



**INSTITUTE FOR ADVANCED COMPUTING  
AND SOFTWARE DEVELOPMENT (IACSD),  
AKURDI, PUNE**

Documentation On

**Roadmate**

**(Making the trip easy and adventurous)**

PG-DAC March 2023

Submitted By:

Group No: 100

Roll No.	Name:
233218	Sourabh Tanaji Deshmukh
233125	Akanksha Parihar

Mrs. Monika Ma'am

Mrs. Harshal Sir

Project Guide

Mr. Rohit Puranik

Centre Coordinator

## ABSTRACT

Roadmate is a project to provide necessary functionalities to the customers to find and select their travel companions. This project is basically divided into 3 modules and can be titled as front end, back end and database. Also, Spring Security is used to provide authentication and authorization with the objective to improve data security.

This web project has applied standard 'SOLID Principles' by maintaining Separation Of Concern and Interface Segregation concepts. To attract more customers, the proposed application also provides attractive Graphical User Interface (GUI) implemented by ReactJS Libraries. As the project involves structured data and fixed schemas, MySQL is used as the database, which is quite preferred for its efficient queries & access.

## ACKNOWLEDGEMENT

I take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. I extend my sincere and heartfelt thanks to our esteemed guide, Mrs. Monika Ma'am and Mrs. Harshal Sir for providing me with the right guidance and advice at the crucial juncture and for showing me the right way. I extend my sincere thanks to our respected Centre Co-Ordinator Mr. Rohit Puranik, for allowing us to use the facilities available. I would like to thank the other faculty members also, at this occasion. Last but not the least, I would like to thank my friends and family for the support and encouragement they have given me during the course of our work.

Sourabh Tanaji Deshmukh (223218)

Akanksha Parihar (223096)

## Table of Contents

ABSTRACT.....	02
ACKNOWLEDGEMENT .....	03
INTRODUCTION .....	06
1 FEATURES .....	07
1.1 PROJECT OBJECTIVE .....	07
1.2 PROJECT OVERVIEW .....	07
1.3 PROJECT SCOPE .....	07
1.4 STUDY OF THE SYSTEM .....	08
1.4.1 MODULES .....	08
2.....	SYETEM
ANALYSIS .....	09
2.1 PROPOSED SYSTEM .....	09
2.2 SYSTEM REQUIREMENT SPECIFICATION .....	10
2.2.1 FUNCTIONAL REQUIREMENTS.....	10
2.2.2 NON FUNCTIONAL REQUIREMENTS .....	13
3 DIAGRAMS	
3.1 USE CASE DIAGRAM .....	16
3.2 ER DIAGRAM .....	18
3.3 CLASS DIAGRAM .....	20
3.4 ACTIVITY DIAGRAM .....	21
3.5 DATA FLOW DIAGRAM .....	22
3.6 DEPLOYMENT DIAGRAM .....	23
3.7 SEQUENCE DIAGRAM .....	23
4 TABLE STRUCTURE .....	24
5 PROJECT DIAGRAMS .....	29
6 CONCLUSION.....	38
7 REFERENCES .....	39

**LIST OF FIGURES**

FIGURE 1: UML USE CASE DIAGRAM -----11

FIGURE 2: E-R DIAGRAM -----13

FIGURE 3: CLASS DIAGRAM ----- 15

FIGURE 4: ACTIVITY DIAGRAM ----- 23

FIGURE 5: 0 LEVEL DATA FLOW DIAGRAM -----24

FIGURE 6: 1 LEVEL DATA FLOW DIAGRAM -----25

FIGURE 7: DEPLOYMENT DIAGRAM ----- 26

FIGURE 8: SEQUENCE DIAGRAM -----27

## INTRODUCTION

Nowadays, the excitement of the customers for clothes shopping is getting crushed by crowdie malls, the long lines involved in the manual process of payments & what not. This is why many customers are opting to enjoy shopping from home instead of shopping at malls, shops etc.

This document includes Software Requirements Specification which is built to describe the agreement between the customer and the developer regarding the specification of the software requested for 'Roadmate Website'. Its primary purpose is to provide a clear and descriptive 'Statement of User Requirements' that can be used as a reference in further development of the software system. This document is broken into a number of sections used to logically separate the whole content for the ease of reference. This Software Requirements Specification aims to describe the Assumptions, Constraints, Scope of Software to be developed, Functional Requirements, Non-Functional Requirements, various diagrams used while software development related to software described throughout the rest of the document.

This SRS describes, in clear terms, the software's primary uses and required functionality needed to general customer. This project of developing a hazel –free online portal for finding the travel companion by providing separate portals for both riders & Admins.

# **1 FEATURES**

## **1.1 PROJECT OBJECTIVE**

Many people find bike riding as their hobby so Roadmate is an application which will help the riders to get along with all other riders on the same route. As travelling alone can be risky and boring so by getting a group of people who are also going for bike ride on the same route can be a good way of communication with strangers and continue to explore new places in an effective manner. By using this application, rider can plan for a trip by mentioning the duration so that he/she can plan accordingly. This application will also help the new riders to get the feedbacks of the experienced riders who had already went through that route in past days. The application also provides facilities like hotels, restaurants available so that the rider can search for the hotels to stay and rest for some time. This application is a one stop solution which will provide a rider all the necessities that he/she may require in his road trip.

## **1.2 PROJECT SCOPE**

This application allows riders to search for a company by logging into the website and provides the details regarding the journey. One can also register himself/herself so that anyone who is willing to go for a ride can also join him. The application also provides the riders with some facilities like hotels, restrooms, and restaurants available in the way so that if they want to rest for some time then they can search for the available options.

## **1.3 STUDY OF THE SYSTEM**

### **1.3.1 MODULES:**

The system after careful analysis has been identified to be presented with the following modules and roles.

The modules involved are:

➤ Rider:-

The rider can browse through the various services available on the website. The authenticated (valid) Rider can add their planned trip details .Rider can also browse to find another rider available on a particular route.

➤ Admin :-

The admin have the authority of deleting any rider from the portal. He/she also can delete the reviews given by the riders if needed.

## 2 SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, diagnosing problems, and using the information to recommend improvements on the system. System analysis is a problem-solving activity that requires intensive communication between the system users and system developers.

System analysis or study is an important phase of any system development process. The system is viewed as a whole, the inputs are identified, and the system is subjected to close study to identify the problem areas. The solutions are given as a proposal. The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.



## 2.1 PROPOSED SYSTEM

The proposed web application provides following functionalities in order to attract more customers for buying clothes online.

- ✓ The whole web application is hassle-free and provided best GUI.
- ✓ Security of the data is maintained properly by using in built Spring Security.
- ✓ Conditional Rendering is also used in order to achieve selective response generation.
- ✓ As it's an online web application and available for 24 x 7, customers may enjoy the shopping without any time constraints.

## 2.2 SYSTEM REQUIREMENT SPECIFICATION

### 2.2.1 FUNCTIONAL REQUIREMENTS

**Registration:**

Users would be able to register by providing their email, username, and password.

**Login Credentials:**

Users would be able to log in using their registered email and password.

**Adding trips:**

After Login the user can add the trip details like source, destination and date of trip.

**Searching Riders:**

The riders can search for any other person using the same route in order to have company.

**Services:**

User who is travelling alone or in group may need assistance during the journey. So in the Roadmate there are several services like hotels, garages, trip planners etc that are available so that the user can take the help of the map or guide to get the distance and approx. time needed to cover the distance. There are also some emergency contact that a person need during the trip.

**Feedbacks:**

The riders who have gone on a road trip can share their experience so that the new rider can access it before planning for the trip.

**2.2.2 NON - FUNCTIONAL REQUIREMENTS**

Roadmate website will provide the following non – functional requirements:-

i. Interface :-

- ~ Roadmate must provide user interactive interface in order to attract more users.
- ~ The application should use best available attractive colour shade combinations.

ii. Performance:-

- ~ Number of Concurrent Users: - The application must handle maximum number of requests.

iii. Security:-

- ~ The Roadmate website must provide maximum level of security regarding data.
- ~ The data of the users, product details, valuable feedbacks, login credentials must be protected in order to maintain high customer satisfaction.
- ~ The application must provide separation via Authorization & Authentication.

iv. Availability:-

- ~ Roadmate must be available 24 X 7 i.e. throughout the day & night, so that Riders can browse and use the application all the time.

v. Reliability:-

- ~ The specified application must be reliable, especially at the time of weekend, festival days, year endings etc.
- ~ The application must be reliable in the perspective of login / payment failures also.

vi.       **Safety:-**

- ~ The online application must be saved against session fixations / SQL injection etc. malicious attacks.
- ~ The whole software must use firewall configurations in order to safeguard the application.

vii.       **Maintainability:-**

- ~ The Roadmate website should be able to maintain with as little efforts & changes as possible.

viii.       **Portability:-**

- ~ The specified application must provide portability in order to change components of architecture in case of emergencies.
- ~ It should have – free facility to replace the databases to enhance the efficiency in needed in future. Like replacement from MYSQL to Oracle or MYSQL to MongoDB.

ix.        **Accessibility:-**

- ~ The online website must be accessible via desktops, laptops, smart devices including mobile phones, tablets etc.
- ~ The UI – UX must not hamper in case of above options. It should remain uniform throughout all the devices.

x.         **Durability:-**

- ~ The overall application should be durable, especially in the terms of data, and uniform performance over time.

