

Visoka škola strukovnih studija za
informacione i komunikacione
tehnologije

CarRent

<http://www.carrent2020.epizy.com/?i=1>

Aleksandar Madić 2/18

Smer: Internet tehnologije

Modul: Web programiranje

Predmet: Praktikum iz Web programiranja PHP

Sadržaj

1.	Uvod.....	5
1.1	Korišćeni programski jezici	5
1.2	Opis funkcionalnosti	5
1.3	Template	6
2.	Organizacija	15
2.1	Šema baze podataka.....	15
2.2	Mapa sajta	16
2.3	Struktura projekta.....	17
2.4	Opis stranica sajta	24
2.4.1	index strana	24
2.4.2	cars strana.....	24
2.4.3	carDetails strana.....	26
2.4.4	about strana	29
2.4.5	autor strana.....	30
2.4.6	contact strana	30
2.4.7	user strana	30
2.4.8	admin strana.....	32
3.	Kodovi.....	37
3.1	PHP	37
3.1.1	index.php.....	37
3.1.2	config/connection.php.....	40
3.1.3	config/config.php	43
3.1.4	models/account/login.php	44
3.1.5	models/account/register.php	46
3.1.6	models/booking/canceluserBook.php	48
3.1.7	models/booking/getAll.php.....	50
3.1.8	models/booking/handleBook.php	51
3.1.9	models/booking/handleBookCancel.php	53
3.1.10	models/brands/delete.php.....	54
3.1.11	models/brands/getAll.php	55
3.1.12	models/brands/getSingle.php	55
3.1.13	models/brands/insert.php	56

3.1.14 models/brands/update.php	58
3.1.15 models/cars/carDetails.php	59
3.1.16 models/cars/delete.php.....	62
3.1.17 models/cars/filterCars.php	63
3.1.18 models/cars/filterFromIndex.php.....	66
3.1.19 models/cars/functionsCars.php	67
3.1.20 models/cars/getAll.php	68
3.1.21 models/cars/getSingle.php	68
3.1.22 models/cars/handleBooking.php.....	70
3.1.23 models/cars/insert.php	72
3.1.24 models/cars/update.php.....	79
3.1.25 models/reviews/changeState.php	83
3.1.26 models/reviews/getAll.php	84
3.1.27 models/reviews/userDelete.php	85
3.1.28 models/reviews/userInsert.php.....	86
3.1.29 models/reivews/userUpdate.php	87
3.1.30 models/functions.php	88
3.1.31 models/initApi.php.....	93
3.1.32 models/createExcel	94
3.1.33 models/createWord.php	95
3.1.34 models/initLoad.php	96
3.1.35 models/statsLog.php	97
3.1.36 views/about.php	98
3.1.37 views/adminPanel.php	100
3.1.38 views/author.php	101
3.1.39 views/carDetails.php	102
3.1.40 views/cars.php.....	109
3.1.41 views/contact.php.....	111
3.1.42 views/index.php	113
3.1.43 views/login.php	117
3.1.44 views/logout.php	118
3.1.45 views/profile.php	118
3.1.45 views/register.php	119

3.2 JS.....	121
3.2.1 main.js.....	Error! Bookmark not defined.
3.2.2 admin.js.....	150

1. Uvod

1.1 Korišćeni programski jezici

Za izradu sajta korišćene su sledeće tehnologije:

- PHP, MySQL
- HTML 5, CSS 3, Bootstrap
- JavaScript, jQuery, AJAX
- PHPMailer
- Visual studio code

1.2 Opis funkcionalnosti

- Registracija i logovanje korisnika uz provere na serverskoj i klijentskoj strani
- Dinamičko ispisivanje menija na svim stranicama iz baze
- Dinamičko ispisivanje automobila iz baze
- Korisnik ima mogućnost da posalje zahtev za iznamljivanje auta, admin treba da odobri u zavisnosti da li je taj auto već iznajmljen za dati datum
- Korisnik ima mogućnost da napiše svoju recenziju samo za automobile koje je iznajmio
- Mogućnost filtriranja automobila putem ajaxa, sa paginacijom
- Statistika pristupa stranicama u txt fajlu
- Logovanje gresaka u log fajl
- Kontakt stranica kojom je omogućeno kontaktiranje admina preko PHPMailera
- Sve forme su proverene i na klijentu i na serveru
- Admin ima da izvrsi CRUD operacije nad automobilima, brendovima, da pogleda statistiku pristupa stranicama, da upravlja iznamljivanjem auta, da izabere koje recenzije će se prikazivati na stranici
- Export podataka o autoru u word fajl na stranici o autoru
- Export podataka iz baze o automobilima u excel fajl na admin stranici
- Na stranicama admin i user, klikom na ajaxom se dohvata sadrzaj u zavisnosti sta se odabere od linkova; tako da se ne refresh-uje stranica
- Obrada grešaka, upisavanje u log fajl errors

1.3 Template

Index strana

The screenshot shows a car listing for a Range Rover. The car is a dark-colored SUV parked in a wooded area. To the left of the car, there is a detailed specification table:

Range Rover	
Doors	5
Seats	5
Lugage	3
Transmission	Automatic
Fuel	Petrol

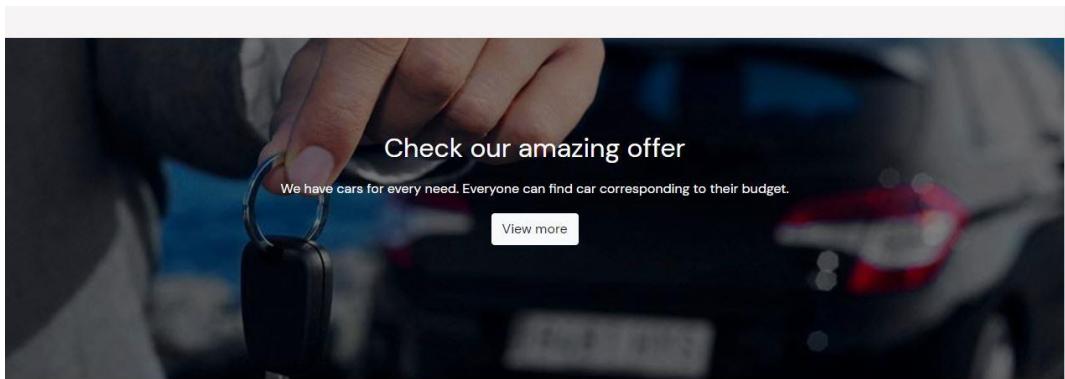
Below the table, the price is listed as \$78/day, with a "View details" button.

The screenshot shows a search interface with the heading "Begin your trip here". It includes search fields for "Search", "Select category", "Select Fuel", and a "Submit" button. To the right, it says "12 cars in our offer".

Below the search interface, there is a section titled "Our Offer" with the sub-instruction "Create account, look for perfect car, submit request and wait for admin's approval".

At the bottom, there are two car listings:

Audi A4	Land Rover Range Rover
\$48/day	\$78/day
Transmission	Manual
Fuel	Diesel
Doors	5
Seats	4



How it works

With these simple steps get you to your dream ride



How it works

With these simple steps get you to your dream ride



Cars strana

CarRent

SUPPORT MAIL:
alemadic@gmail.com

OUR ADDRESS:
777 Brockton Avenue, MA 2351

Blake Smith ▾

Home Cars About Contact

Our For Rent Cars

Here you can see our rich offer of luxury and afordable cars (based on your budget)



Filter

Search

Select category

Select Fuel

Transmission type

Automatic

Car listing



Mercedes-Benz CLS 350



Kia Sportage CRDI

Filter

Search

Select category

Select Fuel

Transmission type

Automatic

Manual

Price range:
\$0 - \$200

Submit

Car listing



Mercedes-Benz CLS 350
\$70/day

Doors	5
Seats	4
Transmission	Manual
Fuel	Diesel

View Details



Kia Sportage CRDI
\$50/day

Doors	5
Seats	4
Transmission	Automatic
Fuel	Diesel

View Details





Chevrolet Cruze

\$45/day

Doors	5
-------	---

Seats	4
-------	---

Transmission	Manual
--------------	--------

Fuel	Petrol
------	--------

[View Details](#)



Volkswagen Sharan

\$72/day

Doors	5
-------	---

Seats	6
-------	---

Transmission	Manual
--------------	--------

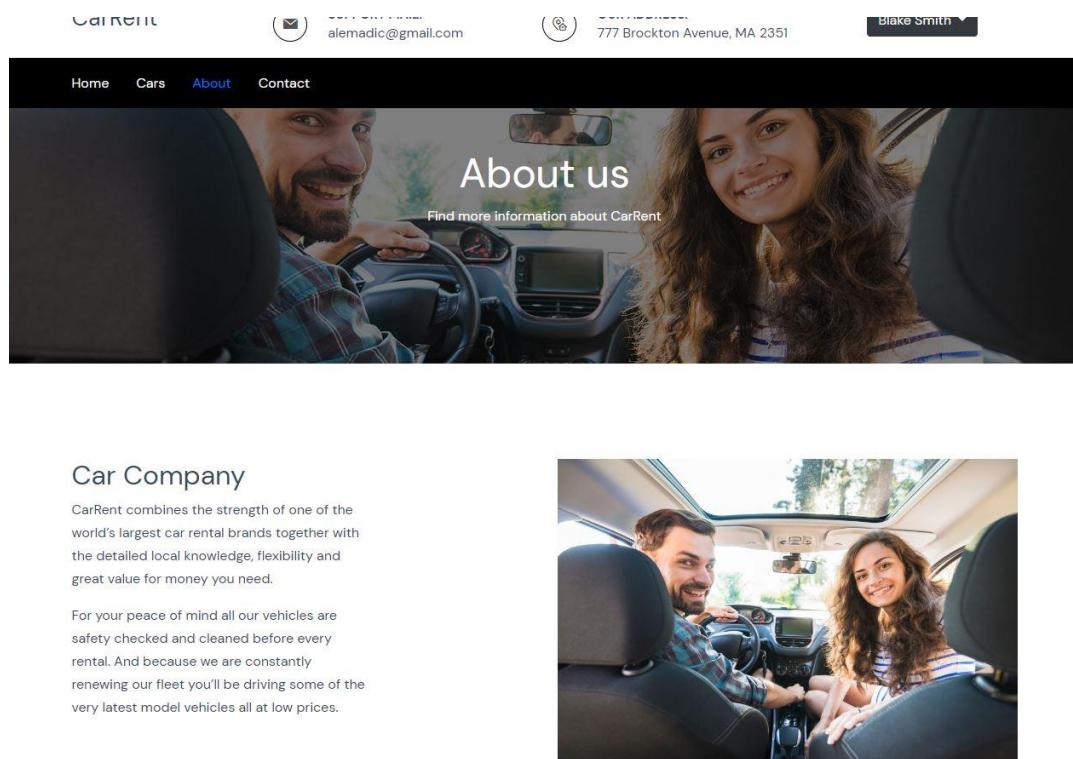
Fuel	Petrol
------	--------

[View Details](#)

1 2

[Facebook](#) / [Instagram](#) / [Youtube](#) / [Docs](#) / [Author](#) /

About strana



The screenshot shows the 'About us' page of the CarRent website. At the top, there's a navigation bar with links for Home, Cars, About (which is highlighted in blue), and Contact. Below the navigation is a large banner image of a man and a woman smiling from inside a car. Overlaid on the banner is the text 'About us' and 'Find more information about CarRent'. The main content area contains two columns. The left column is titled 'Car Company' and includes a paragraph about CarRent's strength as a combination of the world's largest car rental brands and local knowledge. It also mentions safety checks and renewals. The right column is titled 'Our History' and includes a paragraph about CarRent's establishment in 1990 and its global presence. There's also a small image of a Range Rover driving through a forest.

Car Company

CarRent combines the strength of one of the world's largest car rental brands together with the detailed local knowledge, flexibility and great value for money you need.

For your peace of mind all our vehicles are safety checked and cleaned before every rental. And because we are constantly renewing our fleet you'll be driving some of the very latest model vehicles all at low prices.



Our History

Established in 1990, the CarRent brand is licenced within Europe, the Middle East and Africa to Zodiac Europe Ltd (a subsidiary of Avis Budget EMEA Ltd). Budget locations are either directly owned or franchised to local operators across the region.

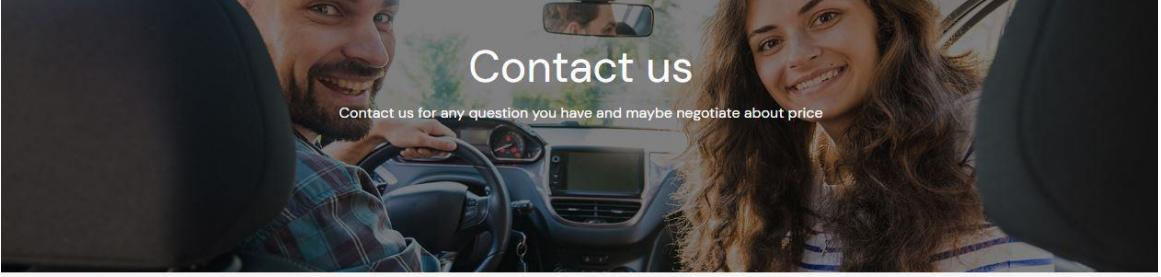
As our name suggests we've always been synonymous with great value for money. We now offer a range of fully inclusive rates, pre-payment options and charge facilities designed to keep you in control and help you save money.

How it works

With these simple steps get you to your dream ride



Contact strana



Home Cars About Contact

Contact us

Contact us for any question you have and maybe negotiate about price

Contact Us Or Use This Form To Rent A Car

First Name

Last Name

Email

Message

Contact Info

Address:
777 Brockton Avenue, MA 2351

Phone:
+1 242 4942 290

Email:
admin@carrent.com

Contact Us

First Name

Last Name

Email

Message

Send Message

Contact Info

Address:
777 Brockton Avenue, MA 2351

Phone:
+1 242 4942 290

Email:
admin@carrent.com

Admin strana

CarRent

SUPPORT MAIL:
admin@current.com

OUR ADDRESS:
777 Brockton Avenue, MA 2351

Admin Pink ▾

Home Cars About Contact

Hello admin
Admin panel of current website

Cars Brands Stats Bookings Reviews Export excel

Facebook / Instagram / Youtube / Docs / Author /

User profile strana

CarRent

SUPPORT MAIL:
admin@current.com

OUR ADDRESS:
777 Brockton Avenue, MA 2351

Ale Madic ▾

Home Cars About Contact

Hello user
User profile

My Bookings My Reviews

Facebook / Instagram / Youtube / Docs / Author /

Login strana

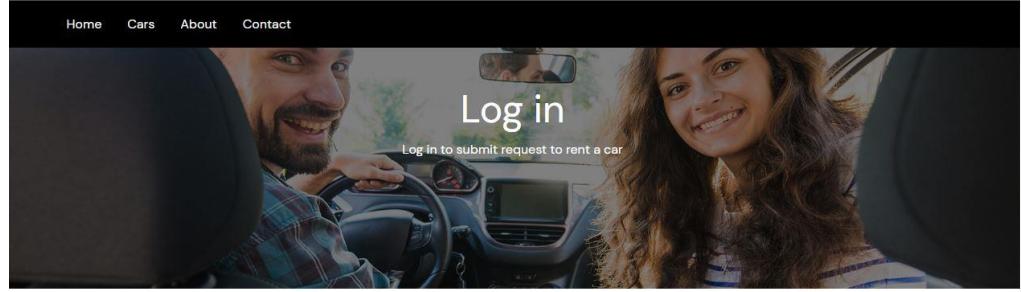
CarRent

SUPPORT MAIL:
admin@carrent.com

OUR ADDRESS:
777 Brockton Avenue, MA 2351

Login / Register

Home Cars About Contact



Log in
Log in to submit request to rent a car

Email

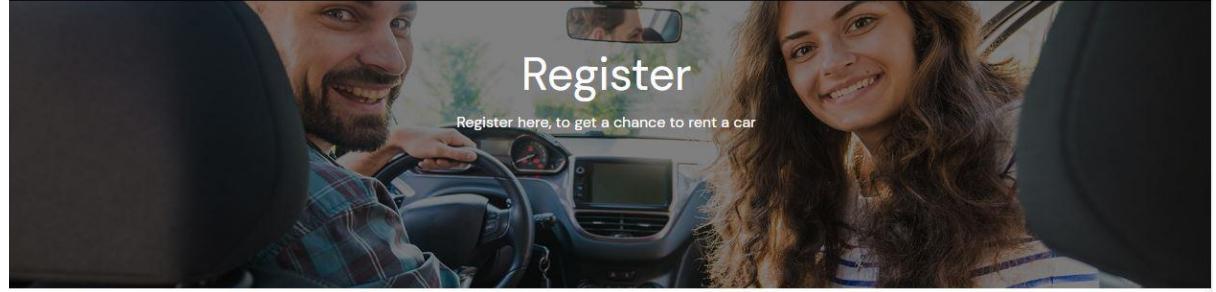
Password

Submit

Don't have account? [Sign Up here](#)

Register strana

Home Cars About Contact



Register
Register here, to get a chance to rent a car

First name Last name

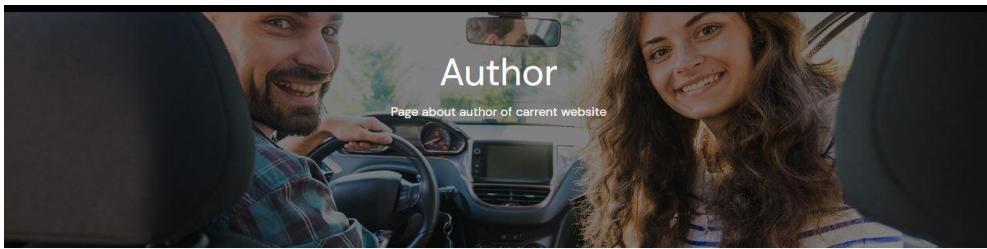
Email

Password

Driver licence No Years of Experience

Submit

Autor strana



Aleksandar Madić 2/18

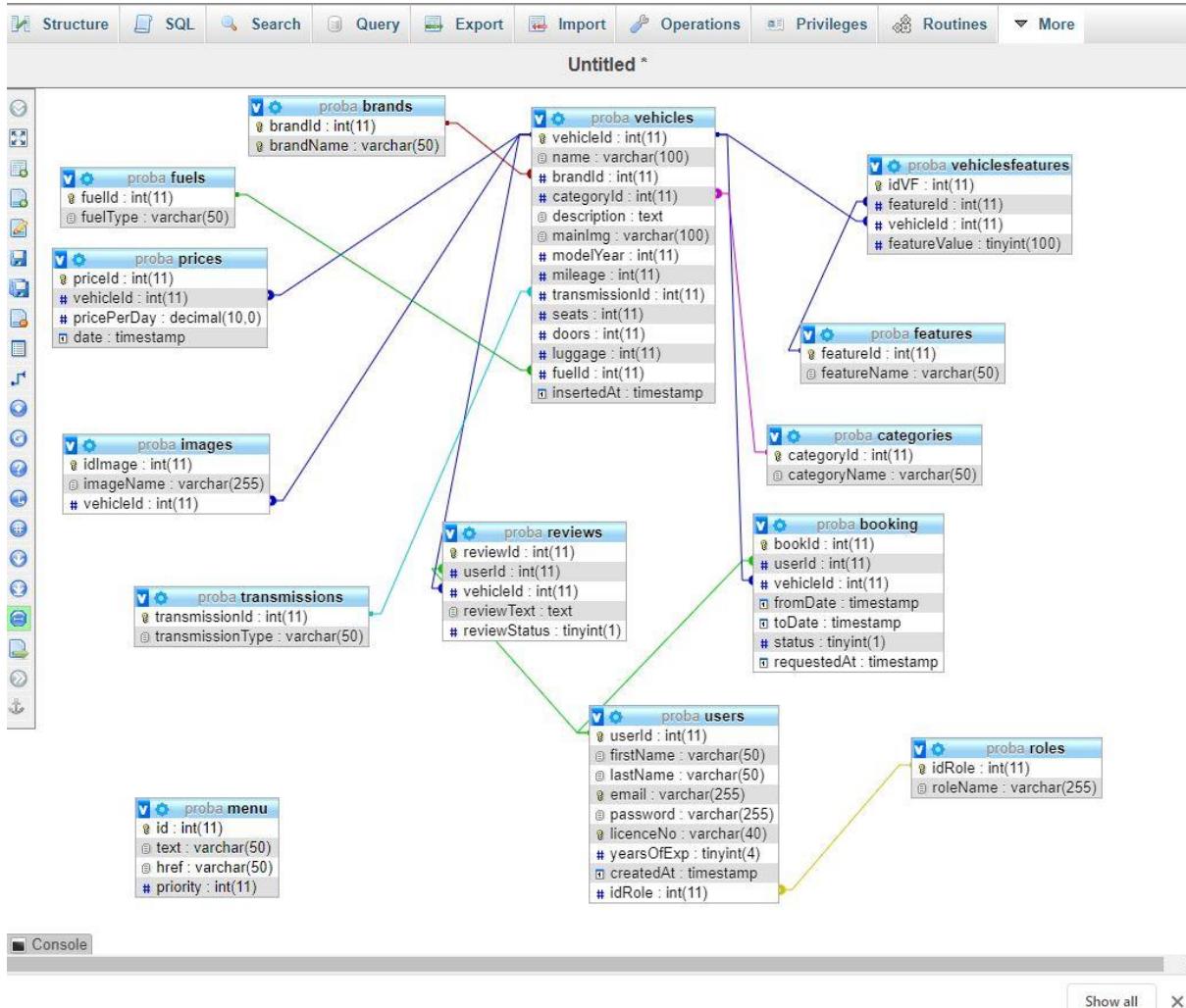


I am currently a student at the ICT College. I was born in Bor, where I finished elementary and high school. In high school, I first encountered web design and a little bit of programming and I liked it. It made me choose this college. In my spare time, I like to play basketball, go to the theater, travel and explore new things.

