American International University- Bangladesh (AIUB)

**Department of Computer Science**
**Faculty of Science &Technology (FST)**
**Fall 20_21**

**Section:**
**Group No: 09**

# Mechanic Koi

A software Engineering project submitted
By

| Serial No. | Student Name | Student ID |
|---|---|---|
| 10 | HAQUE, TASRIFUL | 18-38661-3 |
| 11 | CHAKRABORTY, ADITTYA | 18-38693-3 |
| 12 | Md. Saif Ali | 18-38710-3 |
| 37 | Naim Hasan Mim | 19-40604-1 |

The project will be Evaluated for the following Course Outcomes

| | | |
|---|---|---|
| Requirements Analysis (functional, quality, and project requirements) | [5Marks] | **Total Marks** |
| System Design (UML, UI/UX design) | [5Marks] | |
| Test and Project Management Planning | [5Marks] | |
| Submission, Completeness, Spelling, Grammar and Organization | [5Marks] | |

Submission Date: 19.12.20

# PRODUCT AND PROJECT DESCRIPTION

## 1. System Features

### 1.1 Create account

#### 1.1.1 Description and Priority

If user use this software, then first time they need must create an account in the software. First of all, select what is your account "type customer" or "garage owner". Then, here the necessary information like email, login password of this software, phone number, Location of the users will be stored through this feature. User must have to create account for use this software.

#### 1.1.2 Stimulus/Response Sequences

- Choose 'Create Account' option.
- Select "Customer" Or "Garage Owner"
- Fill up the First Name, Last Name, Name, Email id, Phone number, password,
- Click/press "Signup"
- If the information is successfully imputed then show successful if not then will show error at the wrong point and have to give the information again

#### 1.1.3 Functional Requirements

INPUT: Request for First Name, Last Name, Email id, Phone Number, Password

PROCESSING: Retrieves the provided information and makes a new account for user

OUTPUT: Displayed created account

REQ-1: User Information

### 1.2 Log In

#### 1.2.1 Description and priority:

User need to login for use this software. User have to give user name/email & password. This is mandatory to login.

#### 1.2.2 Stimulus/Response Sequences

- Have to Fill up the two box with Name/Email & password.
- Have To press the 'Log IN' button.
- If the information is successfully imputed then show successful if not then will show error at the wrong point and have to give the information again.

### 1.2.3 Functional Requirements

INPUT: User Name/Email & password

PROCESSING: Retrieves the provided user Name/Email & password and make sure the login successful

OUTPUT: Display the home page.

REQ1: User Name/email

REQ2: User Password

## 1.3 Select Vehicle Features and get support

### 1.3.1 Description and priority:

First of all user need to which Vehicles Features they use “Vehicle Repair”, “Vehicle Maintenance” Or “Vehicle purchase”. then need to on customers GPS location. After that Garage Owner Accept What type of Features or Service Customer needs. Then they will provide it.

### 1.3.2 Stimulus/Response Sequences

- Select Vehicle Feature “Vehicle Repair”, “Vehicle Maintenance” Or “Vehicle purchase”
- Customer Write Details as related to vehicle feature option
- Customer needs to Allow permission for GPS to get that services
- Then have to press ‘Get Service’ button

### 1.3.3 Functional Requirements

INPUT: User selected Vehicle Feature

PROCESSING: Retrieves the provided selected features and processing for next step

OUTPUT: Display another page with some features

REQ1: Allow permission for GPS option

## 1.4 Rent any Vehicle

### 1.4.1 Description and priority:

User have to select which types of Vehicles they need. Then user need to select When and how much time for this vehicle they need. If there is any vehicle available then Customers can book this vehicle.

### 1.4.2 Stimulus/Response Sequences

- Have to select the types of Vehicles
- Select Time and Date
- If there is any vehicle is available then show this to customers.
- Then have to press ‘Book Now’ button

### 1.4.3 Functional Requirements

INPUT: Select the types of Vehicles, time and date

PROCESSING: Search This type of Vehicles and timing schedule is Available or not

OUTPUT: Show the page That Vehicle and that time schedule is available or not for booking

REQ1: Select Vehicles

REQ2: Select Time and Date

## 1.5 Purchase Vehicle or Parts

### 1.5.1 Description and priority:

User have to select Vehicle or parts which they need from this software ecommerce platform. After Select they can purchase it

### 1.5.2 Stimulus/Response Sequences

- Have to select the vehicles or parts product
- Then have to press ‘Purchase’ button

### 1.5.3 Functional Requirements

INPUT: User select Vehicle or parts Product

PROCESSING: Processing for purchase

OUTPUT: Purchase it

REQ1: Select Vehicle or Parts product

## 1.6 Online Payment

### 1.6.1 Description and priority:

After getting Service or select a product for buy the next step is online payment. It is mandatory because without payment order is not successfully happened.

### 1.6.2 Stimulus/Response Sequences

- Have to press on 'Order' option
- Then Select Online payment
- After that the payment platforms will come
- Have to select a platform

### 1.6.3 Functional Requirements

INPUT: Online payment, Payment platform

PROCESSING: Retrieves the provided option in background and show

OUTPUT: Display the Payment platforms and' Order Successful 'message

REQ1: payment platform should be selected

- **Software Quality Attributes**

**Standards Compliance:**

There shall be consistency in variable names within the system. The graphical interface shall have a consistent look and feel.

