

# **PROJECT REPORT**

ON

## **BidRideGo**

In the partial fulfillment of the requirement  
for the program of  
Android Application Programming (CP670)  
in  
Masters of Applied Computing

### **PREPARED BY**

Mitali Padiya (235841100)  
Trushit Patel (235829410)  
Pratham Shah (225847710)  
Bhaskar Vora (235819680)  
Ishaan Morani (235836990)

### **SUBMITTED TO**

**Dr. Abdul-Rahman Mawlood-Yunis**

# **TABLE OF CONTENTS**

<b>Title</b>	<b>Page No</b>
<b>ABSTRACT</b>	<b>4</b>
<b>LIST OF FIGURES</b>	<b>5</b>
<b>LIST OF TABLES</b>	<b>6</b>
<b>ABBREVIATIONS</b>	<b>7</b>
<b>Chapter 1</b>	<b>8</b>
<b>Introduction</b>	<b>8</b>
1.1 Description	8
1.2 Problem Statement	9
1.3 Project Objective	9
1.4 Stakeholders	9
1.5 Features	10
1.6 Tech Stack	10
<b>Chapter 2</b>	<b>11</b>
<b>Project Management</b>	<b>11</b>
2.1 Project Initialisation	12
2.2 Project Planning	12
2.3 Project Timeline	12
2.5 Future plans	12
<b>Chapter 3</b>	<b>13</b>
<b>System Requirements</b>	<b>13</b>
3.1 Functional Requirements	14
3.2 Non-Functional Requirement	14
3.3 Software Requirements	14
3.4 Hardware Requirements	14
3.5 Development Requirements	15
<b>Chapter 4</b>	<b>16</b>
<b>System Design</b>	<b>16</b>
4.1 Flow Diagram	17
4.2 Schema Diagram	18
4.3 Use case diagram	19
4.4 UML Diagram	19
<b>Chapter 5</b>	<b>20</b>
<b>Implementation</b>	<b>20</b>
5.1 Components Used	21
5.2 File Structure	22
<b>Chapter 6</b>	<b>23</b>

<b>Testing</b>	<b>23</b>
6.1 Introduction	24
6.2 Testing Goals	24
6.2 Test Cases	24
6.2.1 RegistrationTest Cases	24
6.2.2 Login Test Cases	25
6.2.3 Ride Test Cases	25
<b>Chapter 7</b>	<b>26</b>
<b>Bibliography</b>	<b>26</b>
<b>Documentations:</b>	<b>27</b>

## **ABSTRACT**

---

The BidRideGo app is a groundbreaking initiative poised to transform the conventional ride-hailing industry. Introducing a dynamic bidding system, this project empowers riders with more choices, cost-efficient options, and flexibility, while simultaneously enhancing earning potential for drivers. By creating a competitive marketplace for on-demand transportation, the BidRideGo app addresses existing challenges in fixed pricing structures and offers a solution for drivers seeking to optimize their earnings.

## LIST OF FIGURES

---

Figures No.	Title	Page No.
Figure 2.1	Project Timeline	12
Figure 3.1	User Flow Diagram	17
Figure 4.2	Driver Flow Diagram	17
Figure 4.3	Schema Diagram	18
Figure 4.4	Use Case Diagram	19
Figure 4.5	UML Diagram	19
Figure 5.1	File Structure Diagram	22

## LIST OF TABLES

---

Table No	Title	Page No.
Table 6.1	Registration Test Cases	24
Table 6.2	Login Test Cases	25
Table 6.3	Ride Test Cases	25

## ABBREVIATION

---

Abbreviations used throughout this whole document for Survey Application are:

MVC	Model View Controller design pattern
XML	Extensible Markup Language
UI	User Interface
UML	Unified Modeling Language

# **Chapter 1**

## **Introduction**

---



## 1.1 Description

---

The BidRideGo app aims to revolutionize the ride-hailing industry by introducing a dynamic bidding system that allows riders to request rides and drivers to bid on those requests. This innovative approach will empower riders with more cost-efficient choices, provide drivers with increased earning potential, and create a competitive marketplace for on-demand transportation services.