[Download word](#)

2. Organizacija

2.1 Šema baze podataka



2.2 Mapa sajta

```
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
<url>
    <loc>https://carrent2020.epizy.com/index.php</loc>
    <lastmod>2020-06-12</lastmod>
    <changefreq>weekly</changefreq>
    <priority>0.95</priority>
</url>
<url>
    <loc>https://carrent2020.epizy.com/index.php?page=cars</loc>
    <lastmod>2020-06-14</lastmod>
    <changefreq>weekly</changefreq>
    <priority>0.9</priority>
</url>
<url>
    <loc>https://carrent2020.epizy.com/index.php?page=about</loc>
    <lastmod>2020-06-1</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.7</priority>
</url>
<url>
    <loc>https://carrent2020.epizy.com/index.php?page=profile</loc>
    <lastmod>2020-06-1</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.75</priority>
</url>
<url>
    <loc>https://carrent2020.epizy.com/index.php?page=login</loc>
    <lastmod>2020-06-10</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.65</priority>
</url>
<url>
    <loc>https://carrent2020.epizy.com/index.php?page=register</loc>
    <lastmod>2020-06-2</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.65</priority>
</url>
<url>
    <loc>https://carrent2020.epizy.com/index.php?page=admin</loc>
    <lastmod>2020-06-1</lastmod>
    <changefreq>weekly</changefreq>
    <priority>0.6</priority>
</url>
<url>
    <loc>https://carrent2020.epizy.com/index.php?page=contact</loc>
```

```

<lastmod>2020-02-5</lastmod>
<changefreq>yearly</changefreq>
<priority>0.55</priority>
</url>

<url>
    <loc>https://carrent2020.epizy.com/index.php?page=logout</loc>
    <lastmod>2020-02-5</lastmod>
    <changefreq>yearly</changefreq>
    <priority>0.55</priority>
</url>
</urlset>

```

2.3 Struktura projekta

root

Name	Date modified	Type	Size
assets	21/05/2020 16:56	File folder	
config	17/05/2020 17:00	File folder	
data	11/06/2020 13:08	File folder	
models	15/06/2020 00:32	File folder	
phpMailer	11/06/2020 15:25	File folder	
views	14/06/2020 23:26	File folder	
index.php	15/06/2020 09:42	PHP File	3 KB
sitemap.xml	15/06/2020 14:01	XML File	2 KB

config

Name	Date modified	Type	Size
.env	17/05/2020 17:01	ENV File	1 KB
config.php	13/06/2020 19:49	PHP File	1 KB
connection.php	14/06/2020 23:30	PHP File	5 KB

assets

Name	Date modified	Type	Size
css	09/05/2020 10:27	File folder	
fonts	09/05/2020 10:46	File folder	
images	14/06/2020 23:29	File folder	
img	15/06/2020 10:35	File folder	
js	04/06/2020 07:33	File folder	
scss	10/04/2019 10:53	File folder	

data

Name	Date modified	Type	Size
errors.txt	15/06/2020 12:28	Text Document	1 KB
log.txt	15/06/2020 13:42	Text Document	348 KB

models

Name	Date modified	Type	Size
account	08/06/2020 20:56	File folder	
booking	15/06/2020 12:17	File folder	
brands	06/06/2020 14:28	File folder	
cars	13/06/2020 19:47	File folder	
reviews	14/06/2020 23:36	File folder	
createExcel.php	15/06/2020 06:49	PHP File	2 KB
createWord.php	15/06/2020 00:30	PHP File	1 KB
functions.php	13/06/2020 19:08	PHP File	7 KB
initApi.php	08/06/2020 17:17	PHP File	2 KB
initLoad.php	09/06/2020 18:17	PHP File	1 KB
statsLog.php	15/06/2020 00:28	PHP File	2 KB

Models/account

Name	Date modified	Type	Size
login.php	15/06/2020 11:02	PHP File	2 KB
register.php	15/06/2020 11:02	PHP File	3 KB

Models/booking

Name	Date modified	Type	Size
cancelUserBook.php	15/06/2020 11:03	PHP File	1 KB
getAll.php	15/06/2020 11:49	PHP File	2 KB
handleBook.php	15/06/2020 12:14	PHP File	3 KB
handleBookCancel.php	15/06/2020 12:22	PHP File	1 KB

Models/ brand

Name	Date modified	Type	Size
delete.php	15/06/2020 11:03	PHP File	1 KB
getAll.php	06/06/2020 14:22	PHP File	1 KB
getSingle.php	05/06/2020 12:24	PHP File	1 KB
insert.php	15/06/2020 11:04	PHP File	2 KB
update.php	15/06/2020 11:04	PHP File	1 KB

Models/car

Name	Date modified	Type	Size
carDetails.php	15/06/2020 12:38	PHP File	4 KB
dataForm.php	15/06/2020 09:28	PHP File	3 KB
delete.php	15/06/2020 11:04	PHP File	1 KB
filterCars.php	12/06/2020 18:56	PHP File	5 KB
filterFromIndex.php	12/06/2020 11:33	PHP File	2 KB
functionsCar.php	13/06/2020 20:41	PHP File	2 KB
getAll.php	12/06/2020 14:05	PHP File	1 KB
getSingle.php	15/06/2020 10:48	PHP File	2 KB
handleBooking.php	15/06/2020 11:43	PHP File	2 KB
insert.php	15/06/2020 11:04	PHP File	9 KB
insertSaFolderima.php	12/06/2020 21:36	PHP File	9 KB
update.php	15/06/2020 11:05	PHP File	5 KB

Models/reviews

Name	Date modified	Type	Size
changeState.php	14/06/2020 23:41	PHP File	1 KB
getAll.php	15/06/2020 07:06	PHP File	2 KB
userDelete.php	15/06/2020 11:06	PHP File	1 KB
userInsert.php	15/06/2020 11:06	PHP File	2 KB
userUpdate.php	15/06/2020 11:06	PHP File	1 KB

Views/

Name	Date modified	Type	Size
admin	14/06/2020 23:47	File folder	
fixed	07/06/2020 22:51	File folder	
profile	09/06/2020 16:42	File folder	
user	14/06/2020 12:46	File folder	
about.php	15/06/2020 13:28	PHP File	7 KB
adminPanel.php	15/06/2020 09:08	PHP File	2 KB
author.php	15/06/2020 00:30	PHP File	1 KB
carDetails.php	15/06/2020 12:39	PHP File	11 KB
cars.php	14/06/2020 08:28	PHP File	3 KB
carsSSR.php	17/05/2020 17:42	PHP File	10 KB
contact.php	15/06/2020 13:32	PHP File	4 KB
index.php	15/06/2020 12:56	PHP File	7 KB
indexRadi.php	13/06/2020 20:54	PHP File	17 KB
login.php	08/06/2020 20:51	PHP File	3 KB
logout.php	08/06/2020 17:48	PHP File	1 KB
profile.php	14/06/2020 21:13	PHP File	1 KB
register.php	08/06/2020 21:28	PHP File	6 KB

Views/admin

Name	Date modified	Type	Size
booking	11/06/2020 18:07	File folder	
brands	05/06/2020 08:07	File folder	
cars	10/06/2020 22:36	File folder	
categories	04/06/2020 07:30	File folder	
reviews	14/06/2020 22:55	File folder	
statsLog.php	15/06/2020 00:28	PHP File	2 KB

Views/admin/Book

Name	Date modified	Type	Size
selectAll.php	11/06/2020 18:44	PHP File	5 KB

Views/admin/brand

Name	Date modified	Type	Size
insertForm.php	06/06/2020 19:34	PHP File	3 KB
selectAll.php	05/06/2020 12:30	PHP File	3 KB

Views/admin/car

Name	Date modified	Type	Size
insertForm.php	10/06/2020 10:56	PHP File	9 KB
selectAll.php	15/06/2020 07:49	PHP File	4 KB
updateForm.php	13/06/2020 19:11	PHP File	11 KB

Views/admin/reviews

selectAll.php	14/06/2020 23:42	PHP File	4 KB
---------------	------------------	----------	------

Views/fixed/

Name	Date modified	Type	Size
footer.php	14/06/2020 08:43	PHP File	3 KB
head.php	11/06/2020 13:30	PHP File	2 KB
header.php	15/06/2020 13:31	PHP File	9 KB
headerTest.php	06/06/2020 12:31	PHP File	8 KB
hero - Copy.php	03/06/2020 17:43	PHP File	3 KB
nav.php	12/06/2020 11:25	PHP File	1 KB
navStari.php	17/05/2020 17:33	PHP File	3 KB

Views/user/

Name	Date modified	Type
booking	11/06/2020 19:33	File folder
reviews	14/06/2020 21:09	File folder

Views/user/booking/

Name	Date modified	Type	Size
selectAll.php	14/06/2020 10:41	PHP File	5 KB

Views/user/review

Name	Date modified	Type	Size
insertForm.php	15/06/2020 12:28	PHP File	4 KB
selectAll.php	15/06/2020 12:26	PHP File	3 KB

2.4 Opis stranica sajta

2.4.1 index strana

Dinamicko ispisivanje menija iz baze. popunjavanje filtera sa pretragu “small search” podacima iz baze. Klikom na submit popupice se podaci iz te male forme proslediti stranici cars I tamo ce se prikazati rezultat filtriranja. Ispisuji se automobili iz baze I prikazuju u vidu slidera. Klik na dugme “view more” vodi na stranicu carDetails gde se prikazuju detalji o konkretnom autu.

2.4.2 cars strana

Na ovoj strani se ispisuje automobile, I koristi se paginacija. Filterima sa strane se mogu izfiltrirati automobile klikom na taster submit. Klikom na dugme view more vodi na stranicu carDetails gde se nalaze dodatne karakteristike automobile.

Filter

Search

Coupe

Diesel

Transmission type

Automatic

Manual

Price range:

\$0 - \$200

Submit

Car listing


BMW 520
\$60/day
Doors 5
Seats 4
Transmission Automatic
Fuel Diesel
[View Details](#)


Mercedes-Benz C 250
\$72/day
Doors 5
Seats 4
Transmission Automatic
Fuel Diesel
[View Details](#)



Filter

Search

Select category

Select Fuel

Transmission type

Automatic

Manual

Price range:
\$48 - \$98

Submit

Car listing



Kia Sportage CRDI
\$50/day

Doors: 5
Seats: 4
Transmission: Automatic
Fuel: Diesel

View Details



BMW 520
\$60/day

Doors: 5
Seats: 4
Transmission: Automatic
Fuel: Diesel

View Details





Filter

Audi dsa

Select category

Select Fuel

Transmission type

Automatic

Manual

Price range:
\$48 - \$98

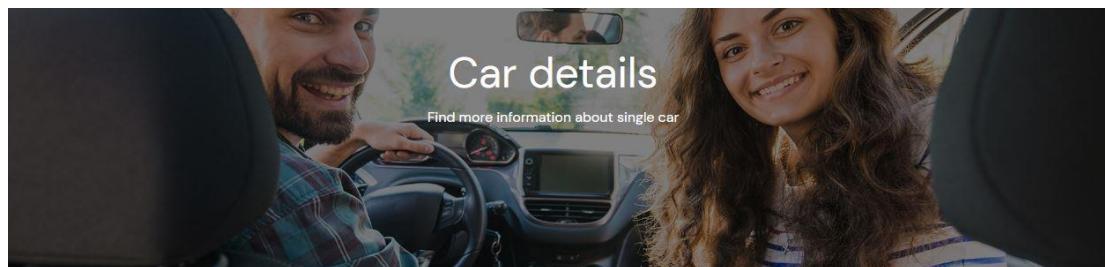
Submit

Car listing

We don't have any car for selected criteria

2.4.3 carDetails strana

Na ovoj strani se ispisuju dodatne karakteristike automobila. Prikazuje se galerija sa slikama automobila koje je admin uneo prilikom inserta u admin panelu. Prikazuju se opste informacije kao i dodatne informacije o autu sta sadrzi. Takodje se ispisuju recenzije korisnika koji su do sad iznajmili taj auto ukoliko ih ima. Ispisuje se forma u kojoj korisnik treba da izabere datum od i datum do kad zeli da iznajmi auto. Vrsi se provera da ne može da se izabere datum u prošlosti i da datum od mora biti manji od datum do. Kada korisnik ispravno popuni formu upisuje se u bazu podataka njegov zahtev. Na drnu strane se prikazuju automobile koji su po tipu prenosa i goriva slični odabranom autu.



COUPE

Mercedes-Benz CLS 350



Mercedes-Benz CLS 350



Mileage
70000



Transmission
Manual



Seats
4



Luggage
2



Fuel
Diesel



Mileage
70000



Transmission
Manual



Seats
4



Luggage
2



Fuel
Diesel

Description

Features

Reviews

The CLS is yet another vehicle from Mercedes-Benz, which gave birth to a new body style that went on to find favour with almost every direct competitor. Back in 2003, the CLS four-door coupe came through as an eye-catching car and found favour among many for its new design. With the CLS, the German automotive giant tried combining the best of two worlds – coupe and four-door saloon. Did they succeed? The answer is a clear affirmative and since October 2004, about 170,000 units of the CLS have moved out of showrooms globally.

Submit a request

From date

To date

Submit

SEARCH CAR

Mileage
70000

Transmission
Manual

Seats
4

Luggage
2

Fuel
Diesel

Description Features Reviews

✓ Airconditioning
✓ AntiLock Braking System
✗ Leather Seats
✓ Brake Assist

✗ Crash Sensor
✓ Onboard computer
✓ GPS
✓ Central Locking

✗ ABS
✗ Bluetooth
✗ Child Seat
✗ Parking Sensors

Submit a request

From date dd/mm/yyyy

To date dd/mm/yyyy

Submit

CHOOSE CAR

Related Cars

Mileage
70000

Transmission
Manual

Seats
4

Luggage
2

Fuel
Diesel

Description Features Reviews

user Ale Madic "This car is amazing. I rented for my short visit of Miami beach and I loved it every second."

Submit a request

From date dd/mm/yyyy

To date dd/mm/yyyy

Submit

CHOOSE CAR

Related Cars

A screenshot of a web browser displaying a car rental application. At the top, there's a navigation bar with a logo and a search bar containing the text "Search for a car". Below the search bar, there are several filters: "Mileage 62300", "Transmission Automatic", "Seats 4", "Luggage 2", and "Fuel Diesel". A large image of a silver car is on the left, and a smaller image of a car interior is on the right. A central modal window has a blue header bar with the text "localhost says" and the message "You must choose date in future". A blue "OK" button is at the bottom of the modal. Below the modal, there are three tabs: "Description", "Features", and "Reviews". Under the "Reviews" tab, it says "No reviews for this car." At the bottom, there's a section titled "Submit a request" with two date input fields: "From date 09/06/2020" and "To date 01/07/2020", both with calendar icons. A blue "Submit" button is below these fields.

A screenshot of a web browser displaying a section titled "Related Cars". The title "CHOOSE CAR" is above the heading "Related Cars". Below the heading are three cards, each showing a car image and details:

- Kia Sportage CRDI**
\$100 day
Doors: 5
Seats: 4
Transmission: Automatic
Fuel: Diesel
[View Details](#)
- Volkswagen Golf 6**
\$100 day
Doors: 3
Seats: 4
Transmission: Automatic
Fuel: Diesel
[View Details](#)
- Ford Fiesta**
\$100 day
Doors: 5
Seats: 4
Transmission: Automatic
Fuel: Diesel
[View Details](#)

2.4.4 about strana

Na ovoj strani se nalazi staticki tekst o kompaniji

2.4.5 autor strana

Na ovoj strani se nalaze podaci o autoru. Takodje klikom na dugme moze se exportovati sadrzaj u word document. Ovo radi u localu, hosting ne prihvata com funkcije

2.4.6 contact strana

Na ovoj strani omogućeno je kontaktiranje admina popunjavanjem forme koja je validirana na klijentu i na serveru. Slanje mejla se vrši pomoću PHPMailer-a

2.4.7 user strana

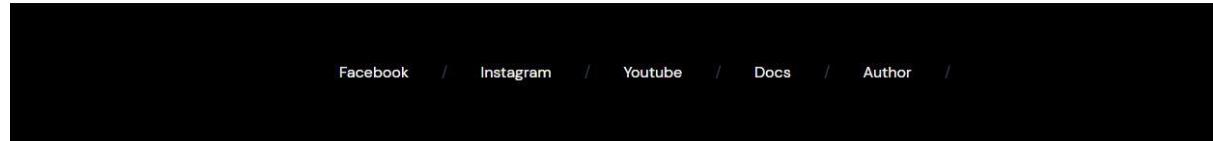
Ova strana namenjena je ulogovanim korisnicima koji mogu da prate status zahteva za iznajmljivanje auta. Omogućeno je da se otkaže zahtev. Omogućeno je da korisnik dodaje recenzije, ali samo za automobile koje je iznajmio. Moguća je izmena i brisanje recenzije.

Sve je realizovano ajaxom.



Manage Booking

Booking details							
#	User name	Car	From Date	To Date	Price	Status	
1	Ale Madic	Mercedes-Benz CLS 350	2020-06-23	2020-07-02	\$70	Confirmed	Cancel
2	Ale Madic	Kia Sportage CRDI	2020-06-23	2020-06-30	\$50	Confirmed	Cancel





My Bookings My Reviews

Manage Reviews

Reviews details

Car name

Review

1

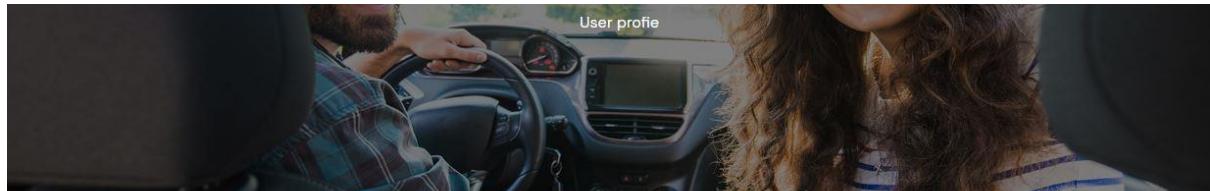
Mercedes-Benz CLS 350

This car is amazing. I rented for my short visit of Miami beach and I loved it every second.



Insert new

Facebook / Instagram / YouTube / Docs / Author /



My Bookings

My Reviews

Post a review

Reviews info

Select vehicle

Select

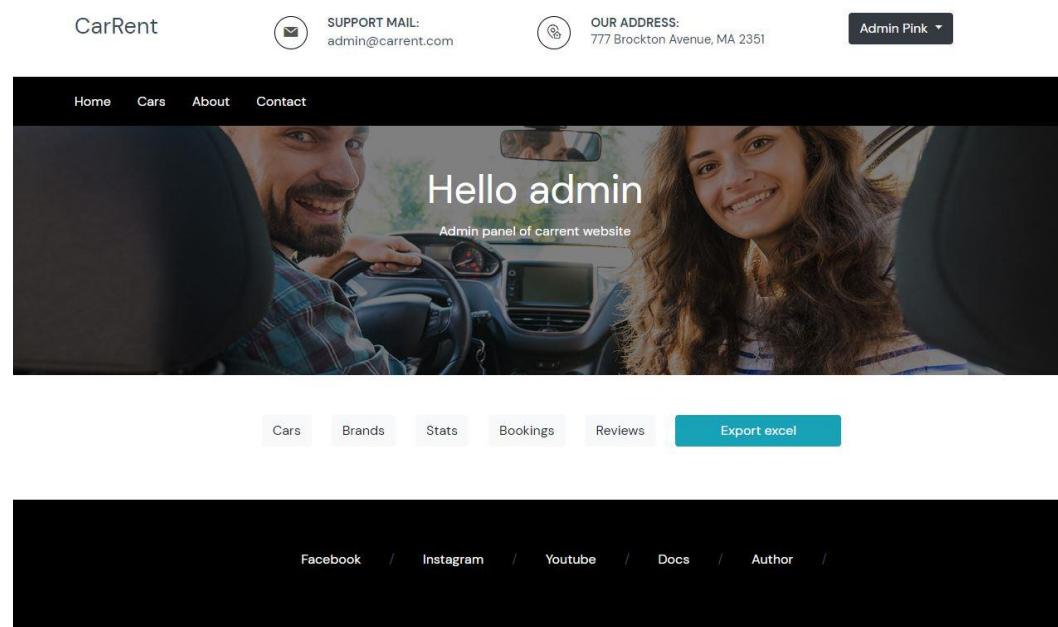
Review text

Submit

2.4.8 admin strana

Na ovoj strani omogućeno je upravljanje sadržajem sajta. CRUD operacije za automobile, prilikom inserta obrađuju se slike na dve veličine kako bi se kasnije prikazale u galeriji na stranici vezanoj za taj auto. CRUD operacije za brendove. Prikaz statistike pristupa stranicama. Upravljanje zahtevima za iznajmljivanje. Vrši se provera da jedan auto u jednom opsegu datuma može biti samo jednom iznajmljen. Tako da ako na primer admin pokuša da prihvati neki zahtev, zahtev će biti prihvaćen jedino ako taj auto nije već izdat u tom vremenskom intervalu. Upravljanje recenzijama, koje recenzije će biti vidljive na sajtu koje ne. Sve forme se validiraju na klijentu i na serveru. Omogućen je export podataka iz baze o automobilima u excel fajl.

Sve se realizuje ajaxom.