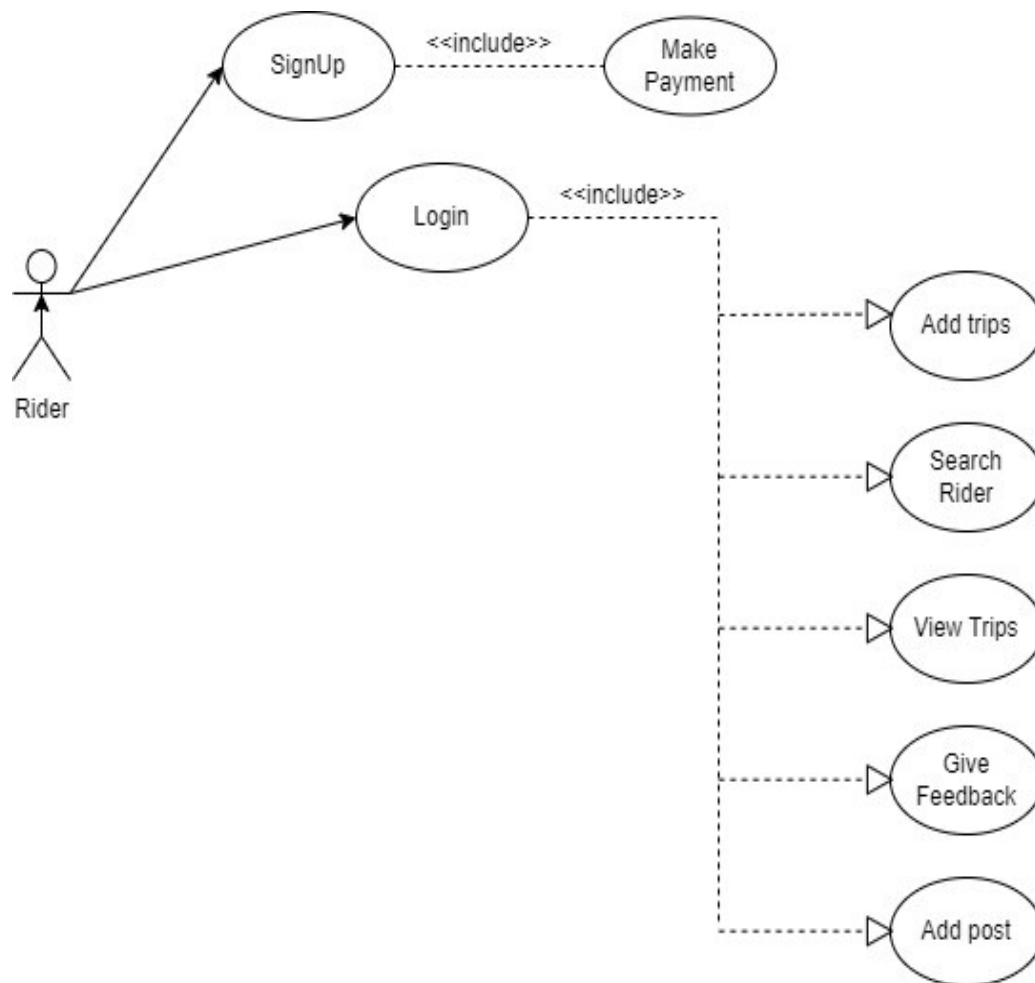
xi.        **Other Requirements:-**

- **Hardware:** - The application is expected to function on Dell G3 – 15 with 1100 MHz Pre Processor Equivalent Or Above, 4 GB RAM, 512 GB HDD.
- **Software:** - The Roadmate website shall work on Microsoft Windows operating systems family (MS Windows XP &

Above). It configures to work with MYSQL database. This System works on Apache Tomcat server. It uses browser Google Chrome Browser.

### 3 DIAGRAMS

#### 3.1 USE CASE DIAGRAM:-



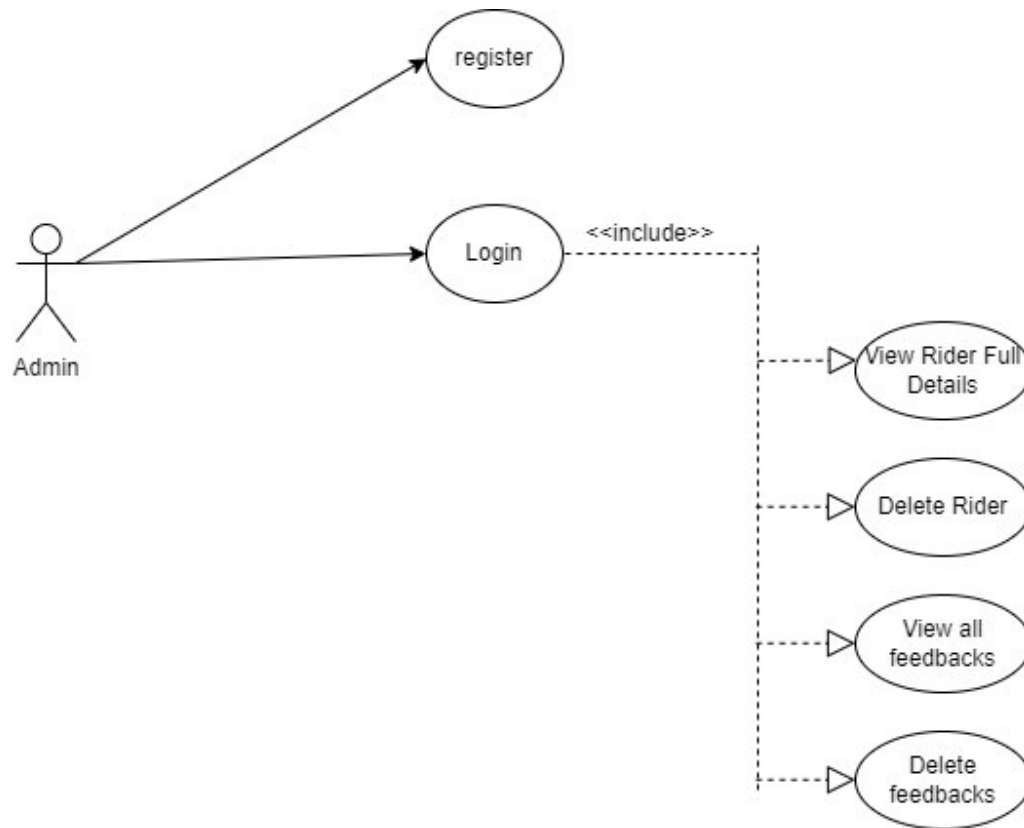


Fig. 1 : UML Use Case Diagram

### 3.2. E-R DIAGRAM:-

Entity Relationship Diagram is used to define the data elements and relationship for a specified application. It develops a conceptual design for the database. It also develops a very simple and easy to design view of the data.

In Entity Relationship Diagram, the data is represented by using various components including entities, attributes, relationships (One To Many / Many To Many etc.)

