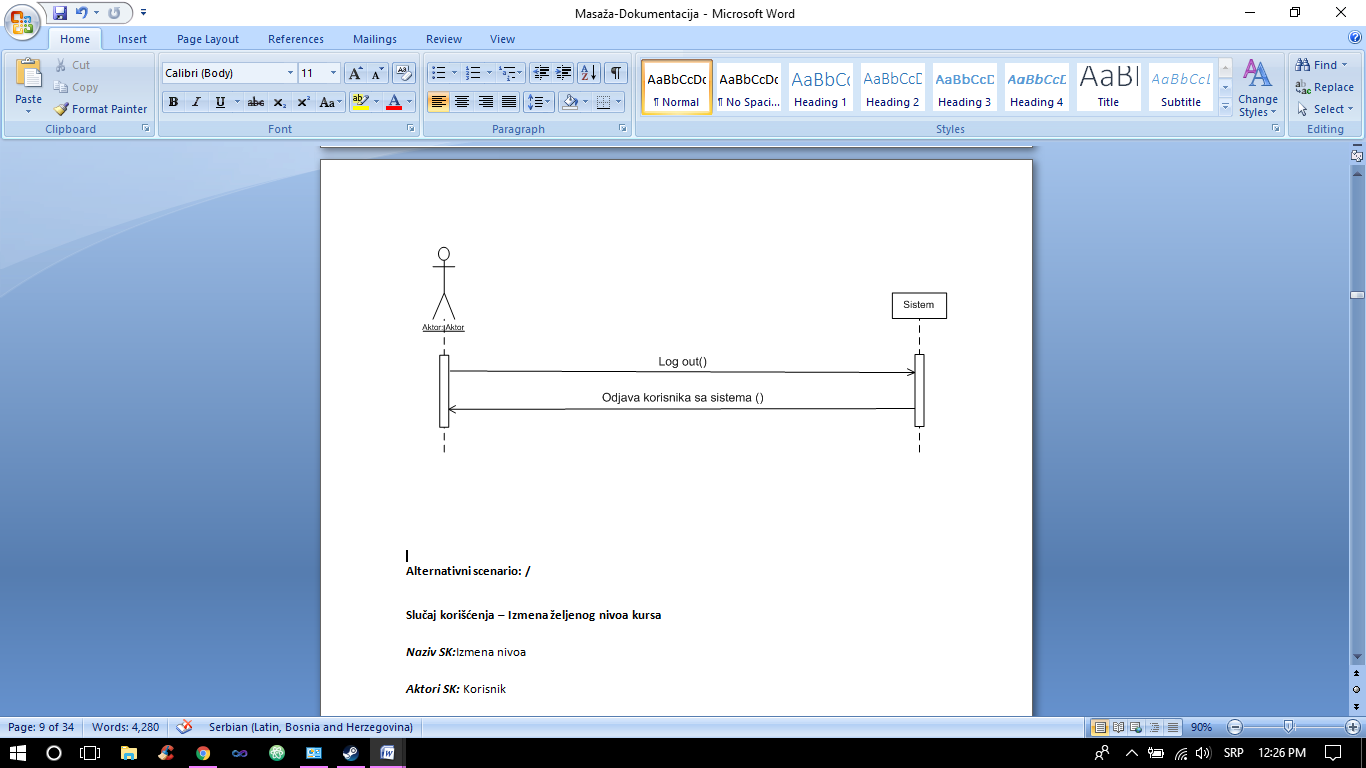
**Akteri SK:** Korisnik

**Preduslov**: Akter je predhodno ulogovan na sistem

***Osnovni scenario:***

1. Korisnik bira opciju „Log out“ iz korisničkog menija.(APSO)
2. Sistem odjavljuje korisnika i vraća ga na stranu za logovanje(IA)

**Dijagram sekvenci:**



### Pregled ponuda

**Naziv SK:** Pregled masaža

**Svrha:** Prikaz trenutne ponude masaža u salonu

**Akteri SK:** Korisnik

**Preduslov**: /

***Osnovni scenario:***

1. Korisnik otvara stranicu za prikaz filmova (APSO)
2. Sistem se povezuje sa bazom i otvara se spisak ponuda (IA)

***Dijagram sekvenci:***

### 28176287_10215478519986082_1337329434_n.jpg

### 2.1.5 Zakazivanje masaže

**Naziv SK:** Zakazivanje masaže

**Svrha:** Zakazivanje željene masaže

**Akteri SK:** Korisnik

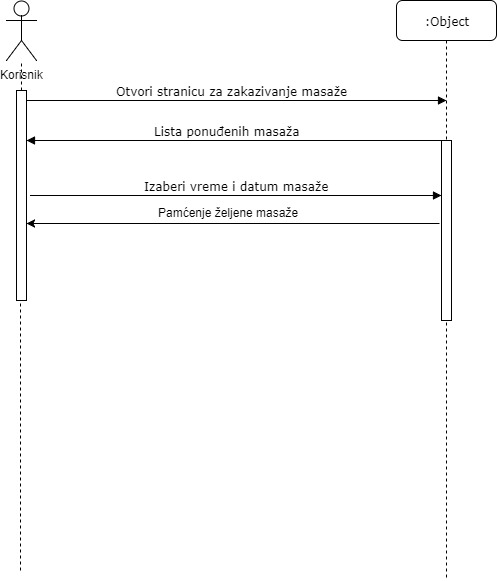
**Preduslov**: Korisnik je ulogovan na sistem

***Osnovni scenario:***

1. Korisnik otvara stranicu za zakazivanje masaže (APSO)
2. Sistem se povezuje sa bazom i stvara se lista mogućih masaža (IA)

3. Korisnik bira masažu koji želi da dobije (APSO)

4. Sistem se povezuje sa bazom, kreira novu rezervaciju i pamti u bazi zeljenu masažu (IA)



## Slučajevi korišćenja za administratora

### 2.2.1 Registracija administratora

**Naziv SK:** Registracija administratora

**Svrha:** Registracija novog administratora na sajt bioskopa

**Akteri SK:** Administrator

**Preduslov**: Korisnik je ulogovan na sistem kao administrator

***Osnovni scenario:***

1. Administrator otvara odgovarajuću formu za registraciju.(APSO)
2. Sistem vraća formu za registraciju.(IA)
3. Nakon unosa podataka korisnik pritiska dugme "Sačuvaj izmene"(APSO)
4. Sistem prihvata podatke, proverava ih i šalje poruku "Uspešno ste registrovali novog korisnika".(IA)

**Dijagram sekvenci:**



**Alternativni scenario:**

4.1. Sistem javlja da uneto korisničko korisničko ili email ime već postoji u bazi porukom “Korisnik sa ovim korisničkim imenom već postoji" ili "Korisnik sa ovim emailom već postoji".

**Dijagram sekvenci:**



### 2.2.1Log in.

**Naziv SK:** Log in

**Svrha:** Logovanje korisnika u cilju pregleda filmova u boskopui i njihove rezervacije

**Akteri SK:** Administrator

**Osnovni scenario:**

1. Korisnik unosi korisnicko ime i sifru i potvrđuje unos.(APSO)
2. Sistem prosleđuje korisnika na odgovarajucu stranicu – stranica dobrodošlice.(IA)

**Dijagram sekvenci:**



**Alternativni scenario:**

2.1. Pogrešno uneti korisničko ime ili šifra, sistem obaveštava korisnika o grešci porukom :“Pogresno korisničiko ime ili šifru. “(IA)

2.2. Ukoliko sistem ne može da se poveže na bazu obaveštava korisnika o tome.(IA)

### 2.2.3 Log out

**Naziv SK:** Log out

**Svrha:** Odjava administratora sa sistema

**Akteri SK:** Administrator

**Preduslov**: Akter je predhodno ulogovan na sistem

***Osnovni scenario:***

1. Korisnik bira opciju „Log out“ iz korisničkog menija.(APSO)
2. Sistem odjavljuje korisnika i vraća ga na stranu za logovanje(IA)



**Alternativni scenario: /**

### 2.2.4 Dodavanje nove masaže

**Naziv SK:** Unos nove masaže

**Svrha:** Unos nove masaže u bazu sistema

**Akteri SK:** Administrator

**Preduslov**: Korisnik je ulogovan kao administrator

1.       Administrator poziva sistem da otvori formu za unos nove masaže (APSO)

2.       Sistem prikazuje administratoru formu za unos podataka o masaži.(IA)

3.       Administratoru poziva sistem da zapamti podatke o masaži. (APSO)

4.       Sistem prikazuje administratoru zapamćenu masažu  poruku:" Uspešno ste dodali novu masažu!".(IA)

### sjsjsjsjjs.jpg

### 2.2.5 Generisanje pdf-a za pregled rezervacija

**Naziv SK:** Pregled i brisanje rezervacija

**Akteri SK:** Administrator

**Preduslov**: Korisnik je ulogovan kao administrator

***Osnovni scenario:***

1. Administrator bira opciju za generisanje pdf-a. (APSO)
2. Sistem se povezuje sa bazom, kreira pdf i čuva ga.(IA)

***Dijagram sekvenci:***



### 2.2.6 Pregled statistike

**Naziv SK:** Pregled statistike

**Akteri SK:** Administrator

**Preduslov**: Korisnik je ulogovan kao administrator, učitana je lista rezisera.

***Osnovni scenario:***

1. Administrator otvara stranicu za prikaz statistike (APSO)
2. Sistem se povezuje sa bazom i otvara se vizuelni prikaz statistike (IA)
3. Korisnik bira kriterijum za prikaz statistike.(APSO)
4. Sistem se povezuje sa bazom i vizuelni prikaz koji zadovoljava dati kriterijum.(IA)

