

[Type here]

[Type here]

## **Visoka ICT škola**

Beograd, Zdravka Čelara

# **Dokumentacija interaktivnog web sajta – Laravel framework (Web programiranje PHP 2)**

**Student:**

**Milica Pavlović 73/17**

Beograd, mart 2021.

## Sadržaj

1.Uvod .....	5
1.1 Korišćeni programski jezici .....	5
1.2 Opis funkcionalnosti .....	5
1.3 Template .....	8
1.4. Slika baze .....	8
2. Organizacija.....	9
2.1 Mapa sajta .....	9
2.2 Slike stranica i opisi funkcionalnosti .....	11
2.3.1 home.blade.php.....	11
2.3.2 product.blade.php .....	13
2.3.3 onama.blade.php .....	14
2.3.4 location.blade.php .....	15
2.3.5 kontakt.blade.php.....	16
2.3.6 galerija.blade.php .....	17
2.3.7 admin.blade.php.....	18
2.3.8 admin/users/index.blade.php .....	19
2.3.9 admin/users/create.blade.php.....	20
2.3.10 admin/users/edit.blade.php.....	20
2.3.11 admin/users/activity.blade.php .....	21
2.3.12 admin/products/index.blade.php.....	22
2.3.13 admin/products/create.blade.php .....	23
2.3.14 admin/products/edit.blade.php .....	24
2.3.15 user/edit.blade.php .....	25
2.3.16 pages/autor.blade.php.....	26
3. Kodovi .....	27
3.0 routes/web.php.....	27
3.1 App/Http/Controllers.....	28
3.1.1 App/Http/Controllers/OsnovniController.php .....	28
3.1.2 App/Http/Controllers/HomeController.php.....	30
3.1.3 App/Http/Controllers/OnamaController.php.....	31

3.1.4 App/Http/Controllers/ProductController.php.....	31
3.1.5 App/Http/Controllers/LokacijeController.php .....	32
3.1.6 App/Http/Controllers/KontaktController.php.....	32
3.1.1 App/Http/Controllers/GalerijaController.php.....	33
3.1.7 App/Http/Controllers/AdminController.php.....	33
3.2 App/Http/Controllers/AdminPanel.....	33
3.2.1 App/Http/Controllers/ProductController.php.....	33
3.2.2 App/Http/Controllers/UserController.php.....	36
3.2.3 App/Http/Controllers/LocationController.php.....	39
3.2.4 App/Http/Controllers/GalleryController.php .....	40
3.3 App/Http/Controllers/UserControllers .....	40
3.3.1 App/Http/Controllers/UserController.php.....	40
3.4 App/Http/Requests.....	41
3.4.1 App/Http/Requests/StoreProductController.php .....	41
3.4.2 App/Http/Requests/StoreUserController.php .....	42
3.5 App/Http/Middleware .....	43
3.5.1 App/Http/Middleware/IpLoggingMiddleware.php .....	43
3.6 App/Models.....	43
3.6.1 App/Models/About_us.php.....	43
3.6.2 App/Models/Autor.php.....	44
3.6.3 App/Models/Category.php.....	44
3.6.4 App/Models/Gallery.php.....	45
3.6.5 App/Models/Location.php .....	45
3.6.6 App/Models/Nav_link.php .....	45
3.6.7 App/Models/Navbar.php .....	46
3.6.8 App/Models/Product.php .....	46
3.6.9 App/Models/Useractivity.php .....	48
3.6.10 App/Models/Users.php .....	48
3.6.11 App/Models/.....	50
3.7 public.....	50
3.7.1 public/assets/css/style.css.....	50
3.7.2 public/assets/js/main.js .....	53
3.8 resources/views/layouts .....	62

3.8.1 resources/views/layouts/layout.blade.php.....	62
3.8.2 resources/views/layouts/admin.blade.php .....	62
3.9 resources/views/fixed.....	63
3.9.1 resources/views/fixed/404.blade.php .....	63
3.9.2 resources/views/fixed/adminPageLinks.blade.php .....	63
3.9.3 resources/views/fixed/footer.blade.php .....	64
3.9.4 resources/views/fixed/head.blade.php .....	67
3.9.5 resources/views/fixed/message.blade.php .....	67
3.9.6 resources/views/fixed/nav.blade.php .....	68
3.9.7 resources/views/fixed/scripts.blade.php.....	69
3.10 resources/views/partials.....	69
3.10.1 resources/views/fixed/product.blade.php.....	69
3.11 resources/views/pages .....	69
3.11.1 resources/views/pages/autor.blade.php .....	69
3.11.2 resources/views/pages/ .....	70

## 1.Uvod

### 1.1 Korišćeni programski jezici

Za izradu sajta *Kaficomika* korišćene su sledeće tehnologije: HTML, CSS, bootstrap, java script, jQuery v3-2-1, AJAX, PHP Laravel framework, rad nad bazom u PHPmyAdmin i skladištenje podataka u sesije.

### 1.2 Opis funkcionalnosti

Ono što važi za ceo *Kaficomika* sajt je:

- Sajt je responsive
- Dinamičko ispisivanje celokupnog sadržaja iz baze
- Organizovan je u 2 layouta, jednim koji ima funkciju da prikaže sadržaj namenjen svima, neautorizovanim korisnicima (layout.blade.php) i jednim namenjenim autorizovanim korisnicima (admin.blade.php)
- Dinamičko ispisivanje grešaka iz javascripta ili controllera

home.blade.php, onama.blade.php, lokacije.blade.php, galerija.blade.php, autor.blade.php:

- Dinamičko ispisivanje celokupnog sadržaja iz baze i dinamičko ispisivanje obaveštenja u slučaju logovanja ili registracije
- Dinamičko ispisivanje grešaka

product.blade.php:

- Funkcionalnost usklađenog filtriranja po kategoriji, sortiranja po ceni i paginacije putem ajax-a
- Klikom na ime ili sliku proizvoda otvara se detaljan prikaz informacija o konkretnom proizvodu

logovanje i registracija:

- Dinamicko ispisivanje obaveštenja o uspešnosti logovanja/registracije na početnoj strani

admin.blade.php:

- Dinamicko ispisivanje svih stavki vezanih za admin panel

users.blade.php:

- Funkcionalnost usklađenog sortiranja po id-ju korisnika, nazivu i momentu registracije, paginacija, pretraga po imenu korisnika – sve putem ajax-a
- Funkcionalnost validacije unetih podataka, ispis grešaka
- Funkcionalnost dodavanja novog korisnika, izmene, brisanja i prikaza aktivnosti konkretnog korisnika

products.blade.php:

- Funkcionalnost usklađenog sortiranja po id-ju proizvoda, nazivu i ceni, paginacija, pretraga po nazivu proizvoda – sve putem ajax-a
- Funkcionalnost dodavanja novog proizvoda, izmene, brisanja putem ajaxa konkretnog proizvoda
- Funkcionalnost validacije unetih podataka, ispis grešaka

activity.blade.php:

- Funkcionalnost usklađenog filtriranja po datumu aktivnosti i paginacije – sve putem ajax-a
- Funkcionalnost brisanja putem ajaxa konkretne aktivnosti odabranog korisnika

user.blade.php:

[Type here]

[Type here]

- Funkcionalnost izmene profila ulogovanog korisnika, konkretno izmena korisničkog imena, lozinke i e-maila, kao i deaktivacija naloga, nakon čega se korisnik automatski logout-uje i više ne može da se uloguje sa starim kredencijalima
- Funkcionalnost validacije unetih podataka, ispis grešaka

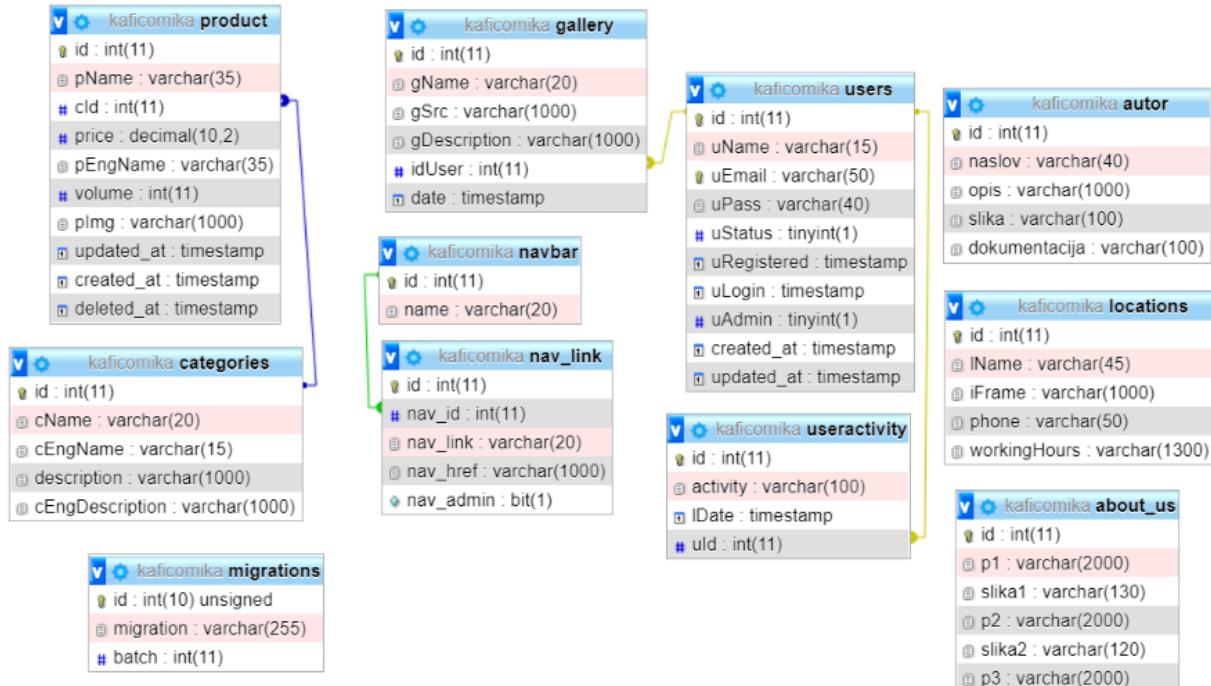
[Type here]

[Type here]

### 1.3 Template

Koristila sam svoj template

### 1.4. Slika baze



## 2. Organizacija

### 2.1 Mapa sajta

```
<?xml version="1.0" encoding="utf-8"?>

<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">

    <url>
        <loc>https:// http://kaficomika.ml/public/</loc>
        <lastmod>2021-13-03</lastmod>
        <changefreq>monthly</changefreq>
        <priority>0.9</priority>
    </url>

    <url>
        <loc>https:// http://kaficomika.ml/public/</loc>
        <lastmod>2021-13-03</lastmod>
        <changefreq>monthly</changefreq>
        <priority>0.8</priority>
    </url>

    <url>
        <loc>https:// http://kaficomika.ml/public/</loc>
        <lastmod>2021-13-03</lastmod>
        <changefreq>monthly</changefreq>
        <priority>0.7</priority>
    </url>

    <url>
        <loc>https:// http://kaficomika.ml/public/</loc>
        <lastmod>2021-13-03</lastmod>
        <changefreq>yearly </changefreq>
        <priority>0.6</priority>
    </url>


```

[Type here]

[Type here]

```
<url>
  <loc>https:// http://kaficomika.ml/public/</loc>
  <lastmod>2021-13-03</lastmod>
  <changefreq>yearly</changefreq>
  <priority>0.5</priority>
</url>
<url>
  <loc>https:// http://kaficomika.ml/public/</loc>
  <lastmod>2021-13-03</lastmod>
  <changefreq>yearly</changefreq>
  <priority>0.4</priority>
</url>
</urlset>
```

[Type here]

[Type here]

## 2.2 Slike stranica i opisi funkcionalnosti

### 2.3.1 home.blade.php

KAFICOMIKA   POČETNA   O NAMA   MENI   LOKACIJE   KONTAKT   GALERIJA   PRIJAVA



**DOBRODOŠLI DŽEZVERI!**

SAJT NA KOM ČEŠ SAZNATI SVE O KAFI, UKUSIMA.