[Cars](#)[Brands](#)[Stats](#)[Bookings](#)[Reviews](#)[Export excel](#)

Manage Vehicles

Vehicle details

Car model	Brand	Price Per day	Fuel type	Model year	Action
CLS 350	Mercedes-Benz	\$70	Diesel	2014	
Sportage CRDI	Kia	\$50	Diesel	2018	
Golf 6	Volkswagen	\$30	Diesel	2012	
520	BMW	\$60	Diesel	2014	
Cruze	Chevrolet	\$45	Petrol	2015	
Sharan	Volkswagen	\$72	Petrol	2016	
Focus	Ford	\$54	Diesel	2016	
Fiesta	Ford	\$58	Diesel	2017	
C 250	Mercedes-Benz	\$72	Diesel	2018	

[Cars](#)[Brands](#)[Stats](#)[Bookings](#)[Reviews](#)[Export excel](#)

Post a vehicle

Basic info

Car name

Valid format: Sportage

Select brand

Car desc

Description must have at least 10 chars

Select category

Select fuel

Update a vehicle

Basic info

Car name

Golf 6

Select brand

Volkswagen

Car desc

The Volkswagen Golf 6 (or VW Typ 5K or MK VI) is a compact car, the sixth generation of the Volkswagen Golf and the successor to the Volkswagen Golf Mk5. It was unveiled at the Paris Motor Show in October 2008. Volkswagen released pictures and information on 6 August 2008, prior to

Select category

Hatchback

Select fuel

Diesel

Model year

2012

Transmission Type

- Automatic
 Manual

Price per day

30

Images

Cover photo

 No file chosen

Other photo

 No file chosen

Other photo

 No file chosen

Accessories

 Airconditioning AntiLock Breaking System Leather Seats Brake Assist Crash Sensor Onboard computer GPS Central Locking ABS Bluetooth Child Seat Parking Sensors

Stats for last 24 hours

Total number: 36

Visits	
Page	Number
/current/views/admin/booking/selectAll.php:	2.78 %: 1 times
/current/views/admin/statsLog.php:	22.22 %: 8 times
/current/index.php:	30.56 %: 11 times
/current/models/initApi.php:	38.89 %: 14 times
/current/models/cars/filterCars.php:	5.56 %: 2 times

[Facebook](#) / [Instagram](#) / [Youtube](#) / [Docs](#) / [Author](#) /

[Cars](#) [Brands](#) [Stats](#) [Bookings](#) [Reviews](#) [Export excel](#)

Manage Booking

Booking details

#	User name	Car	From Date	To Date	Price	Status	Action
1	Ale Madic	Mercedes-Benz CLS 350	2020-06-23	2020-07-02	\$70	Confirmed	Confirm / Cancel
2	Ale Madic	Kia Sportage CRDI	2020-06-23	2020-06-30	\$50	Confirmed	Confirm / Cancel
3	Blake Smith	Mercedes-Benz CLS 350	2020-06-18	2020-06-30	\$70	Canceled	Confirm / Cancel
4	Blake Smith	Mercedes-Benz CLS 350	2020-06-25	2020-06-30	\$70	Not proccesed	Confirm / Cancel
5	Blake Smith	Kia Sportage CRDI	2020-06-30	2020-07-10	\$50	Confirmed	Confirm / Cancel

[Facebook](#) / [Instagram](#) / [Youtube](#) / [Docs](#) / [Author](#) /

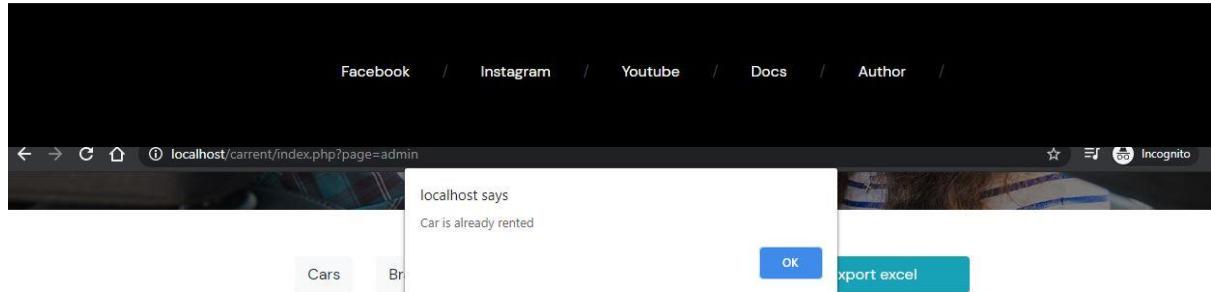


Cars Brands Stats Bookings **Reviews** Export excel

Manage Reviews

Review details

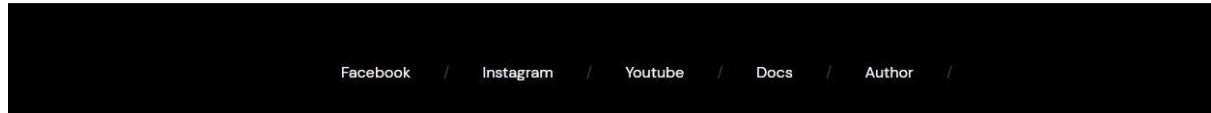
User	#	name	Car	Review	Status	Action
1	Ale Madic		Mercedes-Benz CLS 350	This car is amazing. I rented for my short visit of Miami beach and I loved it every second.	Visible	Show / Hide
2	Blake Smith		Kia Sportage CRDI	This car is amazing for family. It's comfortable and spacious	Visible	Show / Hide



Manage Booking

Booking details

#	User name	Car	From Date	To Date	Price	Status	Action
1	Ale Madic	Mercedes-Benz CLS 350	2020-06-23	2020-07-02	\$70	Confirmed	Confirm / Cancel
2	Ale Madic	Kia Sportage CRDI	2020-06-23	2020-06-30	\$50	Confirmed	Confirm / Cancel
3	Blake Smith	Mercedes-Benz CLS 350	2020-06-18	2020-06-30	\$70	Canceled	Confirm / Cancel
4	Blake Smith	Mercedes-Benz CLS 350	2020-06-25	2020-06-30	\$70	Not proccesed	Confirm / Cancel
5	Blake Smith	Kia Sportage CRDI	2020-06-30	2020-07-10	\$50	Confirmed	Confirm / Cancel



3. Kodovi

3.1 PHP

3.1.1 index.php

```
<?php

session_start();

ob_start();

require "config/connection.php";

if(!isset($_GET['page'])) {
    $page = "index";
} else {
    switch ($_GET['page']) {
        case 'cars':
            $page = "cars";
            break;

        case 'carDetails':
            $page = "carDetails";
            break;

        case 'services':
            $page = "services";
            break;

        case 'about':
            $page = "about";
            break;

        case 'contact':
            $page = "contact";
            break;

        case 'blog':
            $page = "blog";
            break;

        case 'login':
            $page = "login";
            break;

        case 'register':
            $page = "register";
            break;
    }
}
```

```
        case 'logout':
            $page = "logout";
            break;

        case 'profile':
            if(isset($_SESSION['user']) && $_SESSION['user']['idRole'] ==
2) {
                $page = "profile";
            } else if (isset($_SESSION['user']) && $_SESSION['user']['idRo
le'] == 1) {
                $page = "adminPanel";
            } else {
                header("Location: index.php?page=login");
            }

            break;

        case "author":
            $page = "author";

            break;

        case 'admin':

            if(isset($_SESSION['user']) && $_SESSION['user']['idRole'] ==
1) {
                $page = "adminPanel";
            } else {
                header("Location: index.php?page=login");
            }

            break;

        default:
            $page = "index";
            break;
    }

}

$metaPodaci = handleMetaData($page);

require "views/fixed/head.php";
// require "views/fixed/nav.php";
require "views/fixed/header.php";
// var_dump($_SESSION['user']);
```

```
require "views/{$page}.php";  
  
require "views/fixed/footer.php";  
  
?>
```

3.1.2 config/connection.php

```
<?php

require "config.php";

pristupStranici();

try {
    $db = new PDO("mysql:host=" . SERVER . ";dbname=" . DBNAME . ";charset=utf8", USERNAME, PASSWORD);
    $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch(PDOException $ex) {
    // echo $ex->getMessage(); -> u Log fajl
    exit("We have issues with database, please try later");
}

function pristupStranici() {
    $fajl = fopen(LOG_FAJL, "a");
    if($fajl) {
        $datum = date("d-m-Y H:i:s");
        fwrite($fajl, $_SERVER['PHP_SELF'] . "\t" . $datum . "\t" . $_SERVER['REMOTE_ADDR'] . "\t\n");
        fclose($fajl);
    }
}

function handleGresku($err) {
    $fajl = fopen(GRESKE, "a");
    if($fajl) {
        $datum = date("d-m-Y H:i:s");

        fwrite($fajl, $_SERVER['PHP_SELF'] . "\t" . $_SERVER['REQUEST_URI'] . "\t" . $datum . "\t" . $err . "\t\n");
        fclose($fajl);
    }
}

function handleMetaData($page) {
    $meta = [
        "index" => [
            "title" => "Car Rent | Home",
            "titleForTag" => "",
            "desc" => "We are agency for rent a car. Here you can find best deals",
            "keys" => "current, rent a car, best car to rent"
        ],
        "cars" => [
    ];
```

```

        "title" => "Car Rent | Cars",
        "titleForTag" => "Our For Rent Cars",
        "desc" => "Here you can see our rich offer of luxury and affordable cars (based on your budget)",
        "keys" => "cars for rent, luxury car, best deal for car rent"
    ],
    "carDetails" => [
        "title" => "Car Rent | Car details",
        "titleForTag" => "Car details",
        "desc" => "Find more information about single car",
        "keys" => "best car for rent, info about car"
    ],
    "about" => [
        "title" => "Car Rent | About",
        "titleForTag" => "About us",
        "desc" => "Find more information about CarRent",
        "keys" => "bio current, about, about current"
    ],
    "contact" => [
        "title" => "Car Rent | Contact",
        "titleForTag" => "Contact us",
        "desc" => "Contact us for any question you have and maybe negotiate about price",
        "keys" => "contact us, contact current, contact rent a car"
    ],
    "login" => [
        "title" => "Car Rent | Login",
        "titleForTag" => "Log in",
        "desc" => "Log in to submit request to rent a car",
        "keys" => "log in current"
    ],
    "logout" => [
        "title" => "Car Rent | Logout",
        "titleForTag" => "Log out",
        "desc" => "Logout page of current",
        "keys" => "logout current, log out from car rent"
    ],
    "register" => [
        "title" => "Car Rent | register",
        "titleForTag" => "Register",
        "desc" => "Register here, to get a chance to rent a car",
        "keys" => "register current, register page of rent a car"
    ],
    "adminPanel" => [
        "title" => "Car Rent | Admin",
        "titleForTag" => "Hello admin",
        "desc" => "Admin panel of current website",
        "keys" => "adminpanel current, admin panel current"
    ]
]
```

```
        ],
        "profile" => [
            "title" => "Car Rent | User profie",
            "titleForTag" => "Hello user",
            "desc" => "User profie",
            "keys" => "profile at carrent, user at carrent"
        ],
        "author" => [
            "title" => "Car Rent | Author",
            "titleForTag" => "Author",
            "desc" => "Page about author of carrent website",
            "keys" => "about author of carrent, author carrent"
        ]
    ];

    return $meta[$page];

}

?>
```

3.1.3 config/config.php

```
<?php

// define("ABSOLUTE_PATH", $_SERVER['DOCUMENT_ROOT']);
define("ABSOLUTE_PATH", $_SERVER['DOCUMENT_ROOT'] . "/current");

// echo ABSOLUTE_PATH;

define("ENV_FAJL", ABSOLUTE_PATH . "/config/.env");
// define("CONNECTION", ABSOLUTE_PATH . "/config/connection.php");
define("USER_SLIKE", ABSOLUTE_PATH . "/assets/img");
define("SLIKE_FOLDER", "assets/img/");
define("LOG_FAJL", ABSOLUTE_PATH . "/data/log.txt");
define("GRESKE", ABSOLUTE_PATH . "/data/errors.txt");
define("SEPARATOR", "\t");

define("SERVER", izvuciIzEnv("SERVER"));
define("DBNAME", izvuciIzEnv("DBNAME"));
define("USERNAME", izvuciIzEnv("USERNAME"));
define("PASSWORD", izvuciIzEnv("PASSWORD"));

function izvuciIzEnv($naziv) {
    $podaci = file(ENV_FAJL);
    $vrednost = "";
    foreach ($podaci as $key => $value) {
        $red = explode("=", $value);
        if($red[0] == $naziv) {
            $vrednost = trim($red[1]);
        }
    }
    return $vrednost;
}
```

3.1.4 models/account/login.php

```
<?php

session_start();

require "../../config/connection.php";

if(isset($_POST['submitLogin'])) {
    $email = $_POST['email'];
    $pass = $_POST['password'];

    $regexEmail = "/^[\w\.-]{5,100}@[a-z]{2,10}\.[a-z]{2,20}$/";
}

$greskeL = [];

if(!preg_match($regexEmail, $email)) {
    $greskeL[] = "Email format: johnconnaly@gmail.com";
}

if(strlen($pass) < 5) {
    $greskeL[] = "Password must have 5 chars min";
}

if(count($greskeL) == 0) {

    $upit = "SELECT userId, firstName, lastName, email, idRole FROM users WHERE email = :email AND password = :pass";

    $stmt = $db->prepare($upit);

    $pass = md5($pass);

    try {
        $rez = $stmt->execute(["email" => $email, "pass" => $pass]);

        $user = $stmt->fetchAll();

        // echo $stmt->rowCount();

        if($stmt->rowCount() == 1) {
            $_SESSION['user'] = $user[0];

            if($user[0]['idRole'] == 1) {
                header("Location: ../../index.php?page=admin");
            } else {
                header("Location: ../../index.php?page=cars");
            }
        }
    }
}
```

```
var_dump($user[0]);  
  
// header("Location: ../../index.php?page=cars");  
  
} else {  
    $greskaMail = "Wrong combination of email/pass";  
  
    $_SESSION['greskaMail'] = $greskaMail;  
  
    header("Location: ../../index.php?page=login#formContainer");  
  
}  
  
}  
  
}  
  
}  
  
?  
?
```

3.1.5 models/account/register.php

```
<?php

session_start();

require "../config/connection.php";

if(isset($_POST['submitRegister'])) {
    $fname = $_POST['fname'];
    $lname = $_POST['lname'];
    $email = $_POST['email'];
    $password = $_POST['password'];
    $licenceNo = $_POST['licenceNo'];
    $yearsExp = $_POST['yearsExp'];

    $regexName = "/^([A-Z][a-z]{2,20})(\s[A-Z][a-z]{3,20}){0,2}$/";
    $regexEmail = "/^([A-z\d\.-]{5,100})@[a-z]{2,10}\.[a-z]{2,20}$/";
    $regexLicenceNo = "/^([A-Z][\d]{3}-[\d]{3}-[\d]{2}-[\d]{3}-[\d])$/";

    $greskeR = [];

    if(!preg_match($regexName, $fname)) {
        $greskeR[] = "First Name format: John";
    }

    if(!preg_match($regexName, $lname)) {
        $greskeR[] = "Last name format: Connaly";
    }

    if(!preg_match($regexEmail, $email)) {
        $greskeR[] = "Email format: johnconnaly@gmail.com";
    }

    if(strlen($password) < 5) {
        $greskeR[] = "Password must have 5 chars min";
    }

    if(!preg_match($regexLicenceNo, $licenceNo)) {
        $greskeR[] = "Licence No format: F255-921-50-094-0";
    }

    if($yearsExp == "" || $yearsExp < 0) {
        $greskeR[] = "Years of Experience must be positive number";
    }

    if(count($greskeR) == 0) {
```

```

$upit = "INSERT INTO users VALUES(NULL, :fname, :lname, :email, :pass, :licenceNo, :yearsOfExp, :datum, :idRole)";

$stmt = $db->prepare($upit);

$pass = md5($password);

$datum = date("Y-m-d H:i:s");

try {
    $rez = $stmt-
>execute(["fname" => $fname, "lname" => $lname, "email" => $email, "pass" => $pass, "licenceNo" => $licenceNo, "yearsOfExp" => $yearsExp, "datum" => $datum, "idRole" => 2]);

    // header Lokacije na login

    header("Location: ../../index.php?page=login");

} catch(PDOException $ex) {
    handleGresku($ex->getMessage());

    $greskeR[] = "User with this licence number already exist";

    $_SESSION['greskeRegister'] = $greskeR;

    header("Location: ../../index.php?page=register");
}

} else {
    $_SESSION['greskeRegister'] = $greskeR;

    header("Location: ../../index.php?page=register");
}

}

?>

```

3.1.51 models/checkEmail.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";

if(isset($_POST['btnCheckEmail'])) {
    $email = $_POST['email'];

    $upit = "SELECT * FROM users WHERE email = :email";

    $stmt = $db->prepare($upit);

    $stmt->bindParam(":email", $email);

    try {
        $stmt->execute();

        $rez = $stmt->fetchAll();

        if($stmt->rowCount()) {
            $odg = "Email is already taken";
        } else {
            $odg = true;
        }

        http_response_code(200);

    } catch(PDOException $ex) {
        http_response_code(400);
        $odg = "Enter valid email";
    }
}

} else {

    http_response_code(400);
    $odg = "Enter email first!";
}

echo json_encode(["odg" => $odg]);

?>
```

3.1.6 models/booking/canceluserBook.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";

if(isset($_POST['btnUserCancel'])) {
    $bookId = $_POST['bookId'];

    $upitCancel = "UPDATE booking SET status = 2 WHERE bookId = :bookId";

    $stmt = $db->prepare($upitCancel);

    try {
        $stmt->execute(['bookId' => $bookId]);

        $odg = true;
    } catch(PDOException $ex) {
        handleGresku($ex->getMessage());

        $odg = "You can't cancel";
    }

    http_response_code(200);

} else {
    http_response_code(400);

    $odg = "submit form first";
}

echo json_encode(['odg' => $odg]);

?>
```

3.1.7 models/booking/getAll.php

```
<?php

function getAllBooking() {
    // $upit = "SELECT b.*, u.firstName, u.lastName, u.email, u.yearsOfExp
, v.name, p.pricePerDay, br.brandName FROM booking b INNER JOIN users u ON b.u
serId = u.userId INNER JOIN vehicles v ON b.vehicleId = v.vehicleId INNER JOIN
brands br ON v.brandId = br.brandId";

        $upit = "SELECT b.*, v.name, p.pricePerDay, u.firstName, u.lastName, u
.email, u.yearsOfExp, br.brandName FROM ( SELECT DISTINCT vehicleId, pricePerD
ay FROM prices order by date DESC) as p INNER JOIN vehicles v ON p.vehicleId =
v.vehicleId INNER JOIN booking b ON v.vehicleId = b.vehicleId INNER JOIN bran
ds br ON v.brandId = br.brandId INNER JOIN users u ON b.userId = u.userId GROU
P BY p.vehicleId, b.status, u.userId ORDER BY u.userId";

    return izvrsiSelect($upit);
}

function getAllBookingUser() {
    $userId = $_SESSION['user']['userId'];

    $upit = "SELECT b.*, v.name, p.pricePerDay, u.firstName, u.lastName, u
.email, u.yearsOfExp, br.brandName FROM ( SELECT DISTINCT vehicleId, pricePerD
ay FROM prices order by date DESC) as p INNER JOIN vehicles v ON p.vehicleId =
v.vehicleId INNER JOIN booking b ON v.vehicleId = b.vehicleId INNER JOIN bran
ds br ON v.brandId = br.brandId INNER JOIN users u ON b.userId = u.userId WHE
RE u.userId = $userId GROUP BY p.vehicleId";

    return izvrsiSelect($upit);
}

?>
```

3.1.8 models/booking/handleBook.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "../functions.php";

if(isset($_POST['btnHandleBook'])) {
    $bookId = $_POST['bookId'];

    $status = $_POST['btnHandleBook'];

    $upitIzvuciCarId = "SELECT vehicleId, fromDate FROM booking WHERE bookId = $bookId";

    $stmtCar = $db->prepare($upitIzvuciCarId);

    $stmtCar->execute();

    $carResult = $stmtCar->fetch();

    $carId = $carResult['vehicleId'];

    $bookDatum = $carResult['fromDate'];

    $upitSveRezervacijeZaAuto = "SELECT * FROM booking WHERE vehicleId = $carId AND status = 1";

    $stmtRezervacije = $db->prepare($upitSveRezervacijeZaAuto);

    $stmtRezervacije->execute();

    $rezDatumi = $stmtRezervacije->fetchAll();

    $odg = true;

    foreach($rezDatumi as $key => $singleRez) {
        if($bookDatum < $singleRez['toDate']) {
            $odg = false;
            $odgTxt = "Car is already rented";
            break;
        }
    }

    if($odg) {
        $odgTxt = "Moze da se renta";
    }
}
```

```
$upitUpdate = "UPDATE booking SET status = 1 WHERE bookId = $bookId";  
  
$stmtUpdate = $db->prepare($upitUpdate);  
  
try {  
  
    $stmtUpdate->execute();  
  
    $odg = true;  
    $odgTxt = "Successfully approved";  
  
} catch(PDOException $ex) {  
    handleGresku($ex->getMessage());  
  
    $odg = false;  
  
    $odgTxt = "Select valid booking item";  
}  
}  
}  
}  
}  
else {  
    http_response_code(400);  
  
    $odg = false;  
    $odgTxt = "Submit forme prvo";  
}  
  
echo json_encode(['odgovor' => $odg, 'odgovorTxt' => $odgTxt]);  
  
?>
```

3.1.9 models/booking/handleBookCancel.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "../functions.php";

if(isset($_POST['btnHandleBook'])) {
    $bookId = $_POST['bookId'];

    $upitCancel = "UPDATE booking SET status = 2 WHERE bookId = $bookId";

    $stmtCancel = $db->prepare($upitCancel);

    try {
        $stmtCancel->execute();

        http_response_code(200);

        $odg = true;
        $odgTxt = "Successfully canceled booking";

    } catch(PDOException $ex) {
        $odg = false;
        $odgTxt = "Select valid booking item";
    }
}

echo json_encode(['odgovor' => $odg, 'odgovorTxt' => $odgTxt]);

?>
```

3.1.10 models/brands/delete.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "../functions.php";

if(isset($_POST['brandId'])) {
    $brandId = $_POST['brandId'];

    $upit = "DELETE FROM brands WHERE brandId = :brandId";

    $stmt = $db->prepare($upit);

    $stmt->bindParam(":brandId", $brandId);

    $status = "";

    try {
        $stmt->execute();

        $status = true;
        $text = "Successfully deleted";

        http_response_code(200);

    } catch(PDOException $ex) {
        handleGresku($ex->getMessage());

        $status = false;
        $text = "We don't have brand with specified id";
        http_response_code(400);
    }

    echo json_encode(["status" => $status, "text" => $text]);
}

?>
```

3.1.11 models/brands/getAll.php

```
<?php

// require "../functions.php";

function getAllBrands() {
    $upit = "SELECT * FROM brands ORDER BY brandId";

    return izvrsiSelect($upit);
}

?>
```

3.1.12 models/brands/getSingle.php

```
<?php

function getSingleBrand($brandId) {
    global $db;

    $upit = "SELECT * FROM brands WHERE brandId = :brandId";