***Dijagram sekvenci:***



***Alternativni scenario: /***

## 2.3 Model podataka

### 2.3.1 Pmov

1,1

1,1

0,M

0,M

0,M

0,M

Trajanje

Masaža

Zakazivanje

Kosrisnik

Admin

Korisnik

S

### 2.3.2 Dijagram klasa

nazivMasaže

Dodatak ceni

trajanje

osnovna cena

Tip

tipID

1,1

0,M

nazivtipa

trajanjeID#

TipID:int

nazivTipa:string

tipID:int

nazivTipa;String

Tip

1,1

\*

TipID:int

NA

nazivTipa:string

Tip

Tip

Trajanje

trajanjeID: int

trajanje: string

dodatakCeni: double

1,1,1

\*

Korisnik

korisnikID: int

imePrezime: string

email: string

brojTelefon: string

username :string

password: string

pristup: string

Masaža

masazaID: int

nazivMasaze: string

osnovnaCen: int

Zakazivanje

rezervacijaID: int

vreme: date

\*

\*

## 2.4. Specifikacija REST API-ja

|  |  |
| --- | --- |
| Opis funkcije | Pregled svih korisnika |
| HTTP metoda | GET |
| URL | get/zakazivanja |
| URL parametri | (nema) |
| HTTP body parametri | (nema) |
| Format HTTP body parametara | (nema) |
| Izlazni parametri | Primer:  [  {  "zakazivanjeID":"8",  "masazaID":"5",  "korisnikID":"1",  "vreme":"2020-09-01 19:08:38",  "nazivMasaze":"Masaza 4",  "trajanjeID":"4",  "tipID":"2",  "osnovnaCena":"2500",  "nazivTipa":"Relaks masaza",  "trajanje":"120 minuta",  "dodatakCeni":"1250",  "imePrezime":"Marija Maric",  "brojTelefona":"066 097 897",  "email":"marija98@gmail.com",  "username":"maki",  "password":"maki",  "pristup":"1",  "suma":"3750"  },  {  "zakazivanjeID":"9",  "masazaID":"6",  "korisnikID":"1",  "vreme":"2020-09-02 22:46:42",  "nazivMasaze":"Masaza 5",  "trajanjeID":"2",  "tipID":"4",  "osnovnaCena":"1500",  "nazivTipa":"Sportska masaza",  "trajanje":"60 minuta",  "dodatakCeni":"500",  "imePrezime":"Marija Maric",  "brojTelefona":"066 097 897",  "email":"marija98@gmail.com",  "username":"maki",  "password":"maki",  "pristup":"1",  "suma":"2000"  },  {  "zakazivanjeID":"10",  "masazaID":"1",  "korisnikID":"4",  "vreme":"2020-09-04 17:01:00",  "nazivMasaze":"Masaza 1",  "trajanjeID":"1",  "tipID":"1",  "osnovnaCena":"700",  "nazivTipa":"Anticelulit masaza",  "trajanje":"30 minuta",  "dodatakCeni":"0",  "imePrezime":"Petar Petrovic",  "brojTelefona":"065555555",  "email":"petarnjegos@gmail.com",  "username":"pero",  "password":"pero",  "pristup":"0",  "suma":"700"  }  ] |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | vratiZakazaneKorisnike |
| HTTP metoda | GET |
| URL | GET /zakazivanja/@id |
| URL parametri | /zakazivanje/1 |
| HTTP body parametri | [  {  "zakazivanjeID":"8",  "masazaID":"5",  "korisnikID":"1",  "vreme":"2020-09-01 19:08:38",  "nazivMasaze":"Masaza 4",  "trajanjeID":"4",  "tipID":"2",  "osnovnaCena":"2500",  "nazivTipa":"Relaks masaza",  "trajanje":"120 minuta",  "dodatakCeni":"1250",  "imePrezime":"Marija Maric",  "brojTelefona":"066 097 897",  "email":"marija98@gmail.com",  "username":"maki",  "password":"maki",  "pristup":"1",  "suma":"3750"  },  {  "zakazivanjeID":"9",  "masazaID":"6",  "korisnikID":"1",  "vreme":"2020-09-02 22:46:42",  "nazivMasaze":"Masaza 5",  "trajanjeID":"2",  "tipID":"4",  "osnovnaCena":"1500",  "nazivTipa":"Sportska masaza",  "trajanje":"60 minuta",  "dodatakCeni":"500",  "imePrezime":"Marija Maric",  "brojTelefona":"066 097 897",  "email":"marija98@gmail.com",  "username":"maki",  "password":"maki",  "pristup":"1",  "suma":"2000"  },  {  "zakazivanjeID":"10",  "masazaID":"1",  "korisnikID":"4",  "vreme":"2020-09-04 17:01:00",  "nazivMasaze":"Masaza 1",  "trajanjeID":"1",  "tipID":"1",  "osnovnaCena":"700",  "nazivTipa":"Anticelulit masaza",  "trajanje":"30 minuta",  "dodatakCeni":"0",  "imePrezime":"Petar Petrovic",  "brojTelefona":"065555555",  "email":"petarnjegos@gmail.com",  "username":"pero",  "password":"pero",  "pristup":"0",  "suma":"700"  }  ] |
| Format HTTP body parametara | application/json |
| Izlazni parametri | Nema |
| Format izlaznih parametara | application/json |

|  |  |
| --- | --- |
| Opis funkcije | Registracija korisnika |
| HTTP metoda | GET |
| URL | /podaciZaGraf |
| URL parametri | (nema) |
| HTTP body parametri | [  {  "imePrezime":"Marija Maric",  "brojMasaza":"2"  },  {  "imePrezime":"Milica Milic",  "brojMasaza":"1"  }  ] |
| Format HTTP body parametara | application/json |
| Izlazni parametri | / |
| Format izlaznih parametara | application/json |

# 3.Opis tehnologija korišćenih u aplikaciji

U našem radu koristili smo sledeće tehnologije:

1. HTML
2. CSS
3. PHP
4. AJAX
5. JSON
6. JavaScript
7. XML
8. JQuery

## 3.1.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.).

Neke od osnovnih odrednica PHP – a su:

* PHP je akronim za Hypertext Preprocessor
* server – side skriptni programski jezik
* skripte se izvršavaju na serveru
* podržava rad sa različitim bazama podatama (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, itd.)
* open source softver (OSS)
* radi na različitim platformama (Windows, Linux, Unix i td.)
* kompatibilan je sa skoro svim serverima koji se danas koriste (Apache, IIS, itd. )
* lak je za učenje i efikasno funkcioniše na strani servera.

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*”
* Program koji se napiše u PHP-u ne zahteva prevođenje (kompajliranje), nego se interpretira pri svakom izvršavanju. PHP interpreter može raditi po CGI principu, odnosno tako što će interpreter postojati kao eksterna aplikacija koja se poziva da izvrši datu skriptu svaki put kad bude zahtevana od nekog korisnika, a može biti instaliran i kao modul veb-servisa. Druga varijanta je danas u najvećoj upotrebi jer pruža znatno veću brzinu izvršavanja - interpreter je na taj način uvek učitan u memoriju te se ne mora pozivati spoljašnji program.

Uobičajen scenario po kom se izvršavaju PHP skripte na web serveru je sledeći:

* klijent (korisnik Interneta koji koristi neki brauzer) zahteva PHP stranicu sa servera
* server prosleđuje zahtev servisu za veb (program veb-server na serveru)
* veb-server prepoznaje da se traži PHP datoteka
* ne šalje njegov sadržaj klijentu, nego ga izvršava kao program pomoću PHP modula
* izlazni tekst programa (standardni izlaz) se šalje klijentu kao rezultat zahteva
* klijent prepoznaje vrstu rezultata (HTML kod, slika, PDF sadržaj, arhiva itd.)

rezultat se prikazuju klijentu na odgovarajući način

## 3.2.AJAX(Asynchronous JavaScript And XML)

AJAX je skraćenica od “Asynchronous JavaScript and XML”. Termin AJAX prvi put je upotrebljen u februaru 2005. godine, kada je Džesi Džejms Garet (Jesse James Garret), dizajner informacionih sistema i direktor kompanije Adaptive Path, pokušao da nađe odgovarajuću skraćenicu za grupu tehnologija koju je predlagao svom klijentu. Tehnologije potrebne za postojanje AJAX -a su postojale i ranijih godina ali je glavni razlog za slabiju upotrebu svih tehnika korišćenih u AJAX-u, nedostatak podrške od strane web čitača. Mnogi su godinama pre nje koristile slične principe, poput Microsoftovog Remote Scriptinga ili veoma raširenog DHTML-a. Najbitnija stvar koja je uticala na brz razvoj AJAX-a jeste trenutak na tržištu u kojem su veliki igrači Internet industrije želeli da težište korišćenja računara prenesu sa desktopa na web stranice, za šta im je bila potrebna upravo tehnologija.