---

A MOŽETE NAS I KONTAKТИРАТИ

**PONUDA**



COPYRIGHT © KAFICOMIKA 2021

KAFICOMIKA   POČETNA   O NAMA   MENI   LOKACIJE   KONTAKT   GALERIJA   PRIJAVA

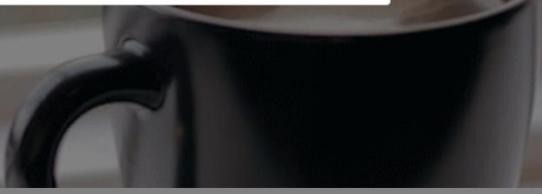
**PRIJAVA**   **REGISTRACIJA**

TVOJ EMAIL

TVOJA LOZINKA

**PRIJAVI SE**

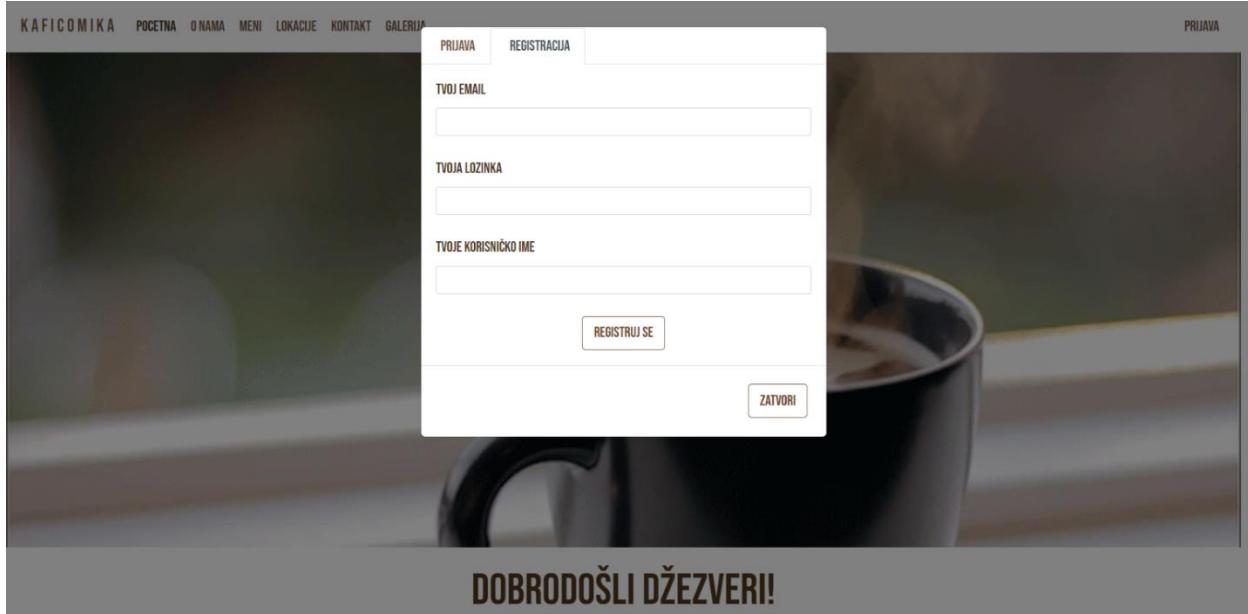
ZABORAVIO/LA SI LOZINKU? **ZATVORI**



**DOBRODOŠLI DŽEZVERI!**

[Type here]

[Type here]



Dinamički unet sadržaj footera i headera. Dinamički ispis obaveštenja i grešaka iz kontrolera, validacija.

[Type here]

[Type here]

## 2.3.2 product.blade.php

KAFICOMIKA POCETNA O NAMA MENU LOKACIJE KONTAKT GALERIJA

PRIJAVA

### ZAVIRITE U NAŠ MENI

SORTIRAJ PO:

CENA OPADAJUĆE

CENA RASTUĆE

KATEGORIJE:

Kafa - COFFEE

HLADNA KAFА - ICED COFFEE

FILTER KAFА - FILTER COFFEE

DODACI - EXTRAS

BILJNO MLEKO - VEGAN MILK

AROMЕ - AROMAS



HEMEKS IKS EL  
CHEMEX XL



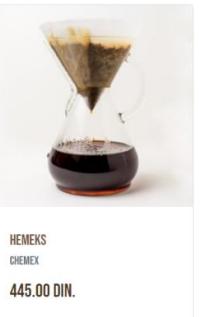
SIFON  
SIPHON



AEROSPRES  
AEROPRESS



FRANCUSKA PRESA  
FRENCH PRESS



HEMEKS  
CHEMEX



MOKA IKS EL  
MOCHA XL

1 2 3 4 5

COPYRIGHT © KAFICOMIKA 2021

KAFICOMIKA POCETNA O NAMA MENU LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA



HEMEKS IKS EL - CHEMEX XL

KOLIČINA: 500ML

CENA: 715.00 DIN.

KATEGORIJA: FILTER KAFА - FILTER COFFEE

FILTER KAFЕ DOBIVAMO PUTEM MEĐU-ČESTIĆNE DIFUZIJE. I IZ TOG RAZLOGA JE PRIPREMA ISTIH DUŽA NEGO PRIPREMA ESPRESA, MEĐUTIM, NIJE TO JEDINA RAZLICA IZMEU OVA DVA TIPOA EKSTRAKCIJE. FILTERI SADRŽE VIŠE KOFENA ZBOG DUŽEG VREMENA EKSTRAKCIJE, A SLABIJE TELO JER NEMA DIREKTNO FORSIRANOG ATMOSFERSKOG PRITiska NA KAFU TOKOM EKSTRAKCIJE. SAMIM TIM FILTER JE MANJE GORAK OD ESPRESA.

THE FILTERED COFFEE IS OBTAINED BY INTER-PARTICLE DIFFUSION. AND FOR THAT REASON, THE PREPARATION OF THE SAME IS LONGER THAN THE PREPARATION OF ESPRESSO, HOWEVER, THAT IS NOT THE ONLY DIFFERENCE BETWEEN THESE TWO TYPES OF EXTRACTION. FILTERS CONTAIN MORE CAFFEINE DUE TO A LONGER EXTRACTION TIME, AND A WEAKER BODY BECAUSE THERE IS NO DIRECTLY FORCED ATMOSPHERIC PRESSURE ON THE COFFEE DURING EXTRACTION. THEREFORE, THE FILTER IS LESS BITTER THAN ESPRESSO.

COPYRIGHT © KAFICOMIKA 2021

Na ovoj stranici dinamički su ispisani proizvodi. Usklađena funkcionalnost filtriranja po kategorijama, sortiranja po ceni i paginacije.

### 2.3.3 onama.blade.php

KAFICOMIKA POCETNA O NAMA MENI LOKACIJE KONTAKT GALERIJA

PRIJAVA

#### O NAMA

STATISTIKA GOVORI DA SE U SRBIJI DNEVNO POPJEVIŠE OD DVA MILIONA ŠOLJICA KAFE. VIŠE OD 70% POPULACIJE PIJE KAFU NA DNEVNOM NIVOU, A ČAK OVE TREĆINE NJIH POSLE POJNE POPUE KAFU IZ UŽIVANJA. NARAVNO, I DALJE SE NAVIŠE KONZUMIRAJ DOMAČA KAFU, ALJ JE GODINAMA UNAZAD SVE VIŠE ONIH KOI SE OKRENU NEKIM DRUGIM VISTAMA I NOVIM UKUSIMA. TAKO SU PRŽNICE KAFE, KOJE SU U SRBIJI BILE KARAKTERIŠĆENE ZA DEVEDESETE GODINE, NA POČETKU DRUGE DECENJE NOVOG MILENIJUMA BILE ZAMENJENE MESTIMA NA KOJIMA SE MOŽE POPITI ODLIČNA KAFU. JEDNO OD TAKVIH MESTA JE JE KAFICOMIKA.

PRVA KAFICOMIKA JE OTVORENA PRE NEŠTO VIŠE OD TRI GODINE NA DORČOLU, A SADA IH JE VEĆ OSAM (SEDEM U BEogradu, uključujući i aerodrom "NIKOLA TESLA", KAO I JEDNA U NOVOM SADU). KAKO ISTIĆU JANO VECINA MLAD BREND, ZA MANJE DO TRI GODINE DOKAZANIM KVALITETOM I KONSTANTNIM INOVACIJAMA POSTALI SU JEĐAN OD NAJBRAŽE RASTUĆIH LANACA KAFETERIJA U OVOM DELU EvROPE. U SVOM ASORTIMANU ZA SADA NUDE VIŠE OD 15 JEDINSTVENIH VISTAVA KAFE SA SVIH STRANA SVETA, MEĐU KOJIMA SU I VEDMA RETKI I ESKLUSIVNI UKUSI. O TOME SMO RAZGOVARALI SA NIKOLOM VUKOMANOVIĆEM, DIREKTOROM KAFETERIJA, U NOVOOTVORENOM "MAGAZINU 1907" U SRCU BEograda.



PRVA KAFICOMIKA JE OTVORENA POČETKOM FEBRUARA 2014. GODINE, A SADA IH IMAMO NA OSAM LOKACIJA U SRBIJI. IDEJA JE POTEKLA OD TOGA ŠTO U BEogradu nije bilo mnogo mesta na kojima biste popili kvalitetnu kafu, a druge strane smo snimili da ljudi u Srbiji, koliko god kafe da pišu, ne znaju mnogo o njoj, i uglavnom neprestano konzumiraju samo jednu vrstu. Htelj smo da ludima približimo uživanje u kafi i počeli smo sa četiri, pet vrsta, a ideja je bila da prve godine svakog meseca predstavimo po jednu novu vrstu kafe. Sve se dogodilo jako brzo, i mi smo za svega nekoliko meseci u ponudu uvrstili još deset vrsta kafe sa raznih strana sveta. Sada, tri godine kasnije, možemo reći da gosti imaju svoje omiljene ukuse u Kaficomiki.



IZA KAFICOMIKE STOJI OZBILJAN TIM LJUDI KOJI SVAKODNEVNO ISTRAŽUJE TRŽIŠTE KAFE. NAMA JE POSEBNO ZADOVOLJSTVO što radimo sa malim plantažama iz svih krajeva sveta i znamo da se neko u potpunosti posvetio izgajavanju određene vrste kafe. Osim što saradujemo sa plantažama, veliku pažnju poslanjamo saradnji sa najboljima iz sveta kafe, koji nam dodatno sugeriraju koja svetska kafa obavezno mora da nade na našem stolu. takođe, pravimo i posebne blendove, sami mešamo određene vrste kafe kako bismo došli do novog ukusa.

KADA ŽELITE DA RADITE KAO BARISTA LIJ KOMBAR U KAFETERIJU, MORATE PROći NIZ EDUKACIJA, UPoznati se sa kompletom ponudom - svim kafama, čajevima i hrannom koja se služi u kafeterijama. Osoblje mora znati sve o kafama, do naistinu detalja, kako bi uvek mogli da preporuče gostu unut po mjeri. Svakogdjevi radimo na tome da edukujemo osoblje na pravi način kako bismo uvek imali spremne odgovore za goste. Tako su naši baristi apsolutni naučnici u svom poslu i možete biti sigurni da nećete pogrešiti kada vam oni něto preloži.



KAFICOMIKA JE MNOGO VIŠE OD OBČINOG LOKALA U KOJU DOLAZITE DA SAMO POPIJETE KAFU. Ona predstavlja jedinstven koncept, koji svakome osim kvalitetne usluge pružači ležernost i hranu. Kao i mate same i svoju dnevnu sobu, od samog početka, kada smo otvorili kafeteriju na Dorćolu - koja je kretnula kao malo mesto za ljude iz kraja, a proširila se na cel grad - obraćala se pažnja na svaki detalj, do toga da kvalitet kafe bude stalno isti i da ona uvek bude sveža, do toga da gosti bude uslužen na vreme i da uživa u ambijentu. Verujem da je upravo to ono što nas razlikuje od drugih.



Sadržaj dinamički učitan iz baze.