    $stmt = $db->prepare($upit);

    $stmt->bindParam(":brandId", $brandId);

    try {
        $stmt->execute();

        $result = $stmt->fetch();

    } catch(PDOException $ex) {
        $result = "Sorry, we don't have brandy with specified id";
    }

    return $result;
}

?>
```

3.1.13 models/brands/insert.php

```
<?php

session_start();

header("Content-Type: application/json");

require "../../config/connection.php";
require "../../functions.php";

if(isset($_POST['btnInsert'])) {
    $brandName = $_POST['brandName'];

    $greske = [];
    $greskaBrand = "";

    $polje = [
        "type" => "input",
        "value" => $brandName,
        "regex" => "/^([A-Z][a-z\d\s\.\.]{3,40})$/",
        "example" => "Mercedes"
    ];

    proveriInput($polje);

    if(count($greske) == 0) {

        $upit = "INSERT INTO brands VALUES(NULL, :brandname)";

        $stmt = $db->prepare($upit);

        $stmt->bindParam(":brandname", $brandName);

        try {
            $rez = $stmt->execute();

            // echo "uneseno \t " . $rez;

            $status = true;
            $text = "Successfully inserted";

            http_response_code(200);
        } catch(PDOException $ex) {
            // u log fajl exception
            handleGresku($ex->getMessage());

            $text = "Brand with this name already exists";
            $status = false;
        }
    }
}
```

```
    http_response_code(400);
    // echo $greskaBrand;

    // $_SESSION['greskaBrand'] = $greskaBrand;

    // header("Location: index.php?page=admin&adminPage=brands");
}

} else {
    $status = false;
    $text = "Error, example format: " . $polje['example'];

    http_response_code(400);

}

echo json_encode(["status" => $status, "text" => $text]);
}

?>
```

3.1.14 models/brands/update.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "../functions.php";

if(isset($_POST['brandId'])) {
    $brandId = $_POST['brandId'];
    $brandName = $_POST['brandName'];

    $upit = "UPDATE brands SET brandName = :brandName WHERE brandId = :brandId";
    $stmt = $db->prepare($upit);

    try {
        $rez = $stmt->
>execute(["brandId" => $brandId, "brandName" => $brandName]);

        $status = true;
        $text = "Successfully updated";

        http_response_code(200);

    } catch(PDOException $ex) {
        handleGresku($ex->getMessage());

        $status = false;
        $text = "Select valid brand";

        http_response_code(400);
    }
}

echo json_encode(["status" => $status, "text" => $text]);
?>
```

3.1.15 models/cars/carDetails.php

```
<?php

function ispisitiOpreme($data) {
    $out = "";

    foreach ($data as $key => $single) {
        $flatKlasa = "";

        if($single['featureValue'] == 1) {
            $flatKlasa = "flaticon-check";
        } else {
            $flatKlasa = "flaticon-cancel";
        }

        $out .= "
            <li class='check'>
                <span class='$flatKlasa'></span>" . $single['featureName']
            . "</li>
        ";
    }

    echo $out;
}

if(isset($_GET['carId'])) {
    $carId = $_GET['carId'];

    $upitCar = "SELECT v.*, b.brandName, f.fuelType, p.pricePerDay, t.transmissionType, c.categoryName
        FROM ( SELECT DISTINCT vehicleId, pricePerDay FROM prices order by date DESC) as p INNER JOIN vehicles v ON p.vehicleId = v.vehicleId INNER JOIN brands b ON v.brandId = b.brandId INNER JOIN fuels f ON v.fuelId = f.fuelId INNER JOIN categories c ON v.categoryId = c.categoryId INNER JOIN transmissions t ON v.transmissionId = t.transmissionId WHERE v.vehicleId = :carId LIMIT 1";
    $stmt = $db->prepare($upitCar);
    $stmt->bindParam(":carId", $carId);

    $upitOprema = "SELECT vf.*, f.featureName FROM vehiclesfeatures vf INNER JOIN features f ON vf.featureId = f.featureId WHERE vehicleId = :carId";

    $stmtOprema = $db->prepare($upitOprema);
    $stmtOprema->bindParam(":carId", $carId);

    $upitReview = "SELECT r.reviewText, u.firstName, u.lastName FROM reviews r INNER JOIN users u ON r.userId = u.userId WHERE reviewStatus = 1 AND vehicleId = :carId";
```

```

$stmtReview = $db->prepare($upitReview);
$stmtReview->bindParam(":carId", $carId);

$upitSlike = "SELECT * FROM images WHERE vehicleId = :carId";

$stmtSlike = $db->prepare($upitSlike);
$stmtSlike->bindParam(":carId", $carId);

try {

    $stmt->execute();

    $car = $stmt->fetch();

    $stmt0prema->execute();

    $oprema = $stmt0prema->fetchAll();

    // var_dump($car);

    $stmtReview->execute();

    $reviews = $stmtReview->fetchAll();

    $stmtSlike->execute();

    $slike = $stmtSlike->fetchAll();

    $relatedCars = handleRelated();

} catch(PDOException $ex) {
    // nema auto sa tim id-jem
}

}

function handleRelated() {
    global $db;
    global $car;

    // $upitRelated = "SELECT * FROM cardetails WHERE fuelId = :carFuel AND transmissionId = :carTransmission AND vehicleId != :carId LIMIT 3";
    $upitRelated = "SELECT v.*, b.brandName, f.fuelType, p.pricePerDay, t.transmissionType
        FROM ( SELECT DISTINCT vehicleId, pricePerDay FROM prices order by date DESC ) as p INNER JOIN vehicles v ON p.vehicleId = v.vehicleId INNER JOIN brands b ON v.brandId = b.brandId INNER JOIN fuels f ON v.fuelId = f.fuelId INN

```

```
ER JOIN transmissions t ON v.transmissionId = t.transmissionId WHERE v.fuelId
= :carFuel AND v.vehicleId != :carId LIMIT 3";

$stmtRelated = $db->prepare($upitRelated);

$stmtRelated->bindParam(":carFuel", $car['fuelId']);
// $stmtRelated-
>bindParam(":carTransmission", $car['transmissionId']);
$stmtRelated->bindParam(":carId", $car['vehicleId']);

try {
    $stmtRelated->execute();

    $rezRelated = $stmtRelated->fetchAll();
} catch(PDOException $ex) {

}

return $rezRelated;
}

?>
```

3.1.16 models/cars/delete.php

```
<?php

require "../../config/connection.php";
require "../../functions.php";
require "functionsCar.php";

header("Content-Type: application/json");

if(isset($_POST['btnDeleteCar'])) {
    $carId = $_POST['carId'];

    $upitDelete = "DELETE FROM vehicles WHERE vehicleId = :carId";

    $stmt = $db->prepare($upitDelete);

    try {
        $stmt->execute(["carId" => $carId]);

        $odg = true;
    } catch(PDOException $ex) {
        handleGresku($ex->getMessage());

        $odg = "Select valid car to delete";
    }
} else {
    $odg = "Submit form first!";
}

echo json_encode(["odg" => $odg]);

?>
```

3.1.17 models/cars/filterCars.php

```
<?php

header("Content-Type: application/json");

require "../../config/connection.php";
// require "models/initLoad.php";

$limit = 6;
$offset = 0;

if(isset($_GET['pageNo'])) {
    $offset = ($_GET['pageNo'] - 1) * $limit;
}

$upitBrPoslova = "SELECT count(*) as total FROM vehicles";
$pripremiBr = $db->prepare($upitBrPoslova);
$pripremiBr->execute();
$brojPoslova = $pripremiBr->fetch();

$total = $brojPoslova[0];

$brStrana = ceil($total / $limit);

$queryStr = "";

if(isset($_GET['btnFilter'])) {

    $upit = "SELECT v.*, b.brandName, CONCAT(b.brandName, ' ', v.name) as vehicleName, t.transmissionType, f.fuelType, p.pricePerDay as latestPrice FROM ( SELECT priceId, vehicleId, pricePerDay, date FROM prices ORDER BY date DESC) as p INNER JOIN vehicles v ON p.vehicleId = v.vehicleId INNER JOIN brands b ON v.brandId = b.brandId INNER JOIN fuels f ON v.fuelId = f.fuelId INNER JOIN transmissions t ON v.transmissionId = t.transmissionId WHERE 1";

    if(isset($_GET['searchKey']) && $_GET['searchKey'] != "") {
        $key = addslashes(strtolower($_GET['searchKey']));
        $upit .= " AND LOWER(CONCAT(b.brandName, ' ', v.name)) LIKE '%$key%'";
        $queryStr .= "&searchKey=$key";
    }

    if(isset($_GET['category']) && $_GET['category'] != "0") {
        $cat = $_GET['category'];
        $upit .= " AND v.categoryId = $cat";
        $queryStr .= "&category=$cat";
    } else {
        $queryStr .= "&category=0";
    }
}
```

```

if(isset($_GET['fuel']) && $_GET['fuel'] != "0") {
    $fuelId = $_GET['fuel'];
    $upit .= " AND v.fuelId = $fuelId";
    $queryStr .= "&fuel=$fuelId";
} else {
    $queryStr .= "&fuel=0";
}

if(isset($_GET['transmission']) && $_GET['transmission'] != "0") {
    $transId = $_GET['transmission'];
    $upit .= " AND v.transmissionId = $transId";
    $queryStr .= "&transmission=$transId";
} else {
    $queryStr .= "&transmission=0";
}

if(isset($_GET['minVal']) && !empty($_GET['minVal'])) {
    $minVal = $_GET['minVal'];
    $upit .= " AND p.pricePerDay >= $minVal";
    $queryStr .= "&minVal=$minVal";
} else {
    $queryStr .= "&minVal=0";
}

if(isset($_GET['maxVal']) && !empty($_GET['maxVal'])) {
    $maxVal = $_GET['maxVal'];
    $upit .= " AND p.pricePerDay <= $maxVal";
    $queryStr .= "&maxVal=$maxVal";
} else {
    $queryStr .= "&maxVal=0";
}

$upit .= " AND p.priceId = ( SELECT MAX(priceId) FROM prices where vehicle
Id = v.vehicleID)";

$upitSvi = $upit;

$pripremiSve = $db->prepare($upitSvi);
$pripremiSve->execute();

$rezSvi = $pripremiSve->fetchAll();

$total = count($rezSvi);

$brStrana = ceil($total / $limit);

$upit .= " LIMIT $limit OFFSET $offset";

```

```

// echo $upit;

$pripremi = $db->prepare($upit);

$pripremi->execute();

$cars = $pripremi->fetchAll();

} else {

    $upit = "SELECT v.*, b.brandName, CONCAT(b.brandName, ' ', v.name) as vehicleName, t.transmissionType, f.fuelType, p.pricePerDay as latestPrice FROM ( SELECT DISTINCT vehicleId, pricePerDay FROM prices order by date DESC) as p INNER JOIN vehicles v ON p.vehicleId = v.vehicleId INNER JOIN brands b ON v.brandId = b.brandId INNER JOIN fuels f ON v.fuelId = f.fuelId INNER JOIN transmissions t ON v.transmissionId = t.transmissionId WHERE 1 GROUP BY p.vehicleId LIMIT $limit OFFSET $offset";

    $pripremi = $db->prepare($upit);

    $pripremi->execute();

    $cars = $pripremi->fetchAll();
}

$queryStr .= "&btnFilter=Submit";

// echo "<p>$upit</p>";
// echo "<p>$upitSvi</p>";

$rez = [
    "cars" => $cars,
    "queryStr" => $queryStr,
    "brStrana" => $brStrana,
    "upit" => $upit
];

echo json_encode($rez);

?>

```

3.1.18 models/cars/filterFromIndex.php

```
<?php if(isset($_GET['btnFilterIndex'])): ?>

<script type="text/javascript">
window.onload = function() {
    let searchKey = "<?= $_GET['searchKey'] ?>";
    let category = "<?= $_GET['category'] ?>";
    let fuel = "<?= $_GET['fuel'] ?>";

    var xhr = new XMLHttpRequest();

    xhr.open("GET", "models/cars/filterCars.php?page=cars&searchKey" + searchKey + "&category=" + category + "&fuel=" + fuel + "&btnFilter=true");

    xhr.onload = function() {
        let odg = JSON.parse(this.response);

        if(odg.cars.length) {
            cars = populateCarDiv(odg.cars);
        } else {
            cars = "<h3 class='mt-5'>We don't have any car for selected criteria</h3>";
        }

        document.querySelector("#carsContainer").innerHTML = cars;

        let paginacija = makePagination(odg);

        document.querySelector("#pagination").innerHTML = paginacija;

        document.querySelectorAll(".linkPagination").forEach(item => {
            item.addEventListener("click", promeniStranuPaginacija);
        });

    }

    xhr.send();

    document.querySelector("#searchKey").value = searchKey;
    document.querySelector("#category").value = category;
    document.querySelector("#fuel").value = fuel;

}

</script>

<?php endif; ?>
```

3.1.19 models/cars/functionsCars.php

```
<?php

function obradiVelicinuSlike($fajl, $novaPutanja) {
    list($sirina, $visina) = getimagesize($novaPutanja);

    $novaSirinaVelika = 720;
    $novaSirinaMala = 200;

    $procenatPromene = $novaSirinaVelika / $sirina;
    $novaVisina = $visina * $procenatPromene;

    $novaSlika = imagecreatetruecolor($novaSirinaVelika, $novaVisina);

    $izvor = "";

    if($fajl['type'] == "image/jpeg") {
        $izvor = imagecreatefromjpeg($novaPutanja);
    } else {
        $izvor = imagecreatefrompng($novaPutanja);
    }

    imagecopyresized($novaSlika, $izvor, 0, 0, 0, 0, $novaSirinaVelika, $novaVisina, $sirina, $visina);

    imagejpeg($novaSlika, $novaPutanja);

    // za malu sliku
    $procenatMala = $novaSirinaMala / $sirina;
    $novaVisinaMala = $visina * $procenatMala;

    $malaSlika = imagecreatetruecolor($novaSirinaMala, $novaVisinaMala);

    imagecopyresized($malaSlika, $izvor, 0, 0, 0, 0, $novaSirinaMala, $novaVisinaMala, $sirina, $visina);

    $nizIme = explode(".", $novaPutanja);
    $nizIme[0] .= "-small";

    $novoImeSlike = implode(".", $nizIme);

    imagejpeg($malaSlika, $novoImeSlike);

}

?>
```

3.1.20 models/cars/getAll.php

```
<?php

    // require "../functions.php";

    function getAllCars() {
        $upit = "SELECT v.*, b.brandName, f.fuelType, p.pricePerDay, t.transmissionType
                FROM ( SELECT DISTINCT vehicleId, pricePerDay FROM prices ORDER BY date DESC) AS p
                INNER JOIN vehicles v ON p.vehicleId = v.vehicleId
                INNER JOIN brands b ON v.brandId = b.brandId
                INNER JOIN fuels f ON v.fuelId = f.fuelId
                INNER JOIN transmissions t ON v.transmissionId = t.transmissionId GROUP BY v.vehicleId";

        return izvrsiSelect($upit);
    }

?>
```

3.1.21 models/cars/getSingle.php

```
<?php

    function getSingleCar() {
        global $carId;

        $upit = "SELECT v.*, b.brandName, f.fuelType, p.pricePerDay, t.transmissionType, c.categoryName
                FROM ( SELECT DISTINCT vehicleId, pricePerDay FROM prices ORDER BY date DESC) AS p
                INNER JOIN vehicles v ON p.vehicleId = v.vehicleId
                INNER JOIN brands b ON v.brandId = b.brandId
                INNER JOIN fuels f ON v.fuelId = f.fuelId
                INNER JOIN categories c ON v.categoryId = c.categoryId
                INNER JOIN transmissions t ON v.transmissionId = t.transmissionId WHERE v.vehicleId = $carId";

        $upitOprema = "SELECT vf.*, f.featureName FROM vehiclesfeatures vf
                INNER JOIN features f ON vf.featureId = f.featureId WHERE vehicleId = $carId";

        $upitSlike = "SELECT * FROM images WHERE vehicleId = $carId";

        $basicInfo = izvrsiSelect($upit);

        $oprema = izvrsiSelect($upitOprema);

        $slike = izvrsiSelect($upitSlike);

        $rez = ["basic" => $basicInfo[0], "oprema" => $oprema, "slike" => $slike];
    }

?>
```

```
        return $rez;
    }

function getSingleIndex() {
    global $carId;
    global $db;

    $upit = "SELECT p.pricePerDay, v.*, b.brandName, f.fuelType, p.pricePerDay, t.transmissionType, c.categoryName
        FROM ( SELECT DISTINCT vehicleId, pricePerDay FROM prices order by date DESC) as p INNER JOIN vehicles v ON p.vehicleId = v.vehicleId INNER JOIN brands b ON v.brandId = b.brandId INNER JOIN fuels f ON v.fuelId = f.fuelId INNER JOIN categories c ON v.categoryId = c.categoryId INNER JOIN transmissions t ON v.transmissionId = t.transmissionId WHERE v.vehicleId = $carId";

    $stmt = $db->prepare($upit);

    $stmt->execute();

    return $stmt->fetch();
}

?>
```

3.1.22 models/cars/handleBooking.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";

if($_POST['btnRentRequest']) {
    $carId = $_POST['carId'];
    $userId = $_POST['userId'];
    $fromDate = $_POST['fromDate'];
    $toDate = $_POST['toDate'];

    $upitCheck = "SELECT * FROM booking WHERE vehicleId = ? AND userId = ? AND status = 0"; //proveriti status dal mu je odobreno rentanje

    $stmtCheck = $db->prepare($upitCheck);

    $stmtCheck->execute([$carId, $userId]);

    $rez = $stmtCheck->fetchAll();

    $odg = "Datumi su u redu";

    $statusOdg = true;

    foreach ($rez as $key => $single) {
        if($fromDate <= $single['toDate']) {
            $statusOdg = false;
            $odg = "You already submitted request to this car";
            break;
        }
    }

    // if($fromDate <= $rez[0]['toDate']) {
    //     $odg = "Vec ste rentalni auto";
    // } else {
    //     $odg = "Datumi su u redu";
    // }

    if($statusOdg) {
        $upit = "INSERT INTO booking VALUES (NULL, :userId, :vehicleId, :fromDate, :toDate, :status, :currentDate)";

        $stmt = $db->prepare($upit);

        $status = 0;
    }
}
```

```
$datum = date("Y-m-d H:i:s");

try {
    $rezRequest = $stmt-
>execute(["userId" => $userId, "vehicleId" => $carId, "fromDate" => $fromDate,
"toDate" => $toDate, "status" => $status, "currentDate" => $datum]);

} catch(PDOException $ex) {
    $odg = $ex->getMessage();
}

$odg = "You successfully requested. Wait for admin's approval";

}

} else {
    $statusOdg = false;
    $odg = "Ulazi u else ";
    http_response_code(400);
}

echo json_encode(['statusOdg' => $statusOdg, 'odg' => $odg]);
```

?>

3.1.23 models/cars/insert.php

```
<?php

require "../../config/connection.php";
require "../../functions.php";
require "functionsCar.php";

header("Content-Type: application/json");

if(isset($_POST['btnInsertCar'])) {
    extract($_POST);
    // var_dump($_POST, $_FILES);

    // echo "<hr />";

    $coverSlika = $_FILES['coverPhoto'];
    $sekSlika1 = $_FILES['otherPhoto'];
    $sekSlika2 = $_FILES['otherPhoto2'];

    $vrednosti = [
        [
            "type" => "input",
            "value" => $carName,
            "regex" => "/^([A-z\s\d\.\.]{2,30})$/",
            "example" => "Sportage"
        ],
        [
            "type" => "ddl",
            "value" => $brandName,
            "error" => "Select brand"
        ],
        [
            "type" => "ddl",
            "value" => $categoryName,
            "error" => "Select category"
        ],
        [
            "type" => "textarea",
            "value" => $desc,
            "error" => "Description must have at least 10 chars"
        ],
        [
            "type" => "text",
            "value" => $modelYear,
            "regex" => "/^(19|20)[0-9]{2}$/",
            "example" => "2000"
        ],
        [
            "type" => "text"
        ]
    ];
}
```

```

        "type" => "number",
        "value" => $seats,
        "error" => "Enter positive number"
    ],
    [
        "type" => "number",
        "value" => $doors,
        "error" => "Enter positive number"
    ],
    [
        "type" => "ddl",
        "value" => $fuelType,
        "error" => "Select fuel type"
    ],
    [
        "type" => "ddl",
        "value" => $transmissiontype,
        "error" => "Select transmission type"
    ],
    [
        "type" => "number",
        "value" => $mileage,
        "error" => "Enter positive number"
    ],
    [
        "type" => "number",
        "value" => $luggage,
        "error" => "Enter positive number"
    ],
    [
        "type" => "numberPrice",
        "value" => $price,
        "error" => "Enter positive number between 10 and 500"
    ],
];
$oprema = [
    [
        "idProp" => "aircondition",
        "text" => "Airconditions"
    ],
    [
        "idProp" => "breakSystem",
        "text" => "AntiLock Breaking System"
    ],
    [
        "idProp" => "leatherSeats",
        "text" => "Leather Seats"
    ]
];

```

```

        ],
        [
            "idProp" => "brakeAssist",
            "text" => "Brake Assist"
        ],
        [
            "idProp" => "crashSenson",
            "text" => "Crash Sensor"
        ],
        [
            "idProp" => "onboardPc",
            "text" => "Onboard computer"
        ],
        [
            "idProp" => "gps",
            "text" => "GPS"
        ],
        [
            "idProp" => "locking",
            "text" => "Central Locking"
        ],
        [
            "idProp" => "abs",
            "text" => "ABS"
        ],
        [
            "idProp" => "bluetooth",
            "text" => "Bluetooth"
        ],
        [
            "idProp" => "childSeat",
            "text" => "Child Seat"
        ],
        [
            "idProp" => "parkingSensor",
            "text" => "Parking Sensors"
        ]
    ];
}

$greske = [];

foreach ($vrednosti as $el) {
    if($el["type"] == "number" || $el['type'] == "ddl" || $el['type'] == "numberPrice") {
        proveriNumber($el);
    } else if($el['type'] == "input") {
        proveriInput($el);
    } else if($el['type'] == "textarea") {

```