Web aplikacije imaju brojne prednosti u odnosu na desktop aplikacije. Web aplikacije dosežu do većeg broja ljudi, lakše se razvijaju, implementiraju i održavaju.

Najveći nedostatak Web aplikacija u odnosu na desktop aplikacije je taj što su desktop aplikacije „bogatije” u smislu funkcija i informacija koje nude. Međutim, taj nedostatak je u velikoj meri otklonjen uz pomoć AJAX tehnologije.

Web aplikacije su one aplikacije čijom funkcionalnošću se upravlja preko web servera i dostavljaju se korisnicima preko mreže kao što je internet ili intranet. I pored mnogostrukih prednosti, klasične web aplikacije imaju određene neostatke, kada je u pitanju interakcija sa korisnicima:

* Spor odgovor na zahteve
* Gubitak sadržaja prilikom ponovnog učitavanja stranice
* Gubitak informacija na ekranu
* Gubitak pozicije skrolovanja
* Bez trenutnog odgovora na korisničke akcije
* Korisnik mora da čeka na učitavanje sledeće strane
* Klikni, čekaj, refresh princip rada na stranici
* Stranica se ponovno učitaca sa servera za sve akcije korisnika
* Sinhroni način rada.

Pre svega, svaki put kada se učitava nova stranica, postoji određeno vreme kada aplikacija «stoji». Veoma često se prilikom malih izmena ili zahteva za delovima stranice, ponovo učitava cela stranica, iako je najveći deo nove stranice u potpunosti identičan sa prethodnom. Da bi se shvatio koncept AJAX tehnologija, neophodno je objasniti razliku izmenu sinhronog i asinhronog prenosa podataka između klijenta i servera. U sinhronom načinu rada procesi se izvode sekvencijalno. Između dva procesa postoji jaz koji je jednak vremenu izvođenja drugog procesa. Dakle, osnovni proces komunikacije između klijenata i servera može se opisati na sledeći način: klijent šalje zahtev ka serveru, podaci se prenose ka serveru, server obrađuje podatke I zatim ih vraća klijentu. Ono što je primetno sa slike dole je da postoji vremenski interval između pravljenja zahteva i odgovora na isti. Za vreme tog intervala klijent čeka, bez mogućnosti da pravi nove zateve ili nastavi svoje korišćenje web aplikacije. Može se zaključiti da kod sinhronog prenosa podataka ne postoji mogućnost za eliminisanje čekanja zato što se procesi obavljaju jedan za drugim, čak i ukoliko za to nema potrebe.

## 3.3.JSON

JSON (JavaScript Object Notation) predstavlja tekstualni format za serijalizaciju podataka. Izveden je od formata za zapis objekta skriptnog jezika JavaScript, međutim danas ovaj format koristi većina savremenih programskih jezika. Osnovna svrha JSON-a je prenos podataka i može se reći da predstavlja svojevrsnu alternativu XML-u. JSON je u memorijskom smislu manje zahtevan od XML-a, što znači da JSON fajl sa određenim podacima zauzima manje memorijskog prostora od XML fajla sa istim podacima. JSON je zasnovan na JavaScript standardu Standard ECMA-262 3rd Edition - December 1999. Iako je JSON usko povezan sa JavaScript programskim jezikom, danas praktično svi programski jezici podržavaju JSON zapis podataka. Među njima su JavaScript, PHP, Java, C, C++, Pyhon, Ruby i drugi.

JSON se zasniva na dve strukture:

* Zbirka parova ime / vrednost. Na raznim jezicima, to je realizovano kao objekat, zapis, struktura, rečnik, heš tabela, lista sa ključevima ili asocijativni niz.
* Uređena lista vrednosti. U većini jezika, ovo je realizovana kao niz, vektor, lista ili sekvenca.

JSON format razlikuje više tipova podataka. Promenljive se u JSON-u nazivaju vrednosti. Od prostih tipova podataka, to su string, number, boolean i null, a od složenih tipova u pitanju su object i array.

## 3.4.JavaScript

JavaScript je objektno zasnovan skriptni jezik. Uključujemo ga u web stranicu da bi je učinili dinamičnijom. HTML (osnovni kod web stranica) se koristi samo za oblikovanje i uređivanje elemenata stranice (tekst, forme, linkove i tabele), ali nema šanse da diktiramo kako će se ti elementi ponašati. Mogućnost uključenja JavaScript skripte daje nam mnogo veću kontrolu kako se web stranica ponaša. Kombinovan sa HTML-om i CSS-om JavaScript čini DHTML (Dinamic HTML).

Objektno je zasnovan jer programer ne definiše samo tip podataka, nego i vrstu operacija (funkcija) koje se mogu primjeniti na strukture podataka. Na ovaj način, struktura podataka postaje objekat koji uključuje i podatke i funkcije. Pored toga, programeri mogu da kreiraju odnose između jednog i drugog objekta. Na primjer, objekti mogu da pridobiju karakteristike od drugih objekata.

Skriptni je jezik jer se sastoji od serije komandi koje se očitavaju u interpreteru (program prevodioc), a da se predhodno ne kompajlira sadržaj (compiler- program prevodioc). Odnosno ne prevodi se u mašinski jezik (binarni kod- 1 i 0) iz koga nikada nećemo saznati originalni jezik, nego se komande direktno "čitaju" iz koda (source code ili bytecode). Zbog ove karakteristike JavaScript se izvršava na strani korisnika (client side), tj. na računaru na kojem je pokrenut sadržaj sa JavaScript-om. Sam po sebi, HTML dozvoljava posjetocu da pošalje podatke ka serveru na obradu. Nažalost ako ti podaci nisu validni cijeli proces se mora ponoviti sve dok se ne unesu validni podaci. Ovo je jedan od osnovnih razloga nastanka JavaScript-a koji provjerava vjerodostojnost podataka na klijentovom pregledniku (browser-u) i tako olakšava posao na web-u.

JavaScript je najpopularniji skriptni jezik na Internetu kojeg podržavaju svi poznatiji preglednici (Internet Explorer, Mozilla Firefox, Netscape, Opera). Evo nekoliko stvari koje je svaki prosječan surfer na internetu vidio: padajući meni, neobični pokazivač miša, iskakajući prozor, sat ... Međutim postoje ozbiljnije primjene JavaScript-a kao što su: Detekcija preglednika -Određuje se tip preglednika koji se koristi pri pregledu vaše web stranice. Zavisno od preglednika, može se prikazati drugačija stranica posebno dizajnirana za taj preglednik. Kontrola prozora -mogu se kontrolisati dimenzije, meniji, dugmadi, vrijeme prikaza itd. Web kolačići (cookies) -pohranjeni podataci na korisnikovom računaru, koji se automatski preuzimaju sljedeci put kada korisnik posjeti vašu web stranicu. Provjera sadržaja -dobar primjer za ovo je provjeravanje da li je korisnik upisao znak @ u polje za upis email adrese.

## 3.5.Jquery

jQuery je biblioteka čija je svrha da olakša upotrebu JavaScript-a na vašim web stranicama. jQuery slogan „write less, do more“ dosta govori – sa jednom linijom jQuery koda možete pozvati funkcionalnosti za koje bi vam trebalo mnogo više rada i truda, ako bi koristili klasični JavaScript metod kodiranja. jQuery obuhvata mnoge zadatke koji zahtevaju veliki broj linija JavaScript-a i omotava ih u metode koje možete pozvati sa samo jednom linijom. jQuery takođe pojednostavljuje i druge komplikovane stvari u JavaScript-u kao što su AJAX i DOM manipulacija.

JQUERY BIBLIOTEKA SADRŽI SLEDEĆE FUNKCIONALNOSTI:

* HTML/DOM manipulacija
* CSS manipulacija
* HTML event metodi
* Efekti i animacije
* AJAX

**Dodatno:**[jQuery poseduje plugin-e](http://www.popwebdesign.net/popart_blog/2013/08/najbolji-jquery-dodaci/) sa kojima možete postići gotovo svaki zahtev koji zamislite.

Najvažniji razlog tolike poularnosti jQuery biblioteke je ekstezivnost. Veliki broj najvećih svetskih kompanija kao što su Google, Microsoft, IBM ga koriste.