[Type here]

[Type here]

## 2.3.4 location.blade.php

KAFICOMIKA POCEVNA O NAMA MENI LOKACIJE KONTAKT GALERIJA

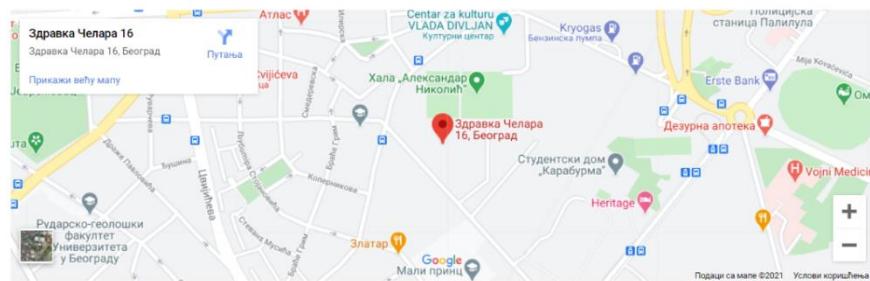
PRIJAVA

### RAZBUDITE SE KOD NAS!

#### ZDRAVKA ČELARA

TELEFON: 011/12345678

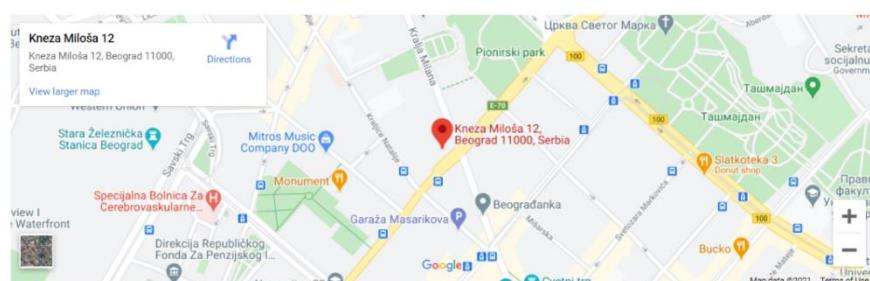
RADNO VРЕME: RADNI DAN: 09:00 - 22:00 VIKEND: 10:00 - 20:00



#### KNEZA MILOŠA 12

TELEFON: 011/12345678

RADNO VРЕME: RADNI DAN: 09:00 - 22:00 VIKEND: 10:00 - 20:00



COPYRIGHT © KAFICOMIKA 2021

Kompletan sadržaj dinamički učitan iz baze.

[Type here]

[Type here]

## 2.3.5 kontakt.blade.php

KAFICOMIKA   POČETNA   O NAMA   MENI   LOKACIJE   KONTAKT   GALERIJA   PRIJAVA

### KONTAKTIRAJTE NAS

---

EMAIL ADDRESS

MESSAGE

SUBMIT



COPYRIGHT © KAFICOMIKA 2021

Stranica namenjena kontaktiranju administratora.

[Type here]

[Type here]

## 2.3.6 galerija.blade.php

KAFICOMIKA [POCETNA](#) [O NAMA](#) [MENI](#) [LOKACIJE](#) [KONTAKT](#) [GALERIJA](#)

[PRIJAVA](#)

### NAŠ KUTAK - VAŠE SLIKE

DOBRA KOMUNIKACIJA PODSTICAJNA JE JEDNAND KAO I  
CRNA KAVA I JEDNAKO JE TEŠKA ZA SPAVANJE.  
— ANNE MURROW LYBBERGH [ABOUT COFFEE](#)

ŽELIŠ I TI DA TVOJE REMEK DELO POSTANE  
DEO NAŠE GALERIJE?

PRIJAVI SE!



19.03.2021. 15:56



3:49 PM



10.03.2021. 16:00



10.03.2021. 15:56



10.03.2021. 15:58



10.03.2021. 15:58



10.03.2021. 15:55



10.03.2021. 16:04



10.03.2021. 15:56



10.03.2021. 16:00



10.03.2021. 16:04



10.03.2021. 15:56

COPYRIGHT © KAFICOMIKA 2021



[Type here]

[Type here]

Dinamički unet sadržaj iz baze. Datum objave preformatiran iz timestamp formata.

### 2.3.7 admin.blade.php

The screenshot shows a web page titled "KAFICOMIKA". At the top, there is a navigation bar with links: POČETNA, O NAMA, MENI, LOKACIJE, KONTAKT, GALERIJA, ADMIN PANEL, MILICA98, and ODJAVA. Below the navigation bar, there is a horizontal menu with three items: ADMIN PANEL (which is highlighted), KORISNICI, and PROIZVODI. A central message reads: "DOBRODOŠLI U ADMIN PANEL, IZABERITE JEDNU OD PONUDENIH KATEGORIJA!". Below this message is a decorative graphic of several coffee beans arranged in a cluster. At the bottom of the page, there is a wide horizontal banner filled with many coffee beans. A small copyright notice "COPYRIGHT © KAFICOMIKA 2021" is visible at the bottom center of the banner.

Logovanjem sa administratorskim parametrima navigacioni meni dobija dva nova linka: admin i odjava. Admin vodi na admin panel koji poseduje podešavanja korisnika, proizvoda, galerije, lokacija i stranice „O nama“. Korisnici se mogu brisati i njihovi podaci editovati i postoji mogućnost dodavanja novog korisnika. Isto važi i za ostale oblasti sajta čijoj izmeni admin panel pruza pristup.

[Type here]

[Type here]

### 2.3.8 admin/users/index.blade.php

KAFICOMIKA

POCETNA O NAMA MENI LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA

ADMIN PANEL KORISNICI PROIZVODI

PRETRAGA

ID RASTUĆE  ID OPADAJUĆE  NAZIV OPADAJUĆE  NAZIV RASTUĆE  REGISTROVAN OPADAJUĆE  REGISTROVAN RASTUĆE

DODAJ KORISNIKA

ID	USERNAME	E-MAIL	LOZINKA	STATUS	REGISTRIRAN	ADMIN	OPCJE		
1	ADMINSAMUA	ADMIN@GMAIL.COM	21232f297a57a5a743894a0e4a801fc3	AKTIVAN	12.03.2021. 22:09:54	DA	<a href="#">AKTIVNOST</a>	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
2	ADMIN2	ADMIN2@GMAIL.COM	21232f297a57a5a743894a0e4a801fc3	AKTIVAN	13.03.2021. 01:05:03	DA	<a href="#">AKTIVNOST</a>	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
3	ADMIN	ADMIN3@GMAIL.COM	A24FED477ABAC2FB11500F5FC3C796FA	AKTIVAN	13.03.2021. 02:53:11	NE	<a href="#">AKTIVNOST</a>	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
5	MILICA98	MILICAP1598@GMAIL.COM	21232f297a57a5a743894a0e4a801fc3	AKTIVAN	13.03.2021. 21:38:52	DA	<a href="#">AKTIVNOST</a>	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
6	PEROSIJE	PERA@GMAIL.COM	21232f297a57a5a743894a0e4a801fc3	AKTIVAN	13.03.2021. 21:56:15	NE	<a href="#">AKTIVNOST</a>	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
7	MILICA	MILICA98@GMAIL.COM	21232f297a57a5a743894a0e4a801fc3	AKTIVAN	14.03.2021. 21:45:21	NE	<a href="#">AKTIVNOST</a>	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
10	STEFANOJLO	AMANSTEFAN@GMAIL.COM	71778f58a35f2c5c105571a1a3603574	AKTIVAN	14.03.2021. 22:06:53	NE	<a href="#">AKTIVNOST</a>	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>

1 2



Dinamički izlistani svi korisnici u tabelarnom prikazu.

Funkcionalnost pretrage username-a, filtriranja po imenu rastuće i opadajuće, id-ju i datumu registracije korisnika.

Datum objave preformatiran iz timestamp formata.

[Type here]

[Type here]

## 2.3.9 admin/users/create.blade.php

KAFICOMIKA POCETNA O NAMA MENI LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA

### KREIRAJ NOVOG KORISNIKA

USERNAME  
E-MAIL  
LOZINKA

ADMIN  
 OBİČAN KORISNIK

AKTIVAN NALOG  
 NEAKTIVAN NALOG

**SAČUVAJ**

COPYRIGHT © KAFICOMIKA 2021

## 2.3.10 admin/users/edit.blade.php

KAFICOMIKA POCETNA O NAMA MENI LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA

### IZMENA KORISNIKA - ADMIN2

ADMIN2  
ADMIN2@GMAIL.COM  
21232F297A5A743894A0E4A801FC3

ADMIN  
 OBİČAN KORISNIK

AKTIVAN NALOG  
 DEAKTIVIRAN NALOG

**IZMENI**

COPYRIGHT © KAFICOMIKA 2021

Forma za izmenu dinamički popunjena podacima iz baze. Validacija sprečava unos korisnika sa e-mailom koji već postoji u bazi. Dinamičko ispisivanje ispravnosti unosa novog korisnika.

[Type here]

[Type here]

## 2.3.11 admin/users/activity.blade.php

KAFICOMIKA POCESTNA O NAMA MENI LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA

ADMIN PANEL KORISNICI PROIZVODI

PRIKAZ AKTIVNOSTI PO DATUMU ZA KORISNIKA ADMINSAMJA

MM.MM.YYYY

ID	AKCIJA	DATUM	OPCIJE
5	KORISNIK SE ULOGOVAO	14.03.2021. 21:51:12	OBRISI
6	KORISNIK SE IZLOGOVAO	14.03.2021. 21:52:28	OBRISI
15	KORISNIK SE ULOGOVAO	14.03.2021. 22:15:25	OBRISI
16	KORISNIK SE IZLOGOVAO	14.03.2021. 22:16:27	OBRISI
20	KORISNIK SE ULOGOVAO	14.03.2021. 23:33:00	OBRISI
21	KORISNIK SE IZLOGOVAO	14.03.2021. 23:57:33	OBRISI
22	KORISNIK SE ULOGOVAO	15.03.2021. 00:02:08	OBRISI

1 2



COPYRIGHT © KAFICOMIKA 2021

Dinamički prikaz svih aktivnosti odabranog korisnika.

Funkcionalnost usklađenih paginacije, filtera po datumu i brisanja aktivnosti odrađeni putem ajax-a.

Datum preformatiran iz timestamp formata koji je došao iz baze.

[Type here]

[Type here]

## 2.3.12 admin/products/index.blade.php

KAFICOMIKA POCECTNA O NAMA MENI LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA

ADMIN PANEL KORISNICI PROIZVODI

PRETRAGA

ID RASTUĆE  ID OPADAJUĆE  NAZIV OPADAJUĆE  NAZIV RASTUĆE  CENA OPADAJUĆE  CENA RASTUĆE

DODAJ PROIZVOD

ID	NAZIV	ENGLESKI NAZIV	CENA	KATEGORIJA	OPCJE	
1	ESPRESSO	ESPRESSO	205.00 DIN.	KAFA	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
2	DUPLI ESPRESSO	DOUBLE ESPRESSO	285.00 DIN.	KAFA	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
3	PRODUZENI ESPRESSO	ESPRESSO LUNGO	205.00 DIN.	KAFA	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
4	RISTRETO	RISTRETTO	205.00 DIN.	KAFA	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
5	MAKIJATO	MACCHIATO	215.00 DIN.	KAFA	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
6	KAPUČINO	CAPPUCINO	235.00 DIN.	KAFA	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>
7	KAPUČINO IKS EL	CAPPUCINO XL	305.00 DIN.	KAFA	<a href="#">IZMENI</a>	<a href="#">OBRIŠI</a>

1 2 3 4



Dinamički prikaz svih proizvoda iz baze.

Funkcionalnost usklađenih pretrage, paginacije, sorta i brisanja proizvoda odrađeni putem ajax-a.

[Type here]

[Type here]

## 2.3.13 admin/products/create.blade.php

KAFICOMIKA POCESTNA O NAMA MENI LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA

### KREIRAJ NOVI PROIZVOD

NAZIV PROIZVODA

ENGLESKI NAZIV

CENA PROIZVODA

KAFĀ - COFFEE

HLAĐENA KAFĀ - ICED COFFEE

FILTER KAFĀ - FILTER COFFEE

DODACI - EXTRAS

BILJNO MLEKO - VEGAN MILK

AROME - AROMAS

ODABERI FOTOGRAFIJU PROIZVODA

CHOOSE FILE

NO FILE CHOSEN

SAČUVAJ

COPYRIGHT © KAFICOMIKA 2021

Inicijalno je selektovana prva kategorija. Dinamička obaveštenja o unosu.



[Type here]

[Type here]

## 2.3.14 admin/products/edit.blade.php

KAFICOMIKA POČETNA O NAMA MENI LOKACIJE GALERIJA ADMIN PANEL

MILICA98 ODJAVA

### IZMENA PROIZVODA - ESPRESO

ESPRESSO

ESPRESSO

205.00

KAFA - COFFEE

HLAĐENA KAFA - ICED COFFEE

FILTER KAFA - FILTER COFFEE

DODACI - EXTRAS

BILJNO MLEKO - VEGAN MILK

AROME - AROMAS

IZMENI FOTOGRAFIJU PROIZVODA

CHOOSE FILE

NO FILE CHOSEN

IZMENI

COPYRIGHT © KAFICOMIKA 2021

Forma za izmenu dinamički popunjena podacima iz baze. Validacija sprečava nepravilan unos. Dinamičko ispisivanje ispravnosti unosa novog proizvoda.