## 1.2 Problem Statement

---

- Limited Pricing Flexibility:
  - The fixed pricing structure often fails to adapt to real-time market demand and supply dynamics.
  - Riders are constrained by predetermined fares, limiting their choices and flexibility in selecting rides.
- Earnings Optimization for Drivers:
  - Drivers encounter difficulties in maximizing their earnings.
  - The lack of control over pricing and the need to wait for rides hinder drivers from actively managing and optimizing their income.

## 1.3 Project Objective

---

- The primary objectives of this project are as follows:
- Develop a user-friendly mobile application that enables riders to request rides and receive bids from drivers.
- Implement a bidding methodology that optimizes driver-rider matching based on factors such as bid amount, driver ratings, proximity, and vehicle type.
- Enhance transparency by providing both riders and drivers with real-time information about the bidding process and bid statuses.
- Improve the overall ride-hailing experience by reducing wait times, increasing driver earnings, and offering riders more pricing options.

## 1.4 Stakeholders

---

- **Riders:** Riders are the primary users of the BidRideGo platform. They are directly impacted by the system as it affects their ability to find rides at competitive prices and with various service options.
- **Drivers:** Drivers are essential stakeholders as the project aims to empower them by allowing them to competitively bid on rides. Their earnings and job satisfaction depend on the success of the BidRideGo platform.

## 1.5 Features

---

- Ride Booking
  - Get current location for pick up
  - Date and Time Selection
  - Number of Seats
  - Initial Budget & Car Type Selection
- Real Time Bidding
  - Witness competitive pricing as drivers bid for your ride, ensuring you get the best value for your journey.
- Map Integration
  - In this automatic current location recognition to simplify the process of entering pick-up and drop-off locations. Users can often input locations by typing or selecting from suggestions.
- Upcoming & Past Trips.
  - Both customers and drivers can access their ride history for reference and review.
- Switch to Driver
  - The User can any time become a driver and the driver can select either the user or drive mode of the application.

## 1.6 Tech Stack

---

- **User Interface:**
  - XML
- **Android App:**
  - Google Maps API
  - Open Cage Data API
  - Java
  - Dependency Injection
  - MVC design pattern
  - Singleton design pattern
- **Database:**
  - Firebase
- **Testing Environment:**
  - Android mobile devices
  - Android emulators

# **Chapter 2**

## **Project Management**

---

## 2.1 Project Initialisation

---

During the project initialization phase, the problem statement and the requirements for the BidRideGo app were identified. This involved defining the project scope and setting clear objectives to ensure that the system fulfilled the desired functionalities and modules. In addition, all necessary resources, including software tools, were selected to facilitate the project planning phase. By establishing a clear understanding of the project requirements and goals, it was possible to ensure that the subsequent planning and development phases were appropriately aligned with project expectations.