```

        proveriTExtarea($el);
    }
}

$opremaVrednosti = [];

foreach($oprema as $single) {
    if(isset($_POST[$single['idProp']])) {
        $opremaVrednosti[] = 1;
    } else {
        $opremaVrednosti[] = 0;
    }
}

// var_dump($opremaVrednosti);

// $folderCar = USER_SLIKE . "/$carName/";

// if(!file_exists($folderCar)) {
//     mkdir($folderCar, 0777, true);
// } else {
//     // echo "Folder already exists <br />";
// }

$folderCar = USER_SLIKE;

$mainImg = $folderCar . "/" . $coverSlika['name'];

$sekImg = $folderCar . "/" . $sekSlika1['name'];

$sekImg2 = $folderCar . "/" . $sekSlika2['name'];

$putanjaZaBazu ="/" . $coverSlika['name'];

$putanjaZaBazu1 ="/" . $sekSlika1['name'];
$putanjaZaBazu2 ="/" . $sekSlika2['name'];

$rezSlike = move_uploaded_file($coverSlika['tmp_name'], $mainImg);

$rezSekSlike = move_uploaded_file($sekSlika1['tmp_name'], $sekImg);
$rezSekSlike2 = move_uploaded_file($sekSlika2['tmp_name'], $sekImg2);

if(count($greske) == 0 || $rezSlike == 1) {

    obradiVelicinuSlike($_FILES['coverPhoto'], $mainImg);

    obradiVelicinuSlike($_FILES['otherPhoto'], $sekImg);
}

```

```

obradiVeliciNuSlike($_FILES['otherPhoto2'], $sekImg2);

// echo "Uneli ste sve bez greske";

$upit = "INSERT INTO vehicles VALUES(NULL, :carName, :brandId, :ca
toryId, :desc, :mainImg, :modelYear, :mileage, :transmissionId, :seats, :doo
rs, :luggage, :fuelId, :insertedat)";

$insertedat = date("Y-m-d H:i:s");

$dataZaInsert = [
    "carName" => $carName,
    "brandId" => $brandName,
    "categoryId" => $categoryName,
    "desc" => $desc,
    "mainImg" => $putanjaZaBazu,
    "modelYear" => $modelYear,
    "mileage" => $mileage,
    "transmissionId" => $transmissiontype,
    "seats" => $seats,
    "doors" => $doors,
    "luggage" => $luggage,
    "fuelId" => $fuelType,
    "insertedat" => $insertedat
];
// echo $rezSlike . " slika je prenestena <br />";

// $rezInsert = izvrsiInsert($upit, $dataZaInsert);

$stmt = $db->prepare($upit);

try {
    $stmt->execute($dataZaInsert);

} catch(PDOException $ex) {
    handleGresku($ex->getMessage());
}

$vehicleId = $db->lastInsertId();

// echo $rezInsert . "<br />";

// echo "Poslednji indeks je $vehicleId";

$featureId = 7;

foreach($opremaVrednosti as $key => $ovrednost) {

```

```

$upitOprema = "INSERT INTO vehiclesfeatures VALUES (NULL, :featureId, :carId, :val)";

$featuresData = [$featureId, $vehicleId, $ovrednost];

// $rez = izvrsiInsert($upit, $featuresData);

$stmtOprema = $db->prepare($upitOprema);

try {
    $stmtOprema-
>execute(["featureId" => $featureId, "carId" => $vehicleId, "val" => $ovrednos
t]);

        // echo "uspesno insert u vehiclefeatures";

} catch (PDOException $ex) {
    echo $ex->getMessage();
}

$featureId++;

// echo "$rez <br /> ";

// echo $upit . "<<br />" . $featureId;
}

$upitCena = "INSERT INTO prices VALUES (NULL, :vehicleId, :price,
:datum)";

$stmtCena = $db->prepare($upitCena);

$datum = date("Y-m-d H:i:s");

try {
    $stmtCena-
>execute(["vehicleId" => $vehicleId, "price" => $price, "datum" => $datum]);
} catch(PDOException $ex) {
    // $odg = ""

    echo $ex->getMessage();
}

$slikeZaBazu = [$putanjaZaBazu1, $putanjaZaBazu2];

foreach ($slikeZaBazu as $key => $slika) {
    $upitSlike = "INSERT INTO images VALUES (NULL, ?, ?)";

```

```
$stmtSlike = $db->prepare($upitSlike);

$stmtSlike->execute([$slika, $vehicleId]);
}

http_response_code(200);

$odg = true;

} else {
    // echo "Ima gresaka";

    http_response_code(400);
    $odg = $greske;
}

} else {
    http_response_code(400);
    $odg = "Submit form first!";
}

echo json_encode(["odg" => $odg]);
```

?>

3.1.24 models/cars/update.php

```
<?php

require "../../config/connection.php";
require "../../functions.php";
require "functionsCar.php";

header("Content-Type: application/json");

if(isset($_POST['btnUpdateCar'])) {
    extract($_POST);

    require "dataForm.php";

    $greske = [];

    foreach ($vrednosti as $el) {
        if($el["type"] == "number" || $el['type'] == "ddl" || $el['type'] == "numberPrice") {
            proveriNumber($el);
        } else if($el['type'] == "input") {
            proveriInput($el);
        } else if($el['type'] == "textarea") {
            proveriTextarea($el);
        }
    }

    $opremaVrednosti = [];

    foreach($oprema as $single) {
        if(isset($_POST[$single['idProp']])) {
            $opremaVrednosti[] = 1;
        } else {
            $opremaVrednosti[] = 0;
        }
    }

    $folderCar = USER_SLIKE;

    if(!file_exists($folderCar)) {
        mkdir($folderCar, 0777, true);
    }

    if(isset($_FILES['coverPhoto'])) {
        $coverSlika = $_FILES['coverPhoto'];

        $mainImg = $folderCar . "/" . $coverSlika['name'];
    }
}
```

```

$putanjaZaBazu = "/" . $coverSlika['name'];

$rezSlike = move_uploaded_file($coverSlika['tmp_name'], $mainImg);

obradiVelicinuSlike($_FILES['coverPhoto'], $mainImg);
}

if(isset($_FILES['otherPhoto'])) {
    $sekSlika1 = $_FILES['otherPhoto'];

    $sekImg = $folderCar . "/" . $sekSlika1['name'];

    $putanjaZaBazu1 = "/" . $sekSlika1['name'];

    $rezSekSlike = move_uploaded_file($sekSlika1['tmp_name'], $sekImg)
;

    obradiVelicinuSlike($_FILES['otherPhoto'], $sekImg);
}

if(isset($_FILES['otherPhoto2'])) {
    $sekSlika2 = $_FILES['otherPhoto2'];

    $sekImg2 = $folderCar . strtolower($sekSlika2['name']);

    $putanjaZaBazu2 = "/" . strtolower($carName) . "/" . $sekSlika2['name'];

    $rezSekSlike2 = move_uploaded_file($sekSlika2['tmp_name'], $sekImg2);

    obradiVelicinuSlike($_FILES['otherPhoto2'], $sekImg2);
}

if(count($greske) == 0) {
    $upit = "UPDATE vehicles SET name = :carName, brandId = :brandId,
categoryId = :categoryId, description = :desc,";

    if(isset($coverSlika)) {
        $upit .= " mainImg = $putanjaZaBazu,";

    }

    $upit .= " modelYear = :modelYear, mileage = :mileage, transmissionId = :transmissionId, seats = :seats, doors = :doors, luggage = :luggage, fuelId = :fuelId WHERE vehicleId = :carId";

    $dataZaUpdate = [
        "carName" => $carName,

```

```

        "brandId" => $brandName,
        "categoryId" => $categoryName,
        "desc" => $desc,
        "modelYear" => $modelYear,
        "mileage" => $mileage,
        "transmissionId" => $transmissiontype,
        "seats" => $seats,
        "doors" => $doors,
        "luggage" => $luggage,
        "fuelId" => $fuelType,
        "carId" => $carId
    ];
}

$stmt = $db->prepare($upit);

try {
    $stmt->execute($dataZaUpdate);

    // $odg = "promenjeno";

} catch(PDOException $ex) {
    handleGresku($ex->getMessage());
    $odg = "You must select valid vehicle id";
}

$featureId = 7;

foreach($opremaVrednosti as $key => $ovrednost) {
    $upitOprema = "UPDATE vehiclesfeatures SET featureValue = :val
WHERE featureId = :featureId AND vehicleId = :carId";

    $featuresData = [$featureId, $carId, $ovrednost];

    // $rez = izvrsiInsert($upit, $featuresData);

    $stmtOprema = $db->prepare($upitOprema);

    try {
        $stmtOprema-
>execute(["val" => $ovrednost, "featureId" => $featureId, "carId" => $carId]);

        // echo "uspesno insert u vehiclefeatures";

    } catch (PDOException $ex) {
        echo $ex->getMessage();
    }

    $featureId++;
}

```

```
    }

$upitCena = "INSERT INTO prices VALUES (NULL, :vehicleId, :price,
:datum)";

$stmtCena = $db->prepare($upitCena);

$datum = date("Y-m-d H:i:s");

try {
    $stmtCena-
>execute(["vehicleId" => $carId, "price" => $price, "datum" => $datum]);
} catch(PDOException $ex) {
    // $odg = ""

    echo $ex->getMessage();
}

http_response_code(200);

$odg = true;

} else {
    $odg = $greske;
}

}

} else {
    $odg = "Nije kliknuto na submit";
}

echo json_encode(["odg" => $odg]);

?>
```

3.1.25 models/reviews/changeState.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "../functions.php";

if(isset($_POST['btnChangeState'])) {
    $reviewId = $_POST['reviewId'];
    $status = $_POST['reviewStatus'];

        $upit = "UPDATE reviews SET reviewStatus = :noviStatus WHERE reviewId = :reviewId";

    $stmt = $db->prepare($upit);

    try {
        $stmt-
>execute(["noviStatus" => $status, "reviewId" => $reviewId]);

        http_response_code(200);

        $odg = true;
    } catch(PDOException $ex) {
        //zabelezi gresku
        handleGresku($ex->getMessage());

        http_response_code(400);

        $odg = $ex->getMessage();

        $odgTxt = "Select valid review item";
    }

} else {
    http_response_code(400);

    $odg = false;
    $odgTxt = "SUbmit forme first!";
}

echo json_encode(['odg' => $odg]);

?>
```

3.1.26 models/reviews/getAll.php

```
<?php

function getCarsByUser() {
    $userId = $_SESSION['user']['userId'];

    $upit = "SELECT r.*, v.name, b.brandName FROM reviews r INNER JOIN vehicles v ON r.vehicleId = v.vehicleId INNER JOIN brands b ON v.brandId = b.brandId WHERE userId = $userId";

    return izvrsiSelect($upit);
}

function getCarsForInsertReview() {
    $userId = $_SESSION['user']['userId'];

    $upit = "SELECT b.vehicleId, v.name, br.brandName
    FROM booking b
        INNER JOIN vehicles v ON b.vehicleId = v.vehicleId INNER JOIN brands br ON v.brandId = br.brandId
        WHERE userId = $userId AND status = 1 AND b.vehicleId NOT IN ( SELECT vehicleId FROM reviews WHERE userId = $userId)";

    return izvrsiSelect($upit);
}

function getReviewsAdmin() {
    $upit = "SELECT r.*, u.firstName, u.lastName, v.name, br.brandName
    FROM reviews r INNER JOIN users u ON r.userId = u.userId INNER JOIN vehicles v ON r.vehicleId = v.vehicleId INNER JOIN brands br ON v.brandId = br.brandId";

    return izvrsiSelect($upit);
}

?>
```

3.1.27 models/reviews/userDelete.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "../functions.php";

if(isset($_POST['deleteReviewBtn'])) {
    $reviewId = $_POST['reviewId'];

    $upit = "DELETE FROM reviews WHERE reviewId = :reviewId";

    $stmt = $db->prepare($upit);

    try {
        $stmt->execute(["reviewId" => $reviewId]);

        http_response_code(200);

        $odg = true;
    } catch(PDOException $ex) {
        handleGresku($ex->getMessage());

        http_response_code(400);

        $odg = "No review with specified id";
    }
} else {

    http_response_code(400);
    $odg = "Submit form first";
}

echo json_encode(["odg" => $odg]);

?>
```

3.1.28 models/reviews/userInsert.php

```
<?php

    session_start();

    header("Content-Type: application/json");

    require "../../config/connection.php";
    require "../functions.php";

    if(isset($_POST['btnInsertReview'])) {
        $reviewTxt = $_POST['reviewTxt'];
        $userId = $_SESSION['user']['userId'];
        $vehicleId = $_POST['vehicleName'];

        $upit = "INSERT INTO reviews VALUES (NULL, :userId, :vehicleId, :reviewText, :reviewStatus)";

        $stmt = $db->prepare($upit);

        $reviewStatus = 0;

        try {
            $stmt-
>execute(['userId' => $userId, "vehicleId" => $vehicleId, "reviewText" => $reviewTxt, "reviewStatus" => $reviewStatus]);

            http_response_code(200);

            $odg = true;
        } catch(PDOException $ex) {
            handleGresku($ex->getMessage());

            http_response_code(400);

            $odg = "You can't add review to this vehicle";
        }
    } else {

        http_response_code(400);
        $odg = "Submit form first";
    }

    echo json_encode(["odg" => $odg]);
?>
```

3.1.29 models/reivews/userUpdate.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "../functions.php";

if(isset($_POST['editReviewBtn'])) {
    $reviewId = $_POST['reviewId'];
    $reviewTxt = $_POST['reviewTxt'];

    $upit = "UPDATE reviews SET reviewText = :reviewTxt WHERE reviewId = :reviewId";

    $stmt = $db->prepare($upit);

    try {
        $stmt-
>execute(["reviewTxt" => $reviewTxt, "reviewId" => $reviewId]);

        http_response_code(200);

        $odg = true;
    } catch(PDOException $ex) {
        handleGresku($ex->getMessage());

        http_response_code(400);

        $odg = "You can't review this car";
    }
} else {

    http_response_code(400);
    $odg = "Submit form first";
}

echo json_encode(["odg" => $odg]);

?>
```

3.1.30 models/functions.php

```
<?php

function izvrsiSelect($upit) {
    global $db;
    $pripremi = $db->prepare($upit);
    $pripremi->execute();

    return $pripremi->fetchAll();
}

// ne moze jer mora drugaciji exception da se baca za razlicite op
function izvrsiInsert($upit, $data) {
    global $db;
    $stmt = $db->prepare($upit);

    $rez = "";

    try {
        $rez = $stmt->execute($data);

    } catch(PDOException $ex) {
        $rez = $ex->getMessage();
    }

    return $rez;
}

function popuniListe($data, $idProp, $valueProp) {
    $out = "";

    for ($i = 0; $i < count($data); $i++) {
        $out .= "<option value='".$data[$i][$idProp] . "'>" . $data[$i][$valueProp] . "</option>";

        // $out .= "<option value=$data[$i][$idProp]>$data[$i][$valueProp]</option>"; zasto se ovde greska javlja
    }

    return $out;
}

function stampajListu($data, $idProp, $valueProp, $text) {
    $out = "
        <div class='form-group'>
            <label for='seats'>$text</label>
    
```

```

        <select class='form-
control' name='$valueProp' id='$valueProp'>
            <option value='0'>Select </option>
        ";

        $out .= popuniListe($data, $idProp, $valueProp);

        $out .= "
            </select>
        </div>
    ";

    echo $out;
}

// STAMPANJE

function stampajInput($idAttr, $text, $inputType) {
    $out =
        <div class='form-group'>
            <label for='seats'$text</label>
            <input type='".$ . $inputType . "' class='form-
control' name='".$idAttr' id='".$idAttr' />
            <small id='".$ . $idAttr . "Err' class='form-text text-
danger error-field'>Valid format: </small>
        </div>
    ";

    echo $out;
}

function stampajChbs($data, $idProp, $valueProp, $text) {
    $out =
        <div class='form-group'>
            <p>$text</p>
    ";

    for ($i = 0; $i < count($data); $i++) {

        $out .= "
            <div class='custom-control custom-radio'>
                <input type='radio' class='custom-control-
input' id='customRadio$i' name='".$ . str_replace(" ", "", $text) . "' value='".$ .
$data[$i][$idProp] . "' />
                <label class='custom-control-
label' for='customRadio$i'>" . $data[$i][$valueProp] . "</label>
    ";
}

```

```

                </div>
            ";
        }

        $out .= "
            <small id=''" . str_replace(" ", "", $text) . "Err' class='form-
-text text-danger error-field'>Select one option</small>
        </div>
    ";
}

echo $out;
}

function stampajOpremu($opremaIzBaze = NULL) {

    $oprema = [
        [
            "idProp" => "aircondition",
            "text" => "Airconditions"
        ],
        [
            "idProp" => "breakSystem",
            "text" => "AntiLock Breaking System"
        ],
        [
            "idProp" => "leatherSeats",
            "text" => "Leather Seats"
        ],
        [
            "idProp" => "brakeAssist",
            "text" => "Brake Assist"
        ],
        [
            "idProp" => "crashSenson",
            "text" => "Crash Sensor"
        ],
        [
            "idProp" => "onboardPc",
            "text" => "Onboard computer"
        ],
        [
            "idProp" => "gps",
            "text" => "GPS"
        ],
        [
            "idProp" => "locking",
            "text" => "Central Locking"
        ],
    ];
}

```

```

        [
            "idProp" => "abs",
            "text" => "ABS"
        ],
        [
            "idProp" => "bluetooth",
            "text" => "Bluetooth"
        ],
        [
            "idProp" => "childSeat",
            "text" => "Child Seat"
        ],
        [
            "idProp" => "parkingSensor",
            "text" => "Parking Sensors"
        ]
    ];

$out = "";

// var_dump($accessories[1]['idProp'], $accessories[1]['text']);

for ($i = 0; $i < count($oprema); $i++) {

    if(isset($opremaIzBaze) && $opremaIzBaze[$i]['featureValue'] == 1)
    {
        $cekirano = "checked='checked'";
    } else {
        $cekirano = "";
    }

    $out .= "
        <div class='col-md-4 my-2'>
            <div class='form-check form-check-inline'>
                <input class='oprema form-check-input' type='checkbox' id='".$oprema[$i]["idProp"] . "' name='".$oprema[$i]["idProp"] . "' value='1' $cekirano />
                <label class='form-check-label' for='".$oprema[$i]["idProp"] . "'> ". $oprema[$i]["text"] . "</label>
            </div>
        </div>
    ";
}

echo $out;
}

```

```

function stampajFajl($idProp, $textProp) {
    $out = "
        <label for='".$idProp.'>$textProp</label>
        <input type='file' class='form-
control' name='".$idProp' id='".$idProp' />
        <small id='".$idProp . "Err' class='form-text text-danger error-
field'>Attach photo</small>
    ";
    echo $out;
}

// PROVERE

function proveriInput($el) {
    global $greske;

    if(!preg_match($el['regex'], $el['value'])) {
        $greske[] = "Valid format: " + $el['example'];
    }
}

function proveriNumber($el) {
    global $greske;

    if($el['type'] == "ddl" || $el['type'] == "number") {
        if(!isset($el['value']) || $el['value'] < 1) {
            $greske[] = $el['error'];
        }
    } else {
        if(!isset($el['value']) || $el['value'] < 10 || $el['value'] > 500)
        $greske[] = $el['error'];
    }
}

function proveriTextarea($el) {
    global $greske;

    if(!isset($el['value']) || strlen($el['value']) < 10) {
        $greske[] = $el['error'];
    }
}

```

?>

3.1.31 models/initApi.php

```
<?php

header("Content-Type: application/json");

require "../config/connection.php";
require "functions.php";

$upitCars = "SELECT * FROM vehicles";
$cars = izvrsiSelect($upitCars);

// var_dump($cars);

$upitBrands = "SELECT * FROM brands";
$brands = izvrsiSelect($upitBrands);

$upitCats = "SELECT * FROM categories";
$cats = izvrsiSelect($upitCats);

$upitFuels = "SELECT * FROM fuels";
$fuels = izvrsiSelect($upitFuels);

$upitTransmision = "SELECT * FROM transmissions";
$transmission = izvrsiSelect($upitTransmision);

$upitGlavniMeni = "SELECT * from menu WHERE priority = 1";
$glavniMeni = izvrsiSelect($upitGlavniMeni);

$upitMeniKor = "SELECT * FROM menu WHERE priority = 2";
$korMeni = izvrsiSelect($upitMeniKor);

$rez = [
    "cars" => $cars,
    "brands" => $brands,
    "cats" => $cats,
    "fuels" => $fuels,
    "transmission" => $transmission,
    "glavniMeni" => $glavniMeni,
    "korMeni" => $korMeni
];

http_response_code(200);

// var_dump($rez);
echo json_encode($rez);
```

```
?>
```

3.1.32 models/createExcel

```
<?php

require "../config/connection.php";

$output = '';

$upit = "SELECT v.name, c.categoryName, br.brandName, v.description FROM vehicles v INNER JOIN categories c on v.categoryId = c.categoryId INNER JOIN brands br ON v.brandId = br.brandId";

$stmt = $db->query($upit)->fetchAll();

if(count($stmt) > 0){

    $output .= "<table class='excelTable'>
        <thead>

            <tr>
                <td>Vehicle Name</td>
                <td>Category</td>
                <td>Description</td>
            </tr>

        </thead>

        <tbody>";

    foreach($stmt as $rez){

        $output .= "<tr>

            <td>" . $rez['brandName'] . " " . $rez['name'] . "</td>
            <td>" . $rez['categoryName'] . "</td>

            <td>" . $rez['description'] . "</td>
        </tr>";

    }

    $output .= "</tbody></table>";

}
```

```
header("Content-Type: application/xls");
header("Content-Disposition: attachment; filename=vehicles.xls");
echo $output;

?>
```

3.1.33 models/createWord.php

```
<?php
    session_start();

    $word = new COM("word.application") or die("Unable to instantiate Word");

    $word->Visible=0;

    $word->Documents->Add();

    $txt = "Aleksandar Madic";
    $txt1 = "I am currently a student at the ICT College. I was born in Bor, where I finished elementary and high school. In high school, I first encountered web design and a little bit of programming and I liked it. It made me choose this college. In my spare time, I like to play basketball, go to the theater, travel and explore new things./";

    $result = $txt."\n\n\n".$txt1."\n";

    $word->Selection->TypeText($result);

    $filename = tempnam(sys_get_temp_dir(), "word");
    $word->Documents[1]->SaveAs($filename);

    $word->Quit();
    $word = null;

    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=AleksandarMadic.doc");

    readfile($filename);
    unlink($filename);

?>
```

