A PROJECT REPORT

ON

"E-LOGISTIC SYSTEM"

Submitted to

UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER ENGINEERING

BY

GIGANI HEENA MOHD ASHRAF ASMA	**14DCO57**
USMANGANI M.H KADARIYA RABIYA	**12CO33**
SHAIKH IMRAN MUKHTAR AHMED NIMMI	**12CO31**
KAZI MOHID MAHMOOD NASIM	**12CO45**

UNDER THE GUIDANCE OF
PROF. JAVED KHAN SHEIKH

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2017-2018

AFFILIATED TO

UNIVERSITY OF MUMBAI

A PROJECT II REPORT
ON

"E-LOGISTIC SYSTEM"

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN COMPUTER ENGINEERING

BY

GIGANI HEENA MOHD ASHRAF ASMA	14DCO57
USMANGANI M.H KADARIYA RABIYA	12CO33
SHAIKH IMRAN MUKHTAR AHMED NIMMI	12CO31
KAZI MOHID MAHMOOD NASIM	12CO45

UNDER THE GUIDANCE OF
PROF. JAVED KHAN SHEIKH

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2017-2018
AFFILIATED TO

UNIVERSITY OF MUMBAI

Anjuman-I-Islam's Kalsekar Technical Campus

Department of Computer Engineering
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

CERTIFICATE

This is certify that the project entitled

"E-LOGISTIC SYSTEM"

submitted by

GIGANI HEENA MOHD ASHRAF ASMA	**14DCO57**
USMANGANI M.H KADARIYA RABIYA	**12CO33**
SHAIKH IMRAN MUKHTAR AHMED NIMMI	**12CO31**
KAZI MOHID MAHMOOD NASIM	**12CO45**

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at ***Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai*** under the University of MUMBAI. This work is done during year 2017-2018, under our guidance.

Date: / /

Prof. JAVED KHAN SHEIKH
Project Supervisor

Prof. KALPANA R. BODKE
Project Coordinator

Prof. TABREZ KHAN
HOD, Computer Department

DR. ABDUL RAZAK HONNUTAGI
Director

External Examiner

Acknowledgements

We would like to take the opportunity to express our sincere thanks to our guide **Prof. Javed Khan Sheikh**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his invaluable support and guidance throughout my project research work. Without his kind guidance & support this was not possible.

We am grateful to him for his timely feedback which helped us to track and schedule the process effectively. His time, ideas and encouragement that he gave helped us to complete our project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. Tabrez Khan**, Head of Department of Computer Engineering and **Prof. Javed Khan Sheikh**, Project Co-ordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

Gigani Heena Mohd Ashraf Asma(14DCO57)

Usmangani M.H. Kadariya Rabiya(12CO33)

Shaikh Imran Mukhtar Ahmed Nimmi(12CO31)

Kazi Mohid Mahmood Nasim(12CO45)

Project II Approval for Bachelor of Engineering

This project entitled ***"E-Logistic System"*** by ***Gigani Heena Mohd Ashraf Asma***, ***Usmangani M.H. Kadariya Rabiya***, ***Shaikh Imran Mukhtar Ahmed Nimmi*** and ***Kazi Mohid Mahmood Nasim*** is approved for the degree of ***Bachelor of Engineering in Department of Computer Engineering.***

Examiners

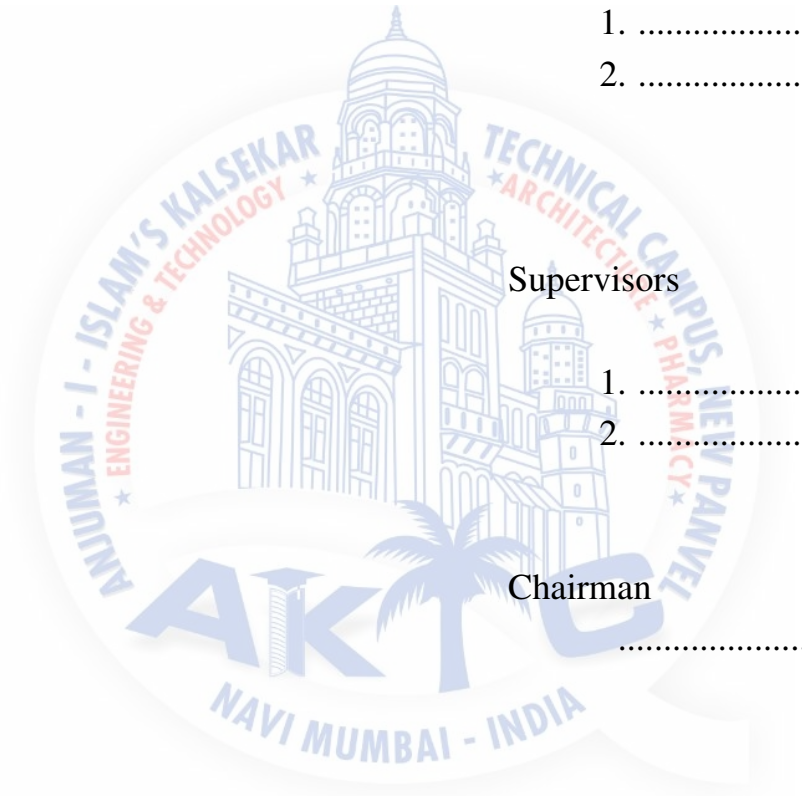
1. .....
2. .....

Supervisors

1. .....
2. .....

Chairman

.....



Declaration

We declare that this written submission represents our ideas in our own words and where others ideas or words have been included, we have adequately cited and referenced the original sources. we also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. we understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Gigani Heena Mohd Ashraf Asma
14DCO57

Usmangani M.H. Kadariya Rabiya
12CO33

Shaikh Imran Mukhtar Ahmed Nimmi
12CO31

Kazi Mohid Mahmood Nasim
12CO45

ABSTRACT

Abstract—In most of the cities, the transportation industry is highly regulated and utilizes the traditional techniques. However, people faces many problems such as searching for the vehicle as per their requirement which results wastage of time as well as it is hectic. The purpose of our project is to build E-Logistics, an online system for the customers where they can search for the appropriate vehicle and book them for their goods transportation with live updates. The system will estimate the cost via algorithm, the required amount of time between the source and destination is calculated using Djktras/ Travelling Sales Problem Algorithm, optimal route is selected and tracking of goods carrying vehicle is done through Google Map's API.

So through our system, customer can easily search for the vehicle within his/her nearby source location as per the requirement and can get live updates that assures for delivery of goods.

Keywords: Booking, Driver, E-Logistics,Intra City, Map, Pick Up, Transportation, Truck, Web Portal Google API, Live Tracking.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Due to the rapid increase in the latest internet/web technologies, people like to work or use the facilities remotely onto one click. In today's busy world, time and money are the two important factors. Taking into consideration the traditional transportation approach doesn't uses any kind of hassle free or internet/web based technologies. If a person wants to transport the goods, he/she has to search for the transportation hub/office in his nearby location which not quite possible. At many times customer dont have time to do all the work. After struggling for the search of the office by reaching destination then only user can enquire about the transportation vehicle, the estimated amount of trip, time required for the transportation and cost of it. If the user is satisfied with their terms and conditions, he has to go through the pen paper based application procedure and payment is done. Driver picks up the goods from the source location and dispatches it to the destination. Here the user can't track the goods location and is unable to get live update of his goods.

There is no proper management of transportation system like there is no drivers and user's information stored neither the records of vehicles, goods, transactions nor database is being maintained. The transaction has also to be done at the time of confirmation of trip only. There is no COD available nor any online transactions. User has to go through many phases for completion of user transportation of goods and undergo predicament. So taking all this into consideration the user suffer problems from the typical transportation system , it became mandatory that there should be an online transportation system through which the user can easily search for the vehicle in their nearby source location, gets the information about the vehicle, payment, driver and the amount needed for the trip onto one click and can book the trip and online payment or cod can also be done. User can maintain their history of trips done weekly, monthly or yearly. Transportation system can maintain system having all the records of users, drivers, trips held in a day for their business analysis. By this the user as well as the transportation system is benefitted and both can use effortlessly without facing any hassle.

1.1 Purpose

The purpose of the project is to build a aggregator platform for customer and truck drivers for inter and intracity pick up and deliveries of the goods. The basic aggregator will connect the customer with driver.The customer can track his goods on map and can check all the bookings on a single platform.

1.2 Project Scope

This will provide efficient service for customer to book transport vehicle within a minutes of time.Customer can book transport vehicle for intracity pickups and deliveries

1.3 Project Goals and Objectives

1.3.1 Goals

Project goal is to provide hectic free goods transportation service to the user and provide employment to the vehicle owners and drivers.

1.3.2 Objectives

There are mainly four objective in this project.The first objective is to show customer that how many vehicles are available around the surroundings.The second objectives to show the optimal route, time and cost by using google map api.The third objective is to provide customer to track the vehicle.The last objective to provide customer previous bookings details.

1.4 Organization of Report

In Chapter 1: we have considered Project overview under which we have explained various important terminologies like Introduction of the project.Motivation (what exactly motivate us to create E-logistics System),problem definition,About current system ,Advantages over current system,Goals and Objectives,Scope and Application.

In Chapter 2: we have discussed about various paper that we have referred for our project. We have mentioned the description,pros and cons and how the overcome the problem under every paper. a total of three paper have been referred.

In Chapter 3: we have discussed about the requirement analysis under it we have

consider about the requirement the platform requirement supporting the os of the software and hardware requirement along with the feasible study.

In Chapter 4: we can see the system design and architecture various diagram can be seen in this chapter which represent the software , diagram including our system architecture usecase diagram dfd diagram class diagram and component diagram.

In Chapter 5: we have seen the methodology here we have explain the project in detail by dividing into module.various module of priority based cab search are explained with the help of few diagram.

In Chapter 6: we have discussed about the implementation details the assumption and dependencies this part contains details of the implementation of methodology that we discuss earlier.

In Chapter 7: we have shown the test cases and result along with analytic discussion this part contained the result of the output of the project.

In Chapter 8: we have concluded the whole project and future scope along with the limitation followed up by reference and chapter 9 with Appendix.

Chapter 2

Literature Survey

2.1 Disruptive Change In Taxi Business The Case Of Uber

In most cities, the taxi industry is highly regulated and utilizes technology developed in the 1940s. Ride sharing service "UBER", which uses modern Internet-based mobile technology to connect Passengers and Drivers, have begun to compete with traditional taxis. This paper examines the efficiency of ride sharing service taxis by comparing the capacity utilization rate of "UberX" drivers with that of traditional taxi drivers in five cities. The capacity utilization rate is measured by the fraction of time a driver has a fare-paying passenger in the car while he or she is working and by the share of total miles that drivers log in which a passenger is in their car. The main conclusion is that, in most cities with data available. "UberX"drivers spend a significantly higher fraction of their time and drive a substantially higher share of miles, with a passenger in their car than do taxi drivers, Four factors likely contribute to the higher capacity utilization rate of "UberX" drivers.

2.1.1 Advantages of Paper

a. It provides for an efficient way for booking a ride.

b. It also provides Live tracking.

c. The price of rides are competitive.

2.1.2 Disadvantages of Paper

a. The Disadvantage of this paper is that it does not support Transportation of Goods/Packages.

b. It operates only in few cities.

2.1.3 How to overcome the problems mentioned in Paper

a. As we an see now a days most of the people are familiar with Ride service/ Ride sharing service, But none of the ride service support a way to transport packages.

b. In our paper we are proposing efficient as well as improved way of transporting Packages.

c. In our scheme, any user can easily login into our system and send the Product to their desired Destination.

2.2 Recent Trends in Intelligent Transportation System: A Review

Managing the increasing traffic is a big problem all over the world, Intelligent Transportation System(ITS) provides solution to these problems with the help of new technologies. ITS is an integrated system that implements a broad range of communication, control, vehicle sensing and electronic technologies to solve and manage the traffic problems. ITS is being used in the developed countries since past two decades, but it is still a new concept when developing countries like India, Brazil, China, South Africa etc. are concerned. N the present study we have studied four major parts of the ITS i.e. Advanced Traveler Information System(ATIS). Objective of this paper is to study various ITS architecture and model and review such model to get in-depth of their architecture. Hence architecture and developed models over the years of 4 major branches of ITS have been reviewed here to make a comparison analysis of different models that have been developed by the researches in their studies. It will lead to the gaps in he knowledge which can be further studied. The paper highlights the conclusion extracted from the studies of different systems and also give the future scope in the field of ITS to make it more user friendly and accessible.

2.2.1 Advantages of Paper

a. It provide the users Optimal trip option with least travel time between the traveler's origin and destination, including walking, waiting , transfer and in-vehicle time.

2.2.2 Disadvantages of Paper

a. To achieve the purpose of providing the optimum route the methodology which was adopted is to consider only those bus stop points which are active (have service) at the time of travel as all the bus stop points don't have the service all 24*7 and considering only active bus stop points results in optimum route.

b. This is one of the major drawback of this system, without active Bus stop point , it wont be able to choose Optimum route

2.2.3 How to overcome the problems mentioned in Paper

a. Our system does not depend on active Bus stop point to find the optimum route, instead we use google made API to find the Optimum route.

2.3 Porter-We Hire Mini-Trucks

Porter is used to hire mini Trucks.Porter is same as OLA.Customers can book mini trucks/ tempo through website and Android application. Customers can also rate their Experience. Customers can track their goods.

2.3.1 Advantages of Paper

a. It provides web-based service as well as Android app and iOS app.

b. A person can add his/her truck in the system.

2.3.2 Disadvantages of Paper

a. It only provides call based Booking.

b. It does not provide Live tracking.

c. It operates only in few cities(i.e. Bangalore, Mumbai, Hyderabad, Chennai and Delhi NCR).

2.3.3 How to overcome the problems mentioned in Paper

a. Ease of booking through instant message.

b. Live Tracking of vehicle can be done

2.4 Technical Review

CodeIgniter:

It is an application development framework, which can be used to develop websites, using PHP. It is an Open Source framework. It has a very rich set of functionality, which will increase the speed of website development work. CodeIgniter is based on the Model-View-Controller (MVC) development pattern. MVC is a software approach that separates application logic from presentation. The Model represents data structures. Model classes contain functions that retrieve, insert and update information into the database. The View is information that is being presented to a user. A View is normally a web page and page can also be fragmented like a header or footer. Here in this project too, the page is being fragmented into header and footer and all this are autoloaded for new page where ever we want them. In view, all the GUI design is done.The Controller serves as an intermediary between the Model, the View, and any other resources needed to process the HTTP request and generate a web page. It just takes the request from view, fetches information from the model and gives information from database i.e. here is model to the view.

Advantages of Technology

a. Doesn't need to load page as it gets autoloaded by use of MVC.

b. Form and Data Validation ensures that the data that we are getting is proper and valid to process.

c. Redirect function to redirect from one to other.

Reasons to use this Technology

Model-View-Controller Based System

Session Management

Data Encryption

PHPMYADMIN:

It is a free and open source tool written in PHP intended to handle the administration of MySQL. with the use of web browser. It is graphical user interface to manage SQL Databases. It performs various tasks such as creating, modifying or deleting tables, fields, rows; executing SQL statements; or managing users and permissions.

Advantages of Technology

a. The advantages of using phpMyAdmin is that it has a user interface and you can run queries within the SQL.

b. Another advantage is that you can paste queries into the SQL to test data output; fom a simple 'Select * FROM tablename' to more advanced relational queries using various tables.

c. The copying and pasting of queries is not available with the mySQL console. If you make a typo with the mySQL console, you must start again.

Reasons to use this Technology

a. Importing A Database From Backup

b. Database Optimization Via phpMyAdmin

c. Checking and repairing tables

Chapter 3

Project Planning

3.1 Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Gigani Heena	UI Design, Database and Documentation.
2	Usmangani Kadariya	UI Design, Development and Database
3	Imran Shaikh	UI Design, Database, Documentation and Logo design
4	Mohid Kazi	UI Design

Work Breakdown Structure

3.2 Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Gigani Heena	Member	Database, Login and Documentation.
2	Usmangani Kadariya	Team Leader	Development, UI and Database
3	Imran Shaikh	Member	UI Design,Documentation and Logo design
4	Mohid Kazi	Member	UI Design and Documentation

3.3 Assumptions and Constraints

Assumption

The driver should login in the system and must not have any current booking to be available to the customer.
Driver GPS should be ON.
Google API's services should be available.

Constraints:-

If google api servers goes down we cannot track drivers as well as we cannot give optimal route cost and time.

