|  |  |
| --- | --- |
| Group Name | Software Buds |
| Group Members | Twinklepreet Kaur (300289566)  Karanbir Singh (300283754)  Nikhil Mahajan (300287270)  Sahil Meangi (300292961) |

Project Report-Car Rental

CSIS 3275 – Software engineering

instructor: IVan Wong

2019

**Car Rental System**

**Overview:**

The project is designed as to be used by Car Rental company specializing in renting car to customers. It is going to be an online system which will help customers view available cars, register, and book a car. This application will also help to view specifications of the selected vehicle. A new up and coming car rental service is wishing to have a user interface that will allow their customers to view the models, descriptions and prices of different cars available. The user has the ability to register and log in to the Web App and track their rental plan. The Web App will be responsive, allowing for the customer to view it on any device, from tablets to mobile phones and desktop computers.

The car rental is a website used to rent a car for a fixed time period by paying a certain fee. The website can be used to book a car by few easy steps. This application helps the people to rent their car of own choice amongst many cars. The cars are also rented accordingly like for Wedding, Tours, Single need etc. This application will be responsive for the logins from laptop, tablets and mobile phones too. People can check the price of the cars and can make the rental plan

**Target Users:** All the audience who are in desperate need of car for some span of time and cannot afford to buy it.

**Problems Solving:** Our product solves a lot of problems, since users can rent a car anytime anywhere. They can open the webpage to check the available cars and their prices rather than calling the offices directly.

**Summary of Problems and challenges:** When vacationing abroad, people have a very limited amount of time to experience activities and often end up cutting things they really want to do out of their itineraries.

If they rent a car when traveling, however, they will have the time and convenience needed to get the most out of their adventures. So turning onto the point, our main focus will be to make the website simple to be understood by the users to rent the car.

**Goals For the Project :**

This software, called ‘Rent and drive’, will allow for the company to access their database securely and safely in a user-friendly online environment. Allowing for them to change car information with ease. The software will be in sync with the both the Web App, allowing for real-time up-to-date services for their customers.

***Scope of the Project:***

**System:**

1. Provide car collection for users as an alternative for them to select car if they want to choose on their own.
2. Allow the Admin to add a new car.
3. Allow admin to get user information from the database based on the user’s ID card number or name.

**User:**

1. All authorized users can access the system

2. Allow the user to view information of the available car.

3. Allow the user to enquire about the car.

**Software Toolset**: The front-end stack would consist of HTML and CSS. HTML is very simple to use, CSS will be used to style the layout and design of the WebApp. The backend would consist of Java, PHP. Java is capable of supporting a wide variety of applications. The Framework or Platform we chose for now is Java spring framework which provides a light-weight framework for batch applications, it is also capable of detecting the device and adapt the application behaviour. It will also provide us with extensible templates. For libraries, we think that for a good start we’ll choose Skeleton. Afterwards, CSS libraries like Animation.css would be helpful in making the web pages more eye-catching. All our information about everything from customer’s information to car’s specifications will be stored in the database using Firebase database in Console because it is pretty straight forward to use. GitHub is the version control system we’ll use to make any changes in the documentation or web application.

**How to run a software:** The project at last works on Localhost with command <http://localhost:8080/> . The port **8080** is often used for this purpose. It is just a random port that web software is listening on because someone chose to set it up that way. ... So, “**localhost 8080**” means “this computer , application on port **8080**”.

**Group Dynamics:** Project manager is the one who is responsible to guide the team and also breaks down the different tasks and functions and then assign the tasks according to people's abilities. If we have to choose project manager our team would like to choose **Twinklepreet Kaur** as she is having good knowledge about the project and some previous experience too. Other members will have to do the remaining tasks like Analysis, design, development, tests, implementation, tracking and maintenance. Yes, all of us will share development work.

These are the most important roles which team members need to do in order to develop a project. If a disagreement arises, we will check on the matter and see if anything needs to be changed or upgraded in any of the processes and declining towards a solution which is acceptable by all the members.