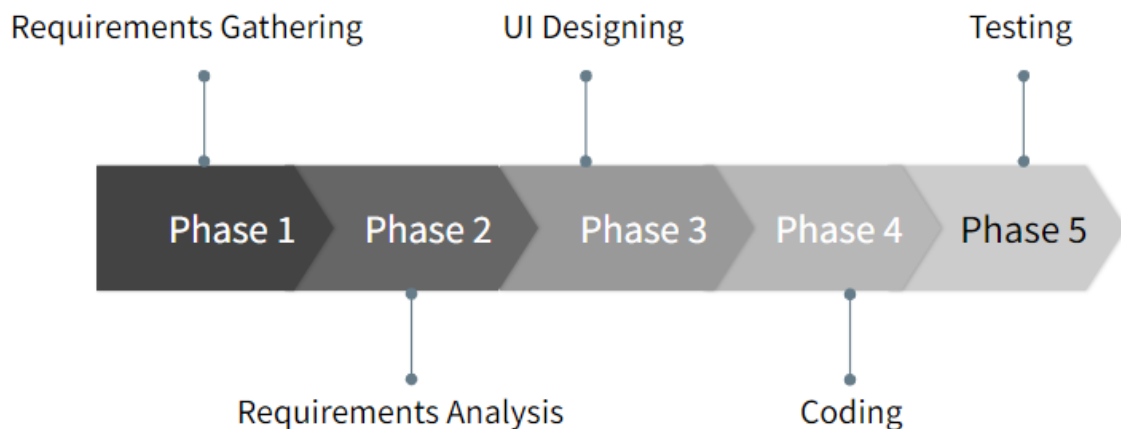
## 2.2 Project Planning

---

During the project planning phase, extensive research was conducted to determine the most suitable tech stack to achieve the desired objectives. The plan involved developing the system architecture, database design, and flow diagrams. After thorough consideration, Firebase was selected as it offers minimal configuration complexities and enables immediate development.

## 2.3 Project Timeline

---



**Fig. 2.1 Project Timeline**

## 2.5 Future plans

---

In addition to regular system updates, bug fixes, and security patches, the future plan for the BidRideGo includes monitoring the system's performance, identifying potential areas for improvement, and implementing necessary changes to ensure smooth and efficient operations. The future plans include integrating a payment API for seamless transactions and social login for customer convenience.

# **Chapter 3**

## **System Requirements**

---

### 3.1 Functional Requirements

---

- System should allow user and driver to register
- System should allow user and driver to login
- System should allow user to post a trip
- System should allow user to add source, destination, trip date and time, number of seats, budget and type of car
- System should allow user to view upcoming and past trips
- System should allow user to user to logout
- System should allow user to view the bids
- System should allow driver to view list of trips
- System should allow driver to bid on a particular trip
- System should allow driver to view his bids and competitive bid price
- System should allow driver to switch himself anytime to user to book a trip
- Driver should be able to log out of the system

### 3.2 Non-Functional Requirement

---

- The system should be scalable to accommodate a growing number of users and products.
- The system should be secure and protect customer data and payment information.
- The system should be responsive and user-friendly across different devices and screen sizes.
- The system should have fast page load times to provide a seamless shopping experience.
- The system should be reliable and available 24/7 with minimal downtime for maintenance and updates.
- The system should be compliant with relevant industry standards and regulations.
- The system should be maintainable and easy to update or modify in the future.

### 3.3 Software Requirements

---

- **Operating System:** Android OS
- **Permissions:** Fine Location, Coarse Location, Background Location, Internet

### 3.4 Hardware Requirements

---

- A device with a screen and input device, such as a tablet, or smartphone.
- An internet connection with sufficient speed and stability to access and use the application without interruptions.

### 3.5 Development Requirements

---

- **Integrated Development Environment (IDE):** An IDE such as Android Studios or EclipseIDE is required to write, test, and debug code efficiently.
- **Database:** Firebase, it is a real-time database that is basically designed for mobile applications.
- **Version Control System:** A version control system like Git should be used to manage source code and collaborate with other developers.

