Hawassa University IOT

School of Informatics

Computer Science Department

Industrial Project Documentation

On

Hawassa City Traffic Control and Management System

Project Advisor Mr. Mulat S.

Submitted to School of Informatics

Submission Date: Feb 23/2016

Members

Name	ID	Sign
Getabalew Amtate	Comp/273/05	
Getnet Azmeraw	Comp/282/05	
Melaku Alehegn	Comp/390/05	
Tirufat Tesfaye	Comp/4750/05	

Advisor

Name	Mr. Mulat Shiferaw (MSc.)
Sign.	
Comment	

Supervisors

Name	Sign.	Remark

Acknowledgment

First of all, we would like to thank God for helping us to reach this time. Then we would like to thank Hawassa University Specially Department of computer science for providing us the opportunity to do this project & to fix several problems of the organization.

Then Special thanks for our advisor Mr. Mulat Sheferaw for continuous support in every phase of the project and who continuously provides us his valuable advice to prepare this project document. Also thanks to the Hawassa city Road Transport Authority employee and Hawassa City Traffic control officers who give us a correct requirement for the system developed by our group members.

Lastly, we would like to acknowledge all our friends for their morals, idea, materials and financial support during the courses of the years.

Abstract

This project is intended to develop web based application system with mobile for Hawassa traffic control and management system and provides simple and effective way of controlling and managing traffic. Ethiopian Transport authority has been performing complex and risky task with greater responsibility, from these responsibilities recording Drivers full information and town traffic flow control are the major tasks that the office undertakes, which needs efficiency and accuracy. To process this task in efficient way, the office should take the full information. The project "Traffic control and Management" has been selected which is aimed for helping the Hawassa Town Transport authority and Traffics police department. Our system will have centralized database and each traffic policemen will have user account that enables them to view and record criminal driver's status like warning and penalty. Authorized Administrators and Employee in the transport authority office will manage the system on their computer browser. They will have privilege to login to the system and they can Register new and view existing Driver Information and record as well as traffic police records.

List of Tables

Table of Contents

List of Acronyms

Ajax: Asynchronous JavaScript and XML.

APN: Access point

App: Application

CSS: Cascaded Style Sheet

DFD: Data Flow Diagram

FK: Foreign key

GUI: Graphical User Interface

HDD: Hard Disk Drive

HTCMS: Hawassa city Traffic control and management system

HTML5: Hyper Text Markup Language version five

JSON: JavaScript Object Notation

LAN: Local Area Network

NAS: Network Attached Storage

PHP: Hypertext Processing

PK: Primary Key

RTA: Road Transport Authority

TPA: Traffic police Administrator

UC: Use Case

UI: User Interface

UML: Unified Modeling Language

Chapter One

1. Introduction

1.1. Background

The Ethiopian Road Transport Authority (R.T.A) is a public transport authority based in Addis Ababa, Ethiopia with regional branches like SNNPR Road Transport Authority. It was originally founded as the Road Transport Administration in 1967 by proclamation No 256/67 but restructured and became the Road Transport Authority (RTA) in 1976, following proclamation No 107/76. The RTA states that its mission is "To ensure the provision of a modern, integrated and safe Road transport services to meet the needs of all the communities for strong and unitary economic and political system in Ethiopia". In doing so they attempt to promote an efficient road service, to coordinate and strengthen the road traffic safety and to develop the transport data base system to enhance research for the development of the sector.

In 2005, the Ethiopian government reorganized the transport sector and although the previously independent Urban Transport Authority, the Railway Regulatory Authority, the Aviation Authority, the Airport Administration Authority and the Maritime Regulatory Authority merged into the Ethiopian Transport Authority, the Road Transport Authority remained independent.

The Hawassa city road transport is branch of RTA which is responsible to SNNPR Road Transport office.
Existing System in Hawassa city road transport authority is manual record System. It has around 50,000 records of drivers including their name, number, category, age, ID and any information regarding their status like they have punished or not by traffic police. The punishment record has the following information: - name, driver license no, time the event has arrived, place, kind, level, plate number.
The main purpose of Hawassa city road transport office is to give driver license by providing an exam for candidate drivers who finished course from legal Drivers training centers. The exam is checked by computer system (network).as the manager of transport authority of Hawassa city tell us it takes 3sec to check their result. And after that every driver must renewed their license in every two years.
Our system focus on two offices Hawassa city road and transport authority and Hawassa city traffic police.

1.2. Problem of statement

The data in this system are kept manually especially after each driver is punished which makes it difficult to search that data after several months or years when a traffic police need it to know the profile of the driver, when the driver need it to know his/her profile, to renew his license, to pay his/her penalty and the data will be destroyed by disaster. It takes time to search large number of files.

In addition to this the existing system has several limitations it is time consuming to record and search files, Wastage of material, Duplication of paper, retrieve in bringing document to those who need it, no system for properly transferring or removing of inactive files which are occasional reference documents, Files get lost, Expensive, not efficient, Error prone, Difficulty to integrate data, Data redundancy and Difficulty of locating & finding files dependent each other.

1.3. Objective

1.3.1. General Objective

To change the manual system which exists in Hawassa city traffic control and management system into computerized System and develop a user friendly system that avoids the need for unnecessary physical interaction with the document of Hawassa city traffic control and management system.

1.3.2. Specific Objective

In order to attain the general objective, the following list of specific objectives is set:

- To make document analysis about traffic management and control system.
- To perform a requirement analysis to find out the system functional and nonfunctional requirements.
- To Design the system using object-oriented models for understanding the system and to make the implementation easy.
- To Design the database for storing data related to Driver, Traffic police and Vehicle using MySQL.
- To avoid data redundancy and to keep data secured
- To save resource
- To develop Mobile app for traffic police to easily manage and control driver license
- To ease data modification like adding, updating and editing user record
- To create better working environment
- To deploy the system

1.4. Scope

Although, design concepts, outputs and other components of the Project can be used for whole Ethiopia, the implementation of this project is on only for Hawassa city transport authority. The scope of the project mainly deals with providing information about traffic polices and drivers, accessing driver profile, register vehicle information.
The main users of the projects are traffic polices or Members and system Administrators. The scope of this new system development will include all the information that needed to handle any problem that may be face the system during processing data of the traffic polices and driver records by manually.

1.5. Limitation of the project

The Ethiopian transport authority system is very wide institution and the activities also wide. So by understanding the time to finish the project there is necessarily a limit to fulfill the required goal in specified time. The new system cannot include the following modules: the scope of our project is only for Hawassa city, it is only used for android phone, it doesn't support local languages and our system needs network connection. This system can't record detailed accident records.

1.6. Tools and Methodology

1.6.1. **Tools**

Since Traffic management and control system is a big system, we design an interactive web and mobile application using android that the authorized user can easily manage and get information from the application and interact with the system.

An interactive web application design requires the following

- •Web Server software
- •Server side scripting language
- •Database system
- •Client side programing language

For this specific project we will use

1. Programing and scripting language

❖ Server side scripting language (Back-end)
- PHP scripting language

❖ Client side programing language (Front-end)
- HTML5

- Bootstrap
- JQuery mobile
- CSS
- Java-script
- Java for Android

❖ Other

- Ajax-**A**synchronous **J**avaScript and **X**ML.
- JQuery - JavaScript Library
- JSON- **J**ava**S**cript **O**bject **N**otation

2. Software

- Microsoft Visio and Astah UML for system prototype design
- Microsoft word and power point
- WAMP (apache and MYSQL)
- Jet Brains PhpStorm
- Adobe Photoshop
- Android Studio

Why PHP?

There are several types of web programming language that are used for making a site more dynamic and interactive. But, for this project we are choosing PHP scripting to make our system more dynamic and interactive. PHP is a widely-used, free, fast and easy, cross platform and efficient alternative to competitors such as Microsoft's ASP.

Why MYSQL?

There are several reasons to use MySQL.

- It’s quick and powerful
- It’s improving all the time
- It’s free
- Handles large database.
- All columns have default values. You can use insert a subset of a table’s columns; those columns that are not explicitly given values are set to their default value.

Why Ajax?

- It is a technique for creating fast and dynamic web pages and web application.
- It allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.

Why JQuery?

- It is a lightweight, "write less, do more", and JavaScript library.

Why java?

- Java is self-interpreted programing language.

Hardware requirements

1. During Development and design
    a. PC
    b. Android Phone
2. During implementation
    a. Web server
    b. NAS
    c. Android Phone

1.6.2. **Methodology**

Data Collection

The method which enables us to capture information about requested system is called Methodology. The methods we employed in the data collection are site observation, interviews and document analysis that are references or indirectly related to Hawassa city traffic control management system.

- ✓ Site observation: actual observation of field officers (traffic police) at work.
- ✓ Open Interview: asking the employee of the organization and See Appendix 1
- ✓ Document analysis: reading the document available in the organization.

Design and Implementation Methodology

The team decides to use object –oriented methodology because it is Easier to maintain, understood by the group members, Unaffected to change, it is flexible in terms of using implementations, Ease of understanding object-oriented models due to a consistent underlying representation throughout the development process, Ease of modification and extensibility of object-oriented models and Ease of reuse of object components from previously designed systems.

1.7. Work Plan(Gant Chart)

Table 1: Work Plan

Activities	**Time**						
	Dec	**Jan**	**Feb**	**Mar**	**Apr**	**May**	**June**
Proposal Preparation	15 Days						
Requirement analysis		20 Days					
Design			25 Days				
Documentation Defense			1 Day				
Implementation				30 Days	30 Days	15 Days	
Evaluation						15 Days	
Final Defense							1 Day

Chapter Two

2. System Requirement Analysis

2.1. Existing system analysis

Currently Hawassa City Traffic management and control system is operated manually. This system has some drawbacks. Registering new certified Driver, updating his penalty record may lead to serious mistakes.

Some of the activities which are performed under the existing systems are:

- ✓ Accept new driver's information from training centers
- ✓ Examine them and give license cards
- ✓ Record traffic accident details
- ✓ Record and update Vehicle information

2.1.1. Problem of the Existing system

Manual processing such as storing, retrieving of data and information, the current system has the problems such as Performance problem, Information Problem, Data storage problem, Efficiency Problem and Security and control problem.

2.1.2. Performance problem

The current system is unable to perform tasks and activities with efficient and required time. It is tedious and not fast communication.

Information Problem

The existing system has many drawback concerning data security such as Lose of data may occur, due to manually collecting of data, there is a redundant record and inconsistency problems, Inaccurate data and information may produce, Incorrect information leads to poor decision making, Poor flow of information between City traffic police and Transport office as well as Higher Authorities like Zone and Regional Offices.

Data storage problem

Problems like Lack of a well-organized database system, Data are not easily accessible due to its integration and placed in different location, Difficult to update information, Data redundancy that leads to inconsistency, it is exposed to Disaster like, Fire, Flood, Volcanic eruption are main data storage problems.

Efficiency Problem and Security

The efficiency of the existing system is not optimal, because storing and locating data takes much more time and Redundancy flow of information.

Control problem

The current system can be accessed by unauthorized person, since it doesn't have any authentication and authorization system.

2.2. Overview of proposed system Analysis

Hawassa city traffic control and management system is a traffic control management Mobile app and web application that provide simple and effective way of controlling and managing traffic. The project "Hawassa city traffic control and management system" is developed in Java for Android, HTML5, JQuery, Ajax and PHP scripting language which mainly focuses on basic operations in a traffic management like registering new Driver, new Traffic Police officer, new vehicle, and updating new information and searching from database.

Hawassa city traffic control and management system is web and mobile application, designed to help drivers and traffic polices. Our software is easy to use for both beginners and advanced users. It provides user-friendly and attractive user interface, combined with strong searching, Insertion and reporting capabilities.

2.3. Requirements Analysis

User Requirements

User requirements are statements, in a natural language plus diagrams, of what services the system is expected to provide to the system users and the constraints under which it must operate. That describes user goals or tasks that the users must be able to perform with the system. User requirements therefore describe what the user will be able to do with the system.

- ✓ The user interface shall be menu driven, it shall provide friendly dialog boxes; drop down lists, radio buttons, and text boxes for user input.
- ✓ The navigation from one screen to the other must be easy.
- ✓ The bill officer wants to get real report according to the report type.
- ✓ Buttons and labels would be indicating exact function that it represented known, example submit represent save, add but not retrieve from database.
- ✓ Drivers, Vehicle and traffic police must fullfil the business rule to be registered.

Functional Requirement

Functional requirements are the description of the facility or feature required. Functional requirements deal with what the system should do or provide for users. They include description of the required functions, outlines of associated reports or online queries, and details of data to be held in the system.

In general, our system includes Web and mobile application. The mobile application that is phone based android application that enables the user to do activities like Punish Driver, register crime or car accident, control fake Driver license and View profile. The web application enables to create

Register Employee, Vehicle, Driver, Traffic police, view Employee profile, Vehicle profile, Driver profile, and Traffic police profile and generate report. The functional requirement of our system Actor include the following.

