Faculty of Technology

Department of Computer Engineering

Dharmsinh Desai University

B.Tech CE Semester-V
Subject: (CE-515) Advanced Technologies

A

PROJECT REPORT ON

WheelUp

a Car Rental Web Application

By

HARSHIT CHUDASAMA (CE026) (18CEUBG007)

Guided by:

Prof. Prashant M. Jadhav
Associate Professor Dept. of
Comp. Engg.DDU

Faculty of Technology

Department of Computer Engineering

Dharmsinh Desai University

CERTIFICATE

This is to certify that the practical / term work

carried out in the subject of

Advanced Technologies and recorded in this journal

is the bonafide work of

HARSHIT CHUDASAMA (CE026) (18CEUBG007)

of B.Tech semester **V** in the branch of **Computer Engineering**

during the academic year **2020-2021**.

Prof. Prashant M. Jadav

Assistant Professor of Department of Computer Engineering, Dharmsinh Desai University, Nadiad

Dr. C. K. Bhensdadia

Head of Department of Computer Engineering, Dharmsinh Desai University, Nadiad

Table of Contents

1. Abstract

The **WheelUp a Car Rental Application** is a web application for to take a car on rental for temporary use. The main objective is to book a car without going to office it's save a lots of time of user. User can choose a car as per their choices. They can check all travel history in this application.

2. Introduction

Brief Introduction

The Car Rental is an application that rents automobiles for a short period of time for a fee whether in a few days or weeks. In this application users can login through email and password. They can see all available cars and select their desired car and book the trip. They can contact us through email or social media platform like Instagram, Facebook. The purpose of this application is to transform the manual process of hiring a car to a computerized system.

Tools/Technologies Used

Technologies :

- Cascading Style Sheet (CSS 3) for styling the HTML pages.
- JavaScript for providing dynamic content for the HTML pages.
- JQuery to provide dynamic data to the request pages.
- Bootstrap4 for styling the HTML pages using predefined classes of bootstrap.
- Angular 6 for component division of our website.
- Node.js for backend purposes.
- Mongo DB as a database manager to save and fetch data from the database using Node.js mongoose module.
- Typescript used in Angular 6 for accessing all the components of the same.

Tools :

- Visual Studio Code
- Umlet
- MongoDb Community

3. Software Requirement Specifications

1 Customer Functionality

R.1 Create an account

⇨ Customer need to create an account to login the system. The customer can create an account using the email address.

⇨ **Input:**

- E-mail address
- Username
- Password
- Basic details

⇨ **Output:**

An email will be sent out to customer requesting to confirm the account.

R.2 Log in to system

⇨ The customer can log into the system using username/email and password

⇨ **Input:**

- Username/email
- Password

⇨ **Output:**

- The customer view of the system will be visible to user

R.3 Select Car Category

- ⇨ The customer will be able to choose vehicle by category
- ⇨ **Input:**
  - Vehicle category
- ⇨ **Output:**
  - All possible result will be shown to the customer.

R.4 View vehicle with details

- ⇨ The customer can view the detail of the vehicle selected
- ⇨ **Input:**
  - Selection of the vehicle
- ⇨ **Output:**
  - Details of the vehicle:
    - Name and model
    - Picture
    - Cost
    - Availability

R.5 Request a email notification when vehicle is available

- ⇨ In case of vehicle is unavailable, the customer can request a Notification when it is available.
- ⇨ **Input:**
  - Selection of the notification feature.
- ⇨ **Output**
  - Email notification request confirm email.

R.6 Reserve vehicle for renting

⇨ Customer can reserve the vehicle after considering all the information.

⇨ **Input:**

- Renting date
- Renting duration

⇨ **Output:**

- An email will be sent to the customer confirming the reservation.

R.7 Update reservation

⇨ Customer can cancel the reservation or update details of the reservation.

⇨ **Input:**

- Cancel request
- Update details

⇨ **Output:**

- Reservation cancellation/ updated notification.

2. System Admin Functionality

R.1 Log in with admin account

⇨ Admin of the system can log into the system using existing account.

⇨ **Input:**

- Admin username and password

⇨ **Output:**

- Admin view of the system

R.2 Add new vehicle

⇨ Admin can add new vehicle to the system

⇨ **Input:**

- Vehicle category details.
- Vehicle details

⇨ **Output:**

- Vehicle successfully added notification.

R.3 Update vehicle details

⇨ Details of existing vehicle can be changed by the admin.

⇨ **Input:**

- Changed details.

⇨ **Output:**

- Updating details successful notification

R.4 Change vehicle status

- ⇨ Admin can change the vehicle status to available or not available.
- ⇨ **Input:**
  - Change status request.
- ⇨ **Output:**
  - Change vehicle status successfully notification.

R.5 Remove vehicle

- ⇨ Admin can remove a vehicle or remove a whole category from the system.
- ⇨ **Input:**
  - Vehicle / category remove request
- ⇨ **Output:**
  - Vehicle / category removed notification

R.6 Change cost details

- ⇨ Admin can change the current rates of the system according to business needs of the company
- ⇨ **Input:**
  - New rates
- ⇨ **Output:**
  - Rate changed notification

R.7 View Report

- ⇨ Admin can see all trip details.
- ⇨ **Input:**
  - Select view report
- ⇨ **Output:**
  - All trips information.