3.4 Project Management Approach

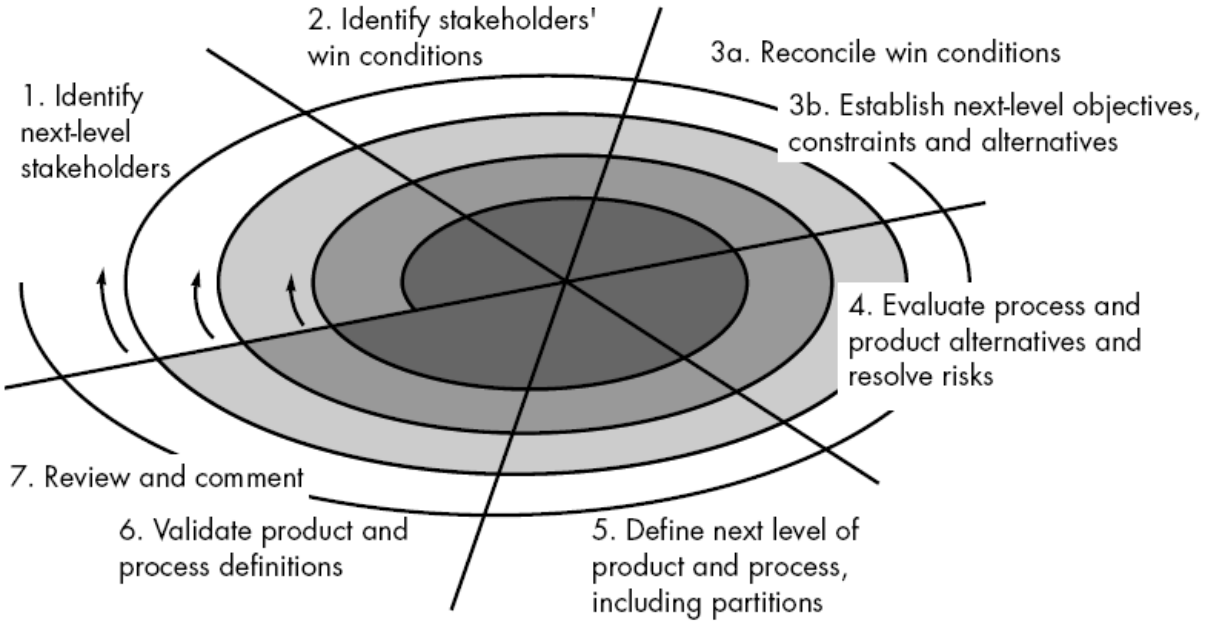


Figure 3.1: Spiral Model

In our project we have used spiral model for implementing all the phases successfully.This model involves strategies, which is a combination of incremental and prototype models. This model is suitable for planning and implementing to achieve the goal of the project.It maintains a systematic step wise approach.

3.5 Ground Rules for the Project

1. We treat each other with respect.

2. We intend to develop personal relationships to enhance trust and open communication.

3. We value constructive feedback. We will avoid being defensive and give feedback in a constructive manner.

4. As team members, we will pitch in to help where necessary to help solve problems and catch-up on behind schedule work.

5. Additional meetings can be scheduled to discuss critical issues or tabled items upon discussion and agreement with the team leader.

6. One person talks at a time; there are no side discussions.

7. When we pose an issue or a problem, we will also try to present a solution.

3.6 Project Budget

The budget for this project is very low as most of the tools we have use are open source.Following open source softwares are used.
1. Operating System:linux mint (Open Source).
2. API:Google API(Open Source).

3.7 Project Timeline

	Name	Duration	Start
1	PLANNING	4 days?	10/10/17 8:00 AM
2	RESEARCH	4 days?	14/10/17 8:00 AM
3	LITERATURE SURVEYS	6 days	20/10/17 8:00 AM
4	SYNOPSIS REPORT	5 days	25/10/17 8:00 AM
5	UI DESIGN	20 days	23/11/17 8:00 AM
6	WORKING ON LOGIN MODULE	10 days	25/12/17 8:00 AM
7	WORKING ON GOOGLE API	10 days	1/1/18 8:00 AM
8	WORKING ON SOURCE AND DESTINATION	10 days	10/1/18 8:00 AM
9	WORKING ON GOOGLE MAP	15 days	25/1/18 8:00 AM
10	PAPER PUBLICATION	23 days	30/1/18 8:00 AM
11	PAPER CUM POSTER PRESENTATION	4 days	5/3/18 8:00 AM
12	DESIGNING OF REVENUE MODEL	3 days	10/3/18 8:00 AM
13	IMPLEMENTATION OF COST CALCULATION	4 days	15/3/18 8:00 AM
14	WORKING ON ADMIN MODULE	4 days	19/3/18 8:00 AM
15	WORKING ON VEHICLES & DRIVERS MODULE	5 days	22/3/18 8:00 AM
16	WORKING ON TRACKING MODULE	10 days	26/3/18 8:00 AM
17	TESTING	3 days	13/4/18 8:00 AM

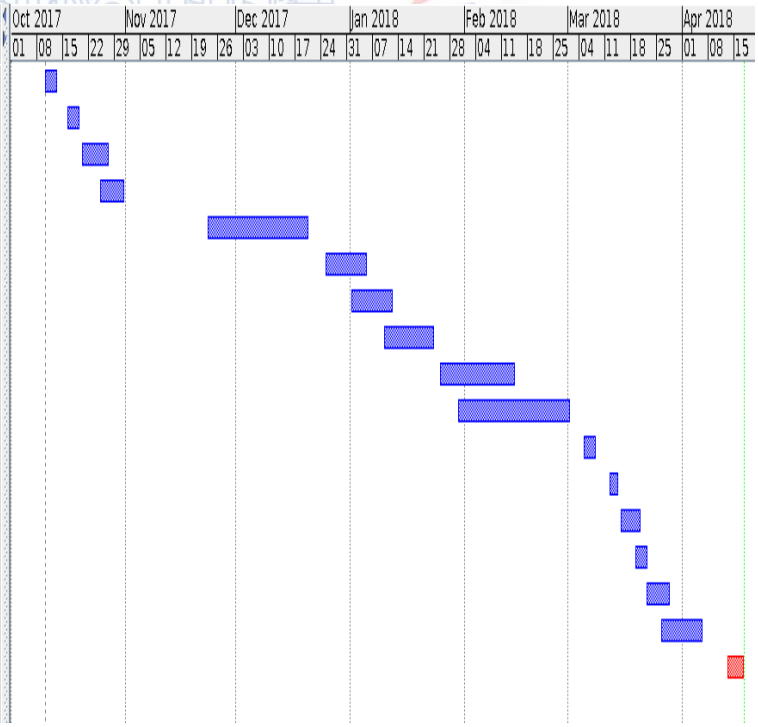


Figure 3.2: Timeline

Chapter 4

Software Requirements Specification

4.1 Overall Description

4.1.1 Product Perspective

The perspective of the website is to avoid the stress and work for the regular users.This web application is standalone system, the webapp is a aggregator between truck drivers and customer.It aims to ease the customer to book the vehicle by providing good UI with google map which provide optimal route time and cost.

4.1.2 Product Features

The major features of the product is that user can book a vehicle in very short time.The web application will provide all features to book vehicle and follow up from pickup to deliveries of goods.The Tracking feature will provide transparencies to the user.The user can also check their previous booking transaction from the given option

4.1.3 User Classes and Characteristics

Different user will use the product differently and the user class related to the app will change according to the need of user. But the pertinent characteristics of the classes will remain the same and user will primary interact will three main class of product that is authentication, maps and booking class, the rest are less important according to this three class. User interaction with this classes will also enhance the user experience with the product.

4.1.4 Operating Environment

The website is environment friendly,the website run on all the updated browser and the website is responsive therefore it can run in mobile on all browser.

4.1.5 Design and Implementation Constraints

The development of the system is based on the google maps api, if the google maps server will get slow down or does not connect properly then we cannot provide live tracking as well as user cannot book the vehicle because it will not calculate cost without route and time which will provided by google map api.

4.2 System Features

The major features of our system is to provide customer optimal route, time and cost before booking.After booking vehicle customer can track their goods till it reach to its destination.

4.2.1 System Feature

1.Optimal route.
2.Optimal time.
3.Optimal cost.
4.Easy booking.
5.Easy tracking.

4.2.2 Description and Priority

1.Optimal route.
This feature will provide optimal route through google map api.
Priority:-High
2.Optimal Time.
This feature will provide optimal time through google map api.
Priority:-Medium
3.Optimal cost.
This feature will provide optimal cost through google map api and algorithm which is build to calculate cost.
Priority:-High
4.Easy booking.
In this feature customer can easily book vehicle.
Priority:-High
5.Easy tracking.
In this feature customer can easily track vehicle.
Priority:-High

4.2.3 Stimulus/Response Sequences

1.The user need to login in to the system.
2.The user will enter the source and destination.
3.The user will select the vehicle and book the vehicle
4.The user can track the vehicle.

4.2.4 Functional Requirements

1.The user should login in to the system.
2.The user should enter correct source and destination location.
3.The range should lies within India
4.The servers should response quickly.

4.3 External Interface Requirements

4.3.1 User Interfaces

1. User shall be able to login in the system.
2. After the login session shall be maintain.
3. The user can enter the source and destination.
4. The user can view the optimal route and time on map.
5. The user can book the transport vehicle.
6. The user can track the vehicle.

4.3.2 Hardware Interfaces

Any device which support internet compatibility with updated browsers.
On driver side driver device which support internet compatibility with updated browsers with gps enbale device

4.3.3 Software Interfaces

Database:MySql Tools:sublime text editor Apis:google maps API.

4.3.4 Communications Interfaces

1.The communication for location is done by google apis.
2.The interface between the MySql and the system will be done byn using http protocol

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

The performance of the system is based on the google map api servers, the route, time and cost will be calculated through map api, if server will respond slow or will not connect then the system will not work

4.4.2 Safety Requirements

The database should be maintained and the should maintain the copy, all the user data will be save in the database, if any damage will occur in database whole data will be destroy therefore copy should maintain for safety.

4.4.3 Security Requirements

The major security requirements for the system will be the safeguarding of the user data from any kind of exploit. Inorder to protect the user data the data is not stored in local databases we will be storing in the cloud for better security.

Chapter 5

System Design

5.1 System Requirements Definition

Our system is an online transportation web based portal by which user can easily searches for the vehicle as per their need, books them, can do payment online and can access it remotely which can be effortless and hassle free for the user and for the transportation management too by making a single complete online system in which they can keep the records and can analyze their business as well as for the assigning of drivers and tracking of the goods.

5.1.1 Functional requirements

1.The customers must register for creating the account and login using username and password to book the cab.

2. Search for the vehicle as per their need.

3. Get estimation of cost. time and distance through algorithm and Google API.

4. Book the vehicle.

Use-case Diagram

This use case diagram give us the ability to build our entire authorization system. Here we have 3 Actors viz. USER, SYSTEM, DRIVER. USER has the ability to interact with every one of the listed component, DRIVER has access to a few of those use cases. Here, USER can rate the DRIVER, even DRIVER have the ability to rate the transport service, which means our SYSTEM let's our DRIVER not only drive but also rate the experience with USER whether it was good or bad.

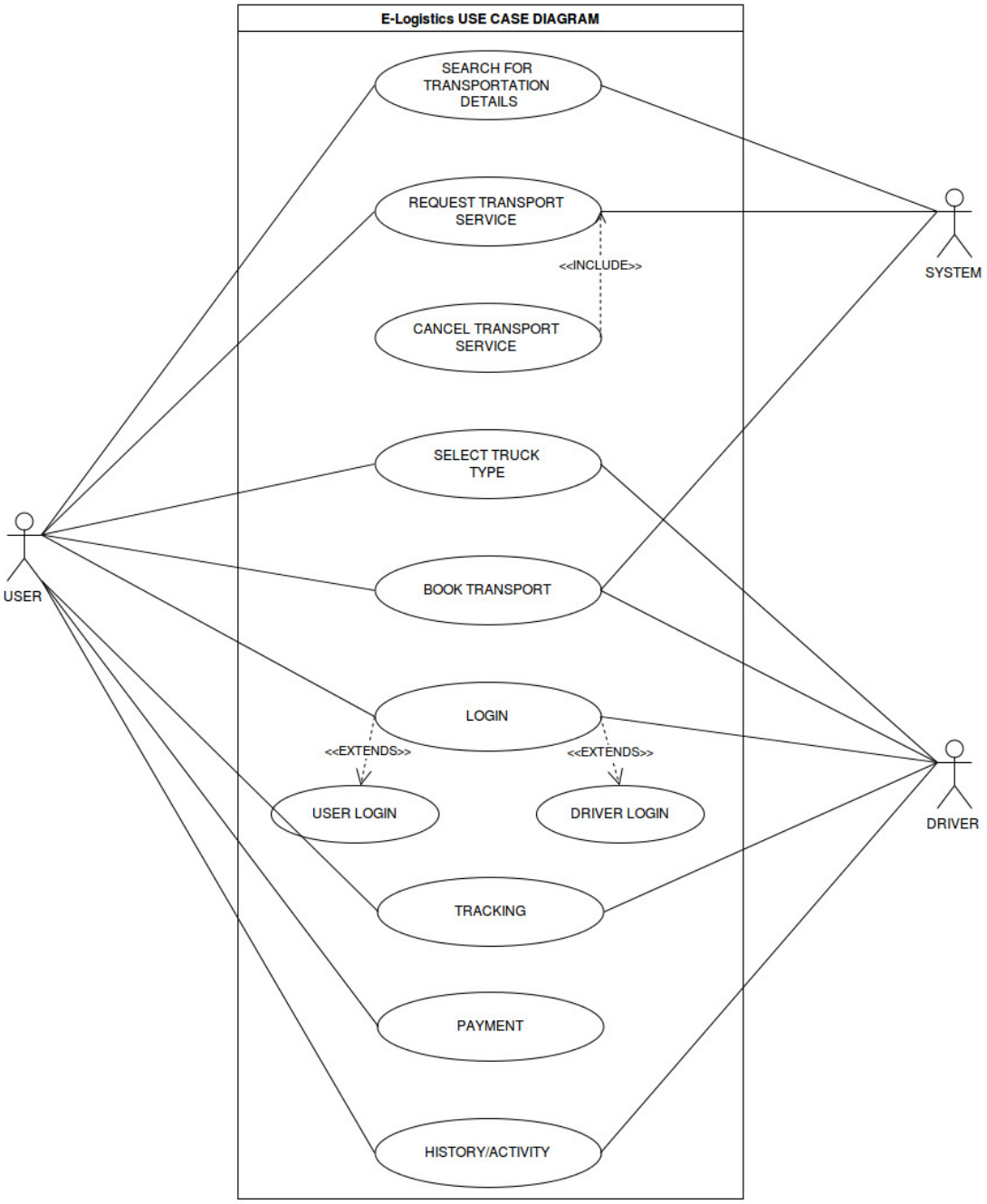


Figure 5.1: Use-case Diagram

Data-Flow Diagram

A data flow diagram (DFD) is a graphical representation of the ”flow” of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated.[2] DFDs can also be used for the visualization of data processing (structured design).[citation needed]. A DFD shows what kind of information will be input to and output from the system, how the data will advance

through the system, and where the data will be stored. It does not show information about process timing or whether processes will operate in sequence or in parallel, unlike a traditional structured flowchart which focuses on control flow, or a UML activity workflow diagram, which presents both control and data flows as a unified model.