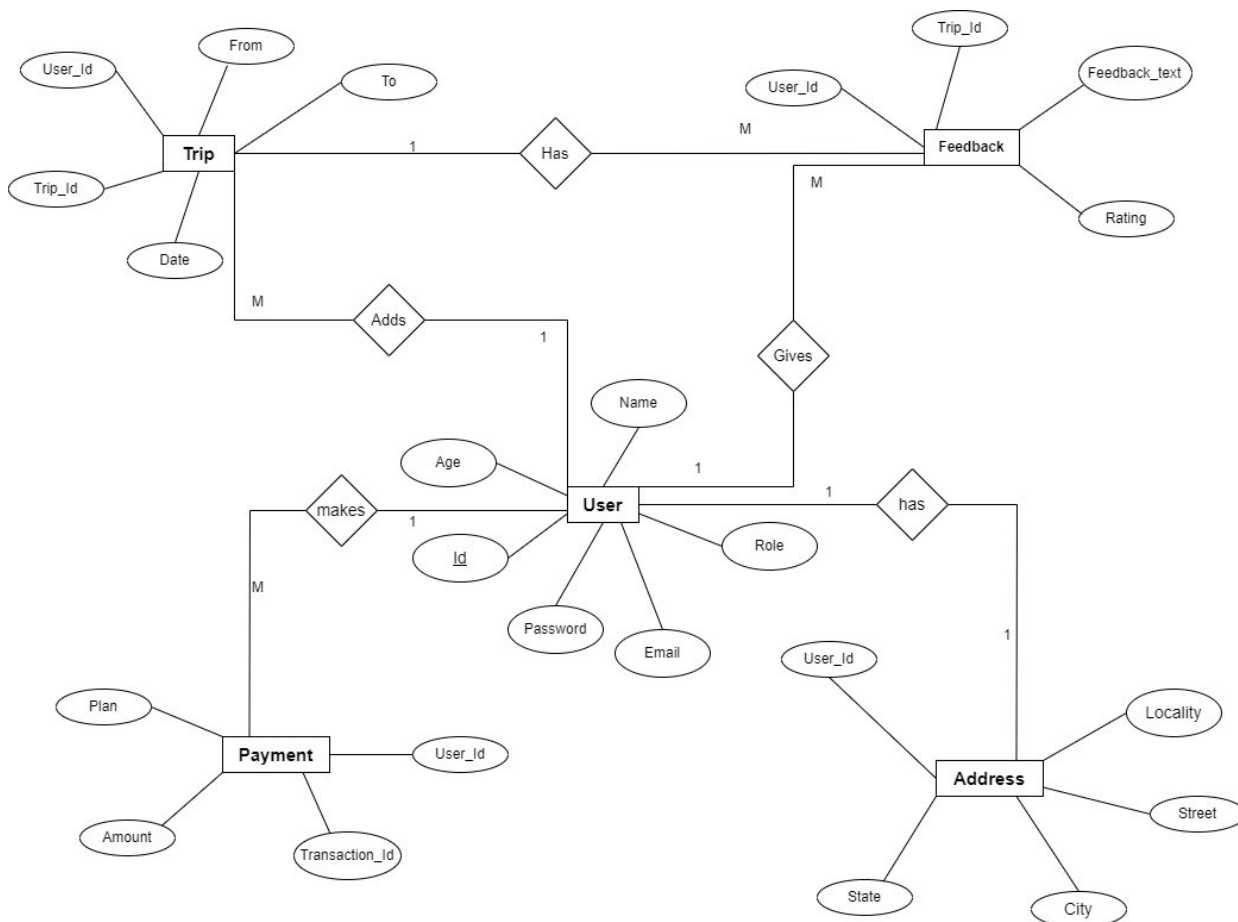


Fig. 2 : ER Diagram

### 3.3. CLASS DIAGRAM:-

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram. The purpose of the class diagram can be summarized as:-

