



WOLKITE UNIVERSITY
COLLEGE OF COMPUTING AND INFORMATICS
DEPARTMENT OF COMPUTER SCIENCE

TITLE OF THE PROJECT
ANDROID BASED TAXI BOOKING SYSTEM
BY

NO	STUDENT NAME	ID NO
1	Tesfahun Alemayehu	CIR/090/11
2	Tsion Tizazu	CIR/094/11
3	Bahiru Yalew	CIR/016/11
4	Bisrat Honja	CIR/021/10

PROJECT ADVISOR: Zerihun M (MSc).

June, 2022

WOLKITE UNIVERSITY
COLLEGE OF COMPUTING AND INFORMATICS
DEPARTMENT OF COMPUTER SCIENCE

TITLE OF THE PROJECT

ANDROID BASED TAXI BOOKING SYSTEM

SUBMITTED TO DEPARTMENT OF COMPUTER SCIENCE
IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR
THE DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE

BY

NO	STUDENT NAME	ID NO
1	Tesfahun Alemayehu	CIR/090/11
2	Tsion Tizazu	CIR/094/11
3	Bahiru Yalew	CIR/016/11
4	Bisrat Honja	CIR/021/10

PROJECT ADVISOR: Zerihun M (MSc).

Wolkite University, Wolkite, Ethiopia

June 16, 2022

DECLARATION

This is to declare that this project work which is done under the supervision of Zerihun M (MSc) and having the title ANDROID BASED TAXI BOOKING SYSTEM is the sole contribution of: Tesfahun Alemayehu, Tsion Tizazu, Bahiru Yalew and Bisrat Honja

No part of the project work has been reproduced illegally (copy and paste) which can be considered as Plagiarism. All referenced parts have been used to argue the idea and have been cited properly. We will be responsible and liable for any consequence if violation of this declaration is proven.

Date: _____

Group Members:

Full Name

Signature

1, Tesfahun Alemayehu

2, Tsion Tizazu

3, Bahiru Yalew

4, Bisrat

Honja

Approval Form

This is to confirm that the project report entitled Android Based Taxi Booking System submitted to **Wolkite University, College of Computing and Informatics, Department of computer science** by: Tesfahun Alemayehu, Tsion Tizazu, Bahiru Yalew and Bisrat Honja is approved for submission.

-----	-----	-----
Advisor Name	Signature	Date
-----	-----	-----
Department Head Name	Signature	Date
-----	-----	-----
Examiner 1 Name	Signature	Date
-----	-----	-----
Examiner 2 Name	Signature	Date
-----	-----	-----
Examiner 3 Name	Signature	Date

Acknowledgment

We would like to thank creator of the universe almighty God for giving us the strength to fulfill our duty. Thank you, God!

We would like to thank our Advisor Zerihun M (MSc) for his valuable comments, support, and guidance throughout this project work. For give us encouragement and constant guidance, to finish this project.

It is a great pleasure to get an opportunity to thank our families for supporting and teaching us. Thank you, our Families! Finally, we would like to thank our lectures and friends for their encouragement and support throughout this this work

Android Based Taxi Booking System

Table of contents

Table of contents	iv
List of Figures	vii
Lists of Tables	ix
ABSTRACT	x
LIST OF ABBREVIATIONS	xi
CHAPTER ONE.....	1
1 INTRODUCTION.....	1
1.1 Background of the Study	1
1.2 Statement of the Problem	2
1.3 Objectives of the Study.....	3
1.3.1 General Objectives	3
1.3.2 Specific Objectives	3
1.4 Feasibility Study	4
1.4.1 Technical Feasibility.....	4
1.4.2 Behavioral Feasibility	4
1.4.3 Operational feasibility	4
1.4.4 Economic Feasibility	4
1.4.5 Schedule feasibility.....	5
1.5 Scope and Limitation of the Study	6
1.5.1. Scope of the Project.....	6
1.5.2 Limitation of the Project.....	6
1.6 Significance of the project.....	6
1.7 Target beneficiaries of the project	7
1.8 Methodology of the Project	7
1.8.1 Data Collection Tools/Technique	8
1.8.2 System Analysis and design	8
1.8.3. System Development Model.....	9
1.8.4. System Testing Methodology	9
1.8.5 Development tools and Technologies	10

Android Based Taxi Booking System

1.9. Documentation	11
CHAPTER TWO.....	12
2 DESCRIPTION OF THE EXISTING SYSTEM.....	12
2.1. Introduction of Existing System	12
2.2. Users of Existing System.....	13
2.3. Major Functions of the Existing System.....	14
2.4. Drawbacks of the Existing System	15
2.5. Business Rules of the Existing System.....	15
CHAPTER THREE	17
3 PROPOSED SYSTEM.....	17
3.1. Functional Requirements	17
3.1.1. Passenger App Module	17
3.1.2. Driver App Module	17
3.1.3. Administration Module.....	18
Admin.....	18
Dispatcher.....	18
3.2. Non-Functional Requirements	19
3.2.1 User Interface and Human Factors.....	19
3.2.2 Hardware Consideration	19
3.2.3. Security Issues	19
3.2.4. Performance Consideration.....	19
3.2.5. Error Handling and Validation.....	20
3.2.6. Quality Issues	20
3.2.7. Backup and Recovery	20
3.2.8. Physical Environment	20
3.2.9. RESOURCE ISSUES	21
3. 2.10. Documentation	21
CHAPTER FOUR	22
4 SYSTEM ANALYSIS.....	22
4.1. Introduction	22

Android Based Taxi Booking System

4.2. System Model	22
4.2.1 Use Case Model.....	23
4.3 Object Model	38
4.4 Dynamic Model	42
4.4.1 Sequence Diagram	42
5 SYSTEM DESIGN.....	54
5.1. Design Goals	54
5.2. Proposed Software Architecture	56
5.3. Packages	64
5.4. Algorithm Design	64
5.5. User Interface Design	66
CHAPTER SIX	70
IMPLEMENTATION AND TESTING	70
6.1. Implementation of the Database	70
6.2. Implementation of the class diagram	70
6.3. Configuration of the application server	70
6.4. Configuration of Application Security	71
6.5. Implementation of User Interface	71
6.6. Testing the system	71
6.6.1. Test Case	72
6.6.2. Testing tools and Environments	73
6.6.3. Unit testing	73
6.6.4. System testing.....	73
6.6.5. Integration testing	73
CHAPTER SEVEN	75
CONCLUSION AND RECOMMENDATION	75
7:1 Conclusion.....	75
7.2 Recommendation	75
References	76

Android Based Taxi Booking System

Figure 4. 1: Use Case Diagram for Android Based Taxi Booking	25
Figure 4. 2: Class Diagram for Android Based Taxi Booking System.....	39
Figure 4. 3: Sequence Diagram for Log In.	43
Figure 4. 4: Sequence Diagram for Booking	44
Figure 4. 5: Sequence Diagram to Registration	45
Figure 4. 6: Sequence Diagram for tracking	46
Figure 4. 7: Activity Diagram for Login.....	47
Figure 4. 8: Activity Diagram for registration	48
Figure 4. 9: Activity Diagram for Book Request.....	49
Figure 4. 10: Activity Diagram for Update.....	50
Figure 4. 11: Activity Diagram for Delete.....	51
Figure 4. 12: Activity Diagram for Cancel	52
Figure 4. 13: State Chart Diagram for Booking.....	53
Figure5. 1: Proposed Software Architecture.....	57
Figure 5. 2: Component Diagram for Android Based Taxi Booking System	59
Figure 5. 3: Deployment Diagram for Android Based Taxi Booking System.....	60
Figure 5. 4: Detailed Class Diagram for Android Based Taxi Booking System	61
Figure 5. 5: Persistence Diagram for Android Based Taxi Booking System.....	62
Figure 5. 6: Package Diagram for android based taxi booking System	64
Figure 5. 7: Passenger Homepage	67
Figure 5. 8: Passenger Sign up Page.....	68
Figure 5. 9: Passenger Sign in Page.....	69

Lists of Tables

Table 4. 1: Use Case Documentation for Login	26
Table 4. 2: Use Case Documentation for Manage Vehicle	27
Table 4. 3: Use Case Documentation for Manage Account	29
Table 4. 4: Use Case Documentation for Tracking Trips.	30
Table 4. 5: Use Case Documentation for Assign Driver.....	31
Table 4. 6: Use Case Documentation for Pickup and Drop off Location	33
Table 4. 7: Use Case Documentation for Book Request.....	34
Table 4. 8: Use Case Documentation for Payment	35
Table 4. 9: Use Case Documentation for Cancel Booking	37
Table 4. 10: Data Dictionary for Admin	40
Table 4. 11: Data Dictionary for Passenger.....	40
Table 4. 12: Data Dictionary for vehicle	41
Table 4. 13: Data Dictionary for Driver	41
Table 4. 14: Data Dictionary for Dispatcher.....	42
Table 5.1: Access control.....	63

Android Based Taxi Booking System

ABSTRACT

This project is an Android Based Taxi Booking System that enables the passengers to book or hail a taxi and drivers to accept or reject booking with the help of a smartphone application.

Currently, in most parts of our country the passengers are just tired of waiting at different locations for a taxi that they have just reserved to pick them up and take them home. What we need is a reliable mobile application which reserves a taxi for us from a specific place at a certain time and which tells its passengers the status of the taxi in order to keep them from waiting for long hours. Taxi booking System serves just for this purpose. This android based taxi booking system allows the Passengers to get a taxi from any location, even if they seem to be lost. One does not have to spend hours on phone with the taxi services to tell them where exactly they are located currently. This application allows filling pickup and destination place to the passengers, and knows which taxi would take to reach to the destination. Once a Passenger reserves a taxi, he is checked for booking whether it is accepted or denied by a driver through the dispatcher. This application enables the passengers to view an available taxi, status of taxi and status of driver.

Android Based Taxi Booking System

LIST OF ABBREVIATIONS

Acronyms	Description
API	Application Program Interface
BR	Business Rule
CPU	Central Processing Unit
FK	Foreign Key
GPS	Global Positioning System
OOA	Object Oriented Analysis
OOD	Object Oriented Analysis
PHP	Hyper Text Pre Processor
PK	Primary Key
P_ID	Passenger ID
RAM	Random Access Memory
SDK	Software Development Kit
UI	Passenger Interface
UC	Use case
UML	Unified Modeling Language

CHAPTER ONE

1 INTRODUCTION

In today's world the numbers of mobile users are increasing and a lot of applications, and services generate a lot of data, and information to mobile users. These mobile users are anyone who can be executive, sales people, service engineers, teachers, etc. In general, everyone is becoming a mobile service user. These users are using different applications; among the many applications used by many users are productivity applications. Our team member will build a productivity application named Android Based Taxi Booking System.

The main objective of the project is to provide an easy to use and handy mobile application to the android users which enables them to book a taxi from any location to any location. The users can keep track of their bookings and the taxi's current status. The aim of this project is to produce a software system that matches customers and taxi drivers in real time using existing cell phone and the designed system consists of a client mobile application for customers, a mobile application for taxi drivers, a server with a database, and web. The purpose here is to book the taxi any point in time using the Android application. The customer just books the taxi using this application and his details and location is sent to the web site where the software finds taxi. After finding the taxi, software sends the customer details to the taxi driver and if the taxi driver accepts the request then the taxi is booked for the given customer; else the customer details are forwarded to other taxi driver.

1.1 Background of the Study

A taxi is a type of vehicle for hire with a driver, used by a single passenger or small group of passengers. Nowadays, taxi services are in everywhere big cities, medium-sized cities and even small cities around the world have reasonable infrastructure to provide this service. Taxi systems try to meet service demands of native population and tourists that visit or work on these places. At most of the cities around the world, it is possible to observe that taxi drivers waste a large portion of time about 50% - at vacant times. Because of this high rate of wasting time, there are a large number of studies that intend to improve the efficiency of these services, without increasing the costs. Some years ago,

Android Based Taxi Booking System

the use of geographical location information has been in expected time to serve. A location-based service (LBS) is a general term denoting software services which use geographic data and information to provide services or information to users. Unfortunately, in most of our country parts the existing system do not use digital map/Google map to give the service based on the use of geographical location information. Therefore, the existing system does not meet the demands on an optimized way.

The existing system in the largest part of our country is a manual system, which means customers stand for a long time on the street by waiting for coming taxi or make a call for driver they already know. Its operation looks like this passenger call phones to taxi drivers or to the taxi service providers otherwise; they make physical contact to the taxi drivers or transport service providers. This project needs to make significant change on the existing system. On the other hand, there are some taxi service applications in our country. For example RIDE, and Feres are a transport hailing and booking platform in Ethiopia, that can be accessed by calling their dispatch centre or by using their free passenger app, which are not manual but most of them are available in Addis Ababa.

1.2 Statement of the Problem

There are a lot of problems in the existing system. Due to that passengers as well as taxi drivers are wasting their time and money, for example, passengers stand for a long time on the street by waiting for coming taxi or make a call for driver they already know, so it is difficult to customers to find a taxi on the road. In other way taxi drivers are also wasting their time and resource by staying somewhere until the customer calls them or come to them or they find customer by roaming. Although as there is no legally specified pick and drop fee for every place based on km or any distance measurement, customers are paying extra money for short route. Road side booking of passengers is inefficient and causes occasional disputes among bookers, passengers and drivers. Therefore, taxi booking project will came up with mobile based system to solve those problems mentioned above by enabling the customer to book taxi and drivers to receive booking order from the system anywhere in the town. This would enable that each customer who made a booking is served within the shortest possible time, as result it will increase operational efficiency and enhancing customer satisfaction. But it is better to use

Android Based Taxi Booking System

location-based response for drivers. Therefore, drivers will respond to customers request by current location. And our system makes payment options simply, customers can pay online while they have access of online payment options (systems).

In ride hailing system, the payment mechanism is in hand to hand payment (or the consumer/passenger pays the cost directly to the driver not to directly to the organization). So, this payment method has so many negative impacts on the organization, government and consumers/passengers. This payment mode doesn't reduce man power specifications.

On the new updated software, order history doesn't show the date on the order detail. As you all know the app didn't support corporate option, you guys also blocked and increased the tariff of telephone line we are using for corporate booking service. It doesn't support daily corporate Passengers. Most of the time this application has a problem of opening through 3G networks and most of our countries phone are using this network.

1.3 Objectives of the Study

1.3.1 General Objectives

The main aim of our project is to develop Android Based Taxi Booking system for Ethiopia.

1.3.2 Specific Objectives

- ✓ To plan the design and development of our system
- ✓ To develop an application that is easy and convenient way in getting a taxi.
- ✓ To develop procedures for appropriate recording of passengers
- ✓ To develop interactive and user-friendly interface
- ✓ To understand the problems in the current system.
- ✓ To plan the solution for the problems identified.
- ✓ To plan the way to ensure the integrity of data in our proposed system.
- ✓ To determine how data will be entered into our system.

1.4 Feasibility Study

1.4.1 Technical Feasibility

The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. The software development tools that we are using in development of the proposed system are freely available (open source) on the internet.

We use XML and android (java) for the Front-end, PHP and MySQL database for the back-end that are not that much new for us. So, the proposed system is technically feasible.

1.4.2 Behavioral Feasibility

The project would be beneficial because it satisfies the objectives of the users and organization legally. The system was developed user-friendly and improves the working environment. So, that our system is using simple and interactive user interface and problem occurs the admin can maintain it easily by using the given settings. We can say our system is free from any political and environmental difficulty. Due to this reason our project is behaviorally feasible.

1.4.3 Operational feasibility

Operational Feasibility deals with the degree to which the proposed application solves business problems. This application will solve the problems in the following ways.

- ✓ By providing a new and reliable means to handling booking taxi.
- ✓ By providing current location and appropriate confirmation for booking.
- ✓ Better management of taxi.

1.4.4 Economic Feasibility

Economic Feasibility is about identifying the costs and benefits related with developing the project. That is why it is sometimes called Cost-Benefit Analysis. It encompasses resources like hardware cost, software cost and time. This project will use free and open-source libraries, frameworks and IDE's to develop the system. The development and deployment of this project result in ultimate benefit to the drivers and customers from economic perspective. The current system used by the taxi booking in most of our

Android Based Taxi Booking System

country parts result in enormous expenditure on phone call, fuel, time and other costs due to improper mechanism in the existing system. Our system resolves this additional requirements and expenditures by using a computerized system. As a result, the proposed system will be economically feasible.

1.4.5 Schedule feasibility

Schedule feasibility determine how the proposed system accomplished with the given time table. It implies effective time management for the system, and the project should finish within deadline. So, the team decides to implement and configure the new system on time without any delay.

Our project has four main phases:

1. Requirement collection: in this phase we collect the requirements that are needed to develop our system.

2. Analysis: in this phase we analyze the requirements collected. It is concerned with becoming aware of the problem, identifying the relevant and most decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution.

The system is going to be conducted including:

Description of the existing system

- ✓ Problem of the existing system
- ✓ Functional and non-functional requirements
- ✓ Proposed system
- ✓ System design
- ✓ Implementation
- ✓ Testing

3. Design: the transformation of analysis model into design model. In this stage we make the diagrams that show how the new system does in its implementation stage. In this stage different modelling technique used for instant use case diagram, sequence diagram etc.

4. Implementation: in this stage the new system run and tested based on the layout specified in the design and requirements of analysis stage.

1.5 Scope and Limitation of the Study

Under this we deal with scope that describes what our project can do and the limitations that we will be faced when doing this project.