[Type here]

[Type here]

### 2.3.15 user/edit.blade.php

KAFICOMIKA POČETNA O NAMA MENI LOKACIJE KONTAKT GALERIJA ADMIN PANEL

MILICA98 ODJAVA

#### IZMENA NALOGA - MILICA98

MILICA98

MILICAP1598@GMAIL.COM

LOZINKA

AKTIVAN NALOG

DEAKTIVIRAN NALOG

IZMENI

COPYRIGHT © KAFICOMIKA 2021

Forma za izmenu dinamički popunjena podacima iz baze, izuzev lozinke. Validacija sprečava izmenu e-maila ako novi već postoji u bazi. Dinamičko ispisivanje ispravnosti unosa novog korisnika.

U slučaju da korisnik odabere deaktiviran nalog, automatski biva izlogovan i više se ne može ulogovati sa istim kredencijalima.

[Type here]

[Type here]

## 2.3.16 pages/autor.blade.php

KAFICOMIKA POČETNA O NAMA MENI LOKACIJE KONTAKT GALERIJA

PRIJAVA

O AUTORU - MILICA PAVLOVIĆ



JA SAM MILICA PAVLOVIĆ [BROJ INDEKSA 73/17]. DOLAZIM IZ BEOGRADA, STUDENTKINJA SAM TREĆE GODINE VISOKE ICT ŠKOLE KOJU SAM UPISALA NAKON ZAVRŠENE TRINAESTE BEOGRADSKE GIMNAZIJE. SAJT KAFICOMIKA PREDSTAVLJA MOJU REALIZACIJU JEDNE OD PREDISPITNIH OBAVEZA NA FAKULTETU IZ PREDMETA WEB PROGRAMIRANJE PHP2.

DOKUMENTACIJA

COPYRIGHT © KAFICOMIKA 2021

Dinamički unet sadržaj o autoru, dokumentacija.

### 3. Kodovi

#### 3.0 routes/web.php

```
<?php

namespace App\Http\Controllers;

use App\Http\Controllers\AdminPanel\UserController;
use Illuminate\Support\Facades\Route;

/*
|--------------------------------------------------------------------------
| Web Routes
|--------------------------------------------------------------------------
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Route::get('/', [HomeController::class, 'index'])->name('home');
Route::get('/onama', [OnamaController::class, 'index'])->name('onama');
Route::get('/kontakt', [KontaktController::class, 'index'])->name('kontakt');
Route::get('/admin', [AdminController::class, 'index'])->name('admin');
Route::get('/lokacije', [LokacijeController::class, 'index'])->name('lokacije');

Route::get('/product', [ProductController::class, 'index'])->name('product');
Route::get('/product/filtered', [ProductController::class, 'getFiltered']);
Route::get('/product/{id}', [ProductController::class, 'show'])->name('product.show');

Route::prefix('admin')->group(function () {
    Route::get('/product/filtered', [AdminPanel\ProductController::class, 'getFiltered']);
    Route::get('/users/filtered', [AdminPanel\UserController::class, 'getFiltered']);
    Route::get('/users/activity/{id}/filtered', [AdminPanel\UserController::class, 'getFilteredActivity']);
});

Route::get('/users/{id}/activity', [AdminPanel\UserController::class, 'activity'])->name('users.activity');
Route::get('/users/activity/{id}', [AdminPanel\UserController::class, 'destroyActivity'])->name('users.destroyActivity');

Route::resource('/products', AdminPanel\ProductController::class);
Route::resource('/users', AdminPanel\UserController::class);
Route::resource('/statistics', AdminPanel\StatisticsController::class);
```

[Type here]

[Type here]

```
};

Route::get('/galerija', [GalerijaController::class, 'index'])->name('galerija');
Route::get("/logout", [OsnovniController::class, 'logout'])->name("logout");
Route::post("/login", [OsnovniController::class, 'login'])->name("login");
Route::post("/register", [OsnovniController::class, 'register'])->name("register");
Route::get("/user", [UserControllers\UserController::class, 'index'])->name("user");
Route::put("/user/{id}/update", [UserControllers\UserController::class, 'update'])->name("user.update");
Route::get("/autor", [OsnovniController::class, 'autor'])->name("autor");
```

### 3.1 App/Http/Controllers

#### 3.1.1 App/Http/Controllers/OsnovniController.php

```
<?php

namespace App\Http\Controllers;

use App\Models\Autor;
use App\Models\Nav_link;
use App\Models\Navbar;
use App\Models\Useractivity;
use App\Models\Users;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;
use Illuminate\Support\Facades\Log;
use Illuminate\Support\Facades\Auth;

class OsnovniController extends Controller
{
    protected $data;
    protected $meni;
    protected $meniLink;
    protected $rola;

    public function construct() {
        $this->data['meni'] = Navbar::all();
        $this->data['linkovi'] = Nav_link::where('nav_admin', 0)->get();
        $this->data['adminLink'] = Nav_link::where('nav_admin', 1)->where('nav_href', 'admin')->get();
    }

    public function logPageAccessActivity($page) {
        try {
            if (session()->has("user"))
                $id = session("user")->getUserId();
            if (session()->has("admin"))
                $id = session("admin")->getUserId();

            Useractivity::create(["activity" => "Korisnik pristupio stranici"]);
        }
    }
}
```

[Type here]

[Type here]

```
$page", "uId" => $id]);
} catch (\Exception $e) {
    Log::error($e->getMessage());
}
}

public function autor()
{
    $this->data['autor'] = Autor::all();
    return view('pages.autor', $this->data);
}

public function logout(Request $request)
try{
    if(session()->has("user")){
        $id = session("user")->getUserId();
        $request->session()->remove("user");
    }

    if(session()->has("admin")){
        $id = session("admin")->getUserId();
        $request->session()->remove("admin");
    }

    Useractivity::create(["activity" => "Korisnik se izlogovao",
"uId" => $id]);

    return redirect()->route('home')->with("success", "Uspesano
logout");
} catch (\Exception $e){
    Log::error($e->getMessage());
    return redirect()->route('home')->with("error", "Neuspesan
logout");
}

}

public function login(Request $request){
    $request->all();
    Users::validateLogin($request);

    try{
        $user = Users::where("uEmail", $request->uEmail)->where("uPass",
md5($request->uPass))->where("uStatus", 1)->first();

        if($user->uAdmin == 0)
            $request->session()->put("user", $user);

        if($user->uAdmin == 1)
            $request->session()->put("admin", $user);

        Useractivity::create(["activity" => "Korisnik se ulogovao", "uId"
=> $user->id]);

        return redirect()->route('home')->with("success", "Uspesano
login");
    }
}
```

[Type here]

[Type here]

```
    }catch (\Exception $e){
        Log::error($e->getMessage());
        return redirect()->route('home')->with("error", "Doslo je do
greske prilikom logovanja.");
    }

}

public function register(Request $request){
    $request->all();
    //Users::validate($request);
    DB::beginTransaction();

    try{
        $pass = md5($request->uPass);

        $mailExixts = Users::where("uEmail", $request->uEmail)->first();

        if(!$mailExixts){
            $user = Users::create($request->except('uPass') + ["uPass" =>
$pass]);
            DB::commit();
            $request->session()->put("user", $user);
            Log::info("Uspesno ste se registrovali.");
            Useractivity::create(["activity" => "Korisnik se
registrovao", "uId" => $user->id]);
            return redirect()->route('home')->with("success", "Uspesno
ste se registrovali.");
        }else{
            Log::error("Korisnik sa unetim e-mailom vec postoji.");
            return redirect()->route('home')->with("error", "Korisnik sa
unetim e-mailom vec postoji.");
        }

    }catch (\Exception $e){
        Log::error($e->getMessage());
        return redirect()->route('home')->with("error", "Doslo je do
greske prilikom registracije.");
    }
}
```

### 3.1.2 App/Http/Controllers/HomeController.php

```
<?php

namespace App\Http\Controllers;

class HomeController extends OsnovniController
{
    public function index(){
        return view('pages.main.home', $this->data);
    }
}
```

[Type here]

[Type here]

### 3.1.3 App/Http/Controllers/OnamaController.php

```
<?php

namespace App\Http\Controllers;
use App\Models\About_us;
use Illuminate\Http\Request;

class OnamaController extends OsnovniController
{
    public function index(){
        $this->data['onama'] = About_us::all();
        $this->logPageAccessActivity("O nama");

        return view('pages.main.onama', $this->data);
    }
}
```

### 3.1.4 App/Http/Controllers/ProductController.php

```
<?php

namespace App\Http\Controllers;
use App\Models\Category;
use App\Models\Product;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

class ProductController extends OsnovniController{

    const PROD_PER_PAGE = 6;
    private $productsModel;

    protected $product;

    public function __construct()
    {
        parent::__construct();
        $this->productsModel = new Product();
    }

    public function index(){
        $this->logPageAccessActivity("Meni");

        $this->data['kategorije'] = Category::all();
        $this->data['proizvod'] = Product::orderBy('price', 'DESC')-
>paginate(self::PROD_PER_PAGE);

        return view('pages.product.index', $this->data);
    }

    public function show(Product $id){
        $this->logPageAccessActivity("Meni - Proizvod id: $id");
    }
}
```

[Type here]

[Type here]

```
$this->data["p"] = $id;
$this->data['k'] = Category::find($id->cId);

return view('pages.product.show', $this->data);
}

public function getFiltered(Request $request)
{
    $categories = $request->categories;
    $sortValue = $request->sortValue;
    $sortColumn = $request->sortColumn;
    $productName = $request->productName;

    $products = $this->productsModel->getFiltered($categories,
$sortValue, $sortColumn, $productName)->paginate(self::PROD_PER_PAGE);

    return response()->json($products);
}

}
```

### 3.1.5 App/Http/Controllers/LokacijeController.php

```
<?php

namespace App\Http\Controllers;

use App\Models\Location;
use Illuminate\Http\Request;

class LokacijeController extends OsnovniController
{
    public function index() {
        $this->data['loc'] = Location::all();
        $this->logPageAccessActivity("Lokacije");

        return view('pages.main.lokacije', $this->data);
    }
}
```

### 3.1.6 App/Http/Controllers/KontaktController.php

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class KontaktController extends OsnovniController
{
    public function index() {
        $this->logPageAccessActivity("Kontakt");

        return view('pages.main.kontakt', $this->data);
    }
}
```

[Type here]

[Type here]

```
}
```

### 3.1.1 App/Http/Controllers/GalerijaController.php

```
<?php

namespace App\Http\Controllers;
use App\Models\Gallery;

class GalerijaController extends OsnovniController{
    public function index(){
        $this->data['img'] = Gallery::all();
        $this->logPageAccessActivity("Galerija");

        return view('pages.gallery.index', $this->data);
    }
}
```

### 3.1.7 App/Http/Controllers/AdminController.php

```
<?php

namespace App\Http\Controllers;

use App\Models\Nav_link;
use Illuminate\Http\Request;

class AdminController extends OsnovniController
{
    public function index(){
        $this->data['adminPageLinks'] = Nav_link::all()->where('nav_admin', 1);

        $this->logPageAccessActivity("Admin panel");

        return view('pages.admin.index', $this->data);
    }
}
```

## 3.2 App/Http/Controllers/AdminPanel

### 3.2.1 App/Http/Controllers/ProductController.php

```
<?php

namespace App\Http\Controllers\AdminPanel;
use App\Http\Requests\StoreProductRequest;
use App\Http\Controllers\OsnovniController;
use App\Models\Nav_link;
use App\Models\Product;
use App\Models\Category;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Log;
```

[Type here]

[Type here]

```
use Illuminate\Support\Facades\DB;
use Mockery\Exception;

class ProductController extends OsnovniController{

    const PROD_PER_PAGE = 7;
    private $productsModel;

    protected $product;

    public function __construct()
    {
        parent::__construct();
        $this->productsModel = new Product();
    }

    public function index(Request $request){
        $this->logPageAccessActivity("Admin panel - Proizvodi");

        $this->data['adminPageLinks'] = Nav_link::all()->where('nav_admin', 1);

        $this->data['proizvod'] = Product::orderBy('id', 'asc')->paginate(self::PROD_PER_PAGE);