3.1.34 models/initLoad.php

```
<?php

// require "../config/connection.php";
require "functions.php";

$upitCars = "SELECT * FROM vehicles";
$cars = izvrsiSelect($upitCars);

// var_dump($cars);

$upitBrands = "SELECT * FROM brands";
$brands = izvrsiSelect($upitBrands);

$upitCats = "SELECT * FROM categories";
$cats = izvrsiSelect($upitCats);

$upitFuels = "SELECT * FROM fuels";
$fuels = izvrsiSelect($upitFuels);

$upitTransmision = "SELECT * FROM transmissions";
$transmission = izvrsiSelect($upitTransmision);

$upitGlavniMeni = "SELECT * from menu WHERE priority = 1";
$glavniMeni = izvrsiSelect($upitGlavniMeni);

$rez = [
    "cars" => $cars,
    "brands" => $brands,
    "cats" => $cats,
    "fuels" => $fuels,
    "transmission" => $transmission,
    "glavniMeni" => $glavniMeni
];

// var_dump($rez);

?>
```

3.1.35 models/statsLog.php

```
<?php

$open = fopen(LOG_FAJL, "r");
if($open) {
    $podaci = file(LOG_FAJL);
    fclose($open);
    $n = count($podaci);
    define("DANSEKUNDE", 24 * 60 * 60);
    $logLast24h = [];

    for($i = 0; $i < $n; $i++) {
        $red = explode("\t", $podaci[$i]);
        $t = $red[1];
        $pre24h = time() - DANSEKUNDE;
        $utT = strtotime($t);
        if($utT >= $pre24h) {
            $logLast24h[] = $red[0];
        }
    }

    $n = count($logLast24h);
    $logLast24hDistinct = [];
    for($i = 0; $i < $n; $i++) {

        if(!in_array($logLast24h[$i], $logLast24hDistinct )) {
            $logLast24hDistinct[] = $logLast24h[$i];
        }
    }

    $arrFinal = [];
    foreach($logLast24hDistinct as $k => $v) {
        $arrFinal[$v] = 0;
    }

    foreach($arrFinal as $k => $v) {
        foreach($logLast24h as $log) {
            if($k == $log) {
                $arrFinal[$k] += 1;
            }
        }
    }

    $total = 0;

    foreach($arrFinal as $ar) {
        $total += $ar;
    }
}
```

```
}
```

```
?>
```

3.1.36 views/about.php

```
<div class="site-section">
    <div class="container">
        <div class="row">
            <div class="col-lg-6 mb-5 mb-lg-0 order-lg-2">
                
            </div>
            <div class="col-lg-4 mr-auto">
                <h2>Car Company</h2>
                <p>
                    CarRent combines the strength of one of the world's largest car rental brands together with the detailed local knowledge, flexibility and great value for money you need.
                </p>
                <p>
                    For your peace of mind all our vehicles are safety checked and cleaned before every rental. And because we are constantly renewing our fleet you'll be driving some of the very latest model vehicles all at low prices.
                </p>
            </div>
        </div>
    </div>

    <div class="site-section">
        <div class="container">
            <div class="row">
                <div class="col-lg-6 mb-5 mb-lg-0">
                    
                </div>
                <div class="col-lg-4 ml-auto">
                    <h2>Our History</h2>
                    <p>
                        Established in 1990, the CarRent brand is licenced within Europe, the Middle East and Africa to Zodiac Europe Ltd (a subsidiary of Avis Budget EMEA Ltd). Budget locations are either directly owned or franchised to local operators across the region.
                    </p>
                </div>
            </div>
        </div>
    </div>
```

```
    As our name suggests we've always been synonymous with great value  
    for money. We now offer a range of fully inclusive rates, pre-  
    payment options and charge facilities designed to keep you in control and help  
    you save money.  
        </p>  
        </div>  
        </div>  
        </div>  
    </div>  
  
<div class="container site-section mb-5">  
    <div class="row justify-content-center text-center">  
        <div class="col-7 text-center mb-5">  
            <h2>How it works</h2>  
            <p>  
                With these simple steps get you to your dream ride  
            </p>  
        </div>  
    </div>  
    <div class="how-it-works d-flex">  
        <div class="step">  
            <span class="number"><span>01</span></span>  
            <span class="caption">Time & Place</span>  
        </div>  
        <div class="step">  
            <span class="number"><span>02</span></span>  
            <span class="caption">Car</span>  
        </div>  
        <div class="step">  
            <span class="number"><span>03</span></span>  
            <span class="caption">Details</span>  
        </div>  
        <div class="step">  
            <span class="number"><span>04</span></span>  
            <span class="caption">Checkout</span>  
        </div>  
        <div class="step">  
            <span class="number"><span>05</span></span>  
            <span class="caption">Done</span>  
        </div>  
    </div>  
</div>
```

3.1.37 views/adminPanel.php

```
<div class="container py-5">
    <div class="mx-auto w-50">

        <?php

        ?>

        <ul class="d-flex justify-content-center" id="adminItems">
            <li>
                <a href="#" data-pagename="cars" class="panelItem mr-4 btn btn-light"> Cars</a>
            </li>

            <li>
                <a href="#" data-pagename="brands" class="panelItem mr-4 btn btn-light"> Brands</a>
            </li>

            <li>
                <a href="#" data-pagename="stats" class="panelItem mr-4 btn btn-light"> Stats</a>
            </li>

            <li>
                <a href="#" data-pagename="bookings" class="panelItem mr-4 btn btn-light">Bookings</a>
            </li>

            <li>
                <a href="#" data-pagename="adminReview" class="panelItem mr-4 btn btn-light">Reviews</a>
            </li>

            <li>
                <a href="models/createExcel.php" class="btn btn-info">Export excel</a>
            </li>

        </ul>
    </div>
</div>

<?php
// require "models/statsLog.php";
?>
```

```
<section id="mainContent">  
    </section>
```

3.1.38 views/author.php

```
<div class="container py-5">  
    <div class="row">  
        <div class="col-md-12 mx-auto text-center my-5">  
            <h2>Aleksandar Madić 2/18</h2>  
        </div>  
        <div class="col-md-6">  
              
        </div>  
  
        <div class="col-md-6">  
            <p> I am currently a student at the ICT College. I was born in  
            Bor, where I finished elementary and high school. In high school, I first enc  
            ountered web design and a little bit of programming and I liked it. It made me  
            choose this college. In my spare time, I like to play basketball, go to the t  
            heater, travel and explore new things.</p>  
  
            <a class="btn btn-  
primary" href="models/createWord.php">Download word</a>  
        </div>  
    </div>  
</div>
```

3.1.39 views/carDetails.php

```
<?php
    require "models/cars/carDetails.php";

    require "models/functions.php";

?>

<?php if(isset($car['name'])): ?>



<div class="container">
        <div class="text-center">
            <div class="mx-auto">
                <p class="text-
uppercase categoryTitle"><?= $car['categoryName'] ?></p>
                <h2><?= $car['brandName'] . " \t " . $car['name']?></h2>
            </div>

            <div class="mx-auto mt-5 mb-5 w-75" id="car-gallery">

                <div class="car-preview">
                    <?php
                        $srcGlavna = explode(".", $car['mainImg']);
                        $srcGlavna[0] .= "-small";
                        $noviSrc = implode(".", $srcGlavna);

                        $maleSlike = [$car['mainImg'], $slike[0]['imageName'], $slike[1]['imageName']];
                    &gt;
                    <foreach($maleSlike as $mala):
                        $noviSrc = explode(".", $mala);
                        $noviSrc[0] .= "-small";
                        $noviSrc = implode(".", $noviSrc); ?>

                    " />
                    <?php endforeach ?>

                    <!--
                     -->
                    <!--  -->
                </div>
            </div>
        </div>
    </div>


```

```

        </div>

        <!-- <div class="text-center mt-5">
            <p>C H E V R L O T</p>
            <h2>Mercedes Grand Sedan</h2>
        </div> -->

        <div class="row">
        <?php

            $mainFeatures = [
                ["Mileage", "flaticon-
dashboard", $car['mileage']], ["Transmission", "flaticon-
piston", $car['transmissionType']],
                ["Seats", "flaticon-safety-seat", $car['seats']],
                ["Luggage", "flaticon-backpack", $car['luggage']],
                ["Fuel", "flaticon-diesel", $car['fuelType']]
            ];

            for($i = 0; $i < count($mainFeatures); $i++): ?>

            <div class="col-md d-flex align-self-stretch ftco-animate">
                <div class="media block-6 features">
                    <div class="media-body py-md-4">
                        <div class="d-flex mb-3 align-items-center">
                            <div class="icon d-flex align-items-
center justify-content-
center"><span class="<?= $mainFeatures[$i][1]?>"></span></div>
                            <div class="text mb-2">
                                <h3 class="heading pl-3">
                                    <?= $mainFeatures[$i][0] ?>
                                    <span class="featureValue"><?= $mainFe
atures[$i][2]?></span>
                                </h3>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>

        <?php endfor; ?>
        </div>

        <div class="row mx-auto justify-content-center mt-4" id="tabs-
container">
            <ul class="d-flex pb-3" id="list-container">
                <li class="tabs active-tab" data-
tabname="desc">Description</li>

```

```

<li class="tabs" data-tabname="features">Features</li>
<li class="tabs" data-tabname="reviews">Reviews</li>
</ul>
</div>

<div class="tab-content mt-5">

    <div class="row">
        <div class="col-md-12 tab-item" id="desc">
            <p><?= $car['description'] ?></p>
        </div>

        <div class="col-md-12 tab-item" id="features">
            <div class="row">
                <div class="col-md-4">
                    <ul class="features">
                        <!--
<li class="check"><span class="flaticon-tick"></span>Airconditions</li>
<li class="check"><span class="flaticon-cancel"></span>Child Seat</li>
<li class="check"><span class="ion-ios-checkmark"></span>GPS</li>
<li class="check"><span class="ion-ios-checkmark"></span>Luggage</li>
<li class="check"><span class="ion-ios-checkmark"></span>Music</li> -->
<?php
// var_dump($oprema);
$prvih = array_slice($oprema, 0, 4);

// foreach ($prvih as $key => $oprema1)
) {
//     $flatKlasa = "";
//     if($oprema1['value'] == 1) {
//         $flatKlasa = "flaticon-
tick";
//     } else {
//         $flatKlasa = "flaticon-
cancel";
//     }

//     echo "<li class='check'><span c
lass='$flatKlasa'></span>" . $oprema1['featureName'] . "</li>";

```

```

        // }

        ispisiliOpreme($prvih);

    ?>
    </ul>
</div>
<div class="col-md-4">
    <ul class="features">
        <?php
            $srednji = array_slice($oprema, 4, 4);

            ispisiliOpreme($srednji);
        ?>
        </ul>
</div>
<div class="col-md-4">
    <ul class="features">
        <?php
            $poslednji = array_slice($oprema, 8);

            ispisiliOpreme($poslednji);
        ?>
        </ul>
    </div>
</div>

<div class="col-md-12 tab-item" id="reviews">
    <?php

        if(count($reviews)):
            foreach($reviews as $review): ?>
                <div class="d-flex justify-content-
between">
                    <h3>User <?= $review['firstName'] . " "
" . $review['lastName'] ?></h3>
                    <p>"<?= $review['reviewText'] ?>"</p>
                </div>

                <hr />

            <?php endforeach; ?>
            <?php else: ?>
                <p> No reviews for this car.</p>
            <?php endif; ?>
    </div>

```

```

<div class="my-5">

    <?php if(isset($_SESSION['user'])): ?>

        <h2>Submit a request</h2>

        <form>

            <div class="d-flex mt-5 mb-2">
                <div class="form-group mr-5">
                    <label for="fromDate">From date</label>
                >
                    <input type="date" id="fromDate" name="fromDate" />
                </div>

                <div class="form-group mx-5">
                    <label for="toDate">To date</label>
                    <input type="date" id="toDate" name="toDate" />
                </div>
            </div>

            <div id="dateErr" class="form-text text-danger error-field mb-2">Date From must be lower than date to</div>

            <input class="btn btn-primary" type="button" name="submitRequest" id="submitRequest" value="Submit" data-userid="<?= $_SESSION['user']['userId'] ?>" data-vehicleid="<?= $car['vehicleId'] ?>" />
        </form>
        <?php else: ?>
            <h2>Log in to submit a request</h2>
        <?php endif; ?>

    </div>

    </div>
</div>

<div class="row justify-content-center text-center mb-5 section-2-title">
    <div class="col-md-6">
        <span class="sub-heading">choose car</span>
        <h2 class="mb-4">Related Cars</h2>

```

```

        </div>
    </div>

    <div class="row" id="relatedCars">

        <?php if(count($relatedCars)): ?>

            <?php foreach($relatedCars as $key => $relatedCar): ?>

                <div class="col-md-4 mb-4">
                    <div class="item-1">
                        
                        <div class="item-1-contents">
                            <div class="text-center">
                                <h3><?= $relatedCar['brandName'] . " \t " . $relatedCar['name'] ?></h3>
                                <div class="rent-price"><span>$100</span>day</div>
                            </div>
                            <ul class="specs">
                                <li>
                                    <span>Doors</span>
                                    <span class="spec"><?= $relatedCar['doors'] ?></span>
                                </li>
                                <li>
                                    <span>Seats</span>
                                    <span class="spec"><?= $relatedCar['seats'] ?></span>
                                </li>
                                <li>
                                    <span>Transmission</span>
                                    <span class="spec"><?= $relatedCar['transmissionType'] ?></span>
                                </li>
                                <li>
                                    <span>Fuel</span>
                                    <span class="spec"><?= $relatedCar['fuelType'] ?></span>
                                </li>
                            </ul>
                            <div class="d-flex action">
                                <a href="index.php?page=carDetails&carId=<?= $relatedCar['vehicleId'] ?>" class="btn btn-primary">View Details</a>
                            </div>
                        </div>
                    </div>
                </div>
            <?php endforeach(); ?>
        <?php endif(); ?>
    </div>

```

```
        </div>
    </div>

    <?php endforeach; ?>
    <?php else : ?>
        <p>There is no related cars </p>
    <?php endif; ?>

    </div>
<!-- KRAJ REDA -->

        </div>
    </div>
<?php else: ?>
    <h2>Select valid car</h2>
<?php endif; ?>
```

3.1.40 views/cars.php

```
<?php

require "models/initLoad.php";

?>

<div class="site-section bg-light">
<div class="container">
<div class="row">

    <div class="col-md-3 bg-white">
        <h3>Filter</h3>
        <form action="<?= $_SERVER['PHP_SELF'] ?>" method="GET" class="filterForm mt-4">

            <input type="text" id="searchKey" name="searchKey" class="form-control" placeholder="Search" />

            <select id="category" name="category" class="form-control">
                <option value="0">Select category</option>

                <?php
                    $kategorije = $rez['cats'];

                    foreach ($kategorije as $key => $kat): ?>
                        <option value="<?= $kat['categoryId'] ?>"><?= $kat['categoryName'] ?></option>
                <?php endforeach; ?>

            </select>

            <select name="fuel" id="fuel" class="form-control">
                <option value="0">Select Fuel</option>

                <?php
                    $fuels = $rez['fuels'];

                    foreach ($fuels as $key => $fuel): ?>
                        <option value="<?= $fuel['fuelId'] ?>"><?= $fuel['fuelType'] ?></option>
                <?php endforeach; ?>

            </select>

            <input type="hidden" id="minVal" name="minVal" value="" />

            <input type="hidden" id="maxVal" name="maxVal" value="" />

        </form>
    </div>
</div>
</div>
```

```
<div>
    <label for="amount">Price range:</label>
    <input type="text" id="amount" readonly />
</div>

<div id="slider-range"></div>

<input type="button" name="btnFilter" id="btnFilter" class="btn btn-primary" value="Submit" />
</form>
</div>

<div class="col-md-9">
    <h3>Car listing</h3>
    <div class="row mt-4" id="carsContainer"></div>

    <div id="pagination">
        <!-- <span class="p-3">1</span>
        <a href="#" class="p-3">2</a>
        <a href="#" class="p-3">3</a>
        <a href="#" class="p-3">4</a> -->

        </div>
    </div>
</div>

</div>
</div>
</div>

<?php
    require "models/cars/filterFromIndex.php";
?>
```

3.1.41 views/contact.php

```
<?php
    require "models/functions.php";
?>

<div class="site-section bg-light" id="contact-section">
    <div class="container">
        <div class="row justify-content-center text-center">
            <div class="col-7 text-center mb-5">
                <h2>Contact Us </h2>
            </div>
        </div>
        <div class="row">
            <div class="col-lg-8 mb-5" >
                <form action="phpMailer/index.php" method="post" onSubmit="return
proveriContact()">
                    <div class="form-group row">
                        <div class="col-md-6 mb-4 mb-lg-0">
                            <!-- <input type="text" class="form-
control" placeholder="First name" > -->

                            <?php
                                stampajInput("firstName", "First Name", "text");

                            ?>
                        </div>
                        <div class="col-md-6">
                            <?php
                                stampajInput("lastName", "Last Name", "text");

                            ?>

                        </div>
                    </div>
                </div>

                <div class="form-group row">
                    <div class="col-md-12">
                        <?php
                            stampajInput("email", "Email", "text");
                        ?>

                    </div>
                </div>

                <div class="form-group row">
                    <div class="col-md-12">
                        <div class='form-group'>
```

```
        <label for='msg'>Message</label>
        <textarea name="msg" id="msg" class="form-control" cols="30" rows="10"></textarea>
            <small id='msgErr' class='form-text text-danger error-field'>Valid format: </small>
        </div>

    </div>
</div>
<div class="form-group row">
    <div class="col-md-6 mr-auto">
        <input type="submit" class="btn btn-block btn-primary text-white py-3 px-5" id="btnContact" name="btnContact" value="Send Message">
    </div>
    </div>
</form>
</div>
<div class="col-lg-4 ml-auto">
    <div class="bg-white p-3 p-md-5">
        <h3 class="text-black mb-4">Contact Info</h3>
        <ul class="list-unstyled footer-link">
            <li class="d-block mb-3">
                <span class="d-block text-black">Address:</span>
                <span>777 Brockton Avenue, MA 2351

</span></li>
            <li class="d-block mb-3"><span class="d-block text-black">Phone:</span><span>+1 242 4942 290</span></li>
            <li class="d-block mb-3"><span class="d-block text-black">Email:</span><span>admin@carrent.com</span></li>
        <ul>
    </div>
    </div>
</div>
</div>
</div>
```

3.1.42 views/index.php

```
<?php

    require "models/initLoad.php";

?>

<div class="site-section pt-0 pb-0 bg-light">
    <div class="container">
        <div class="row">
            <div class="col-12">

                <form class="trip-form" action="index.php" method="GET">
                    <div class="row align-items-center mb-4">
                        <div class="col-md-6">
                            <h3 class="m-0">Begin your trip here</h3>
                        </div>
                        <div class="col-md-6 text-md-right">
                            <span class="text-
primary" id="numberCars">32</span> <span>cars in our offer</span></span>
                        </div>
                    </div>
                    <div class="row">
                        <div class="form-group col-md-4">

                            <input type="hidden" name="page" value="cars" />

                            <input type="text" id="searchKey" name="searchKey" class="form-control" placeholder="Search" />

                        </div>
                        <div class="form-group col-md-4">

                            <select id="category" name="category" class="form-control">
                                <option value="0">Select category</option>
                                <?php
                                    $kategorije = $rez['cats'];

                                    foreach ($kategorije as $key => $kat): ?>
                                        <option value="<?= $kat['categoryId'] ?>"><?= $kat['ca
toryName'] ?></option>
                                <?php endforeach; ?>
                            </select>

                            <!--
                            <select name="transmission" id="transmission" class="form-control">
                                <option value="0">Select Transmission</option>
                            </select> -->
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>
```

```
</div>
<div class="form-group col-md-4">

    <select name="fuel" id="fuel" class="form-control">
        <option value="0">Select Fuel</option>
        <?php
            $fuels = $rez['fuels'];

            foreach ($fuels as $key => $fuel): ?>
                <option value="<?= $fuel['fuelId'] ?><?= $fuel['fuelType'] ?>"><?= $fuel['fuelType'] ?></option>
            <?php endforeach; ?>
        </select>

    </div>

</div>
<div class="row">
    <div class="col-lg-6">
        <input type="submit" name="btnFilterIndex" id="btnFilterIndex" value="Submit" class="btn btn-primary">
    </div>
    </div>
</form>
</div>
</div>
</div>

<div class="site-section bg-light">
    <div class="container">
        <div class="row">
            <div class="col-lg-3">
                <h3>Our Offer</h3>
                <p class="mb-4">Create account, look for perfect car, submit request and wait for admin's approval</p>
                <p>
                    <a href="#" class="btn btn-primary custom-prev">Previous</a>
                    <span class="mx-2"/>
                    <a href="#" class="btn btn-primary custom-next">Next</a>
                </p>
            </div>
            <div class="col-lg-9">
```

```

<div class="nonloop-block-13 owl-carousel" id="carItems">

    <?php

        require "models/cars/getAll.php";

        $allCars = getAllCars();

        foreach ($allCars as $car): ?>

            <div class="item-1">
                " alt="Image <?= $car['name'] ?>" class="img-fluid">
                <div class="item-1-contents">
                    <div class="text-center">
                        <h3><?= $car['brandName'] . " " . $car['name'] ?></h3>

                        <div class="rent-price"><span>$<?= $car['pricePerDay'] ?>/</span>day</div>
                    </div>
                    <ul class="specs">
                        <li>
                            <span>Transmission</span>
                            <span class="spec"><?= $car['transmissionType'] ?></span>
                        </li>
                        <li>
                            <span>Fuel</span>
                            <span class="spec"><?= $car['fuelType'] ?></span>
                        </li>
                        <li>
                            <span>Doors</span>
                            <span class="spec"><?= $car['doors'] ?></span>
                        </li>
                        <li>
                            <span>Seats</span>
                            <span class="spec"><?= $car['seats'] ?></span>
                        </li>
                    </ul>
                    <div class="d-flex action">
                        <a href="index.php?page=carDetails&carId=<?= $car['vehicleId'] ?>" class="btn btn-primary">View more</a>
                    </div>
                </div>
            </div>

            <?php endforeach; ?>

        </div>