Administrator

REQ-1: The system shall require login before provide any function for administrator.

REQ-2: The system shall display an error message "Incorrect password or username" when administrator try to login with wrong password or username.

REQ-3: The system shall allow the administrator to control the overall activities in the system.

REQ-4: The system shall allow administrator to change his /her account information.

REQ-5: The system should allow the administrator to create user account for the system user.

REQ-6: The system shall Inquiry all current enrolled/registered users to view their details except password.

REQ-7: The system shall remove wrong entries from the system.

REQ-8: the system shall view reports in different operations in the system.

REQ-9: The system shall give permission categorically, also enabling or disabling of user's permission can be set.

REQ-10: The system shall give the administrator to enable and disable users of the system.

REQ-11: The system shall ensure that the information entered is of the correct format.

REQ-12: The system should generate report.

Employee

REQ-1: The system shall require login before provide any function for any staff user.

REQ-2: The system shall display an error message "Incorrect password or username" when staff user tries to login with wrong password or username.

REQ-3: The system shall allow the staff user to register the drivers (here the driver must pass the exam provided by the organization).
REQ-4: The system shall allow the stuff user to change his/her account information Example password, profile picture.
REQ-5: The system shall import vehicle information from vehicle database when required.
REQ-7: The system shall allow uploading notice from the staff members/users.
REQ-8: The system shall provide the calendar to the staff members'.

REQ-9: The system should allow staff user to search for any Driver and vehicle detailed information.

Drivers:

REQ-1: The system shall authenticate before accessing system.
REQ-2: The system shall allow drivers to change his /her account password.
REQ-3: The system shall allow drivers to view his/her detail information or records.

Vehicle:

REQ-1: The system shall require login before allowing manipulation of vehicle information.
REQ-2: The system shall allow the authorized user to modify vehicle information example vehicle type, level, owner.
REQ-3: The system shall allow search for any vehicle detailed information.
REQ-4: The system shall generate annual reports of the vehicle.

Traffic police:

REQ-1: The system shall require login before provide any function for Traffic police.
REQ-2: The system shall display an error message "Incorrect password or username" when Traffic police try to login with wrong password or username.
REQ-3: The system shall allow Traffic police to change his /her account information except user name.
REQ-4: The system should allow Traffic police to view drivers detail information.
REQ-5: The system shall calculate number of records of the Driver for any Traffic police.
REQ-4: The system shall allow Traffic police to punish the driver and update Driver details to the database.
REQ-7: The system shall search for any driver detailed information to the Traffic police.
REQ-8: The system shall approve Driver License and vehicle plate number.

Traffic police Administrator(TPA)

REQ-1: The system shall require login before provide any function for TPA.

REQ-2: The system shall display an error message "Incorrect password or username" when TPA try to login with wrong password or username.

REQ-3: The system shall allow the TPA to control the overall activities of the Traffic police.

REQ-4: The system shall allow TPA to change his /her account information.

REQ-5: The system should allow the TPA to create user account for the Traffic Police.

REQ-6: The system shall Inquiry all current enrolled/registered Traffic police to view their details except password.

REQ-7: The system shall give permission categorically, also enabling or disabling of traffic Police permission can be set.

Non-Functional Requirements

Non-functional requirements are requirements that are not directly concerned with the specific services delivered by the system to its users. They may relate to emergent system properties such as reliability, response time, and store occupancy. Alternatively, they may define constraints on the system implementation such as the capabilities of I/O devices or the data representations used in interfaces with other systems. Non-functional requirements, such as performance, security, or availability, usually specify or constrain characteristics of the system as a whole. Such as

Usability

- ✓ The system provides a help and support menu in all interfaces for the user to interact with the system.
- ✓ The user can use the system (app) by reading help and support.
- ✓ The user can use the system (app) by looking descriptive icons

Security

- ✓ The system provides username and password to prevent the system from unauthorized access.
- ✓ The user's password must be greater than eight characters including letters, numbers and special characters.
- ✓ Traffic Polices can be registered by the authorized staff admin only
- ✓ Drivers registered by staff admins

Performance

- ✓ The system response time for every instruction conducted by the user must not exceed more than a minimum of few minutes and seconds.
- ✓ The system works at high performance rate when executing user's input and should be able to provide response within a short time span for highly complicated tasks.
- ✓ The system uses Advanced Languages to retain good performance like Ajax

Availability

- ✓ The system should always be available on web and mobile phone app for access at 24 hours, 7 days a week. Also in the occurrence of any major system malfunctioning, the system should be available in hours.
- ✓ The system should be available on the internet always to allow administrators and other staff to keep in touch everywhere

Chapter Three

3. System Modeling: Object Oriented Design

3.1. System use case

A system use case model is composed of a use case diagram and the accompanying documentation describing the use cases, actors, and associations. System use cases reflect analysis decisions and, arguably, even design decisions. A use case describes a sequence of actions that provide a measurable value to an actor and is drawn as a horizontal ellipse. An actor is a person, organization, or external system that plays a role in one or more interactions with your system. Actors are drawn as stick figures. Associations between actors and classes are indicated in use case diagrams, a relationship exists whenever an actor is involved with an interaction described by a use case. Associations are modeled as lines connecting use case and actors to one another.

3.1.1. **Use case Diagram**

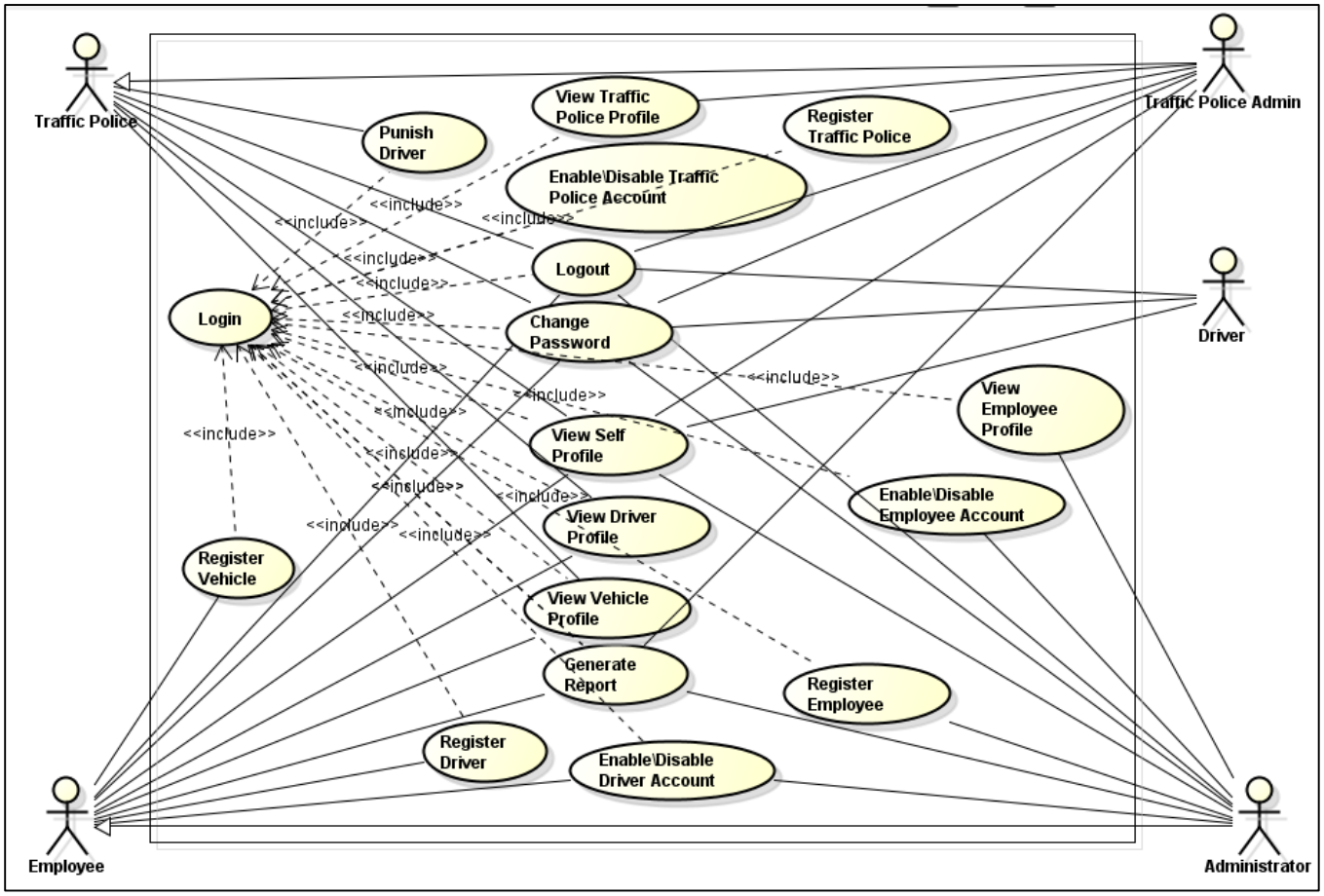


Figure 1: Use case Diagram

3.1.2. Use Case Description

Login

Use-case Number	UC-01	
Use-Case Name	Log in	
Priority	High	
Actor	Administrator, Employee, TPA, Traffic police, Driver	
Description	This use case describes how each stakeholder login into the System.	
Precondition	None	
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	**User Action**	**System Response**
	1. The Users are try to login to the system. 2. The User enter user name and password and click login button.	3.The system verifies the password and username. 4. if the username and password are valid the system redirects to the main page else 4.1 5 use case exit.
Alternate course of Action	4.1 The system displays error message and redirect to the login page.	

Table 2:Log in - use case description

Generate Report

Use-Case Number	UC-02	
Use-Case Name	Generate Report	
Priority	High	
Actor	Administrator, TPA, Employee	
Description	These use case allow the Administrator, TPA and Employee to generate report to the necessary activities	
Precondition	Administrator, TPA and Employee should have LAN connection, and they should login to the system.	
Post Condition	If this use case ends successfully, the system generates monthly or weekly Report Information	
Basic Course of Action	**User Action**	**System Response**
	1. The users(Actor) log in to the system 2. User clicks the generate report menu item. 3. The user clicks on report type menu item 5. The user fills the form 6. The user click on generate report button 9.End use case	4. The system display report form that contain the following: - ✓ Date of report ✓ Type of report ✓ For whom the report is prepared

		7. The system will check or validate the filled form elements 8. If the form elements are filled correctly, the system generate the report to the selected user else got to#8.1.
Alternate course of action	8.1 If the filled data are incorrect the system displays error message and point were the error is occurred.	

Table 3:Generate Report Use Case Description

Enable Account

Use-case Number	UC-03	
Use-Case Name	Enable Account	
Priority	High	
Actor	Administrator and TPA	
Description	This use case describes how Administrator enable employee account and TPA enable traffic police's account. When Employee/Traffic police Resign job, their account should be disabled and when they return to their job their account will be Enabled by Administrator/TPA.	
Precondition	Employee/traffic police return to their job	
Post-condition	If the use case was successful, traffic police/Employee can access to their account	
Basic course of Action	**User Action**	**System Response**
	1. Admin/TPA log in to the system click 'search 'menu item under Manage Account menu 3. Admin/ TPA fills Employee id/Traffic police id. 5. Admin/TPA click 'Enable user' button.	2. The system displays search Employee/Traffic police form field. 4. The system displays Employee/Traffic police profile if found else go to #4.1. 6. The system Deactivates user's account and display 'successful' message
Alternate course of Action	4.1 The system Displays 'no such Employee/Traffic police found' message.	

Table 4:Enable Account Use Case Description

Disable Account

Use-case Number	UC-04
Use-Case Name	Disable Account
Priority	High
Actor	Administrator and TPA

Description	This use case describes how Administrator enable employee account and TPA enable traffic police's account. When Employee/Traffic police Resign job, their account should be disabled.	
Precondition	Employee and traffic police is not working currently	
Post-condition	If the use case was successful, traffic police/Employee can't access to their account	
Basic course of Action	**User Action**	**System Response**
	1. Admin/TPA log in to the system and click 'search 'menu item under 'Manage Account' menu 3. Admin/ TPA fills Employee id/Traffic police id. 5. Admin/TPA click 'Disable user' button.	2. The system displays search Employee/Traffic police form field. 4. The system displays Employee/Traffic police profile if found else go to #4.1. 6. The system Deactivates user's account and display 'successful' message
Alternate course of Action	4.1 The system Displays 'no such Employee/Traffic police found' message.	