# **Chapter 4**

## **System Design**

---



#### 4.1 Flow Diagram

---

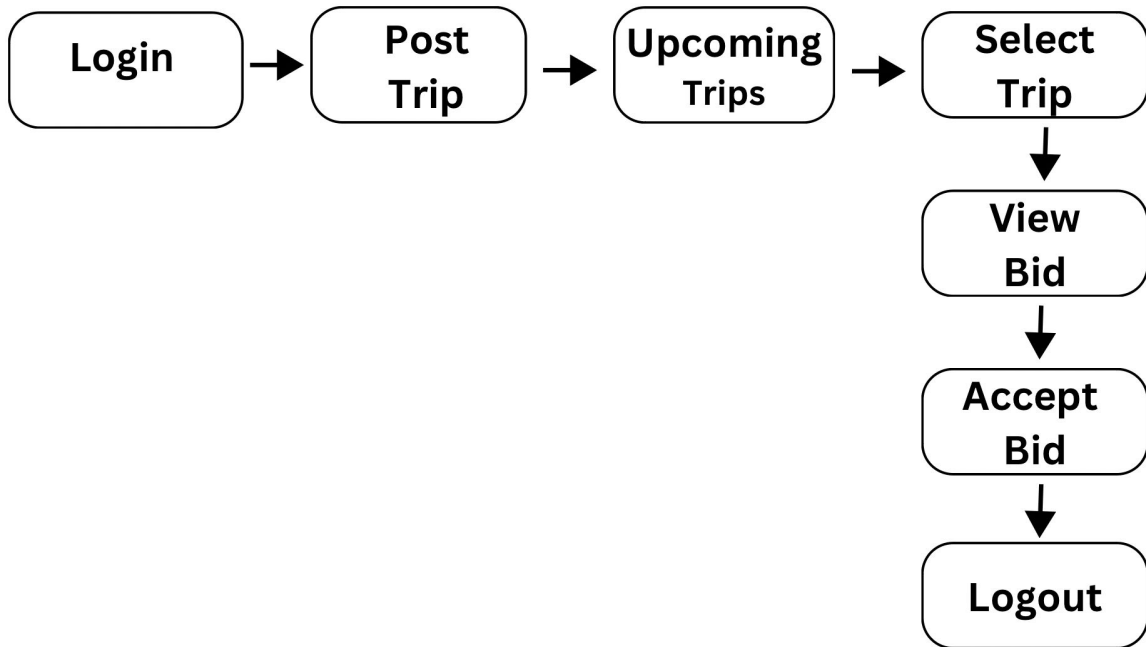