4. Design

1. DTD

1.1 User

```
<?xml version="1.0"?>
<!DOCTYPE user [
<!ELEMENT user
(id,email_id,username,password,phoneno,gender,pic,role)>
<!ELEMENT id (#PCDATA)>
<!ELEMENT email_id (#PCDATA)>
<!ELEMENT username (#PCDATA)>
<!ELEMENT password (#PCDATA)>
<!ELEMENT phoneno (#PCDATA)>
<!ELEMENT gender (#PCDATA)>
<!ELEMENT pic (#PCDATA)>
<!ELEMENT role (#PCDATA)>
]>
<user>
<id>ObjectId(---)</id>
<email_id>XYZ@abc.com</email_id>
<username>XYZ</username>
<password>****</password>
<phoneno>1234567890</phoneno>
<gender>Male</gender>
<pic>123456.jpg</pic>
<role>user</role>
</user>
```

1.2 Car

```
<?xml version="1.0"?>
<!DOCTYPE car [
<!ELEMENT user
(id,name,price,transmission,category,available,img1,img2)>
<!ELEMENT id (#PCDATA)>
<!ELEMENT name (#PCDATA)>
```

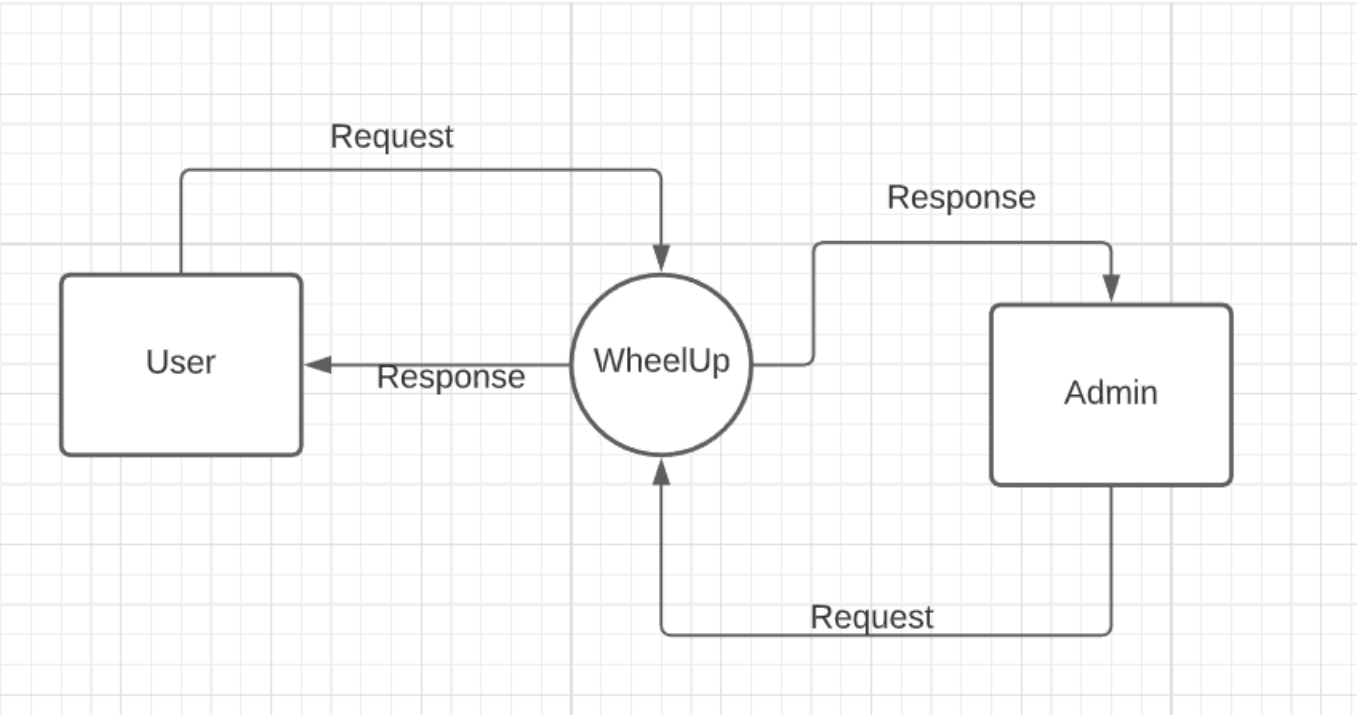
```
<!ELEMENT price (#PCDATA)>
<!ELEMENT transmission (#PCDATA)>
<!ELEMENT category (#PCDATA)>
<!ELEMENT available (#PCDATA)>
<!ELEMENT img1 (#PCDATA)>
<!ELEMENT img2 (#PCDATA)>
]>
<car>
<id>ObjectId(---)</id>
<name>Ford Figo</name>
<price>2500 </price>
<transmission>Manual</transmission>
<category>Economy Car</category>
<available>true</available>
<img1>123456.jpg</img1>
<img2>123458.jpg</img2>
</car>
```

1.3 Trip

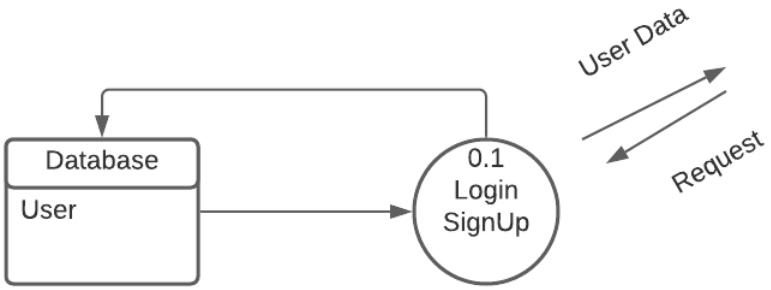
```
<?xml version="1.0"?>
<!DOCTYPE trip [
<!ELEMENT trip (id,startDate,endDate,amount,car_id,user_id)>
<!ELEMENT id (#PCDATA)>
<!ELEMENT startDate (#PCDATA)>
<!ELEMENT endDate (#PCDATA)>
<!ELEMENT amount (#PCDATA)>
<!ELEMENT car_id (#PCDATA)>
<!ELEMENT user_id (#PCDATA)>
]>
<trip>
<id>ObjectId(---)</id>
<startDate>2020-10-22T00:00:00.000+00:00 </startDate>
<endDate>2020-10-22T00:00:00.000+00:00 </endDate>
<amount>2500 </amount>
<car_id>123456789…</car_id>
<user_id>658463…</user_id>
</trip>
```