1.5.1. Scope of the Project

The main aim of our project is to develop Android Based Taxi Booking system for Ethiopia.

The android based taxi booking system will perform the following activities:-

- ✓ **Easy Booking:** passenger can book or order taxi by sending their pick up, drop of place, time and date.
- ✓ **Fast information exchange between drivers and customers:** The driver can send his details to the customer including taxi current status.
- ✓ **Easy and fast administration:** Easy controlling and managing the whole process between taxi drivers and passengers.
- ✓ **Location Tracking:** The driver can track the current location of passengers easily when the passenger send request by specifying starting location, destination location and arrival time. In the same way, the passenger can view the live location of driver.

1.5.2 Limitation of the Project

Just as any researcher is bounded and restricted to encounter some limitation, this project is also faced with the following limitations.

- ✓ This project is only focus on contract taxi service
- ✓ Works in android operating system only.
- ✓ It would not work without the internet connection.

1.6 Significance of the project

The significance of the study is to create android based taxi booking system that will provide services to customers and easy to use. Some of the significances of the proposed system described as follows:

Android Based Taxi Booking System

This Taxi Booking Application is the ideal taxi booking application for individuals on the travel. This user-friendly application offers convenience by allowing users to pre-set their favorite locations and journeys for their taxi bookings. It even allows users to book a taxi at their current location. For ease of convenience, users can also access and book from a list of completed trips.

Instead of dialing to the service center, taxi booking application will help the users deliver their requests directly to the nearest taxi drivers through tracking their current location. It greatly saves the trouble calling to service center and waiting reply from it. It will serve to:

- ✓ Reduce waiting time for users
- ✓ Customers can easily send pickup, drop off place and arrival time to the drivers
- ✓ Taxi drivers can decline or accept the request
- ✓ Avoid paper based or hand recording system.
- ✓ Prevent the users from stress and confusion.

1.7 Target beneficiaries of the project

Different parts are going to be benefited from the new system such as:

- ✓ **Drivers:** - drivers will be benefited from this system by tracking the pickup and destination place of the passenger and decrease tiredness that comes from searching the passengers on the road. Drivers who use this system will perform their work effectively and efficiently without time loss and loss of resource.
- ✓ **Customers:** -This will remove wastage of customers' time that are standing for a long time on the street waiting for coming taxi, again they will gain fast access and save their time.
- ✓ **Transport office:** -By using this system transport office can properly manage all taxi those are inside the service provider.

1.8 Methodology of the Project

Methodology is the way or mechanism in which we gather the information's to develop the system. To develop appropriate android-based taxi booking mobile application the current system must investigate thoroughly and enough information has to be gathered.

Android Based Taxi Booking System

1.8.1 Data Collection Tools/Technique

Methods and methodology, we are going to use in this project to collect essential information and requirements. We will use the following techniques to gather requirements and to describe the ABTBS application.

- ✓ Internet is our major source to gather information.
- ✓ Review related works
- ✓ By observing the existing problem: - We observe the place where taxi exists and by observing the actual work of taxi transportation we will determine how taxi is going to be available for the customers.
- ✓ Discussing and analyzing the problems with project team members: - To get the necessary information we will explore and analyze written materials that describe the operations conducted in taxi booking.

1.8.2 System Analysis and design

We decide to use object-oriented system analysis and design methodology because of the following reason:

- ✓ **Object-oriented techniques** work well in situations where complicated systems are undergoing continuous maintenance, adaptation and design
- ✓ **Simplicity:** software objects model real world objects, so the complexity is reduced and the program structure is very clear.
- ✓ **Reusability:** the object oriented provides opportunities for reuse through the concepts of inheritance, polymorphism, encapsulation and modularity.
- ✓ **Increased Quality:** Increases in quality are largely a by-product of this program reuse.
- ✓ **Increased extensibility:** -when we need to add new feature to the system we only need to make changes in one part of the applicable class.
- ✓ **Maintainable:** OOP methods make code more maintainable. Objects can be maintained separately, making locating and fixing problems easier. The principles of good OOP design contribute to an application's maintainability.

Android Based Taxi Booking System

- ✓ **Modifiability:** It is easy to make minor changes in the data representation or the procedures in an OO program.

1.8.3. System Development Model

We select software development life cycle models is iterative because of the following reason: -

- ✓ Iterative process starts with a simple implementation of small sets of the software requirements.
- ✓ Iteratively enhances the evolving version until the complete system is implemented and ready to deploy.
- ✓ It doesn't attempt to start with a full specification of requirements instead development begins by specifying and implementing just part of the software which is then reviewed in order to identify further requirements
- ✓ Easier to test and debug during a smaller iteration.
- ✓ Easier to manage risk because risky pieces are identified and handled during its iteration.

1.8.4. System Testing Methodology

- ✓ **Unit testing:** every developer and analyst would test each module. In our system we will test whether the return type of functions is correct, whether functions are called correctly, whether correct output is produced for different inputs and efficiency of the code in terms of CPU and memory usage.
- ✓ **System testing:** after the whole integration of all expected system modules, the whole system would be tested using sample. In this level of testing process, we will have examined how the whole subsystems came together to achieve the desired goal. The goals of system testing are to detect faults that can only be exposed by testing the entire integrated system or some major part of it.
- ✓ **Integration Testing:** -is the activity of software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before acceptance testing. Thus, Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested.

Android Based Taxi Booking System

1.8.5 Development tools and Technologies

1.8.5.1 Frontend Technologies: - It's all about what a user is able to see.

- ✓ XML
- ✓ Android(java) programming

1.8.5.1 Backend Technologies: of anything is not visible to general user or administrator. It can be a piece of code or a program running on the server machine to serve the user need.

- ✓ PHP
- ✓ MySQL Database

1.8.5.3 Documentation and Modeling Tools

- ✓ Microsoft word 2019: -To write the documentation
- ✓ Microsoft PowerPoint 2019: -For presentation at the end of the project
- ✓ Draw.io: - to draw different UML diagrams such as use cases, sequence diagrams, activity diagrams, class diagrams.....and so on.
- ✓ Editors: -Sublime, android studio for writing programs or codes
- ✓ Browser: -Google chrome used to run the program.

1.8.5.4 Software Requirements

For the development of this project, the following software requirements will be considered.

- ✓ Operating System: Windows 10
- ✓ Language: Android SDK 3.1 or above, Java
- ✓ Database: MySQL version 5.1
- ✓ Tools: Eclipse Juno IDE or android studio
- ✓ Technologies: Java, MySQL, Android, XML,
- ✓ Network: Mobile network and Internet (cellular or Wi-Fi)

1.8.5.6 Hardware Requirements

For the development of this project, the following hardware requirements will be considered.

- ✓ RAM: 500 MB

Android Based Taxi Booking System

- ✓ Space on disk: 250MB or higher
- ✓ Device: Android phone with version 4.0 or higher

1.9. Documentation

The document organization of our project is well organized and documented. The project documentation has five chapters. Chapters' organization is based on the WKU CCI industrial project guidelines.

Chapter one contains the introduction part of the project. It contains the statements of the problems, objectives, significances, beneficiaries, methodologies and approaches that we use are described well as necessarily.

Chapter two contains the description of the existing system: - It describes activities and major functions, users of the system and the business rules or the work flow as well as the drawbacks of the existing system.

Chapter three contains the proposed system: - in this chapter the functional and non-functional requirements of the system are described well. All the security and performance related issues and hardware and software requirements, the error handling mechanisms, and quality related concepts are also discussed in this chapter.

Chapter four contains System Analysis: - in the system analysis section all the system model (use-case model and diagrams, use-case descriptions and scenarios), object model (class diagram and data dictionary) and the dynamic model (sequence, activity and state cart diagrams) of the system are described clearly and precisely.

Chapter five contains system design: - the system design section discussed about the design goal, system architecture, system decompositions, software and hardware mapping and packages, detailed class and consistent data management also described in this section. The abbreviated words used in the document are presented with their description in a table form for clear understanding.

CHAPTER TWO

2 DESCRIPTION OF THE EXISTING SYSTEM

2.1. Introduction of Existing System

Existing system is totally on book and thus a great amount of manual work has to be done. In existing system, there are various problems like customer/taxi finding, prices of services and fixing bill generation on each bill and also finding out details regarding any information is very difficult. Major problem was lack of security because customers don't have any guarantee about the driver's identity. With this existing low priority in terms of exchanging of data within short period of time. And usually there is no published fares and the customers will end up paying what the Driver demand for the service amount and also there is no standard system that will number the distance in the sense that customers are not assured of the distance preference.

RIDE is the transport hailing and booking platform in Ethiopia that gives an access in Addis Ababa. It has ride free Passenger App to book/hail next trip on the streets of Addis Ababa or call the dispatch center at 8294.

In addition to call center facility (8294) Ride service features encompass the following key items:

Passenger App/ online booking Features

- ✓ Displays available online drivers live on a map
- ✓ Allows users to hail RIDE on demand or pre-booked for later use
- ✓ Past ,current and future trip history with detailed pieces of info such as driver Name, license plate, trip distance, amount charged, duration and method of payment

Online platform

- ✓ Book online on a desktop and manage trips live
- ✓ Review and manage E-wallet and corporate accounts
- ✓ Consolidated past trip history with detailed pieces of info such as driver ID, license plate, distance travelled marked on a map, amount charged, duration and

Android Based Taxi Booking System

exact time of service, method of payment and rating of driver

- ✓ Full periodical digital reports and cab print out receipts to corporate and individual users to budget efficiently
- ✓ Location tracking see the area which the driver and the customer located.
- ✓ Assign driver to customers through nearest location of drivers for customers.
- ✓ Fare calculator for payment options
- ✓ Capacity to refresh vehicle data and transfer photographs of the vehicle for documentation reason and care [1].

Feres: As one of the prestigious selections of GPS following framework supplier in Ethiopia. FERES offers Driver GPS tracker, vehicle GPS tracker, and customer GPS tracker. It has an opportunity to make taxi booking sufficiently. It can handle and gives an appropriate ordering and servicing for appropriate costumer requests.

FERES has different functions to perform:

- ✓ Location tracking for services.
- ✓ Driver and customer registration for security issues.
- ✓ Capacity to refresh customer, driver and vehicle data/information for transfer to documentation for future use.
- ✓ Capacity to audit daily, monthly and yearly usages.
- ✓ Capacity for price calculation in the payment system through appropriate distancetraveled [2].

2.2. Users of Existing System

Passengers/customers: Those are consumers who has to be serviced by an organization for their transportation system. These consumers can communicate with the organization through a passenger app module in their mobile phone.

There are some functionalities of these consumers such as:-

- ✓ Register/log-in
- ✓ Booking requests to Admins
- ✓ Pick-up location (of passengers) to the driver app
- ✓ Drop-off location (of passengers) to the driver app

Android Based Taxi Booking System

- ✓ Price calculator
 - ✓ Tracking taxi
 - ✓ Interactive map

Driver: These are peoples who have a job in the organization to drive taxis to give services for consumers. And they have accessed the organization and the consumers through a driver app module on their android phone. The have some functionalities such as: -

- ✓ Registration
- ✓ Pick-up & drop-off navigation

Admin: Those are the organizations administrator or manager who controls the whole functionalities inside the organization. And it manages and controls the drivers, cars, cost estimation, and so on inside the organization.

- ✓ Driver management
- ✓ Vehicle management
- ✓ Payment management
- ✓ Generate report
- ✓ Tracking trips
- ✓ Fare estimation
- ✓ Receipt
- ✓ Notification
- ✓ Manage service requests
- ✓ Matching drivers & passengers

2.3. Major Functions of the Existing System

✓ For the Transportation Service

Upon service request and using GPS technology, customer and the third-party service provider will have real time communication and established connection to detect location information to the service provider.

If the Service Provider accepts your request, the Application program notifies you and provides information regarding the Service Provider - including driver's name, vehicle type and vehicle plate number, and the ability to contact the Service Provider by

Android Based Taxi Booking System

telephone.

The Application also allows you to view the Service Provider's progress towards the pick-up point, in real time.

✓ **For Better Security**

This application gives a better security for passengers and drivers. It protects everyone's privacy with respect to login systems. Although these applications are legally registered, so the passengers didn't afraid of any unauthorized peoples.

2.4. Drawbacks of the Existing System

As the fast growing of technologies in the country there are a lot of new software's and applications are developed rapidly.

In ride hailing system, the payment mechanism is in hand to hand payment (or the consumer/passenger pays the cost directly to the driver not to directly to the organization). So, this payment method has so many negative impacts on the organization, government and consumers/passengers. This payment mode doesn't reduce man power specifications.

On the new updated software, order history doesn't show the date on the order detail. As you all know the app didn't support corporate option, you guys also blocked and increased the tariff of telephone line we are using for corporate booking service. It doesn't support daily corporate Passengers. It doesn't support all android devices which doesn't access play store like Huawei android phones. Most of the time this application has a problem of opening through 3G networks and most of our countries phone are using this network.

2.5. Business Rules of the Existing System

Business rule is effectively an operating principle or policies that we try to specify for both the existing system and the new system must satisfy. The business rule is principle or a policy in which the proposed system operates accordingly. There are some business rules and constraint to prevent any violation during the process. This focuses on access control issue.

BR01: Drivers must be approved by the admin to start delivering their services.

Android Based Taxi Booking System

BR02: All vehicles serving under the system should be either code 1 (vehicle licensed to carry paying passengers) and code 3(Commercial vehicles).

BR03: Drivers must have a valid driving license (License type permitted by insurance company) to be registered in to the system.

BR04: Passengers should be registered before to access the services that are provided by an organization.

BR05: In this system the initial cost estimation is 50 ETB. And the cost is calculated through ten ETB per one kilometer.

CHAPTER THREE

3 PROPOSED SYSTEM

3.1. Functional Requirements

3.1.1. Passenger App Module

- ✓ **Register/log-in:** All the passengers who use this system should register for the first time. And login through using Passenger name and password for security purposes and privacy.
- ✓ **Booking requests:** passengers requests booking to get services through login in the system and by locating their location to book a service.
- ✓ **Locate Pick-up and Drop-off location (of passengers) to the driver app:** By turning on their GPS passengers can locate their pick-up location. On the other hand, by locating the destination area they can locate their drop-off location.
- ✓ **Tracking taxi:** This module can track the taxi status and the driver status.
- ✓ **Interactive map:** The passenger's app module has a Google map that can control and lookup for the starting location to the destination location.
- ✓ **Real-time GPS tracking:** allows users to see the live locations of drivers while waiting for them and shows the available taxis nearby.
- ✓ **Make Payment:** After getting services, the passenger makes payment through the app

3.1.2. Driver App Module

- ✓ **Registration & Login:** when the driver first comes to the system it should be registered all of his statuses to identify and awareness of the driver to the system.
- ✓ **Pick-up & drop-off navigation:** In the driver app module the driver uses a Googlemap to navigate pickup and drop-off location.
- ✓ **Get Vehicle:** If driver is not owner of vehicle he can get it from organization.
- ✓ **Get Notification:** The driver is notified if someone requests access from him.
- ✓ **Confirm Booking:** If the driver accepts the passenger's requests then he can confirm it.
- ✓ **Accept Booking:** the driver accepts the passenger's requests.

Android Based Taxi Booking System

3.1.3. Administration Module

Admin

- ✓ **Driver management:** This module controls and manages driver's status like pick-up and drop-off location, driving licenses, registering new drivers, and so on.
- ✓ **Vehicle management:** It controls and manages the vehicle (taxi) status like which taxi is given to which driver, code (Targa number), registering new taxi, and so on.
- ✓ **Payment management:** It controls the payment mechanism that is whether the passenger's paid successfully or not.
- ✓ **Generate report:** This module sends a report to passengers when they paid and sends a report to driver if there is change of fare tariff.
- ✓ **Tracking trips:** In this module, the administrators manage the passengers and location and assign which driver to which passenger by their pickup location. And the driver is assigned to a passenger through its nearby location.
- ✓ **Fare estimation:** In here the price should be calculated based upon the length of distance the customers traveled.
- ✓ **Account management:** managing and controlling accounts within the organization.

View scheduled taxi: checking and managing taxi requests comes from customers & transfers requests to the dispatcher.

Dispatcher

- ✓ **Register/Login:** These are the peoples who are employed in the organization and registered by the administrator.
- ✓ **Tracking trips:** track the location of passengers and drivers, when there is a service request from customers.
- ✓ **Manage service requests:** controlling and managing service requests which have to be delivered.
- ✓ **Assign drivers for Passengers:** Assigning of drivers to customers through their distance and nearby by location.
- ✓ **View request history:** checking and updating histories that are

Android Based Taxi Booking System

requested by the customers and the provider's/driver.

- ✓ **Assign Vehicle for Driver:** Assigning of vehicle to driver if the driver is not owner.
- ✓ **View scheduled taxi:** checking and managing taxi requests comes from customers& transfers requests to the dispatcher.

3.2. Non-Functional Requirements

In this section, we have discussed the quality attributes in which our system should have to perform the functional requirements. Non-functional requirements according to this system are listed below:

3.2.1 User Interface and Human Factors

The User Interface of the proposed system is simple to understand; the user interface components are leading and action indicators. Thus, the users of the proposed system can easily perform the tasks that they intended to do. Every component of the user interface is arranged based on understanding our users in terms of their goals, skills, preferences, and tendencies. Therefore, anyone (Beginner, Intermediate, and Advanced) who can access a mobile phone also can access this system, because it doesn't have any complex structure.

3.2.2 Hardware Consideration

This system is integrated with any android device which has an API version is 16 and above (Android 4.0 and above devices). In addition to that, the system works with slower internet connections and needs devices that can support Global positioning system (GPS).