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

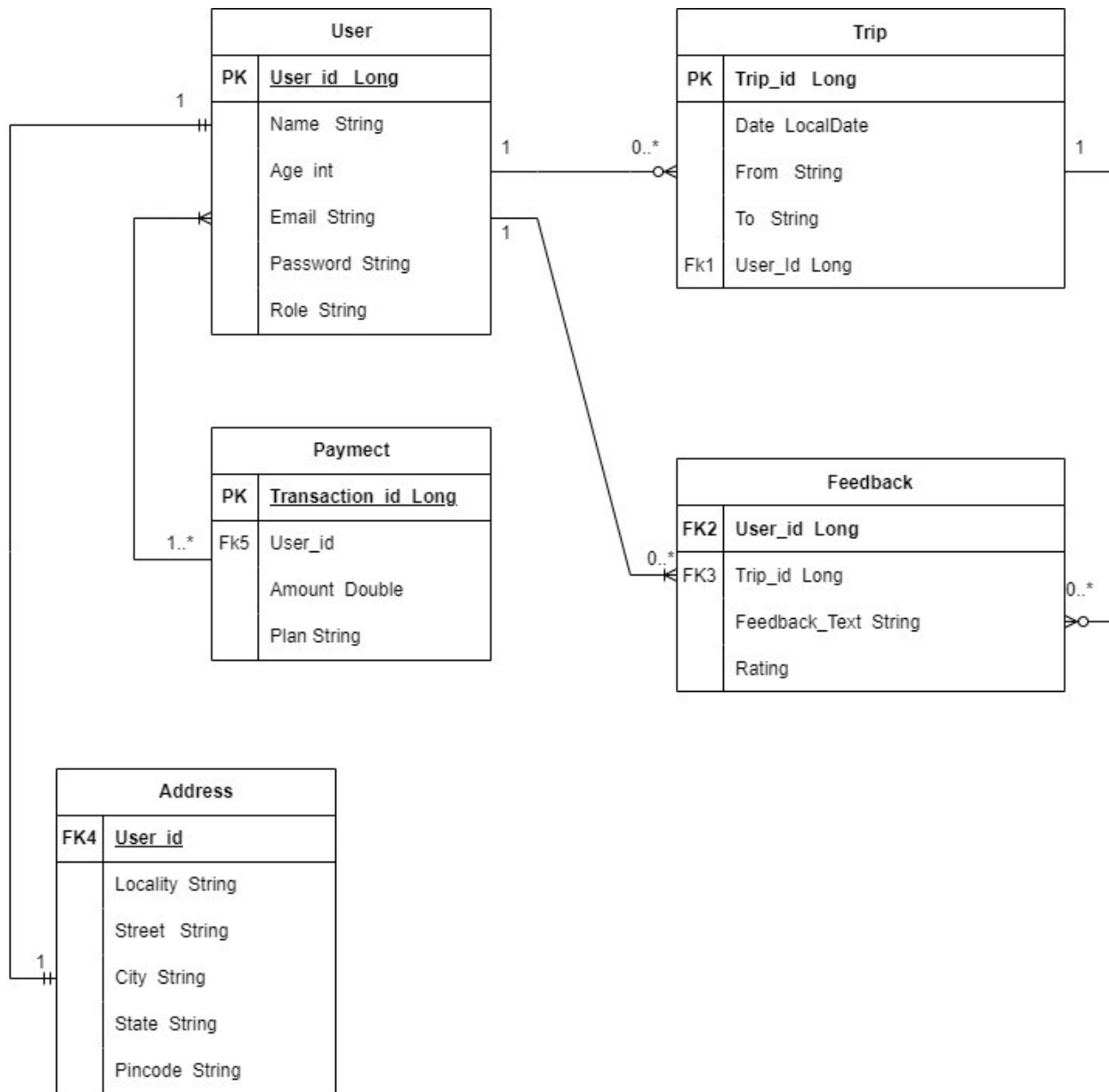


Fig. 3 : Class Diagram

### 3.4. ACTIVITY DIAGRAM:-



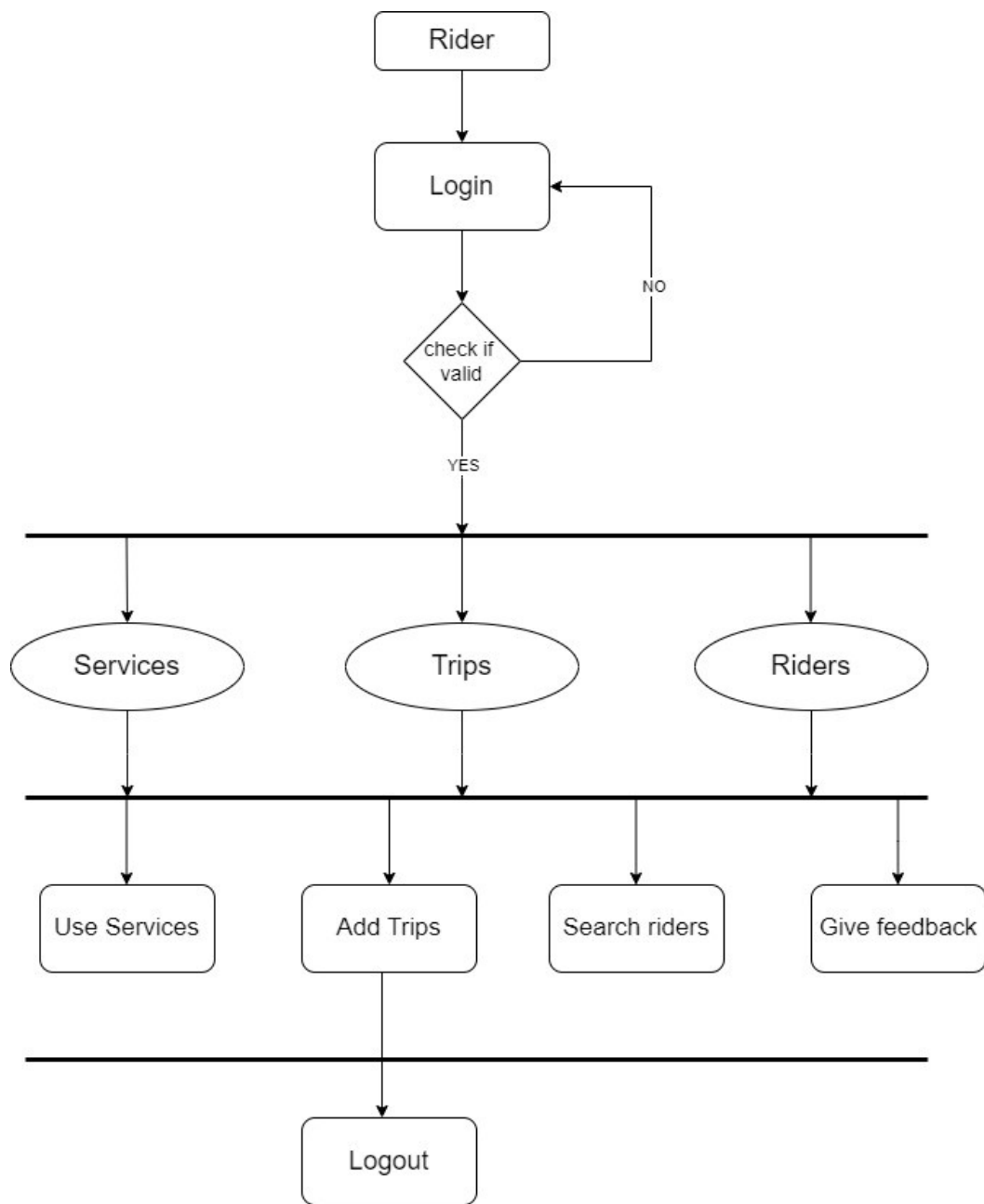


Fig. 4 : Activity Diagram

### 3.5. DATA FLOW DIAGRAMS:-

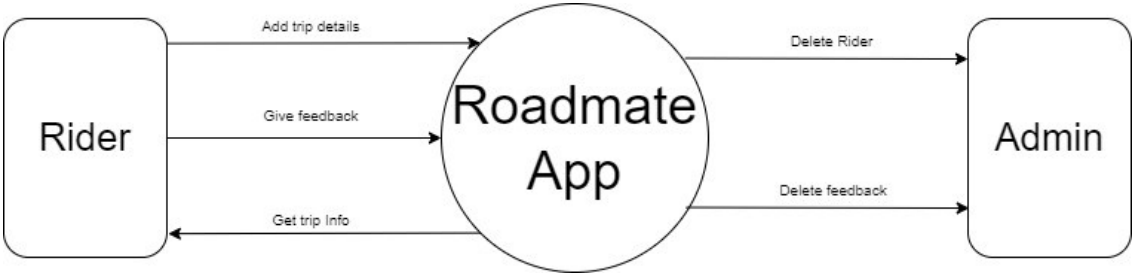


Fig. 5: Zero Level Data Flow Diagram

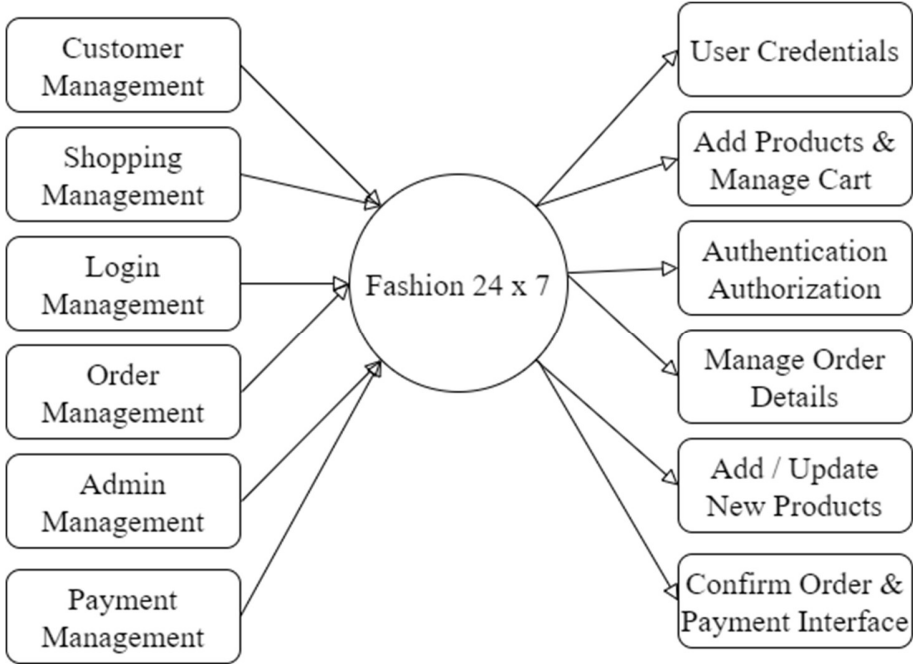


Fig. 6: First Level Data Flow Diagram

3.5. SEQUENCE DIAGRAM:

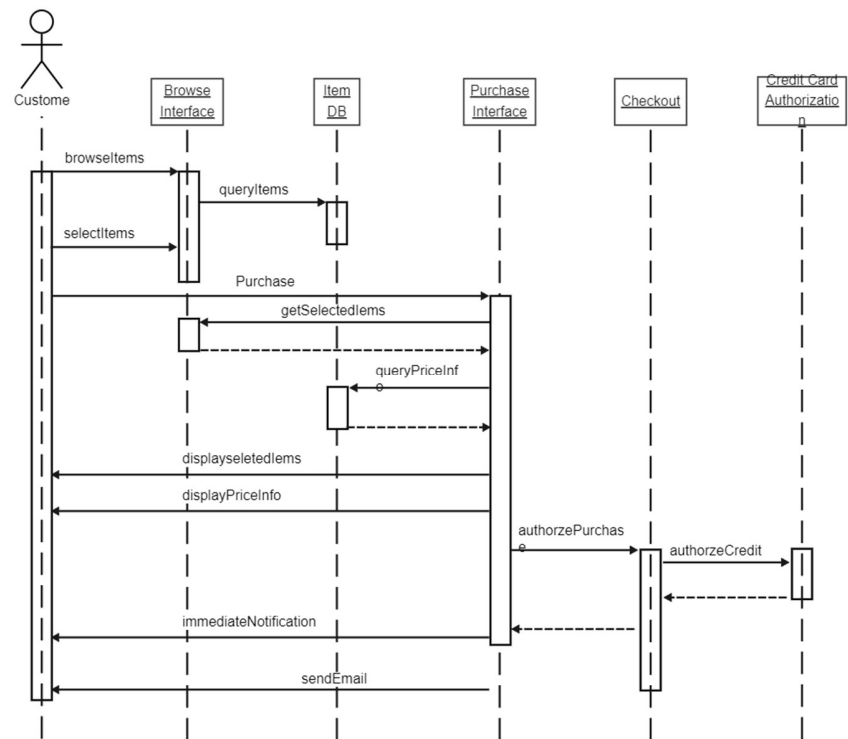


Fig. 8: Sequence Diagram

4 TABLE STRUCTURES

Roadmate Application generate following tables in the database:-

```
mysql> desc riders;
```

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
name	varchar(255)	YES		NULL	
age	int	YES		NULL	
email	varchar(255)	YES		NULL	
password	varchar(255)	YES		NULL	
role	varchar(255)	YES		NULL	
a_id	bigint	YES	MUL	NULL	

```
mysql> desc trips;
```

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
date	varchar(255)	YES		NULL	
from_location	varchar(255)	YES		NULL	
to_location	varchar(255)	YES		NULL	
rider_id	bigint	YES	MUL	NULL	

```
mysql> desc address;
```

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
city	varchar(15)	YES		NULL	
state	varchar(15)	YES		NULL	
locality	varchar(255)	YES		NULL	
street	varchar(30)	YES		NULL	

```
mysql> desc feedback;
```

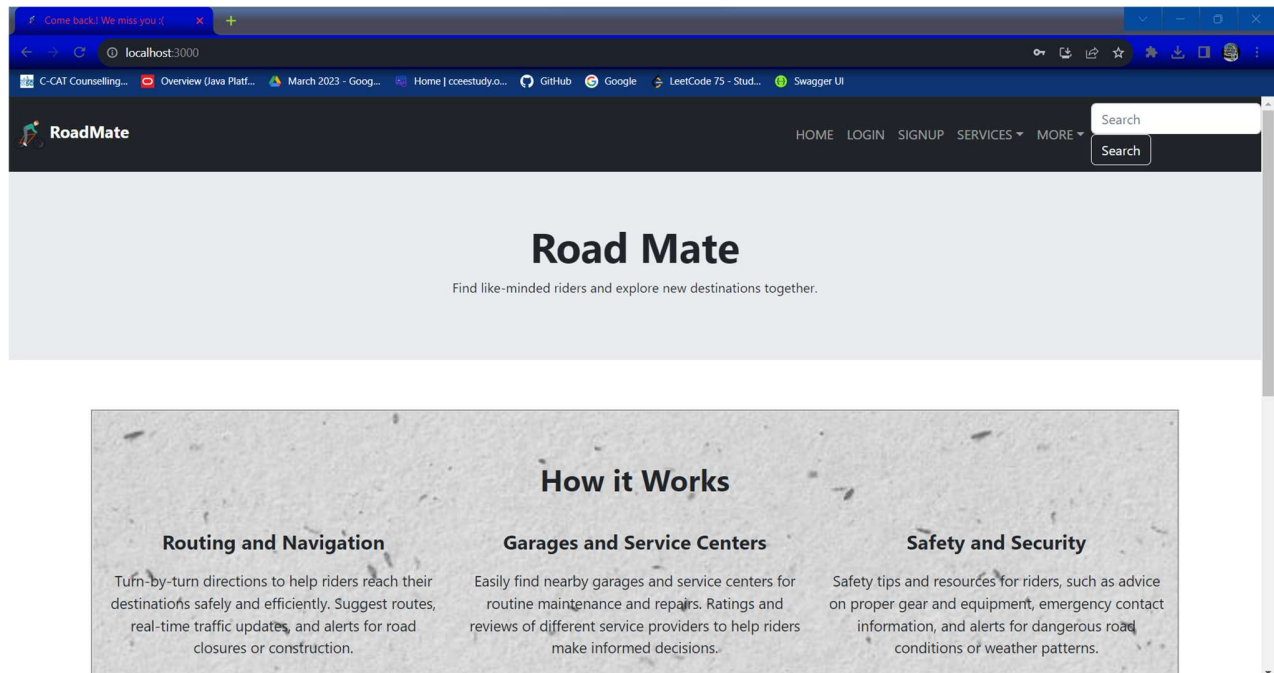
Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
text	varchar(255)	YES		NULL	
rider_id	bigint	YES	MUL	NULL	
trip_id	bigint	YES	MUL	NULL	

```
mysql> desc post;
```