Tri stvari su značajne i izdvojile su jQuery u odnosu na druge konkurente:

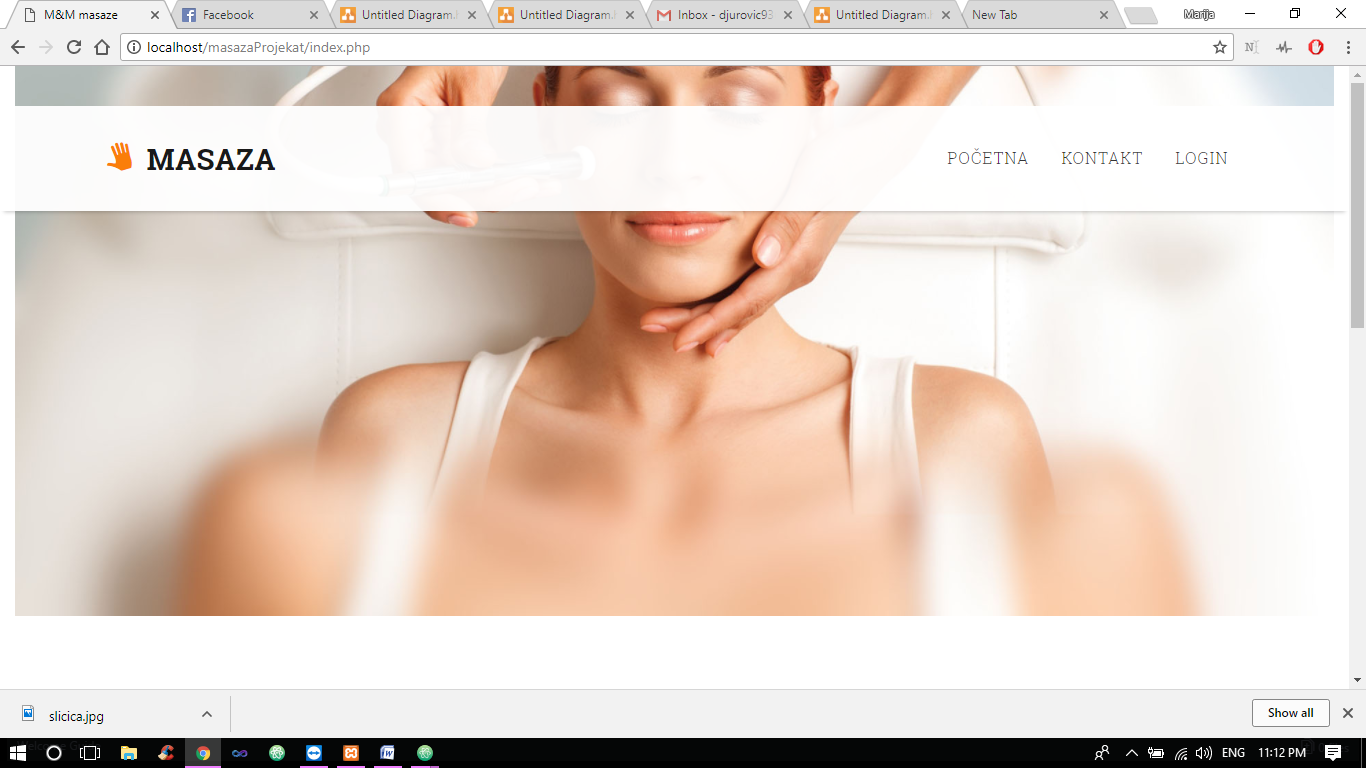
* korišćenje CSS selektora
* unobtrusive (nenametljiv) JavaScript
* jednostavnost upotrebe

jQuery se pojavio u vreme kada su se front-end developeri navikavali na kreiranje web stranica sa boljim semantičkim obeležavanjem, a učenje CSS-a je bilo zastupljeno širom web-a. jQuery koristi potpuno iste [CSS selektore](http://www.popwebdesign.net/popart_blog/2013/08/10-korisnih-css-selektora/) koje developeri koriste za stilizovanje web stranica da bi dodali ponašanje i nema potrebe za promenom konteksta kada se referencira na iste elemente. „Unobtrusive JavaScript“je metodologija koja definiše napredan način inplementiranja JavaScript koda na web stranicama.

# 4.Korisničko uputstvo

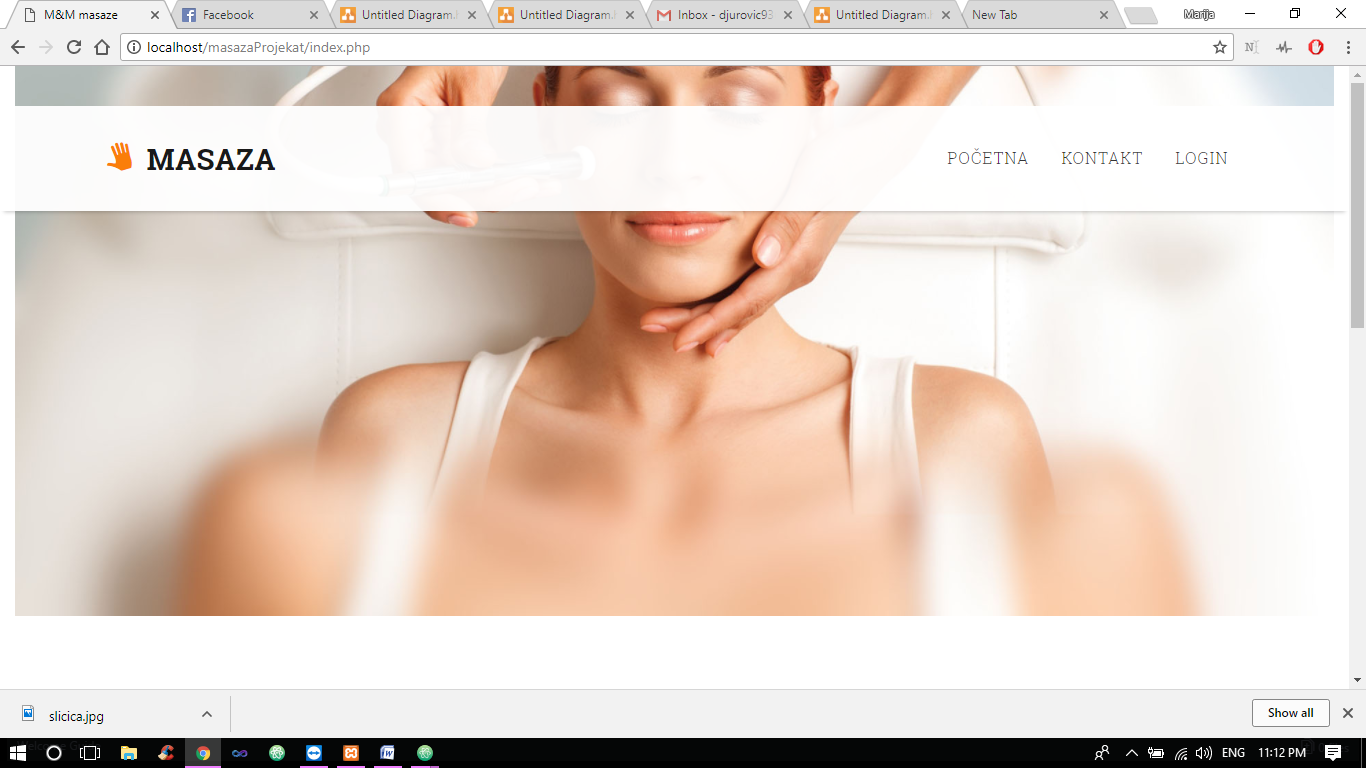
## 4.1.Korisnik

Pri inicijalnom otvaranju sajta korisnik dobija prikaz početne strane na kojoj može dobiti neke od prvih informacije o aktuelnim masažama i mogućnosti za stampanje pdf file-a, kontakt, kao i opciju logovanja.



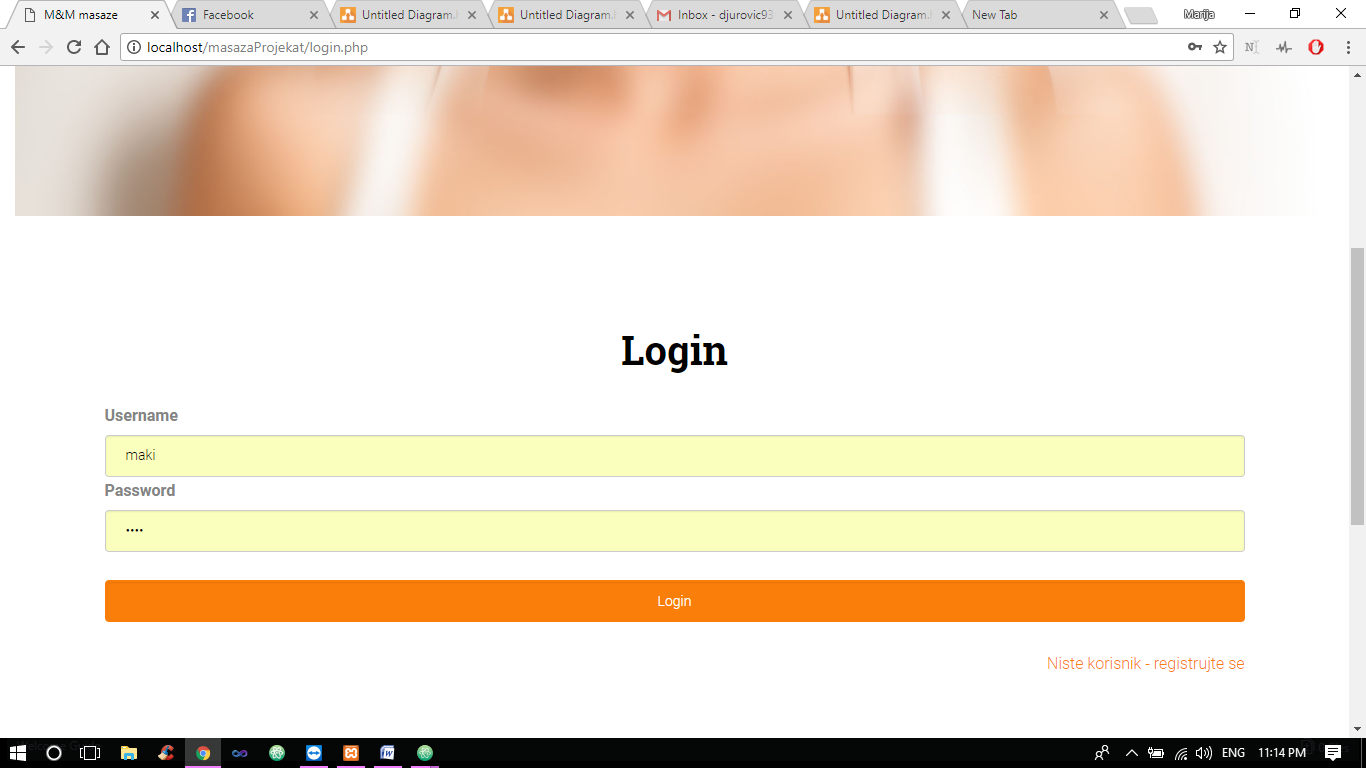
Početna strana 1

Na vrhu početne strane nalazi se meni.