Table 5: Disable Account Use Case Description

View profile(self)

Use-case Number	UC-05	
Use-Case Name	View profile(self-profile)	
Priority	High	
Actor	Admin, employee, traffic police, driver and TPA	
Description	This use case describes how users view their profile	
Precondition	User logged to the system	
Post-condition	If the use case was successful, the user can view their own profile.	
Basic course of Action	**User Action**	**System Response**
	1. The user log in to the system 2.The user click 'profile' menu item 4. Users click 'close' button to return to homepage 6.use case end	3. The system displays their own profile 5. The system displays homepage
Alternate course of Action		

Table 6:View Own Profile Use Case Description

Change Password

Use-case Number	UC-06
Use-Case Name	Change Password
Priority	High
Actor	Administrator, Employee, Traffic Police, Driver and TPA

Description	This use case describes how each User change their password	
Precondition	Each User logged to the system	
Post-condition	If the use case was successful, the User will have new password	
Basic course of Action	**User Action**	**System Response**
	1. Users try to log to the system with valid user name and password. 3.User click 'change password' menu item under Manage Account menu 5. User fills the form components and Click 'Apply' button 8. End of use case	2. The system Displays Homepage. 4. The system Displays change password form that contain the following: - ✓ old password ✓ New password ✓ Confirm password ✓ hint 6. System checks old password and password policy 7. If old password is correct and new password fulfill password policy, then replace old password with new password in the database and Displays 'Successful' message and go to Homepage Else go to 7.1
Alternate course of Action	7.1. If the new password doesn't fulfil the password policy or the old password is incorrect then the system displays Error message and go to 4	

Table 7: Change Password Use Case Description

Register Traffic Police

Use-case Number	UC-07	
Use-Case Name	Register Traffic police	
Priority	High	
Actor	TPA	
Description	This use case describes how TPA register traffic police	
Precondition	Traffic police wants to registered	
Post-condition	If the use case was successful, Traffic police gets user name and unique id	
Basic course of Action	**User Action**	**System Response**
	1.TPA log in to the system and click 'Register Traffic police' menu item under Register Menu 3.TPA fills the form and click Register button 7. use case end	2. The system displays Register Traffic police form that contain the following: - ✓ Full name ✓ Title ✓ Address ✓ Age ✓ Sex ✓ Educational status

		4. The system checks or validate the form fields. 5. if the form fields are correctly filled the system register Traffic police and save data to the data base else go to use case 5.1 6. The system displays 'successfully register Traffic police' message.
Alternate course of Action	5.1 if the form fields are not correctly filled, the system display error message and TPA fills the form again.	

Table 8: Register Traffic Police Use Case Description

***Register* Driver**

Use-case Number	UC-08	
Use-Case Name	Register Driver	
Priority	High	
Actor	Employee	
Description	This use case describes how Employee Register Driver	
Precondition	Driver must pass both field and paper and computer exam. In addition, he/she must able to bring his/her grade 10 certificate.	
Post-condition	If the use case was successful, Driver gets his/her driver license and new user id and password in order to use his new mobile app.	
Basic course of Action	**User Action**	**System Response**
	1.Employee log in to the system and click 'Register Driver' menu item 3.Employee fills the form and click Register button 7. use case end	2. The system displays Register Driver form that contain the following: - ✓ Full name ✓ sex ✓ birth region ✓ birth date ✓ Driver license number ✓ Level ✓ Home Town and city ✓ Wereda ✓ Kebele ✓ House number 4. The system checks or validate the form fields. 5. if the form fields are correctly filled then the system register Driver and save the data to the data base else go to use case 5.1 6. The system displays 'successful register Driver' message.
Alternate course of Action	5.1 if the form fields are not correctly filled, the system display error message and go to #3.	

Table 9: Register Driver Use case Description

***Register* Vehicle**

Use-case Number	UC-9	
Use-Case Name	Register Vehicle	
Priority	High	
Actor	Employee	
Description	This use case describes how Employee Register Vehicle	
Precondition	Employee should login to the system	
Post-condition	If the use case was successful, the Vehicle gets plate number	
Basic course of Action	**User Action**	**System Response**
	1.Employee log in to the system and click 'Register Vehicle' menu item 3.Employee fills the form and click Register button 7. use case end	2. The system displays Register Vehicle form that contain tow components 1. Vehicle description • Vehicle type • Made in • Model • Chassis number • Motor number • Fuel type • Weight • Plate Number 2. Owner description • Name • Sex • Country • Region • Wereda • Phone 4. The system checks or validate the form fields. 5. if the form fields are correctly filled then the system register Vehicle and save the data to the data base else go to use case 5.1 6. The system displays 'successful register Vehicle message.
Alternate course of Action	5.1 if the form fields are not correctly filled, the system display error message and go to #3.	

Table 10: Register Vehicle Use case Description

***Register* Employee**

Use-case Number	UC-10
Use-Case Name	Register Employee
Priority	High

Actor	Administrator	
Description	This use case describes how Administrator Register Employee	
Precondition	Employee must pass the exam provided by the organization	
Post-condition	If the use case was successful, Employee registered and gets his/her new user id and password	
Basic course of Action	**User Action**	**System Response**
	1.Administrator log in to the system and click 'Register Employee 'menu item 3.Administrator fills the form and click Register Employee button 7. use case end	2. The system displays Register Driver form that contain the following: - ✓ Full name ✓ Role ✓ sex ✓ birth region ✓ birth date ✓ experience year in number ✓ Educational status ✓ Home Town and city ✓ Wereda ✓ Kebele ✓ House number 4. The system checks or validate the form fields. 5. if the form fields are correctly filled then the system register Employee and save the data to the Employee data base else go to use case 5.1 6. The system displays 'successful register Employee message.
Alternate course of Action	5.1 if the form fields are not correctly filled, the system display error message and go to #3.	

Table 11: Register Employee Use case Description

View *Employee* profile

Use-case Number	UC-11	
Use-Case Name	View Employee profile	
Priority	Medium	
Actor	Administrator	
Description	This use case describes how Administrator View Employee profile	
Precondition	Administrator logged to the system	
Post-condition	If the use case was successful, the Administrator can view his/her Employee profile.	
	User Action	**System Response**

Basic course of Action	1. The Administrator log in to the system 2.The Administrator click 'Employee profile' menu item 4. Administrator select Employee and click 'View profile' button 6. The Administrator click 'close' button to return to the home page. 8.use case end	3. The system displays list of Employee 5. The system displays Employee profile 7. the system redirects to the home page.
Alternate course of Action		

Table 12: View Employee Profile Use case Description

View Driver Profile

Use-case Number	UC-12	
Use-Case Name	View Driver profile	
Priority	High	
Actor	Administrator, Employee, TPA, Traffic police	
Description	This use case describes how the Authorized View Driver profile	
Precondition	Authorized user logged to the system	
Post-condition	If the use case was successful, the Authorized user can view driver profile.	
Basic course of Action	**User Action**	**System Response**
	1. The user log in to the system 2.The user click 'Driver profile' menu item 4. user(Actor) select Driver and click 'View profile' button 6. The user click 'close' button to return to the home page. 8.use case end	3. The system displays list of Driver's 5. The system displays Driver profile 7. the system redirects to the home page.
Alternate course of Action		

Table 13: View Driver Profile Use case Description

View Traffic Police Profile

Use-case Number	UC-13
Use-Case Name	View Traffic police profile
Priority	Medium
Actor	TPA
Description	This use case describes how TPA view traffic police profile
Precondition	TPA logged to the system
Post-condition	If the use case was successful, TPA can view driver profile.

Basic course of Action	**User Action**	**System Response**
	1. TPA log in to the system 2.TPA click 'Traffic police profile' menu item 4. TPA select Traffic police row and click 'View profile' button 6. The user click 'close' button to return to the home page. 8.use case end	3. The system displays list of Traffic police 5. The system displays Traffic police profile 7. the system redirects to the home page.
Alternate course of Action		

Table 14: View Traffic Police Profile Use case Description

View Vehicle Profile

Use-case Number	UC-14	
Use-Case Name	View vehicle profile	
Priority	High	
Actor	TPA, Administrator, Employee	
Description	This use case describes how Authorized user view Vehicle profile	
Precondition	user logged to the system	
Post-condition	If the use case was successful, user can view vehicle profile.	
Basic course of Action	**User Action**	**System Response**
	1. authorized user log in to the system 2.Authorized user click 'Vehicle profile 'menu item 4. authorized user select Vehicle row and click 'View profile' button 6. The Authorized user click 'close' button to return to the home page. 8.use case end	3. The system displays list of vehicle' 5. The system displays Vehicle profile 7. the system redirects to the home page.
Alternate course of Action		

Table 15: View Vehicle information Use case Description

Punish Driver

Use-case Number	UC-15
Use-Case Name	Punish Driver
Priority	High
Actor	Traffic Police
Description	This use case describes how Traffic Police Punish Driver

Precondition	Driver violate the rule and Traffic Police must be logged in to the system and have internet connection	
Post-condition	If the use case was successful, Each Driver who violate the rule punished by the Traffic police.	
Basic course of Action	**User Action**	**System Response**
	1. Traffic Police log in to the system 3.Traffic police browse driver profile and tap on punish button 5. Traffic Police fill form element and click on Punish Driver Button 8.use case end	2. The system Display Home page. 4. The system Display Punish Driver form that contain the following: - • Vehicle plate number • Driver name • Vehicle owner name • From------To------ • Type • Accident type • Date • Time • Place • Payment amount 6. The system Inspect form element 7.If Form element are correct, the system saves the data to the database and display successful message else go to #7.1
Alternate course of Action	7.1 The system displays error message and go back to #5	

Table 16: Punish Driver Use case Description

Logout

Use-Case Number	UC-16	
Use-Case Name	Log out	
Priority	High	
Actor	Administrator, staff, TPA, traffic police and driver	
Description	These use case allow Administrator, staff, TPA, traffic police and driver log out from the system at a time of accomplishing their work.	
Precondition	UC-1	
Post Condition	The System save the data into the session or the database then logout from the system.	
Basic Course of Action	**User Action**	**System Response**
	1. The users(Actor) try to log out from the system. 2. The users(Actor) clicks the log out button	3. The system saves the data and responds to the requested action. 4. The system displays a message that the Administrator, staff, TPA, traffic police and driver logged out from the system and redirects to the login page.

Table 17: Logout Use case Description

3.2. SequenceDiagram

Sequence diagrams are used to model the logic of usage scenarios. A usage scenario is exactly what its name indicates- the description of a potential way your system is used. A usage scenario is exactly what its name indicates- the description of a potential way your system is used. UML sequence diagrams model the flow of logic within your system in a visual manner, enabling you both to document and validate your logic, and are commonly used for both analysis and design purposes. Sequence diagrams are the most popular UML artifact for dynamic modeling, which focuses on identifying the behavior within your system. Objects, classes, and actors are depicted in sequence diagram. Sequence diagrams are typically used to model:

1. **Usage scenarios**. A usage scenario is a description of a potential way your system is used. The logic of a usage scenario may be part of a use case, perhaps an alternate course. It may also be one entire pass through a use case, such as the logic described by the basic course of action or a portion of the basic course of action, plus one or more alternate scenarios. The logic of a usage scenario may also be a pass through the logic contained in several use cases. For example, a student enrolls in the university, and then immediately enrolls in three seminars.

2. **The logic of methods**. Sequence diagrams can be used to explore the logic of a complex operation, function, or procedure. One way to think of sequence diagrams, particularly highly detailed diagrams, is as visual object code.

3. **The logic of services**. A service is effectively a high-level method, often one that can be invoked by a wide variety of clients.