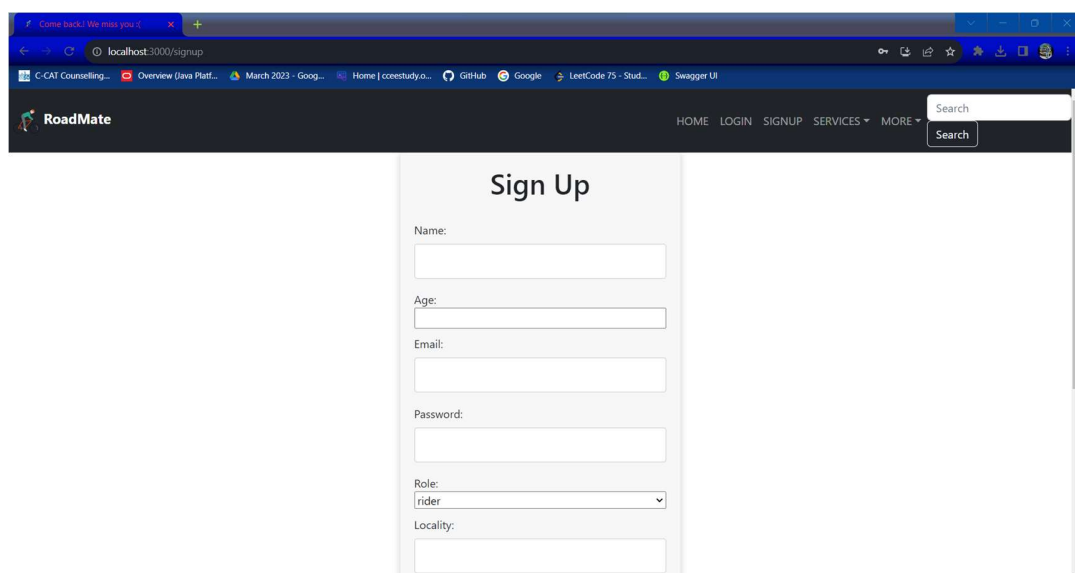
Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	
experience	varchar(255)	YES		NULL	
name	varchar(255)	YES		NULL	