**Availability:** The system shall be available 24*7 times.

**Maintainability:** The ability to maintain, modify information and update fix problem of the system.

**Usability:** software can be used again and again without distortion.

**Accessibility:** Administrator and many other users can access the system but the access level is controlled for each user according to their work scope.

## 1.7 Project Requirements

**Duration to complete:** Total 3 months need for complete this project.

**Developer need:** 6 developers are needed.

**Cost:** 50 thousand taka per month for each developer.

**Period:** total 6 periods in 3 months.

**Alpha Test:** $4^{th}$ number of period alpha test.

**Beta Test:** 5th number of period beta test.

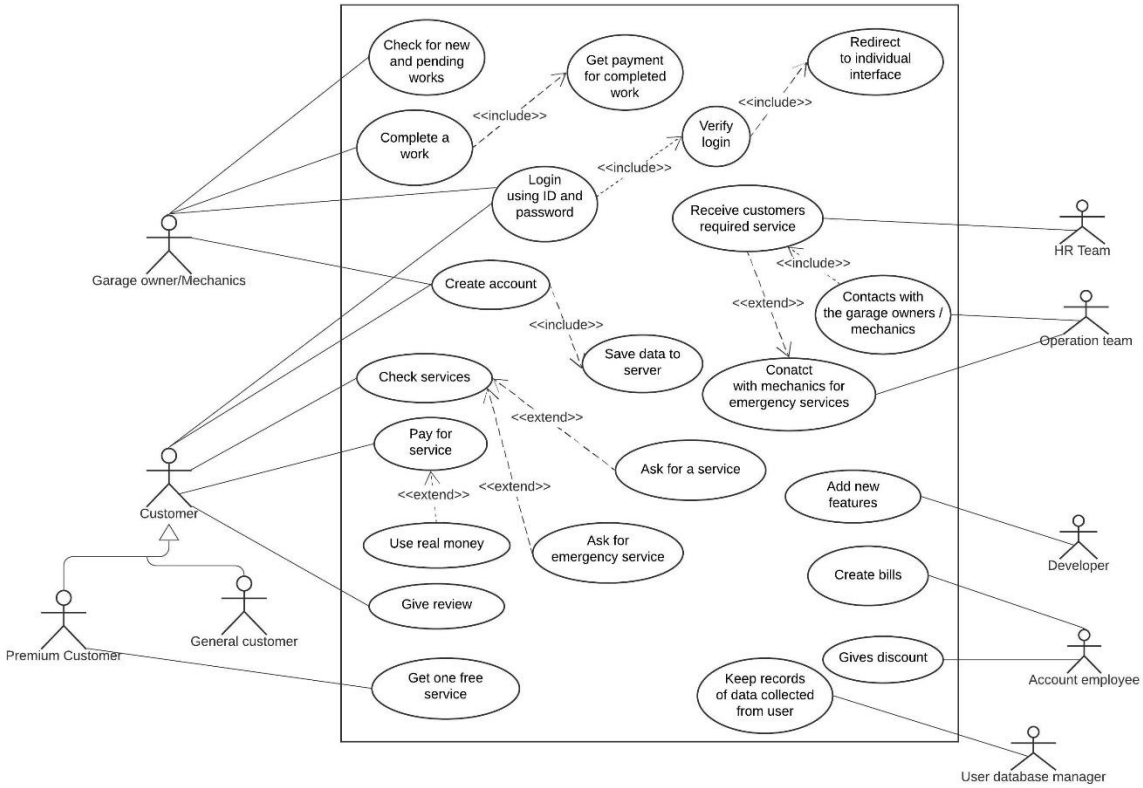
**App launch:** App launch in 6th number of period.