        $this->data['kategorije'] = Category::all();

        return view('admin.products.index',
            ['proizvod' => $this->data['proizvod'],
            'linkovi' => $this->data['linkovi'],
            'adminLink' => $this->data['adminLink'],
            'meni' => $this->data['meni'],
            'adminPageLinks' => $this->data['adminPageLinks']]);
    }

    public function store(StoreProductRequest $request) {
        DB::beginTransaction();

        try {
            Log::error($request);
            Log::error($request->pImg);

            $image = Product::uploadImage($request->pImg);
            $proizvod = Product::create($request->except('pImg') + ["pImg" => $image]);
            DB::commit();

            // $proizvod->save();

            return redirect()->route('products.index')->with("success", "Uspesno dodavanje proizvoda.");
        } catch (Exception $e) {
            Log::error($e->getMessage());
        }
    }
}
```

[Type here]

[Type here]

```
        DB::rollBack();
        return redirect()->route('products.index')->with("error", "Doslo
je do greske prilikom dodavanja proizvoda.");
    }

}

public function create(){
    $this->logPageAccessActivity("Admin panel - Kreiraj proizvod");

    return view('admin.products.create', [
        'proizvodi' => Product::all(),
        'kategorije' => Category::all(),
        'adminLink' => $this->data['adminLink'],
        'linkovi' => $this->data['linkovi'],
        'meni' => $this->data['meni']
    ]);
}

public function edit($id){
    $this->logPageAccessActivity("Admin panel - Izmena proizvoda
id:$id");

    return view('admin.products.edit', [
        'proizvodi' => Product::findOrFail($id), //automatski vraca 404
ako ne postoji
        'kategorije' => Category::all(),
        'adminLink' => $this->data['adminLink'],
        'linkovi' => $this->data['linkovi'],
        'meni' => $this->data['meni']
    ]);
}

public function update(Request $request, $id){
    try {
        $data = $request->all();
        unset($data["_token"]);
        unset($data["_method"]);

        $product = Product::find($id);

        foreach($data as $key => $value){
            $product->$key = $value;
        }

        $product->save();
        return redirect()->route('products.index')->with("success",
"Proizvod je uspešno izmenjen.");
    } catch (Exception $e){
        Log::error($e->getMessage());
        return redirect()->route('products.index')->with("error", "Doslo
je do greske prilikom brisanja proizvoda.");
    }
}
```

[Type here]

[Type here]

```
}

public function destroy($id) {
    try {
        Product::destroy($id);
        return redirect()->route('products.index')->with("success",
"Uspesno brisanje proizvoda.");

    }catch (Exception $e){
        Log::error($e->getMessage()); //greska se upisuje u log fajl
        return redirect()->route('products.index')->with("error", "Doslo
je do greske prilikom brisanja proizvoda.");
    }
}

public function getFiltered(Request $request)
{
    $categories = $request->categories;
    $sortValue = $request->sortValue;
    $sortColumn = $request->sortColumn;
    $productName = $request->productName;

    $products = $this->productsModel->getFiltered($categories,
$sortValue, $sortColumn, $productName)->paginate(self::PROD_PER_PAGE);

    return response()->json($products);
}

}
```

### 3.2.2 App/Http/Controllers/UserController.php

```
<?php

namespace App\Http\Controllers\AdminPanel;
use App\Http\Requests\StoreProductRequest;
use App\Http\Controllers\OsnovniController;
use App\Http\Requests\StoreUserRequest;
use App\Models\Nav_link;
use App\Models\User;
use App\Models\Useractivity;
use App\Models\Users;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Log;
use Illuminate\Support\Facades\DB;
use Mockery\Exception;

class UserController extends OsnovniController{

    const PROD_PER_PAGE = 7;
    private $usersModel;
    private $activityModel;
    protected $user;

    public function __construct()
```

[Type here]

[Type here]

```
{  
    parent::__construct();  
    $this->usersModel = new Users();  
  
}  
  
public function index(){  
    $this->logPageAccessActivity("Admin panel - Korisnici");  
  
    $this->data['user'] = Users::orderBy('id', 'asc')-  
>paginate(self::PROD_PER_PAGE);  
  
    $this->data['adminPageLinks'] = Nav_link::all()->where('nav_admin',  
1);  
  
    return view('admin.users.index', $this->data);  
}  
  
public function show(){  
    return view('admin.users.show', $this->data);  
}  
  
public function store(StoreUserRequest $request){  
    DB::beginTransaction();  
  
    try {  
        $pass = md5($request->uPass);  
  
        $mailExixts = Users::where("uEmail", $request->uEmail)->first();  
  
        if(!$mailExixts){  
            Users::create($request->except('uPass') + ["uPass" =>  
$pass]);  
            DB::commit();  
            return redirect()->route('users.index')->with("success",  
"Uspesno dodavanje korisnika.");  
        }else{  
            return redirect()->route('users.index')->with("error",  
"Korisnik sa unetim e-mailom vec postoji.");  
        }  
  
    } catch (Exception $e){  
        Log::error($e->getMessage());  
    }  
  
}  
  
public function create(){  
    $this->logPageAccessActivity("Admin panel - Kreiraj korisnika");  
  
    $this->data['user'] = Users::all();  
    return view('admin.users.create', $this->data);  
}
```

[Type here]

[Type here]

```
public function edit($id) {
    $this->logPageAccessActivity("Admin panel - Edit korisnika id:$id");

    $this->data['user'] = Users::findOrFail($id);

    return view('admin.users.edit', $this->data);

}

public function update(Request $request, $id){
    try {
        $data = $request->all();
        unset($data["_token"]);
        unset($data["_method"]);

        $user = Users::find($id);

        $mailExixts = Users::where("uEmail", $request->uEmail)->first();

        if($user->id == $mailExixts->id || !$mailExixts) {
            if($data["uPass"] != $user->uPass){
                $data["uPass"] = md5($data["uPass"]);
            }
            foreach ($data as $key => $value) {
                $user->$key = $value;
            }

            $user->save();
            return redirect()->route('users.index')->with("success",
"Korisnik je uspešno izmenjen.");
        } else{
            return redirect()->route('users.index')->with("error",
"Korisnik sa unetim e-mailom vec postoji.");
        }
    } catch (Exception $e){
        Log::error($e->getMessage());
        return redirect()->route('users.index')->with("error", "Neuspešna
izmena korisnika.");
    }
}

public function destroy($id) {
    try {
        Users::destroy($id);
        return redirect()->route('users.index')->with("success", "Uspesno
brisanje korisnika.");

    } catch (Exception $e){
        Log::error($e->getMessage());
        return redirect()->route('users.index')->with("error", "Doslo je
do greske prilikom brisanja korisnika.");
    }
}

public function getFiltered(Request $request)
{
```

[Type here]

[Type here]

```
$sortValue = $request->usrSortValue;
$sortColumn = $request->usrSortColumn;
$uName = $request->uName;

$users = $this->usersModel->getFiltered($sortValue, $sortColumn,
$uName)->paginate(self::PROD_PER_PAGE);

return response()->json($users);
}

public function getFilteredActivity(Request $request) {
    dd($request);
    $date = $request->dateOfActivity;
    $activities = $this->activityModel->getFiltered($date)-
>paginate(self::PROD_PER_PAGE);

    return response()->json($activities);
}

public function destroyActivity($id) {
    try {
        Useractivity::destroy($id);
        return redirect()->route('users.activity')->with("success",
"Uspesno brisanje aktivnosti.");
    } catch (Exception $e) {
        Log::error($e->getMessage());
        return redirect()->route('users.activity')->with("error", "Doslo
je do greske prilikom brisanja aktivnosti.");
    }
}

public function activity($id = null) {
    if (!$id)
        return redirect()->route('users.index')->with("error", "Korisnik
za prikaz aktivnosti nije pronadjen.");

    $this->data['user'] = Users::findOrFail($id);
    $this->data['activity'] = Useractivity::where("uId", $id)-
>paginate(self::PROD_PER_PAGE);
    $this->data['adminPageLinks'] = Nav_link::all()->where('nav_admin',
1);

    return view('admin.users.activity', $this->data);
}
}
```

### 3.2.3 App/Http/Controllers/LocationController.php

[Type here]

[Type here]

### 3.2.4 App/Http/Controllers/GalleryController.php

## 3.3 App/Http/Controllers/UserControllers

### 3.3.1 App/Http/Controllers/UserController.php

```
<?php

namespace App\Http\Controllers\UserControllers;
use App\Http\Controllers\OsnovniController;
use App\Models\Users;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Log;
use Mockery\Exception;

class UserController extends OsnovniController
{
    protected $user;

    public function index() {
        return view('pages.user.index', $this->data);
    }

    public function update(Request $request, $id){
        //dd($id);
        //Users::validateLogin($request);
        try {
            $data = $request->all();
            unset($data["_token"]);
            unset($data["_method"]);
            //dd($data["uStatus"]);

            if($data["uPass"]){
                $data["uPass"] = md5($data["uPass"]);
            }else{
                unset($data["uPass"]);
            }

            if($data["uEmail"]){
                $data["uEmail"];
            }else{
                unset($data["uPass"]);
            }

            $user = Users::find($id);

            foreach ($data as $key => $value) {
                $user->$key = $value;
            }

            $user->save();
            if(session()->has("user"))
                $rola = "user";
            if(session()->has("admin"))
                $rola = "admin";
        }
    }
}
```

[Type here]

[Type here]

```
if($data["uStatus"] == 1){
    $request->session()->put($rola, $user);
    return redirect()->route('user')->with("success", "Profil je
uspešno izmenjen.");
} else{
    try{
        $request->session()->remove($rola);
        return redirect()->route('home')->with("success",
"Uspesna deaktivacija");
    } catch (\Exception $e){
        Log::error($e->getMessage());
        return redirect()->route('home')->with("error",
"Neuspesna deaktivacija");
    }
}

} catch (Exception $e){
    Log::error($e->getMessage());
    return redirect()->route('user')->with("error", "Greška pri
izmeni profila.");
}
}
```

### 3.4 App/Http/Requests

#### 3.4.1 App/Http/Requests/StoreProductController.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class StoreProductRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return true
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
```

[Type here]

[Type here]

```
{  
    return [  
        'pName' => 'required|min:3|max:35',  
        'price' => 'required|numeric',  
        'pEngName' => 'required|min:3|max:35',  
        'image' => 'required|image'  
    ];  
}  
  
public function messages(){  
    return [  
        'required' => 'Polje :attribute je obavezno.',  
        'pName.min' => 'Naziv mora imati više od :min karaktera.',  
        'pName.max' => 'Naziv mora imati manje od :max karaktera.',  
        'pEngName.min' => 'Engleski naziv mora imati više od :min  
karaktera.',  
        'pEngName.max' => 'Engleski naziv mora imati manje od :max  
karaktera.',  
        'numeric' => 'Polje :attribute mora biti broj.',  
        'image.image' => 'Postavljeni fajl mora biti tipa image.'  
    ];  
}
```

### 3.4.2 App\Http\Requests\StoreUserController.php

```
<?php  
  
namespace App\Http\Requests;  
  
use Illuminate\Foundation\Http\FormRequest;  
  
class StoreUserRequest extends FormRequest  
{  
    /**  
     * Determine if the user is authorized to make this request.  
     *  
     * @return true  
     */  
    public function authorize()  
    {  
        return true;  
    }  
  
    /**  
     * Get the validation rules that apply to the request.  
     *  
     * @return array  
     */  
    public function rules()  
    {  
        return [  
            'uName' => 'required|min:3|max:15',  
            'uEmail' => 'required|unique:users|min:11|max:50',  
            'uPass' => 'required'  
        ];  
    }  
}
```

[Type here]

[Type here]

```
}
```

```
public function messages() {
    return [
        'required' => 'Polje :attribute je obavezno.',
        'unique' => 'Polje :attribute mora biti jedinstveno.',
        'uName.min' => 'Ime ne sme biti kraće od :min karaktera.',
        'uName.max' => 'Ime ne sme biti duže od :max karaktera.',
        'uEmail.min' => 'Email ne sme biti kraći od :min karaktera.',
        'uEmail.max' => 'Email ne sme biti duži od :max karaktera.'
    ];
}
```

```
}
```

### 3.5 App/Http/Middleware

#### 3.5.1 App/Http/Middleware/IpLoggingMiddleware.php

```
<?php
```

```
namespace App\Http\Middleware;
```

```
use App\Providers\RouteServiceProvider;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\Log;
```

```
class IpLoggingMiddleware
{
    /**
     * Handle an incoming request.
     *
     * @param \Illuminate\Http\Request $request
     * @param Closure $next
     * @param string|null ...$guards
     * @return mixed
     */
    public function handle(Request $request, Closure $next)
    {
        Log::info("Http request from". $request->ip());

        return $next($request);
    }
}
```