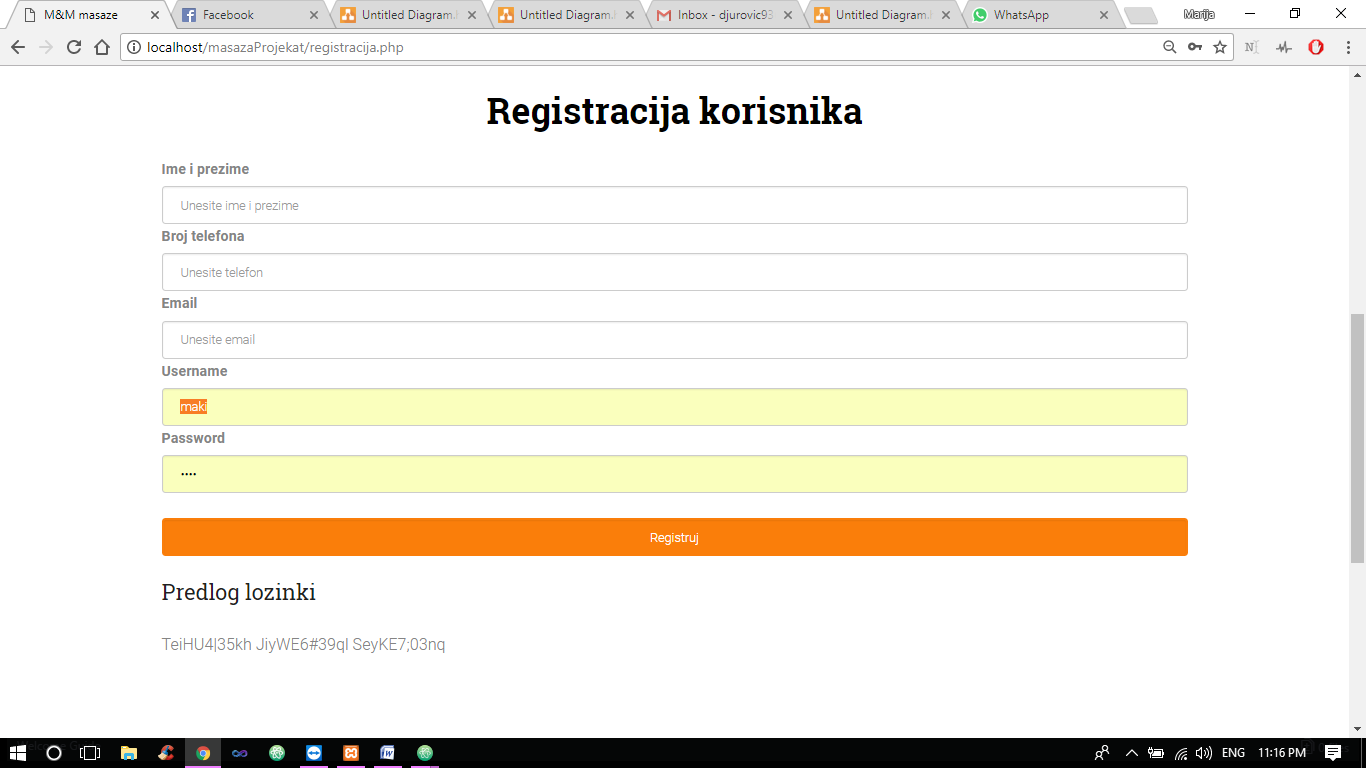


Meni 1

Na kartici „Kontakt“ korisnik može dobiti informacije o kontakt telefonu i adresi, kao i da pristup Google mapi.

Na kartici „Login“ korisnik unosi username i password pomoću koje se kasnije loguje . 

Ukoliko korisnik ne postoji u bazi može se registrovati unosenjem imena i prezimena, broja telefona, e-mail adrese, username – a, i password-a. Korisnik takodje ima i ponudjen predlog lozinki koje može da iskoristi.



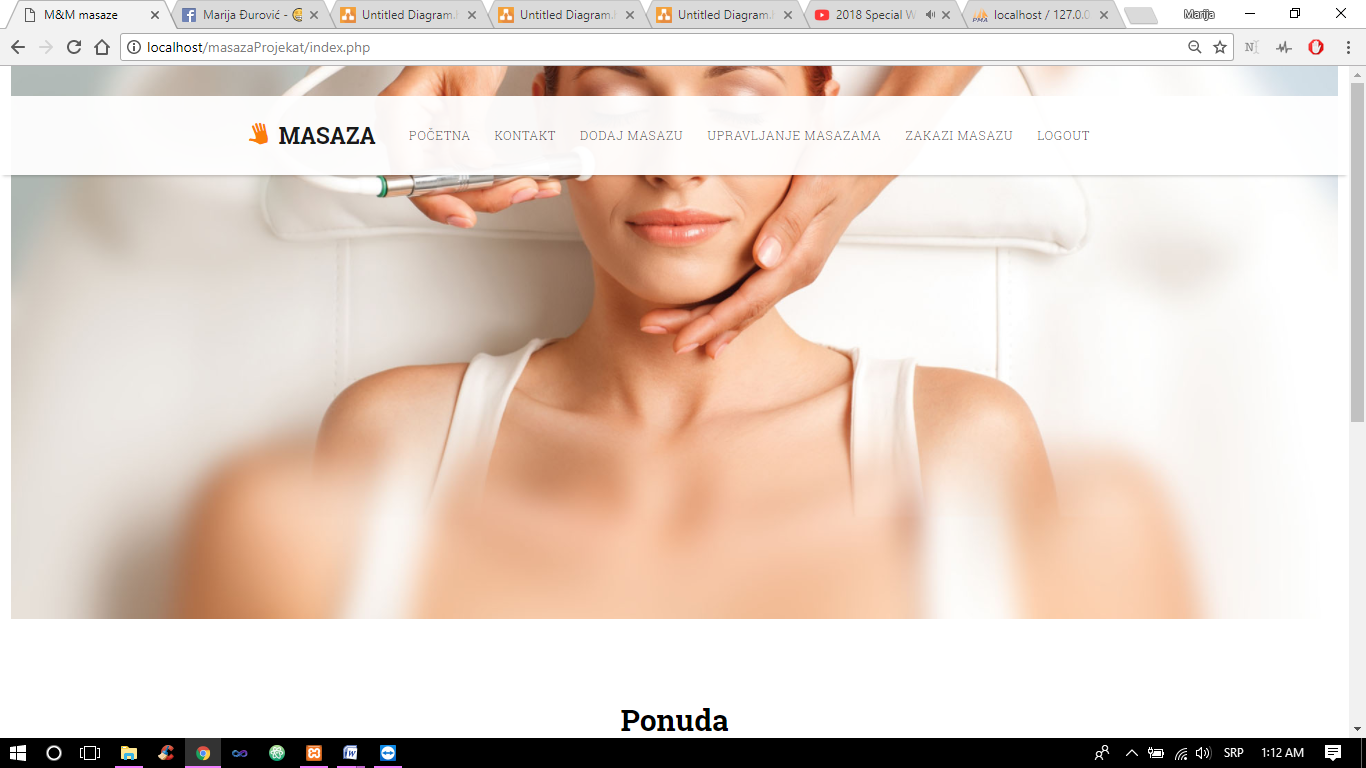
Na kartici „Početna“ korisnik može videti sve moguće masaže. Postoji dve opcije logovanja, kao korisnik I kao administrator. Ukoliko se logovanje vrsi kao korisnik to izgleda ovako.

Jedna od bitnijih kartica je „Zakazi masažu“. Kada korisnik klikne na tu karticu u meniju otvara mu se strana sledećeg sadržaja. Pored ove funcije imamo mogućnost stampanja ponude u pdf formatu.

Korisniku se prikazuje padajući meni gde moze da izabere masazu i da zakaze tacno vreme i datum masaze. Ukoliko se korisnik uloguje kao administrator njegov meni izgleda ovako.