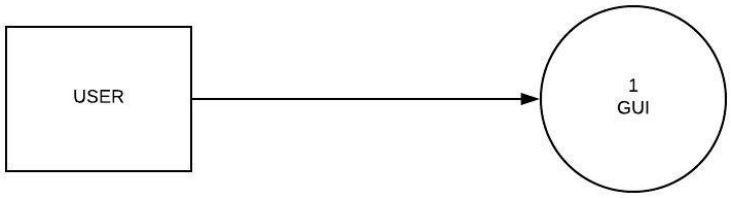


Figure 5.2: DFD Level 0

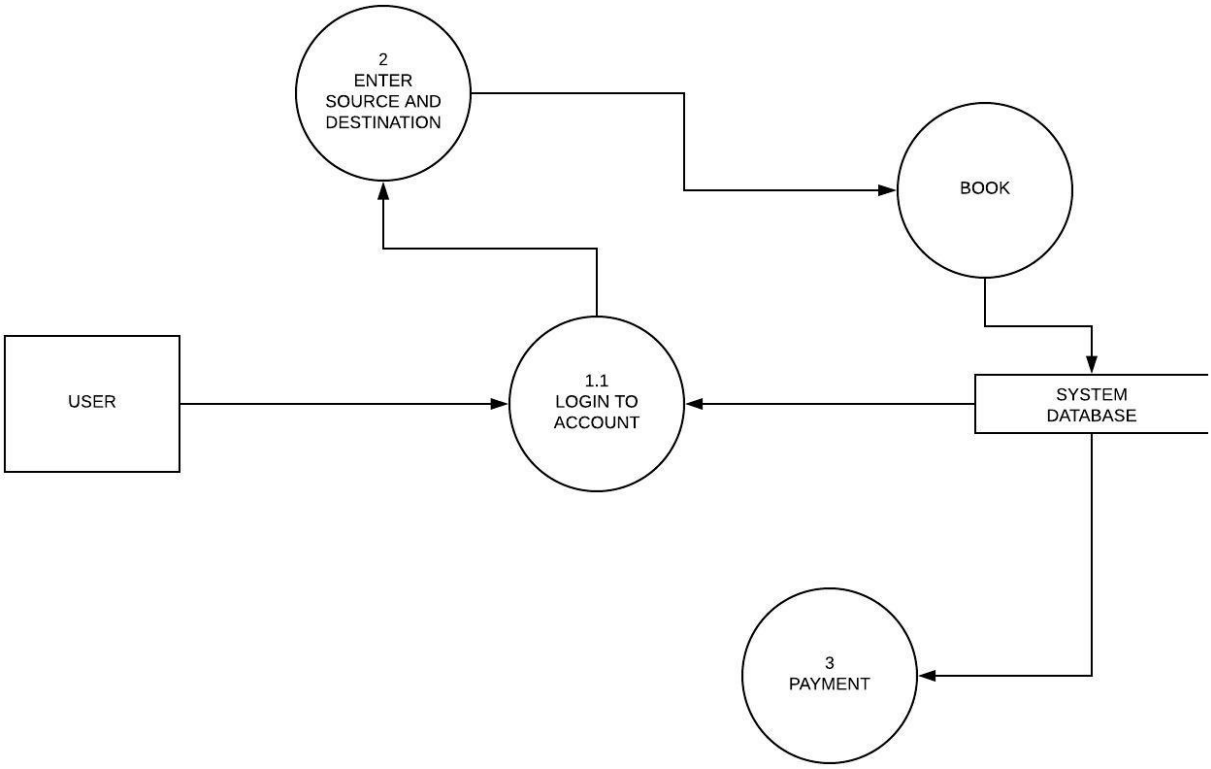


Figure 5.3: DFD Level 1

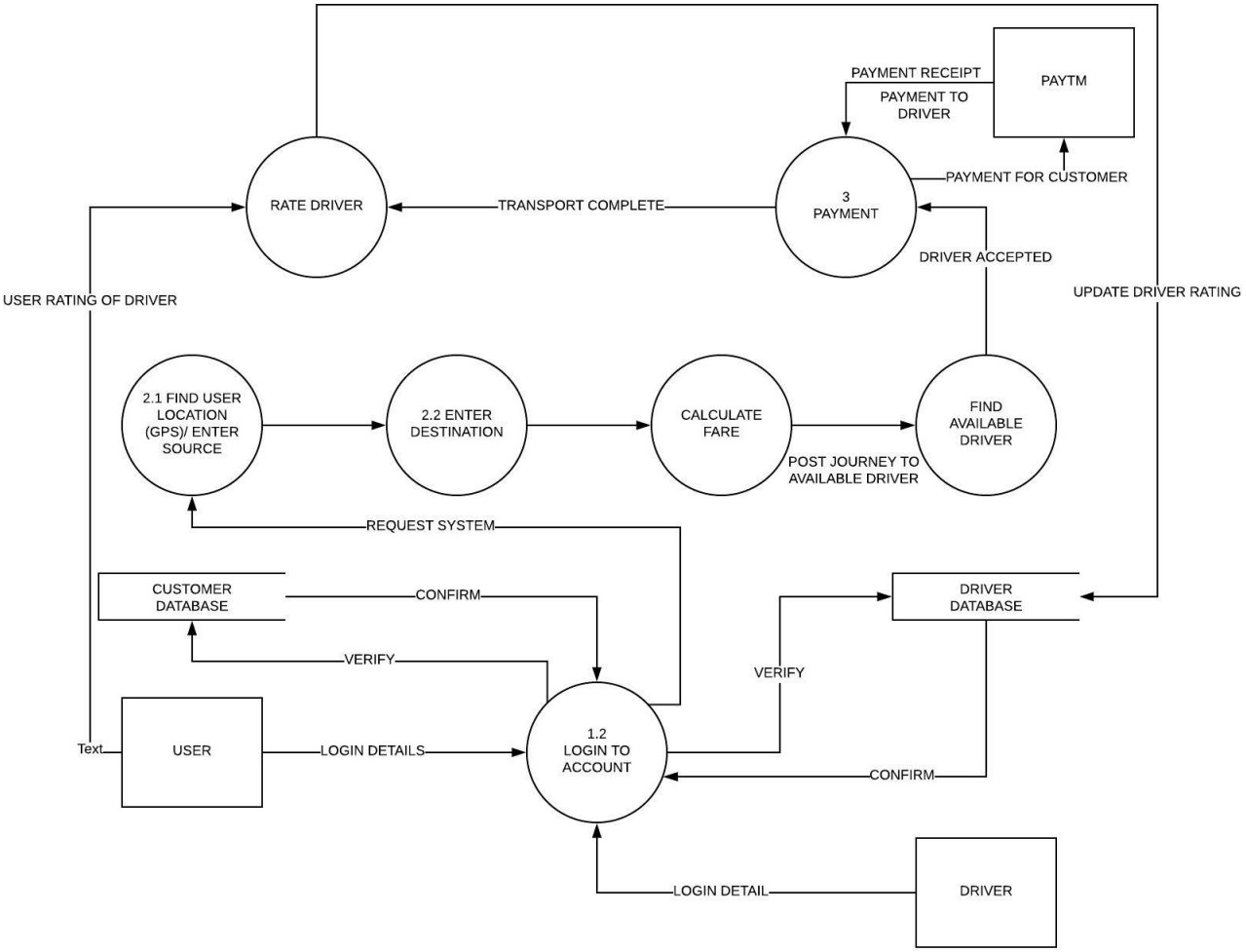


Figure 5.4: DFD Level 2

Entity-Relationship Diagram

n entity–relationship model (ER model for short) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types. An entity–relationship diagram for an MMORPG using Chen's notation.In software engineering, an ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model, that defines a data or information structure which can be implemented in a database, typically a relational database.Entity–relationship modeling was developed for database design by Peter Chen and published in a 1976 paper.[1] However, variants of the idea existed previously.[2] Some ER models show super and subtype entities connected by generalization-specialization relationships,[3] and an ER model can be used also in the specification of domain-specific ontologies.

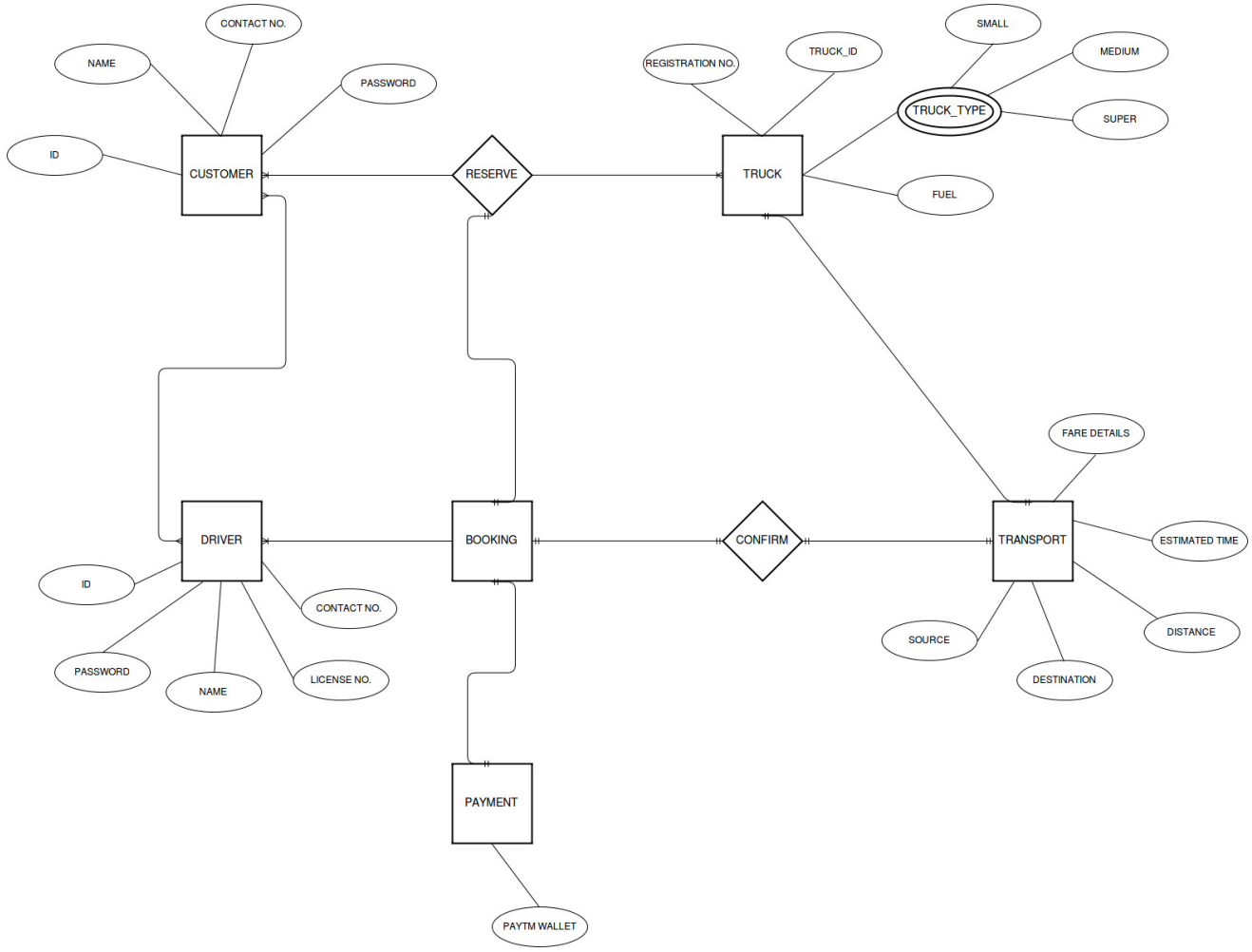


Figure 5.5: Entity-Relationship Diagram

Activity Diagram

Starting right here at the very top left and side this is going to be when the truck has been ordered and we're going to start immediately right here and start with a branch this is a branch that simply have multiple inputs for SET SOURCE. Snow from there it sets the DESTINATION.Now with these two items, the system can check to see if the transport is possible. If this trip is not possible then it will show you error. Now if the transport is possible,we go down and get transport details and finally user can book transport service.Once booked user will have to pay for transport service and once his payment is approved , user's transport service is started or user can cancel the payment and if the payment is cancelled, the booking is also cancelled.

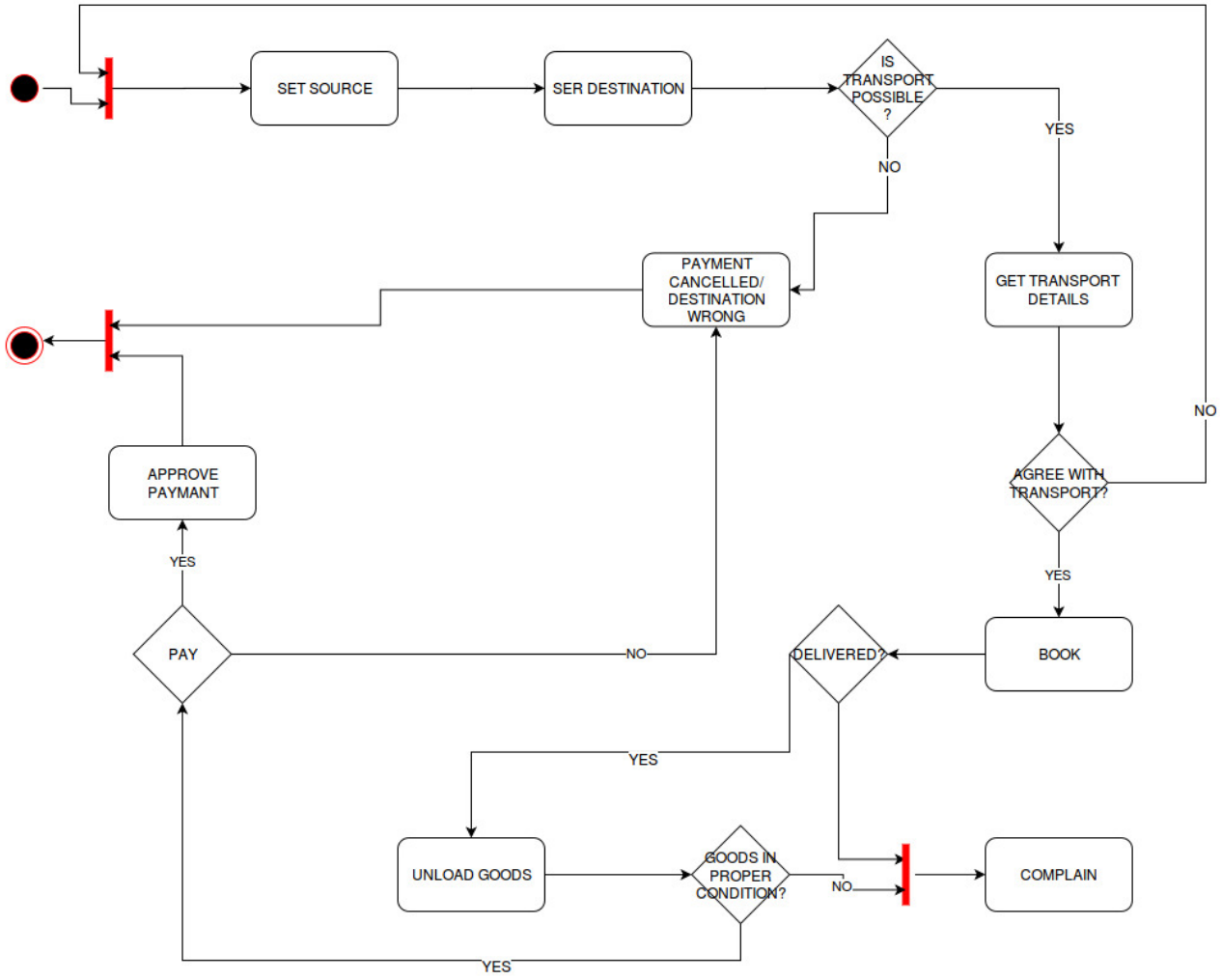


Figure 5.6: Activity Diagram

Object Diagram

In the Unified Modeling Language (UML), an object diagram focuses on some particular set of objects and attributes, and the links between these instances. A correlated set of object diagrams provides insight into how an arbitrary view of a system is expected to evolve over time. In early UML specifications the object diagram is described as ”An object diagram is a graph of instances, including objects and data values. A static object diagram is an instance of a class diagram; it shows a snapshot of the detailed state of a system at a point in time. The use of object diagrams is fairly limited, namely to show examples of data structure.The latest UML 2.5 specification does not explicitly define object diagrams, but provides a notation for instances of classifiers.Object diagrams and class diagrams are closely related and use almost identical notation.Both diagrams are meant to visualize static structure of a system. While class diagrams show classes, object diagrams display instances of classes (objects). Object diagrams are more concrete than class diagrams. They are

often used to provide examples or act as test cases for class diagrams. Only aspects of current interest in a model are typically shown on an object diagram.

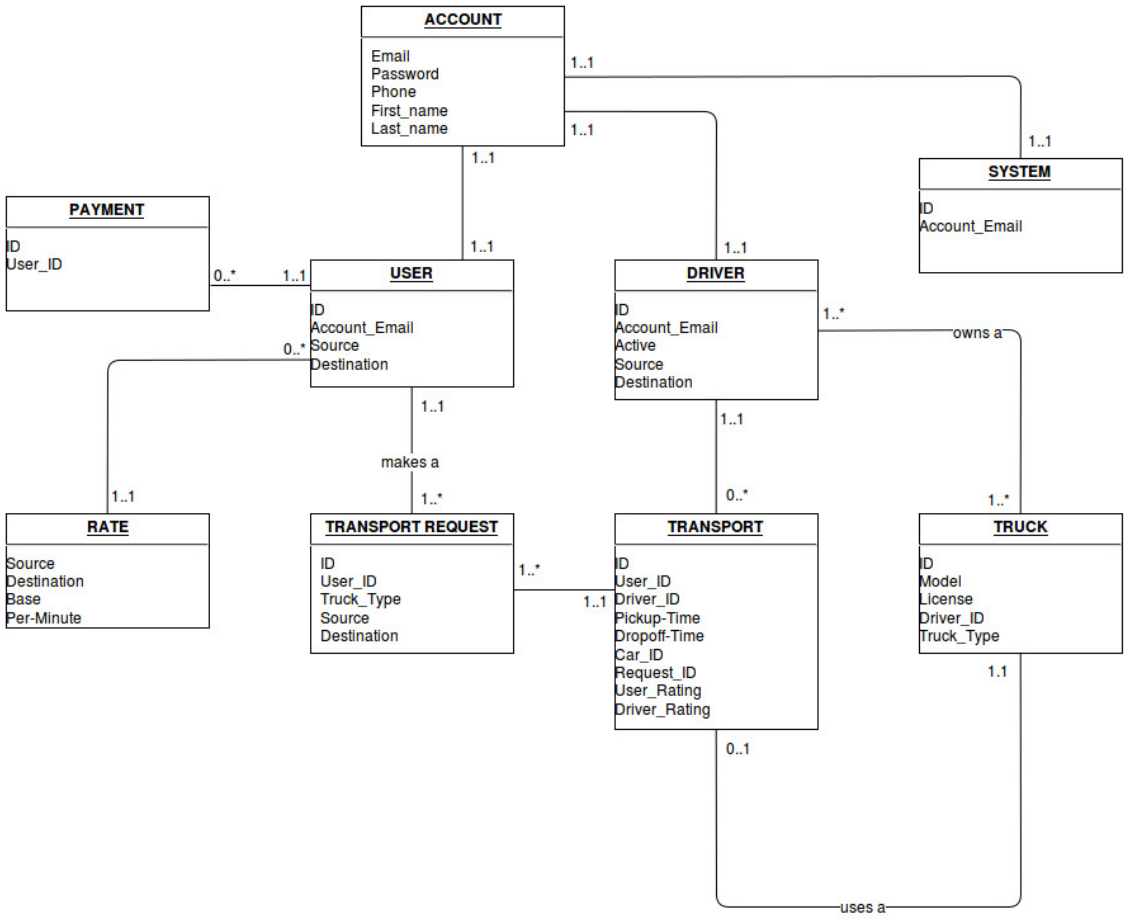


Figure 5.7: Object Diagram

Collaboration Diagram

A collaboration diagram is a type of visual presentation that shows how various software objects interact with each other within an overall IT architecture and how users can benefit from this collaboration. A collaboration diagram often comes in the form of a visual chart that resembles a flow chart. It can show, at a glance, how a single piece of software complements other parts of a greater system.In many cases, a collaboration diagram will show how a system made up of individual software pieces works in real time. In other cases, the flow chart objects may represent a more abstract interaction, such as a general cause-and-effect or event-driven collaboration that may happen over time.