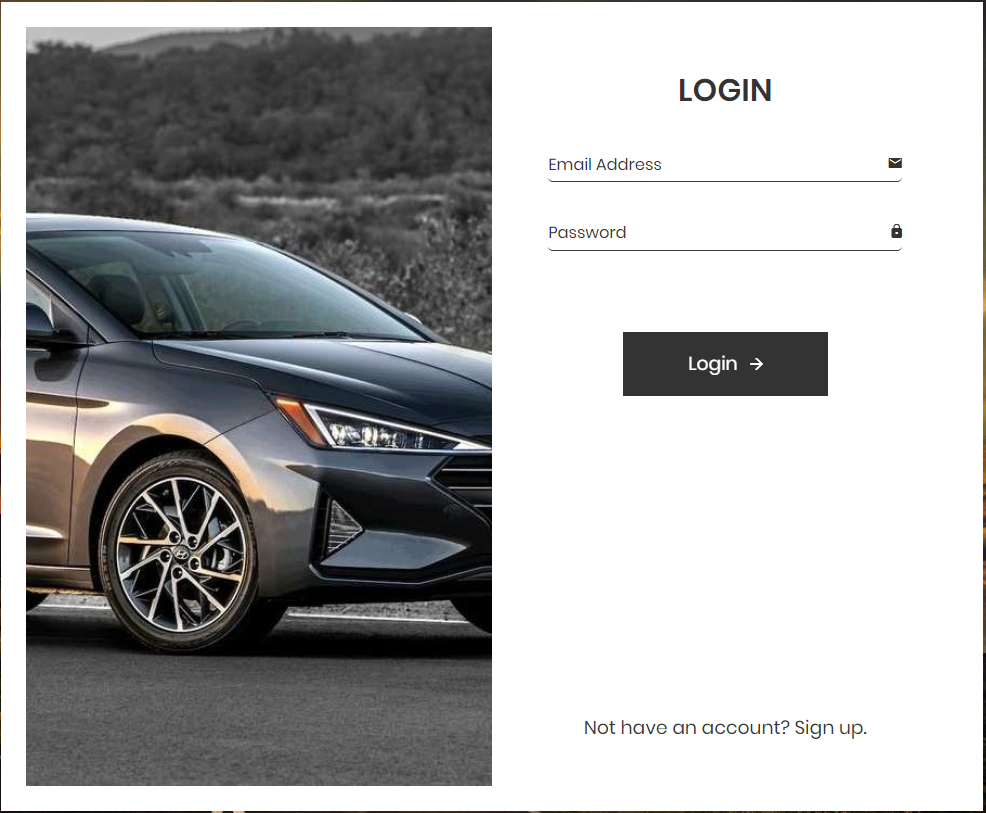
**Karanbir Singh:** Worked for html , javascript pages in the project

**Twinklepreet Kaur & Sahil Meangi** : Set up all the eclipse problems with error codes and running code

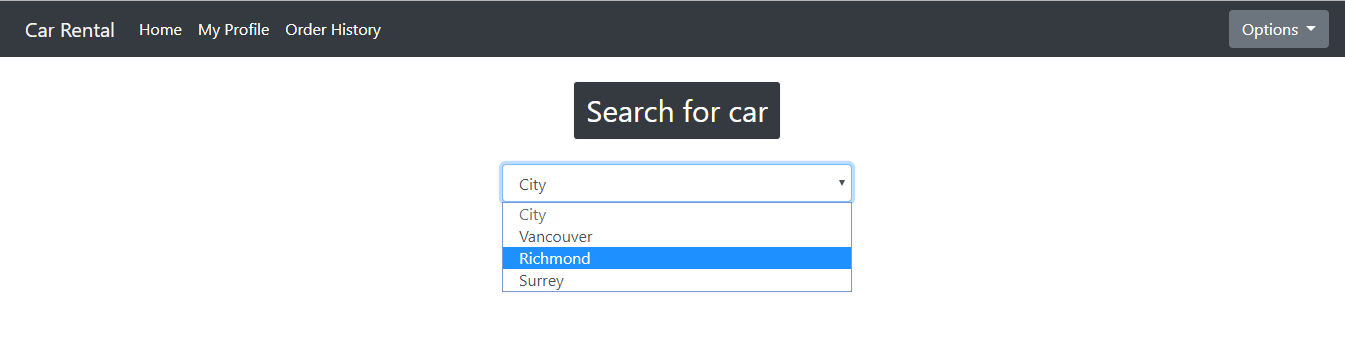
**Nikhil Mahajan & Twinklepreet Kaur**: Helped with hand written documents and setting up meeting minutes with team members.