3.2.3. Security Issues

We worry about security on greater extent to enable our users to trust our system. No one can access our system unless he or she is privileged to use our system services. We encrypted password using SHA -256 encryption algorithm.

3.2.4. Performance Consideration

- ✓ **Response Time:** when accessing the system to get services it is fast to respond and take an action for passenger's requests. When passengers request

Android Based Taxi Booking System

booking to the system then dispatcher look up their pickup location then assign a driver and send a notification as a request is accepted. We use Caching that ensures fast delivery to visitors. Caching speeds up loading time.

- ✓ **Supplies:** It can support many passengers at a given time until all the drivers are taken by other customers.
- ✓ It can load an infinite number of requests because the system is available all the time.

3.2.5. Error Handling and Validation

- ✓ The system display message if the Passenger enters incorrect input into different fields.
- ✓ The system display message if the customer enters a large quantity that is greater than the existing product in-store.
- ✓ The passenger cannot book if a taxi is not available.
- ✓ The system display message if the Passenger tries to open the privilege page using the wrong password and wrong Passenger name.

3.2.6. Quality Issues

- ✓ **Reliability:** This system has developed through different programming languages and works effectively and efficiently to perform the given services to Passengers as the customer's needs.
- ✓ **Availability:** The system should always be available for access for 24 hours, 7 days a week.
- ✓ This system is an automated system, so it has many advantages to Passengers by reducing wastage of time, ease of access, availability, and so on. Passengers are highly benefited by the system.

3.2.7. Backup and Recovery

Our system should backup data using portable storage device and store in cloud so that it is used to restore the original data after a data loss event existing system.

3.2.8. Physical Environment

Android Based Taxi Booking System

This system is deployed or addressed to a particular organization that has its business conduct. And that company can deploy the different modules of the system to Passengers and other aspects of the organization. These modules can distribute to Passengers on different websites.

3.2.9. RESOURCE ISSUES

The system provides available resources for the community as well as the members of the firm at a specified time as needed. Some limitations that should be taken into consideration are related to the security and reliability of the system.

The system should be designed in such a way that a third party can't see or change data in any way. Such an intrusion could allow a third party to act as a taxi driver and could cause significant loss for the taxi organization. Furthermore, the system should be available at all times, and resistant to the database, network, and other failures. Such an error could cause a loss of communication between taxi drivers and dispatchers and should be treated accordingly.

3. 2.10. Documentation

The new system required full documentation, help contents, and tips to allow further maintainability and to support and guide passengers on how to use the new system. For a system user level document is provided that the user can read the document to know how to use the system and what functionality should the system support to the user. And also, the development process is documented such that user can read to know about the process and what type of model we used to develop this system.

CHAPTER FOUR

4 SYSTEM ANALYSIS

4.1. Introduction

After careful observing and analyzing the problem of this type mobile application our team decide to design and develop the android based Taxi Booking system as a solution. The new mobile application is a faster, accurate and a bit more personal application, specially designed to make your taxi reservation application is better and also provide a sustainable way to get taxi easily when you want to go anywhere.

The proposed application will use the major functionality of the prior related work application to advance the prior related work. This means that taxi booking provides information to the Passenger about the current location they exist and track the nearest taxi based on transportation mode.

4.2. System Model

System Model is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. It is about representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML). Models help the analyst to understand the functionality of the system; they are used to communicate with customers.

For this project development in we used UML (unified model language) to specify the work of the proposed system diagrammatically through different angles. The UML (unified model language) development approach gives easier and simple, small and manageable components so that it reduces the vague appearance of the big problem. The system model document describes all data, functional and behavioral requirements of the software under production or development.

This model can explain the system from different perspectives: -

An external perspective, where we model the context or environment of the system,

An interaction perspective, where we model the interactions between a system and its environment, or between the components of a system,

Android Based Taxi Booking System

A structural perspective, where we model the organization of a system or the structure of the data that is processed by the system,

A behavioral perspective, where we model the dynamic behavior of the system and how it responds to events.

We used five types of UML diagrams that are the most useful for system modeling: Activity diagrams; which show the activities involved in a process or in data processing, Use case diagrams; which shows the interactions between a system and its environment, Sequence diagrams; which show interactions between actors and the system and between system components, Class diagrams, which shows the object classes in the system and the associations between these classes, State diagrams, which shows how the system reacts to internal and external events [3].

The following lists of objects are System Models for Taxi Booking System.

- ✓ Use Case Diagram
- ✓ Use Case diagram list and their details description
- ✓ Description of actors
- ✓ Activity diagram
- ✓ Class diagram
- ✓ Sequence diagram
- ✓ Passenger Interface prototyping diagram

4.2.1 Use Case Model

This Use case Model shows that Administrator, Dispatcher, Driver and Passengers are interact with the system. And all those actors have to make their own role to the system with respect to different classes of objects.

Use Cases

- ✓ Login
- ✓ Register Passenger
- ✓ Make payment
- ✓ Send Request
- ✓ Cancel Booking

Android Based Taxi Booking System

- ✓ Navigate pickup & drop-off
- ✓ Calculate fare
- ✓ Accept Booking
- ✓ Confirm Booking
- ✓ manage Account
- ✓ Assign Vehicle
- ✓ View Scheduled Taxi
- ✓ View Request History
- ✓ Track trip
- ✓ Assign Driver
- ✓ Manage Vehicle
- ✓ Manage Dispatcher

4.2.2. Use Case Diagram

Use case diagram is a graphical representation of use case model of our project. In our system there are four actors and twenty-five attributes which satisfies and fulfills all the system requirements. Each actor has their attributes to identify the functionalities of each actor on the system.

Android Based Taxi Booking System

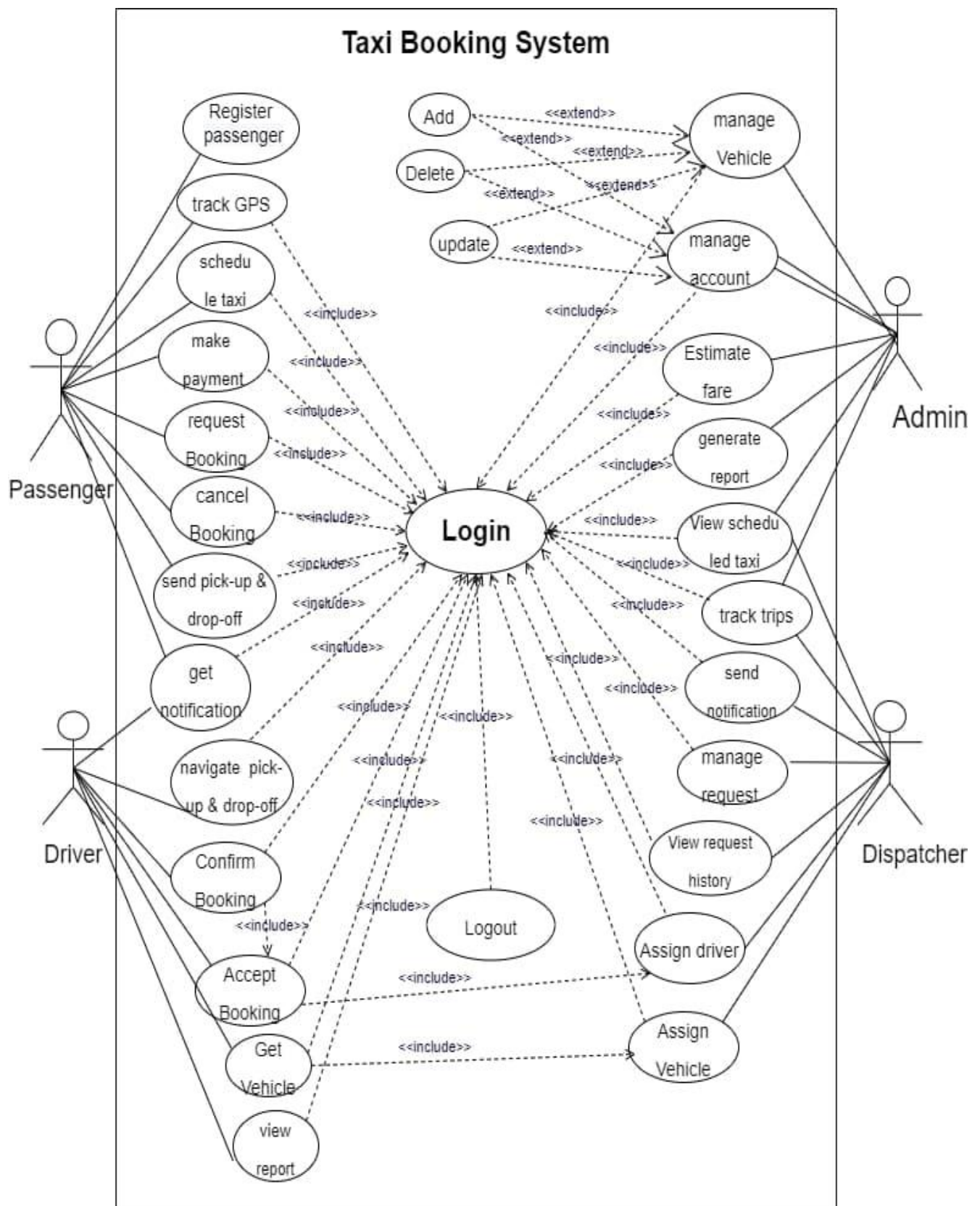


Figure 4. 1: Use Case Diagram for Android Based Taxi Booking

Android Based Taxi Booking System

4.2.2 Use Case Description

Table 4. 1: Use Case Documentation for Login

Section	Purpose	
Use case number	UC 01	
Name	Log in	
Actor	admin, driver, Passenger, and dispatcher	
Description	This use case describes how admin, driver, Passenger, and dispatcher to login into the System.	
Goal	To login into the system	
Precondition	The actor is on the homepage	
Post condition	If the use case was successful, the actor is now logged into the system. If not, the system state is unchanged.	
Basic course of action		
	Actor Action	System Response

Android Based Taxi Booking System

	1 The admin, Passenger, dispatcher and driver is on the home page to login to the system. 3. The admin, Passenger, dispatcher and driver enters Passenger name and password, Click on Login Button.	The system promotes the admin, Passenger, dispatcher, and drivers to enter Passenger name, Password. The system verifies that all the field have been filled out and valid. The actor successfully logged in the system. 6. use case exit
Alternate course of action	A6 If all fields are not filled out and not matched to their name and password then the system notifies the actor a message check your name or password and then goes back or returns to step 4 of basic course of Action to enter again.	

Table 4. 2: Use Case Documentation for Manage Vehicle

Section	Purpose
Use case no	UC 02
Name	manage Vehicle
Actor	Admin

Android Based Taxi Booking System

Description	Allow admin to add, delete and update Vehicle	
Goal	To manage Vehicle	
Precondition	Login	
Post condition	Registered	
Basic course of action	Actor Action	System report
	1 The administrator wants to manage Vehicle The admin enters or fill necessary information about Vehicle.	2 The system allows admin to enter Vehicle's information. The system verifies whether all the field have been filled out and valid. 5 The system notifies if Vehicle's registration is successful 6 Use case ends
Alternate course of action	A.5 The system indicates the administer that he/she entered an invalid information or misses something. A.6 the use case continues step3 and the basic course of action ends.	

Android Based Taxi Booking System

Table 4. 3: Use Case Documentation for Manage Account.

Section	Purpose
Use case no	UC 03
Name	Manage Account.
Actor	Admin
Description	Allow admin and Dispatcher control and manage over the account.
Goal	To manage and control account
Precondition	Admin should login
Post condition	Get full permissions on account to manage and control.
Basic course of action	The admin want to manage account. 2 The system allows admin to manage account information. 4 The system notifies that the admin manage and control account is successful Use case end
Alternate course of action	A.5 the use case continues step 2 , 3,4,5 and use case ends

Android Based Taxi Booking System

Table 4. 4: Use Case Documentation for Tracking Trips.

Section	Purpose	
Use case no	UC 04	
Name	tracking trips	
Actor	Dispatcher, Driver, passenger	
Description	Dispatcher, Driver and Passenger can check and look after tracking trips of drivers and passengers.	
Goal	To control and look after the way that the driver and passenger has assigned.	
Precondition	The Driver, Dispatcher and passenger should Login, enable GPS	
Post condition	Management on the location tracking	
Basic course of action	Passenger Action	System report
	1 The Dispatcher, Driver and passenger wants to look and manage tracking trips 3 Dispatcher, Driver and passenger should be logged into the system.	2 The system allows Dispatcher, Driver and passenger see and manage tracking trips. 4 The system verifies and shows all tracking trips information. 5 The system notifies tracking trips when login is successful Use case end

Android Based Taxi Booking System

Alternate course of action	A.5 the system indicates the Passenger that he/she enters invalid information. the use case continues step3, 4, 5 and use case end.
----------------------------	---

Table 4. 5: Use Case Documentation for Assign Driver.

Section	Function	
Use case no	UC 05	
Name	Assign Driver	
Actor	Dispatcher	
Description	Dispatcher should Assign Driver to give services for passenger's request.	
Goal	To Assign Driver for passenger	
Precondition	Login, booking request, tracking trips	
Post condition	Assign driver to each request by their nearest location.	
Basic course of action	Passenger Action	System report

Android Based Taxi Booking System

	1. The Dispatcher wants to Assign driver to each request respectively. 3. The Dispatcher can control over the driver information's and can assign driver to passengers based on the nearest location.	2 The system shows all driver information's to Dispatcher. 4 The system verifies that the driver is assigned to the passengers correctly. 5. The system notifies that the Driver Assigning is successful Use case ends
Alternate course of action	The system indicates when invalid address gets caught, A.6 the use case continues step3 and the basic course of action ends.	

Android Based Taxi Booking System

Table 4. 6: Use Case Documentation for Pickup and Drop off Location

Section	Purpose
Use case number	UC 06
Name	pickup and drop off location
Actor	Drivers and Passenger
Description	Passenger send pickup and drop off location and driver view pickup and drop off location of Passengers
Goal	To match driver and Passengers
Precondition	UI01
Post condition	The driver gets pickup and drop off location of Passenger
Basic course of action	1 the drivers want to view Passenger's pickup and drop off location 2 The system allows drivers to view Passenger's pickup and drop off location 3 Use case ends
Alternate course of action	

Android Based Taxi Booking System

Table 4. 7: Use Case Documentation for Book Request

Section	Purpose	
Use case number	UC 07	
Name	book request	
Actor	Passenger	
Description	The Passenger can send book request online on his mobile device	
Goal	To order taxi online	
Precondition	UI01	
Post condition	Passenger booked taxi	
Basic course of action	Passenger Action	System Response
	1.Passengers want to make booking 3. Passenger fills up booking information. 4 The passenger submits booking.	2 The system allows passengers to enter basic booking information. 5 The system confirms booking. 6 The use case end.
Alternate course of action	A.5 the system indicates that passenger enters invalid information or jumps fields. A.6 the system continues at step three and four. Finally exit form use case.	

Section	Purpose	
Use case number	UC 08	
Name	Payment	
Actor	Passenger	
Description	The Passenger can pay by using mobile device online	
Goal	To make payment online	
Precondition	UI01, view fare calculation	
Post condition	Make payment for the service online	
Basic course of action	Passenger Action	System Response
	1. Passengers want to pay for the service 3.Passenger fill the required form to get billing system and transfer the money that shown at the fare calculation	2.The system leads the Passenger to payment system and shows the payment options 4. The system the system generates report 5. The use case end.

Android Based Taxi Booking System

Alternate course of action	the Passenger inserts invalid information and return 3
----------------------------	--

Android Based Taxi Booking System

Table 4. 9: Use Case Documentation for Cancel Booking

Section	Purpose
Use case number	UC 09
Name	Cancel booking
Actor	Passenger
Description	Allows Passengers to cancel booking if there is booked before.
Goal	To cancel booking
Precondition	Make booking
Post condition	Booking canceled
Basic course of action	1. passengers want to cancel booking. 2 The system allows dispatcher and passengers to enter (“cancel booking form”). 3 Passengers enter booking code. 4 The system verifies the booking code. 5 Use case ends.
Alternate course of action	A.3 the system notifies that the Passenger enters invalid booking code. the system continues to step 4 and use case end.

4.3 Object Model

Object Model is a logical interface, software or system that is modeled through the use of object-oriented techniques. It enables the creation of an architectural software or system model prior to development or programming. In our system we use object model like class diagram and data dictionary.