### 3.6 App/Models

#### 3.6.1 App/Models/About\_us.php

```
<?php
```

```
namespace App\Models;
```

[Type here]

[Type here]

```
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\DB;

class About_us extends Model
{
    protected $table = 'about_us';
    public $timestamps = 'false';
    protected $primaryKey = 'id';
    use HasFactory;
    public function aboutUs() {
        return DB::table('about_us')->get();
    }
}
```

### 3.6.2 App/Models/Autor.php

```
<?php

namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\DB;

class Autor extends Model
{
    protected $table = 'autor';
    public $timestamps = 'false';
    protected $primaryKey = 'id';
    use HasFactory;

}
```

### 3.6.3 App/Models/Category.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Category extends Model
{
    use HasFactory;

    protected $table = 'categories';
    public $timestamps = 'false';
    protected $primaryKey = 'id';

    public function product(){
        return $this->hasMany(Product::class, 'id', 'cId');
    }
}
```

[Type here]

[Type here]

### 3.6.4 App/Models/Gallery.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\DB;

class Gallery extends Model
{
    protected $table = 'gallery';
    public $timestamps = 'false';
    protected $primaryKey = 'id';
    use HasFactory;
    public function getGallery() {
        return DB::table('gallery')->get();
    }
}
```

### 3.6.5 App/Models/Location.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\DB;

class Location extends Model
{
    protected $table = 'locations';
    public $timestamps = 'false';
    protected $primaryKey = 'id';
    use HasFactory;
    public function getLocations() {
        return DB::table('locations')->get();
    }
}
```

### 3.6.6 App/Models/Nav\_link.php

```
<?php

namespace App\Models;

use Illuminate\Support\Facades\DB;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Nav_link extends Model
```

[Type here]

[Type here]

```
{  
    protected $table = 'nav_link';  
    public $timestamps = 'false';  
    protected $primaryKey = 'id';  
    use HasFactory;  
  
    public function navbar(){  
        return $this->belongsTo(Navbar::class);  
    }  
}
```

### 3.6.7 App/Models/Navbar.php

```
<?php  
  
namespace App\Models;  
  
use Illuminate\Database\Eloquent\Factories\HasFactory;  
use Illuminate\Database\Eloquent\Model;  
use Illuminate\Support\Facades\DB;  
  
class Navbar extends Model  
{  
    use HasFactory;  
  
    protected $table = 'navbar';  
    public $timestamps = 'false';  
    protected $primaryKey = 'id';  
  
    public function nav_link(){  
        return $this->hasMany(NavLink::class);  
    }  
}
```

### 3.6.8 App/Models/Product.php

```
<?php  
  
namespace App\Models;  
  
use Illuminate\Database\Eloquent\Factories\HasFactory;  
use Illuminate\Database\Eloquent\Model;  
use Illuminate\Support\Facades\DB;  
use Illuminate\Support\Facades\Storage;  
use Illuminate\Support\Facades\Log;  
  
class Product extends Model  
{  
    use HasFactory;  
  
    protected $table = 'product';  
    public $timestamps = 'false';  
    protected $primaryKey = 'id';
```

[Type here]

[Type here]

```
protected $fillable = ["pName", "price", "volume", "pEngName", "pImg",  
"cId"];  
  
public function categories() {  
    return $this->belongsTo(Category::class, 'cId', 'id');  
}  
  
public static function uploadImage($image) {  
    Log::error($image);  
    $path = Storage::disk('public')->putFile('assets/img/product',  
$image);  
    $exploded = explode('/', $path);  
    return $exploded[count($exploded) - 1];  
}  
  
public static function deleteImage($image) {  
    Storage::disk('public')->delete('assets/img/product/' . $image);  
}  
  
public static function validate($request) {  
    $request->validate([  
        'pName' => 'required|min:3|max:35',  
        'price' => 'required|numeric',  
        'pEngName' => 'required|min:3|max:35',  
        'image' => 'required|image'  
    ], [  
        'required' => 'Polje :attribute je obavezno.',  
        'pName.min' => 'Naziv mora imati više od :min karaktera.',  
        'pName.max' => 'Naziv mora imati manje od :max karaktera.',  
        'pEngName.min' => 'Engleski naziv mora imati više od :min  
karaktera.',  
        'pEngName.max' => 'Engleski naziv mora imati manje od :max  
karaktera.',  
        'numeric' => 'Polje :attribute mora biti broj.',  
        'image.image' => 'Postavljeni fajl mora biti tipa image.'  
    ]);  
}  
  
public function getFiltered($categories, $sortValue, $sortColumn,  
$productName) {  
    $products = Product::with('categories');  
  
    if(is_array($categories)) {  
        /*  
         * $products->whereHas('categories', function ($query)  
use ($categories) {  
            return $query->whereIn('id', $categories);  
        });*/  
        $products->whereIn('cId', $categories);  
    }  
    if($sortValue && $sortColumn) {  
        $products->orderBy($sortColumn, $sortValue);  
    }  
  
    if($productName) {  
        $products->where('pName', 'like', '%' . $productName . '%');  
    }  
}
```

[Type here]

[Type here]

```
        }

        return $products;
    }

}
```

### 3.6.9 App/Models/Useractivity.php

```
<?php

namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Useractivity extends Model
{
    use HasFactory;

    protected $table = "useractivity";
    public $timestamps = 'false';
    const UPDATED_AT = null;
    const CREATED_AT = null;
    protected $primaryKey = 'id';

    protected $fillable = ['activity', 'lDate', 'uId'];

    public function users()
    {
        return $this->belongsTo(User::class, 'uId', 'id');
    }

    public function getFiltered($date)
    {
        $activities = Useractivity::with('users');

        if ($date) {
            $activities->where('lDate', $date);
        }

        return $activities;
    }
}
```

### 3.6.10 App/Models/Users.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Query\Builder;
```

[Type here]

[Type here]

```
class Users extends Model
{
    use HasFactory;

    protected $table = 'users';
    public $timestamps = 'false';
    protected $primaryKey = 'id';

    protected $fillable = ['uName', 'uEmail', 'uPass'];

    public function useractivity()
    {
        return $this->hasMany(Useractivity::class, 'id', 'uId');
    }

    public static function validateLogin($request)
    {
        $request->validate([
            'uEmail' => 'required|min:11|max:50',
            'uPass' => 'required|min:3|max:50'
        ], [
            'required' => 'Polje :attribute je obavezno.',
            'uEmail.min' => 'Email ne sme biti kraći od :min karaktera.',
            'uEmail.max' => 'Email ne sme biti duži od :max karaktera.',
            'uPass.min' => 'Lozinka ne sme biti kraća od :min karaktera.',
            'uPass.max' => 'Lozinka ne sme biti duža od :max karaktera.'
        ]);
    }

    public static function validateStore($request)
    {
        $request->validate([
            'uName' => 'required|min:3|max:15',
            'uEmail' => 'required|unique:users|min:11|max:50',
            'uPass' => 'required|min:3|max:50'
        ], [
            'required' => 'Polje :attribute je obavezno.',
            'unique' => 'Polje :attribute mora biti jedinstveno.',
            'uName.min' => 'Ime ne sme biti kraće od :min karaktera.',
            'uName.max' => 'Ime ne sme biti duže od :max karaktera.',
            'uEmail.min' => 'Email ne sme biti kraći od :min karaktera.',
            'uEmail.max' => 'Email ne sme biti duži od :max karaktera.',
            'uPass.min' => 'Lozinka ne sme biti kraća od :min karaktera.',
            'uPass.max' => 'Lozinka ne sme biti duža od :max karaktera.'
        ]);
    }

    public function getFiltered($sortValue, $sortColumn, $uName)
    {
        $users = Users::with('useractivity');

        if ($sortValue && $sortColumn) {
            $users->orderBy($sortColumn, $sortValue);
        }

        if ($uName) {

```

[Type here]

[Type here]

```
    $users->where('uName', 'like', '%' . $uName . '%');

}

return $users;
}

public function getUserId()
{
    return $this->id;
}

}
```

### 3.6.11 App/Models/

## 3.7 public

### 3.7.1 public/assets/css/style.css

```
*{
    margin: 0;
    padding: 0;
    text-decoration: none;
    list-style-type: none;
    box-sizing: border-box;
    font-family: 'Bebas Neue', cursive;
    /*font-family: 'Raleway', sans-serif;*/
    font-size: 18px;
}

body, .jumbotron{
    background-color: white;
    color: #4f361f;
}

a{
    color: #775847;
    text-decoration: none;
}

a:hover{
    color: #4f361f;
    text-decoration: none;
}

a .active{
    color: #372311 !important;
    font-weight: bold;
    text-decoration: none;
}
nav
{
```

[Type here]

[Type here]

```
position: relative;
overflow: auto;
}

.logo{
    width: 4%;
}

.navbar{
    margin-bottom: 0;
    border-radius: 0;
    /*background-color: rgba(21, 14, 2, 0.5);*/
    /*position: fixed;*/
    z-index: 1030;
    width: 100%;
}

.navbar-toggler-icon {
    background-image: url("data:image/svg+xml; charset=utf8, %3Csvg viewBox='0 0 32 32' xmlns='http://www.w3.org/2000/svg'%3E%3Cpath stroke='rgba(79, 54, 31, 0.5)' stroke-width='2' stroke-linecap='round' stroke-miterlimit='10' d='M4 8h24M4 16h24M4 24h24'/%3E%3C/svg%3E") !important;
}

.navbar-toggler {
    border-color: #775847 !important;
}

.btn-outline-secondary:hover, .list-group-item:hover{
    background-color: #775847 !important;
    border-color: #4f361f;
    color: rgb(255, 255, 255) !important;
}
.btn-outline-secondary, .border-secondary{
    color: #6e4423;
    border-color: #775847;
}
.nav-link, .greska{
    color:#775847;
}
.nav-link:hover{
    color:#775847;
}

.nav-pills .active{
    background-color: rgb(255, 255, 255) !important;
    color: #412e1a !important;
    border-color: #4f361f !important;
}

.ispodMenija{
    padding-top: 10%;
}

.levo{
    float: left;
```

[Type here]

[Type here]

```
}

.desno{
    float: right;
}
#slide{
    background-image: url('../img/s5.jpg');
    background-repeat: no-repeat;
    padding-top: 0;
    width: 100%;
    background-size: 100%;

    display: block;
    position: static;
    background-position: center;
}

#image-inner {
    display: block;
    height: 0;
    padding-top: 40%;
}

.jumbotron{
    text-transform: uppercase;
    padding: 2% !important;
}

.pagination > li > a
{
    color: #6e4423;
    border-color: #6e4423;
}

.pagination > li > a:focus,
.pagination > li > a:hover,
.pagination > li > span:focus,
.pagination > li > span:hover
{
    background-color: #775847;
    border-color: #6e4423;
    color: rgb(255, 255, 255);
    box-shadow: unset;
}

.pagination > .active > a
{
    background-color: #6e4423;
    border-color: #6e4423;
    color: rgb(255, 255, 255);
}

.pagination > .active > a:hover
{
    color: #775847;
    border-color: #6e4423;
```

[Type here]

[Type here]

```
}

.page-item.active .page-link{
    background-color: #6e4423;
    border-color: #6e4423;
    color: rgb(255, 255, 255);
}

.page-item.active .page-link:hover{
    color: #775847;
    border-color: #6e4423;
}

.footer{
    height: 5%;
    background-image: url('../img/ftr.jpg');
    background-repeat: repeat-x;
    background-position: center;
    width: 100%;
}

.cleaner{
    clear: both;
}

.bg-secondary{
    background-color: #6e4423 !important;
}
```

### 3.7.2 public/assets/js/main.js

```
window.onload = function () {

    url=window.location.href+"/";

    if(url.indexOf("product") != -1 || url.indexOf("admin/product") != -1) {

        console.log("usao u product ili admin product")

        $(".page-link").click(showMore);
        $(".categories").click(sortAndFilter);
        $(".sort").click(sortAndFilter);
        $(document).on("click",".deleteProduct", deleteProduct);

        $(".search").keyup(search);

    }

    if(url.indexOf("admin/users") != -1) {

        console.log("usao u users")
    }
}
```

[Type here]