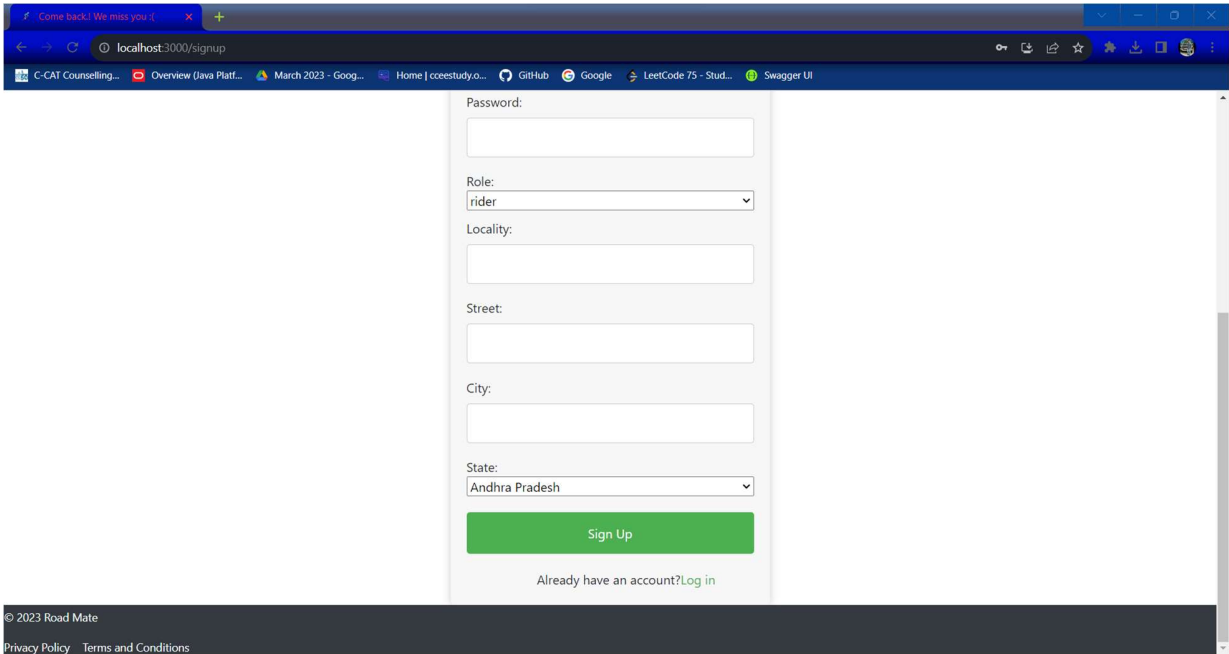
## 5 PROJECT DIAGRAMS

- Home Page

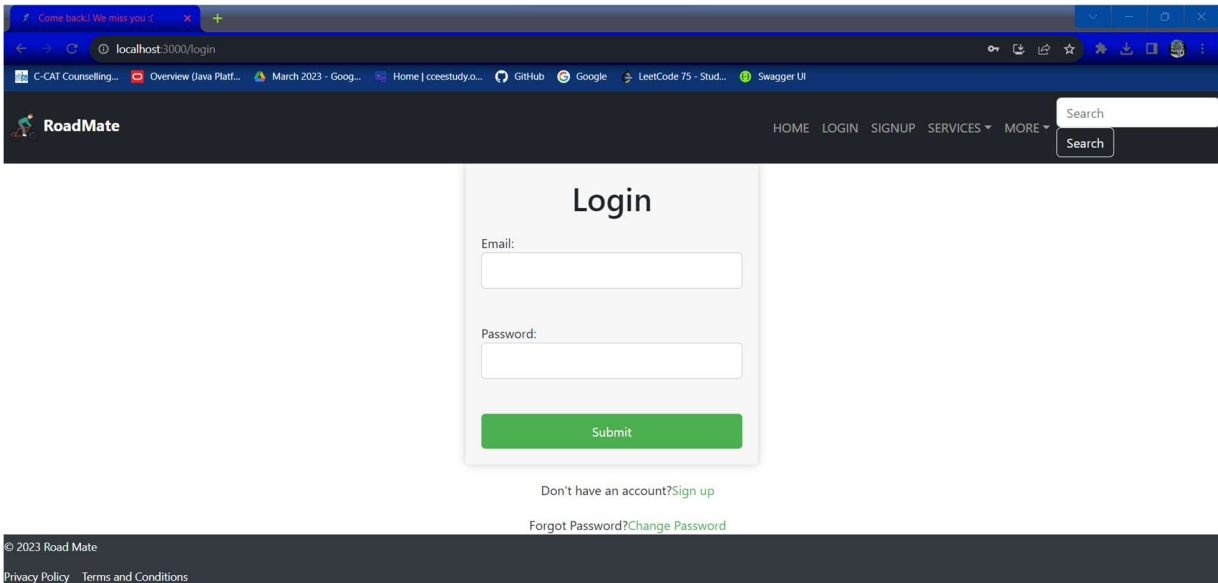


- Sign up:-

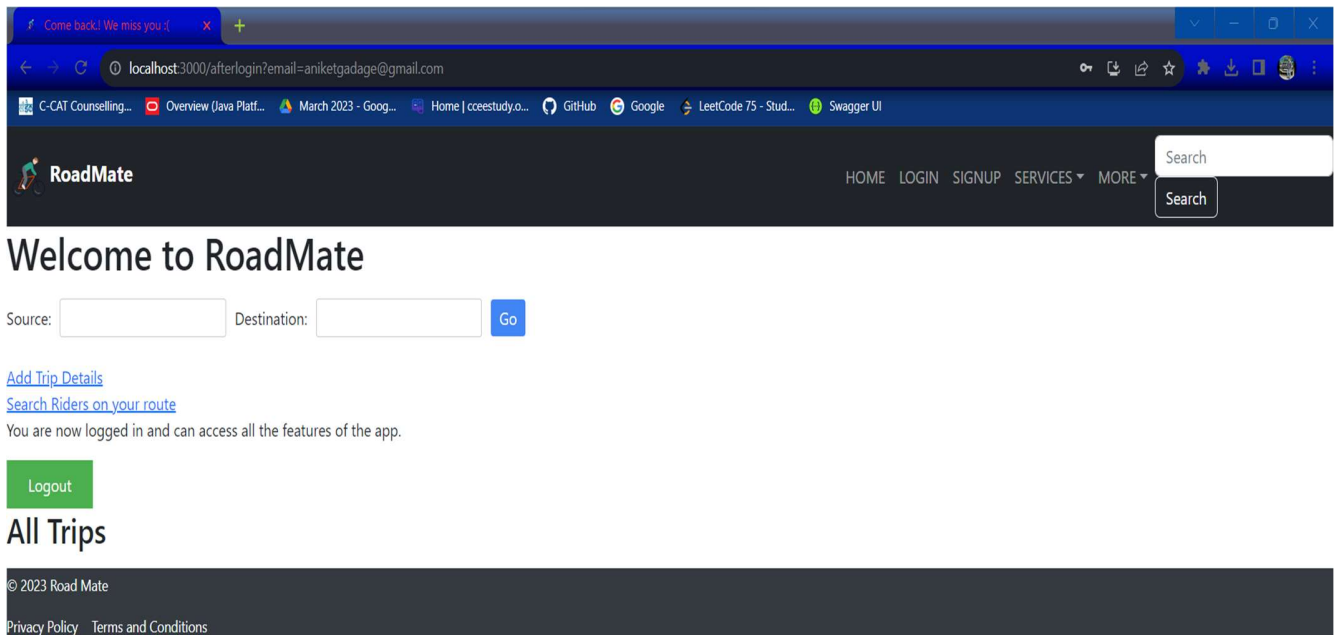
A screenshot of a web browser displaying the RoadMate sign-up page. The browser's address bar shows 'localhost:3000/signup'. The page has a dark blue header with the 'RoadMate' logo on the left and navigation links (HOME, LOGIN, SIGNUP, SERVICES, MORE) on the right. A search bar is also present in the header. The main content area has a light gray background with the heading 'Sign Up'. Below the heading, there is a form with the following fields: 'Name:' (text input), 'Age:' (text input), 'Email:' (text input), 'Password:' (text input), 'Role:' (dropdown menu with 'rider' selected), and 'Locality:' (text input).



- Sign in:-

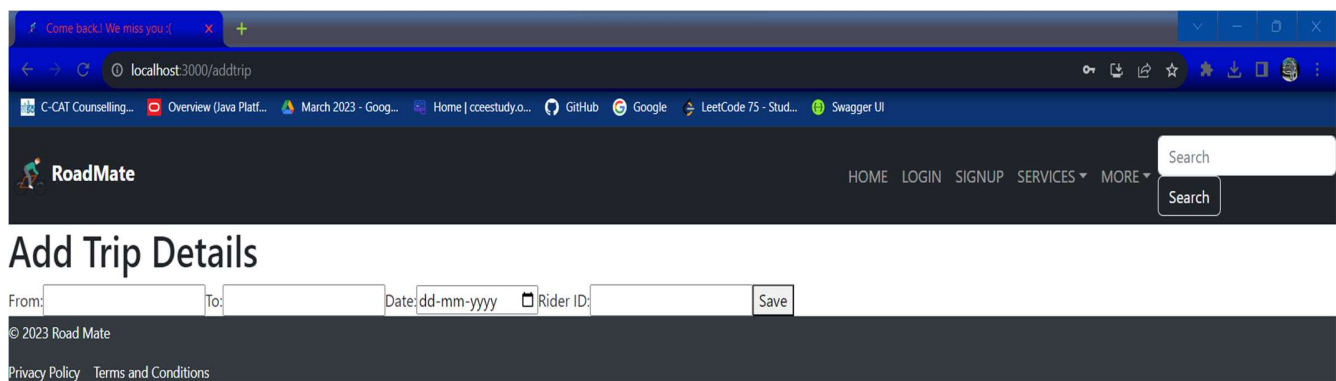


- After Login:-



The screenshot shows the RoadMate web application interface after a successful login. The browser address bar displays the URL `localhost:3000/afterlogin?email=aniketgadage@gmail.com`. The application header includes the RoadMate logo, navigation links (HOME, LOGIN, SIGNUP, SERVICES, MORE), and a search bar. The main content area features a large heading "Welcome to RoadMate" and a form with "Source:" and "Destination:" input fields, followed by a "Go" button. Below the form are links for "Add Trip Details" and "Search Riders on your route", and a message stating "You are now logged in and can access all the features of the app." A green "Logout" button is also present. The footer contains copyright information for 2023 Road Mate and links to "Privacy Policy" and "Terms and Conditions".

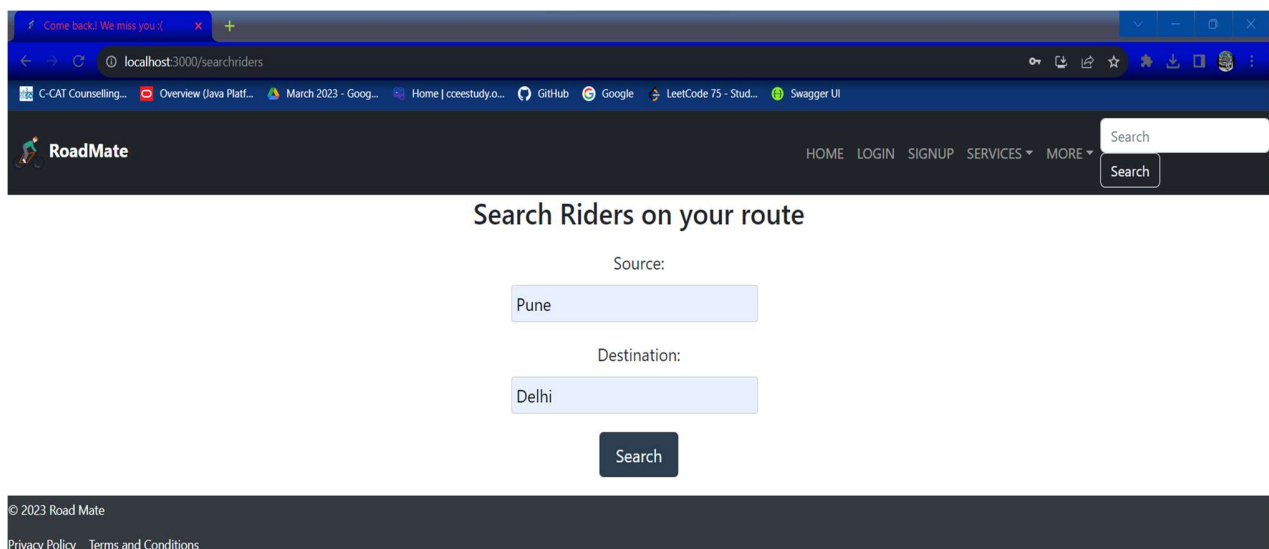
- Add trip details:-



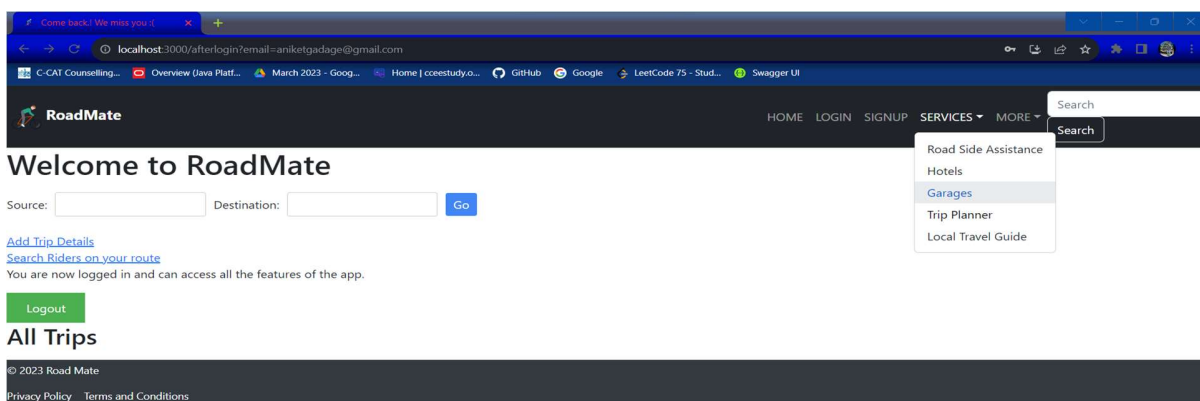
The screenshot shows the "Add Trip Details" page of the RoadMate web application. The browser address bar displays the URL `localhost:3000/addtrip`. The application header is consistent with the previous screenshot. The main content area features a heading "Add Trip Details" and a form with "From:" and "To:" input fields, a "Date:" field with a date picker (showing dd-mm-yyyy), a "Rider ID:" input field, and a "Save" button. The footer contains copyright information for 2023 Road Mate and links to "Privacy Policy" and "Terms and Conditions".