4.3.1 Class Diagram

Class diagrams should show the objects the system is comprised of and how they are interrelated. Class models contain a wealth of information; it can be used for both the analysis and design of systems. To create and evolve a class model:

Android Based Taxi Booking System

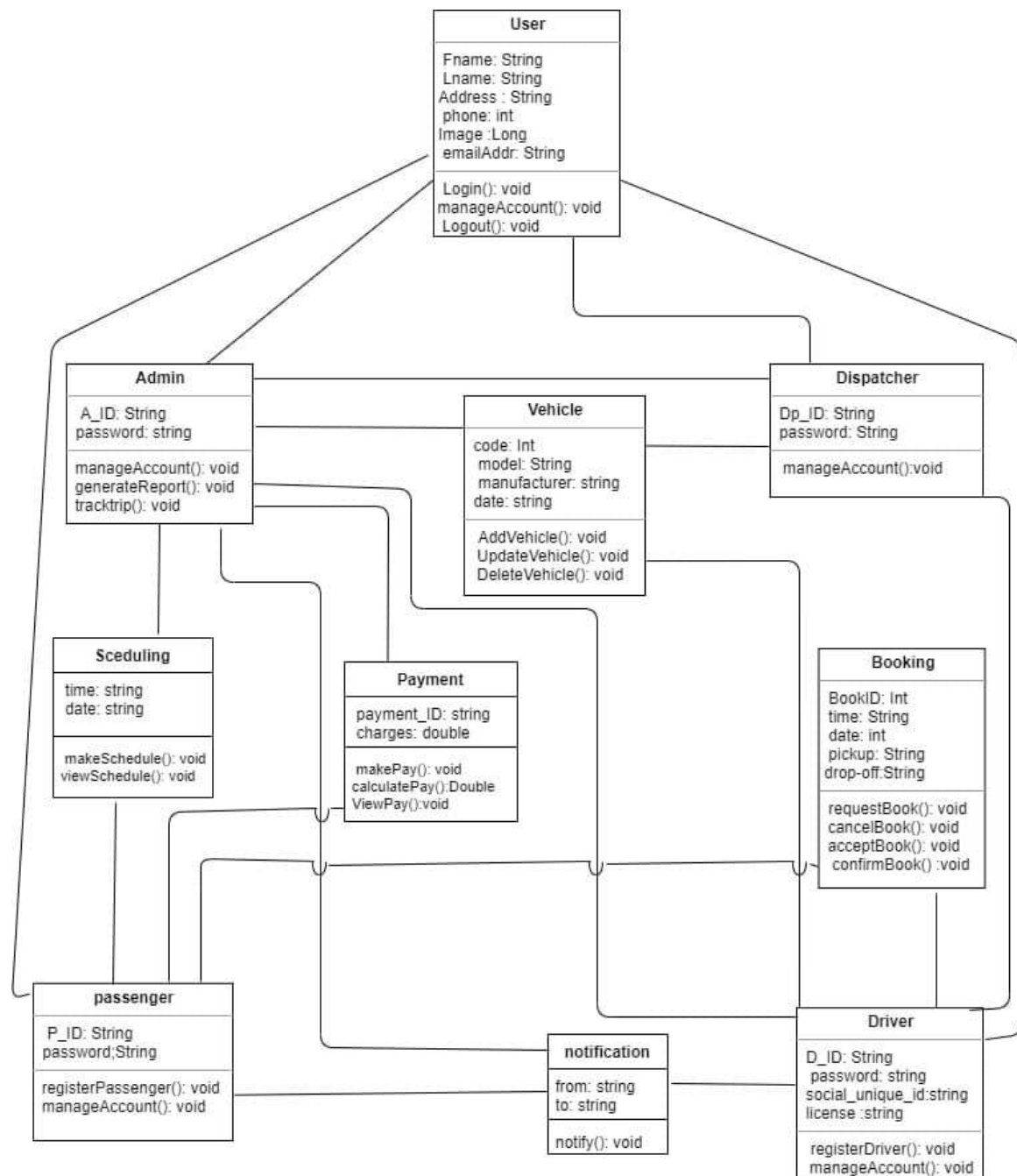


Figure 4. 2: Class Diagram for Android Based Taxi Booking System

Android Based Taxi Booking System

4.3.2 Data Dictionary

A data dictionary is a file or a set of files that includes a database's metadata. The data dictionary holds records about other objects in the database, such as data ownership, data relationships to other objects, and other data. In our system the following table describe datadictionary.

Table 4. 10: Data Dictionary for Admin

Attribute	Data type	Data size	Constraint
ID	string	30	Primary key
Name	String	50	NOT NULL
Email	String	50	NOT NULL
Password	String	50	NOT NULL
Image	Long	100	NOT NULL

Table 4. 11: Data Dictionary for Passenger

Attribute	Data type	Data size	Constraint
ID	String	50	Primary key
name	String	50	NOT NULL
Email	String	50	NOT NULL
Password	String	50	NOT NULL
phone	Int	50	NOT NULL
Social unique ID	String	50	

Android Based Taxi Booking System

Table 4. 12: Data Dictionary for vehicle

Attribute	Data type	Data size	Constraint
code	int	8	Primary key
model	string	50	
manufacturer	string	50	
date	string	30	

Table 4. 13: Data Dictionary for Driver

Attribute	Data type	Data size	Constraint
ID	string	50	Primary key
Fname	String	50	
Lname	String	50	
Email	String	50	NOT NULL
Password	String	50	NOT NULL
Mobile	Int	50	NOT NULL

Android Based Taxi Booking System

Table 4. 14: Data Dictionary for Dispatcher

Attribute	Data type	Data size	Constraint
ID	String	50	NOT NULL
Name	String	50	NOT NULL
Mobile	Int	50	NOT NULL
Remember token	String	40	

4.4 Dynamic Model

A dynamic model represents the behavior of an object over time. It is used where the object's behavior is best described as a set of states that occur in a defined sequence.

4.4.1 Sequence Diagram

Sequence diagrams illustrate how the different parts of a system interact with each other to carry out a function, and the order in which the interactions occur when a particular use case is executed. A sequence diagram shows object interactions arranged in time sequence.

Android Based Taxi Booking System

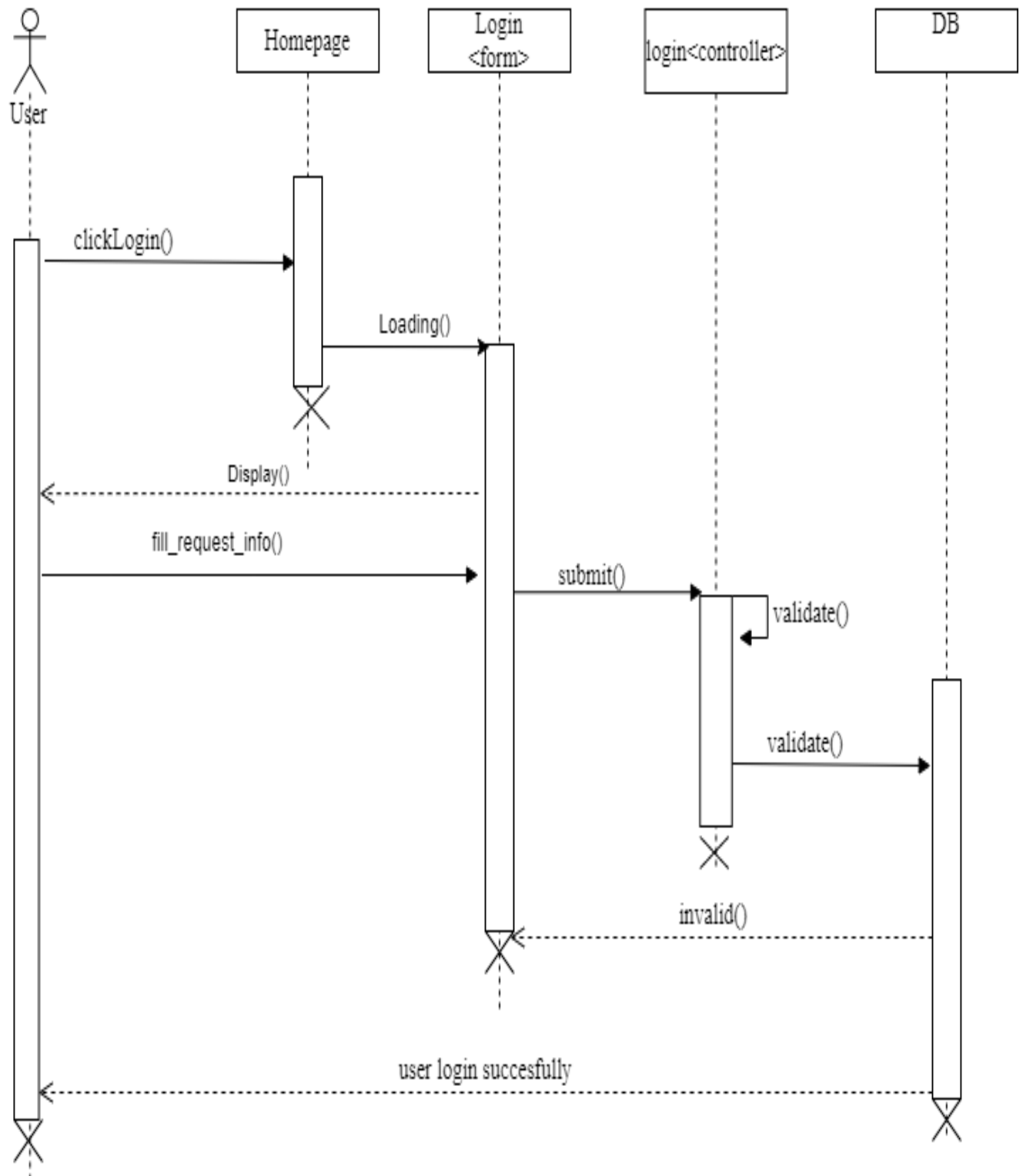


Figure 4. 3: Sequence Diagram for Log In.