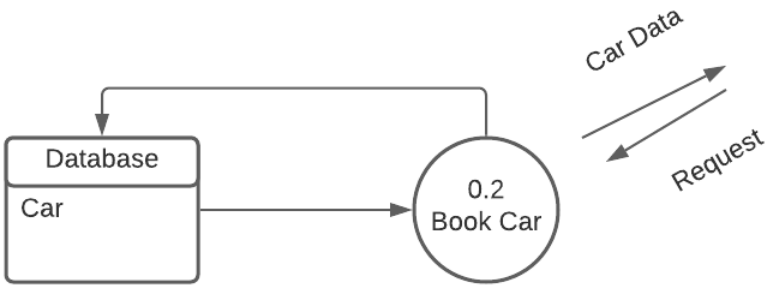
2. DFD Diagram

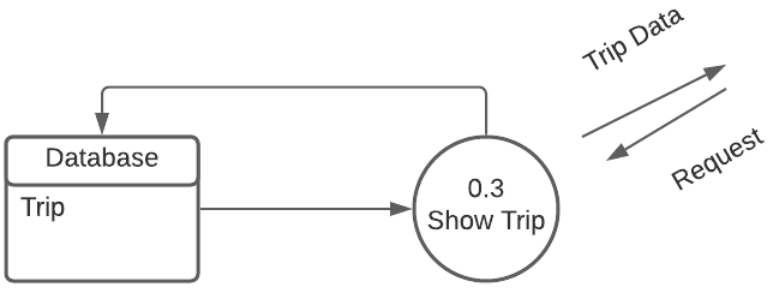
Level 0



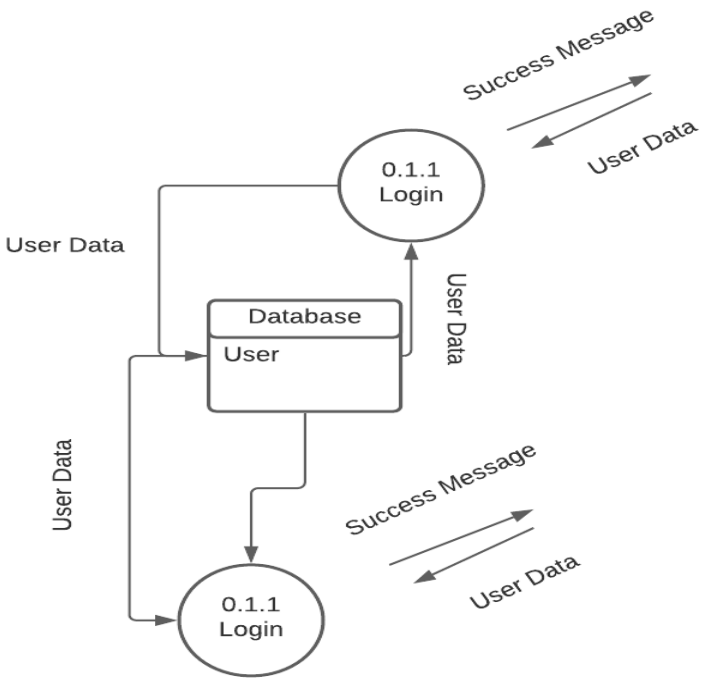
Level 1

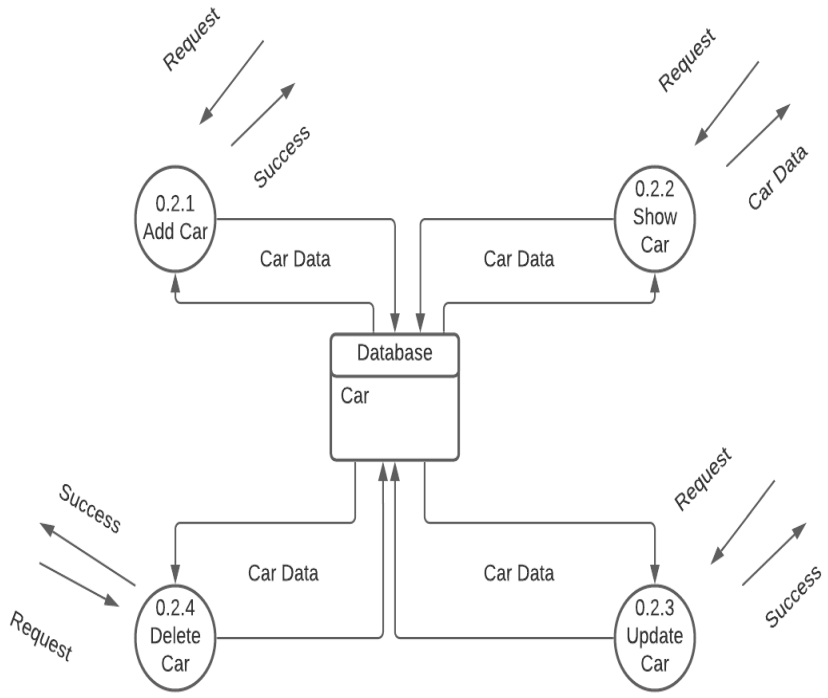


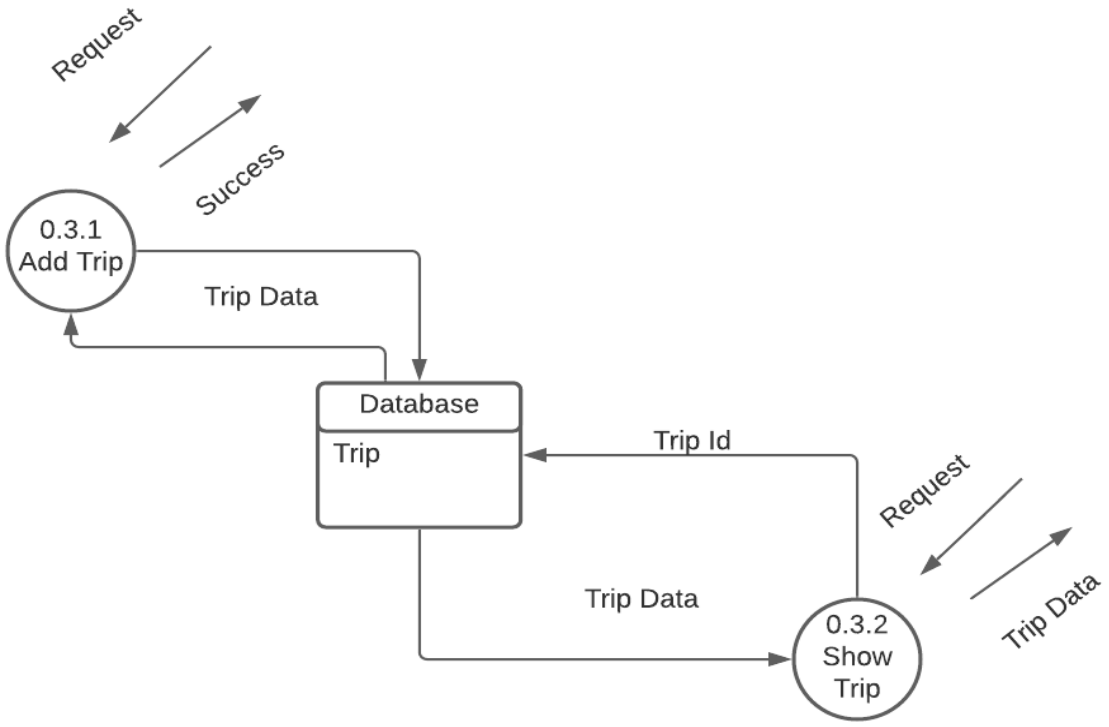

Car Data
Request
Database
Car
0.2
Book Car


Trip Data
Request
Database
Trip
0.3
Show Trip

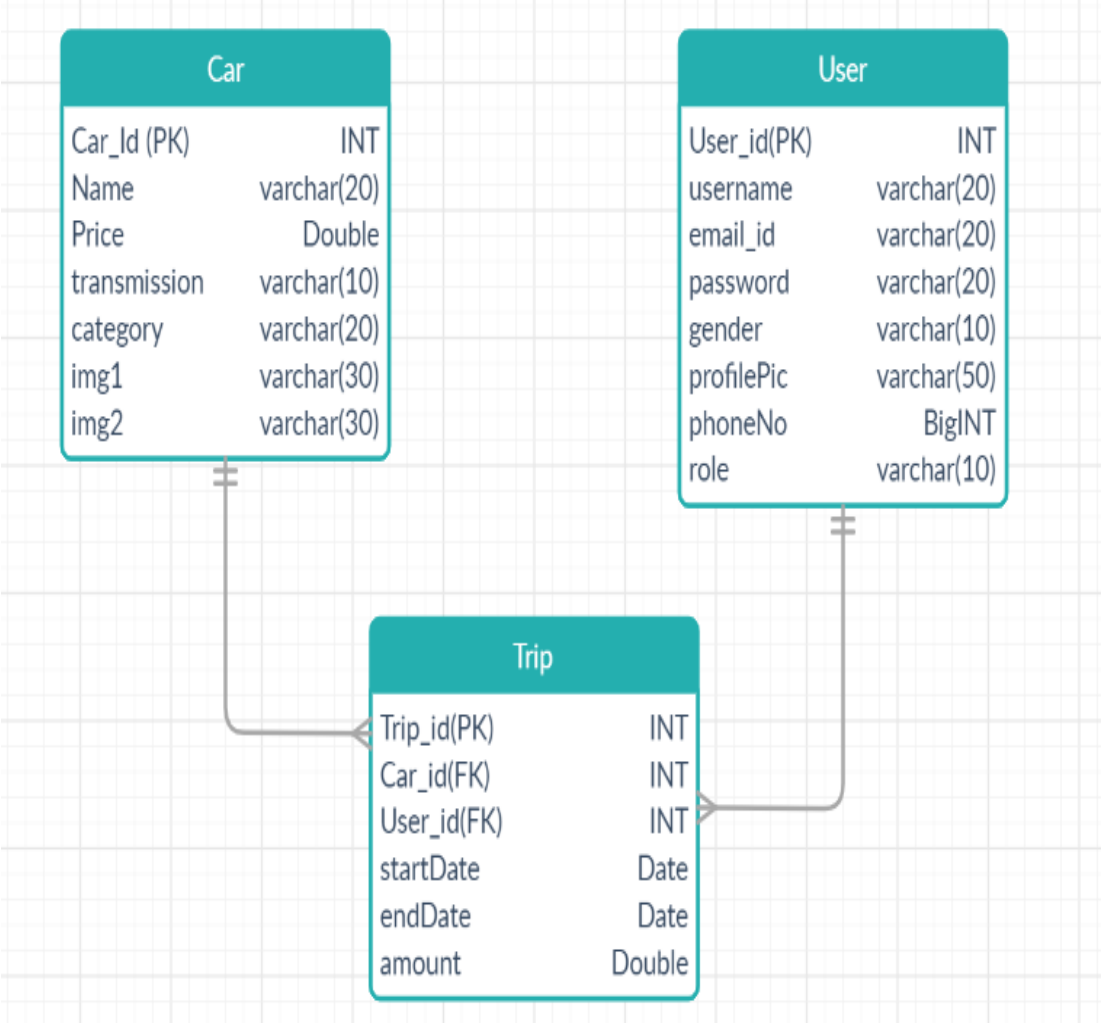
Level 2


Success Message
User Data
0.1.1
Login
User Data
User Data
Database
User
User Data
Success Message
User Data
0.1.1
Login


Request
Success
Request
Car Data
0.2.1
Add Car
0.2.2
Show
Car
Car Data
Car Data
Database
Car
Success
Request
Car Data
Car Data
Request
Success
0.2.4
Delete
Car
0.2.3
Update
Car


Request
Success
0.3.1
Add Trip
Trip Data
Database
Trip
Trip Id
Request
Trip Data
Trip Data
0.3.2
Show
Trip

3. E-R Diagram


Car
Car_Id (PK) INT
Name varchar(20)
Price Double
transmission varchar(10)
category varchar(20)
img1 varchar(30)
img2 varchar(30)
User
User_id(PK) INT
username varchar(20)
email_id varchar(20)
password varchar(20)
gender varchar(10)
profilePic varchar(50)
phoneNo BigINT
role varchar(10)
Trip
Trip_id(PK) INT
Car_id(FK) INT
User_id(FK) INT
startDate Date