## 2. SYSTEM DESIGN SPECIFICATION

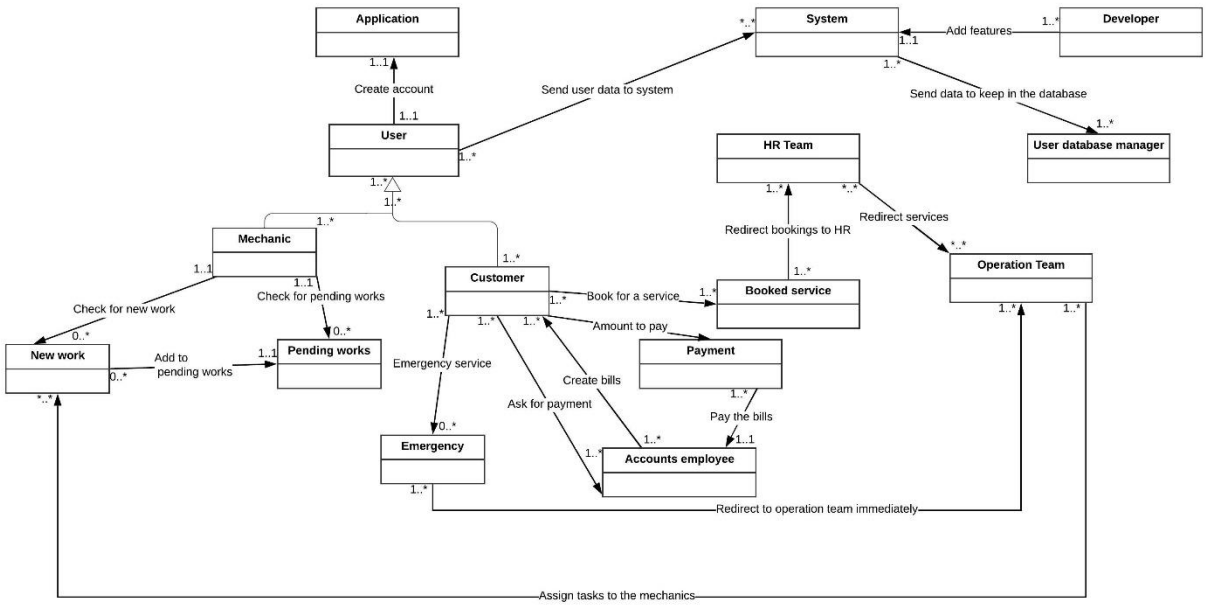
**Scenario:** "Mechanic Koi" is an online platform that provides 360-degree vehicle solutions & on-demand emergency repair services. With the motto of being the roadside best friends, they want to create the biggest innovative revolution in the automobile industry within five years. They want to build an application for their customers and garage owners to manage everything efficiently. In their app, there will be a login page. If the customer or garage owner already have an account they can sign in; otherwise, they have to create an account. Each account will be an individual account, and the account database for the customers, garage owners and admin panel will be different. With a single application, everyone can use every required feature for them. The website of the company will connect the software. After a successful login, a customer can choose any options, like - Home & On-road Emergency repair Support, Vehicle Health Consultancy, Vehicle Parts e-commerce platform and they can check the nearest garage from their location. On the other hand, the garage owner or manager who will use the app to communicate with the customers can quickly locate where the customer is. Whether the customer will come to the garage or they need to send a mechanic if it is an on-road emergency repair or else what should a garage manager or owner know. Admins of that company can also log in and check everything about the app. The company will update the software after some specific time for next five years depending on the customer feedback and upcoming new services. Mobile data or wifi will be needed to use the application. There will be some options for payment. The customers can use e-banking system for payment, or they can choose cash. They can buy automobile-related products using the app, and they can also pay using e-banking or cash on delivery. The software will connect the company's chatbot so that anyone can check their car or bike problems using the bot. They can detect a problem by themselves.

### 2.1 System Design

- USE Case diagram:

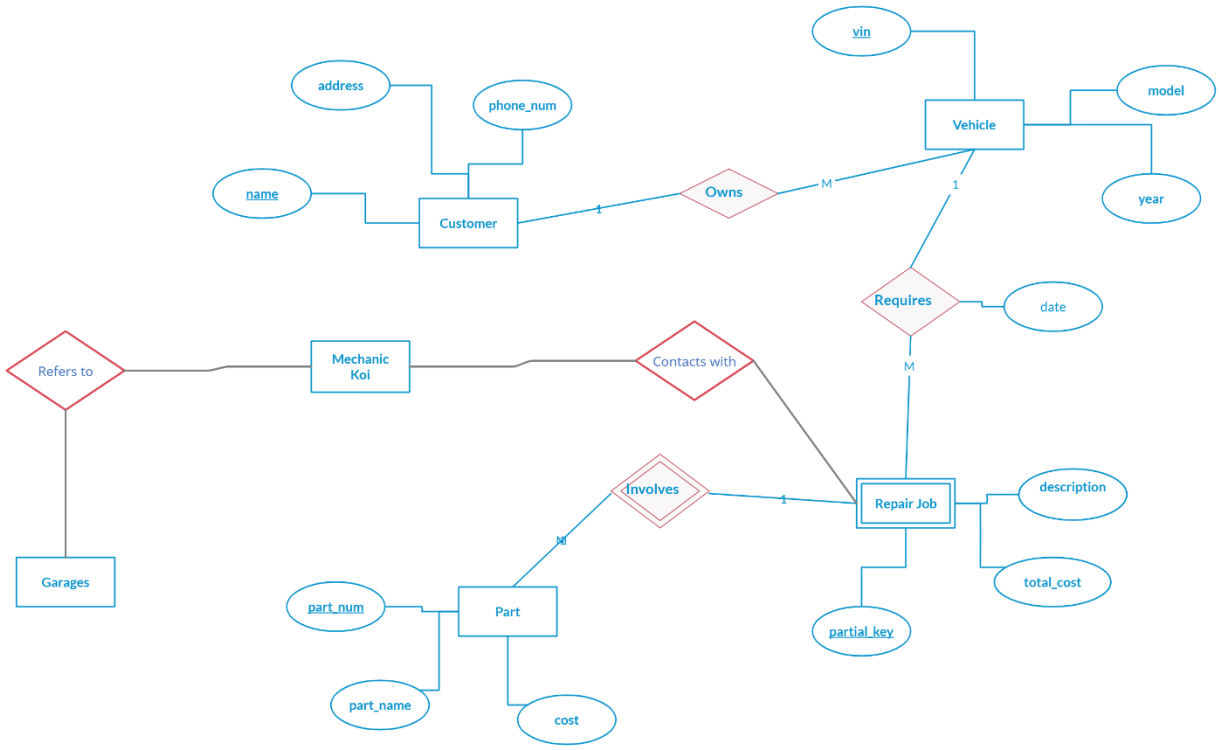



- Class Diagram:




- Identify various elements such as controller class, objects, boundaries, messages etc. of the sequence diagram. And, Draw the **sequence diagram** as per the norms.