Android Based Taxi Booking System

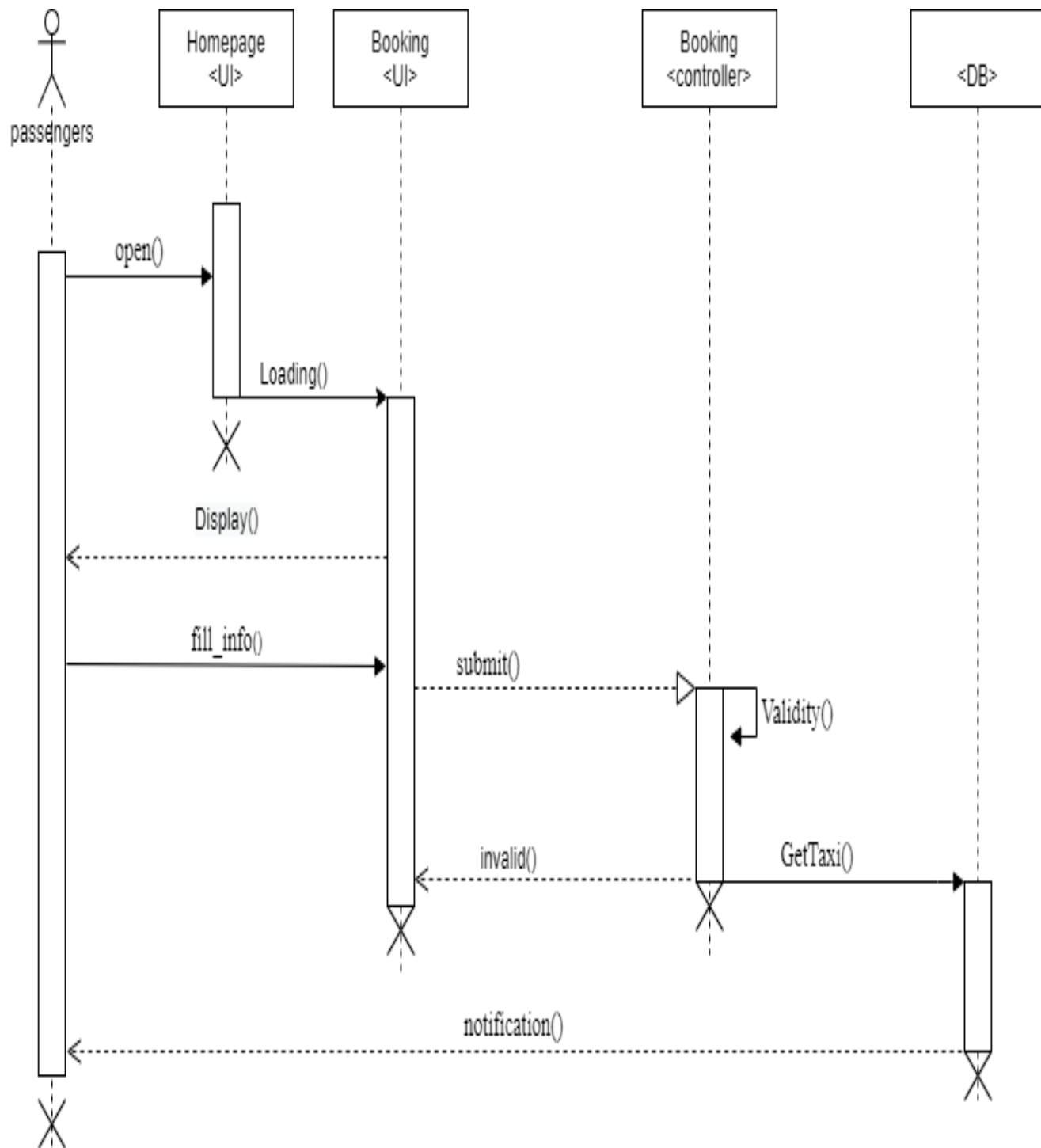


Figure 4. 4: Sequence Diagram for Booking

Android Based Taxi Booking System

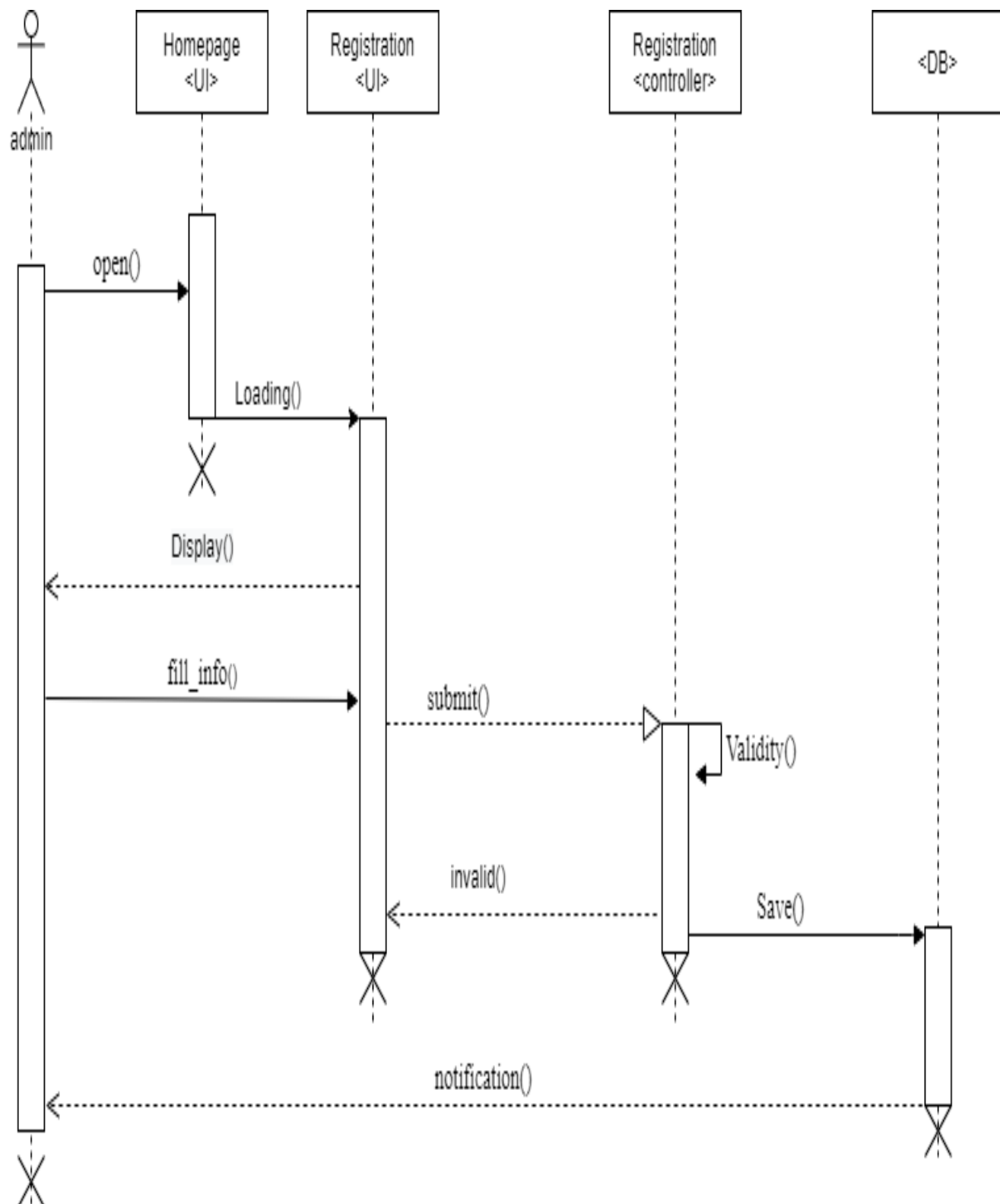


Figure 4. 5: Sequence Diagram to Registration

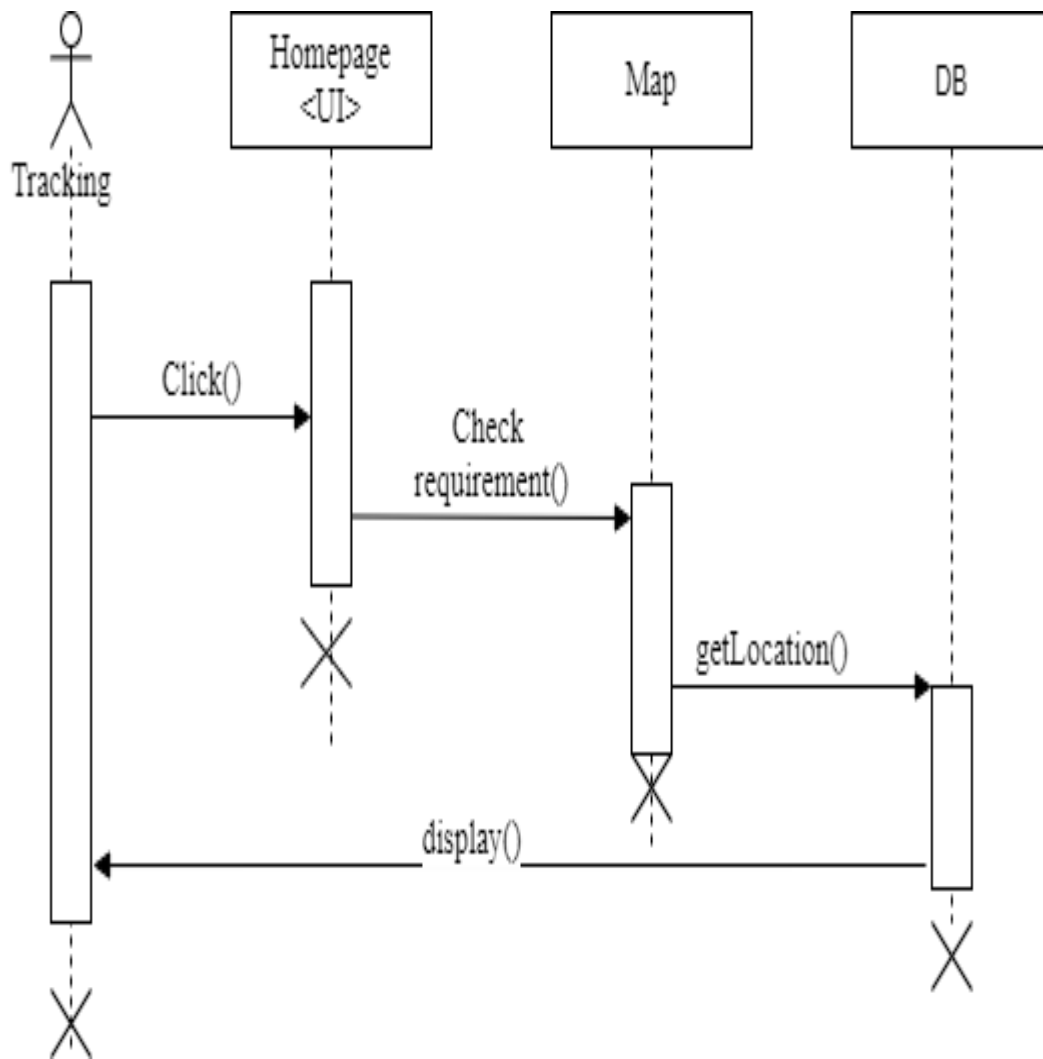


Figure 4. 6: Sequence Diagram for tracking

4.4.2. Activity Diagram

An Activity diagram is similar to a flowchart to represent the flow from one activity to another activity. Activity diagram focuses on the flow of activities involved in a single process. The Activity diagram shows how these single-process activities depend on one another.