endDate Date
amount Double

4. Data Dictionary

User							
Sr No.	Field Name	Data Type	Width	Required	Unique	PK/FK	Reference
1	User_Id	INT	20	Yes	Yes	PK	
2	username	Varchar	20	Yes	No		
3	Email_id	Varchar	20	Yes	Yes		
4	Phoneno	Big INT	10	Yes	No		
5	Gender	Varchar	10	Yes	No		
6	ProfilePic	Varchar	40	Yes	No		
7	Password	Varchar	10	Yes	Yes		
8	Role	Varchar	10	Yes	No		

Car							
Sr No.	Field Name	Data Type	Width	Required	Unique	PK/FK	Reference
1	Car_Id	INT	20	Yes	Yes	PK	
2	name	Varchar	20	Yes	No		
3	price	Double	10	Yes	No		
4	Transmission	Varchar	20	Yes	No		
5	Category	Varchar	10	Yes	No		
6	Img1	Varchar	40	Yes	No		
7	Img2	Varchar	40	Yes	No		

Trip							
Sr No.	Field Name	Data Type	Width	Required	Unique	PK/FK	Reference
1	Trip_Id	INT	20	Yes	Yes	PK	
2	Car_id	INT	20	Yes	Yes	FK	Car
3	User_id	INT	10	Yes	Yes	FK	User
4	stardDate	datetime		Yes	No		
5	endDate	Datetime		Yes	No		
6	Amount	Double	10	Yes	No		

5. Implementation Details

Home Module:

In the home page, a slideshow is shown along with the developer details and a navigation bar to go the desired location to go to the other pages like login module or signup module. The home page also consists of a footer which includes the contact details of the developer.

Login and Sign Up Module:

User must enter his/her username, email id, phone no. and a password to sign up. To login on the web application user must enter a valid email id and password. We have used ajax like call to validate user.

Car Details Module:

After user selects his/her car option from navigation menu, the page containing all available cars and its details are displayed. We have fetched the car details using angular module and for fetching data from the database where all the cars and its details are stored. For fetching data, the angular component sends a request to the http server which processes the request and fetches the details from the database and send the data to the angular component.

Booking Details Module:

When user clicks on book A Car option on the car details page, a Bootstrap Collapse feature of Bootstrap 4 show down card for user to enter booking details. They include pickup date and dropping date.

Admin Module:

Admin can add, delete and update car details and he/she can see all user and trips details.

6. Testing

Angular Testing Methods:

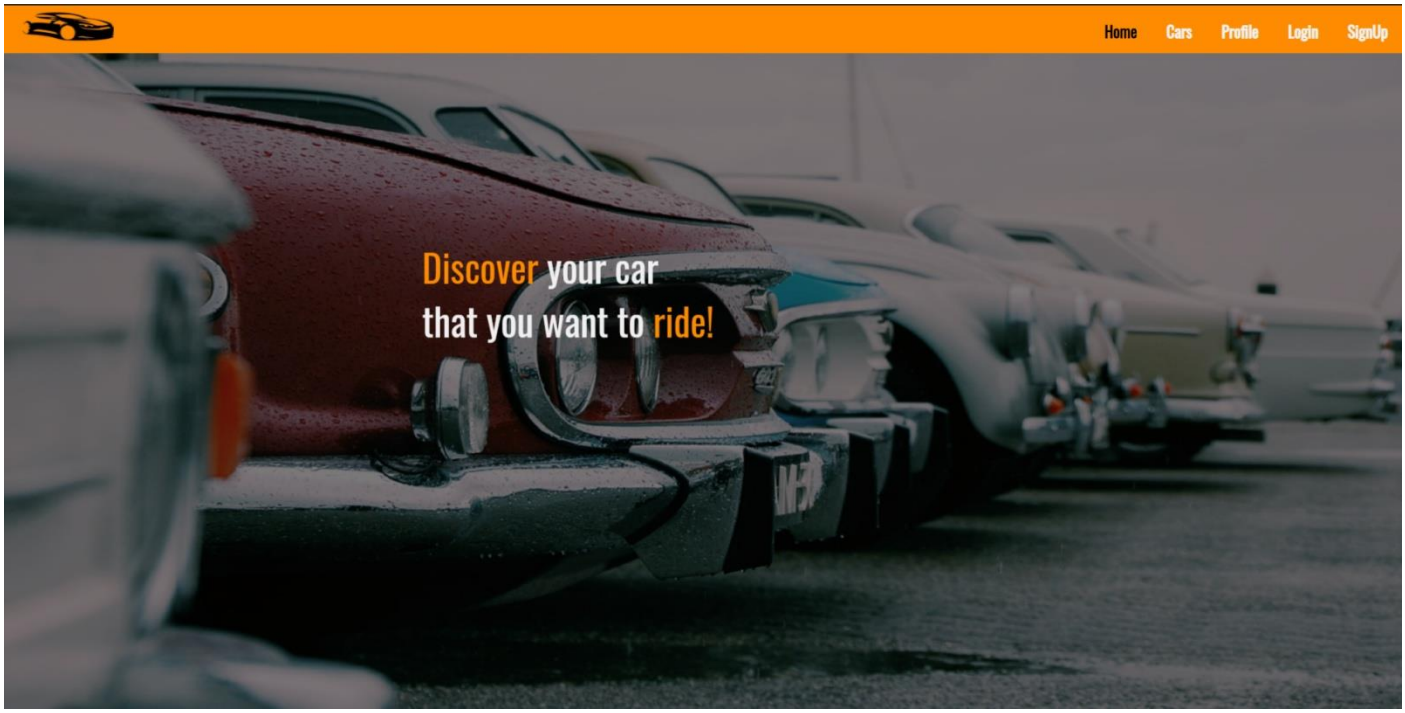
- We used Mozilla Firefox to do front-end testing.
- ng serve on terminal, also node.js needs to be executed for the DataBase connection.
- We have used node functionality for the sever.
- We used nodemon as a developers command for testing the condition of the node server.
- We used postman to test back-end APIs.

Test Cases:

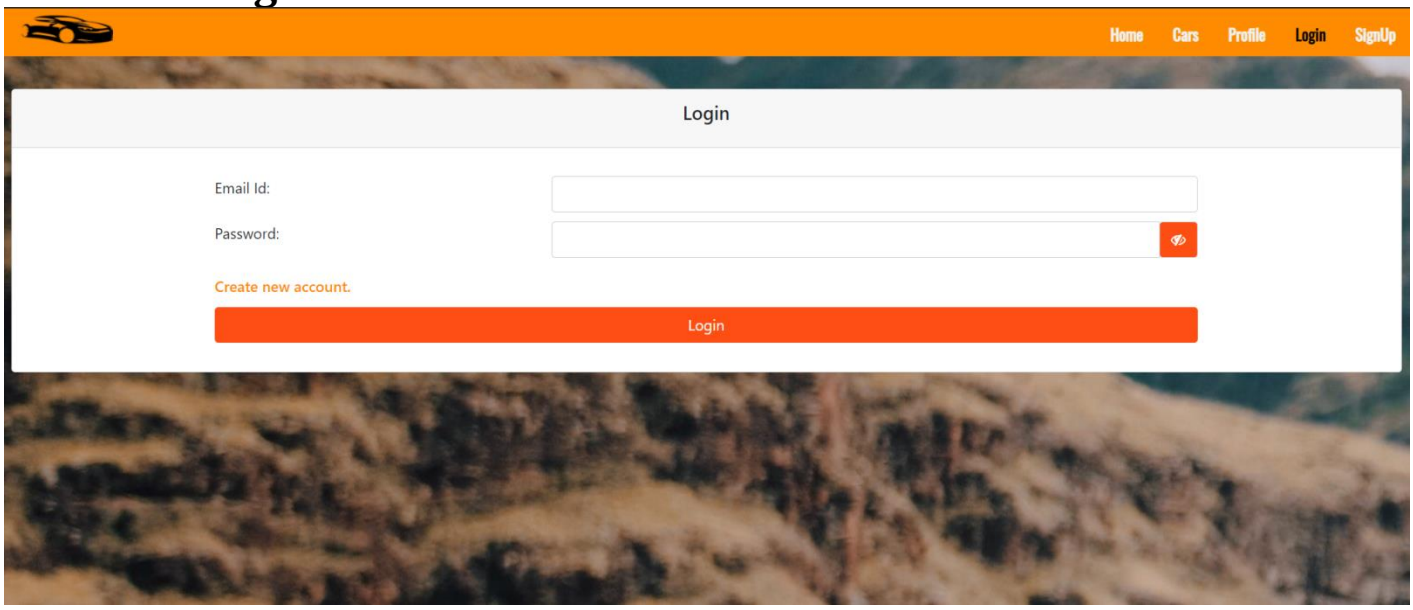
Module name	Field Input	Expected output	Actual output
Login	Username & Password (true input)	Redirect to Home page	Redirect to Home page
Login	Username & Password (wrong password)	Error message and Stay to login page	Error message and Stay to login page
SignUp	Email(email which Is already taken)	Warning message and stay to signup page	Warning message and stay to signup page
Car Details	Selection of valid car	Car details appear on screen	Car details appear on screen
Booking Details	Select date for booking (Car available)	Booking Done message	Booking Done message
Booking Details	Select date for booking (Car not available)	Error message to choose other dates	Error message to choose other dates