- Identify various actions and corresponding events of triggering actions. And, draw the **activity diagram** as per the norms.
- ER Diagram:



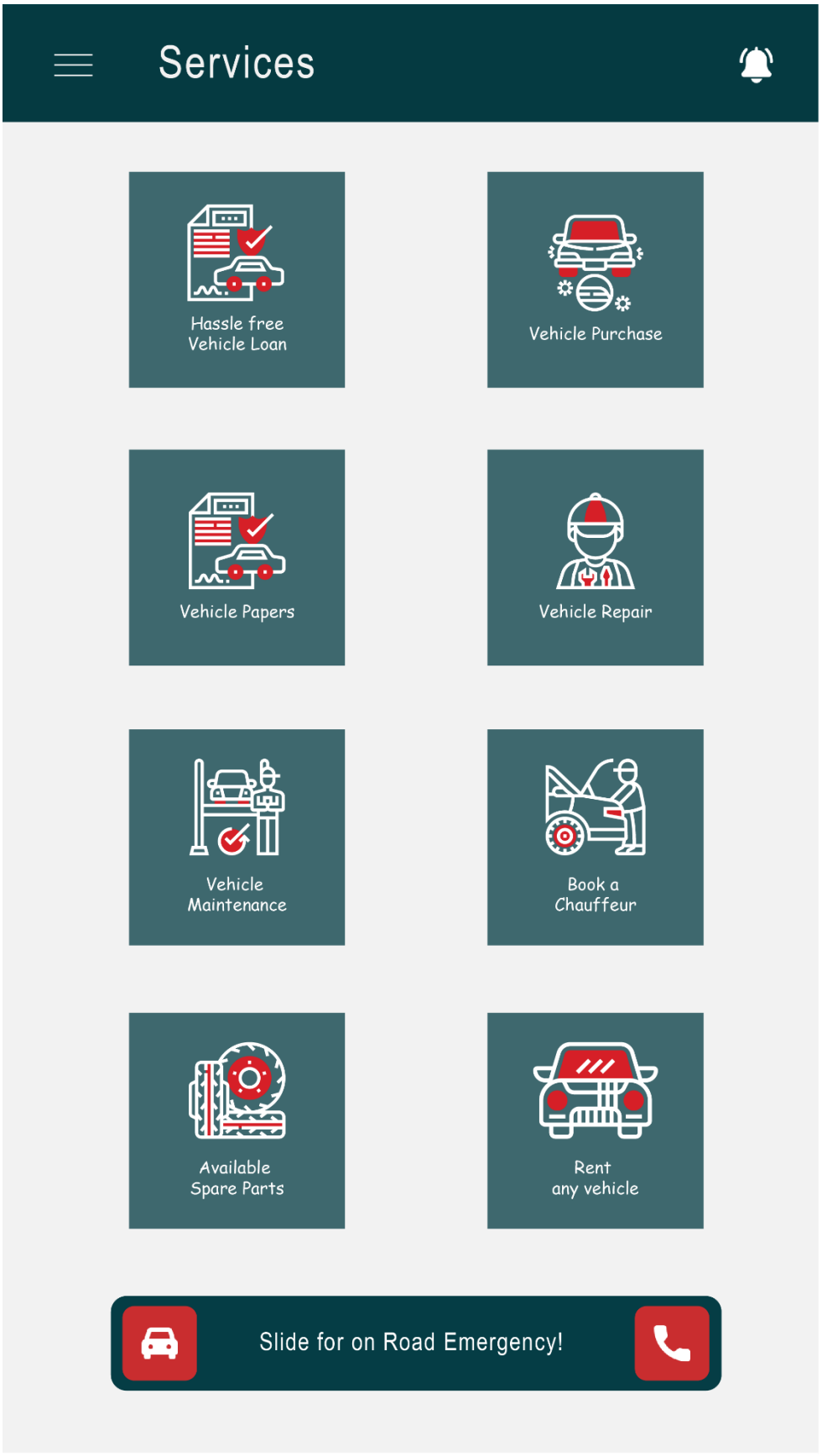

## 2.2 UI/UX Design

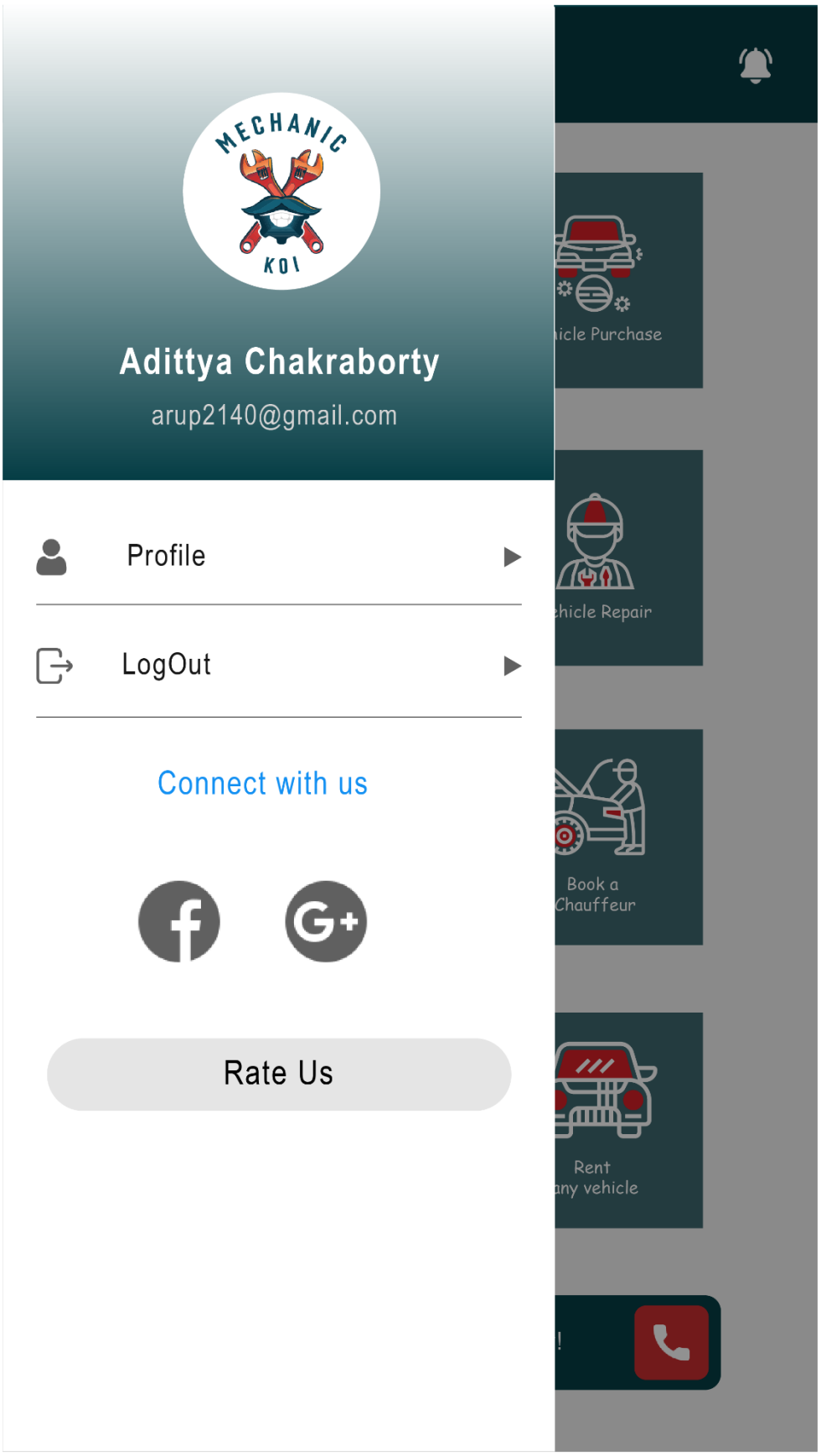
- Here is some design of our proposed mobile software prototype:


# Make Appointment

Auto Service Manager will then contact you to confirm
your appointment via email or phone.


MECHANIC
KOI
Name
Address
Area & City
Car Name & Model
Phone no
Email
Password
ConfirmPassword
Register


Services
Hassle free
Vehicle Loan
Vehicle Purchase
Vehicle Papers
Vehicle Repair
Vehicle
Maintenance
Book a
Chauffeur
Available
Spare Parts
Rent
any vehicle
Slide for on Road Emergency!


MECHANIC
KOI
Adittya Chakraborty
arup2140@gmail.com
Profile
LogOut
Connect with us
Rate Us
icle Purchase
hicle Repair
Book a
Chauffeur
Rent
ny vehicle