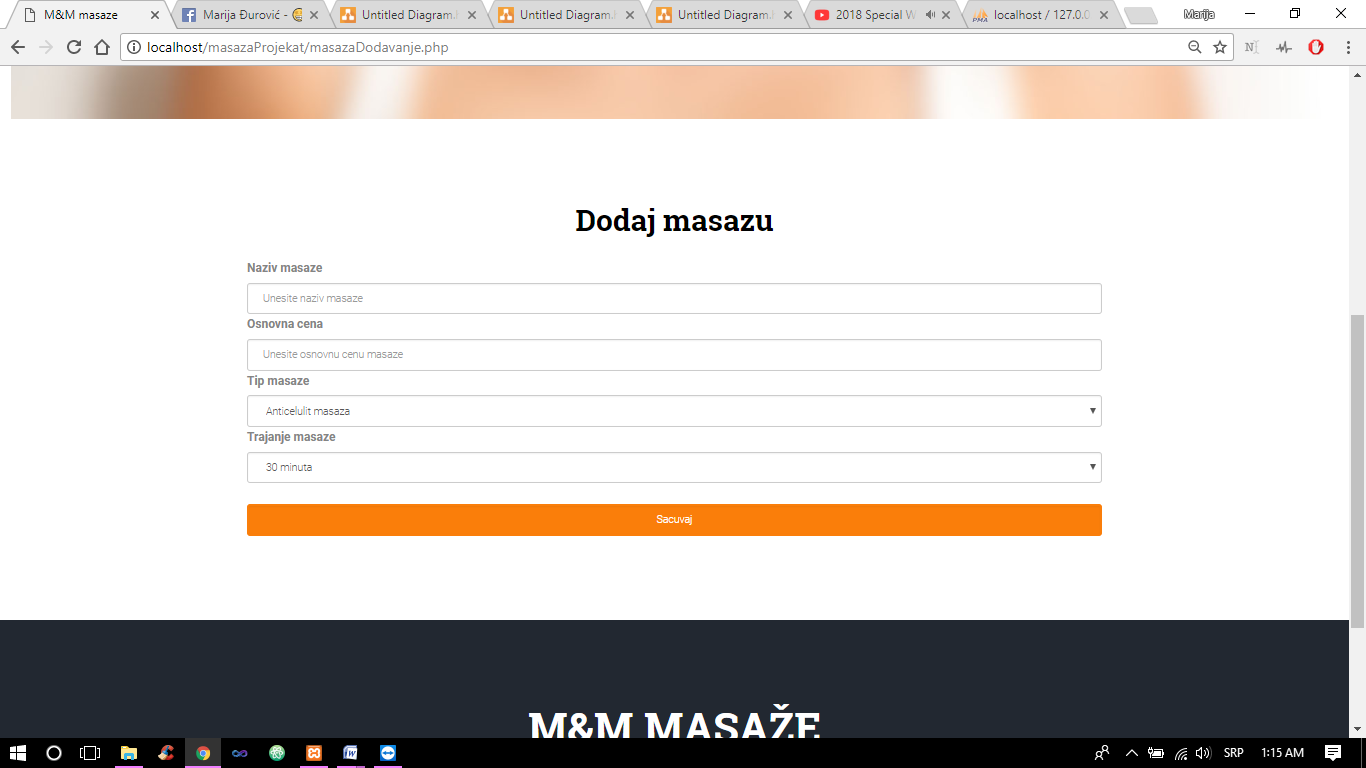
## 4.2.Admin

Ukoliko je korisnik na sajt ulogovan kao admin meni koji mu se prikazuje se razlikuje od onog koji običan registrovan korisnik vidi.



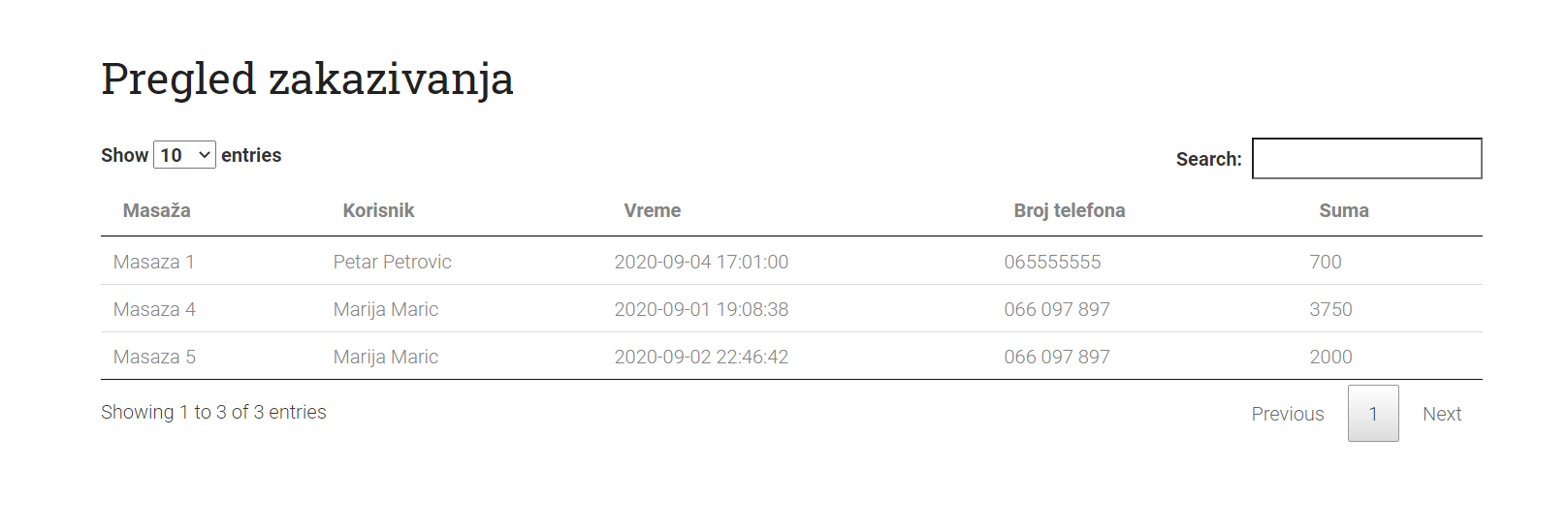
Kao što se sa slike vidi, admin takodje ima pristup početnoj strani, kontakt, dodaj masažu, uredi masažu, zakaži masažu i logout.

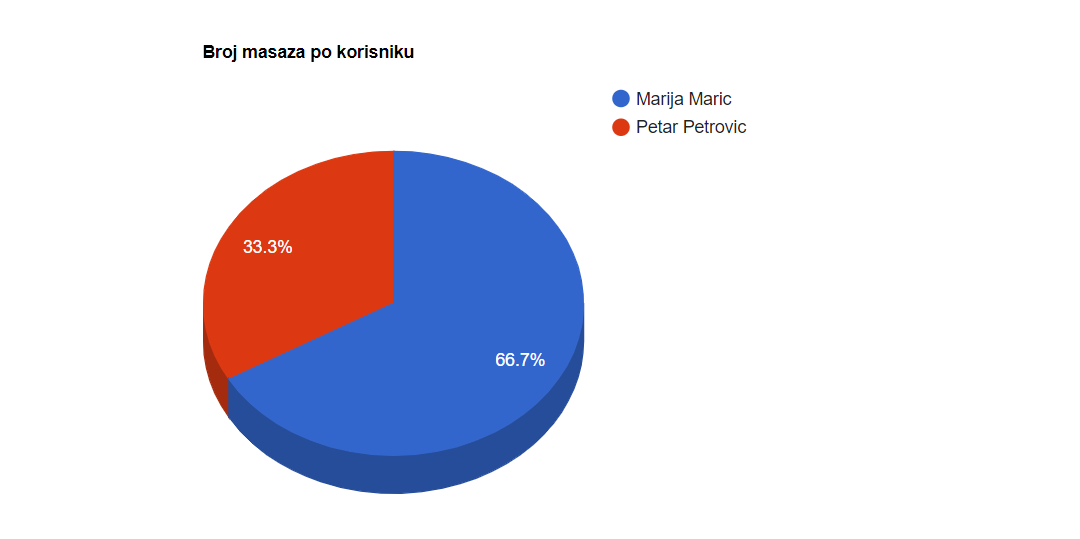
Razlika je u karticama dodaj masažu, upravljaj masažama.



Kada admin klikne na „upravljaj masažama“ dobija sledeći prikaz

U administraciji se prikazuje spisak svih naziva masaža, trajanja, tip masaža, osnovna cena ukupna cena. Tabela se moze sortirati po odredjenim kriterijumima. Takodje sa desne strane se nalaze opcije za brisanje i izmenu. U okviru ove opcije imamo grafički prikaz broja masaža po korisniku.