```

```
        </div>
    </div>
</div>
</div>

<div class="site-section section-3" style="background-image: url('assets/images/call.jpg');">
    <div class="container">
        <div class="row">
            <div class="col-12 text-center mb-5">
                <h2 class="text-white">Check our amazing offer</h2>
                <p class="text-white my-3">We have cars for every need. Everyone can find car corresponding to their budget.</p>
                <a class="btn btn-light" href="index.php?page=cars">View more</a>
            </div>
        </div>
    </div>

    </div>
</div>

<div class="container site-section mb-5">
    <div class="row red justify-content-center text-center">
        <div class="col-7 text-center mb-5">
            <h2>How it works</h2>
            <p>With these simple steps get you to your dream ride</p>
        </div>
    </div>
    <div class="how-it-works d-flex">
        <div class="step">
            <span class="number"><span>01</span></span>
            <span class="caption">Time & Place</span>
        </div>
        <div class="step">
            <span class="number"><span>02</span></span>
            <span class="caption">Car</span>
        </div>
        <div class="step">
            <span class="number"><span>03</span></span>
            <span class="caption">Details</span>
        </div>
        <div class="step">
            <span class="number"><span>04</span></span>
            <span class="caption">Request</span>
        </div>
        <div class="step">
            <span class="number"><span>05</span></span>
        </div>
    </div>
</div>
```

```

        <span class="caption">Admin aprove</span>
    </div>

    </div>
</div>
```

3.1.43 views/login.php

```

<?php
    require "models/functions.php";
?>

<div class="container-fluid py-5" id="formContainer">

    <?php

        if(isset($rez) && $rez == "") {
            foreach($greskeL as $g) {
                echo "<p>$g</p>";
            }
        }

        unset($greskeL);

        if(isset($_SESSION['greskaMail'])) {
            echo "<p>{$_SESSION['greskaMail']}</p>";
        }

        unset($_SESSION['greskaMail']);

?>

    <form action="models/account/login.php" method="POST" onSubmit="return pro
veriLogin();">

        <?php

            stampajInput("email", "Email", "email");

            stampajInput("password", "Password", "password");

?>

        <!-- <input type="email" id="email" name="email" class="form-
control" placeholder="Email address" />
```

```

        <input type="password" id="password" name="password" class="form-control" placeholder="Password"> -->

        <input type="submit" class="btn btn-primary" name="submitLogin" value="Submit" />

    </form>

    <p class="mt-5">
        Don't have account?
        <a class="text-primary" href="index.php?page=register">Sign Up here</a>
    </p>

</div>

```

3.1.44 views/logout.php

```

<?php

if(isset($_SESSION['user'])) {
    unset($_SESSION['user']);

    header("Location: index.php");
} else {
    echo "Vec nema usera";

    header("Location: index.php");
}

?>

```

3.1.45 views/profile.php

```

<div class="container py-5">
    <div class="mx-auto w-50">

        <?php

    ?>

        <ul class="d-flex justify-content-center">
            <li>
                <a href="#" data-pagename="userBook" class="panelItem mr-4 btn btn-light"> My Bookings</a>
            </li>
        </ul>
    </div>
</div>

```

```

        <li>
            <a href="#" data-
        pagename="userReview" class="panelItem mr-4 btn btn-light"> My Reviews</a>
        </li>
    </ul>
</div>
</div>

<?php
    // require "models/statsLog.php";
?>

<section id="mainContent">

</section>

```

3.1.45 views/register.php

```

<?php

require "models/functions.php";

?>

<div class="container my-5">
    <div class="row">
        <div class="offset-md-2 col-md-8">
            <?php
                if(isset($_SESSION['greskeRegister'])) {
                    foreach ($_SESSION['greskeRegister'] as $key => $gresk
a) {
                        echo "<p class='text-danger'>$greska</p>";
                    }
                }

                unset($_SESSION['greskeRegister']);
            }

            ?>

            <form action="models/account/register.php" method="POST" onSub
mit="return proveriRegister();">
                <div class="row">
                    <div class="col-md-6">
                        <?php
                            stampajInput("fname", "First name", "text");
                        ?>
                    </div>

                    <div class="col-md-6">

```

```
<?php
    stampajInput("lname", "Last name", "text");
?
</div>

</div>

<?php
    stampajInput("email", "Email", "email");

    stampajInput("password", "Password", "password");
?

<div class="row">
    <div class="col-md-6">
        <?php
            stampajInput("licenceNo", "Driver licence No",
"text");
        ?
        </div>

    <div class="col-md-6">
        <?php
            stampajInput("yearsExp", "Years of Experience"
, "number");
        ?
        </div>
    </div>

        <input type="submit" class="btn btn-primary" name="submitRegister" value="Submit" />

    </form>

    </div>
</div>
</div>
```

3.2 JS

3.2.1 main.js

```
$(document).ready(function () {
    var currentPage = location.href;
    console.log(currentPage);

    initMeni();

    if (currentPage.indexOf("page") == -1 || currentPage.indexOf("page=index") != -1) {
        initIndex();

        $(".owl-nav").hide();
    }

    if (currentPage.indexOf("cars") != -1) {

        ajaxZahtev("models/initApi.php", function (data) {

            // let cats = populateList(data.cats);
            // $("#category").append(cats);

            // let fuels = populateList(data.fuels);
            // $("#fuel").append(fuels);

            let transmissions = printChbs(data.transmission);
            $(transmissions).insertAfter("#fuel");

            $(".prenos").click(function (e) {
                if (e.ctrlKey) {
                    $(this).prop("checked", false);
                }
            })
        });

        ajaxZahtev("models/cars/filterCars.php", function (data) {
            let cars = populateCarDiv(data.cars);
            $("#carsContainer").html(cars);

            let paginacija = makePagination(data);
            $("#pagination").html(paginacija);

            $(".linkPagination").click(promeniStranuPaginacija);
        });
    }
});
```

```

$( "#btnFilter" ).click(filterCars);

$( "#slider-range" ).slider({
    range: true,
    min: 0,
    max: 200,
    values: [ 0, 200 ],
    slide: function (event, ui) {
        $( "#amount" ).val("$" + ui.values[0] + " - $" + ui.values[1]);
        $( "#minVal" ).val(ui.values[0]);
        $( "#maxVal" ).val(ui.values[1]);
    }
});
$( "#amount" ).val("$" + $( "#slider-range" ).slider("values", 0) +
" - $" + $( "#slider-range" ).slider("values", 1));

}

if (currentPage.indexOf("carDetails") != -1) {

    imageGallery();

    $(".tabs").click(changeTab);

    $(".tab-item:not(:first)").hide();

    $(".error-field").hide();

    $("#submitRequest").click(saljiRentZahtev);
}

if (currentPage.indexOf("login") != -1 || currentPage.indexOf("register") != -1 || currentPage.indexOf("contact")) {
    $(".error-field").hide();
}

if (currentPage.indexOf("profile") != -1) {
    $(".panelItem").click(promeniStranu);
}

if (currentPage.indexOf("register") != -1) {
    $("#email").blur(checkEmail);
}

// za admin panel
if (currentPage.indexOf("formaZaInsert") != -1) {
    $(".error-field").hide();
}

```

```
        $(".tabs").click(changeTab);

        $(".tab-item:not(:first)").hide();

    }

    initFooter();
});

function promeniStranu(e) {
    e.preventDefault();

    let strana = $(this).data("pagename");

    $(".panelItem").removeClass("btn-primary");
    $(this).removeClass("btn-light");
    $(this).addClass("btn-primary");

    let url = "";
    switch (strana) {
        case "cars":
            let kola = new Car();
            kola.stampajZaCars();

            break;

        case "brands":
            let brand = new Brand();
            brand.stampajZaBrand();

            break;

        case "bookings":
            let booking = new Booking();
            booking.stampajZaBooking();

            break;

        case "stats":
            let stats = new Stats();
            stats.stampajStats();
            break;

        case "userBook":
            let userBooking = new UserBook();
            userBooking.stampajZaUserBooking();
            break;
    }
}
```

```
        case "userReview":  
            let userReview = new UserReview();  
            userReview.stampajUserReview();  
            break;  
  
        case "adminReview":  
            let adminReview = new AdminReview();  
            adminReview.stampajAdminReview();  
            break;  
  
        default:  
            break;  
    }  
  
    // console.log(url);  
}
```

```
/* INICIJALNA UCITAVANJA */  
  
function ajaxZahtev(url, callback) {  
    $.ajax({  
        url: url,  
        success: callback,  
        error: function (xhr, error, status) {  
            console.log(xhr, error, status);  
        },  
    });  
}  
  
function ajaxZahtevPost(url, data, callback) {  
    $.ajax({  
        url: url,  
        type: "POST",  
        dataType: "json",  
        data: data,  
        success: callback,  
        error: function (xhr) {  
            console.log(xhr);  
  
            if (xhr.responseJSON.odg) {  
                alert(xhr.responseJSON.odg);  
            }  
  
            if (xhr.responseJSON.text) {  
                alert(xhr.responseJSON.text);  
            }  
        }  
    });  
}
```

```

        }

    });

}

function initIndex() {
    ajaxZahtev("models/initApi.php", (data) => {
        let carsOut = populateCarDiv(data.cars.slice(0, 4));

        // $("#carItems").html(carsOut);

        let numberOfCars = data.cars.length;

        console.log(numberOfCars);

        $("#numberCars").html(numberOfCars);
    });
}

function initMeni() {
    ajaxZahtev("models/initApi.php", data => {
        printGlavniMeni(data.glavniMeni);

        printKorMeni(data.korMeni);

        $("#glavniMeni li a").each(function () {
            // if (currentPage.indexOf($(this).attr("href")) != -1) {
            //     $(this).addClass("active");
            // }
            // console.log($(this).attr("href"));

            let glavnaStrana = location.href.split("=")[1];

            let strana = $(this).attr("href").split("=")[1];

            console.log(glavnaStrana, strana);

            if (glavnaStrana == strana) {
                $(this).addClass("activeLink");
            }

            if (glavnaStrana == undefined || glavnaStrana == 1) {
                $("#glavniMeni li a:first").addClass("activeLink");
            }
        });
    });
}

```

```
)  
}  
  
function printGlavniMeni(data) {  
    let out = "";  
  
    for (const el of data) {  
        out += `  
            <li class="nav-item">  
                <a class="nav-link" href="index.php?page=${el.href}">${el.text}</a>  
            </li>  
        `;  
    }  
  
    $("#glavniMeni").html(out);  
}  
  
function printKorMeni(data) {  
    let out = "";  
  
    for (const el of data) {  
        if (el.href == "logout") {  
            out += `  
                <a class="dropdown-item" href="index.php?page=${el.href}">${el.text}</a>  
            `;  
        } else {  
            out += `  
                <a class="dropdown-item" href="index.php?page=profile&userPage=${el.href}">${el.text}</a>  
            `;  
        }  
    }  
  
    $("#korMeni").html(out);  
}  
  
function initFooter() {  
    let socials = [  
        ["Facebook", "https://www.facebook.com"],  
        ["Instagram", "https://www.instagram.com"],  
        ["Youtube", "https://www.youtube.com"],  
        ["Docs", "docs.pdf"],  
        ["Author", "index.php?page=author"]  
    ];
```

```

let out = "";

for (const el of socials) {
    out += `
        <li>
            <a href="${el[1]}" target="_blank">${el[0]}</a>
        </li> <i class="slash"> / </i>
    `;
}

$("#social").html(out);
}

// function changeTab() {
//     let tabName = $(this).data("tabname");
//     $(".tabs").removeClass("active-tab");
//     $(this).addClass("active-tab");

//     $(".tab-item").hide();
//     $("#" + tabName).fadeIn();

// }
// }

function checkEmail() {
    let email = $(this).val();

    let podaciMail = {
        email: email,
        btnCheckEmail: true
    }

    ajaxZahtevPost("models/checkEmail.php", podaciMail, data => {
        if (data.odg != true) {
            let out = `<small class='form-text text-danger error-field' id='wrongEmail'>${data.odg}</small>`;

            $(out).insertAfter("#email");

            $("#email").addClass("border-danger");

        } else {
            $("#wrongEmail").remove();
            $("#email").removeClass("border-danger");
            $("#emailErr").hide();
        }
    })
}

```

```

function filterCars() {
    let keyword = $("#searchKey").val();
    let catId = $("#category").val();
    let fuelId = $("#fuel").val();
    let transId = $("input[name='transmission']:checked").val() ? $("input[name='transmission']:checked").val() : "0";

    let minValue = $("#minVal").val();
    let maxValue = $("#maxVal").val();

    let btnVal = $(this).val();

    let podaci = [];

    podaci.push(keyword, catId, fuelId, transId, minValue, maxValue, btnVal);

    $.ajax({
        url: "models/cars/filterCars.php?page=cars&searchKey=" + keyword + "&category=" + catId + "&fuel=" + fuelId + "&transmission=" + transId + "&&minVal=" + minValue + "&&maxVal=" + maxValue + "&btnFilter=true",
        success: function (data) {
            console.log(data);

            let cars = "";

            if (data.cars.length) {
                cars = populateCarDiv(data.cars);
            } else {
                cars = "<h3 class='mt-5'>We don't have any car for selected criteria</h3>";
            }

            $("#carsContainer").html(cars);

            let paginacija = makePagination(data);
            $("#pagination").html(paginacija);

            $(".linkPagination").click(promeniStranuPaginaciju);
        },
        error: function (xhr) {
            console.log(xhr);
        }
    })
}

function makePagination(data) {

```

```

let out = "";

for (let i = 1; i <= data.brStrana; i++) {
    out += `
        <a class="p-
2 linkPagination" href="index.php?page=cars&pageNo=${i}${data.queryStr}">${i}<
/a>
    `;
}

// console.log(out, data.brStrana);

return out;
}

function promeniStranuPaginacija(e) {
    e.preventDefault();

    let hrefAttr = $(this).attr("href").split("&");

    let queryStr = "";

    for (let i = 1; i < hrefAttr.length; i++) {
        // queryStr += "" + hrefAttr[i];

        if (i == 1) {
            queryStr += hrefAttr[i];
        } else {
            queryStr += "&" + hrefAttr[i];
        }
    }

    $.ajax({
        url: "models/cars/filterCars.php?" + queryStr,
        success: function (data) {
            console.log(data);

            let cars = populateCarDiv(data.cars);
            $("#carsContainer").html(cars);
        },
        error: function (xhr) {
            console.log(xhr);
        }
    })
}

```

```

function changeTab() {
    let tabName = $(this).data("tabname");
    $(".tabs").removeClass("active-tab");
    $(this).addClass("active-tab");

    $(".tab-item").hide();
    $("#" + tabName).fadeIn();

}

function imageGallery() {
    let highlight = $(".gallery-highlight");
    let previews = $(".car-preview img");

    previews.click(function () {
        let smallSrc = $(this).attr("src");
        let bigSrc = smallSrc.replace("small", "").replace("-",
        "").replace("//", "/");
        highlight.attr("src", bigSrc);
        previews.removeClass("car-active");
        $(this).addClass("car-active");
    });
}

function saljiRentZahnev() {
    let carId = $(this).data("vehicleid");
    let userId = $(this).data("userid");

    let fromDate = $("#fromDate").val();
    let toDate = $("#toDate").val();

    const current = new Date();
    const currentTmp = Date.UTC(current.getFullYear(), current.getMonth(), current.getDate());

    const date1 = new Date(fromDate);
    const date2 = new Date(toDate);

    const tmp1 = Date.UTC(date1.getFullYear(), date1.getMonth(), date1.getDate());
    const tmp2 = Date.UTC(date2.getFullYear(), date2.getMonth(), date2.getDate());

    if (tmp1 < currentTmp || tmp2 < currentTmp) {
        alert("You must choose date in future");
        return false;
    }
}

```

```

        }

    if (isNaN(tmp1) || isNaN(tmp2)) {
        alert("You must choose both dates");

        return false;
    }

    if (tmp1 > tmp2 || tmp1 == tmp2) {
        $("#dateErr").html("From date must be lower than date to");
        $("#dateErr").fadeIn();
        return false;
    } else {
        console.log("OKEJ SU DATUMI");
        $("#dateErr").fadeOut();
    }

    let podaci = {
        carId,
        userId,
        fromDate,
        toDate,
        btnRentRequest: true
    };

    ajaxZahtevPost("models/cars/handleBooking.php", podaci, data => {
        if (data.statusOdg) {
            alert(data.odg);
            $("#dateErr").fadeOut();
        } else {
            console.log(data.odg)
            $("#dateErr").html(data.odg);
            $("#dateErr").fadeIn();
        }
    });
}

/* FUNKCIJE ADMIN PANEL */

/* STAMPANJE */

function populateCarDiv(data) {
    let out = "";

```

```

for (const el of data) {
  out += `
    <div class="col-sm-12 col-md-6 mb-4">
      <div class="item-1">
        
        <div class="item-1-contents">
          <div class="text-center">
            <h3>${el.brandName} ${el.name}</h3>

            <div class="rent-
price"><span>$$ ${el.latestPrice}</span>day</div>
          </div>
          <ul class="specs">
            <li>
              <span>Doors</span>
              <span class="spec">${el.doors}</span>
            </li>
            <li>
              <span>Seats</span>
              <span class="spec">${el.seats}</span>
            </li>
            <li>
              <span>Transmission</span>
              <span class="spec">${el.transmissionType}</span>
            </li>
            <li>
              <span>Fuel</span>
              <span class="spec">${el.fuelType}</span>
            </li>
          </ul>
          <div class="d-flex action">
            <a href="index.php?page=carDetails&carId=${el.vehicleId}" clas
s="btn btn-primary">View Details</a>
          </div>
        </div>
      </div>
    `;
}

return out;
}

function populateList(data) {
  let out = "";

```

```

for (const el of data) {

    if (el.hasOwnProperty("categoryId")) {
        out += `
            <option value="${el.categoryId}">${el.categoryName}</option>
        `;
    } else if (el.hasOwnProperty("fuelType")) {
        out += `
            <option value="${el.fuelId}">${el.fuelType}</option>
        `;
    }
}

return out;
}

function printChbs(data) {
    let out = "<p>Transmission type <br /> <small>Press ctrl and click to deselect</small></p>";

    for (let i = 0; i < data.length; i++) {
        out += `
            <div class="custom-control custom-radio">
                <input type="radio" class="custom-control-
input prenos" id="customRadio${i}" name="transmission" value="${data[i].transm
issionId}" />
                <label class="custom-control-
label" for="customRadio${i}">${data[i].transmissionType}</label>
            </div>
        `;
    }

    return out;
}

/* PROVERE FORME */

var podaci = [];
var greske = [];

var regexName = /^[A-Z][a-z]{2,20}(\s[A-Z][a-z]{3,20}){0,2}$/;
var regexEmail = /^[A-z\d\.-]{5,100}@[a-z]{2,10}\.[a-z]{2,20}$/;
var regexLicenceNo = /^[A-Z][\d]{3}-[\d]{3}-[\d]{2}-[\d]{3}-[\d]$/;

var objValid = {
    fname: "John",
    lname: "Conally",
}

```

```

email: "johnconnaly@gmail.com",
licenceNo: "F255-921-50-094-0",
};

function proveriPolje(regex, polje) {
    if (regex.test(polje.val())) {
        podaci.push(polje.val());
        polje.removeClass("border-danger");
    } else {
        greske.push(polje.attr("id") + " nije dobro");
        polje.val("");
        polje.addClass("border border-danger");
        let pId = polje.attr("id");
        polje.attr("placeholder", objValid[pId]);
        return false;
    }
}

function proveriRegister() {
    podaci = [];
    greske = [];

    let fname = $("#fname");
    let lname = $("#lname");
    let email = $("#email");
    let password = $("#password");
    let licenceNo = $("#licenceNo");
    let yearsExp = $("#yearsExp");

    // proveriPolje(regexName, fname);
    // proveriPolje(regexName, lname);
    // proveriPolje(regexEmail, email);
    // proveriPolje(regexLicenceNo, licenceNo);

    // if (password.val().Length < 5) {
    //     password.attr("placeholder", "Password must have 5 chars min");
    //     password.addClass("border border-danger");
    //     greske.push("Password must have 5 chars min");
    // } else {
    //     password.removeClass("border-danger");
    // }

    // if (yearsExp.val() < 0 || yearsExp.val() == "") {
    //     yearsExp.attr("placeholder", "Years must be positive number");
    //     yearsExp.val("");
    //     yearsExp.addClass("border border-danger");
    //     greske.push("Years of experience must be positive number");
    // } else {
}

```

```

//    yearsExp.removeClass("border-danger");
// }

let polja = [
{
  selector: "#fname",
  regex: /^[A-Z][a-z]{2,20}(\s[A-Z][a-z]{3,20}){0,2}$/,
  type: "input",
  example: "John"
},
{
  selector: "#lname",
  regex: /^[A-Z][a-z]{2,20}(\s[A-Z][a-z]{3,20}){0,2}$/,
  type: "input",
  example: "Connaly"
},
{
  selector: "#email",
  regex: /^[A-z\d\.-]{5,100}@[a-z]{2,10}\.[a-z]{2,20}$/,
  type: "input",
  example: "johnconnaly@gmail.com"
},
{
  selector: "#password",
  regex: /^[A-z\d\.-]{5,100}@[a-z]{2,10}\.[a-z]{2,20}$/,
  type: "password",
  example: "johnconnaly@gmail.com"
},
{
  selector: "#licenceNo",
  regex: /^[A-Z][\d]{3}-[\d]{3}-[\d]{2}-[\d]{3}-[\d]$/,
  type: "input",
  example: "F255-921-50-094-0"
},
{
  selector: "#yearsExp",
  type: "number",
  example: "Enter positive number"
},
]

for (const polje of polja) {
  if (polje.type == "input") {
    proveriInputPolje(polje);
  } else if (polje.type == "number") {
    proveriNumberPolje(polje);
  } else if (polje.type == "password") {