## Adittya Chakraborty

User

| Services | Review | Total Spent |
|---|---|---|
| 0 | 0 | 0 |

### Your Information

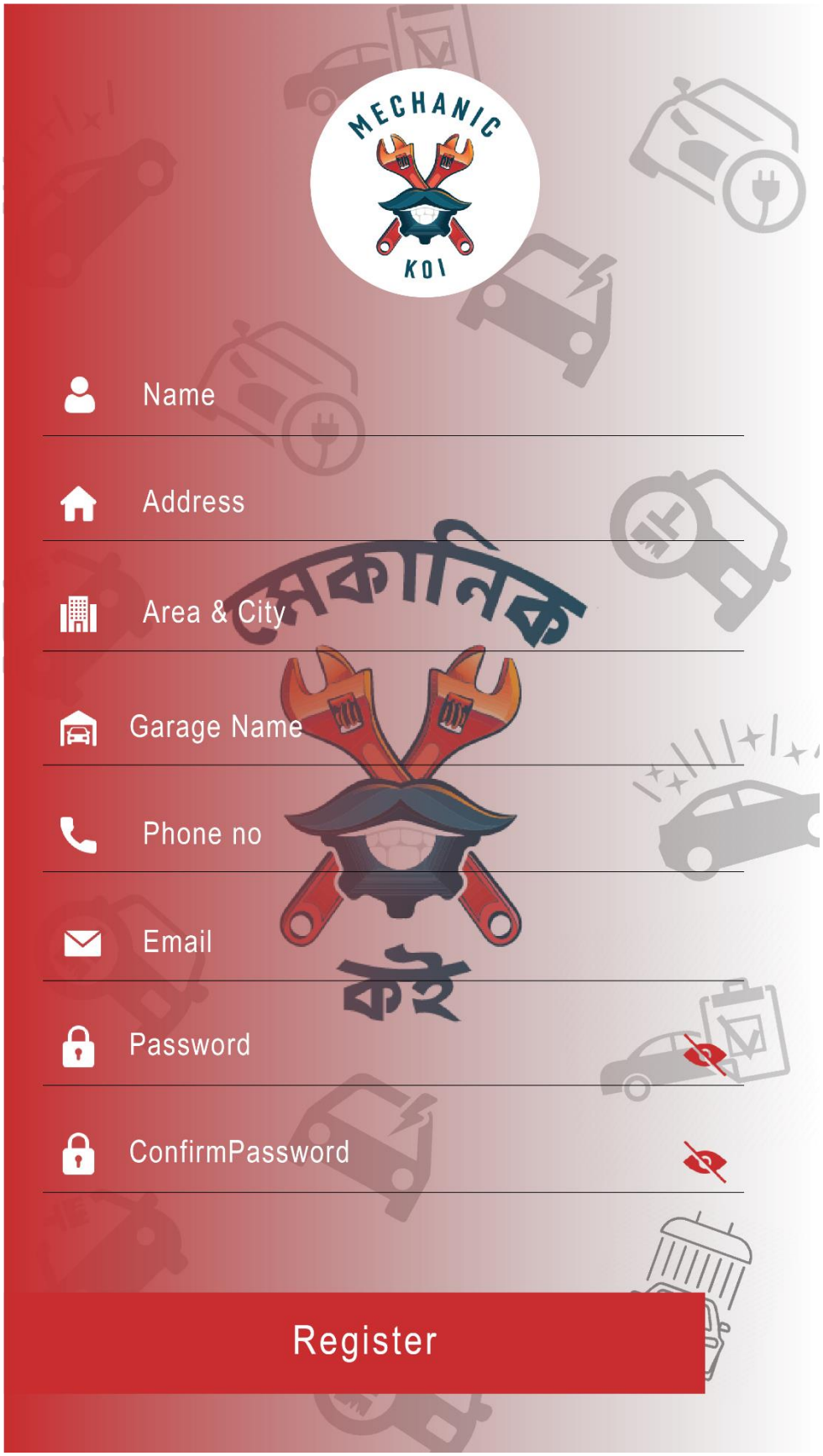
**Name**
Adittya Chakraborty

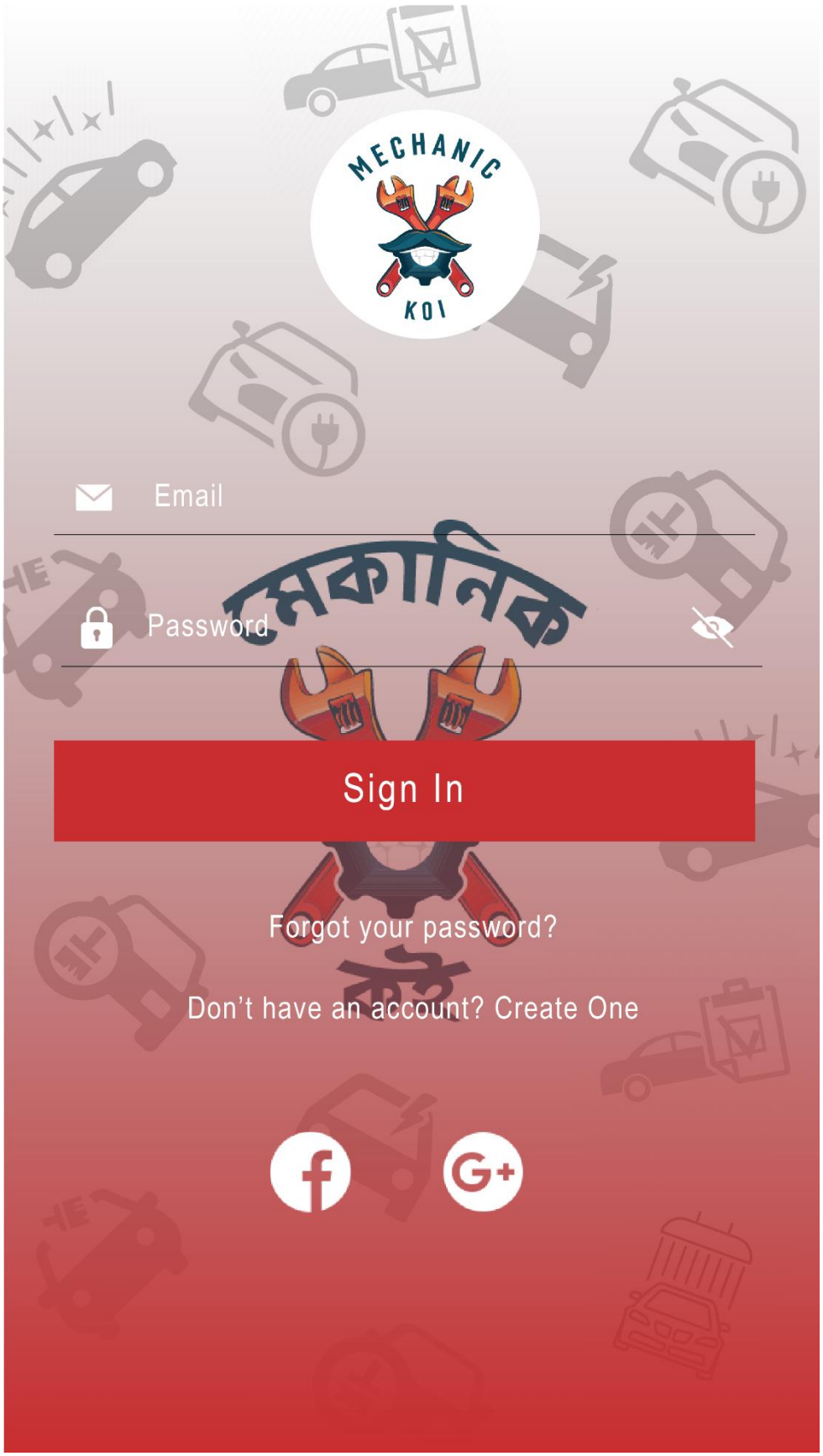
**Phone**
01751331694

**Email**
arup2140@gmail.com

**Address**
1373, Nagua, Kishoreganj, Dhaka

**Car Name & Model**
Allion, 2015


MECHANIC
KOI
Name
Address
Area & City
Garage Name
Phone no
Email
Password
ConfirmPassword
Register
মেকানিক
কই