1. LogIn

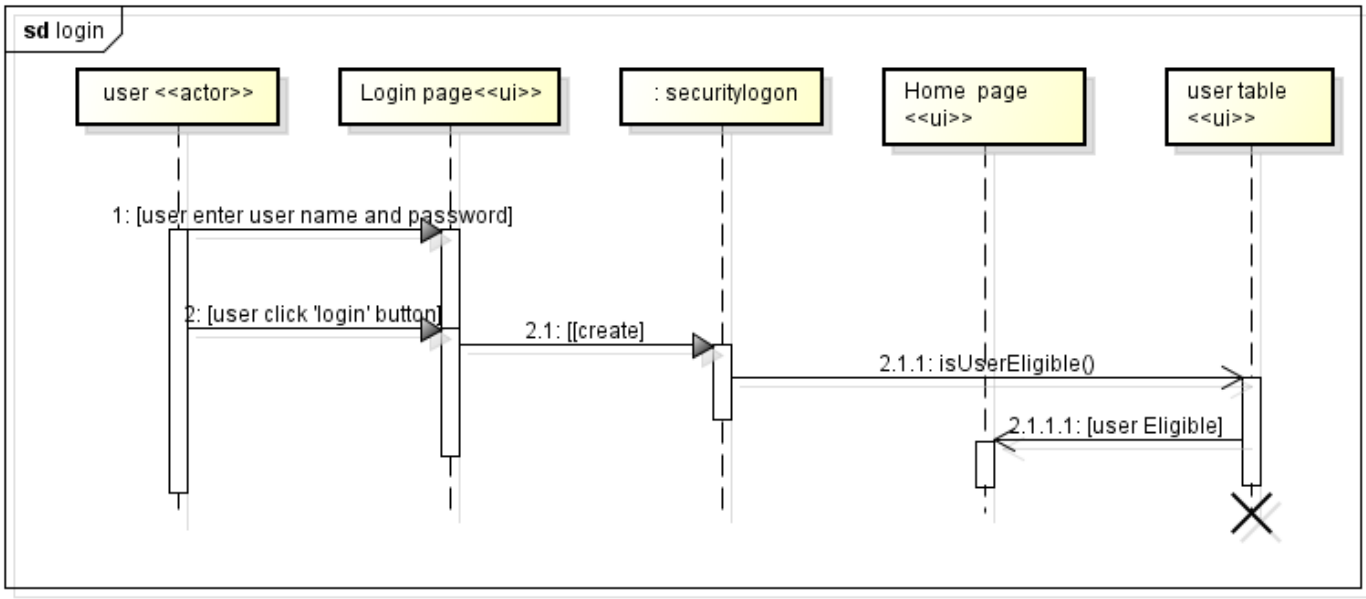


Figure 2: Login - Sequence Diagram

Generate Report

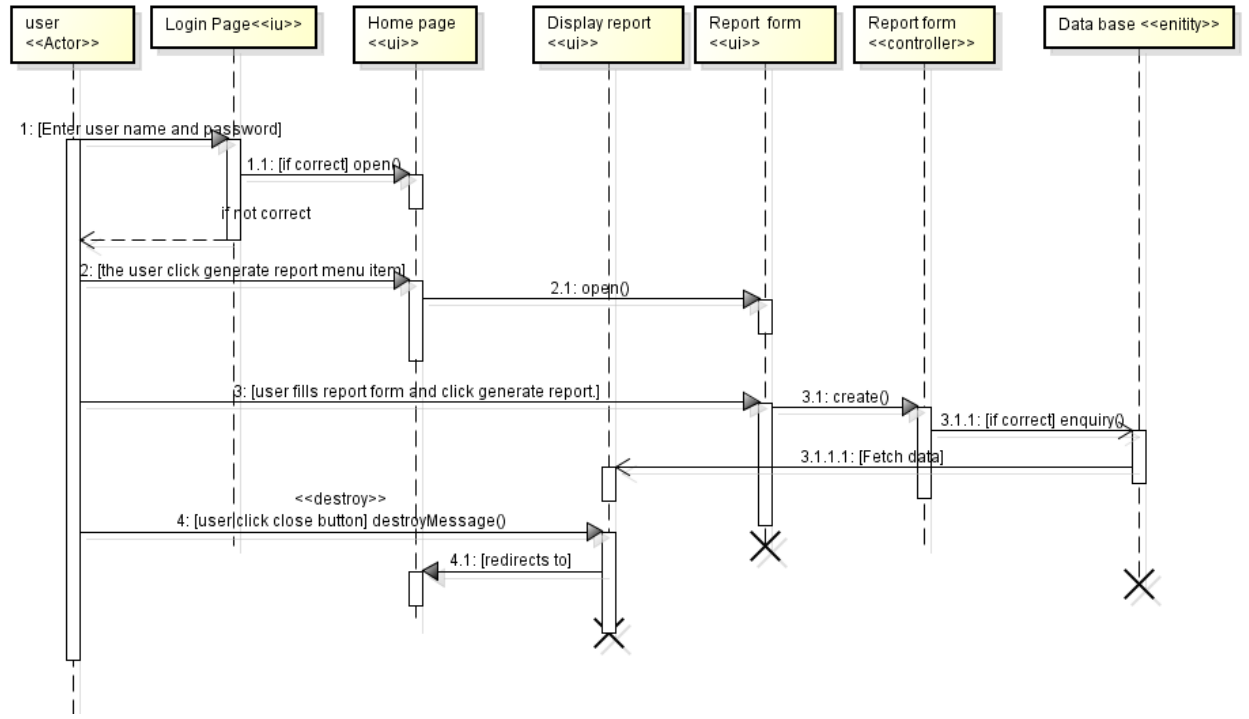


Figure 3: Generate Report - Sequence Diagram

Enable Account

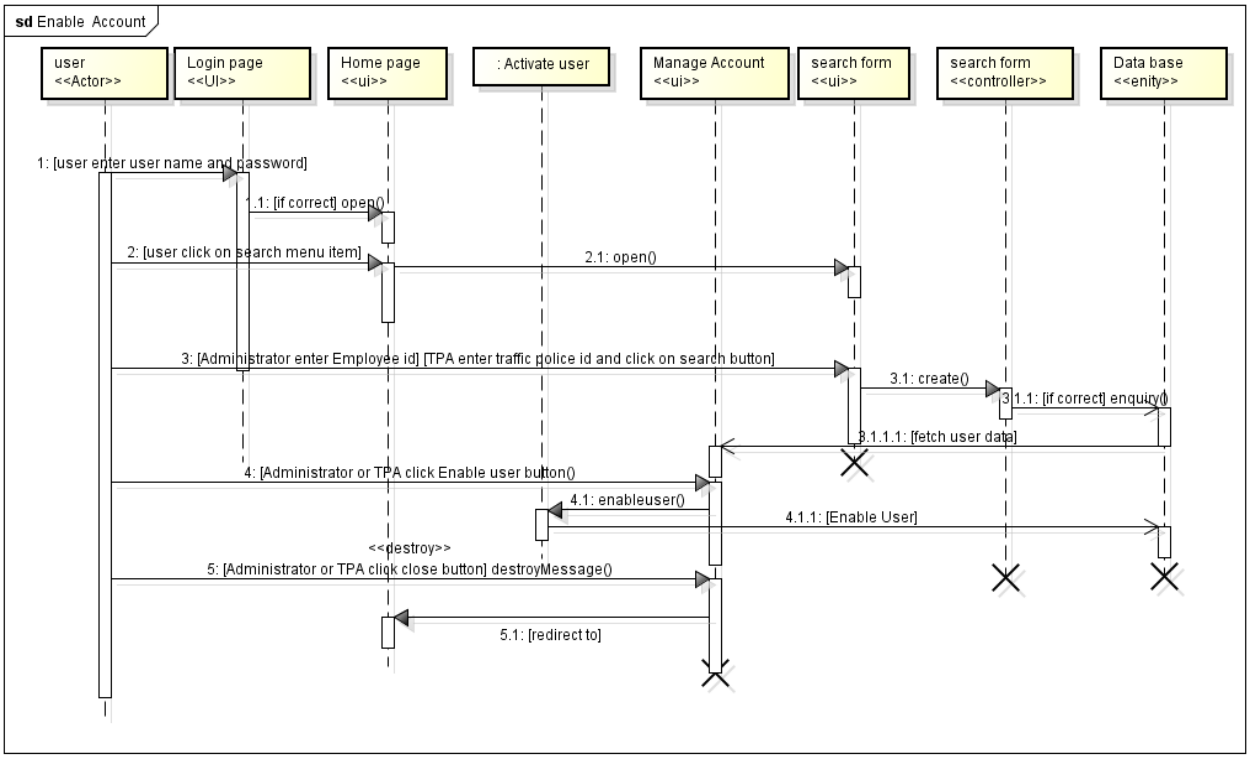


Figure 4: Enable Account - Sequence Diagram

Disable Account

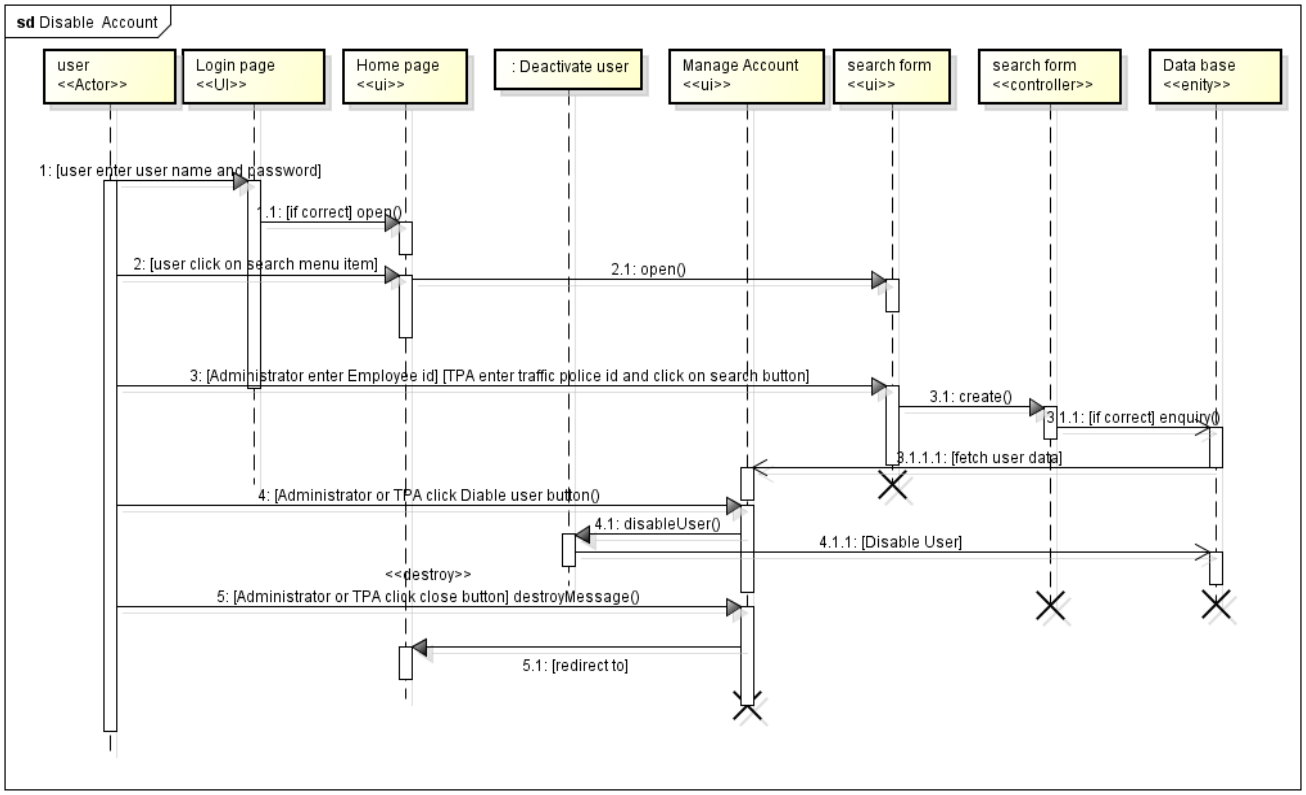


Figure 5: Disable Account - Sequence Diagram

View Self Profile

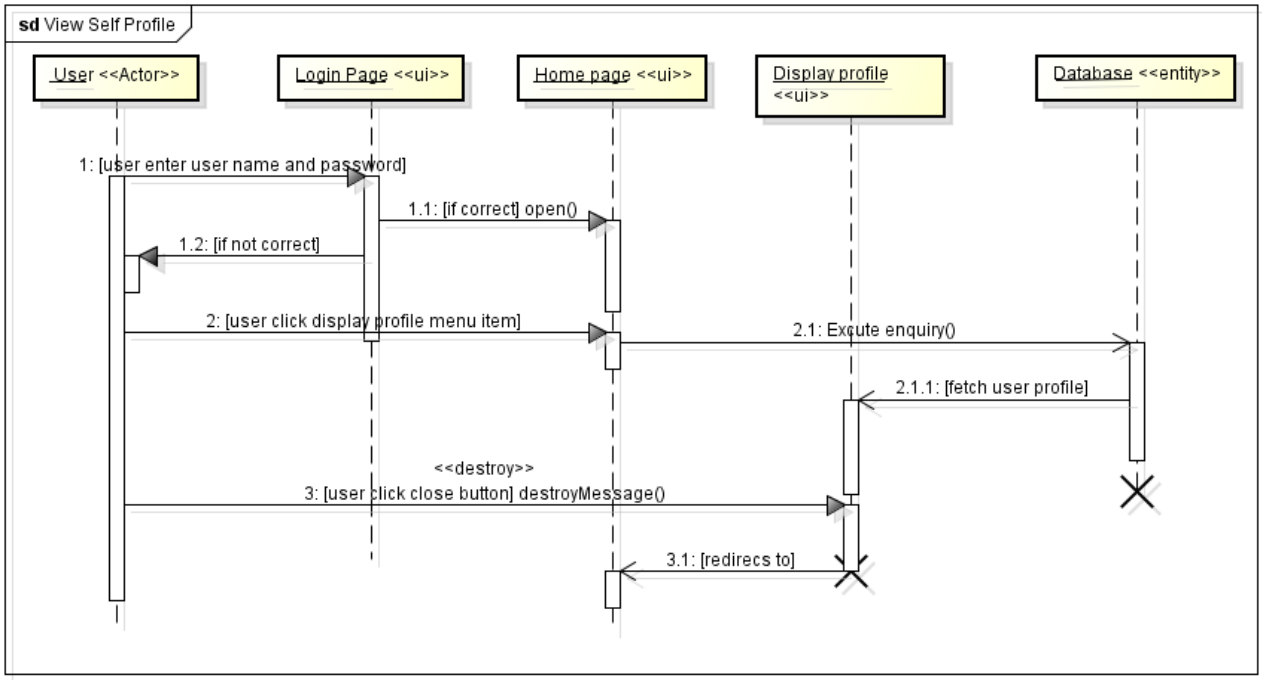


Figure 6: View Self Profile - Sequence Diagram