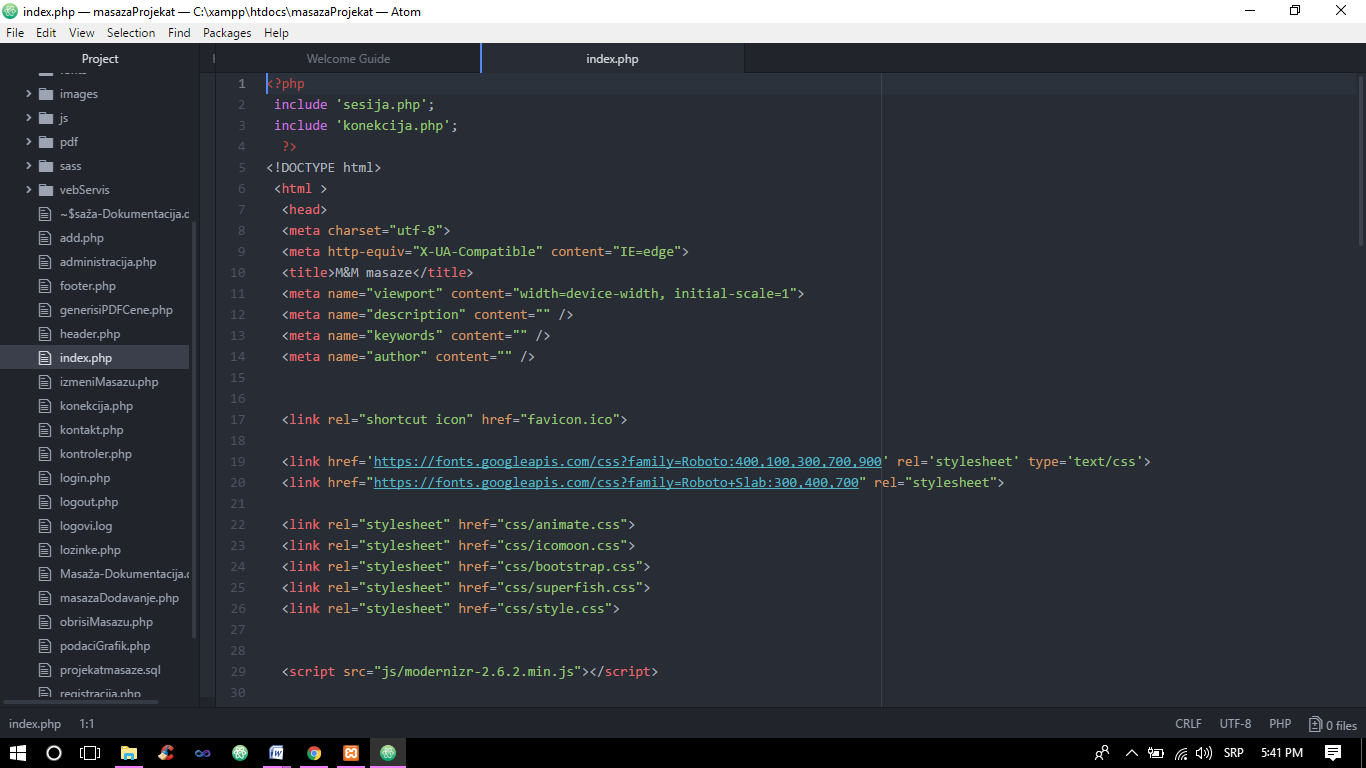


Na vrhu se nalazi combo box iz kojeg možemo birati koliko ćemo zakazanih termina da prikažemo. Sa gornje desne strane opcija Search nam nudi željenu pretragu. Omogućen nam je prikaz zakazanih termina i podataka o korisnicima koji su zakazali. Nakon završetka rada, na isti način kao i običan korisnik admin pritiska „Logout“ i odjavljuje se.

# 5.Prikaz reprezentativnih delova koda

Na sledećih nekoliko slika biće prikazani delovi koda i objašnjene njihove funkcionalnosti.

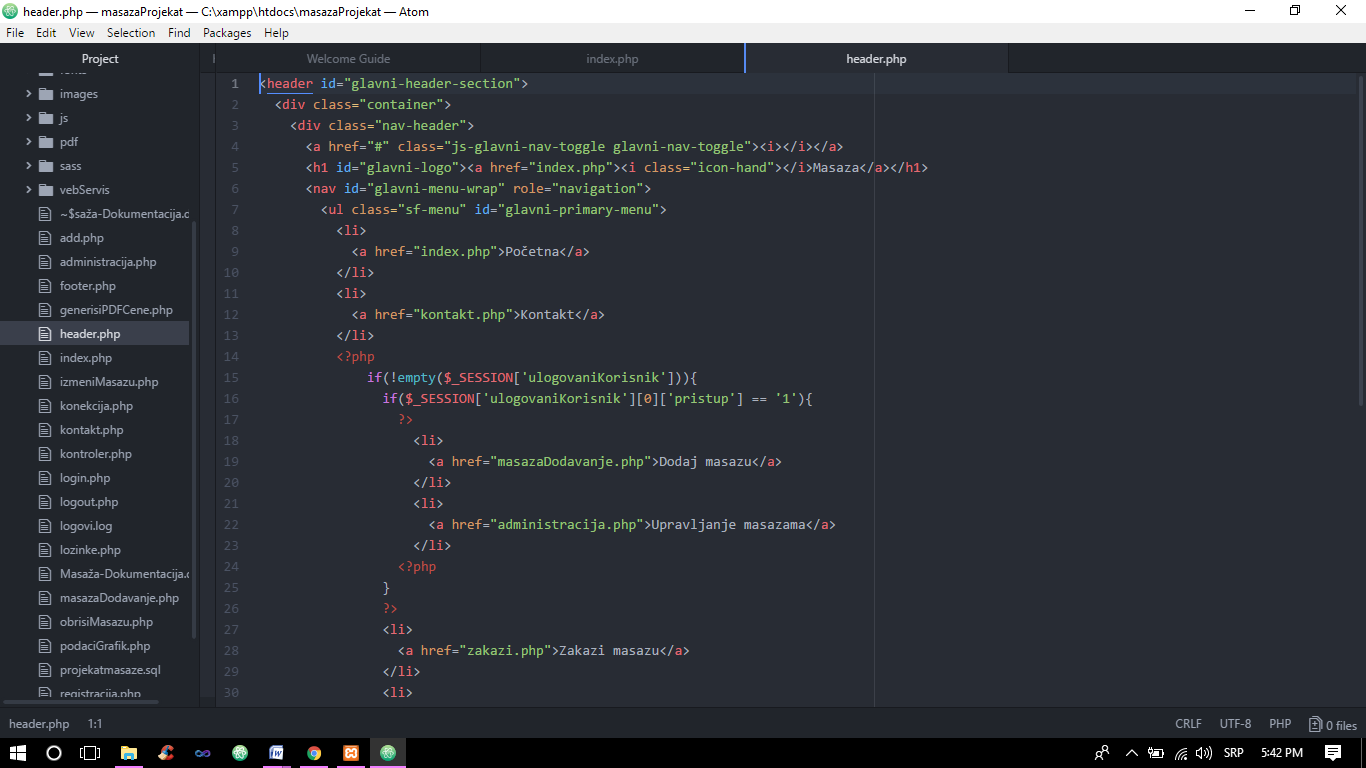
Index.php



U primeru iznad, prikazano je korišćenje HTML5. Pomoću funkcije include uključili smo fajlove header.php, konekcija.php i footer.php, koji se nalaze u folderu potrebno, na našu početnu stranicu kako ne bismo morali svaki put da ih definišemo kroz kod.

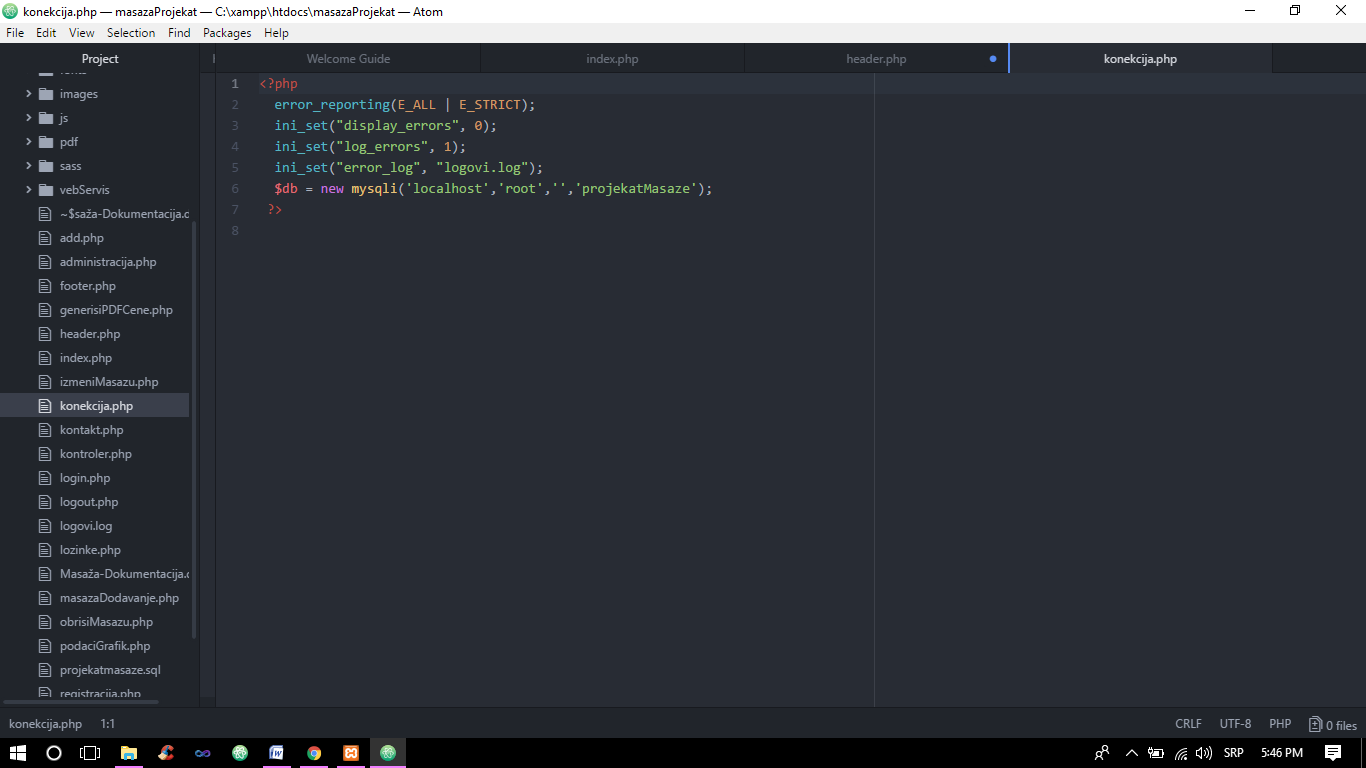
Pored toga na početnoj stranici nalazi se i slajder, kao i opis njegovih karakteristika.