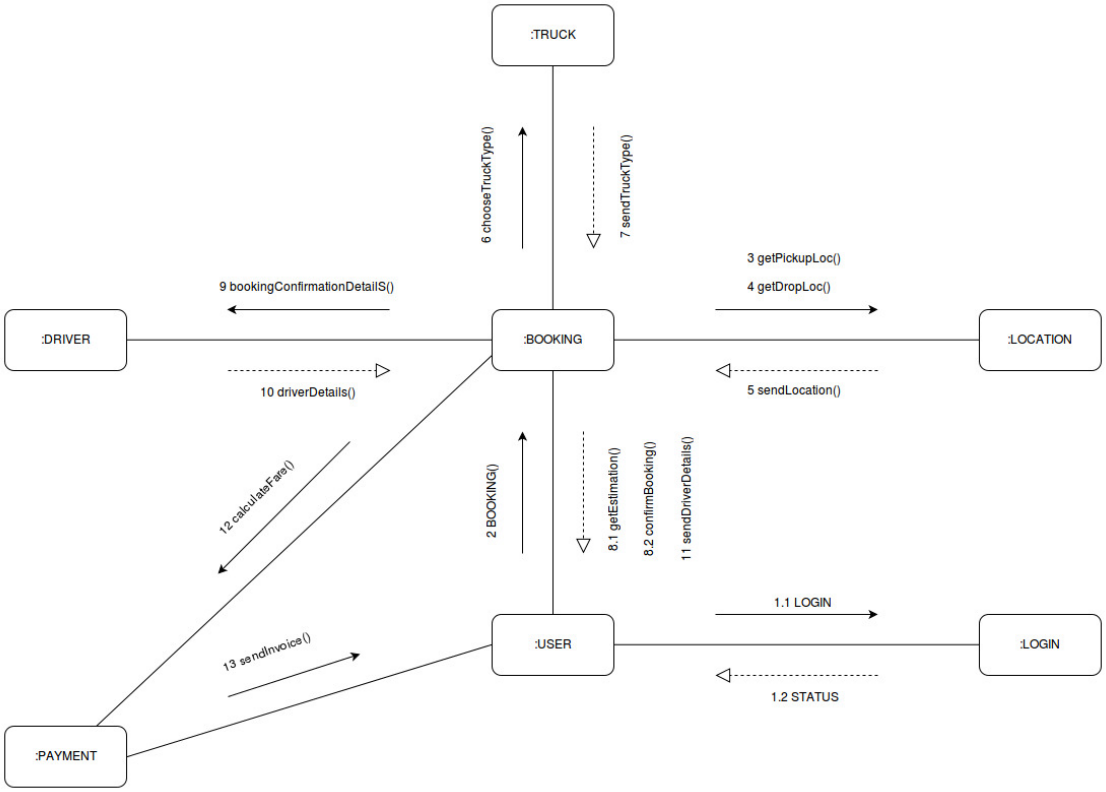


Figure 5.8: Collaboration Diagram

5.1.2 System requirements (non-functional requirements)

5.1.3 Availability

The system should be available all the time when user is in need of it. Any driver must be there logged into the system.

5.1.4 Performance Requirement

For security purpose, user has to gets register with his mail id, phone no and address. Requires less amount of space or memory.

Database Schema/ E-R Diagram

5.2 System Architecture Design

Firstly, the user needs to sign up into the system for the purpose of booking of the trip or vehicle. By entering the source and destination location, user will get the estimated cost, total amount of time and distance of the route. After booking of the vehicle, confirmation of the trip is done instantly and details of the trip along with the user id and consignment no and driver's details is notified to the user's contact no. Particular driver is being assigned to the trip, that driver picks up the goods from

source location and dispatches it to destination address within specified amount of time. User can view the live tracking of their goods carrying vehicle through google API map. After the delivery of goods, payment is done by COD (cash on delivery) or at the time of confirmation it is being done by net banking or through debit/credit card banking.

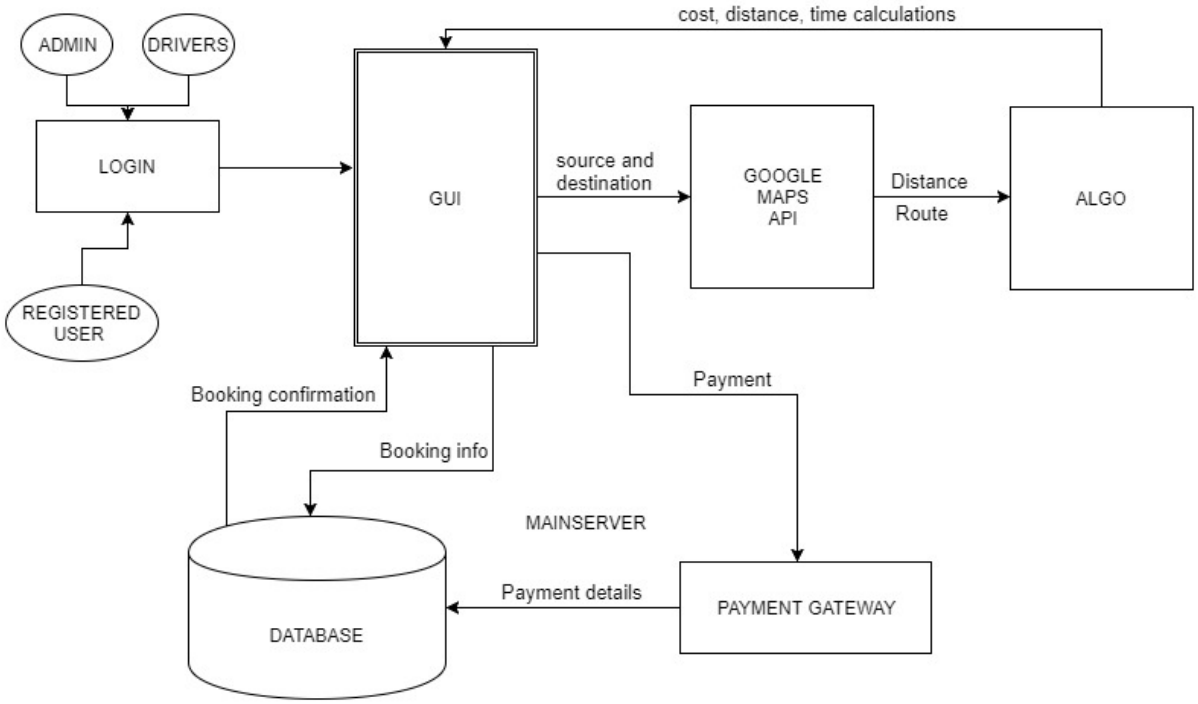


Figure 5.9: System Architecture

5.3 Sub-system Development

5.3.1 Login

Login: Here the user logs in into the system for booking of the vehicle, driver logs in to gets connected with the system and admin ,logs in to see the status of the system.

5.3.2 Booking

Booking of the vehicle is done after login into the system.After booking a booking id is being assigned to all the user, invoice is being generated and a particular driver is assigned to the user.

5.3.3 LIVE TRACK OF GOODS

User can view the live update of the vehicle lively through Google Map API. By using Google Map's API, the estimated amount of time, cost and distance is also

calculated.

5.3.4 Driver

Driver picks up the good from the source location and dispatches them to the destination location of the user.

5.4 Systems Integration

First module of the system is login module where the user sign up and logs into the system, searches for the vehicle and gets estimated amount of time and distance through google API and cost through algorithm. Second module is booking, where user books the vehicle. third module is tracking of vehicle carrying goods. Forth module is driver where a driver is assigned to a user who takes the goods from source and dispatches them to the destination and lastly admin module who looks up for the whole system functioning; add drivers, add trucks view the whole system.

Class Diagram

In a class diagram, the classes are arranged in groups that share common characteristics. A class diagram resembles a flowchart in which classes are portrayed as boxes, each box having three rectangles inside. The top rectangle contains the name of the class; the middle rectangle contains the attributes of the class; the lower rectangle contains the methods, also called operations, of the class. Lines, which may have arrows at one or both ends, connect the boxes. These lines define the relationships, also called associations, between the classes.

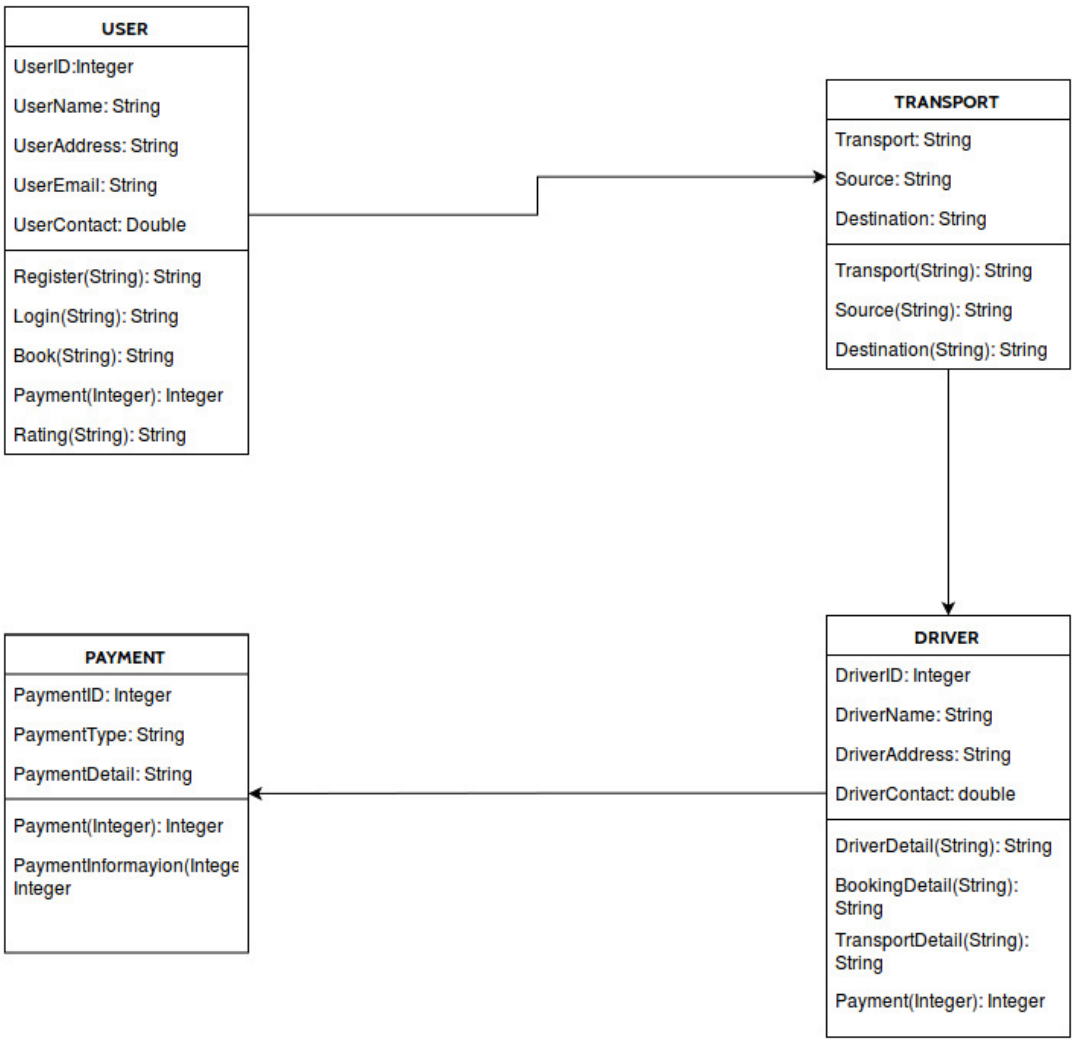


Figure 5.10: Class Diagram

5.4.1 Sequence Diagram

A sequence diagram shows object interactions arranged in time sequence.It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development.Sequence diagrams are sometimes called event diagrams or event scenarios.A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them,in the order in which they occur.This allows specification of simple runtime scenarios in a graphical manner.

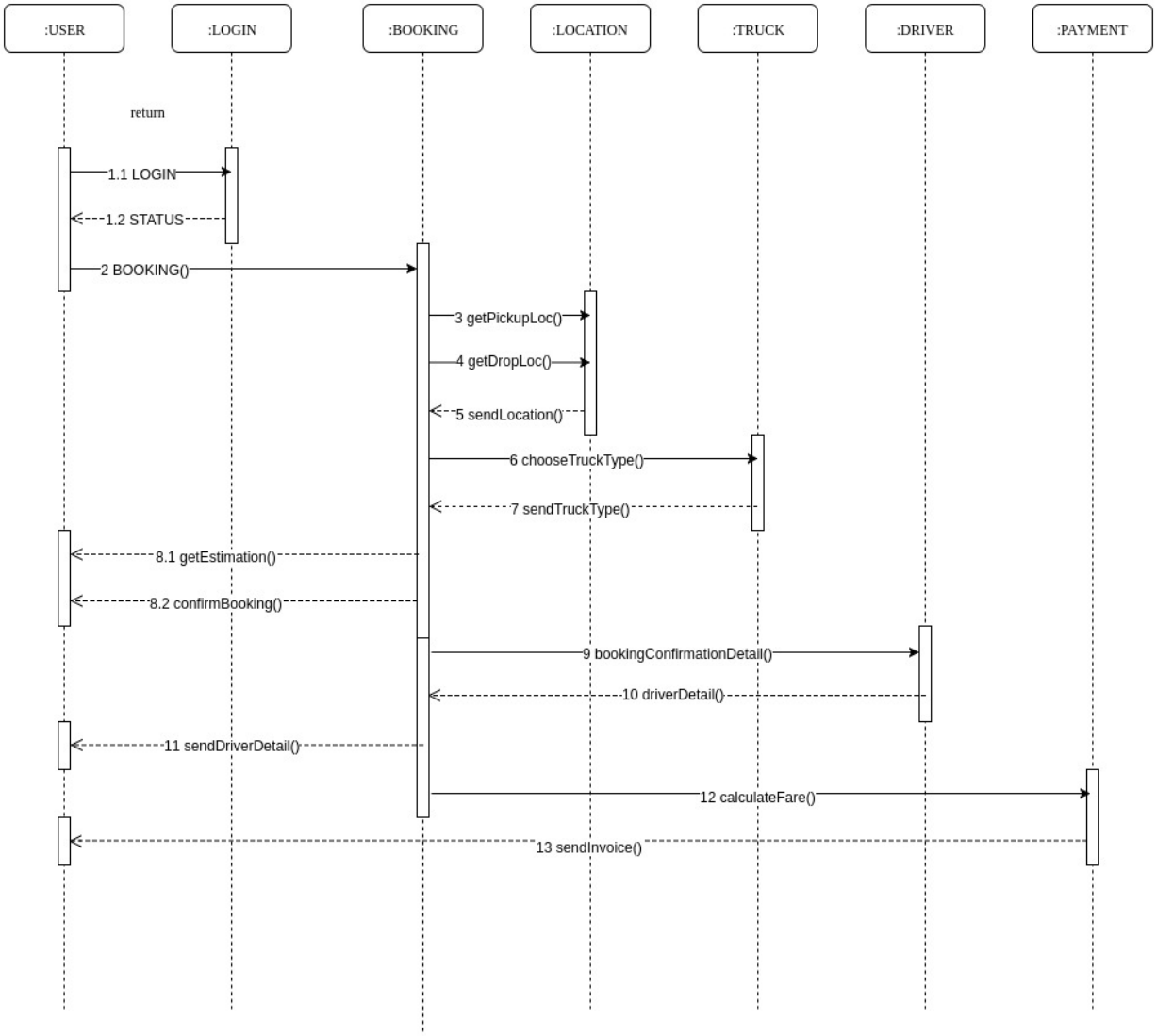


Figure 5.11: Sequence Diagram

Chapter 6

Implementation

There are various modules for the E-Logistic system viz; login module, booking, location tracker and the driver module.

6.1 Login

Login modules contains three sub modules of login that is user's login, driver's login and admin's login.

User Login- User registers or sign-up into the system if they are new to the system and the user signs in by entering their credentials. Without signing in, no user can book their trip. So logging into the system is mandatory for the purpose of booking and delivery of goods.

Driver Login- Driver signs into the system by using his driver-id to get connected with the system. If the driver is on duty, he has to get logged into the system. Drivers who are on duty are assigned to the users at the time of booking.