7. Screen Shots of the system

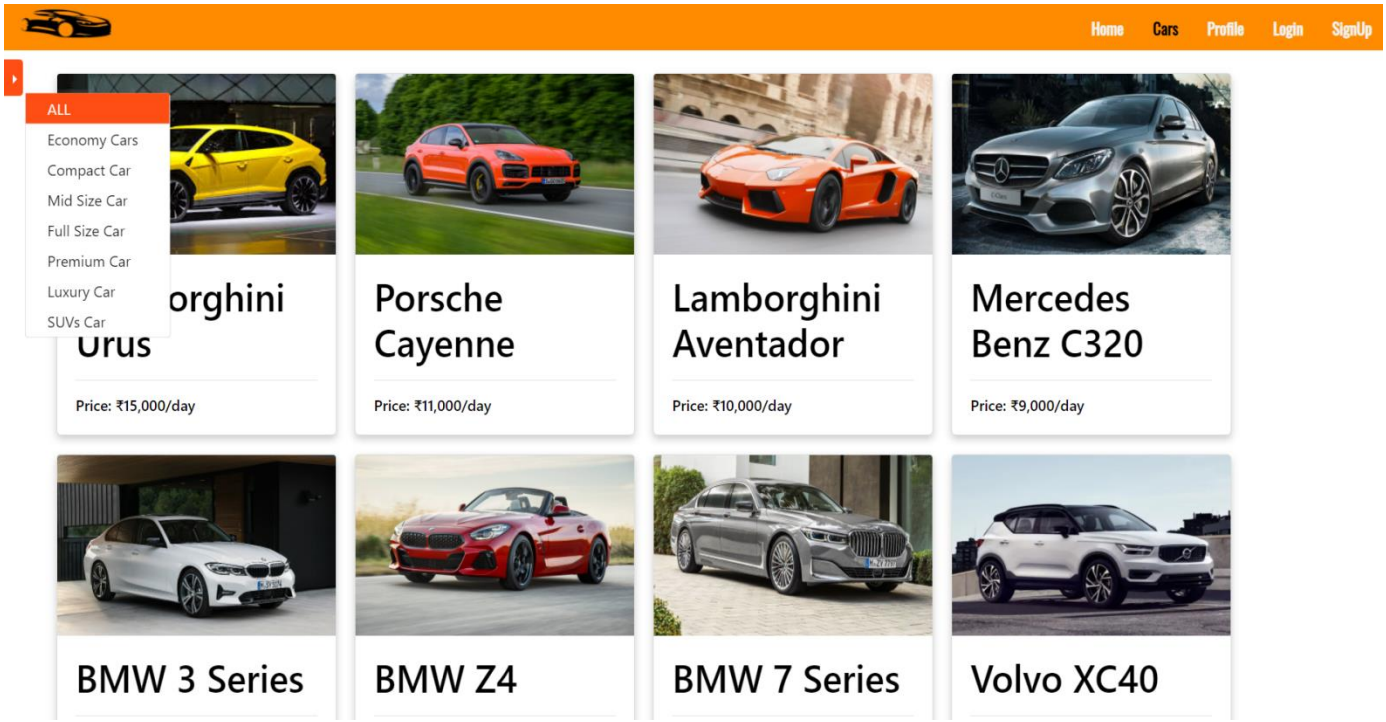
1. Home Page



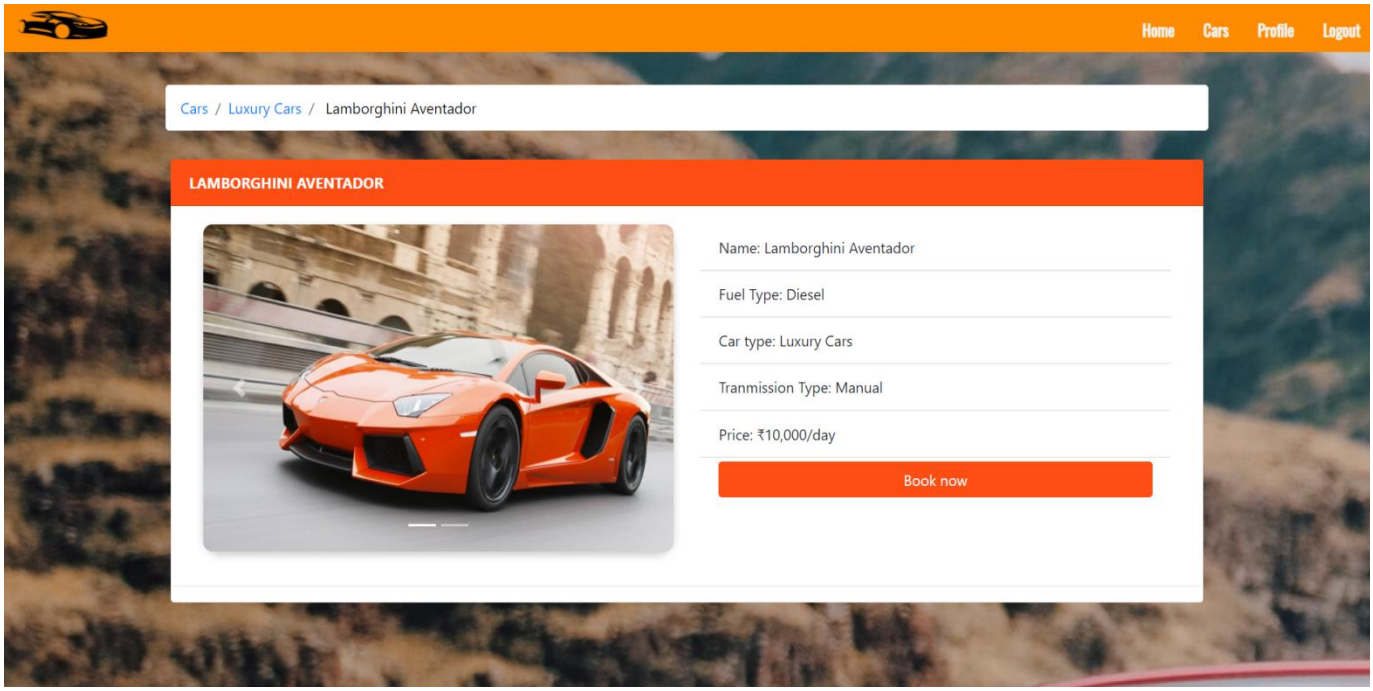
2. Login



3. Cars

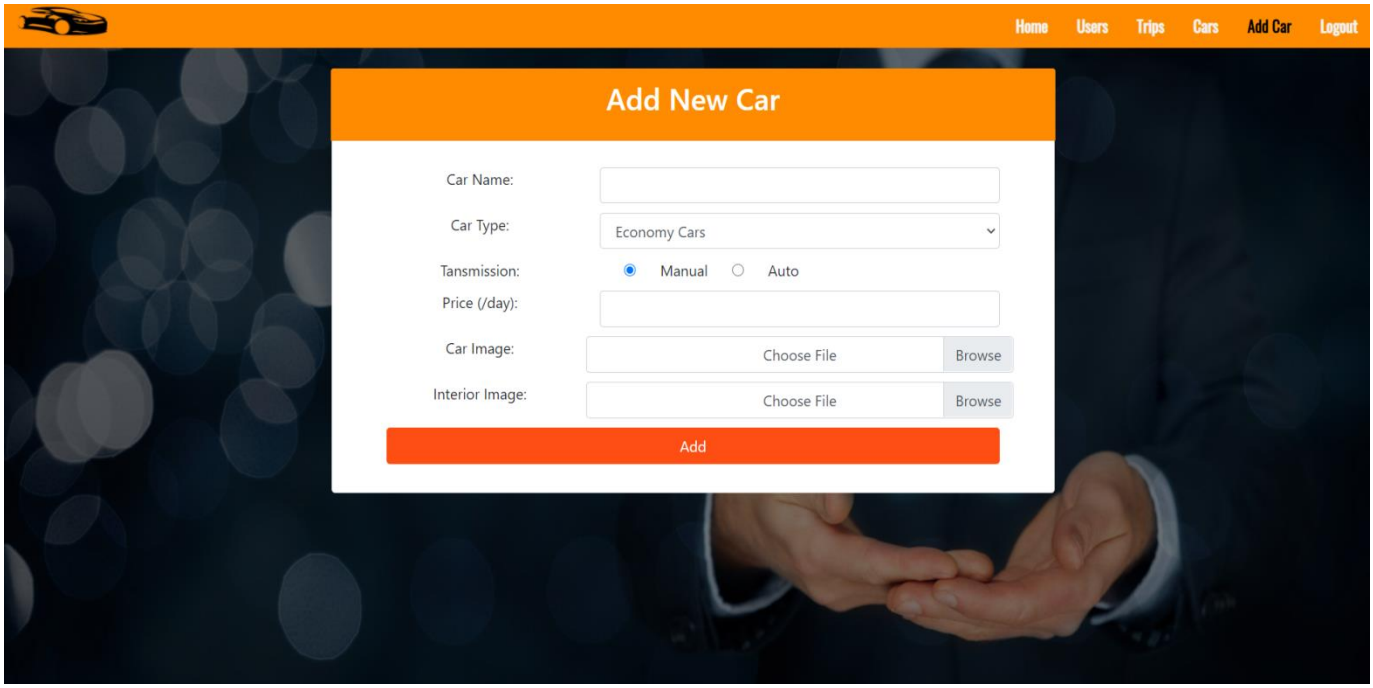


4. Car details



5. Admin Side

6. Add Car



8. Conclusion

Login functionality was successfully implemented by the use of Nodejs. Home page animation, careousels, footer was successfully implemented using CSS, Bootstrap and Javascript. Retrieving of cars is implemented using service class. Car details page was successfully implemented by fetching the details from database and displayed on the browser using CSS, Bootstrap and Javascript. Admin can add, delete and update car details and see all user and trip details.

9. Limitation and Future Extension

Limitations:

- Payment module is not implemented.
- Admin cannot see statistics of users.
- User can rent car for maximum a week.
- There is only one city.

Future Extension:

- Prices can be dynamic.
- Different cities can be included.
- Admin can see the user statistics.
- Email notifications(Email Verification)
- Add different offers

10. Bibliography

Stack Overflow:

https://stackoverflow.com/

W3School:

- HTML5: https://www.w3schools.com/html/default.asp
- CSS: https://www.w3schools.com/css/default.adp
- JavaScript: https://www.w3schools.com/js/default.asp
- AJAX: https://www.w3schools.com/js/js_ajax_intro.asp
- Jquery: https://www.w3schools.com/jquery/default.asp
- XML: https://www.w3schools.com/xml/default.asp
- JSON: https://www.w3schools.com/js/js_json_intro.asp

Bootstrap4:

https://getbootstrap.com/docs/4.3/getting-started/introduction/

Angular6:

https://angular.io/docs

Node.js:

https://nodejs.org/en/

Mongo DB:

https://www.mongodb.com/