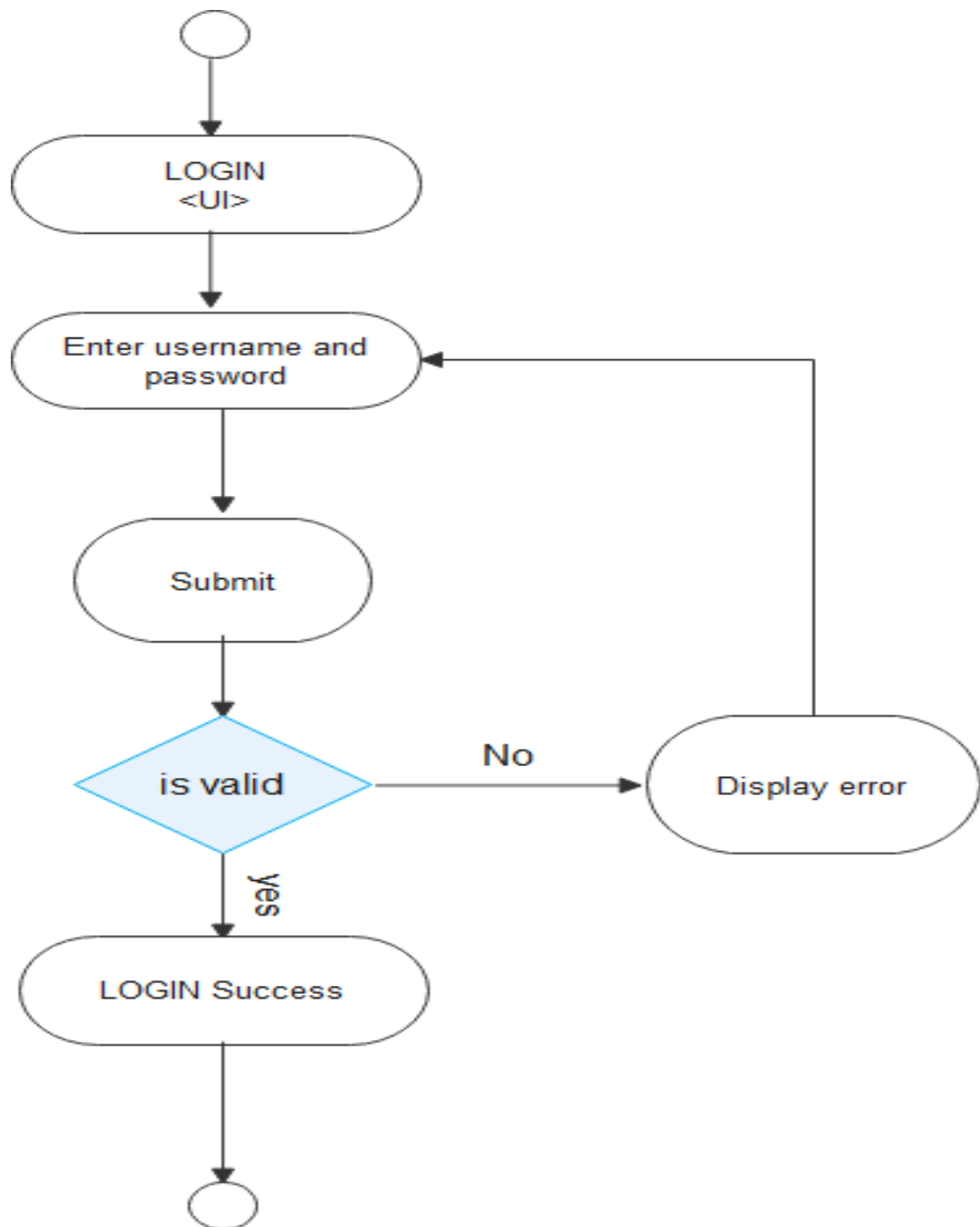


Figure 4. 7: Activity Diagram for Login

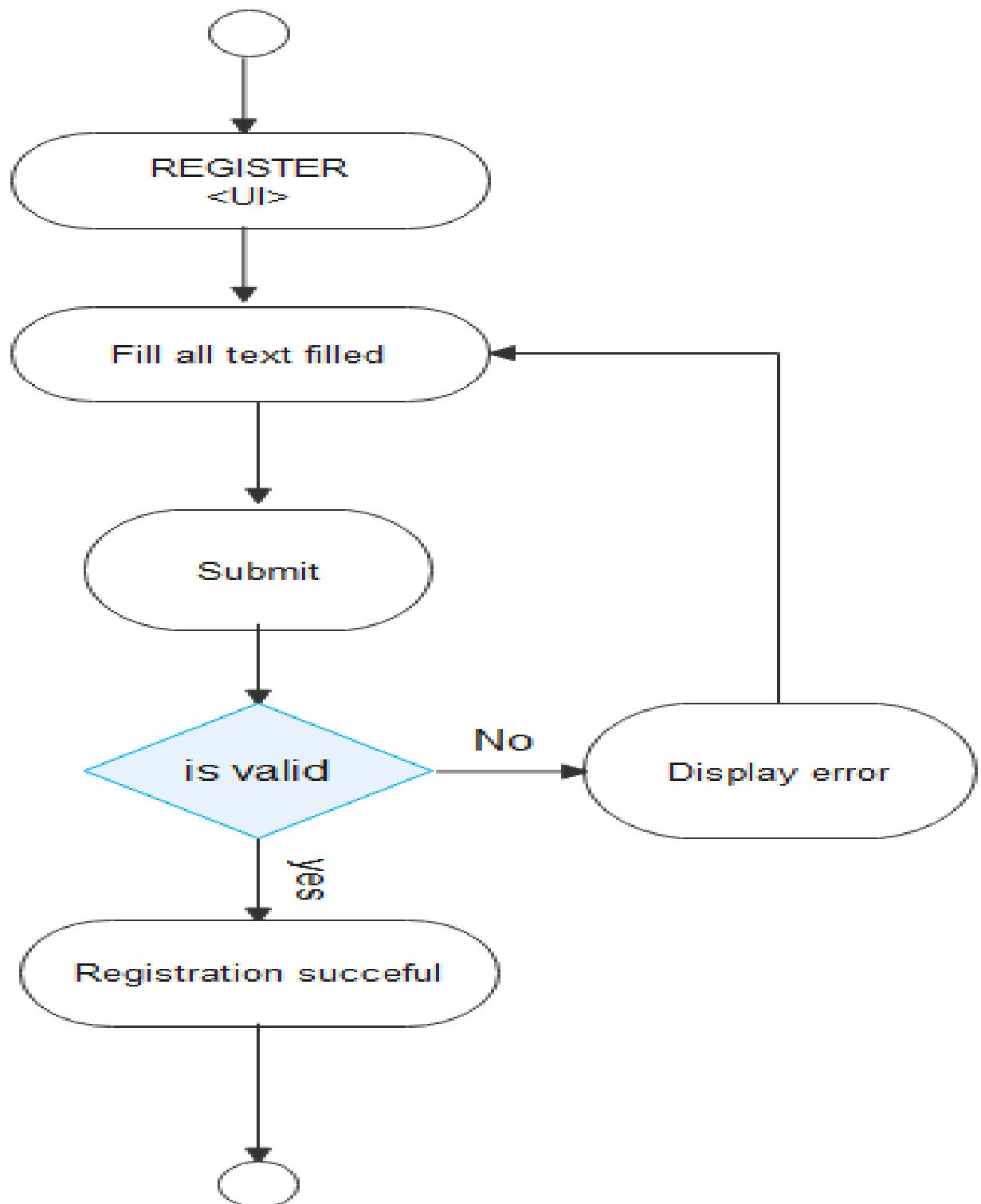


Figure 4. 8: Activity Diagram for registration

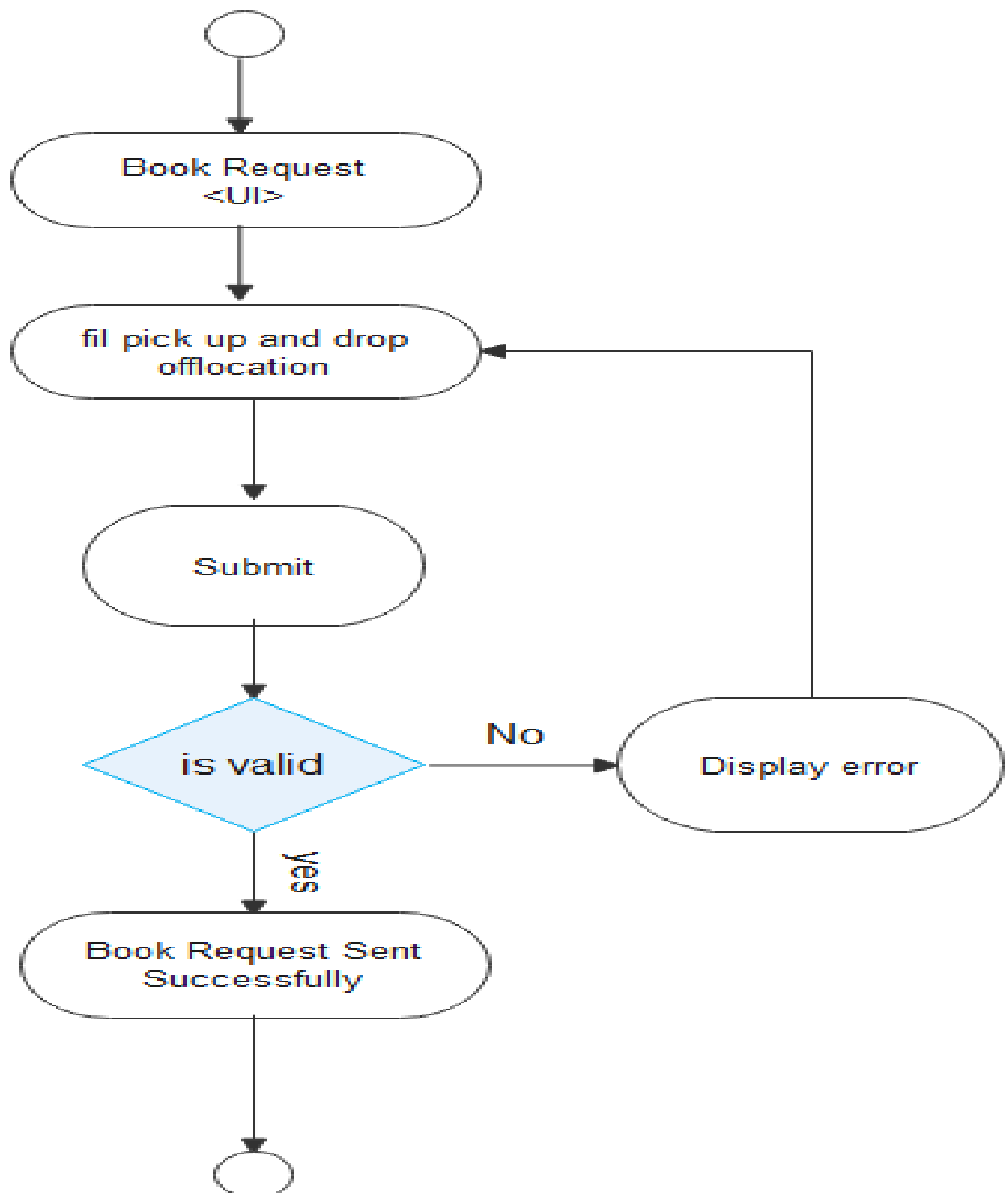


Figure 4. 9: Activity Diagram for Book Request

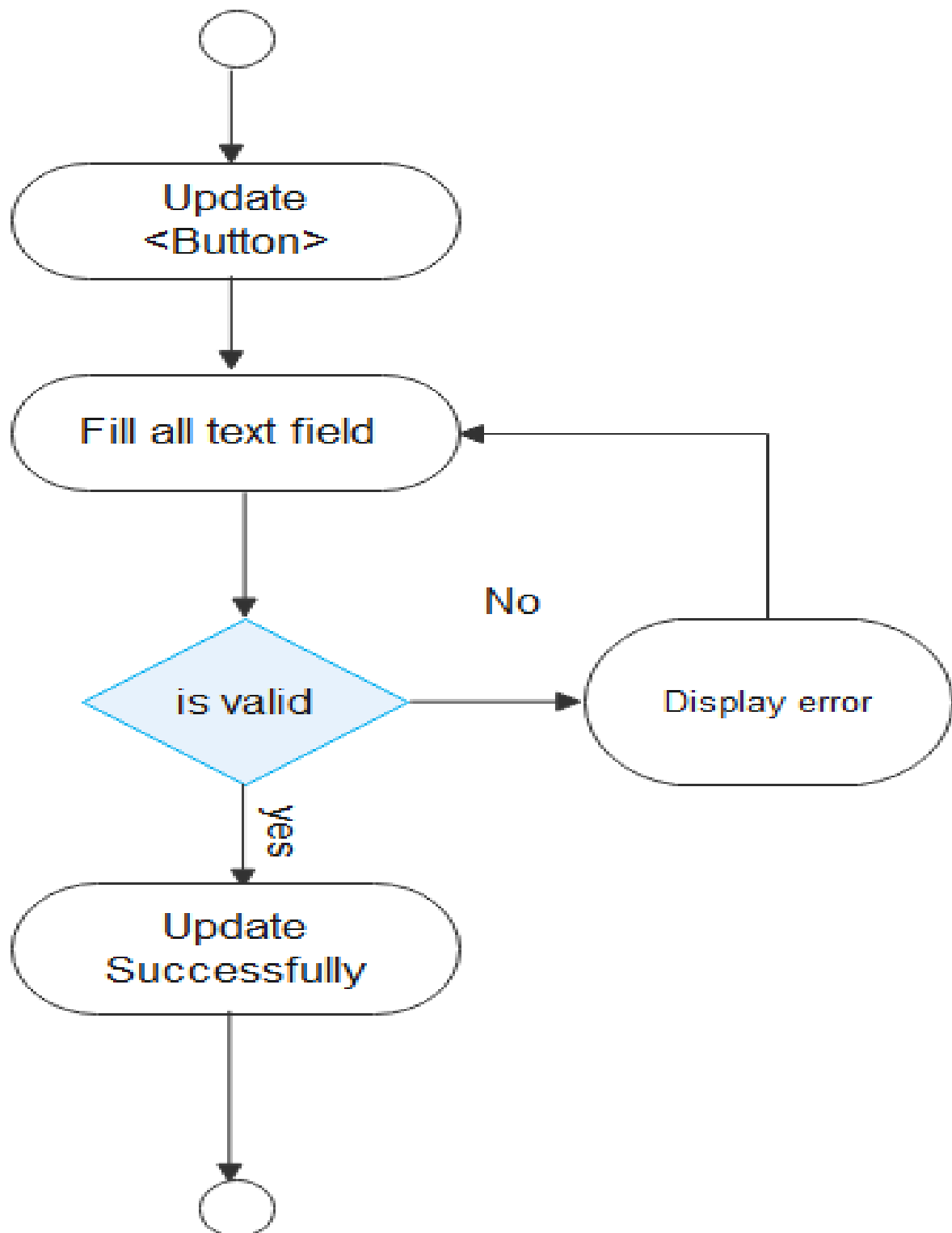


Figure 4. 10: Activity Diagram for Update

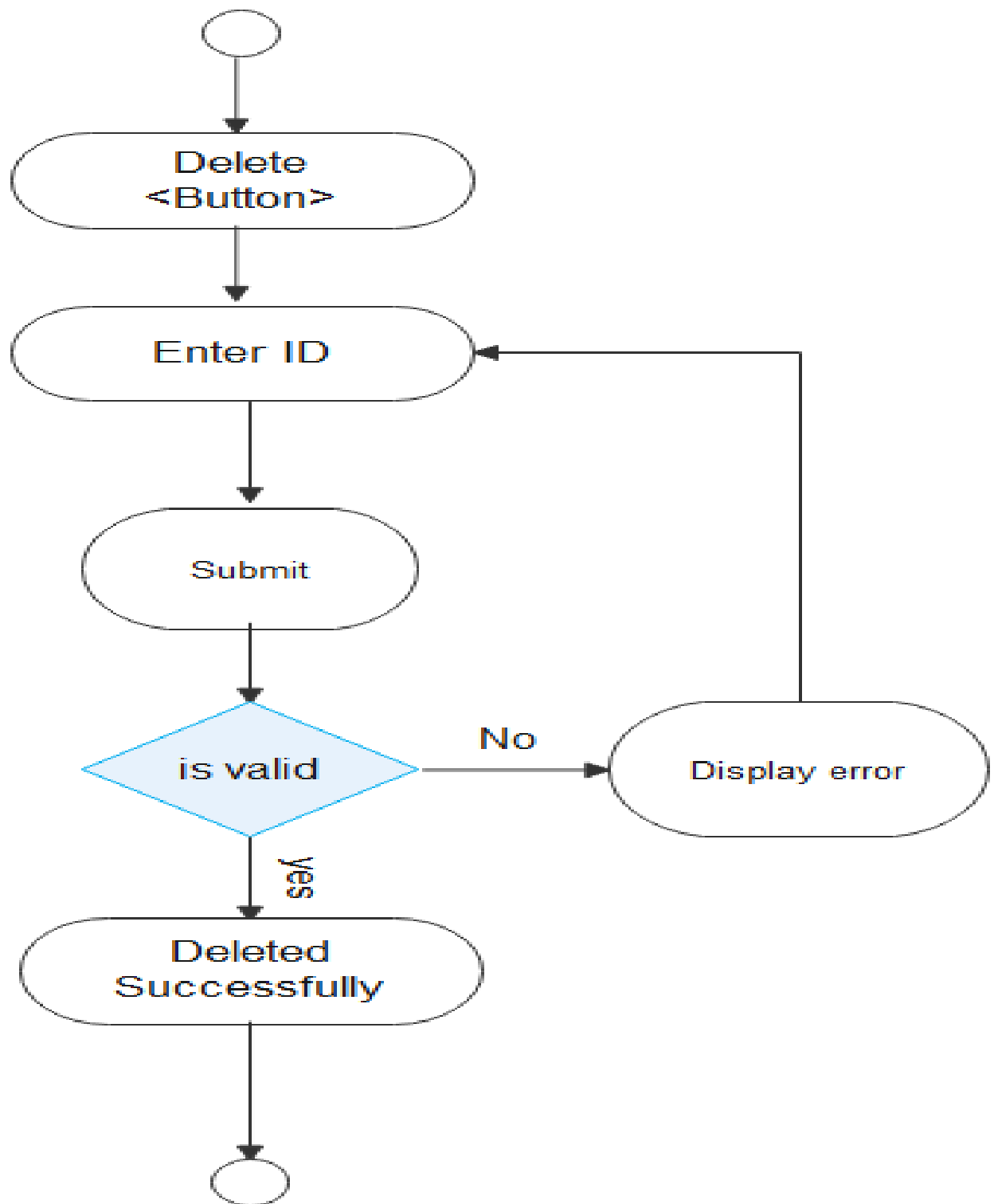


Figure 4. 11: Activity Diagram for Delete

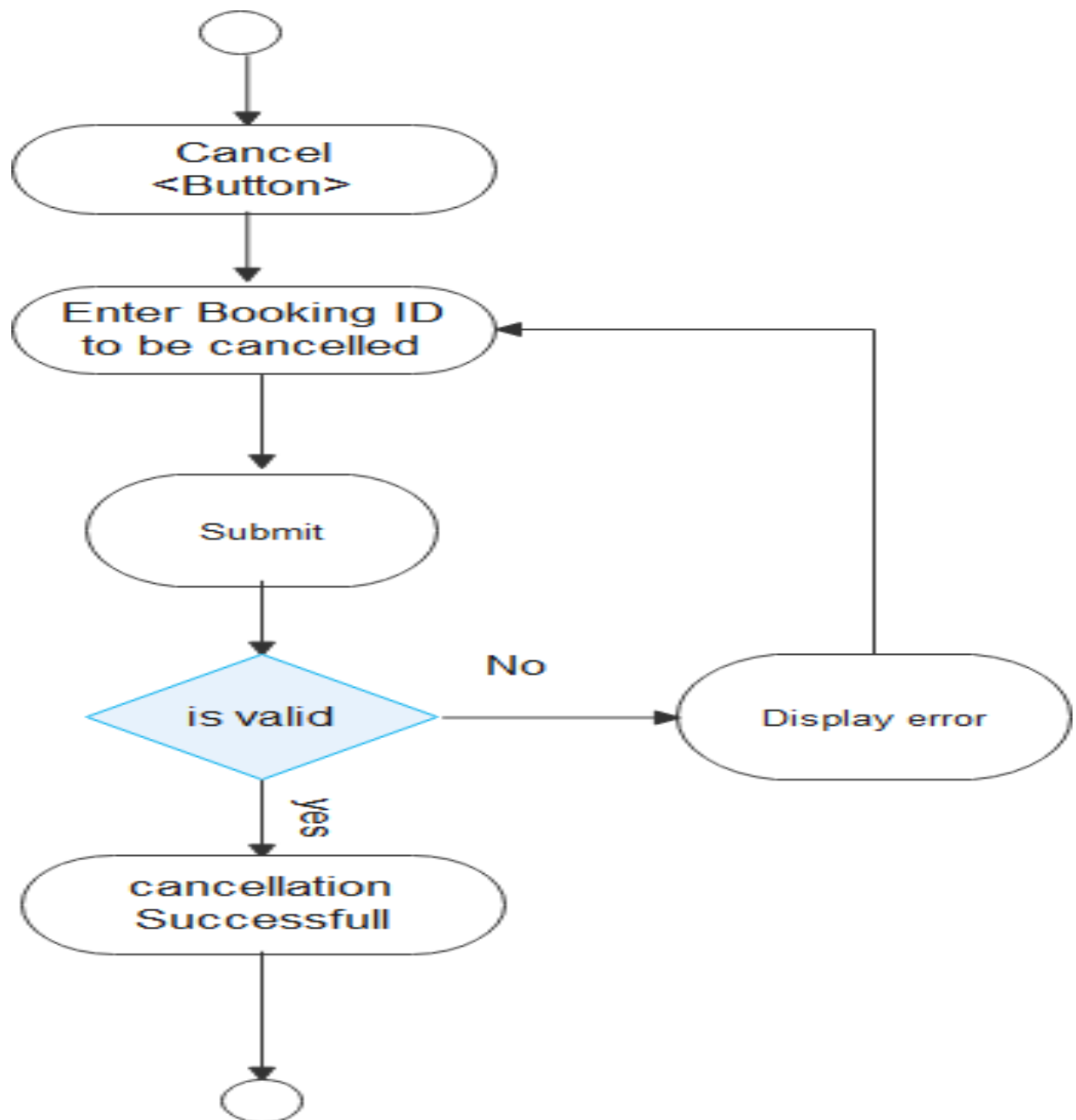


Figure 4. 12: Activity Diagram for Cancel

4.4.3. State Chart Diagram

State Chart Diagram is an illustration of the states an object can attain as well as the transitions between those states in the Unified Modeling Language (UML). State chart diagram describes the flow of control from one state to another state.

Android Based Taxi Booking System

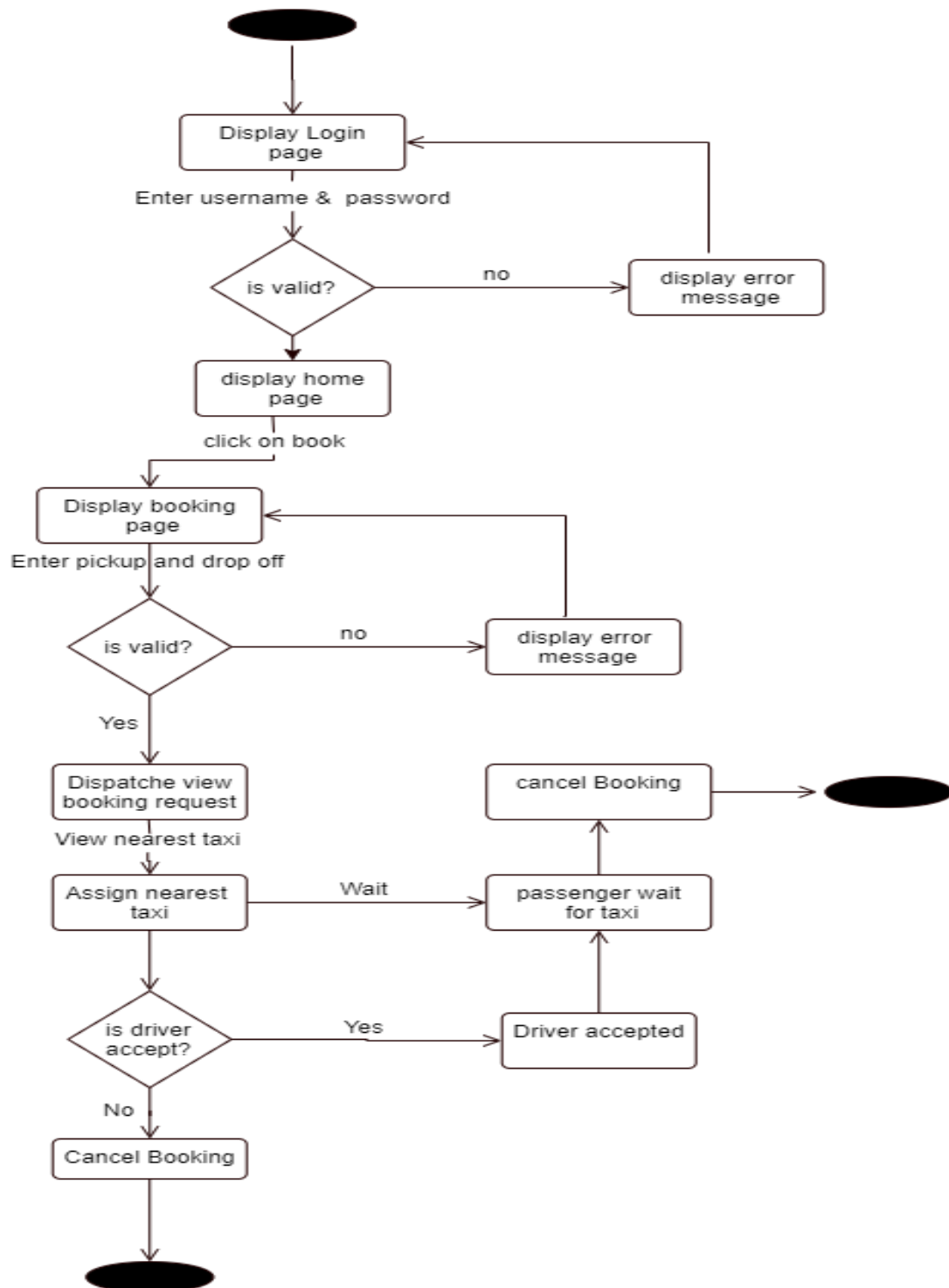


Figure 4. 13: State Chart Diagram for Booking

CHAPTER FIVE

5 SYSTEM DESIGN

The system design of this project defines the elements of the android-based taxi booking system like modules, architecture, components, and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing, and designing systems that satisfy the specific needs and requirements of the stakeholder and also defines the coherent and well running system. So, if the system is designed effectively, it will be easy to make changes to it.

5.1. Design Goals

The design goals are derived from the non-functional requirements of the system, which describes the qualities of the android-based taxi booking system that should be achieved and addressed during the design of the system.

The following are mentioned as the design goals of Android Based Taxi Booking System.

User interface: - The User Interface of the proposed system is simple to understand; the user interface components are leading and action indicators. Thus, the users of the proposed system can easily perform the tasks that they intended to do. Every component of the user interface is arranged based on understanding our users in terms of their goals, skills, preferences, and tendencies.

Hardware consideration: - the hardware requirements of our system are considerably affordable and accessible. The system will use computer devices, cellphones, and deployment servers. These devices are already available. No need of a super computer or complex electronics devices.

Client side

Android phones with at least 1GB RAM and 2GB memory.

Server side

At the server side the system requires server computer with a tera byte of storage capacity 8GB RAM memory. It also requires high processor (2.9 GHZ processor) for speedy response time.

Android Based Taxi Booking System

Security: - Unauthorized access to the system should be restricted because the proposed system verifies the users by their username and password to check whether they are authorized or not. The password is encrypted using SHA-256 to secure our system from guessing the password and to eliminate the security problems.

Step-by-step SHA-256 work as:-

Step 1 – Pre-Processing

Step 2 – Initialize Hash Values (h)

Step 3 – Initialize Round Constants (k)

Step 4 – Chunk Loop

Step 5 – Create Message Schedule (w)

Step 6 – Compression

Step 7 – Modify Final Values

Step 8 – Concatenate Final Hash

Backup and Recovery: - data backup and recovery are the process of backing up data in the event of a loss and setting up secure systems that allow us to recover data as a result. It requires the copying and achieving of computer data to make it accessible in case of data corruption or deletion. The system will use physical (copy of physical database files that stores database information in another location and forms the foundation of the database recovery mechanism) and logical data (contains the logical data that is extracted from the database and consists of tables, procedures, views and functions) backup and recovery mechanism

Quality issue: - Our system has high reliability due to real-time tracking. In the quality assessing the users will be involved by a feedback mechanism in which they can give comments on the system. The system should be available every time the Passenger needs to access it.

Performance consideration: - performance of our system is an efficient. The proposed system has a better response time and it can support many users concurrently

Error Handling and Validation: - The proposed system will check user inputs to the

Android Based Taxi Booking System

system to handle errors. It handles and shows errors in a user-friendly manner, without exaggerating the user.

Resource Issues: - Since the proposed system is android based system, it needs high computing power in order to behave efficiently and in a good performance. The performance of the proposed system is dependent on the computing power such as RAM, Processor, GPU of mobile and computer. Cost is difficult for us in order to buy materials like Hard Disk for backup files.

5.2. Proposed Software Architecture

The architecture which will be used for the proposed system is a three-tier architecture where the client or the Passenger side is a mobile phone containing Passenger interfaces like data entry interfaces, it is used to display information to the Passenger. Passenger directly interacts with the system through the interfaces on this layer.

The data layer or the database is responsible for storing all information needed for the system to function correctly.

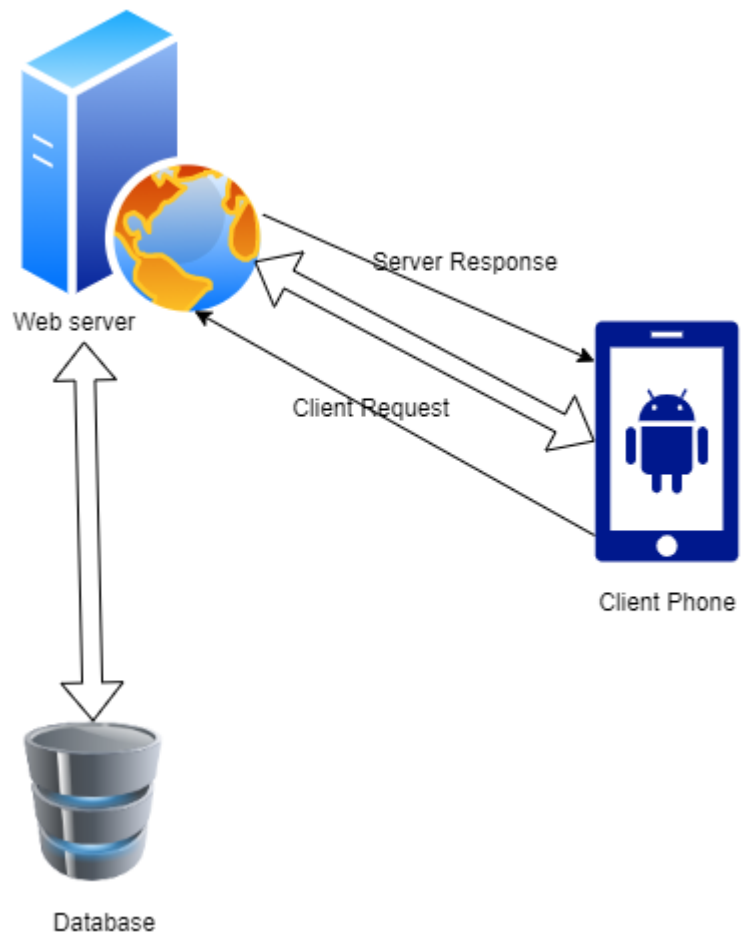


Figure 5. 1: Proposed Software Architecture

5.2.1. Subsystem Decomposition and Description

Subsystem decomposition is the process of dividing the Android Based Taxi Booking System into manageable subsystems from the analysis model of the system. Our goal in the subsystem decomposition is to reduce the complexity of the design model of ABTBS and to distribute the class of the system into large-scale and cohesive components. The different components of our system are likely to interact while the system is in operation to provide the expected services. In our system the following subsystems are available.