[Type here]

```
$(".page-link").click(showMoreUsers);
$(".sort").click(sortAndFilterUsers);
$(document).on("click",".deleteUser", deleteUser);

$(".search").keyup(searchUsers);

}

if(url.indexOf("activity") != -1) {
    console.log("usao u activity")
    $(".page-link").click(showMoreActivities);
    $(".date").change(showActivitiesByDate);
    $(document).on("click",".deleteActivity", deleteActivity);
}

};

/****************************************** AKTIVNOSTI *****/
*****/


var dateOfActivity;
var page;

function showMoreActivities(e) {
    e.preventDefault();
    page = this.innerText;
    getFilteredActivities(page, dateOfActivity);
    $(".deleteActivity").click(deleteActivity);
}

function showActivitiesByDate() {
    dateOfActivity = $(this).val();
    getFilteredActivities(1, dateOfActivity);
}

function getFilteredActivities(page, dateOfActivity) {

    const caller = arguments.callee.caller.name;
    console.log(caller);
    $.ajax({
        url: "users/activity/1/filtered",
        method: "GET",
        data: {
            page, dateOfActivity
        },
        dataType: "JSON",
        success: function (response) {

            showActivitiesForAdmin(response.data);

            if(caller == 'showActivitiesByDate') {
                changePagination(response.last_page, response.current_page,
showMoreActivities);
            }
            if(caller == 'showMoreActivities') {
```

[Type here]

[Type here]

```
        changePagination(response.last_page, response.current_page,
showMoreActivities);
            changeActivePageLink(response.current_page);
        }
    },
error: function (err, xhr, data) {
    console.log(err);
    console.log(data);

}
} );
}

function deleteActivity() {
let id = $(this).val();
let csrf = document.querySelector('meta[name="csrf-
token"]').getAttribute('content');

$.ajax({
url: "/users/activity/"+id,
method: "POST",
headers: {
"Accept": "application/json"
},
data: {
"_token": csrf,
"_method": "DELETE"
},
success: function () {
$("#a_id "+id).remove();
$(".greskaAdmin").html("<div class='alert alert-success my-
3'>Uspesno brisanje aktivnosti.</div>")
},
error: function (err, xhr, data) {
console.log(err);
console.log(xhr);
console.log(data);
$(".greskaAdmin").html("<div class=\\"alert alert-danger my-
3\\>Doslo je do greske prilikom brisanja aktivnosti.</div>")
}
});
}

function showActivitiesForAdmin(products) {
let html = "";

function ispisDatum(datum) {
d = new Date(datum);
var dat = d.getDate();
var mes = d.getMonth() + 1;
var god = d.getFullYear();

var s = d.getSeconds();
var m = d.getMinutes();
var h = d.getHours();
```

[Type here]

[Type here]

```
var dateStr = dat + "." + mes + "." + god+"."+ " "+ h +"："+m +"："+ s;
    return dateStr;
}
for(let p of products) {
    html += `
<tr id="a_id_${p.id}">
    <td>${p.id}</td>
    <td>${p.activity}</td>
    <td>${ispisDatum(p.lDate)}</td>
    <td>`;
    html+=`<input type="hidden" name="_token" value="${csrf}" />
            <button type="submit" class="btn btn-outline-secondary
deleteActivity" value="${p.id}">Obriši</button>`;
    html+=`</td>

</tr>`;
}
$("#adminAktivnosti").html(html);
}

/********************************************* KORISNICI *****/
*****/


var usrSortValue;
var usrSortColumn;
var uName;

function getFiltersForUsers(){
    usrSortValue = $("input[name='sort']:checked").val();
    usrSortColumn = $("input[name='sort']:checked").data('id');
}

function showMoreUsers(e) {
    getFiltersForUsers();
    e.preventDefault();
    page = this.innerText;
    getFilteredUsers(page, usrSortValue , usrSortColumn, uName);
    $(".deleteUser").click(deleteUser);
}

function sortAndFilterUsers() {
    getFiltersForUsers();
    getFilteredUsers(1, usrSortValue , usrSortColumn, uName);
}

function searchUsers(){
    getFiltersForUsers();
    uName = $(this).val();
    getFilteredUsers(1 ,usrSortValue ,usrSortColumn, uName);
}

function getFilteredUsers(page, usrSortValue , usrSortColumn, uName) {
    const caller = argumentscallee.caller.name; //koja funkcija je pozvala
    console.log(caller);
```

[Type here]

[Type here]

```
$.ajax({
    url: "users/filtered",
    method: "GET",
    data: {
        page, usrSortValue , usrSortColumn, uName
    },
    dataType: "JSON",
    success: function (response) {

        showUsersForAdmin(response.data);

        if(caller == 'sortAndFilterUsers' || caller=='searchUsers'){
            changePagination(response.last_page, response.current_page,
showMoreUsers);
        }
        if(caller == 'showMoreUsers'){
            changePagination(response.last_page, response.current_page,
showMoreUsers);
            changeActivePageLink(response.current_page);
        }
    },
    error: function (err, xhr, data) {
        console.log(err);
        console.log(data);

    }
));
}

function deleteUser(){
    let id = $(this).val();
    let csrf = document.querySelector('meta[name="csrf-
token"]').getAttribute('content');

    $.ajax({
        url: "users/"+ id,
        method: "POST",
        headers:{
            "Accept":"application/json"
        },
        data: {
            "_token": csrf,
            "_method": "DELETE"
        },
        success: function () {
            $("#u_id_"+id).remove();
            $(".greskaAdmin").html("<div class='alert alert-success my-
3'>Uspesno brisanje korisnika.</div>")
        },
        error: function (err, xhr, data) {
            console.log(err);
            console.log(xhr);
            console.log(data);
            $(".greskaAdmin").html("<div class=\"alert alert-danger my-
3\">Doslo je do greske prilikom brisanja korisnika.</div>")
        }
    });
}
```

[Type here]

[Type here]

```
});  
}  
  
function showUsersForAdmin(products) {  
    let html = "";  
  
    function ispisDatum(datum) {  
        d = new Date(datum);  
        var dat = d.getDate();  
        var mes = d.getMonth() + 1;  
        var god = d.getFullYear();  
  
        var s = d.getSeconds();  
        var m = d.getMinutes();  
        var h = d.getHours();  
  
        var dateStr = dat + "." + mes + "." + god + ":" + " " + h + ":" + m + ":" + s;  
        return dateStr;  
    }  
    for(let p of products) {  
        html += `|  
            <td>${p.id}</td>  
            <td>${p.uName}</td>  
            <td>${p.uEmail}</td>  
            <td>${p.uPass}</td>  
            <td>${p.uStatus == 1 ? "Aktivan" : "Deaktiviran"}</td>  
            <td>${ispisDatum(p.uRegistered)}</td>  
            <td>${p.uAdmin == 1 ? "Da" : "Ne"}</td>  
  
            <td colspan="2">  
                <a href="users/${p.id}/activity" class="btn btn-outline-secondary" role="button">Aktivnost</a>  
            </td>  
            <td colspan="">  
                <a href="users/${p.id}/edit" class="btn btn-outline-secondary" role="button">Izmeni</a>  
            </td>  
            <td>;  
            html+=`<input type="hidden" name="_token" value="${csrf}" />  
                  <button type="submit" class="btn btn-outline-secondary deleteUser" value="${p.id}">Obriši</button>`;  
            html+=`</td>  
            </tr>`;  
    }  
    $("#adminKorisnici").html(html);  
}  
  
/***************************** PROIZVODI *****/

```

[Type here]

[Type here]

```
var categories = [];
var sortValue;
var sortColumn;
var productName;

function getFilters(){
    categories = [];
    sortValue = $("input[name='sort']:checked").val();
    sortColumn = $("input[name='sort']:checked").data('id');
    $.each($("input[name='categories']:checked"), function() {
        categories.push($(this).val());
    });
}

function showMore(e){
    getFilters();
    e.preventDefault();
    page = this.innerText;
    getFiltered(page,categories,sortValue, sortColumn, productName);
    $(".deleteProduct").click(deleteProduct);
}

function sortAndFilter() {
    getFilters();
    getFiltered(1, categories,sortValue, sortColumn, productName);
}

function search() {
    getFilters();
    productName = $(this).val();
    getFiltered(1 ,categories,sortValue, sortColumn, productName);
}

function getFiltered(page, categories, sortValue, sortColumn, productName) {

    const caller = arguments.callee.caller.name; //koja funkcija je pozvala
    console.log(caller);
    $.ajax({
        url: "product/filtered",
        method: "GET",
        data: {
            page, categories, sortValue, sortColumn, productName
        },
        dataType: "JSON",
        success: function (response) {

            showProducts(response.data);
            showProductsForAdmin(response.data);

            if(caller == 'sortAndFilter' || caller=='search'){
                changePagination(response.last_page, response.current_page,
showMore); // ovde umesto show more stavis show more users/activity
                //changeActivePageLink(response.current_page); razmislijam sta
je jos trebalo da zapisem
                //odvoj kod
            }
        }
    })
}
```

[Type here]

[Type here]

```
        }
        if(caller == 'showMore'){
            changePagination(response.last_page, response.current_page,
showMore);
            changeActivePageLink(response.current_page);
        }
    },
    error: function (err, xhr, data) {
        console.log(err);
        console.log(data);
    }
});

}

function deleteProduct(){
    let id = $(this).val();
    let csrf = document.querySelector('meta[name="csrf-
token"]').getAttribute('content');

    $.ajax({
        url: "products/"+ id,
        method: "POST",
        headers:{
            "Accept":"application/json"
        },
        data: {
            "_token": csrf,
            "_method": "DELETE"
        },
        success: function () {
            $("#product_id_"+id).remove();
            $(".greskaAdmin").html("<div class='alert alert-success my-
3'>Uspesno brisanje proizvoda.</div>")
        },
        error: function (err, xhr, data) {
            console.log(err);
            console.log(xhr);
            console.log(data);
            $(".greskaAdmin").html("<div class=\\"alert alert-danger my-
3\\>Doslo je do greske prilikom brisanja proizvoda.</div>")
        }
    });
}

function changePagination(totalLinks, currentPage, onClick){
    let html = "";
    for(let i = 1; i <= totalLinks; i++){
        if(i != currentPage){
            html += `<li class="page-item"><a class="page-link" id="link${i}" data-page="${i}" href="#">${i}</a></li>`;
        }else{
            html += `<li class="page-item active"><a class="page-link" id =
"link${i}" data-page="${i}" href="#">${i}</a></li>`;
        }
    }
}
```

[Type here]

[Type here]

```
        }

    $(".pagination").html(html);
    $(".page-link").click(onClick);
}

function changeActivePageLink(currentPage) {
    $('.page-item').removeClass('active');
    $('#link' + currentPage).parent().addClass('active');
}

function showProducts(products) {
    let html = "";
    for(let product of products) {
        html += `
            <div class="col-lg-4 col-md-6 my-4">
                <div class="card h-100">
                    <a href="product/${product.id}"><img class="card-img-top"
src=""`;

        if(product.pImg != null) {
            html+=`assets/img/product/${product.pImg}`;
        } else{
            html+=`assets/img/default.jpg`;
        }

        html+=`" alt="${product.pName}"></a>
            <div class="card-body">
                <h4 class="card-title">
                    <a href="product/${product.id}">${product.pName}</a>
                    <p class="card-text"><small class="text-
muted">${product.pEngName}</small></p>
                </h4>
                <h5>${product.price} din.</h5>
            </div>
        </div>
    `;
    }
    $("#proizvodi").html(html);
}

let csrf = document.querySelector('meta[name="csrf-
token"]').getAttribute('content');

function showProductsForAdmin(products) {
    let html = "";
    for(let p of products) {
        html += `
            <tr id="product_id_${p.id}">
                <td>${p.id}</td>
                <td>${p.pName}</td>
                <td>${p.pEngName}</td>
                <td>${p.price} din.</td>
                <td>${p.categories.cName}</td>
            `;
    }
}
```

[Type here]

[Type here]

```
<td colspan="1">
    <a href="products/${p.id}/edit" class="btn btn-outline-secondary" role="button">Izmeni</a>

    </td>
    <td>;
        html+=`<input type="hidden" name="_token" value="${csrf}" />
        <button type="submit" class="btn btn-outline-secondary deleteProduct" value="${p.id}">Obriši</button>`;
    /*           </form>* /
        html+=`</td>

    </tr>`;
}
$( "#adminProizvodi" ).html( html );
}
```

### 3.8 resources/views/layouts

#### 3.8.1 resources/views/layouts/layout.blade.php

```
<!DOCTYPE html>
<html lang="en">
    @include('fixed.head')
<body>
<div class="wrapper">
    @include('fixed.nav')
    @yield('slider')
    <div class="container">
        @include('fixed.message')
        @yield('content')
    </div>
    @include('fixed.footer')
    @include('fixed.scripts')
</div>
</body>

</html>
```