Fig. 4.1 User Flow diagram

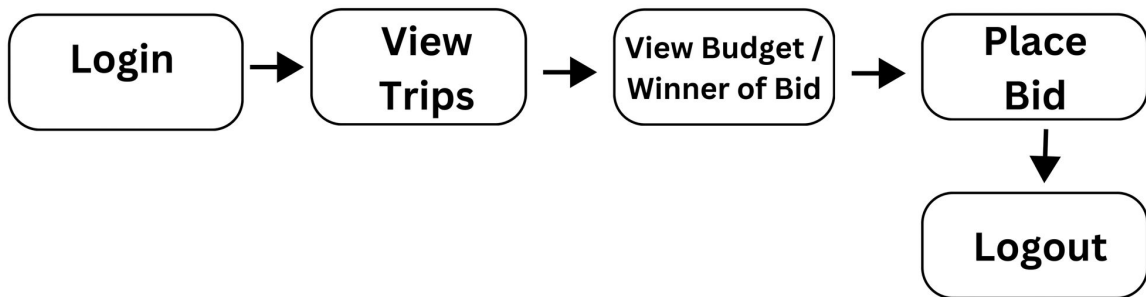
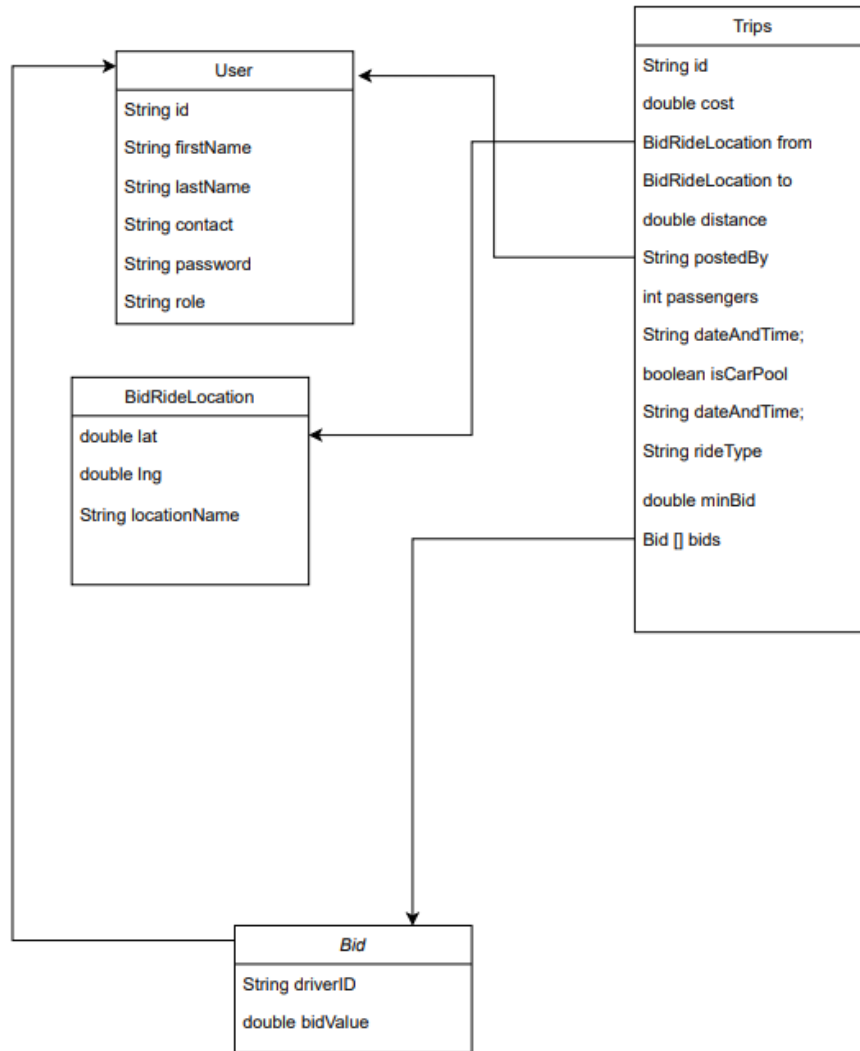