Header.php



Kod startovanja konekcije sa bazom podataka, globalna promenljiva je postavljena na nije ulogovan tačnije id=0, ukoliko se korisnik uloguje id se menja u 1.

Konekcija.php

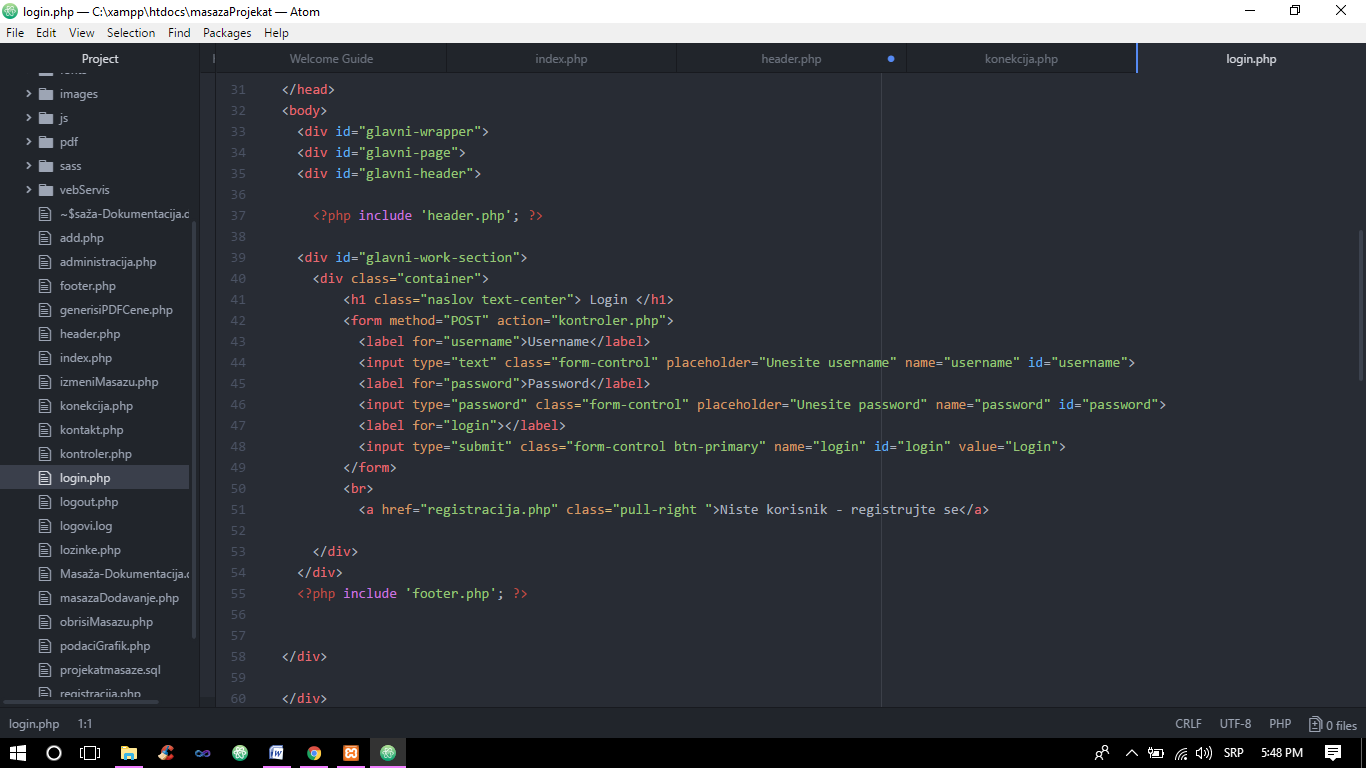


Kao i kod svakog rada sa bazom podataka, neophodno je izvršiti konkciju sa istom.

U prvih četiri koraka prosledjujemo parametre neophodne za konekciju, a to su server, user, password i naziv baze. Vrši se povezivanje preko funkcije mysqli u kojoj prosledjujemo unete parametre.

Ukoliko nije uspešno izvršena konekcija ispisuje se poruka o grešci.

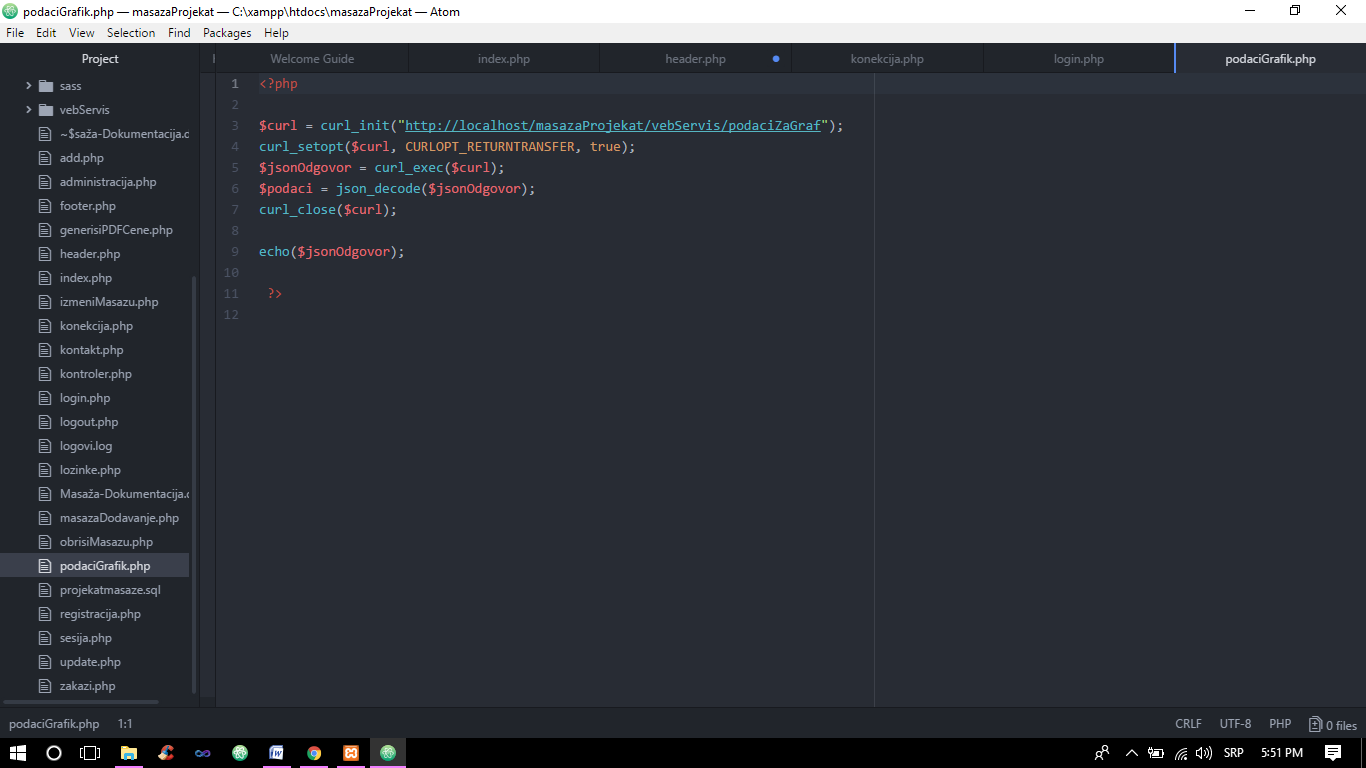
Login.php



Bitnost koda iznad se ogleda u tome što smo za prenos podataka pri loginu koristili funkciju POST koja ne prikazuje parametre kroz komandnu liniju već se šalje kontroleru zahtev koji ga dalje prosledjuje. Ukoliko nije uspesno logovanje, sistem će ispisati poruku.

podacizaGrafik.php

Sledeci kodovi se odnose na vizuelizaciju podataka o broju izdatih knjiga.



Sledeci kod se odnosi na vizelizaciju u obliku pie.