#### 3.8.2 resources/views/layouts/admin.blade.php

```
<!DOCTYPE html>
<html lang="en">
@include('fixed.head')
<body>
<div class="wrapper">

    @include('fixed.nav')
    @if(session() ->has("admin"))
        @php $rola = "admin"; @endphp
        @if(request() ->routeIs('user'))
            <div class="container">
                <div class="row">
                    @include('fixed.message')
                    @yield('content')
                </div>
        
```

[Type here]

[Type here]

```
</div>

@else
    @include('fixed.adminPageLinks')
    <div class="container">
        <div class="row">
            @include('fixed.message')
            @yield('content')
        </div>
    </div>
@endif

@elseif(session() ->has("user"))
    @php $rola = "user"; @endphp

    <div class="container">
        <div class="row">
            @include('fixed.message')
            @yield('content')
        </div>
    </div>

@elseif(!session() ->has("user") && !session() ->has("admin"))
    <div class="container">
        <div class="row">
            @include('fixed.404')
        </div>
    </div>

@endif
    @include('fixed.footer')
    @include('fixed.scripts')

</div>
</body>

</html>
```

### 3.9 resources/views/fixed

#### 3.9.1 resources/views/fixed/404.blade.php

```
<div class="container padding">
    <div class="row text-center my-3">
        
    </div>
</div>
```

#### 3.9.2 resources/views/fixed/adminPageLinks.blade.php

```
<ul class="nav nav-tabs">
    <?php
        foreach($adminPageLinks as $h):
    ?>
```

[Type here]

[Type here]

```
<li class="nav-item">
    <a class="nav-link" {{request() -> routeIs($h->nav_href) ? 'active' : ''}} href="{{route($h->nav_href)}}">{{ $h->nav_link }}</a>
</li>
<?php endforeach; ?>
</ul>
```

### 3.9.3 resources/views/fixed/footer.blade.php

```
<!-- Footer -->
<footer class="py-5 footer">
    <div class="container">
        <p class="m-0 text-center"><a href="{{route('autor')}}">Copyright
&copy; KAFICOMIKA 2021</a></p>
    </div>
    <div class="modal fade" id="modallRForm" tabindex="-1" role="dialog"
aria-labelledby="myModalLabel" aria-hidden="true">
        <div class="modal-dialog cascading-modal" role="document">
            <div class="modal-content">
                <div class="modal-c-tabs">
                    <ul class="nav nav-tabs md-tabs tabs-2" role="tablist">
                        <li class="nav-item">
                            <a class="nav-link active" data-toggle="tab"
href="#panel7" role="tab"><i class="fas fa-user mr-1"></i>
                                Prijava</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link" data-toggle="tab"
href="#panel8" role="tab"><i class="fas fa-user-plus mr-1"></i>
                                Registracija</a>
                        </li>
                    </ul>
                    <div class="tab-content">

                        <!-- LOG IN
PANEL ----->
                        <div class="tab-pane fade in show active" id="panel7"
role="tabpanel">

                            <!-- LOG IN
FORMA ----->
                            <form action="{{route('login')}}" method="POST">
                                @csrf
                                <div class="modal-body mb-1">
                                    <div class="md-form form-sm mb-4">
                                        <i class="fas fa-envelope
prefix"></i>
                                        <input name="uEmail" type="email"
id="modallRInput10" class="form-control form-control-sm validate">
                                            <label data-error="wrong" data-
success="right" for="modallRInput10">Tvoj email</label>
                                        </div>

                                        <div class="md-form form-sm mb-4">
                                            <i class="fas fa-lock prefix"></i>
                                            <input name="uPass" type="password">
                                        </div>
                                    </div>
                                </div>
                            </form>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</footer>
```

[Type here]

[Type here]

```
id="modalLRInput11" class="form-control form-control-sm validate">
    <label data-error="wrong" data-
success="right" for="modalLRInput11">Tvoja lozinka</label>
    </div>
    @if($errors->any())
        <div class="alert alert-danger my-3">
            @foreach ($errors->all() as
$error)
                {{ $error }}<br/>
            @endforeach
        </div>
    @endif
    <div class="text-center mt-2">
        <input type="hidden"
name="prijavljen">
        <button class="btn btn-outline-
secondary">Prijava se</button>
    </div>
</form>
<div class="modal-footer">
    <div class="options text-center text-md-right
mt-1">
        <p>Zaboravio/la si <a
href="?page=Kontakt" class="crveni-text">lozinu?</a></p>
        </div>
        <button type="button" class="btn btn-outline-
secondary" data-dismiss="modal">Zatvori</button>
    </div>
</div>

<!--
REGISTRACIJA PANEL -----
-->
<div class="tab-pane fade" id="panel8"
role="tabpanel">

<!--
REGISTRACIJA FORMA -----
-->
<form action="{{route('register')}}" method="POST">
    @csrf
    <div class="modal-body">
        <div class="md-form form-sm mb-4">
            <i class="fas fa-envelope
prefix"></i>
            <label data-error="wrong" data-
success="right" for="modalLRInput12">Tvoj email</label>
            <input name="uEmail" type="email"
id="mailRegistracija modalLRInput12" class="form-control form-control-sm
validate">
            <small id="greskaMail" class="greska
form-text"></small>
        </div>
        <div class="md-form form-sm mb-4">
            <i class="fas fa-lock prefix"></i>
            <label data-error="wrong" data-
```

[Type here]

[Type here]

```
success="right" for="modalLRInput13">Tvoja lozinka</label>

id="passRegistracija modalLRInput13" class="form-control form-control-sm validate">
Lozinka mora imati bar 4 karaktera i broj.</small>
</small>
</div>

<div class="md-form form-sm mb-4">
<i class="fas fa-lock prefix"></i>
<label data-error="wrong" data-success="right" for="modalLRInput14">Tvoje korisničko ime</label>
<input name="uName" type="text" id="userRegistracija modalLRInput14" class="form-control form-control-sm validate">
</small>
</div>

@if($errors->any())


@foreach ($errors->all() as $error)
{{ $error }}<br/>
@endforeach
@endif

<div class="text-center form-sm mt-2">
<input type="hidden" name="registrovanje">
<input type="submit" class="btn btn-outline-secondary" value="Registruj se">
</div>
</div>
</form>

<div class="modal-footer">
<button type="button" class="btn btn-outline-secondary" data-dismiss="modal">Zatvori</button>
</div>
</div>
<!--
REGISTRACIJA KRAJ PANELA ----->
</div>
</div>
</div>
</div>
<div class="cleaner">
</div>


```

[Type here]

[Type here]

```
<!-- /.container -->
</footer>
```

### 3.9.4 resources/views/fixed/head.blade.php

```
<head>

    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no" />
    <meta name="description" content="@yield('description')" />
    <meta name="keywords" content="@yield('keywords')" />
    <meta name="csrf-token" content="{{ csrf_token() }}">

    <title>KAFICOMIKA - @yield('title')</title>

    <!-- Bootstrap core CSS -->

    <!-- Custom styles for this template -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-
beta1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP1"
crossorigin="anonymous">
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.cs
s" integrity="sha384-
ggOyR0iXcbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-
select@1.13.14/dist/css/bootstrap-select.min.css">
    <link rel="stylesheet" href="{{asset('assets/css/style.css')}}" type="text/css">
    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link
href="https://fonts.googleapis.com/css2?family=Bebas+Neue&display=swap"
rel="stylesheet">
    <link rel="shortcut icon" href="{{asset('assets/img/logo/logo.png')}}" class="logo">

</head>
```

### 3.9.5 resources/views/fixed/message.blade.php

```
<div class="greskaAdmin">
    @if(session()->has('success'))
        <div class="alert alert-success my-3">
            {{session()->get('success')}}
        </div>
    @endif

    @if(session()->has('error'))
        <div class="alert alert-danger my-3">
            {{session()->get('error')}}
        </div>
    @endif
</div>
```

[Type here]

[Type here]

### 3.9.6 resources/views/fixed/nav.blade.php

```
<div id="header">
    <nav class="navbar navbar-expand-lg">
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarTogglerDemo03" aria-controls="navbarTogglerDemo03" aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>

        <a class="navbar-brand" href="{{ route('home') }}>K A F I C O M I K
A</a>

        <div class="collapse navbar-collapse" id="navbarTogglerDemo03">
            <ul class="navbar-nav mr-auto mt-2 mt-lg-0">

                @foreach($linkovi as $link)
                <li class="nav-item nav-pills">
                    <a class="nav-link" {{ request()>routeIs($link->nav_href) ? 'active' : '' }} href="{{ route($link->nav_href) }}>{{$link->nav_link}}</a>
                </li>
                @endforeach
                @if(session()->has("admin"))
                @foreach($adminLink as $l)
                <li class="nav-item">
                    <a class="nav-link" {{ request()>routeIs($l->nav_href) ? 'active' : '' }} href="{{ route($l->nav_href) }}>{{$l->nav_link}}<span class="sr-only">(current)</span></a>
                </li>
                @endforeach
                @endif

            </ul>
            <ul class="nav navbar-nav ml-auto levo">
                @if(session()->has("user") || session()->has("admin"))
                <li><a href="{{ route("user") }}" class="nav-link nav-pills">{{ session("user") ? session("user")->uName : session("admin")->uName }}</a></li>
                <li><a href="{{ route("logout") }}" class="nav-link nav-pills">Odjava</a></li>
                @endif

                @if(!session()->has("user") && !session()->has("admin"))
                <li>
                    <a class="nav-link" href="#" data-toggle="modal" data-target="#modallRForm">Prijava<span class="sr-only"></span></a>
                </li>
                @endif
            </ul>
        </div>
    </nav>
</div>
```

[Type here]

[Type here]

### 3.9.7 resources/views/fixed/scripts.blade.php

```
<script src="https://code.jquery.com/jquery-3.1.1.min.js">
    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzOOrT7abK41JStQIAqVgRVzbzo5smXKp4YfRvH+8abTE1Pi6jizo"
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.j
s" integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1c1HTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap-
select@1.13.14/dist/js/bootstrap-select.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-
beta1/dist/js/bootstrap.bundle.min.js" integrity="sha384-
ygbV9kiqUc6oa4msXn9868pTtWMgiQaeYH7/t7LECLbyPA2x65Kgf800JFdroafW"
crossorigin="anonymous"></script>
<script type="text/javascript" src="{{ asset('assets/js/main.js') }}"></script>
```

### 3.10 resources/views/partials

#### 3.10.1 resources/views/fixed/product.blade.php

```
<div class="col-lg-4 col-md-6 my-4">
    <div class="card h-100">
        <a href="{{ route('product.show', [$p->id]) }}">pName }}"></a>
        <div class="card-body">
            <h4 class="card-title">
                <a href="{{ route('product.show', [$p->id]) }}">{{ $p-
>pName }}</a>
                <p class="card-text"><small class="text-muted">{{ $p-
>pEngName }}</small></p>
            </h4>
            <h5>{{ $p->price }} din.</h5>
        </div>

    </div>
</div>
```

### 3.11 resources/views/pages

#### 3.11.1 resources/views/pages/autor.blade.php

```
@extends("layouts.layout")
@section("content")
<div class="container">
```

[Type here]

[Type here]

```
<div class="row justify-content-center col-md-12 text-center my-3">
    <div class="row">

        <?php
        foreach($autor as $a):
        ?>
        <div class="col-md-12">
            <h5 class="card-title">{{ $a->naslov }}</h5>
            slika) }}" class="img-fluid w-
50" alt="{{ $a->naslov }}">
        </div>
        <div class="col-md-12">
            <div class="card-body my-3">

                <p>
                    {{ $a->opis }}
                </p>
            </div>
            <div class="row text-center padding">
                <div class="col-md-4 social padding"><a
href="https://www.instagram.com/comikomakeup/"><i class="fa fa-
instagram"></i></a></div>
                <div class="col-md-4 social padding"><a
href="https://www.facebook.com"><i class="fab fa-facebook"></i></a></div>
                <div class="col-md-4 social padding"><a
href="https://www.tumblr.com"><i class="fab fa-tumblr"></i></a></div>
            </div>
            <div>
                <a href="{{ $a->dokumentacija }}" class="btn btn-outline-
secondary">Dokumentacija</a>
            </div>
        </div>
        <?php
        endforeach;
        ?>
        </div>
    </div>
</div>
@endsection
```

### 3.11.2 resources/views/pages/

[Type here]

[Type here]