MECHANIC
KOI
Email
মেকানিক
Password
Sign In
Forgot your password?
Don’t have an account? Create One

# Vehicle Papers

Name*

Email*

Phone No*

Appoinment Date*

mm/dd/yy

Preferred Time*

Choose

Select Services Needed*

Choose

Vehicle Brand*

Vehicle Model*

Vehicle Year*

Submit

## 3. SYSTEM TEST PLAN

- We are selecting mobile application for system test plan and have identified 4 modules of the system to test alone.
- Here are the test cases of our selected modules:

**Login session test plan:**

<table>
<tr><td colspan="3">Project Name: Mechanic Koi</td><td colspan="2">Test Designed by: Group-9</td></tr>
<tr><td colspan="3">Test Case ID: MK_1</td><td colspan="2">Test Designed date: 15.12.20</td></tr>
<tr><td colspan="3">Test Priority (Low, Medium, High): High</td><td colspan="2">Test Executed by: Adittya Chakraborty</td></tr>
<tr><td colspan="3">Module Name: Login Session</td><td colspan="2">Test Execution date: 17.12.20</td></tr>
<tr><td colspan="5">Test Title: verify login with valid username and password</td></tr>
<tr><td colspan="5">Description: Test application login</td></tr>
<tr><td colspan="5">precondition (If any): User must have valid username and password</td></tr>
<tr><td>Test Steps</td><td>Test Data</td><td>Expected Results</td><td>Actual Results</td><td>Status (Pass/Fail)</td></tr>
<tr><td>1. launch the application<br>2. Enter the username<br>3. Enter password<br>4. Click submit</td><td>Username:<br>12121219<br>Password:<br>789</td><td>User should log in to the application</td><td>As expected,</td><td>Pass</td></tr>
<tr><td colspan="5">Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.</td></tr>
</table>

**Vehicle repair session test plan:**

| | | | | |
|---|---|---|---|---|
| Project Name: Mechanic Koi | | | Test Designed by: Group-9 | |
| Test Case ID: MK_2 | | | Test Designed date: 15.12.20 | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Adittya Chakraborty | |
| Module Name: Vehicle repair session | | | Test Execution date: 17.12.20 | |
| Test Title: Verify an appointment system for vehicle repair. | | | | |
| Description: Test vehicle repair appointment. | | | | |
| precondition (If any): User need to be logged in the application and go to the service section. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
| 1.open app<br>2.log in-app<br>3.go to services<br>4. Go to the vehicle repair<br>5. Fill up the appointment requirements. | User name: aaaa<br>User number:000<br>User car details: kkk | User should be able to book an appointment perfectly. | As expected, | Pass |
| Post Condition: User need to book an appointment with the company and every data of the appointment related query is being stored in database. | | | | |

**Emergency call button session test plan:**

| | |
|---|---|
| Project Name: Mechanic Koi | Test Designed by: Group-9 |
| Test Case ID: MK_3 | Test Designed date: 15.12.20 |
| Test Priority (Low, Medium, High): High | Test Executed by: Adittya Chakraborty |
| Module Name: Emergency call button Session | Test Execution date: 17.12.20 |
| Test Title: Verify if emergency service call button is working properly. | |
| Description: To test the emergency service call button if it is working properly or not. | |
| Precondition (If any): User need to open the application and they can find the emergency service button there. | |

| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
|---|---|---|---|---|
| 1.Open app<br>2.Tap the button<br>3.Make a call from mobile phone | Checking the extended feature, no particular data | The user should be able to make a call while he or she presses the emergency call button. | As expected, | Pass |
| Post Condition: User is being redirected to the emergency service of that company and they will easily find the solution in no time. | | | | |

**Sign up session test plan:**

| | |
|---|---|
| Project Name: Mechanic Koi | Test Designed by: Group-9 |

| Test Case ID: MK_4 | | | Test Designed date: 15.12.20 | |
|---|---|---|---|---|
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Adittya Chakraborty | |
| Module Name: Sign up Session | | | Test Execution date: 17.12.20 | |
| Test Title: verify sign up with valid email and password | | | | |
| Description: Test application sign up | | | | |
| precondition (If any): User must have the application to sign up. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
| 1.Launch the application<br>2. Enter the email<br>3. Enter password<br>4.Enter user details<br>5. Click submit | Email:<br>abc@gmail.com<br>Password:<br>767<br>Details:<br>ghgh | User should sign up to the application | As expected, | Pass |
| Post Condition: User is validated with database and successfully signed up. The account details are stored in the database. | | | | |

**Spare parts shop test session:**

| | |
|---|---|
| Project Name: Mechanic Koi | Test Designed by: Group-9 |
| Test Case ID: MK_5 | Test Designed date: 15.12.20 |
| Test Priority (Low, Medium, High): Medium | Test Executed by: Adittya Chakraborty |
| Module Name: Available spare parts session | Test Execution date: 17.12.20 |

Test Title: Verify spare parts e-commerce section

Description: The spare parts shop is an e-commerce platform of that company.