Admin Login- Admin is the main module of the project. Admin assigns the driver to particular user, checks various activities and performs daily schedule of the system. He/she analyses the system, can also modifies the system. All the responsibilities is on the shoulder of the admin.

```

ADMIN LOGIN:

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Admin_c extends CI_Controller
{
  public function __construct(){
        parent::__construct();
```

```
        $this->load->model('Login_m');
        $this->load->model('Details_m');
        $this->load->model('Booking_m');
    }

    public function index(){
      $this->load->view('admin/header');
        $this->load->view('admin/a_login_v');
        $this->load->view('admin/footer');
    }

    public function a_login(){

        $validation = array(
            array('field' => 'a_email', 'rules' => 'required'),
            array('field' => 'a_password', 'rules' => 'required')
        );
        $this->form_validation->set_rules($validation);

        $a_email=$this->input->post('a_email', TRUE);
        $a_password=$this->input->post('a_password', TRUE);
        $en_pass=hash('sha1',$a_password);

        $auth=$this->Login_m->a_login($a_email,$en_pass);

        if($this->form_validation->run() === FALSE){
        //$this->session->set_flashdata('error', validation_errors());
            redirect(base_url('admin_c'));
        }

        if($auth){
             //$r['result']= $res;
             //$data['result']=$res;
            session_start();
            $r = $auth;
            echo $r;
            $_SESSION['aname'] = $r;
            //$_SESSION['email']=$email;
            redirect(base_url('Admin_c/content'));

            //print_r($r);
        }
        else{
            redirect(base_url('Admin_c'));;
        }
    }

    public function content(){
        $data['h']=$this->Booking_m->estimate();
        $this->load->view('admin/header');
        $this->load->view('admin/content',$data);
        $this->load->view('admin/footer');
    }

    public function users(){
        $this->load->view('admin/header');
        $this->load->view('admin/users');
        $this->load->view('admin/footer');
    }

    // Add new Truck
```

```
72      public function add_truck(){
73          $validation1 = array(
74              array('field' => 't_name', 'rules' => 'required'),
75              array('field' => 'd_name', 'rules' => 'required'),
76              array('field' => 'truck_no', 'rules' => 'required'),
77              array('field' => 'd_phone_no', 'rules' => 'required'),
78              array('field' => 't_bprice', 'rules' => 'required'),
79              array('field' => 't_rate', 'rules' => 'required')
80          );
81          $this->form_validation->set_rules($validation1);
82
83          $tname=$this->input->post('t_name', TRUE);
84          $dname=$this->input->post('d_name', TRUE);
85          $truckno=$this->input->post('truck_no', TRUE);
86          $dphoneno=$this->input->post('d_phone_no', TRUE);
87          $tbprice=$this->input->post('t_bprice', TRUE);
88          $trate=$this->input->post('t_rate', TRUE);
89
90          $auth=$this->Details_m->add_driver($tname,$dname,$truckno,$dphoneno,
                $tbprice,$trate);
91
92          if($this->form_validation->run() === FALSE){
93              redirect(base_url('Admin_c/content'));
94          }
95
96          if($auth){
97              redirect(base_url('Admin_c/content'));
98              echo "Truck Added Sucessfully";
99          }
100     }
101
102     //Edit Truck Details
103     public function update_truck(){
104         $d_number=$this->input->post('dnumber', TRUE);
105         $b_price=$this->input->post('bprice', TRUE);
106         $rate=$this->input->post('rate', TRUE);
107         $t_id=$this->input->post('tid', TRUE);
108
109         $update=$this->Details_m->update_truck($d_number,$b_price,$rate,$t_id);
110         if($update){
111             redirect(base_url('Admin_c/content'));
112             echo "Truck Added Sucessfully";
113         }
114     }
115
116     // Delete Truck
117     public function delete_truck(){
118         $d_id = $this->input->get('d_id');
119         $dlt=$this->Details_m->delete_driver($d_id);
120         if($dlt){
121             redirect(base_url('Admin_c/content'));
122             echo "Truck Deleted Succesfully";
123         }
124     }
125
126     public function a_logout(){
127         unset($_SESSION['aname']);
128         session_destroy();
129         redirect(base_url('Admin_c'));
130     }
131 }
```

```
?>

USER LOGIN:
class Login_c extends CI_Controller
{
    public function __construct(){
        parent::__construct();
        $this->load->model('login_m');
    }

    public function signup(){
        $validation1 = array(
            array('field' => 'email', 'rules' => 'required'),
            array('field' => 'username', 'rules' => 'required'),
            array('field' => 'mobile', 'rules' => 'required'),
            array('field' => 'password', 'rules' => 'required')
        );
        $this->form_validation->set_rules($validation1);

        $email1=$this->input->post('email', TRUE);
        $uname1=$this->input->post('username', TRUE);
        $number1=$this->input->post('mobile', TRUE);
        $password1=$this->input->post('password', TRUE);
        $en_pass1=hash('sha1',$password1);

        $auth1=$this->login_m->signup($email1,$uname1,$number1,$en_pass1);

        if($this->form_validation->run() === FALSE){
            redirect(base_url('Home/signup'));
        }
        if($auth1){
            redirect(base_url('Home/login'));
            echo "Signed In";
        }

    }

    public function login(){

        $validation = array(
      array('field' => 'email', 'rules' => 'required'),
      array('field' => 'password', 'rules' => 'required')
    );
        $this->form_validation->set_rules($validation);

        $email=$this->input->post('email', TRUE);
        $password=$this->input->post('password', TRUE);
        $en_pass=hash('sha1',$password);

        $auth=$this->login_m->login($email,$en_pass);

        if($this->form_validation->run() === FALSE){
        //$this->session->set_flashdata('error', validation_errors());

        redirect(base_url('Home/login'));
        }

        if($auth){
             //$r['result']= $res;
             //$data['result']=$res;
            session_start();
```

```
193                 $r = $auth;
194                 echo $r;
195                 $_SESSION['uname'] = $r;
196                 //$_SESSION['email']=$email;
197                 redirect(base_url('Home/book'));
198 
199                 //print_r($r);
200             }
201             else{
202                 redirect(base_url('Home/login'));;
203             }
204         }
205 
206         public function logout(){
207             unset($_SESSION['uname']);
208             session_destroy();
209             redirect(base_url());
210         }
211 
212 }
213 
214 
215 ?>
216 
217 
218 DRIVER:
219 <?php
220 defined('BASEPATH') OR exit('No direct script access allowed');
221 
222 class Driver extends CI_Controller
223 {
224   public function __construct(){
225             parent::__construct();
226             $this->load->model('Booking_m');
227             $this->load->model('Details_m');
228         }
229 
230   public function index()
231     {
232       $this->load->view('header/header');
233         $this->load->view('driver/content');
234         $this->load->view('footer/footer');
235     }
236 }
237 
238 ?>
```

6.2 Booking

After searching for the particular vehicle, if user is satisfied with the terms and conditions, he books up the vehicle through booking module. Firstly, he has to get log into the system and then only he/she can book it.

A booking id is being assigned after booking is done to all the user, invoice is being generated and a particular driver is assigned to the user. All these details get stored into the database of the system and can be accessed remotely by the admin.

```
1
2   <title>MIHU|Booking</title>
3 <?php error_reporting(0); ?>
4 <section>
5   <?php
6     session_start();
7     if(!isset($_SESSION['uname'])){
8       header('Location:'.base_url('Home/login'));
9     }
10  ?>
11    <h2>Bookings</h2>
12  <?php
13    echo "Your current Booking<br>";
14    echo "Your distance is: ".$dist = $_REQUEST['dist']."<br>";
15    echo "Your Estimated Price is:".$out = $_REQUEST['out']."<br><br>";
16    echo "<button class='btn blue'>Book</button>";
17    // $cost = $_REQUEST('');
18  ?>
19 </section>
20
21 <script>
22   onload{
23     initMap();
24   }
25 </script>
26
27 <div style="height:150px"></div>
```

6.3 Tracking

User can view the live update of the vehicle means he/she can track the vehicle lively through Google Map API. By using Google Map's API, the estimated amount of time, cost and distance is also calculated.

```
<script>
 onload{
   initMap();
 }
</script>


function initMap() {
  var origin = $('#start').val();
  var destination = $('#end').val();
  // var cost = $('#mobile').val();
  if(origin == "" || destination == "")
    return false;
  else{
    var directionsDisplay = new google.maps.DirectionsRenderer();
    var directionsService = new google.maps.DirectionsService();
    var bounds = new google.maps.LatLngBounds;
    var from = document.getElementById('from');

    var autocomplete = new google.maps.places.Autocomplete(from);
    var map = new google.maps.Map(document.getElementById('map'), {});

    var service = new google.maps.DistanceMatrixService;
    directionsDisplay.setMap(map);
    service.getDistanceMatrix({
      origins: [origin],
      destinations: [destination],
      travelMode: 'DRIVING',
      unitSystem: google.maps.UnitSystem.METRIC,
      avoidHighways: false,
      avoidTolls: false },
      function(response, status) {
        if (status !== 'OK') {
          alert('Error was: ' + status);
        } else {
          var originList = response.originAddresses;
          var destinationList = response.destinationAddresses;
          var outputDiv = document.getElementById('output');
          var waqt = $('#waqt');
          var doori = $('#doori');


          outputDiv.innerHTML = '';
          var showGeocodedAddressOnMap = function(asDestination) {
            return function(results, status) {
              if (status === 'OK') {
                map.fitBounds(bounds.extend(results[0].geometry.location));
                markersArray.push(new google.maps.Marker({
                  map: map,
                   position: results[0].geometry.location,
```

```
                }));
              } else {
                alert('Geocode was not successful due to: ' + status);
              }
            };
          };

          function calculateRoute(){
            var request = {
              origin: origin,
              destination: destination,
              travelMode:'DRIVING'
            };
            directionsService.route(request,function(result,status){
              console.log(result,status);
              directionsDisplay.setDirections(result);
            });}
            calculateRoute();

            for (var i = 0; i < originList.length; i++) {
              var results = response.rows[i].elements;

              for (var j = 0; j < results.length; j++) {
              // outputDiv.innerHTML =
              // '<strong>Results</strong><br>'+
              // 'Your Destination is '+
              // results[j].distance.text + ' away &<br> it will take ' +
              // results[j].duration.text + ' to reach there.';
              // waqt.text = results[j].duration.text;
              // doori.text = results[j].distance.text;
              // alert(waqt,doori);
              var waqt= results[j].duration.text;
              var res = waqt.slice(0, -4);
              var doori = results[j].distance.text;
              var res1 = doori.slice(0, -2);
              var cost = $('#truck').val();

              d = doori;
              out2= res1*cost;
              out3= out2.toFixed(2);
              outputDiv.innerHTML = '<strong>Results</strong><br>'+
              'Your Destination from source is: '+ doori + ' away.<br>'+
              'Your estimated time is: '+ waqt + '.<br>'+
              'Your estimated cost is: Rs.'+ out3 + '.<br>'+
              '<button class="btn blue" onclick="datasend()">Book</button>';

            }
          }
        }
      });
  }}
  function datasend(){
    var dist = d;
    var out = out3;
                window.location.href="Home/book?dist="+dist+"&out="+out;
              }

              function place() {

                var countryRestrict = {'country': 'in'};
                var countries = {
```

```
                'in': {
                  center: {lat: 19.0760, lng: 72.8777},
                  zoom: 12},
                };
                var start = document.getElementById('start');
                var autocomplete = new google.maps.places.Autocomplete(start);
                autocomplete = new google.maps.places.Autocomplete((start),{
                    componentRestrictions: countryRestrict});

                var start = document.getElementById('end');
                var autocomplete = new google.maps.places.Autocomplete(end);
                autocomplete = new google.maps.places.Autocomplete((end),{
                    componentRestrictions: countryRestrict});

              }
              $(document).ready(function() {

                $("#start-link, #end-link").click(function(event) {
                  var addressId = this.id.substring(0, this.id.indexOf("-"));
                  navigator.geolocation.getCurrentPosition(function(position)
                      {
                    var geocoder = new google.maps.Geocoder();
                    geocoder.geocode({
                      "location": new google.maps.LatLng(position.coords.
                          latitude, position.coords.longitude)
                    },
                    function(results, status) {
                      $("#" + addressId).val(results[0].formatted_address);
                    });
                  },
                  function(positionError){},
                  {
                    enableHighAccuracy: true,
            timeout: 10 * 1000 // 10 seconds
          });
                });

              });
```

6.4 Driver

Driver has to get sign in to the system on duty time. Whoever the driver is online, gets connected to the system and a user is assigned to them. Driver picks up the goods from the source location and dispatches them to the destination location of the user. As the driver is off duty, he gets sign out of the system.

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Driver extends CI_Controller
{
  public function __construct(){
        parent::__construct();
        $this->load->model('Booking_m');
        $this->load->model('Details_m');
    }

  public function index()
    {
      $this->load->view('header/header');
        $this->load->view('driver/content');
        $this->load->view('footer/footer');
    }
}

?>

Fetching Driver's location:

<!DOCTYPE html>
<html>
<head>
  <title>Geolocation</title>
  <meta name="viewport" content="initial-scale=1.0, user-scalable=no">
  <meta charset="utf-8">
  <style>
      /* Always set the map height explicitly to define the size of the div
      * element that contains the map. */
      #map {
        height: 100%;
      }
      /* Optional: Makes the sample page fill the window. */
      html, body {
        height: 100%;
        margin: 0;
        padding: 0;
      }
  </style>
</head>
<body>
    <div id="map"></div>
  <script>
      // Note: This example requires that you consent to location sharing when
      // prompted by your browser. If you see the error "The Geolocation service
```

```
      // failed.", it means you probably did not give permission for the browser to
      // locate you.
      var map, infoWindow;
      function initMap() {
        map = new google.maps.Map(document.getElementById('map'), {
          center: {lat: 19.0760, lng: 72.8777},
          zoom: 20
        });
        infoWindow = new google.maps.InfoWindow;

        // Try HTML5 geolocation.
        if (navigator.geolocation) {
          navigator.geolocation.getCurrentPosition(function(position) {
            var pos = {lat: position.coords.latitude,lng: position.coords.longitude};
            document.getElementById('lol').innerHTML = pos;
            infoWindow.setPosition(pos);
            infoWindow.setContent('Gotch You');
            infoWindow.open(map);
            map.setCenter(pos);
          }, function() {
            handleLocationError(true, infoWindow, map.getCenter());
          });
        } else {
          // Browser doesn't support Geolocation
          handleLocationError(false, infoWindow, map.getCenter());
        }
      }
    }
</script>
<p id="lol"></p>