Booking management Subsystem

- ✓ Request Booking
- ✓ Accept Booking
- ✓ Confirm Booking
- ✓ Cancel Booking

Account management Subsystem

- ✓ Create account
- ✓ Delete account
- ✓ Update account
- ✓ View account

Notification Subsystem

- ✓ Send Notification
- ✓ view Notification

Payment Management Subsystem

- ✓ Make payment
- ✓ Calculate payment
- ✓ View payment

Tracking Subsystem

- ✓ Tracking trip
- ✓ Tracking GPS

Vehicle management Subsystem

- ✓ Add Vehicle
- ✓ Delete Vehicle
- ✓ Update Vehicle
- ✓ View vehicle

Android Based Taxi Booking System

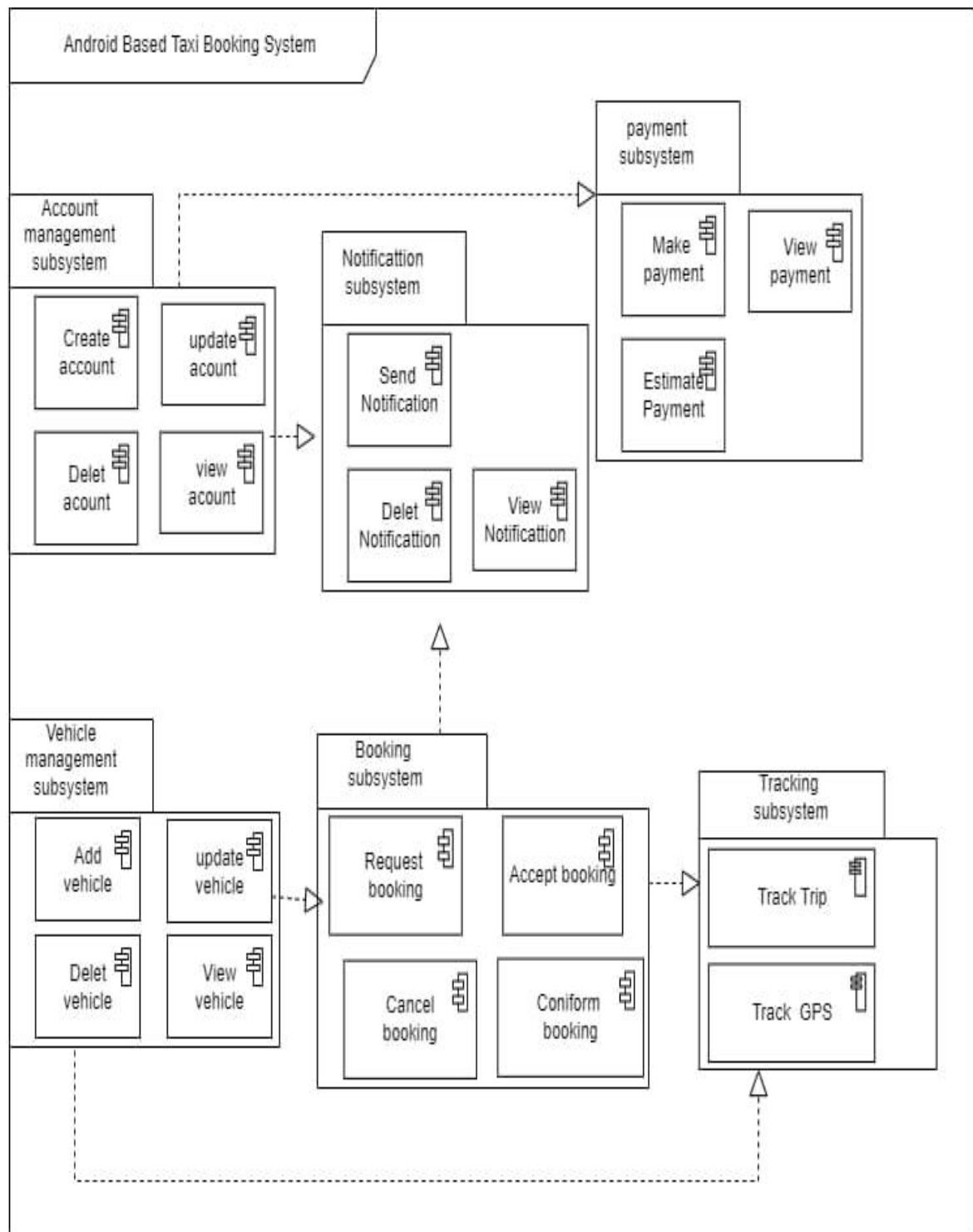


Figure 5. 2: Component Diagram for Android Based Taxi Booking System

Android Based Taxi Booking System

5.2.2. Hardware/Software Mapping

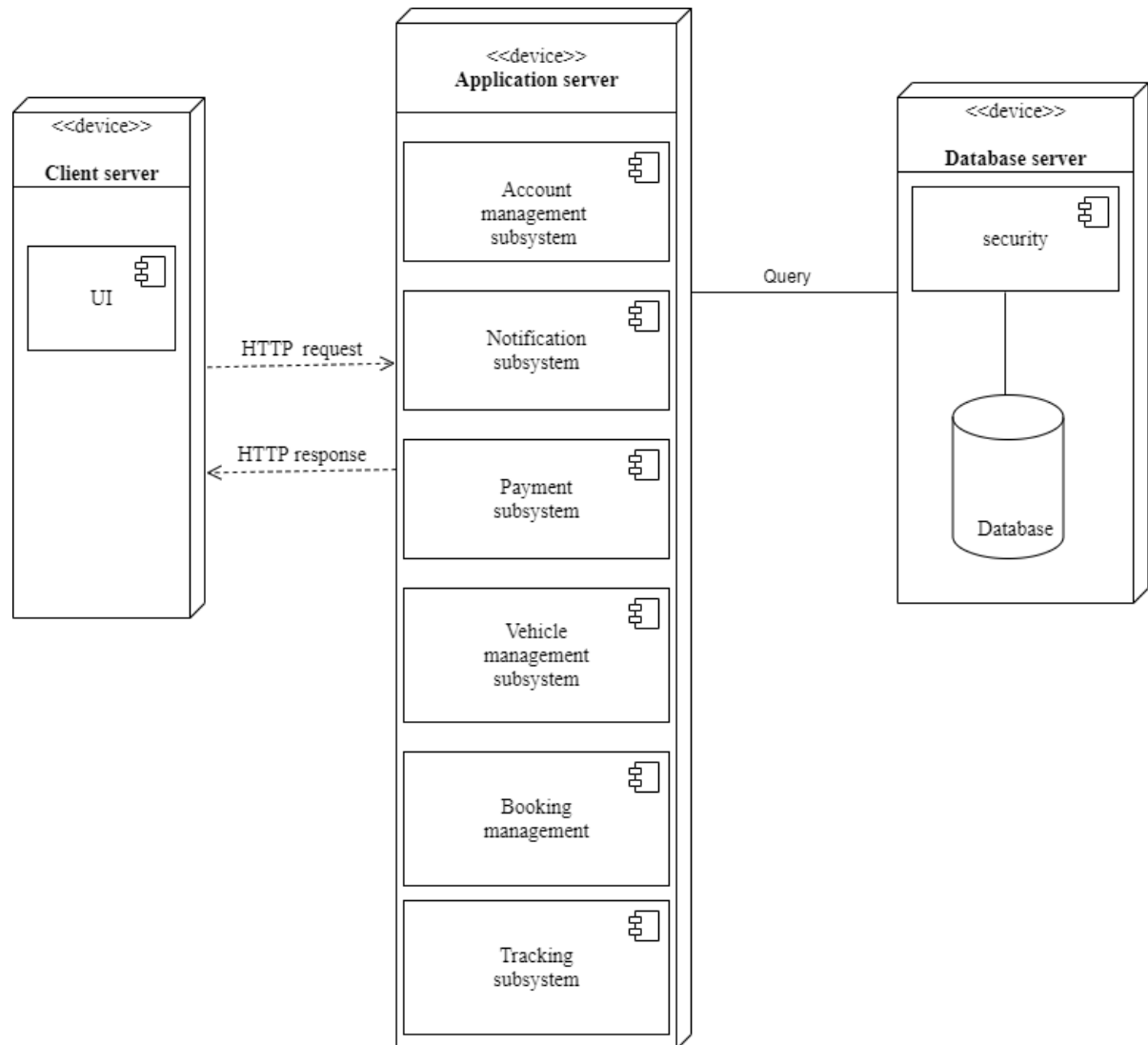


Figure 5. 3: Deployment Diagram for Android Based Taxi Booking System

Android Based Taxi Booking System

5.2.3. Detailed Class Diagram

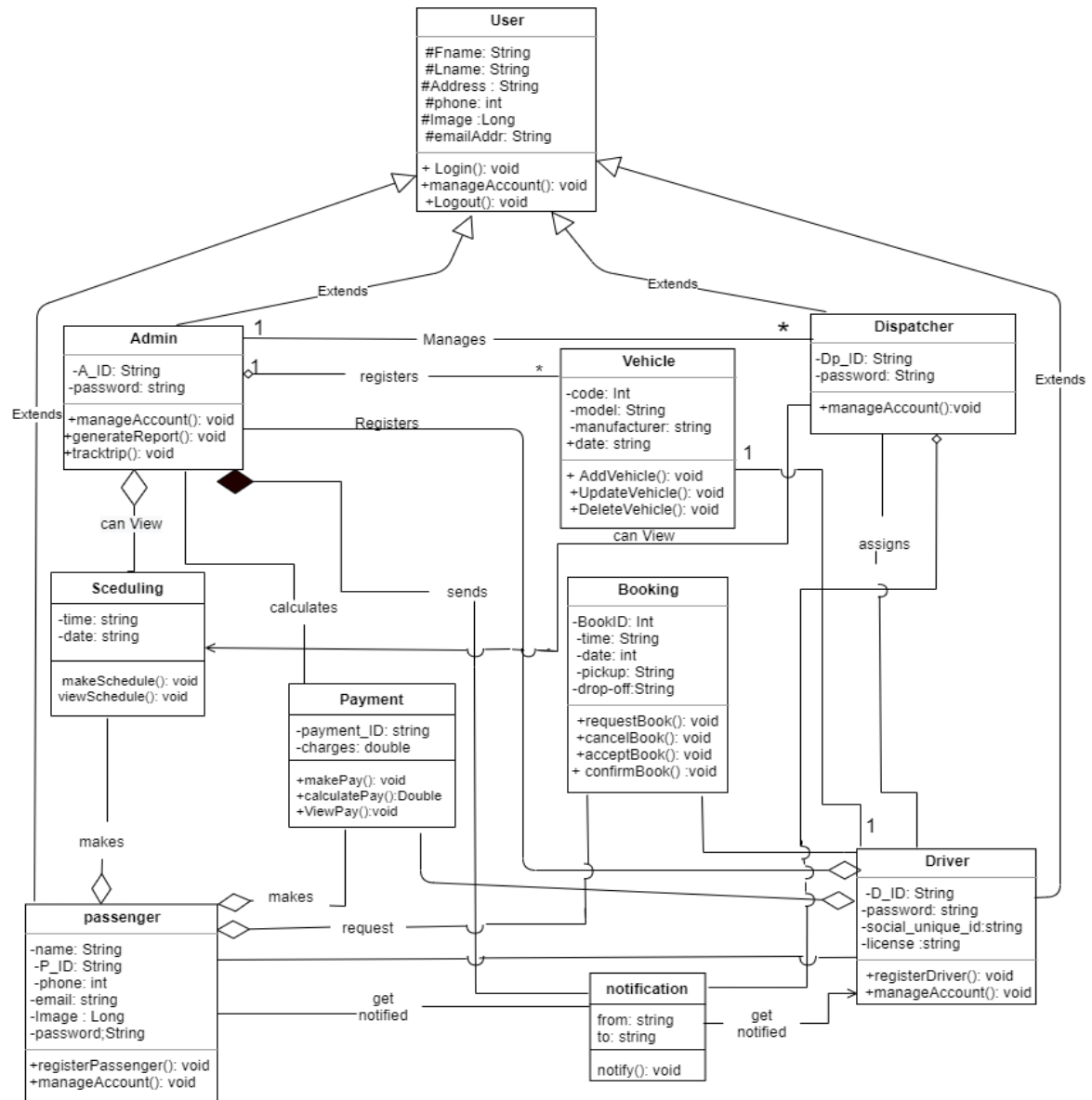


Figure 5. 4: Detailed Class Diagram for Android Based Taxi Booking System

5.2.4. Persistent Data Management

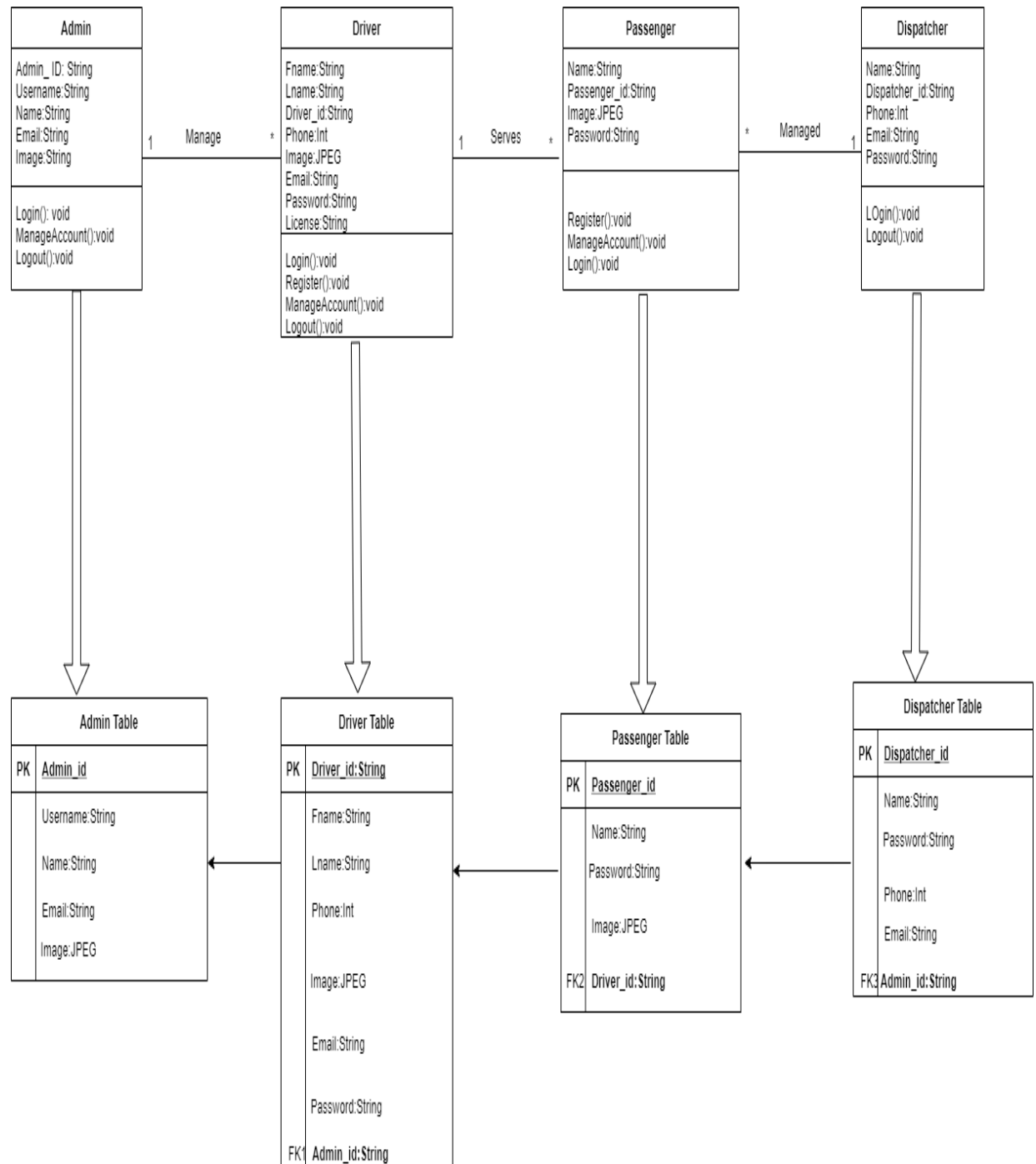


Figure 5. 5: Persistence Diagram for Android Based Taxi Booking System

Android Based Taxi Booking System

5.2.5 Access Control and Security

In the systems, different actors have access to different functionality and data. The system has different actors such as passengers, administrator and driver.