**Features and Functions**

1. The Login Page: The first page allows the user to login or to sign up if you are not already registered on the website.

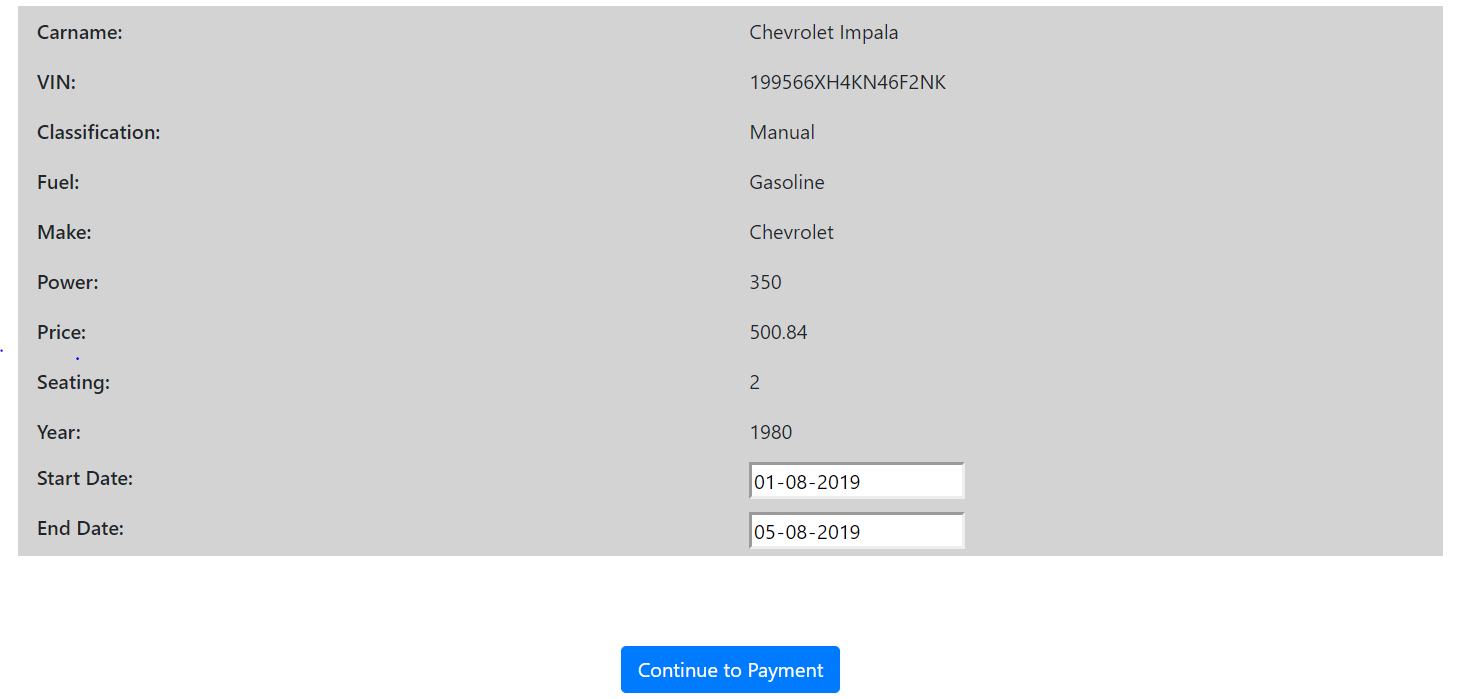


1. The second page will allow the user to choose cars from different cities: Richmond, Surrey, Vancouver.

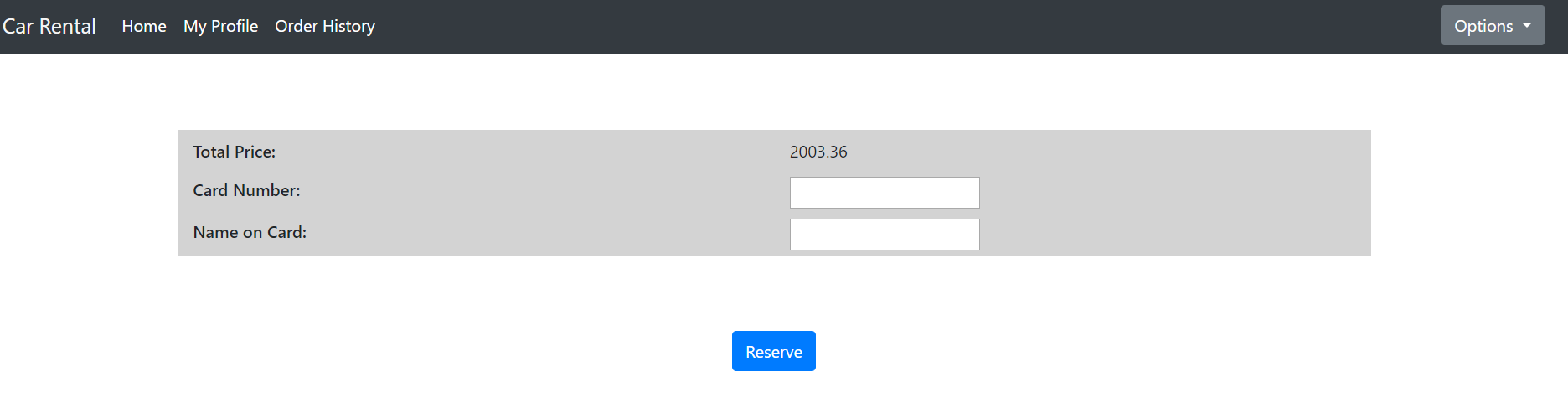


1. Users can select their desired vehicle and can check their features, and desired dates for booking.
2. After that there will be the list of available cars in that appropriate city





1. Choosing a car with all details a user can proceed with payment.
2. The last page will show up the payment details of card and reserve the seat.



**Use Case#1**:Register to Car Rental

Primary actor: General user

**Goal in context:** To register to the Car rental when the user visits the Car rental website

**Precondition:** Good internet on the device and a user can access the correct URL

**Trigger:** In the login page, click ‘Sign up’ button

Scenario:

1) User: type the car rental URL to access the system

2) User: click ‘Sign up’ button under log in page

3) User: input the user profile

4) User: click ‘Submit’

Exceptions:

1) The server or database is not installed properly, then it shows error message when click the ‘Sign up’

2) Invalid ID, name and password are input in the registration form: ‘Submit’ doesn’t work.

3) Duplicate ID does not allow in the registration form: ‘Submit’ doesn’t work.

Priority: Essential, must be implemented before login

When available: In the first login page

Frequency of use: Once

Channel to actor: Via web browser

Secondary actors: User manager or Administrator

Channels to secondary actors: Support via email address

Open issues:

1) Is there a way to change the user ID?

2) Should the administrator assign the role to users?

3) Is there a way to approve without user ID?