<script async defer
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBJNd0XSpFhQMeQPOFylt3TJhIk1c6t3_0&callback=initMap">
</script>
</body>
</html>
```

Chapter 7

System Testing

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Login	Should be register user	Will connect to database	Should login into system
T02	Map	Should show route with time	Will connect to Google API	Calculate the route and time
T03	Tracking	Should track the device	Device sending longitude and latitude through gps	Drivers is tracked

7.2 Sample of a Test Case

Title: Login Page

Description: A registered user should be able to successfully login.

Precondition: The user must already be registered with an email address and password.

Assumption: A supported browser is being used.

Test Steps:

1. Enter source and destination
2. Route showing through google map api.
3. Click the ‘Book’ button.

4. Driver Assigned
5. Delivery Confirmed.

Expected Result: The page will display a google map and the selected route, after booking the page will display estimated time and cost.

Actual Result:

The page is displaying results from source to destination distance in kilometers, the page is also displaying estimation time and also showing cost.The page is also showing all input which is given by user and showing on map.

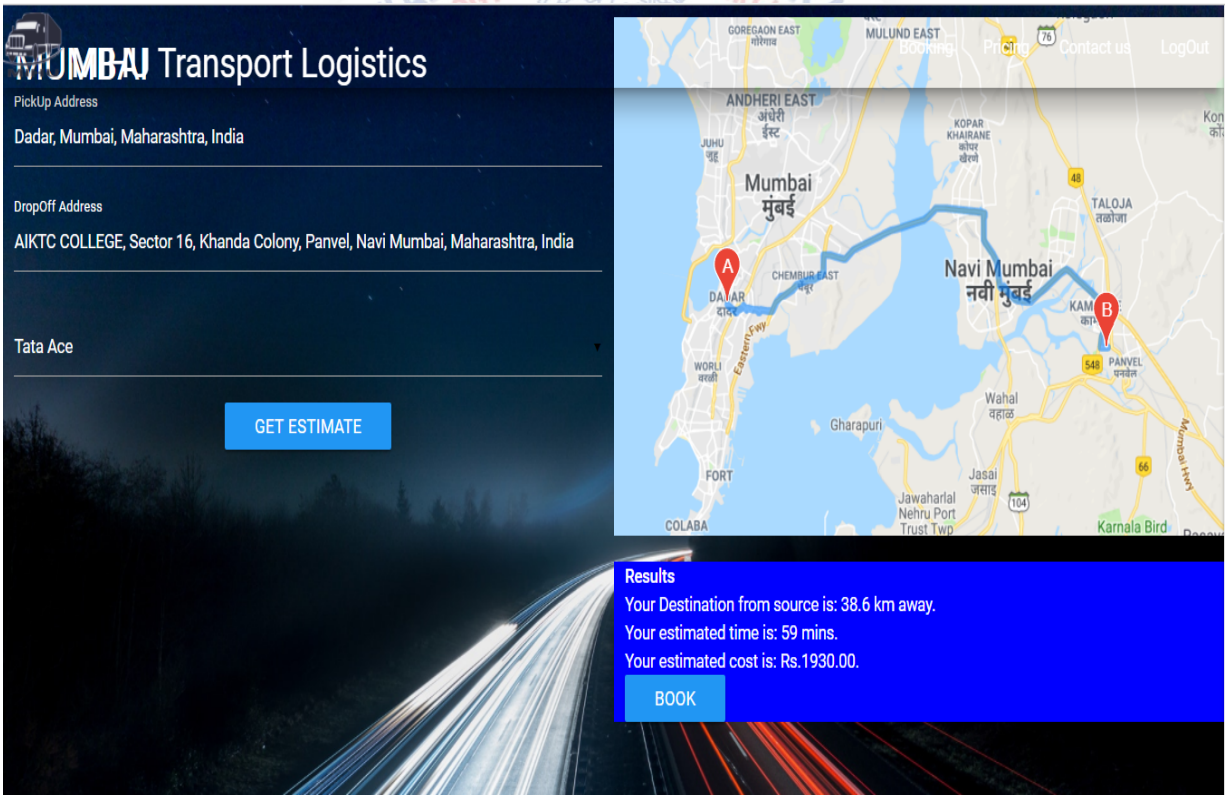


Figure 7.1: Test

7.2.1 Software Quality Attributes

1. AVAILABILITY: The system should not be down,whenever the user use the system the specific data should be available to the user.