Change Password

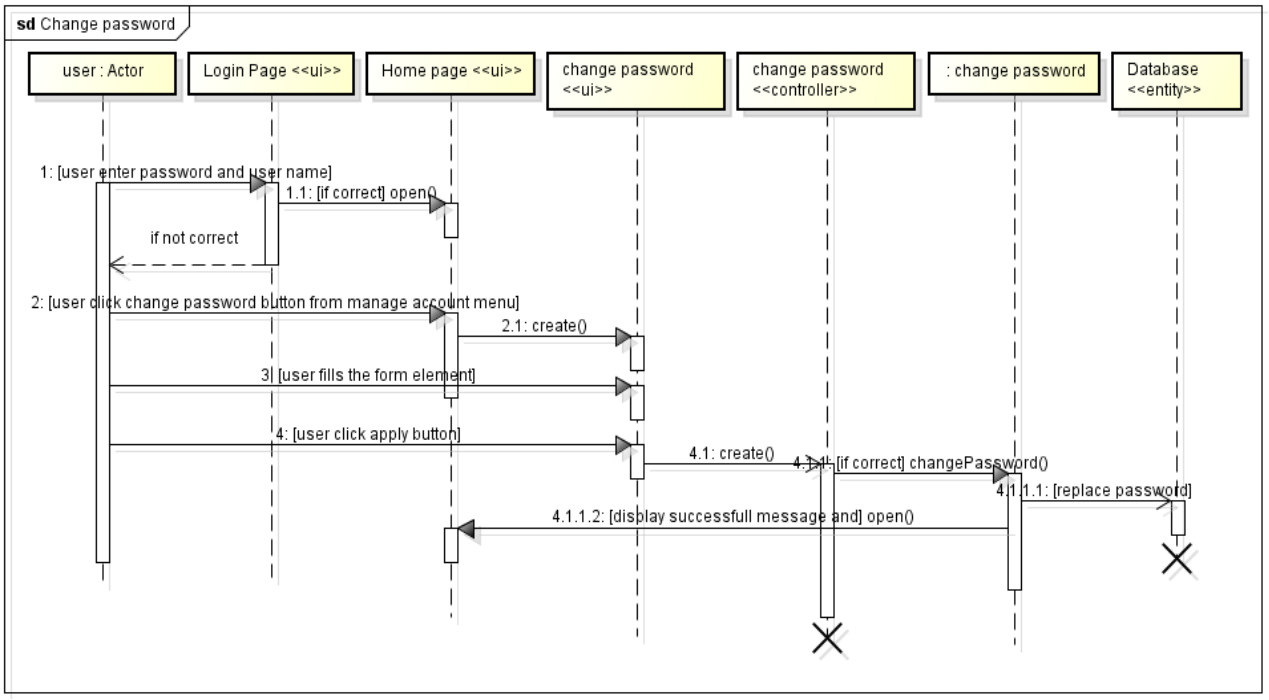


Figure 7: Change Password - Sequence Diagram

Register Traffic Police

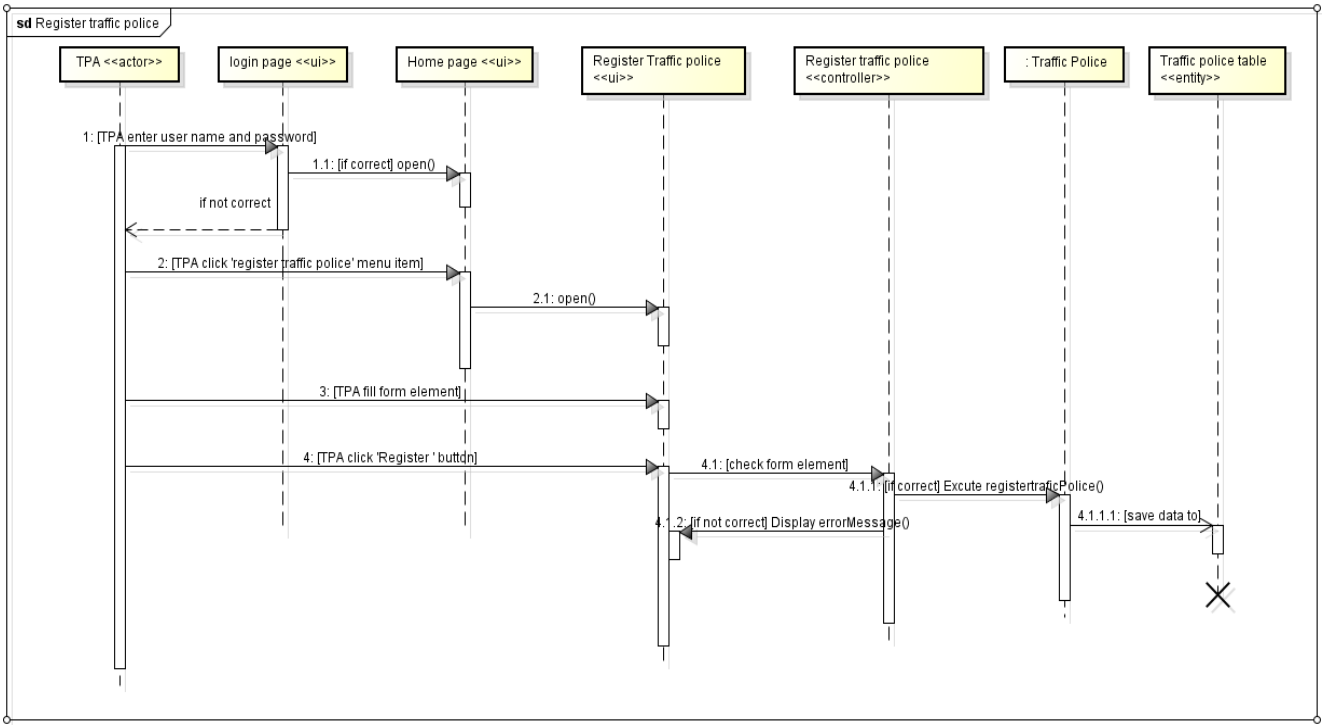


Figure 8: Register Traffic Police - Sequence Diagram

Register Driver

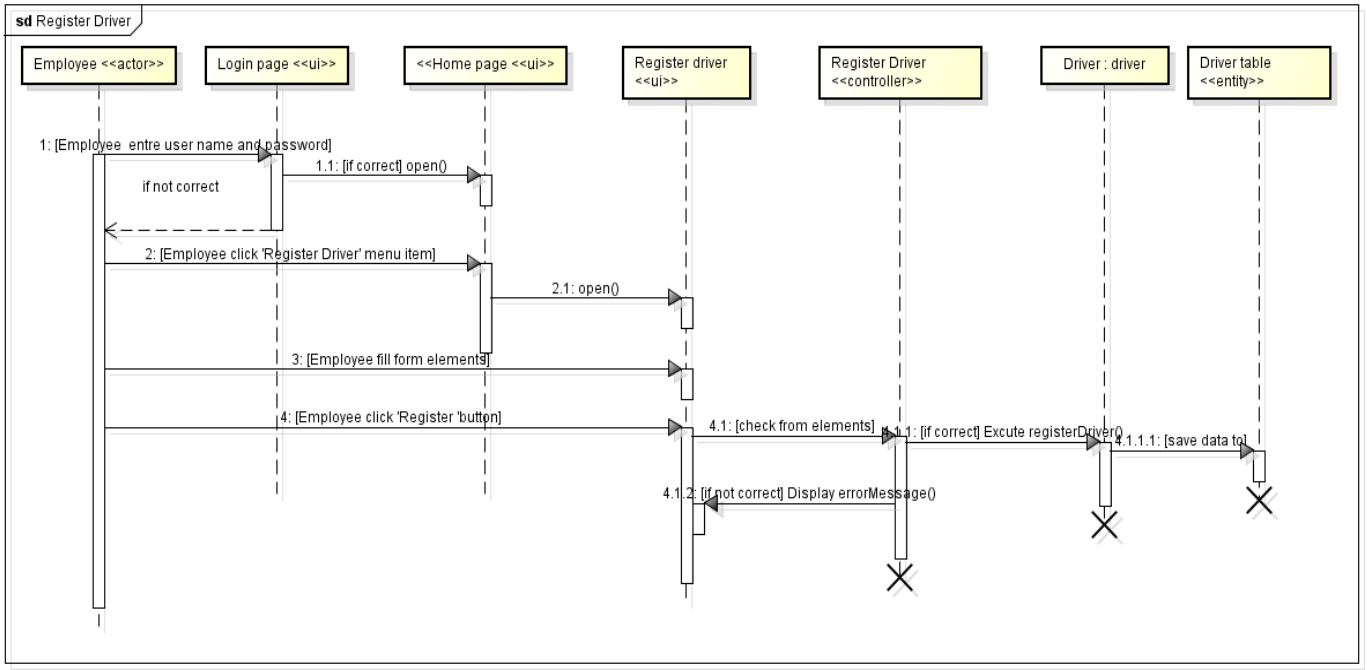


Figure 9: Register Driver - Sequence Diagram

Register Vehicle

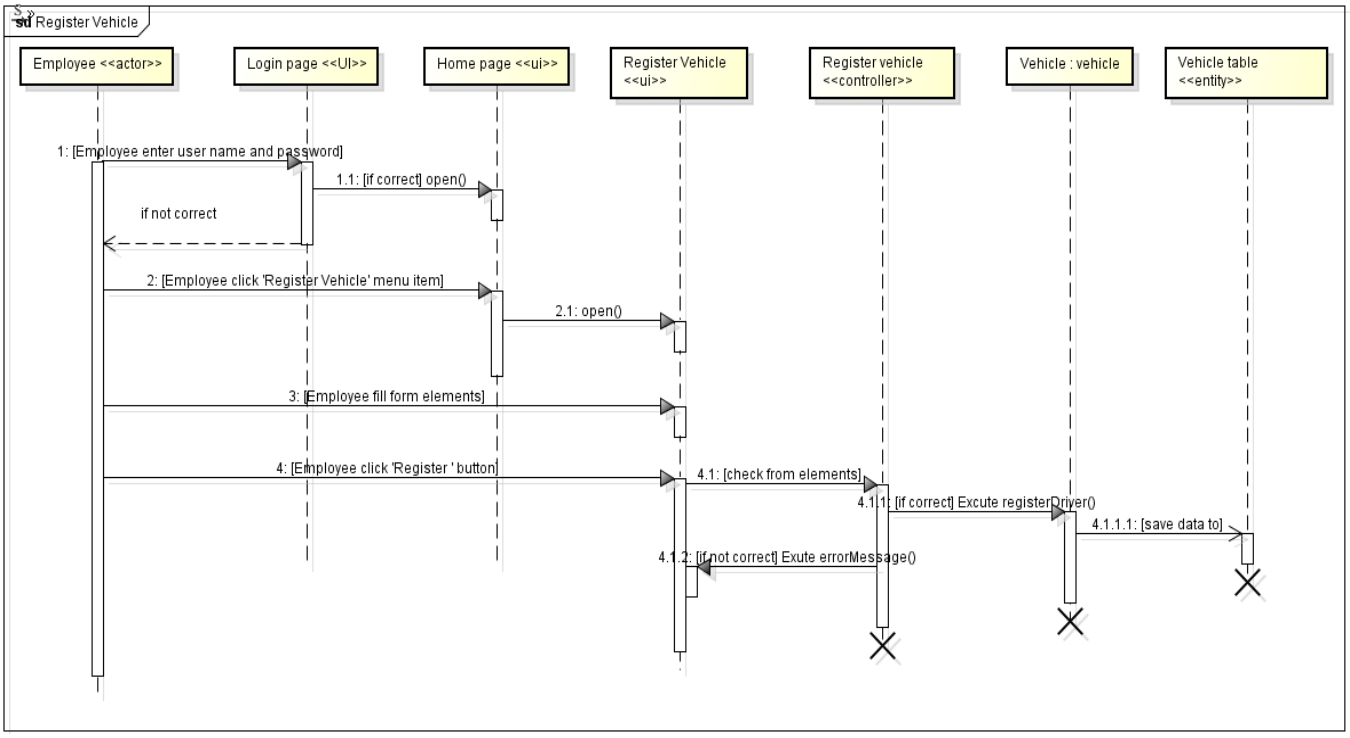


Figure 10: Register Vehicle - Sequence Diagram

Register Employee

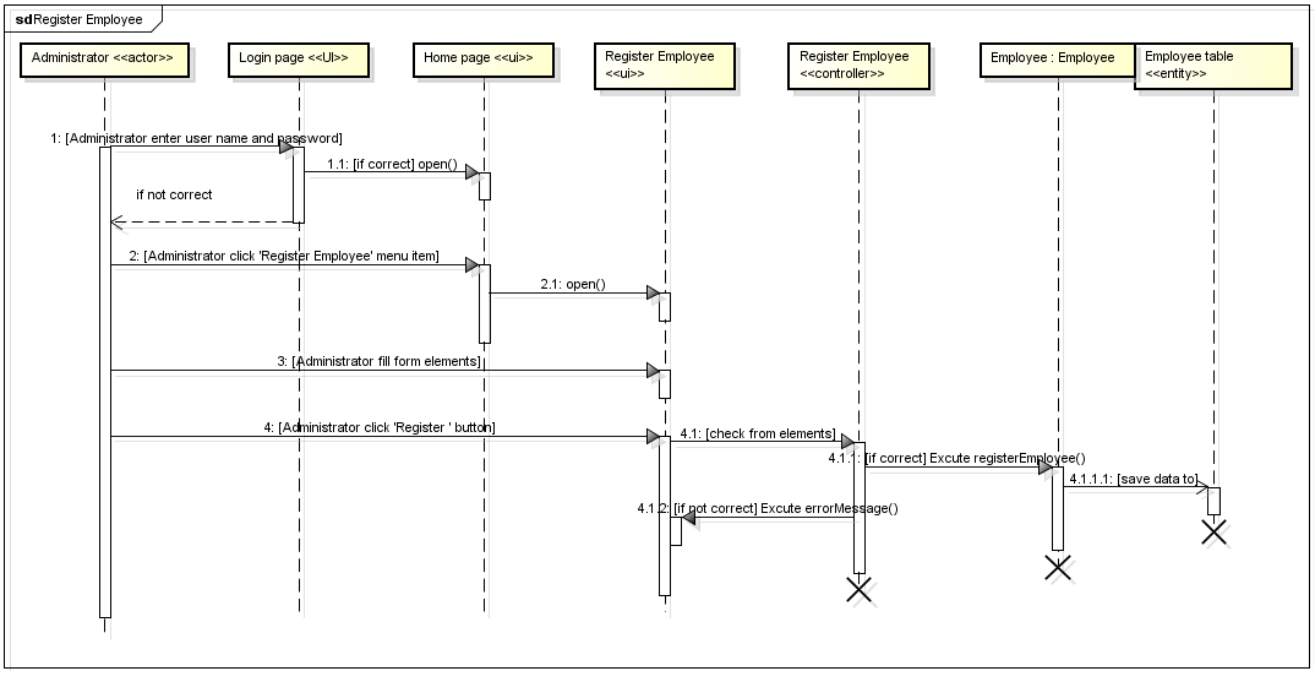