Precondition (If any): Need to login to the account first.

| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
|---|---|---|---|---|
| 1.open app<br>2.log in-app<br>3.go to services<br>4. Go to Available spare parts shop<br>5. Pick required parts | User details:<br>abcd<br>Choose parts: 123<br>Price:<br>54698 | User should be able to buy anything perfectly. | As expected, | Pass |

Post Condition: User have to download the application and the device must support the application.

**Vehicle paper session test plan:**

| | | | | |
|---|---|---|---|---|
| Project Name: Mechanic Koi | | | Test Designed by: Group-9 | |
| Test Case ID: MK_6 | | | Test Designed date: 15.12.20 | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Adittya Chakraborty | |
| Module Name: Vehicle paper Session | | | Test Execution date: 17.12.20 | |
| Test Title: Verify if the customer can fill the details in order to get his/her vehicle paper by the company. | | | | |
| Description: The user will be able to get his/her vehicle paper with a hassle free service. | | | | |
| precondition (If any): Need to login to the account first. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
| 1.open app<br>2.log in-app<br>3.go to services<br>4. Go to the vehicle papers | User data: aaaa<br>Car details: aaaa | The user must be able to fill up all the required data. | As expected, | Pass |
| Post Condition: User have to download the application and the device must support the application. | | | | |

**Book a chauffeur test plan:**

| | |
|---|---|
| Project Name: Mechanic Koi | Test Designed by: Group-9 |

| Test Case ID: MK_7 | | | Test Designed date: 15.12.20 | |
|---|---|---|---|---|
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Adittya Chakraborty | |
| Module Name: Book a Chauffeur Session | | | Test Execution date: 17.12.20 | |
| Test Title: Verify if an user can book a chauffeur. | | | | |
| Description: The user can book any driver at any time if he/she uses the application. | | | | |
| Precondition (If any): Need to login to the account first. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
| 1.open app<br>2.log in-app<br>3.go to services<br>4. Go to the Book a Chauffeur | User address:<br>vvvv<br>Car details:<br>nnnn<br>Time:<br>00:00 | User should be able to book a chauffeur. | As expected, | Pass |
| Post Condition: User have to download the application and the device must support the application. | | | | |

**Menu button test plan:**

| | |
|---|---|
| Project Name: Mechanic Koi | Test Designed by: Group-9 |
| Test Case ID: MK_8 | Test Designed date: 15.12.20 |
| Test Priority (Low, Medium, High): Medium | Test Executed by: Adittya Chakraborty |
| Module Name: Menu button test | Test Execution date: 17.12.20 |

Test Title: Verify if an user can perfectly use the menu button.

Description: The user can check the menu and the available service options at any time if he/she uses the application.

Precondition (If any): Need to login to the account first.

| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
|---|---|---|---|---|
| 1.open app<br>2.log in-app<br>3.Press menu button<br>4. Select any service | It's only a test for checking the menu button. So we can't find any specific data there. | User should be able to see the service option whenever he/she presses the menu button. | As expected, | Pass |

Post Condition: User have to download the application and the device must support the application.

**Back button test plan:**

| | |
|---|---|
| Project Name: Mechanic Koi | Test Designed by: Group-9 |
| Test Case ID: MK_9 | Test Designed date: 15.12.20 |
| Test Priority (Low, Medium, High): Medium | Test Executed by: Adittya Chakraborty |
| Module Name: Back button test | Test Execution date: 17.12.20 |
| Test Title: Verify if an user can perfectly use the back button to get back to previous page. | |
| Description: The user can get back to the previous page whenever he/she presses the back button. | |
| Precondition (If any): Need to login to the account first. | |

| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
|---|---|---|---|---|
| 1.open app<br>2.log in-app<br>3. Go to any page after logging in<br>4.Press back button | It's only a test for checking the back button. So we can't find any specific data there. | User should be able to get back to the previous page. | As expected, | Pass |

| |
|---|
| Post Condition: User have to download the application and the device must support the application. |

**Log out test plan:**

| | | | | |
|---|---|---|---|---|
| Project Name: Mechanic Koi | | | Test Designed by: Group-9 | |
| Test Case ID: MK_10 | | | Test Designed date: 15.12.20 | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Adittya Chakraborty | |
| Module Name: Log out test | | | Test Execution date: 17.12.20 | |
| Test Title: Verify if an user can perfectly log out of the session. | | | | |
| Description: The user can log out from the session whenever he/she uses the log out button. | | | | |
| Precondition (If any): Need to login to the account first. | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | Status (Pass/Fail) |
| 1.open app<br>2.log in-app<br>3. Go to any page after logging in<br>4.Press back button<br>5. Go to the log out | It's only a test for logging out of the current session. So we can't find any | User should be able to log out of that session. | As expected, | Pass |

| | | | | |
|---|---|---|---|---|
| section<br><br>6. Press log out | specific<br><br>data there. | | | |
| Post Condition: User have to download the application and the device must support the application. | | | | |

**Text Format:**

- Font Style: Times New Roman
- Font Size: 12
- Line space: 1
- Alignment: Justify
- Report Length: Maximum 10 pages