- Search riders on your route:-

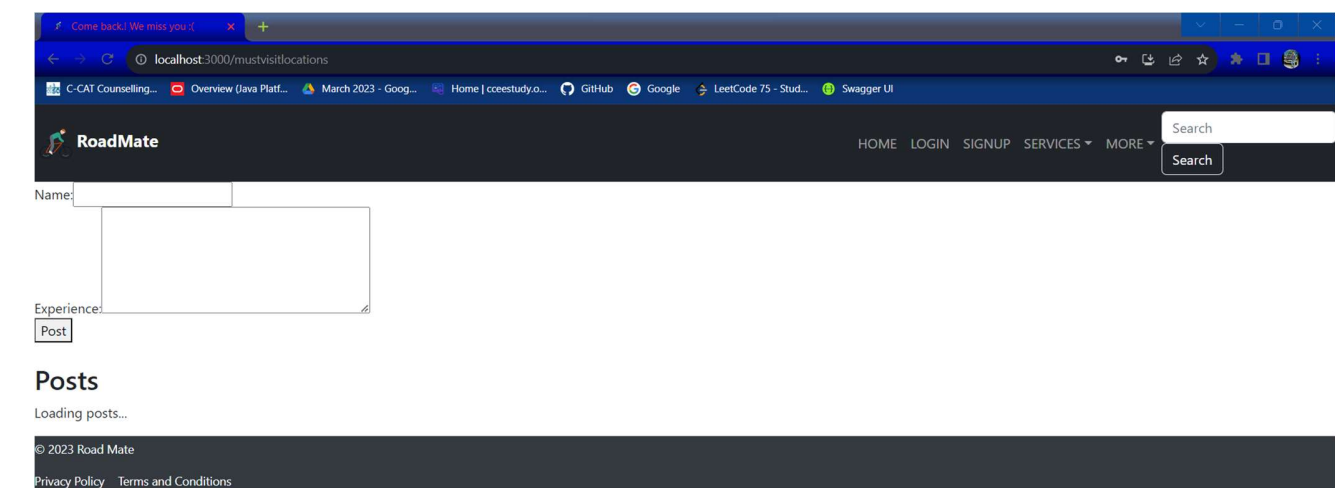




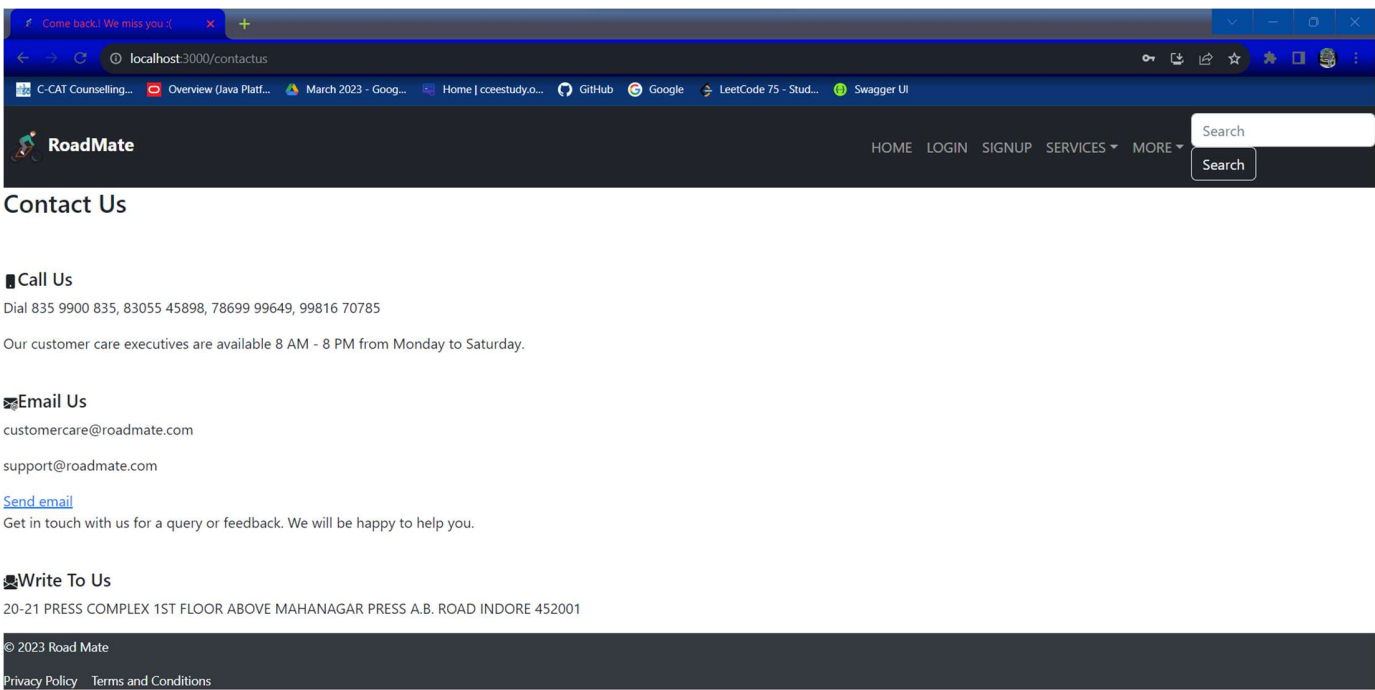
- Services:-



- Add post :-



- Contact us:-



REFERENCE

Following references are considered throughout the development of Roadmate Application:-

- <https://www.tutorialspoint.com/java/>
- <http://www.javatpoint.com/java-tutorial>
- <https://docs.oracle.com/javase/tutorial/>
- <http://www.tutorialspoint.com/mysql/>
- <https://stackoverflow.com/>





## **CLOSURE REPORT**

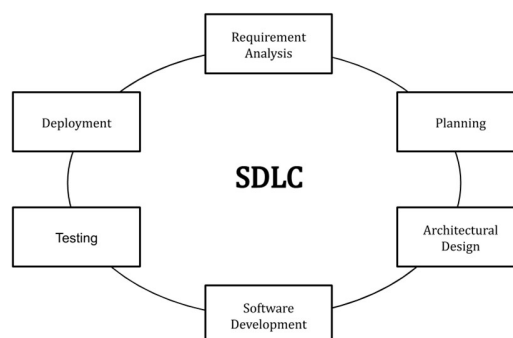
### **1. GENERAL INFORMATION**

The general information about the roadmate is as follows:-

#### **1.1. PRODUCTIVITY:-**

- As per the requirements gathered from the customer during initial phase of the software development, the functionalities like CRUD operations for Roadmate app has been added successfully. That is a registered customer can add, remove his scheduled trips from the list and also can give feedback to his previous travelled trips.
- Admin can delete a rider from the database. Admin can also delete a particular review.
- The new i.e. unregistered customers may browse through added services even without official registration. This functionality is useful to achieve non functional requirement of the website, which is to provide hassle-free use of the website as far as customers are concerned.
- This proposed website of Roadmate application can be accessed by any latest version of Google Chrome through smart devices as well as personal computers, laptops.
- Spring Security concept is used effectively to secure data by including 'Authentication, Authorization'.

#### **1.2. PROCESS USED & DEVIATION:-**



Roadmate app has been successfully implemented the SDLC Lifecycle in which it went through different phases, which are described as follows:-

1. Requirement Analysis: - Basic requirements were gathered from the customer and analyzed in a proper way so as to set aim, objectives, and time constraints of the proposed project.

2. Planning:- To meet the specified requirements given by the customer, planning phase is conducted in which the various technologies were taken into consideration for the online clothes shopping website project.
  3. Architectural Design: - In this phase, basic flow of the whole project was finalized. Also, included the necessary security measures in the architectural design of the project.
  4. Software Development: - In this phase of software development, actual code was written in selected programming languages and finalized frameworks. Roadmate has been successfully implemented by using Spring Boot Dynamic Web Project (MVC) and ReactJS as a front end of the project.
  5. Testing: - Agile Methodology is used for increasing efficiency of the proposed project. In which, testing is done in parallel with software development. Also, the concept of 'V-Model' is used here effectively, which involves Verification conducted by the developer itself so as to mitigate the defects at earlier stages of the application development.
  6. Deployment: - The completed project has been deployed on the EC2 Instance hosted by Amazon Web Services (AWS).
- During the whole process of execution, as some of the requirements were added by the customer in later phases of the Software Development Lifecycle. These changes were effectively considered and revised the structure of the proposed project by adding necessary functionalities. These additions improvised the SRS as per the specified new added requirements.