Figure 11: Register Employee

View Employee Profile

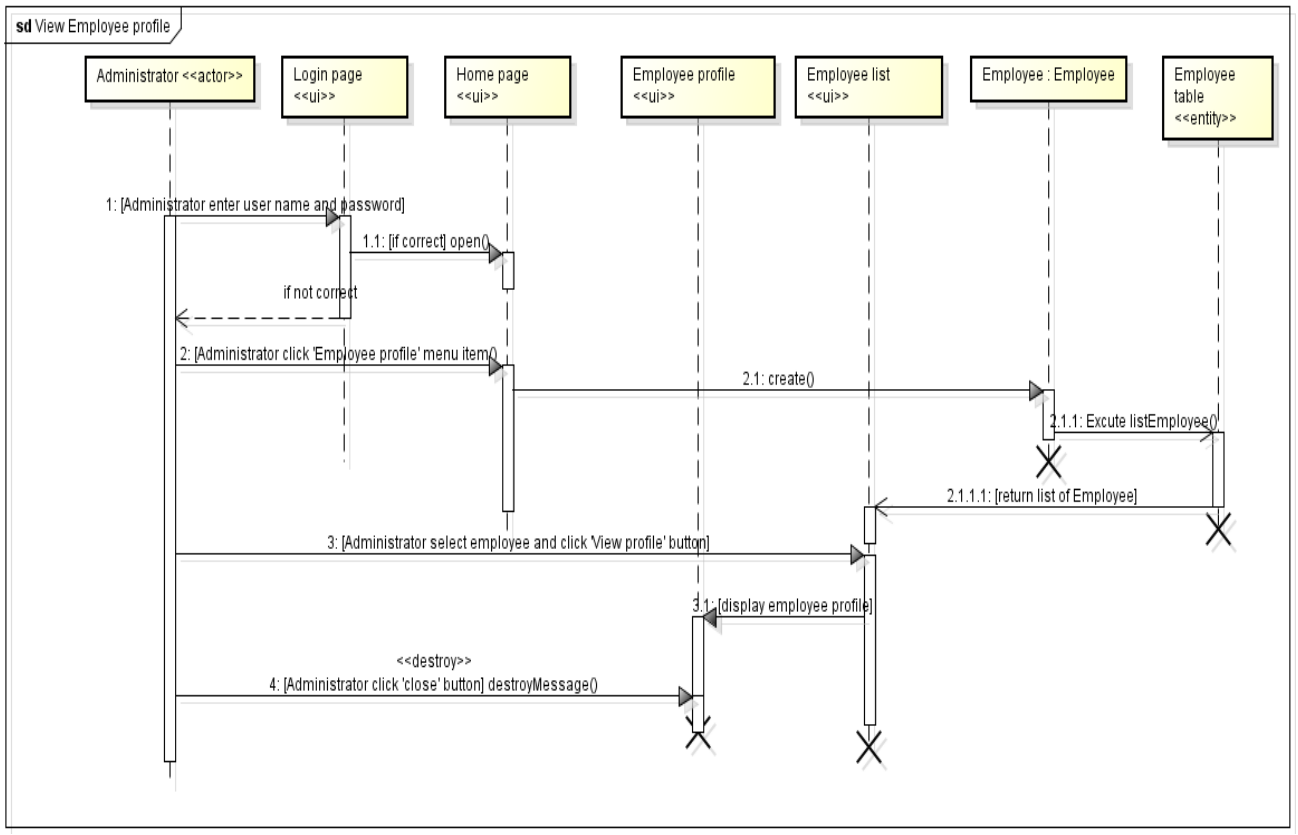


Figure 12: View Employee Profile - Sequence Diagram

View Driver Profile

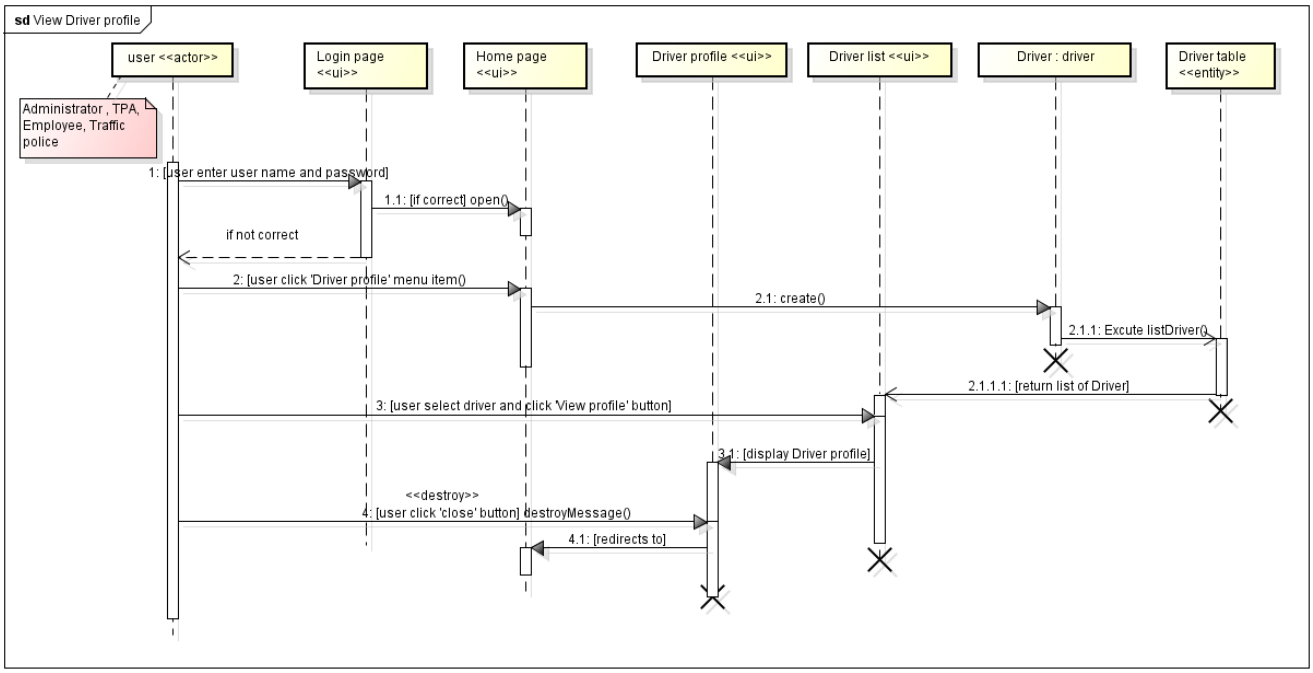


Figure 13: View Driver Profile - Sequence Diagram

View Traffic Police Profile

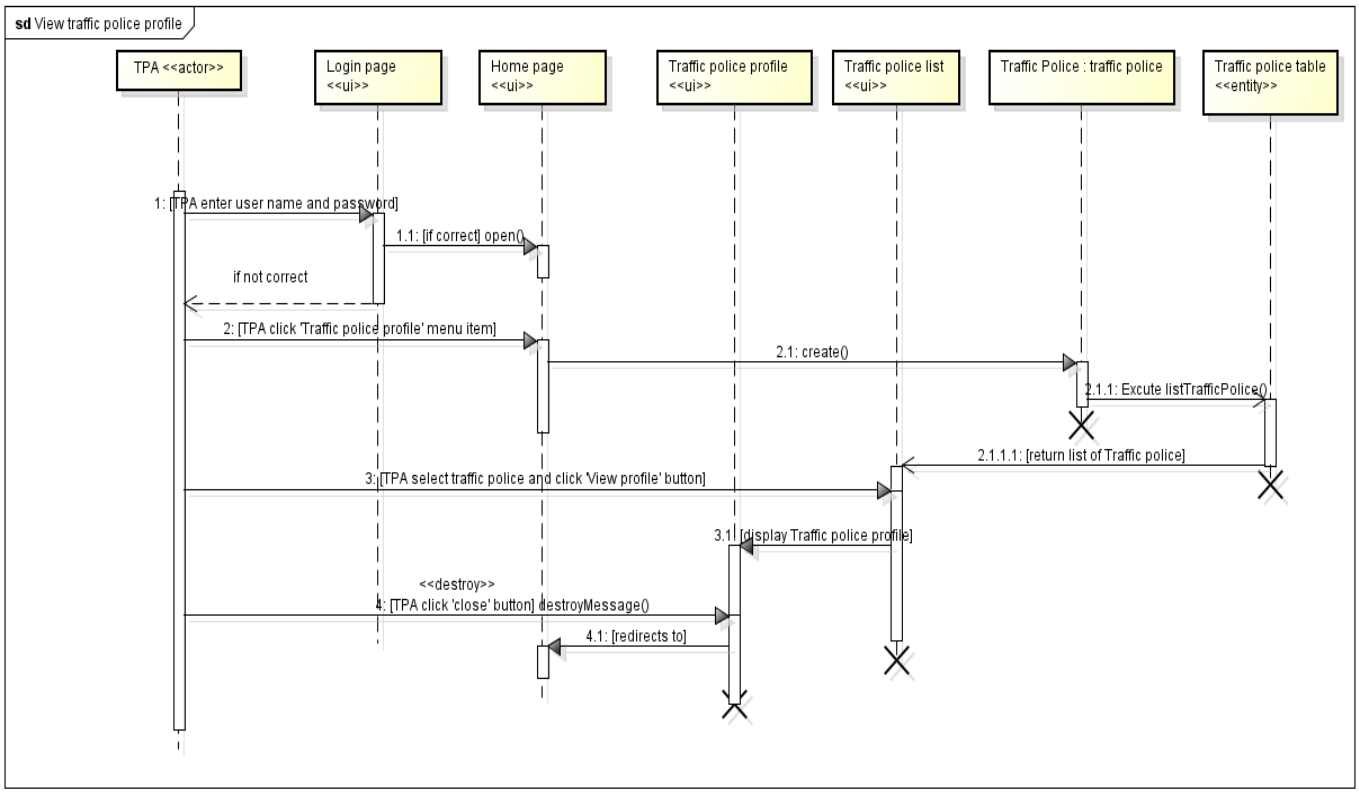


Figure 14: View Traffic Police Profile

View Vehicle Profile

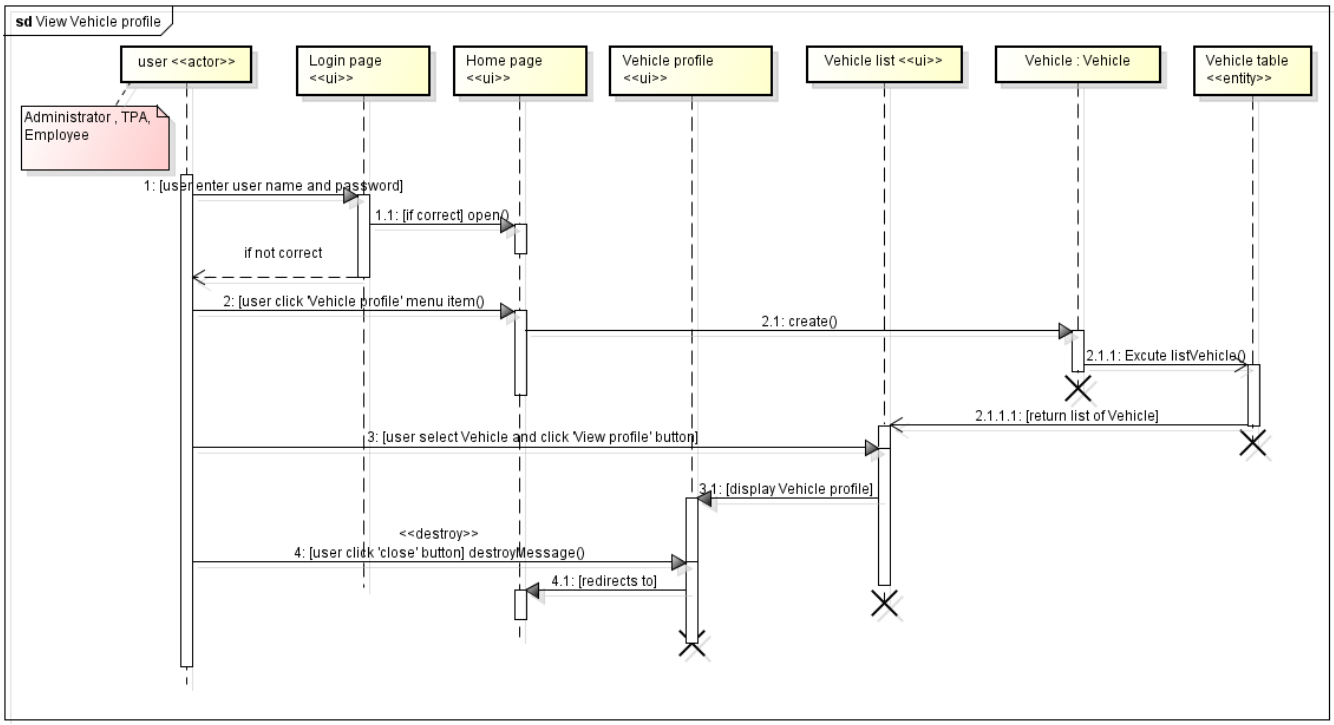


Figure 15: View Vehicle profile- Sequence Diagram

Punish Driver