The passengers: Represents unauthenticated Passenger. It is used by without Login subsystems to book taxi. And also, they responsible to fill booking form, select, search booking.

Driver: Is an authenticate Passenger who is responsible to accept and reject booking, change taxi current status and cancel booking.

Administrator: Is an authenticate Passenger. It is used by login subsystem. He or she is responsible for managing drivers, taxi and control booking.

Table 5. 1: Access control

Actor	Operations				
	Manage User account	Manage Book	Vehicle	Notification	Report
Administrator	CRUD	-R--	CRUD	CRUD	CRUD
Dispatcher	-RUD	-R--	-R--	----	-R-D
Passenger	CRUD	CRU-	---	-R-D	-R-D-
Driver	-RUD	-UD	-R--	-R--D	- R-D-

5.3. Packages

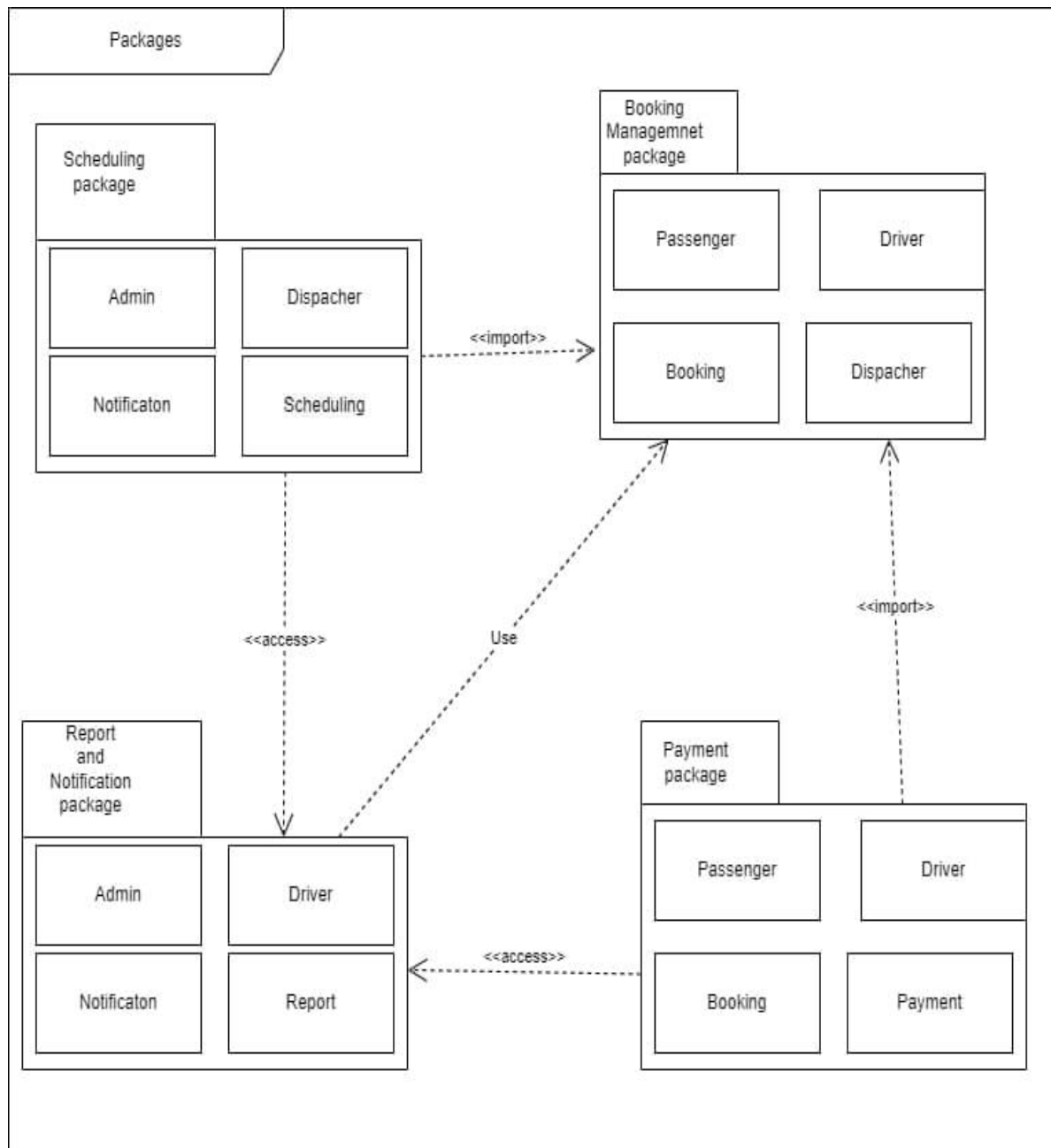


Figure 5. 6: Package Diagram for android based taxi booking System

5.4. Algorithm Design

Pseudo code algorithm for login

Begin

If (user click the Login button)

Android Based Taxi Booking System

Fill Login Form

If (valid)

 Redirect to authenticated page

 Allow to perform privileged task

Else

 Display error message.

 Redirect to login page

 Ask the user to refill the form

End if

End if

End

Pseudo code algorithm for booking

Begin

If user click on Book button

 Fill booking Form

 If valid

 if click submit button

 Assign nearest driver and forward driver information to passenger

 If click cancel button

 Return to homepage successfully

 Else

 Display error message and Redirect to book page

 Ask the user to refill the form

 End if

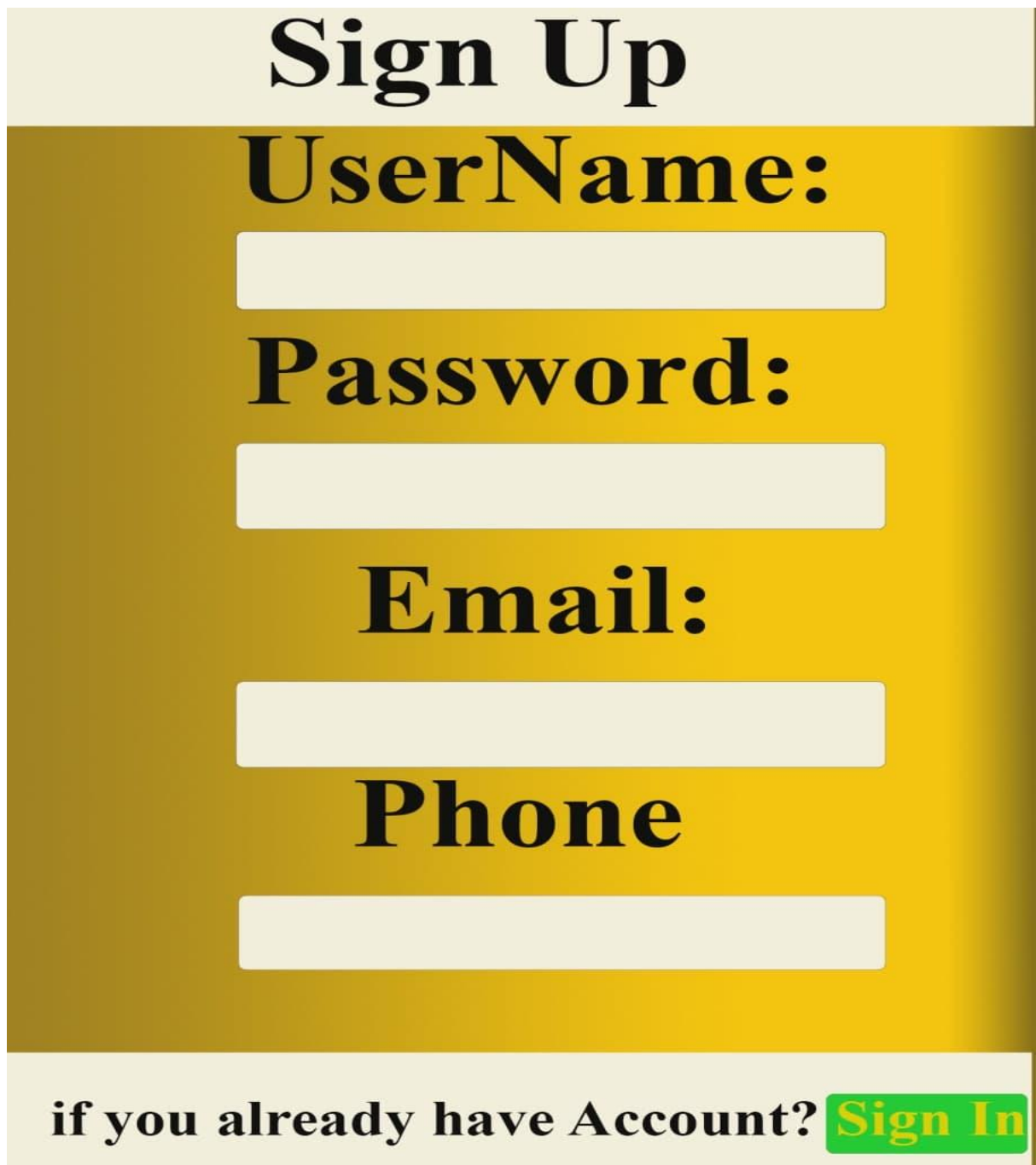
End if

5.5. User Interface Design

The goal of User interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing Passenger goals (Passenger-centered design) so in our project we have designed Passenger interfaces that increase the Passenger experience.



Figure 5. 7: Passenger Homepage



The image shows a sign-up form for a taxi booking system. It has a yellow background with a darker yellow header and footer. The form fields are white with rounded corners. The text is in a bold, black, serif font.

Sign Up

UserName:

Password:

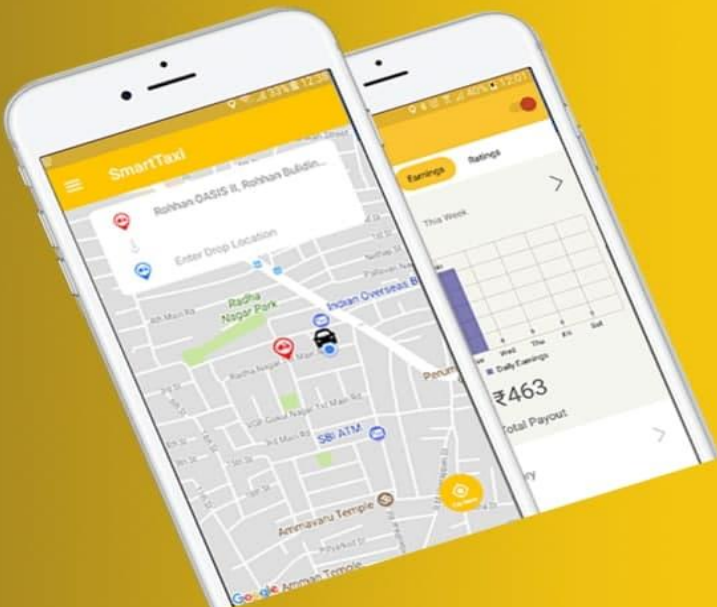
Email:

Phone

if you already have Account? [Sign In](#)

Figure 5. 8: Passenger Sign up Page

Sign In



Enjoy Emanda System

username

password

Continue

Driver passenger

Figure 5. 9: Passenger Sign in Page

CHAPTER SIX

IMPLEMENTATION AND TESTING

6.1. Implementation of the Database

In this project Firebase database management system used for the implementation of the database. The Firebase Real time Database is a cloud-hosted database. Data is stored as JSON and synchronized in real-time to every connected client. When you build cross-platform apps with our Apple platforms, Android, and JavaScript SDKs, all of your clients share one Real-time Database instance and automatically receive updates with the newest data.

6.2. Implementation of the class diagram

The implementation of the system is based on the design of the class diagram and the business logic with the targeted programming language. The attributes are properly defined with their data types and access visibility (modifiers). Methods are also defined appropriately with their return types, parameters, and data types including access visibility. The document of this project is well organized and documented based on the document design.

6.3. Configuration of the application server

In this project, we used Firebase as an application server.because it is a free, open-source framework and for the purpose of developmental speed and performance as well as for security purposes. It offers interactive shortcuts for full integration with the system databases.□

- ✓ Folders and files are properly organized.
- ✓ □ The application server properly starts and shutdowns.
- ✓ □ Its configuration enables the remote access of the system

6.4. Configuration of Application Security

From the security aspect of any system, students should take in to account the following tasks.

- ✓ ☐ Implement all input validations properly
- ✓ ☐ Implement encryption/decryption
- ✓ ☐ Roles must be defined clearly
- ✓ ☐ User accounts must be assigned with necessary access privileges
- ✓ ☐ All none functional requirements as described in the system feature part must be checked

6.5. Implementation of User Interface

The user interface of this system is implemented based on the design. It is clear and interactive; as much as possible complexity is handled in this user interface.

- ✓ ☐ It reduces memory loads of the users.
- ✓ ☐ It is consistent and interactive.
- ✓ ☐ It is easy and clear to use.

6.6. Testing the system

The functionality of the developed system must be tested because it has to provide its intended goal or

functions. The following features are tested:-

- ✓ ☐ Input and output function
- ✓ ☐ Subsystem communication
- ✓ ☐ Database and application server communication
- ✓ ☐ Databases and user interface interactions
- ✓ ☐ Graphical user interface.

Android Based Taxi Booking System

- ✓ ☐ Validations and authentication
- ✓ ☐ Security mechanisms also tested in this system.

6.6.1. Test Case

Sample Test Cases

Registration Form having following fields:-

-Email

-password

Test Case1: check user registration and approval
Test case objective : To create account
Test case description: first any user enters basic info of the user, then presses register button. Client program validate appropriate client side validation and contacts with server, server contacts with the database, and database checks for registration and if use is registered successfully the system asks for approval of registered user. If user is registered and approved then user can create his/her own account by using registered username.
Requirements Verified: Yes
Test Environment: Firebase server must be in running state, Database Should contain appropriate table and link must be established between server and client Program.

6.6.2. Testing tools and Environments

In this project different testing tools are used for testing the system development process progress and for debugging purposes.

Software testing tools

- ✓ ☐ Window 10 operating system.
- ✓ ☐ Android operating system.

Hardware testing tools

- ✓ Mobile phone
- ✓ Internet cable

6.6.3. Unit testing

We used unit testing at the coding level to test the methods' functional correctness of the independent modules, the unit testing technique was used. We used it to isolate a section of codes, verify the correctness of the code, the functions and procedures of the system, to fix debug in development cycle and helped us to understand the code base, reuse code, and enable us to make quick changes. Generally, we used a unit test to correct code errors at the right time.

6.6.4. System testing

We applied System testing at high-level design to ensure the functional and non-functional requirements' correctness, performance, accuracy, functionality validations, and security have been met correctly. We tested the complete and fully integrated software product. We examined how the whole subsystems work together to achieve the desired goal of the user's requirements from the system.

6.6.5. Integration testing

At low-level design, we used integration testing to examine how the different procedures work together to achieve the goal of the system. Check the interaction between individual functionality which performs the specific task. All the required outputs were produced

successfully as expected from the systems and inputs were validated and stored in the correct formats.

Those are:

- ✓ Onesignal for push message
- ✓ Google map
- ✓ GPS tracking
- ✓ Firebase
- ✓ Cloud messaging

6.6.6. Acceptance testing

We used acceptance testing to determine whether or not the software system has met the required specifications. We used it to evaluate the system's compliance with the business required and verify if it has met the required criteria for delivery to end-users. This system is ready for end-users because we tested it based on the following acceptance testing criteria.

- ✓ ☐ Its functionally correct and complete.
- ✓ ☐ It is usable.
- ✓ ☐ Data integrity and confidentiality are ensured.
- ✓ ☐ Saleable and available.
- ✓ ☐ Has high performance and security.
- ✓ ☐ It is well documented.

CHAPTER SEVEN

CONCLUSION AND RECOMMENDATION

7:1 Conclusion

In this project, we have developed a mobile based Taxi Booking application that facilitates various booking activities of people. The system developed in the project consists of mobile and web applications. These are two different applications on the same database. The mobile application enable user's activities like making reservation, edit booking and preview booking histories. The web application facilitates to handle managing driver information, control booking details and taxi information. Our model contains analysis model which contains the Functional and Non-functional requirements, use case, sequence, state chart, activity diagrams, conceptual modelling of classes, and identifying user interface prototypes. And also contains designs model which consists subsystem structuring, component structuring, deployment model and the persistent modelling of database.

7.2 Recommendation

According to scope of our project the team develops both web application and mobile application. Because of the time constraint we may have some limitations which should be taken in considerations, but in the future the team believes that this system can be fully operational by having some functionality that are not included in the proposed system like online payment for the service we gate.

During the development of this project the group members faced many challenges. However, by the cooperation of all group members and our advisors the team is now able to reach to the final result through this way we try to construct the road and hence this project has a big role in Ethiopia.

Finally the team would recommend that further work should done on the system in order to make the system perform better for interested organizations who would like to use online taxi reservation system in Ethiopia.

- ✓ PayPal system will be integrated

Android Based Taxi Booking System

- ✓ Driver will use voice record
- ✓ include digital map as soon as Google map is available in our country.

References

- [1] "Ride," [Online]. Available: <https://ride8294.com/service-types/>. [Accessed 2021/22].
- [2] "Feres," [Online]. Available: <https://feres.co/>. [Accessed 2021/2022].
- [3] "uml diagrams tutorial," [Online]. Available: https://www.tutorialspoint.com/uml/uml_standard_diagrams.htm. [Accessed 2021/2022].