**Use Case #2** : Change Car Rental Date

Primary Actor: Car Renter

Goal in Context: To change date of when Car Renter wants car

Preconditions: Car Renter has made a rental and wishes to change date of car rental

Trigger: Car Renter has switch the date of when Car is being rented on

Scenario:

1. Car Renter Logs onto Car Rental service site (Enters Username/ Password)

2. Car Renter Selects “My Car Rental” on website.

3. Car Renter Selects “Car Rental” on website.

4. Car Renter Selects “Change Date” on website.

5. Car Renter selects new date for car rental and accepts the new date

7. Car Renter is prompt to enter “password” sent to his/her E-mail to validate the switch

8. Car Renter enters “password” and sees the new date for the car rental

Exceptions:

1. Car Renter denied request, Car Rental date is too close to actual date of the rental,

cannot make change.

2. Car Renter submits wrong “password”; is prompt to re-enter otherwise the date will not be changed

3.Car Renter enters wrong date and accepts changes; is prompt to call Customer service

and talk to representative about issue

Priority: Essential, must be implemented

When available: First increment

Frequency of use: Once per quarter to Registered Members

Channel to actor: Car Rental Website

Secondary Actors: Car Rental Customer Service

Channels to Secondary Actors:

1. Customer Service: Phone line

Open Issues:

1. How many times can a Car Rental member request this service in a given period of time?

2. Should the Car Rental member be charged extra for a convenience fee for using this

service?

**Use Case #3**: Checking Car Status

Primary Actor: Car Renter

Goal in Context: To see if car rental is on route to Car Renter

Preconditions: Car Renter has made a rental in the past and is nearing the time that the car should be delivered to the car renter

Trigger: The Car Renter knows the current information of where his/her car is at.

Scenario:

1. Car Renter: Logs onto Car Rental service site (Enters Username/ Password)

2. Car Renter: Selects “My Car Rental” on website.

3. Car Renter: Selects “Car Status” on website.

4. Car Renter: Observes status of car to know current information on car.

Exceptions:

1. Car Rental service Username/Password incorrect: Car Renter is sent to main page to reenter credentials

2. Password is not recognized: E-mail is sent to Car Renters current E-mail and prompt to enter new password

3. “Car Status” is unavailable: Car Renter is prompt to call Car Rental Service for further

details on Car Status.

Priority: Essential, must be implemented

When available: First increment

Frequency of use: Couple times per day/ week

Channel to actor: Car Rental Website

Secondary Actors: Car Rental Customer Service

Channels to Secondary Actors:

1. Customer Service: Phone line / Online chat support

2. Automated E-mail: Sent to Car Renters E-mail for status request

Open Issues:

1. Should there be an option to opt out of the Car Rental on the day of the Car Rental

date?

2. Should the Car Renter have an option to change what car he/she wants after confirming what car they decided on(i.e.: going from a stock model to luxury)?

3. How much time does the Car Renter have on their account before an amount of time

has passed before the system automatically logs them out due to inactivity?

**How to setup or install the software and see the homepage**

There are several steps to set up the whole development environment. Here are the requirements.

Step1. Install JDK 1.8 or higher in your pc.

Step2. Install IDE Eclipse (different version of Eclipse)

Step3. Create the new maven project or pull <https://github.com/Twinklepreet27/Project_CarRental> and import to the project

Step4.After file synchronization, make sure the maven build. If it shows the problem, then try to

download libraries for dependency.

Step5. Run the project with selection of maven build and try to open a tab in any browser (google chrome is used ) and add write <http://localhost:8080/> . Without this, it will work properly.

Step6. Use firebase database to show the details of the registered accounts setting for database link.

Step7.In the first page, try to click user registration and then you can login the system properly.



**Conclusion :**

Car Rental Management System will improve the management of rentals and bookings of cars when people face difficulty to go to a particular renting dealership of cars and they have to go again and again to make changes . This will save a lot of time and people will feel glad to make anytime changes as they like . This system developed in order to overcome the problems faced using the manual way. It acts as agent for car owners to rent out their car and the company can also earn profit from each of the transaction. Lastly, our team did a lot of efforts to make the project working and error free . It was interesting to work in a team and learn a lot from each member.