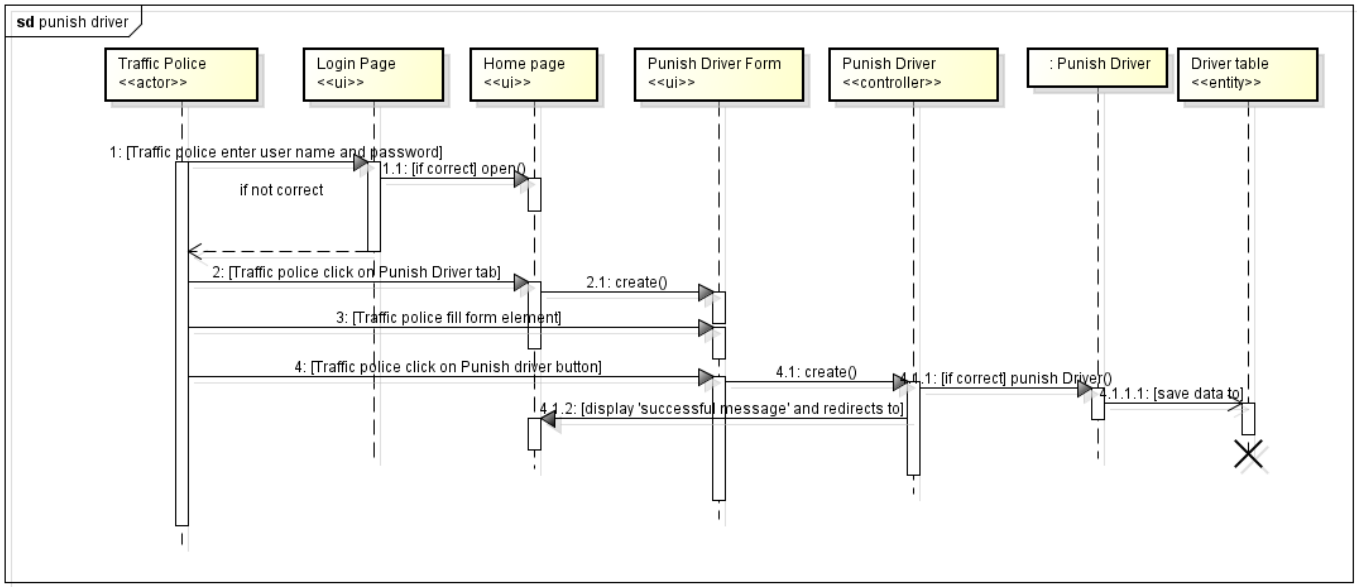


Figure 16: Punish Driver - Sequence Diagram

3.3. Activity Diagrams

Activity diagrams are used to model the logic of a business process, use case, or method. They are typically used for business process modeling, for modeling the logic captured by a single use case or usage scenario, or for modeling the detailed logic of a business rule. In many ways UML activity diagrams are the object-oriented equivalent of flow charts and data flow diagrams (DFDs) from structured development.