```

```

        proveriInputPass(polje);
    }
}

console.log(greske);

if (greske.length == 0 && !$("#wrongEmail").length) {
    return true;
} else {
    return false;
}
}

function proveriLogin() {
    podaci = [];
    greske = [];

    let email = $("#email");
    let password = $("#password");

    // proveriPolje(regexEmail, email);

    // if (password.val().Length < 5) {
    //     password.attr("placeholder", "Password must have 5 chars min");
    //     password.addClass("border border-danger");
    //     greske.push("Password must have 5 chars min");
    // } else {
    //     password.removeClass("border-danger");
    // }

    let polja = [
        {
            selector: "#email",
            regex: /^[A-z\d\.-]{5,100}@[a-z]{2,10}\.[a-z]{2,20}$/,
            type: "text",
            example: "somebody12@gmail.com"
        },
        {
            selector: "#password",
            type: "password"
        }
    ];

    for (const el of polja) {
        if (el.type == "text") {
            proveriInputPolje(el);
        } else if (el.type == "password") {
            proveriInputPass(el);
        }
    }
}

```

```
        }

    }

    console.log(greske);

    if (greske.length == 0) {
        return true;
    } else {
        return false;
    }
}

var podaciCar = new FormData();

var poljaCar = [
{
    selector: "#carName",
    regex: /^[A-z\s\d\.\.]{2,30}$/,
    type: "input",
    example: "Sportage"
},
{
    selector: "#brandName",
    regex: "",
    type: "ddl"
},
{
    selector: "#categoryName",
    regex: "",
    type: "ddl"
},
{
    selector: "#desc",
    regex: "",
    type: "textarea"
},
{
    selector: "#modelYear",
    regex: /^(19|20)[0-9]{2}$/,
    type: "input",
    example: 2000
},
{
    selector: "#seats",
    regex: "",
    type: "number"
},
{
```

```
        selector: "#doors",
        regex: "",
        type: "number"
    },
    {
        selector: "#fuelType",
        regex: "",
        type: "ddl"
    },
    {
        selector: "input[name=transmissiontype]",
        regex: "",
        type: "radio",
        errSelector: "#transmissiontypeErr",
        name: "transmissiontype"
    },
    {
        selector: "#mileage",
        regex: "",
        type: "number"
    },
    {
        selector: "#luggage",
        regex: "",
        type: "number"
    },
    {
        selector: "#coverPhoto",
        regex: "",
        type: "file"
    },
    {
        selector: "#otherPhoto",
        regex: "",
        type: "file"
    },
    {
        selector: "#otherPhoto2",
        regex: "",
        type: "file"
    },
    {
        selector: "#price",
        regex: "",
        type: "number"
    }
];

```

```

function proveriFormuCar() {
    greske = [];

    for (const el of poljaCar) {
        if (el.type == "input") {
            proveriInputPolje(el, podaciCar);
            // Let propName = el.selector.substr(1);

        } else if (el.type == "number") {
            proveriNumberPolje(el, podaciCar);
        } else if (el.type == "ddl") {
            proveriDdl(el, podaciCar);
        } else if (el.type == "radio") {
            let radio = $(el.selector + ":checked");

            if (radio.length == 0 || radio.val() == "") {
                greske.push("Select one option");
                $(el.errSelector).fadeIn();
                return false;
            } else {
                $(el.errSelector).hide();
            }

            podaciCar.append(el.name, radio.val());
        }

        } else if (el.type == "textarea") {
            let area = $(el.selector);

            if (area.val().length < 10 || !area.val().trim().length) {
                greske.push("Desc must have at least 10 chars");
                area.addClass("border border-danger");

                $(el.selector + "Err").fadeIn();

                return false;
            } else {
                $(el.selector + "Err").hide();
                area.removeClass("border-danger");

                let propName = el.selector.substr(1);
                podaciCar.append(propName, area.val());
            }
        } else if (el.type == "file") {
            if (!$(el.selector).val()) {
                greske.push("Upload photo err");
            }
        }
    }
}

```

```
$(el.selector + "Err").fadeIn();
return false;
} else {
$(el.selector + "Err").hide();

let slika = document.querySelector(el.selector).files[0];

let propName = el.selector.substr(1);

podaciCar.append(propName, slika);
}

}

let oprema = $(".oprema:checked");

if (oprema.length) {
let opremaNiz = Array.from(oprema);

for (const el of opremaNiz) {
let elId = $(el).attr("id");
let elVal = $(el).val();

podaciCar.append(elId, elVal);
}
}

console.log(greske);

if (greske.length) {
return greske;
} else {

podaciCar.append("btnInsertCar", "true");
return podaciCar;
}

}

var podaciCarUpdate = new FormData();

function proveriFormuCarUpdate() {

podaciCarUpdate.append("carId", $("#carId").val());

for (const el of poljaCar) {
```

```

if (el.type == "input") {
    proveriInputPolje(el, podaciCarUpdate);
    // Let propName = el.selector.substr(1);

} else if (el.type == "number") {
    proveriNumberPolje(el, podaciCarUpdate);
} else if (el.type == "ddl") {
    proveriDdl(el, podaciCarUpdate);
} else if (el.type == "radio") {
    let radio = $(el.selector + ":checked");

    if (radio.length == 0 || radio.val() == "") {
        greske.push("Select one option");
        $(el.errSelector).fadeIn();
        return false;
    } else {
        $(el.errSelector).hide();

        podaciCarUpdate.append(el.name, radio.val());
    }
}

} else if (el.type == "textarea") {
    let area = $(el.selector);

    if (area.val().length < 10) {
        greske.push("Desc must have at least 10 chars");
        area.addClass("border border-danger");

        $(el.selector + "Err").fadeIn();

        return false;
    } else {
        $(el.selector + "Err").hide();
        area.removeClass("border-danger");

        let propName = el.selector.substr(1);
        podaciCarUpdate.append(propName, area.val());
    }
}

} else if (el.type == "file") {
    if ($(el.selector).val()) {
        $(el.selector + "Err").hide();

        let slika = document.querySelector(el.selector).files[0];

        let propName = el.selector.substr(1);

```

```
podaciCarUpdate.append(propName, slika);

}

}

let oprema = $(".oprema:checked");

if (oprema.length) {
    let opremaNiz = Array.from(oprema);

    for (const el of opremaNiz) {
        let elId = $(el).attr("id");
        let elVal = $(el).val();

        podaciCarUpdate.append(elId, elVal);
    }
}

console.log(greske);

if (greske.length) {
    return greske;
} else {

    podaciCarUpdate.append("btnUpdateCar", "true");
    return podaciCarUpdate;
}
}

function proveriContact() {
    greske = [];

    let polja = [
        {
            selector: "#firstName",
            regex: /^[A-Z][a-z]{2,20}(\s[A-Z][a-z]{3,20}){0,2}$/,
            type: "input",
            example: "John"
        },
        {
            selector: "#lastName",
            regex: /^[A-Z][a-z]{2,20}(\s[A-Z][a-z]{3,20}){0,2}$/,
            type: "input",
            example: "Smith"
        },
    ];
}
```

```

{
  selector: "#email",
  regex: /^[A-z\d\.-]{5,100}@[a-z]{2,10}\.[a-z]{2,20}$/,
  type: "input",
  example: "johnsmith@gmail.com"
},
{
  selector: "#msg",
  type: "textarea",
  example: "Your message must have at least 10 chars"
}
];

for (const el of polje) {
  if (el.type == "input") {
    proveriInputPolje(el);
  } else if (el.type == "textarea") {
    let area = $(el.selector);

    if (area.val().length < 10) {
      greske.push("Message must have at least 10 chars");
      area.addClass("border border-danger");

      $(el.selector + "Err").html(el.example);
      $(el.selector + "Err").fadeIn();

      return false;
    } else {
      $(el.selector + "Err").hide();
      area.removeClass("border-danger");
    }
  }
}

console.log(greske);

if (greske.length) {
  return false;
} else {
  return true;
}

}

function proveriInputPolje(el, podaci = new FormData()) {
let polje = $(el.selector);
let regex = el.regex;
let poljeErr = $(el.selector + "Err");

```

```

let propName = el.selector.substr(1);

if (regex.test(polje.val())) {
    polje.removeClass("border-danger");
    poljeErr.hide();
    podaci.append(propName, polje.val());
} else {
    greske.push(el.selector + " not in right format");
    polje.val("");
    polje.addClass("border border-danger");

    poljeErr.html("Valid format: " + el.example);
    poljeErr.fadeIn();

    return false;
}
}

function proveriNumberPolje(el, podaci = new FormData()) {
    let polje = $(el.selector);
    let poljeErr = $(el.selector + "Err");
    let propName = el.selector.substr(1);

    if (polje.val() < 0 || polje.val() == "") {
        polje.addClass("border border-danger");
        greske.push(el.selector + " enter positive number");

        poljeErr.html("Valid format: enter positive number");
        poljeErr.fadeIn();

        return false;
    } else {
        polje.removeClass("border-danger");
        poljeErr.hide();

        podaci.append(propName, polje.val());
    }
}

function proveriDdl(el, podaci = new FormData()) {
    let polje = $(el.selector);
    let propName = el.selector.substr(1);

    if (polje.val() == 0) {
        polje.addClass("border border-danger");
        greske.push(el.selector + " select one element");
        return false;
    } else {

```

```

polje.removeClass("border-danger");

    podaci.append(propName, polje.val());
}
}

function proveriInputPass(el, podaci = new FormData()) {
    let polje = $(el.selector);
    let propName = el.selector.substr(1);

    console.log(polje.val());

    if (polje.val() == "" || polje.val().length < 5) {
        let greskaPolje = $(el.selector + "Err");
        greskaPolje.fadeIn();
        greskaPolje.html("Password must have 5 chars min");

        greske.push("Password has less than 5 chars");
        polje.addClass("border border-danger");
        return false;
    } else {
        polje.removeClass("border-danger");

        podaci.append(propName, polje.val());
    }
}

// KLASE
class UserBook {
    stampajZaUserBooking() {
        let url = "views/user/booking/selectAll.php";

        ajaxZahtev(url, data => {
            $("#mainContent").html(data);

            $(".cancelBookingUser").click((e) => {
                e.preventDefault();

                let bookId = e.target.dataset.bookingid;

                this.saljiCancelBooking(bookId);
            })
        });
    }

    saljiCancelBooking(bookId) {
        let status = $("#status").data("statusid");
    }
}

```

```

if (status == 1) {
    let url = "models/booking/cancelUserBook.php";

    let podaciCancel = {
        bookId: bookId,
        btnUserCancel: true
    };

    $.ajax({
        url: url,
        type: "POST",
        dataType: "json",
        data: podaciCancel,
        success: data => {
            if (data.odg) {
                alert("You successfully canceled");
                this.stampajZaUserBooking();
            } else {
                alert("You can't cancel");
            }
        },
        error: function (xhr) {
            console.log(xhr);
        }
    })
} else {
    alert("Admin didn't aproved your request, you can't cancel yet");
}

}

class UserReview {
    stampajUserReview() {
        let url = "views/user/reviews/selectAll.php";

        ajaxZahtev(url, data => {
            $("#mainContent").html(data);

            $(".editReview").click(e => {
                e.preventDefault();

                let reviewId = e.target.parentElement.dataset.reviewid;

                this.handleEditReview(reviewId);
            });
        });
    }
}

```

```
$(".deleteReview").click(e => {
  e.preventDefault();

  let reviewId = e.target.parentElement.dataset.reviewid;

  this.handleDeleteReview(reviewId);
});

$(".insertReview").click(() => {
  this.stampajInsertReview();
})

});

handleEditReview(reviewId) {
  let reviewTxt = $("#review" + reviewId);

  // Let regexReview = /^[A-z](A-z\d\..)*$/;

  if (reviewTxt.val().trim().length) {
    let url = "models/reviews/userUpdate.php";

    let podaciReview = {
      reviewId: reviewId,
      reviewTxt: reviewTxt.val(),
      editReviewBtn: true
    };

    ajaxZahtevPost(url, podaciReview, data => {
      if (data.odg) {
        alert("Updated successfully");

        this.stampajUserReview();
      } else {
        alert(data.odg);
      }
    });
  } else {
    alert("Enter valid text");
  }
}

handleDeleteReview(reviewId) {

  let url = "models/reviews/userDelete.php";
```

```
let podaci = {
    reviewId: reviewId,
    deleteReviewBtn: true
};

ajaxZahtevPost(url, podaci, data => {
    if (data.odg) {
        alert("Successfully deleted");

        this.stampajUserReview();

    } else {
        alert(data.odg);
    }
});

}

stampajInsertReview() {
    let url = "views/user/reviews/insertForm.php";

    ajaxZahtev(url, data => {
        $("#mainContent").html(data);

        $(".error-field").hide();

        $("#btnInsertReview").click(() => this.proveriInsertReview());
    });
}

proveriInsertReview() {
    greske = [];

    let podaciReview = {};

    let polja = [
        {
            selector: "#vehicleName",
            type: "ddl"
        },
        {
            selector: "#reviewTxt",
            regex: "",
            type: "textarea"
        }
    ];
}
```

```

for (const el of polja) {
    let propName = el.selector.substr(1);

    if (el.type == "ddl") {
        if ($(el.selector).val() != 0) {
            $(el.selector + "Err").hide();
            $(el.selector).removeClass("border-danger");
            podaciReview[propName] = $(el.selector).val();

        } else {
            greske.push(el.selector + " you must choose");

            $(el.selector).addClass("border border-danger");

            return false;
        }
    } else {
        let area = $(el.selector);

        if (area.val().length < 10 || !area.val().trim().length) {
            greske.push("Review must have at least 10 chars");
            area.addClass("border border-danger");

            $(el.selector + "Err").fadeIn();

            return false;
        } else {
            $(el.selector + "Err").hide();
            area.removeClass("border-danger");

            podaciReview[propName] = area.val();
        }
    }
}

if (!$("#vehicleName").length) {
    console.log("Ne postoji ddl na strani");
    return false;
}

podaciReview.btnInsertReview = true;

if (greske.length) {
    return false;
}

let url = "models/reviews/userInsert.php";
ajaxZahtevPost(url, podaciReview, data => {

```

```

        if (data.odg) {
            alert("Successfully inserted review");

            this.stampajUserReview();
        } else {
            alert(data.odg);
        }
    });

}

}

```

3.2.2 admin.js

```

window.onload = function () {

    let currentPage = location.href;

    $(".panelItem").click(promeniStranu);

}

// function ajaxZahtev(url, callback) {
//     $.ajax({
//         url: url,
//         success: callback,
//         error: function (xhr) {
//             console.log(xhr);
//         }
//     })
// }

class Car {

    stampajZaCars() {
        let url = "views/admin/cars/selectAll.php";
        ajaxZahtev(url, data => {
            // console.log(data);
            // console.log(this);

            $("#mainContent").html(data);

            $(".editCar").click((e) => {
                e.preventDefault();

                let carId = e.target.parentElement.dataset.carid;

```

```

        this.stampajUpdateCar(carId);
    });

    $(".deleteCar").click((e) => {
        e.preventDefault();

        let carId = e.target.parentElement.dataset.carid;

        console.log(carId);

        this.saljiZahtevZaDelete(carId);
    });

    $(".insertCar").click(() => this.stampajInsertCar());
}
}

stampajInsertCar() {
    let url = "views/admin/cars/insertForm.php";
    ajaxZahtev(url, data => {
        $("#mainContent").html(data);

        $(".error-field").hide();

        // $(".tabs").click(changeTab);

        // $(".tab-item:not(:first)").hide();

        $("#btnInsertCar").click(() => this.saljiFormuZaInsert());

        // $("#btnInsertCar").click(this.saljiFormuZaInsert);
    });
}

saljiFormuZaInsert() {
    let podaciCar = proveriFormuCar();

    $.ajax({
        url: "models/cars/insert.php",
        method: "POST",
        data: podaciCar,
        contentType: false,
        processData: false,
        success: (odgovor) => {
            if (odgovor.odg) {
                alert("Successfully inserted!");
                this.stampajZaCars();
            }
        }
    });
}

```

```

        console.log(this);

    } else {
        console.log(odgovor.odg);
    }
},
error: function (xhr) {
    console.log(xhr);
}
});
}

saljiZahtevZaDelete(carId) {
let podaci = {
    carId: carId,
    btnDeleteCar: true
}

ajaxZahtevPost("models/cars/delete.php", podaci, data => {
if (data.odg) {
    alert("Successfully deleted");
    this.stampajZaCars();
}
});
}

stampajUpdateCar(carId) {
let url = "views/admin/cars/updateForm.php?carId=" + carId;

ajaxZahtev(url, data => {
$("#mainContent").html(data);

$(".error-field").hide();

$("#btnUpdateCar").click(() => this.saljiZahtevZaUpdate());
})
}

saljiZahtevZaUpdate() {
let podaciCar = proveriFormuCarUpdate();

console.log(podaciCar);

//provera slike koci

$.ajax({
    url: "models/cars/update.php",
    method: "POST",

```

```

        data: podaciCar,
        contentType: false,
        processData: false,
        success: (data) => {
            console.log(data);
            if (data.odg) {
                alert("Updated successfully");
                this.stampajZaCars();
            } else {
                alert("Updated error, try again");
            }
        },
        error: function (xhr) {
            console.log(xhr);
        }
    );
}

}

class Brand {

stampajZaBrand() {
    let url = "views/admin/brands/selectAll.php";

    ajaxZahtev(url, data => {
        $("#mainContent").html(data);

        $(".editBrand").click((e) => {
            e.preventDefault();
            let brandId = e.target.parentElement.dataset.brandid;

            console.log(brandId, "Parent element je " + e.target.parentElement);

            this.prikaziFormuZaEdit(brandId);
        });

        $(".deleteBrand").click((e) => {
            e.preventDefault();

            let brandId = e.target.parentElement.dataset.brandid;

            console.log(brandId);

            this.saljiZahtevZaDelete(brandId);
        });
    });
}
}

```

```
        });

        $(".insertBrand").click(() => this.prikaziFormuZaInsert());

    });

prikaziFormuZaEdit(brandId) {
    // let brandId = $(this).data("brandId");
    let url = "views/admin/brands/insertForm.php?brandId=" + brandId;

    ajaxZahtev(url, data => {
        $("#mainContent").html(data);

        $(".error-field").hide();

        // let brandForm = document.querySelector("#brandForm");
        // brandForm.addEventListener("submit", this.proveriBrandFormu);

        $("#btnInsert").click(() => this.proveriBrandFormu());
    });
}

prikaziFormuZaInsert() {

    let url = "views/admin/brands/insertForm.php";

    ajaxZahtev(url, data => {
        $("#mainContent").html(data);

        $(".error-field").hide();

        console.log(this);

        // let brandForm = document.querySelector("#brandForm");
        // brandForm.addEventListener("submit", this.proveriBrandFormu);

        $("#btnInsert").click(() => this.proveriBrandFormu());
    });
}

proveriBrandFormu(e) {
    console.log("jel se poziva");
    let greske = [];

    let polje = {
```

```
        selector: "#brandName",
        regex: /^[A-Z][a-z\d\s\.\.]{3,40}$/,
        type: "input",
        example: "Mercedes"
    }

proveriInputPolje(polje);

let podaci = {
    brandName: $("#brandName").val(),
    btnInsert: true
}

let url = "models/brands/insert.php";

if ($("#brandId").length) {
    podaci.brandId = $("#brandId").val();
    url = "models/brands/update.php";
}

console.log(greske, url);

if (greske.length) {
    e.preventDefault();
}

ajaxZahtevPost(url, podaci, data => {
    console.log(data);

    if (data.status) {
        alert(data.text);
        this.stampajZaBrand();

    } else {
        alert(data.status);
    }
});

}

saljiZahtevZaDelete(brandId) {
    let url = "models/brands/delete.php";
    // ajaxZahtev(url, data => {
    //     console.log(data);
    // })
}

let podaci = {
    brandId: brandId,
```

```
        btnInsert: true
    }

    ajaxZahtevPost(url, podaci, data => {
        console.log(data);

        if (data.status) {
            alert(data.text);

            this.stampajZaBrand();
        }
    )
}

}

class Booking {
    stampajZaBooking() {
        let url = "views/admin/booking/selectAll.php";

        ajaxZahtev(url, data => {
            $("#mainContent").html(data);

            $(".confirmBooking").click((e) => {
                e.preventDefault();

                let bookId = e.target.dataset.bookingid;

                this.saljiPotvrduBooking(bookId);
            });

            $(".cancelBooking").click((e) => {
                e.preventDefault();

                let bookId = e.target.dataset.bookingid;

                this.saljiCancelBooking(bookId);
            })
        });
    }

    saljiPotvrduBooking(bookId) {
        let url = "models/booking/handleBook.php";
    }
}
```

```
let podaci = {
    bookId: bookId,
    btnHandleBook: "true"
};

ajaxZahtevPost(url, podaci, data => {
    alert(data.odgovorTxt);

    this.stampajZaBooking();
})
}

saljiCancelBooking(bookId) {
    let url = "models/booking/handleBookCancel.php";

    let podaci = {
        bookId: bookId,
        btnHandleBook: "false"
    };

    ajaxZahtevPost(url, podaci, data => {
        alert(data.odgovorTxt);

        this.stampajZaBooking();
    })
}
}

class Stats {
    stampajStats() {

        console.log("jel se radi ovde nesto");

        let url = "views/admin/statsLog.php";

        ajaxZahtev(url, data => {
            $("#mainContent").html(data);

        });
    }
}

class AdminReview {
    stampajAdminReview() {
        let url = "views/admin/reviews/selectAll.php";

        ajaxZahtev(url, data => {
```

```
$("#mainContent").html(data);

$(".changeStatus").click((e) => {
    e.preventDefault();

    let reviewId = e.target.dataset.reviewid;

    let reviewStatus = e.target.dataset.reviewstatus;

    this.changeReviewStatus(reviewId, reviewStatus);
});

});

changeReviewStatus(reviewId, reviewStatus) {
    let url = "models/reviews/changeState.php";

    let podaci = {
        reviewId: reviewId,
        reviewStatus: reviewStatus,
        btnChangeState: true
    };

    ajaxZahtevPost(url, podaci, data => {
        if (data.odg) {
            alert("State successfully changed");
            this.stampajAdminReview();
        }
    });
}

}
```