2. CORRECTNESS: As per the user search the correct data should be shown to the user like user is selecting the route the same route should be shown to the user.
3. MAINTAINABILITY: The administrators of the system will maintain the system with effective updates though on air update if needed.
4. EXTENSIBILITY: The system is capable to be modified by changing some modules or by adding some features to the existing system

Chapter 8

Screenshots of Project

8.1 Home Page

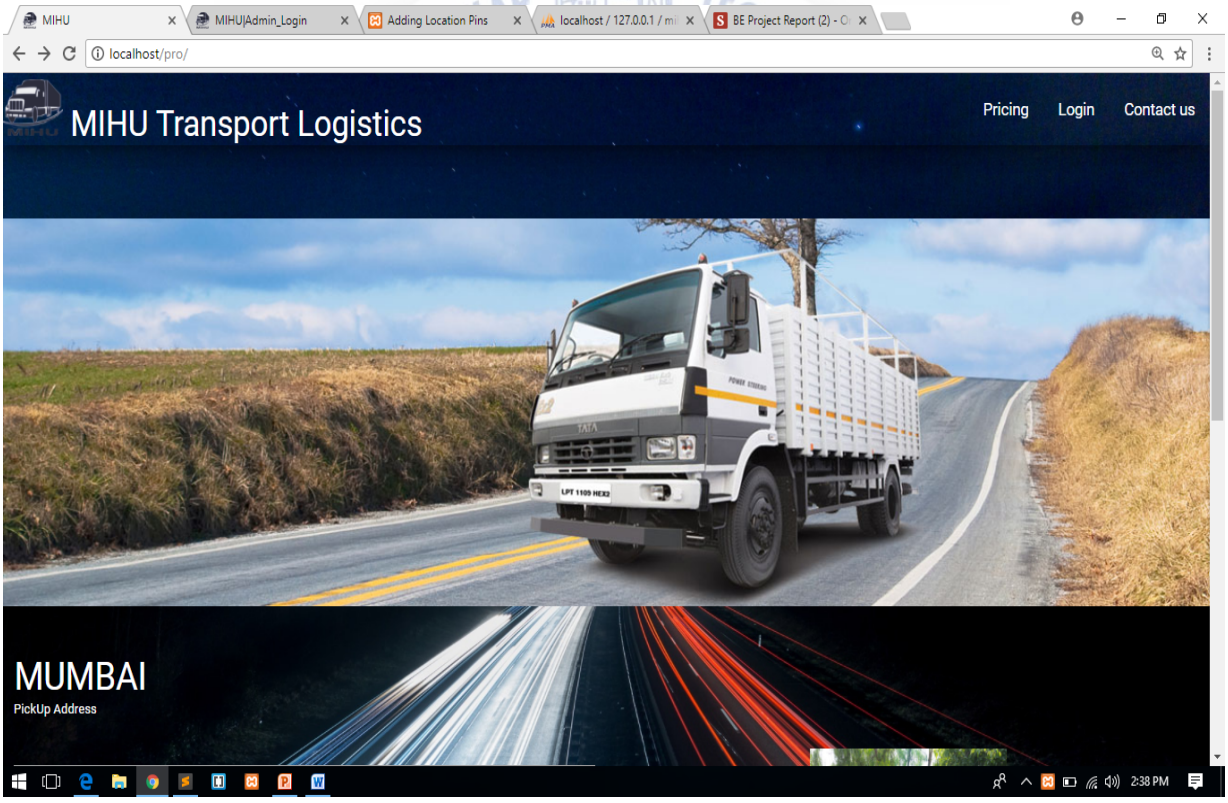


Figure 8.1: Home Page

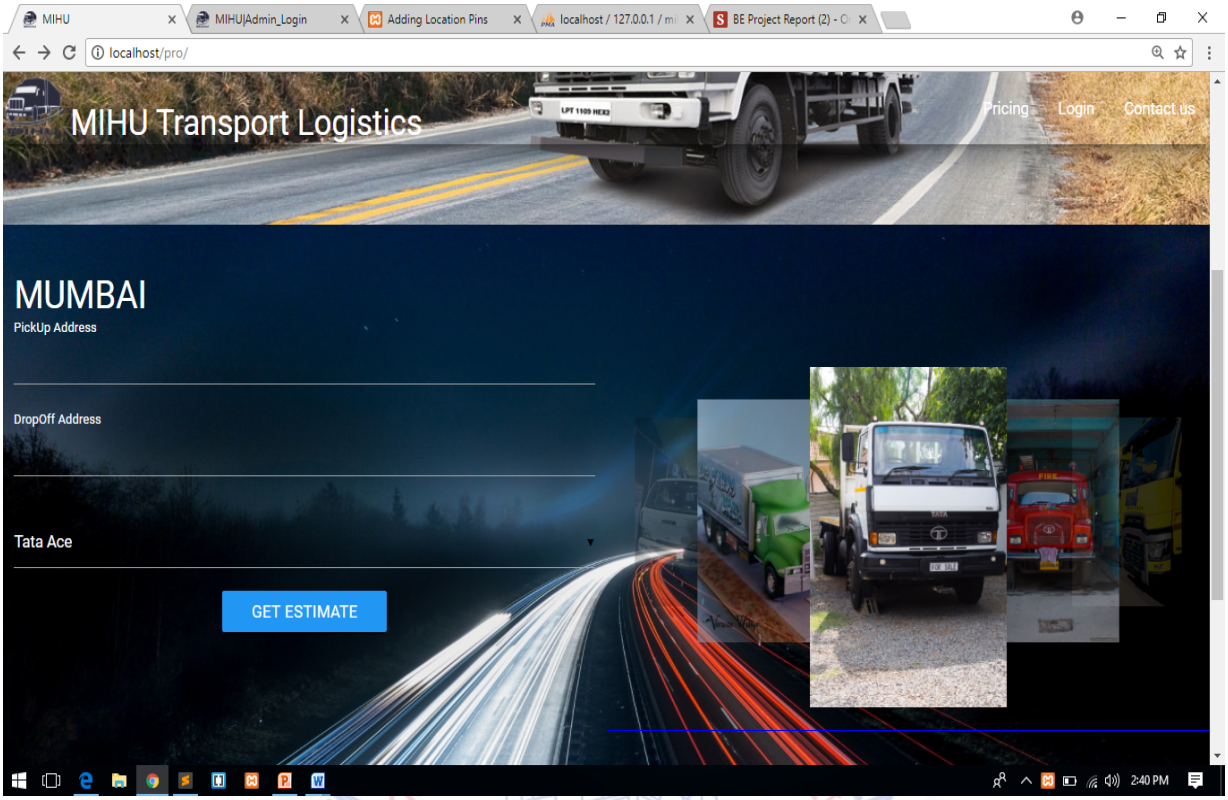


Figure 8.2: Home Page

8.2 User Signup

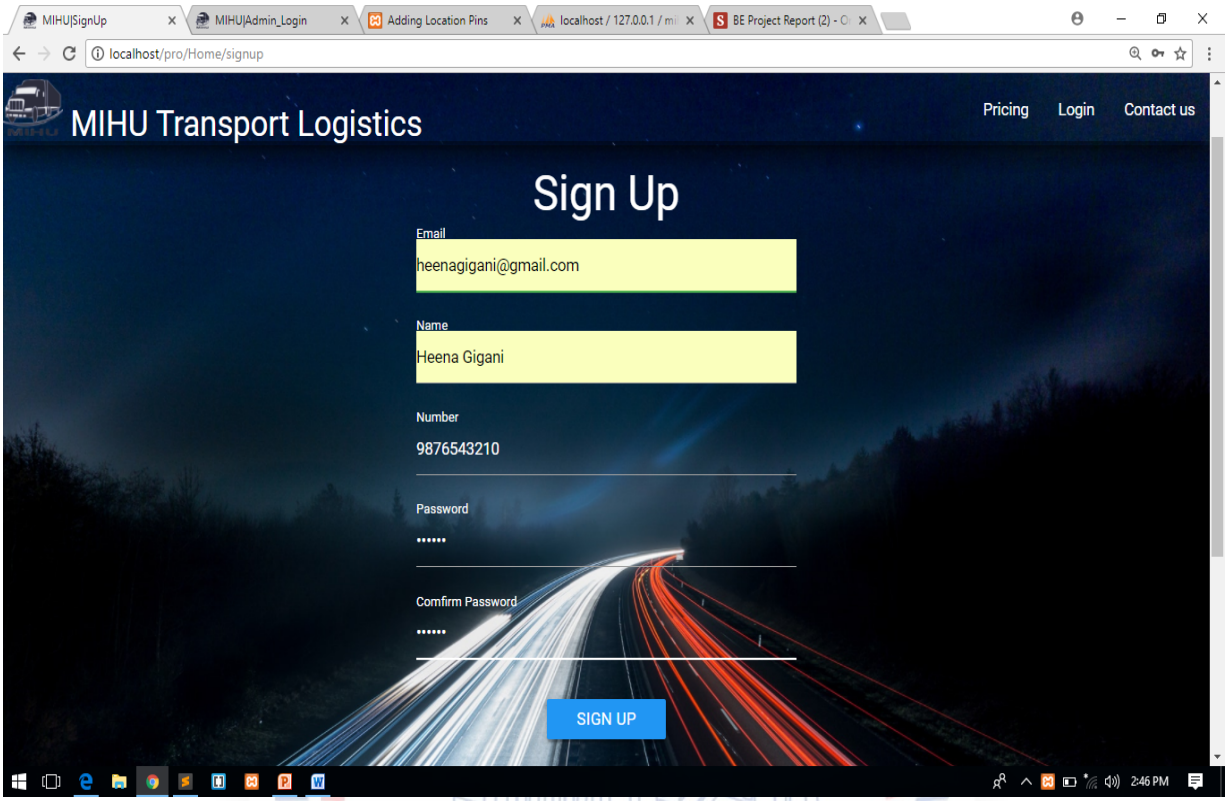


Figure 8.3: User Signup

8.3 User Login

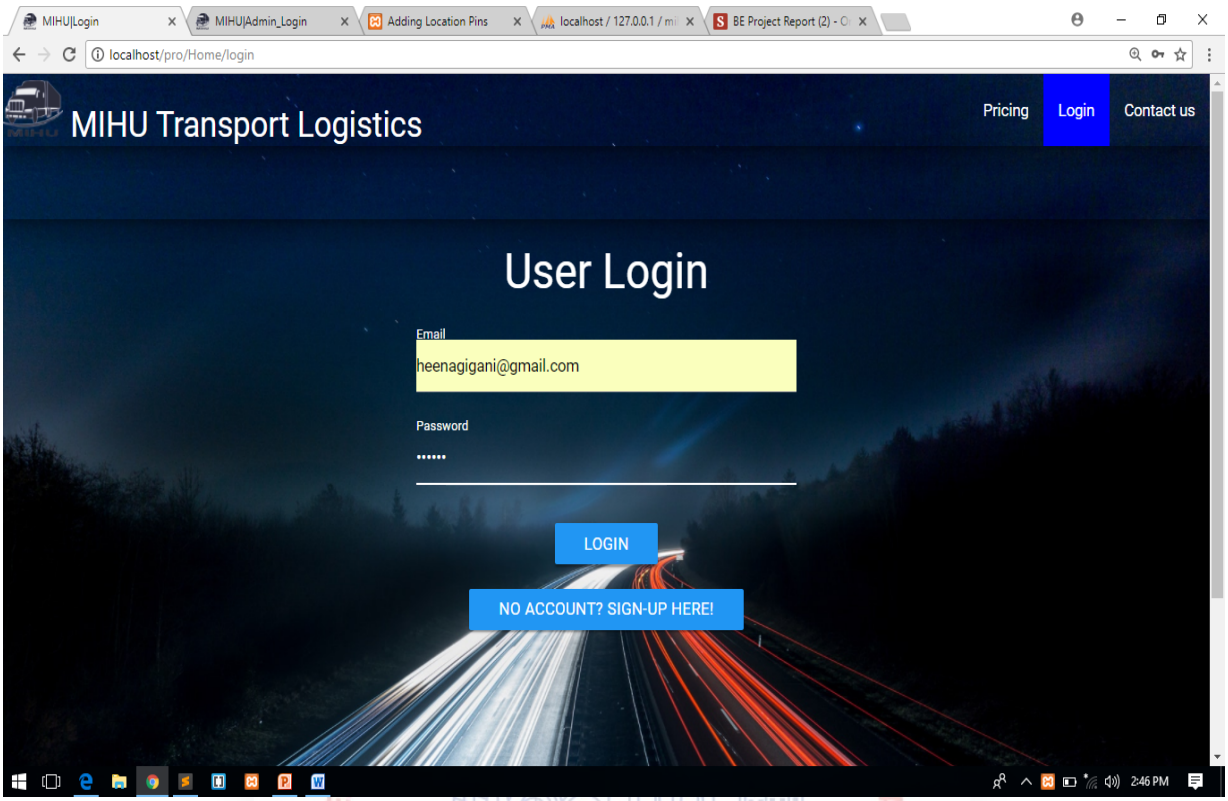


Figure 8.4: User Login

8.4 Price of vehicles and Booking

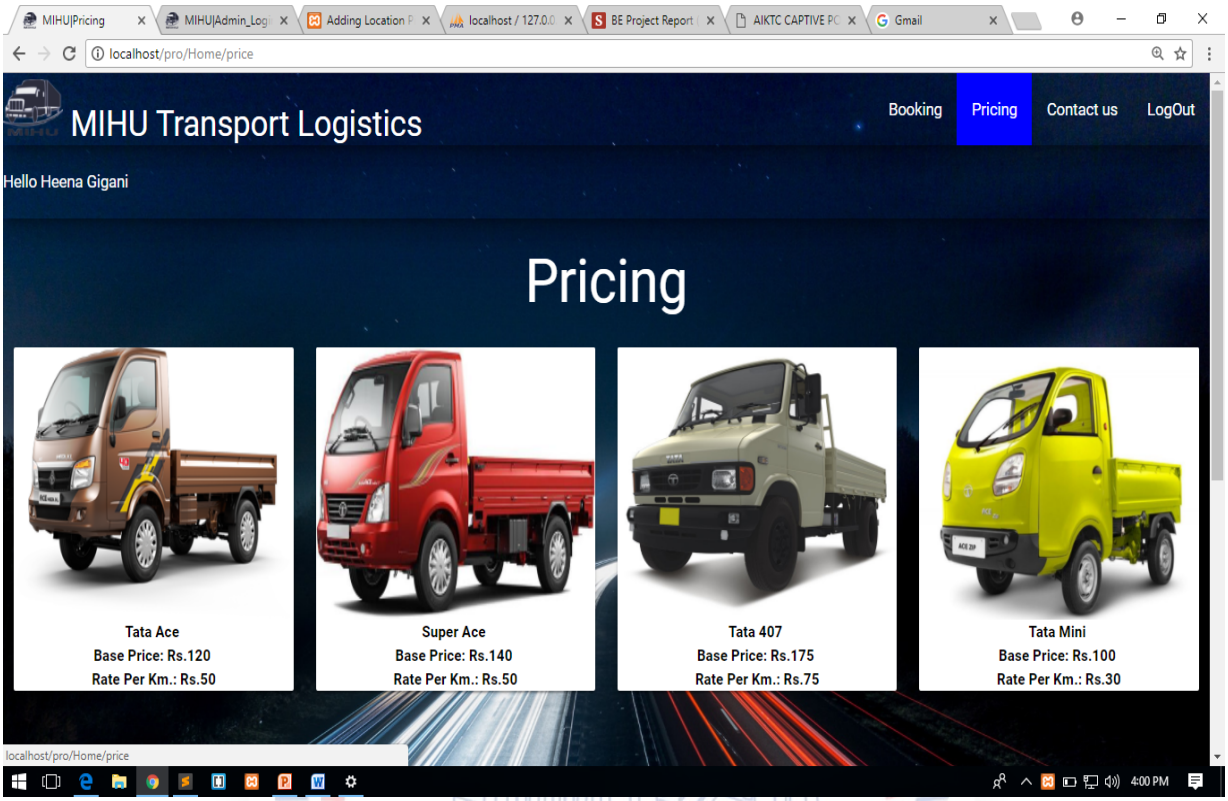


Figure 8.5: Rate

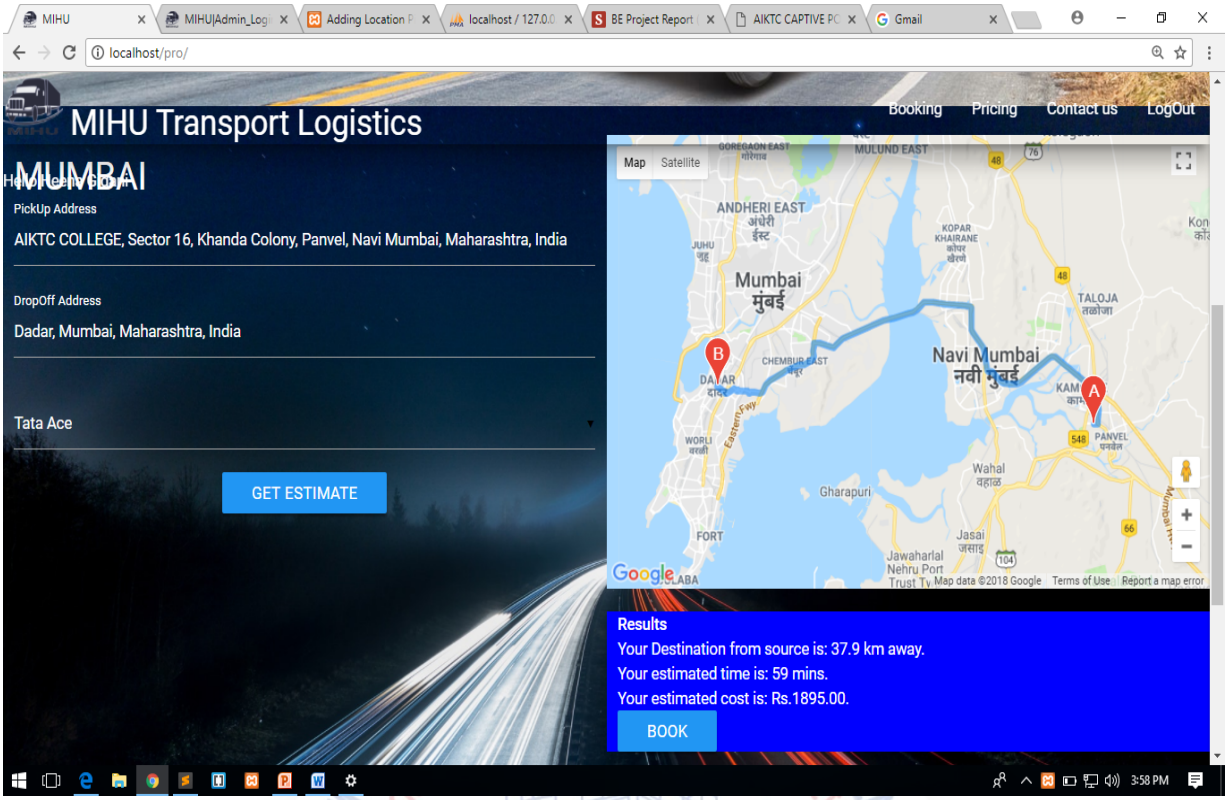


Figure 8.6: Rate

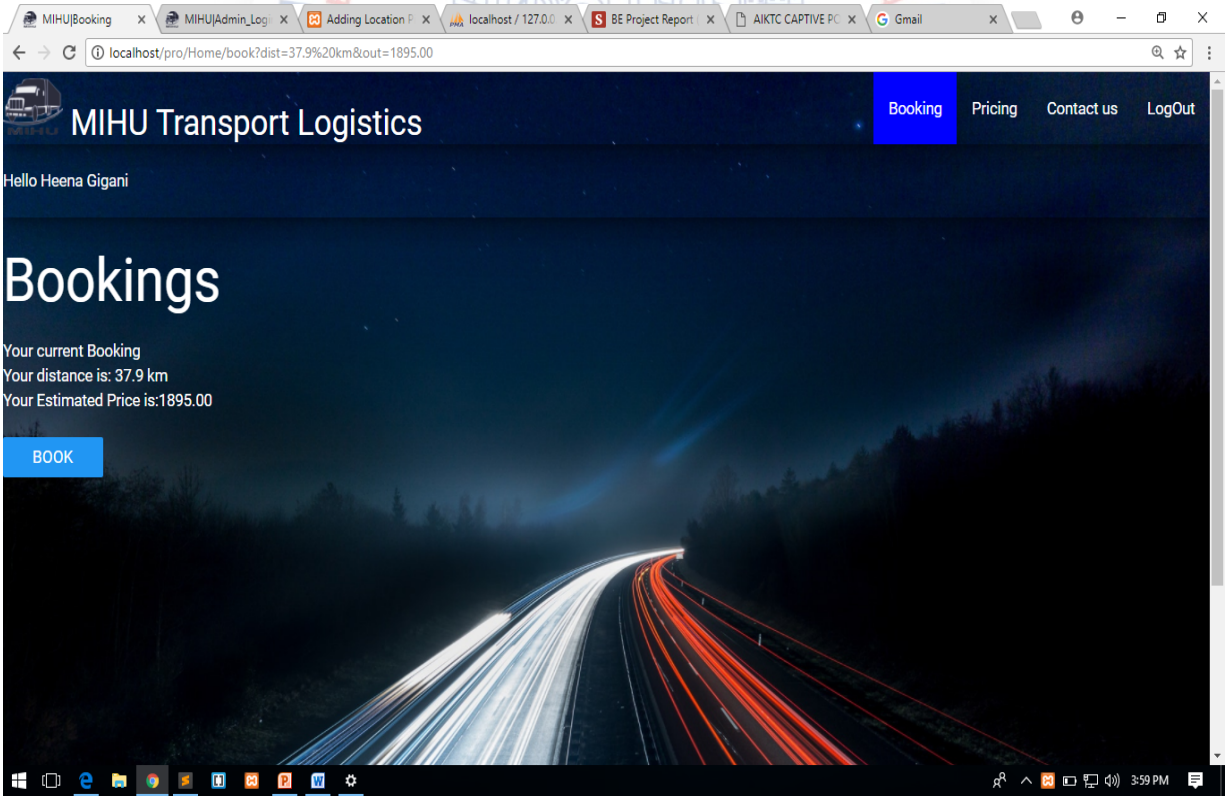


Figure 8.7: Book

8.5 Driver Login

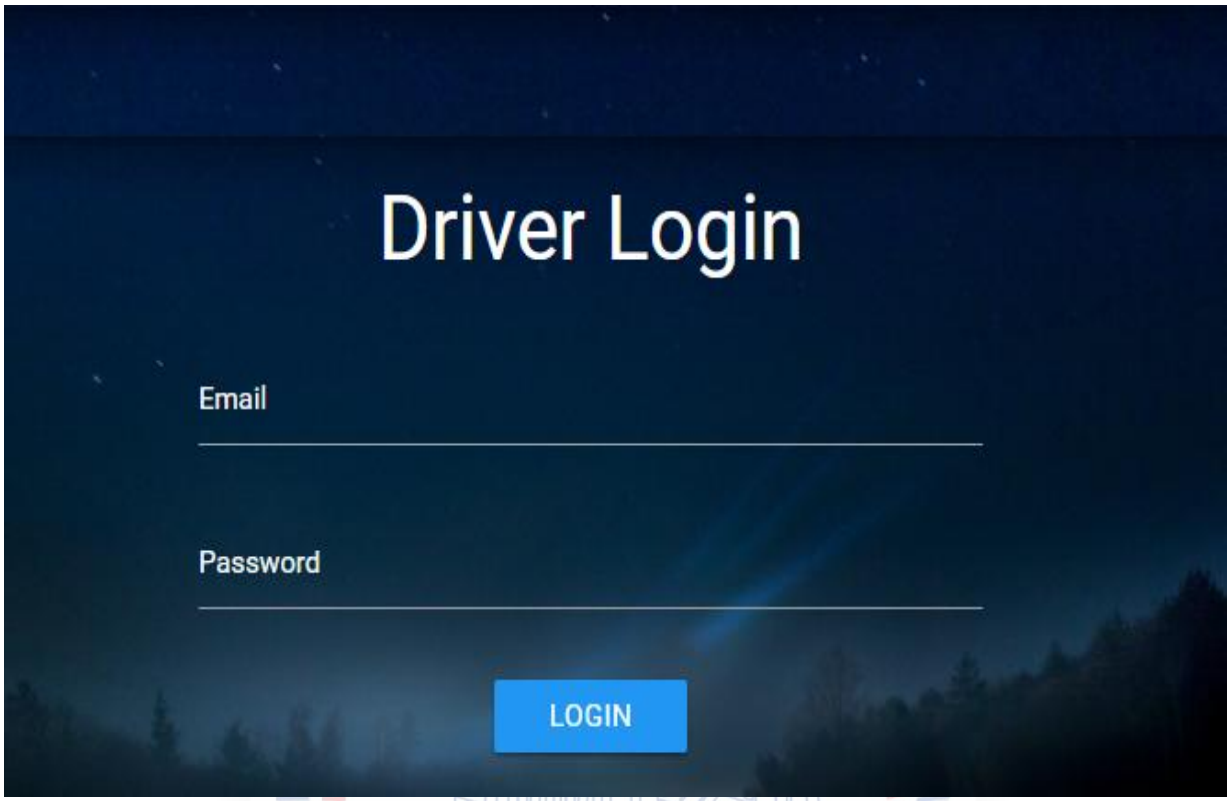


Figure 8.8: Driver Login

8.6 Admin Login

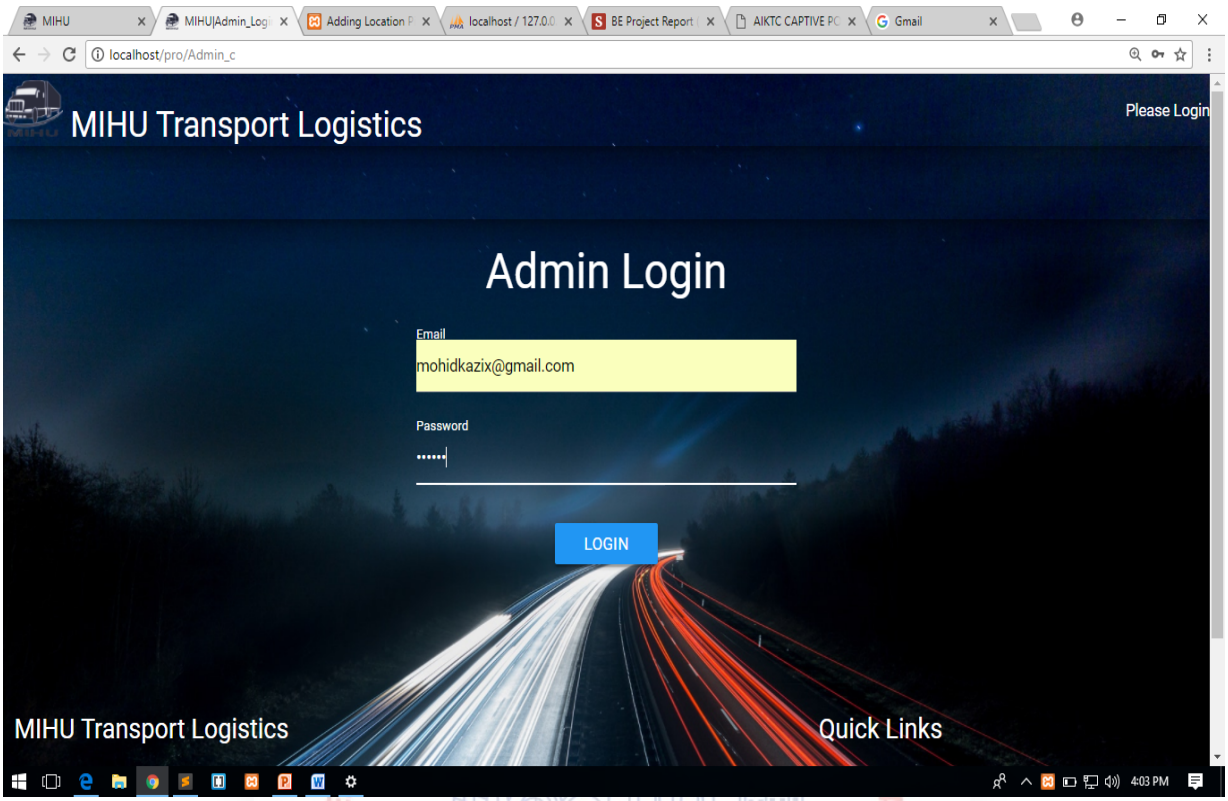


Figure 8.9: Admin Login

8.7 Add Truck

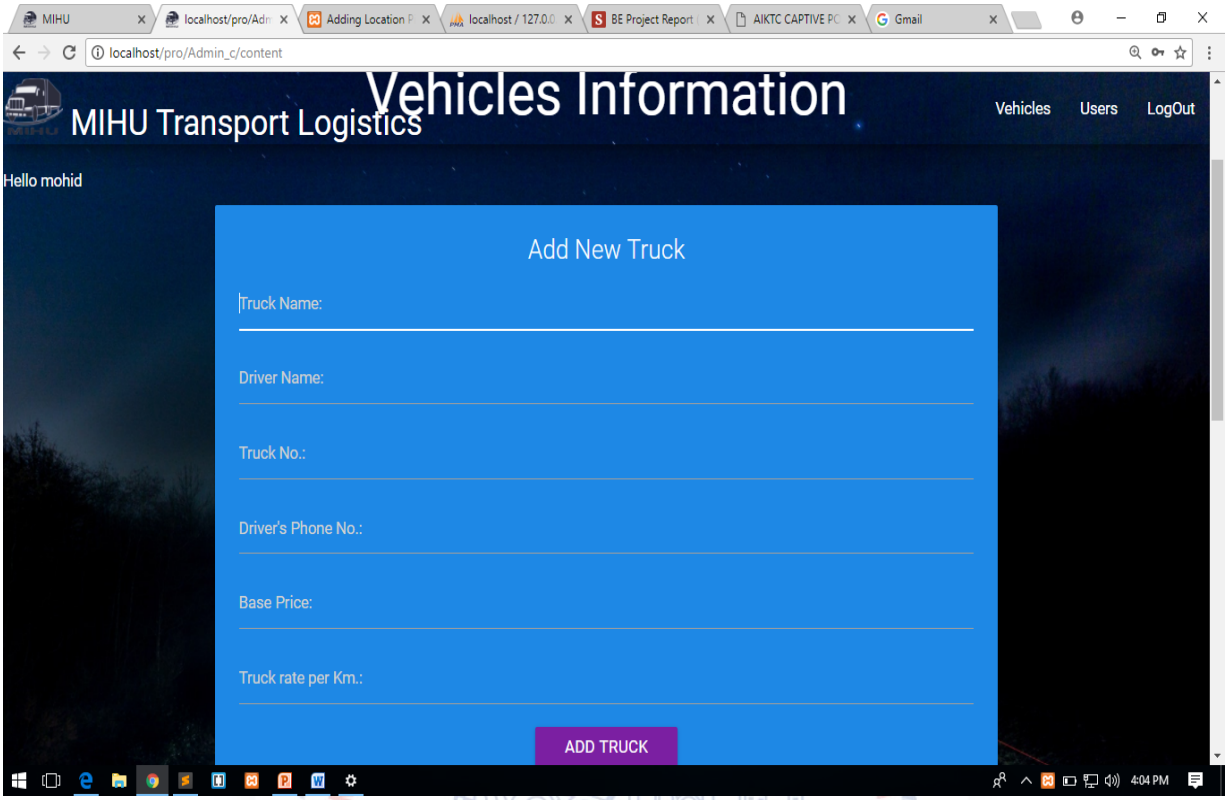


Figure 8.10: Add Trucks

8.8 Update truck

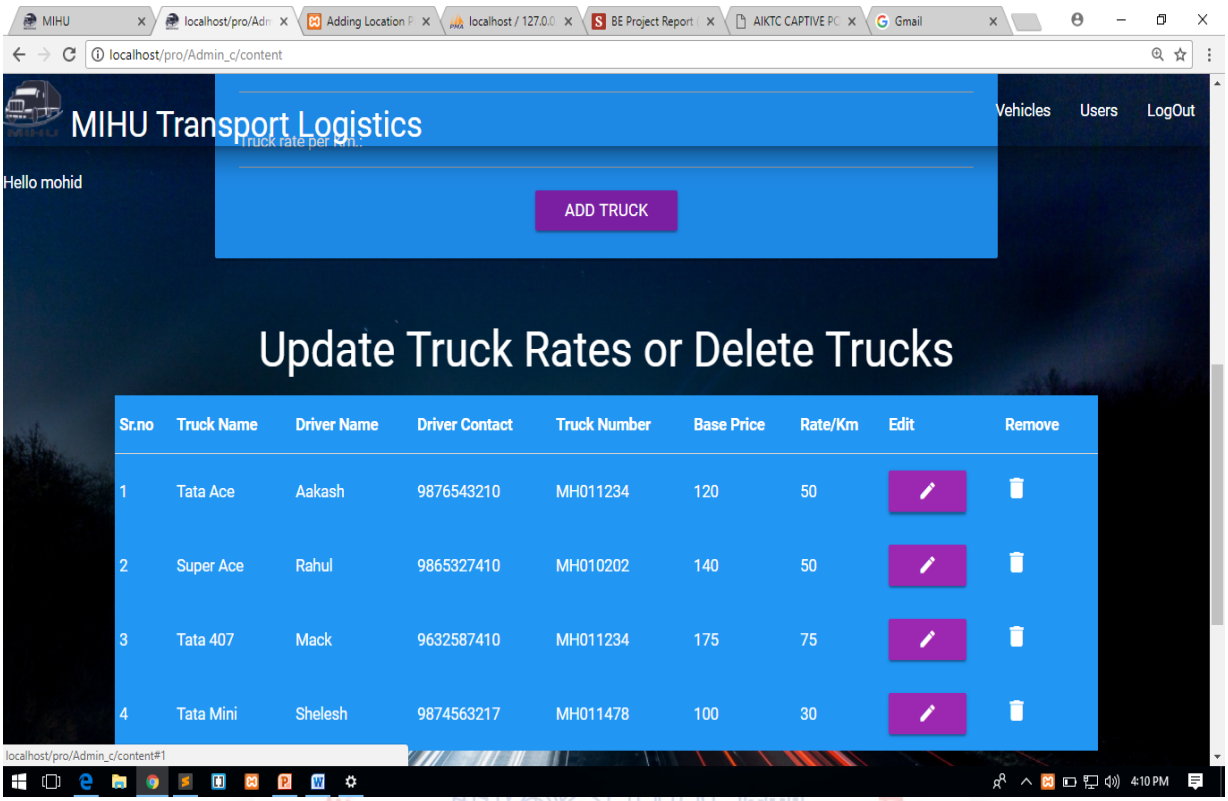


Figure 8.11: Update truck

Chapter 9

Conclusion and Future Scope

9.1 Conclusion

Through this system, user can easily search for the vehicle and get the estimated amount of time, cost and distance for the transportation of goods and can book them on one click. Here the user can lively track their goods carrying vehicle and stays up to date.