Fig. 4.2 Driver Flow diagram

## 4.2 Schema Diagram

---



**Fig. 4.3** Schema diagram

### 4.3 Use case diagram

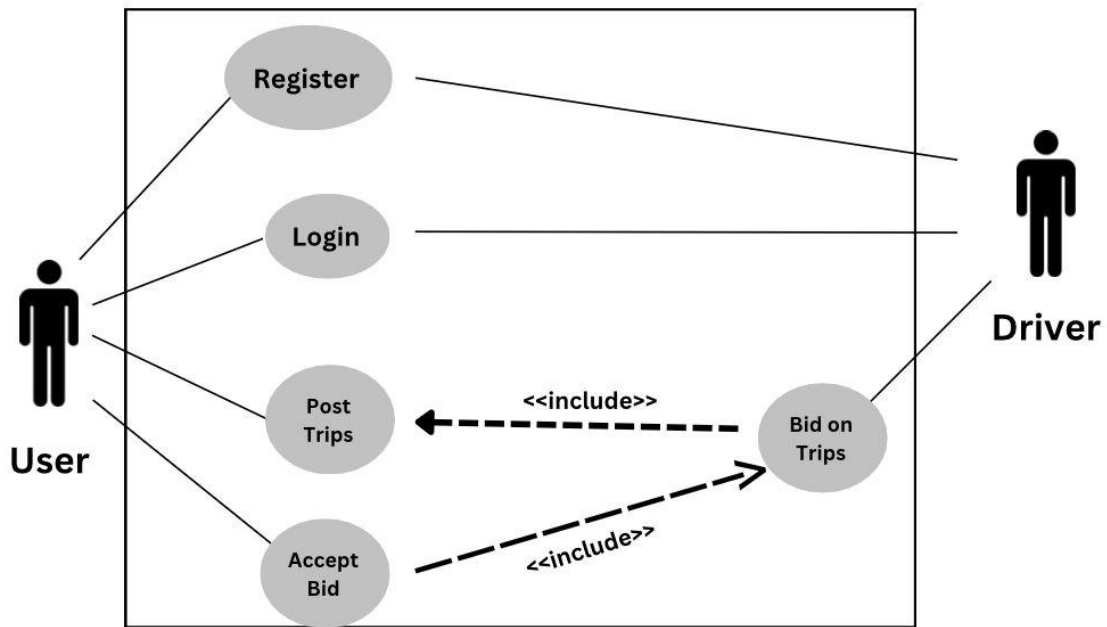


Fig. 4.4 Use Case Diagram

### 4.4 UML Diagram

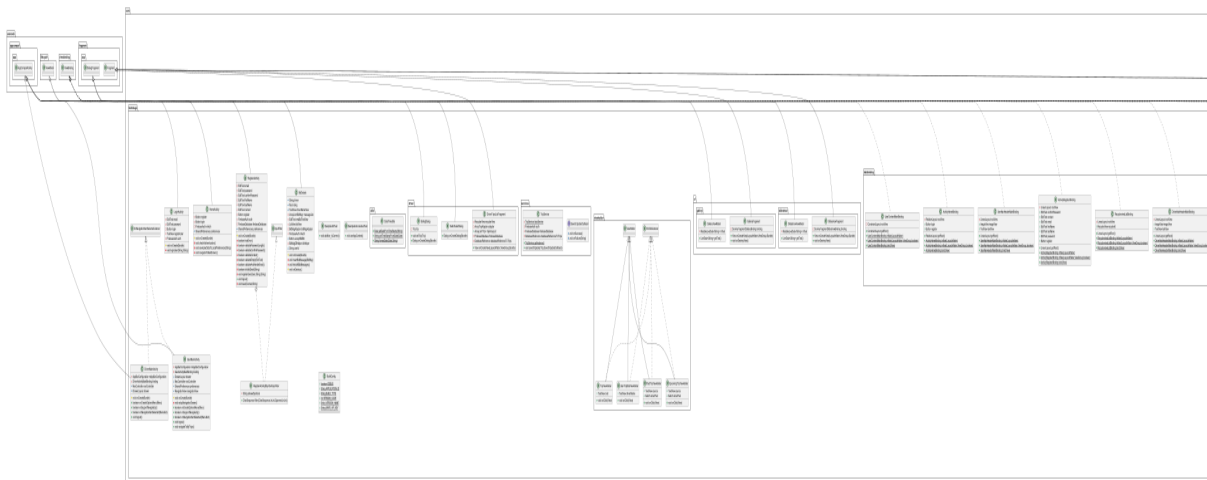


Fig. 4.5 UML diagram

# Chapter 5

## Implementation

---

The implementation of BidRideGo application project involved a comprehensive and systematic approach that addressed various aspects of the project. The project was divided into five phases that were carefully planned and executed to ensure the successful delivery of the end product.