### **1.3. ESTIMATED & ACTUAL START – END DATES OF THE PROJECT:-**

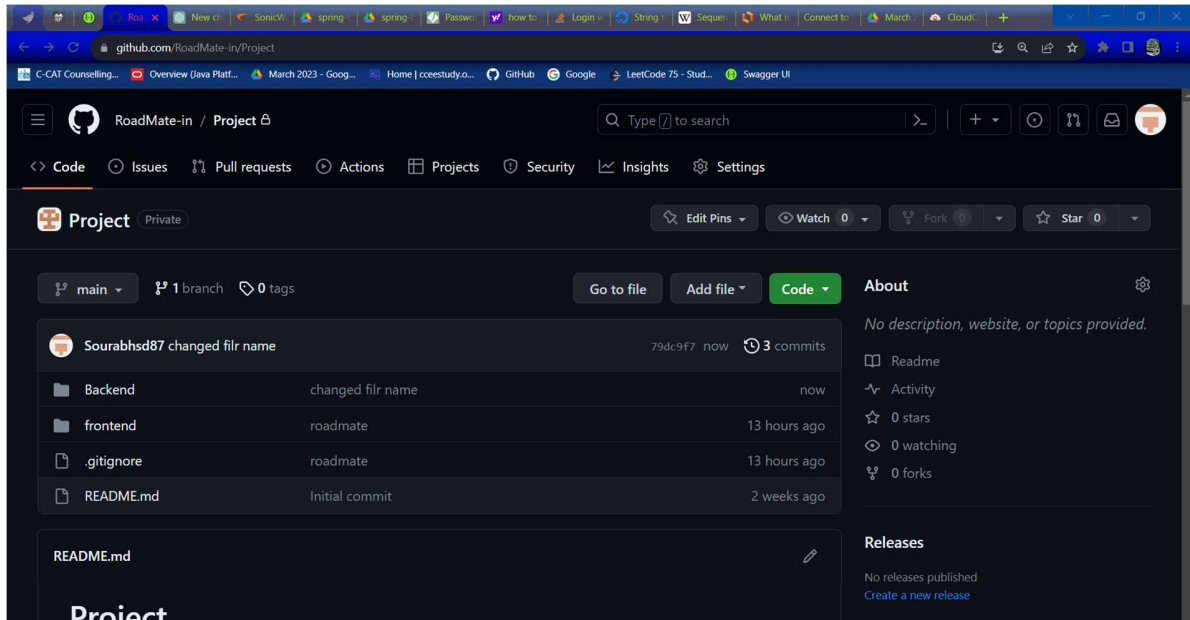
- Estimated Dates:-
  - Start of the Software Development: - 1<sup>st</sup> August, 2023
  - End of the Software Development: - 25<sup>st</sup> August, 2023
- Actual Dates:-
  - Start of the Software Development: - 11<sup>th</sup> August, 2023
  - End of the Software Development: - 31<sup>st</sup> August, 2023

### **1.4. TOOLS USED:-**

Following are the tools used effectively during the software development of Roadmate

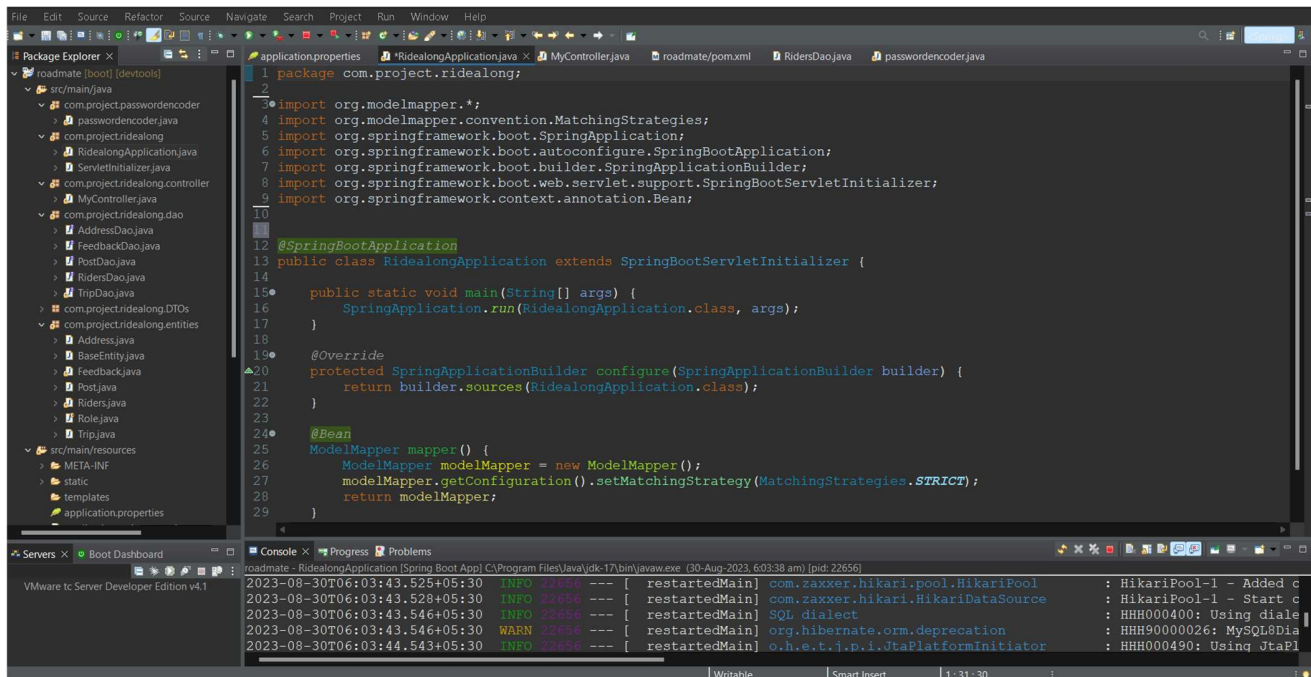
App:-

- GitHub: - This tool is used to control versions of the application throughout the development phase. In which separate repositories were created so as to maintain modularity throughout the project.



- Spring Tool Suite (STS):- This tool contains the necessary JVM to compile the written code throughout the Software Development Phase of SDLC. Basic Debugger also used to detect errors in codes by the developer. Version: 3.9.18 RELEASE, Build Id: 2021091440905, Platform: Eclipse 2021-09 (4.21.0) is used in the project.





- Visual Studio Code: - Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE. This tool provided the necessary GUI for developing front end code in ReactJS.

## 2. SIZE

### 2.1. ESTIMATED AND ACTUAL SIZE (IN KLOC):-

‘KLOC’ (thousands of lines of code) is a traditional measure of how large a computer program is or how long or how many people it will take to write it. The code measured is usually source code. The KLOC metric is often used when evaluating an application's total number of lines of code (LOC) -- or source lines of code (SLOC).

To find KLOC value, divide total number of lines of code by 1000.

- Estimated Size:-
  - Total Lines of Code = 25000
  - KLOC Value = Total Lines of Code / 1000  
= 25000 / 1000 = 25.00
- Actual Size:-
  - Total Lines of Code = 26156
  - KLOC Value = Total Lines of Code / 1000

$$= 26156 / 1000 = 26.15$$

### 3. DEFECTS

#### 3.1. SDLC Stage Wise Defects:-

Defects (also known as bugs or issues) can occur at various stages of the Software Development Life Cycle (SDLC). Each stage of the SDLC involves different activities, and defects can arise during any of these activities. Here's a breakdown of the common SDLC stages and the types of defects that can be found at each stage:

- Requirements Gathering and Analysis: Requirements Defects: Incomplete, ambiguous, conflicting, or incorrect requirements can lead to defects down the line.
- System Design: Design Defects: Inadequate or flawed design can result in defects during implementation or testing.
- Implementation (Coding): Coding Defects: Errors introduced during coding, such as syntax errors, logic errors, or incorrect algorithm implementations.
- Security Defects: Vulnerabilities due to improper handling of inputs, lack of validation, or other security-related issues.
- Deployment: Configuration Defects: Issues with the deployment environment's configuration that affect the software's behavior.
- Maintenance and Support: Maintenance Defects: Defects that are identified after the software has been released and is in production use.

#### 3.2. Distribution of Defects:-

Defects classification as major or minor can vary based on project specific criteria and severity levels defined by the development team. Here, in this project, the defects are distributed as follows:-

- Major Defects :
  - Requirements Defects
  - System Design
  - Coding Defects
  - Security Defects
  - Configuration Defects
  - Maintenance Defects
- Minor Defects :
  - Performance Defects