9.2 Future Scope

In future, various modules can be added for extending the feasibility of the system. Somewhat are mentioned below.

- Application in Android and iOS will be built for more feasibility.
- It can be made for intercity transportation.

References

[1] *Recent Trends in intelligence transportation systems:*; a review'1 IIT Roorkee, Roorkee, Uttarakhand, India 2 IIT, BHU, Varanasi, India

[2] *Disruptive change in the Taxi Business*; a review'1 IIT Roorkee, Roorkee, Uttarakhand, India 2 IIT, BHU, Varanasi, India

[3] *Transportation site link* `http://www.porter.in`

Achievements

1. Publications

   (a) *E-Logistic System*; Heena Mohd. Ashraf Gigani , Usman Gani M.H. Kadariya , Imran Mukhtar Ahmed Shaikh , Mohid Mahmood Kazi , Prof. Javed Khan Sheikh,IJISRT (International Journal of Innovative Science and Research Technology) , 2nd February 2018(`https://ijisrt.com/e-logistic-system`)

2. Project Competitions

   (a) *E-Logistic System*; Heena Mohd. Ashraf Gigani , Usman Gani M.H. Kadariya , Imran Mukhtar Ahmed Shaikh , Mohid Mahmood Kazi , Prof. Javed Khan Sheikh, Poster Competition at Universal College of Engineering , 9th March 2018(Venue : Vasai)

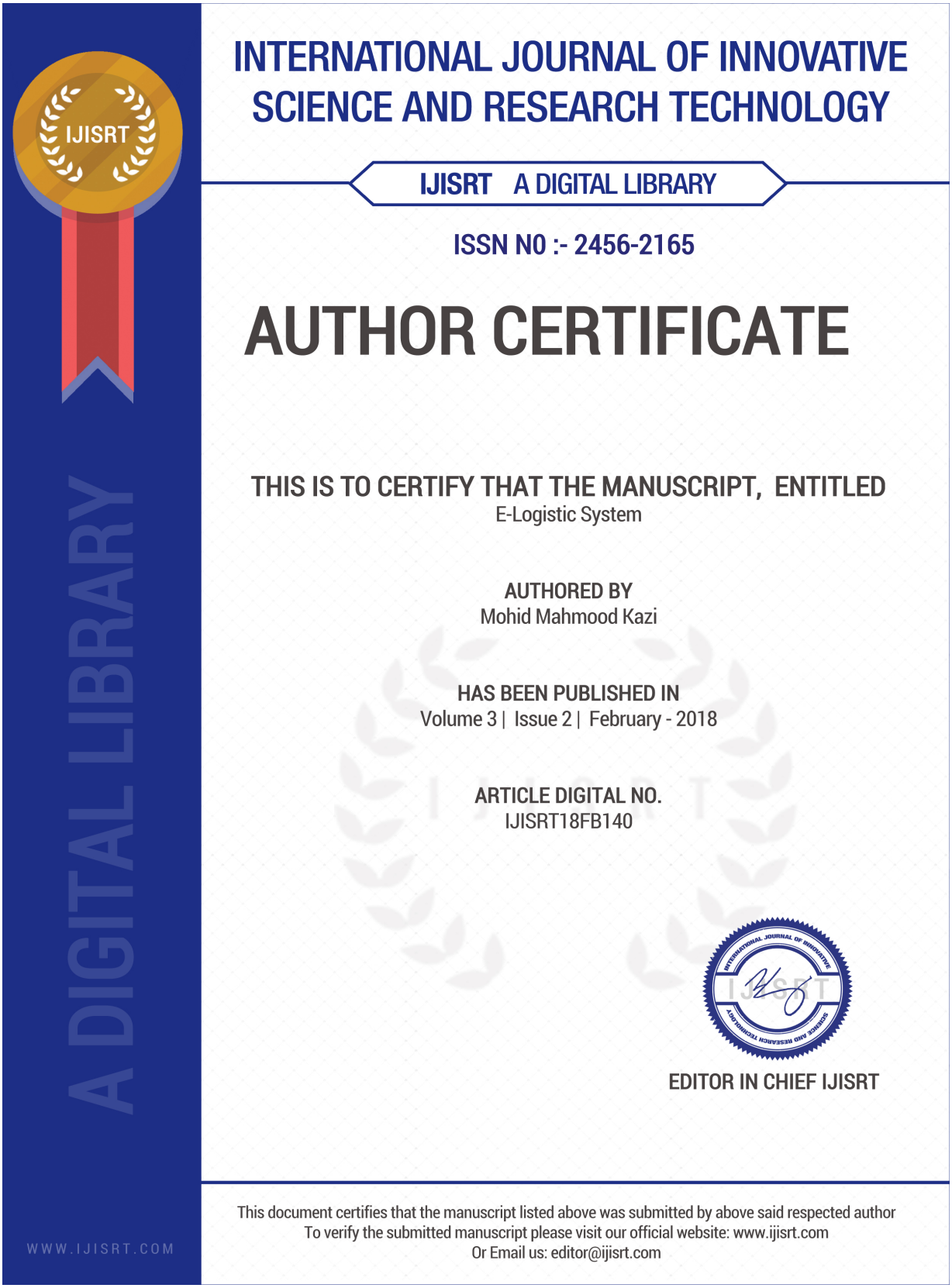


Figure 9.1: Certificate

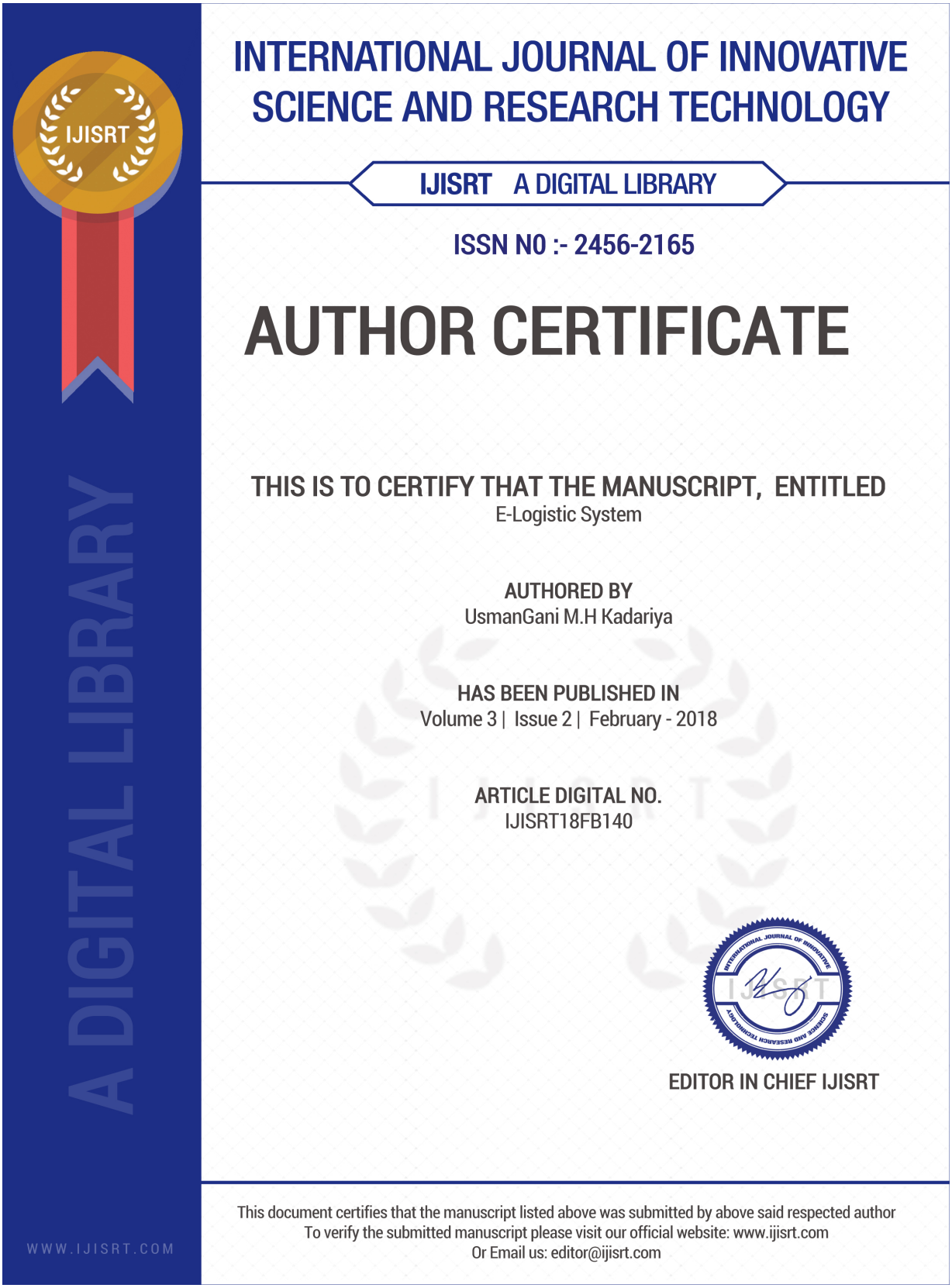


Figure 9.2: Certificate

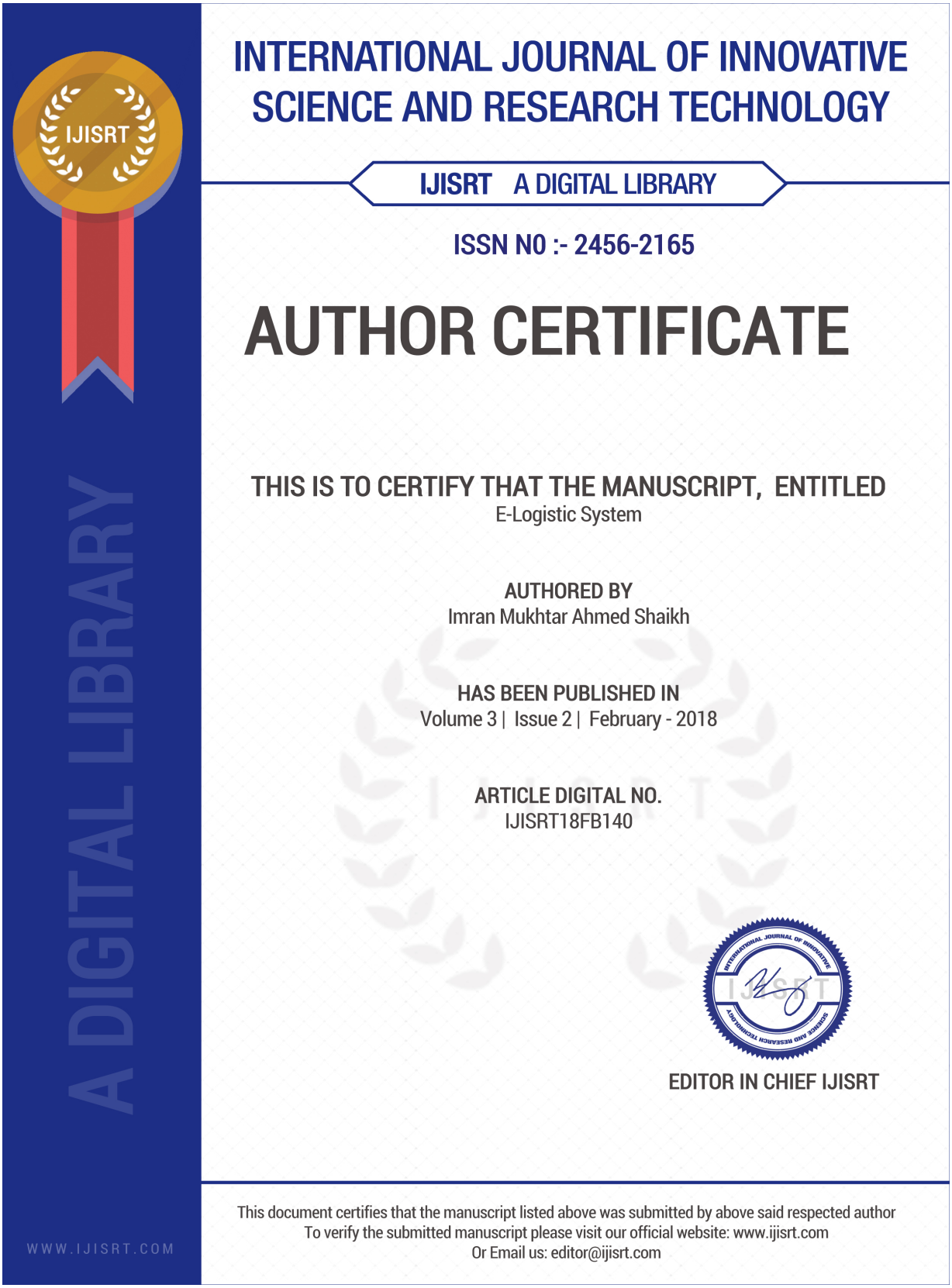


Figure 9.3: Certificate

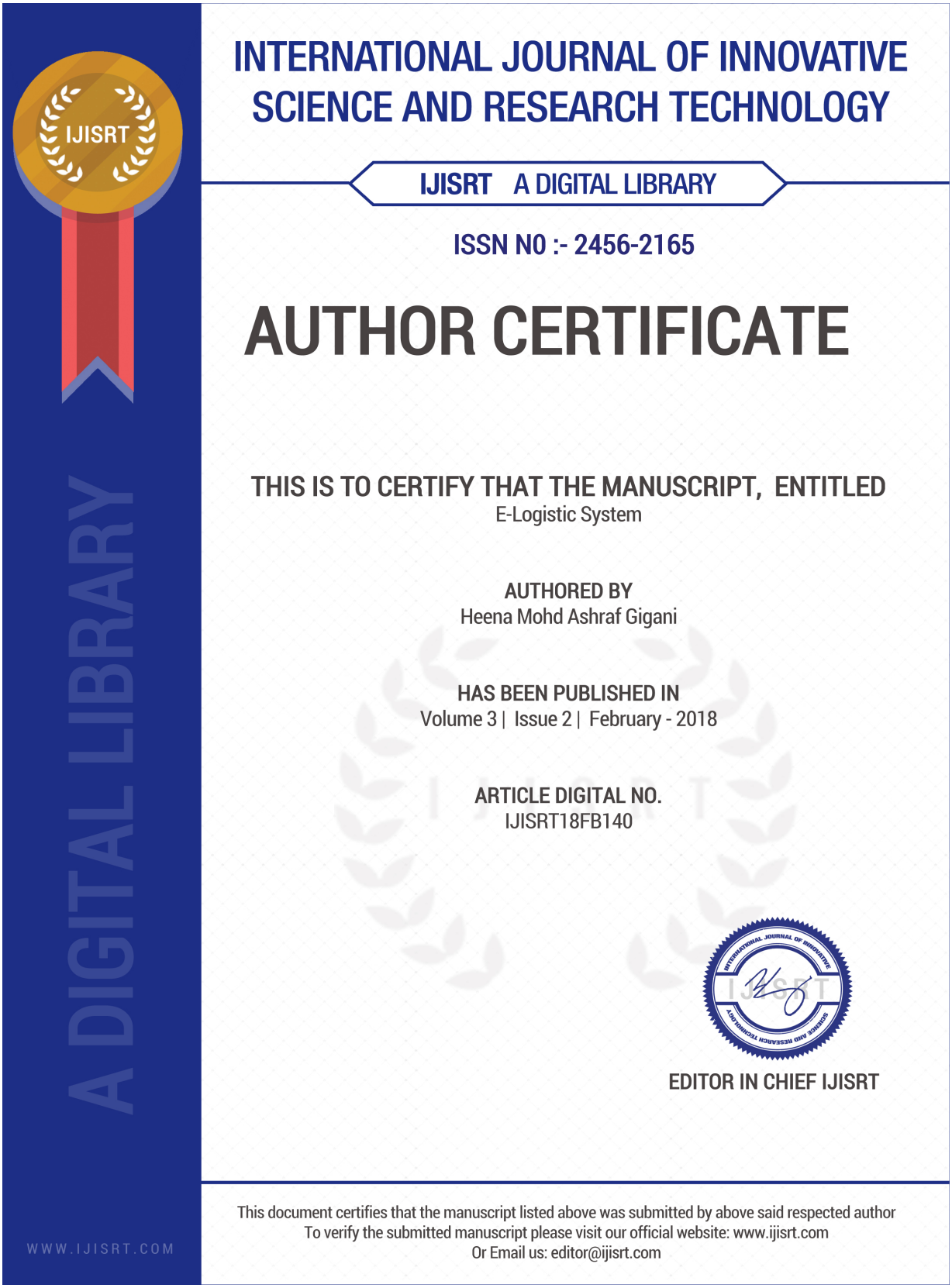


Figure 9.4: Certificate

Figure 9.5: Certificate

Figure 9.6: Certificate

Figure 9.7: Certificate

Figure 9.8: Certificate