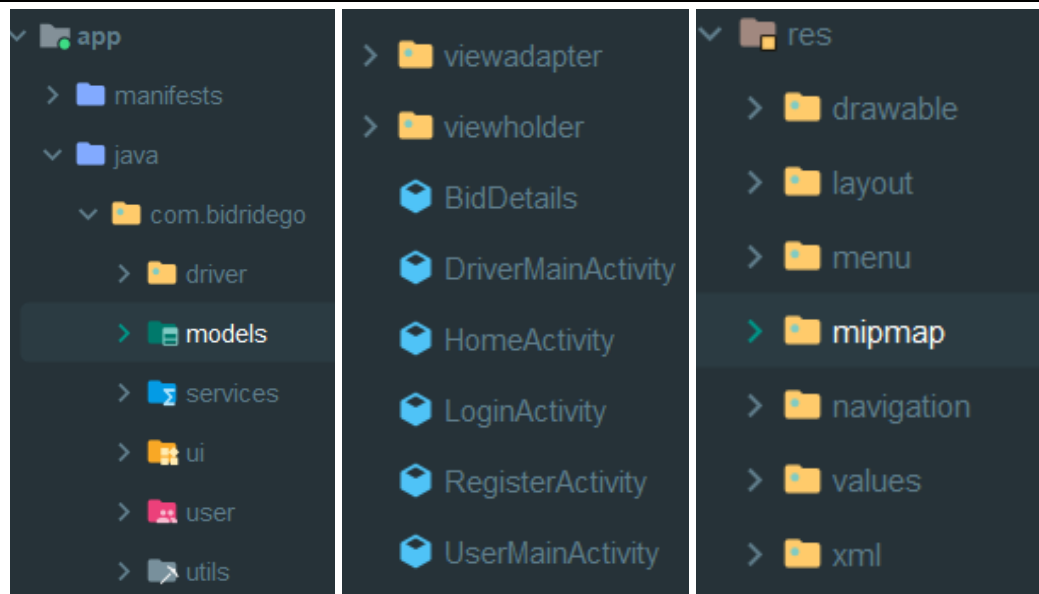
## 5.1 Components Used

---

- **Navigation drawer:** User navigation drawer for navigation options for both user and driver
- **Fragments:** Code dependent on fragments heavily on fragments as we are using navigation drawer as a user and drawer activity. As the navigation drawer is mostly dependent on fragments, we have used fragments in our code.
- **Maps Integration:** We have used maps for fetching the current location, getting latitude, and longitude as per the selected location, setting marker on the maps, when source and destination is added, have used polyline to show the distance between source and destination.
- **OpenCageData API:** As GeoLocation API and Places API from Google are paid, we have used OpenCageData API for showing the suggestions of auto-complete for source and destination.
- **Activity:** We have used activity for Registration, login, start activity for user and driver, and showing trip details
- **Dialog:** We have used Dialogs for showing prompt messages, adding bid value in driver view, and to accept the bid for a trip in user view.
- **Toast and Snackbar:** We have used Toast and Snackbar to show any updates or error messages to the user and driver.

## 5.2 File Structure

---



**Fig. 5.1** File structure

# Chapter 6

## Testing

---

## 6.1 Introduction

Testing is a crucial aspect of ensuring that BidRideGo works as intended. This section outlines the testing approach, goals, and types of tests performed.

## 6.2 Testing Goals

The primary testing goal is to identify and fix any issues or bugs that may arise in BidRideGo. The aim is to ensure that all features and functionality work correctly and that the application is easy to use for both customers and drivers. Additionally, the performance and security of BidRideGo will be validated to ensure that it can handle high traffic volumes and that sensitive customer data is protected.

## 6.2 Test Cases

### 6.2.1 Registration Test Cases

Registration Test Case				
Sr. No	Action	Expected Result	Output	Test Result
1	Any field is empty	Field cannot be empty Error	Field cannot be empty error	Pass
2	Number of digits in contact are not 10.	Number can be 10 digits only error	Number can be 10 digits only error	Pass
3	Email is not proper	Invalid Email address	Invalid Email address	Pass
4	Password length is less than 6	Password length must be 6 characters	Password length must be 6 characters	Pass
5	Confirm Password is not same as password	Should be same as password error	Should be same as password error	Pass
6	All inputs are correct	Registration Successful	Registration Successful	Pass

**Table 6.1 Registration test cases**



### 6.2.2 Login Test Cases

Login Test Case				
Sr. No	Action	Expected Result	Output	Test Result
1	Any or both fields are empty	Empty Credentials Toast	Empty Credentials Toast	Pass
2	In correct credentials	Invalid Credentials Toast	Invalid Credentials Toast	Pass

**Table 6.2 Login Test Cases**

### 6.2.3 Ride Test Cases

Ride Test Case				
Sr. No	Action	Expected Result	Output	Test Result
1	Selecting a previous date	Please select a future date Toast	Please select a future date Toast	Pass
2	Proper fields entered	Ride Now Button is activated	Ride Now Button is activated	Pass

**Table 6.3 Ride Test Cases**

# **Chapter 7**

## **Bibliography**

---

## **Documentations:**

- Android : <https://developer.android.com/guide>
- Firebase: <https://firebase.google.com/docs/guides>