Login

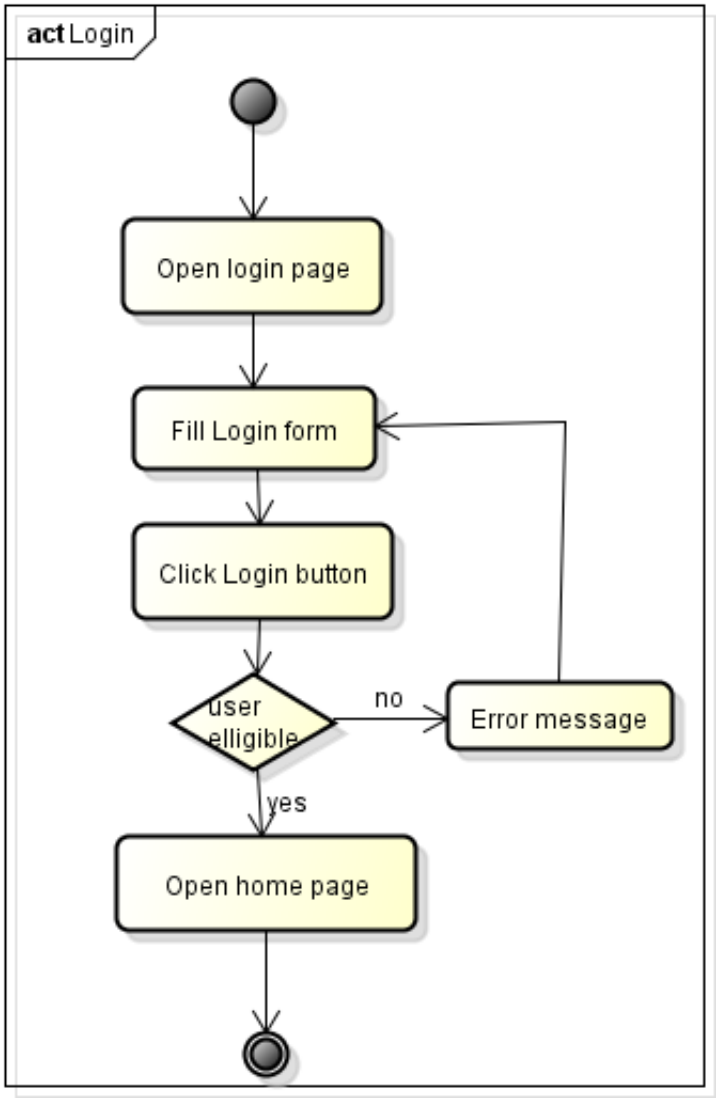


Figure 17: Log in - Activity Diagram

Logout

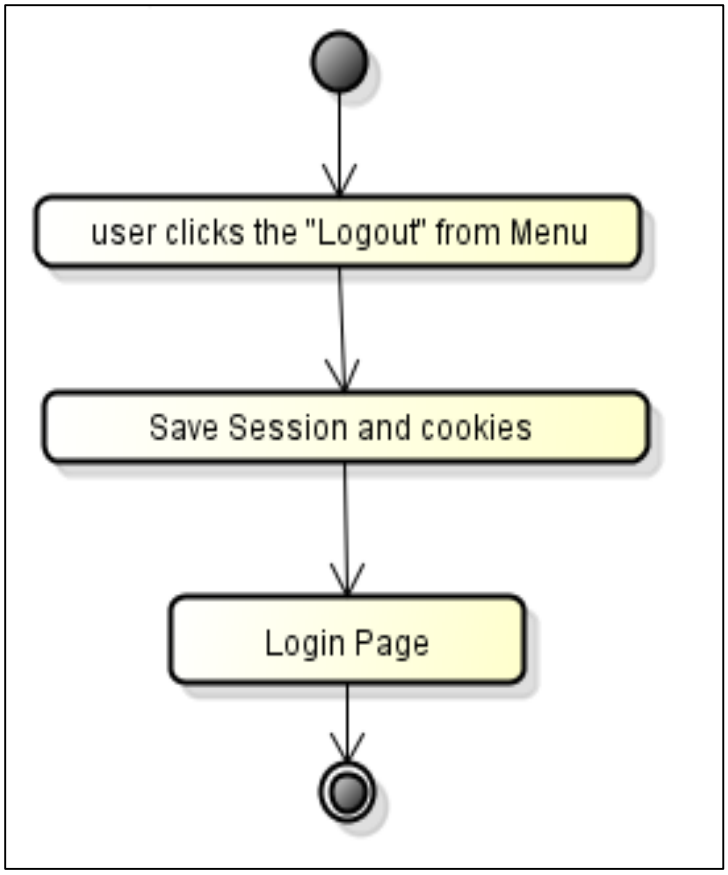


Figure 18: - Logout - Activity Diagram

Generate Report

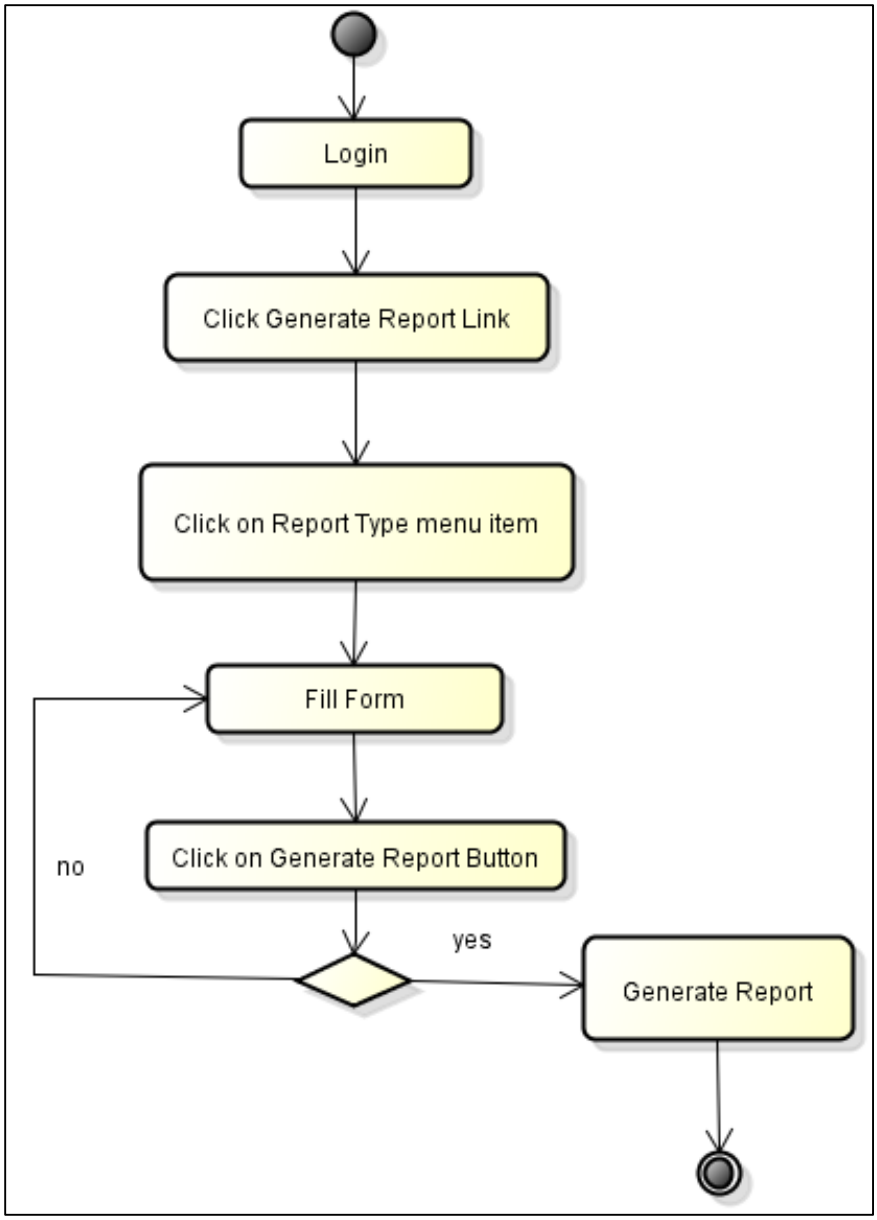


Figure 19: Generate Report - Activity Diagram

Enable/Disable user

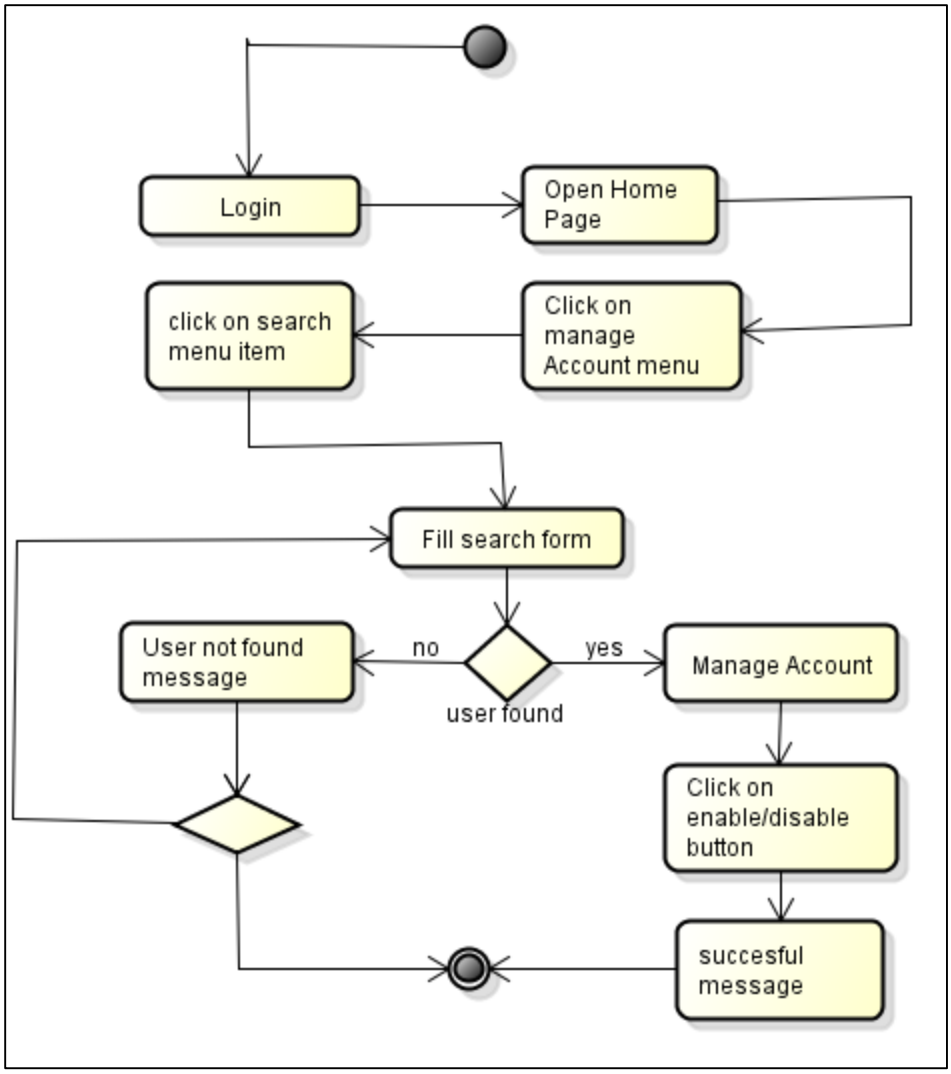


Figure 20: Enable/Disable user - Activity Diagram

View self-profile

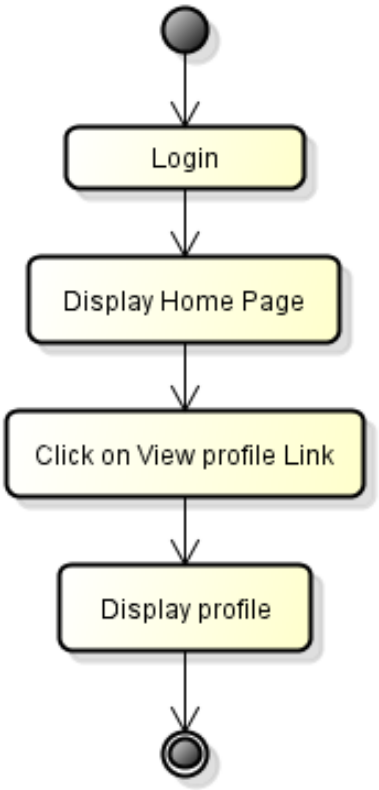


Figure 21: View self-profile - Activity Diagram

Change password

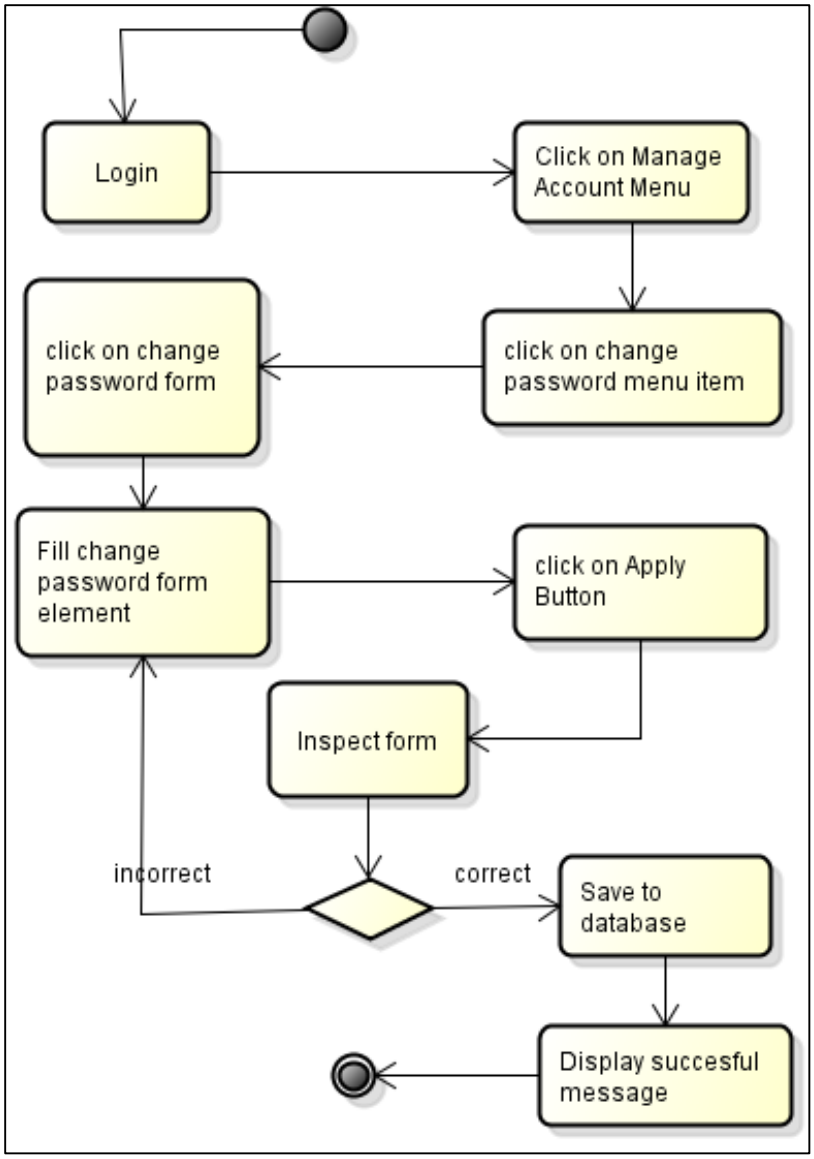


Figure 22: Change password- Activity Diagram

Register Traffic Police

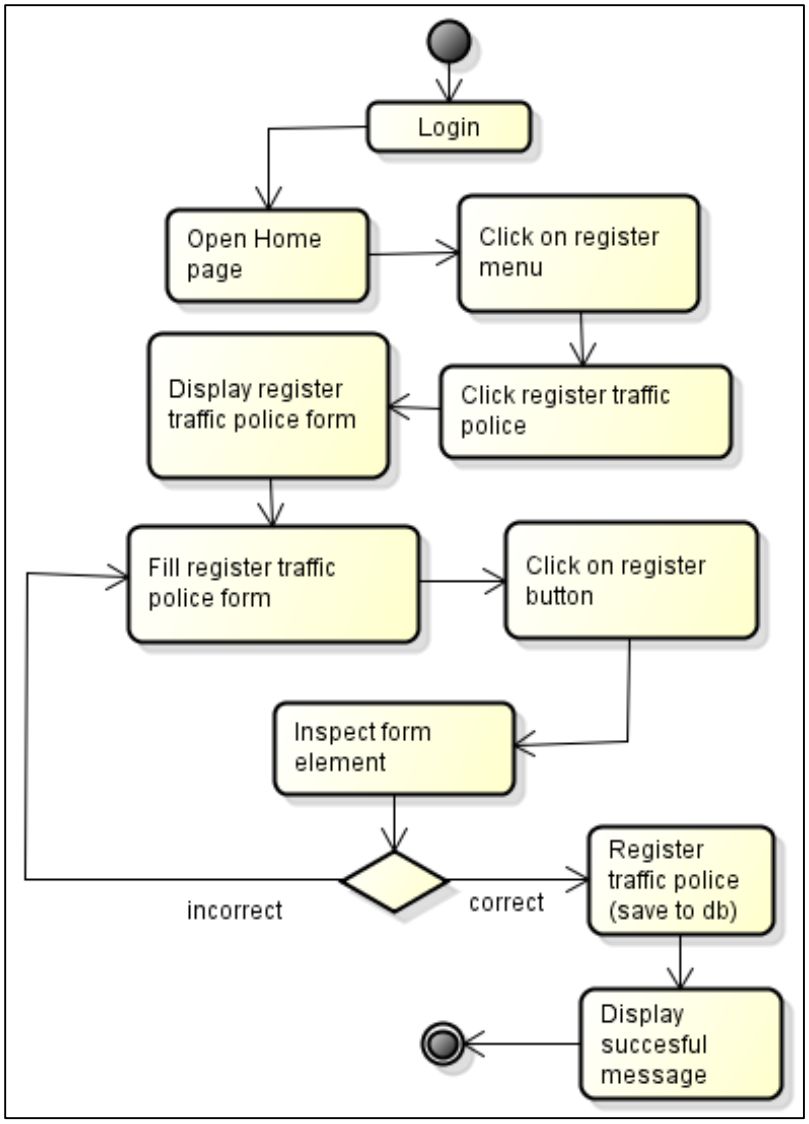


Figure 23: Register Traffic police - Activity Diagram

Register Vehicle

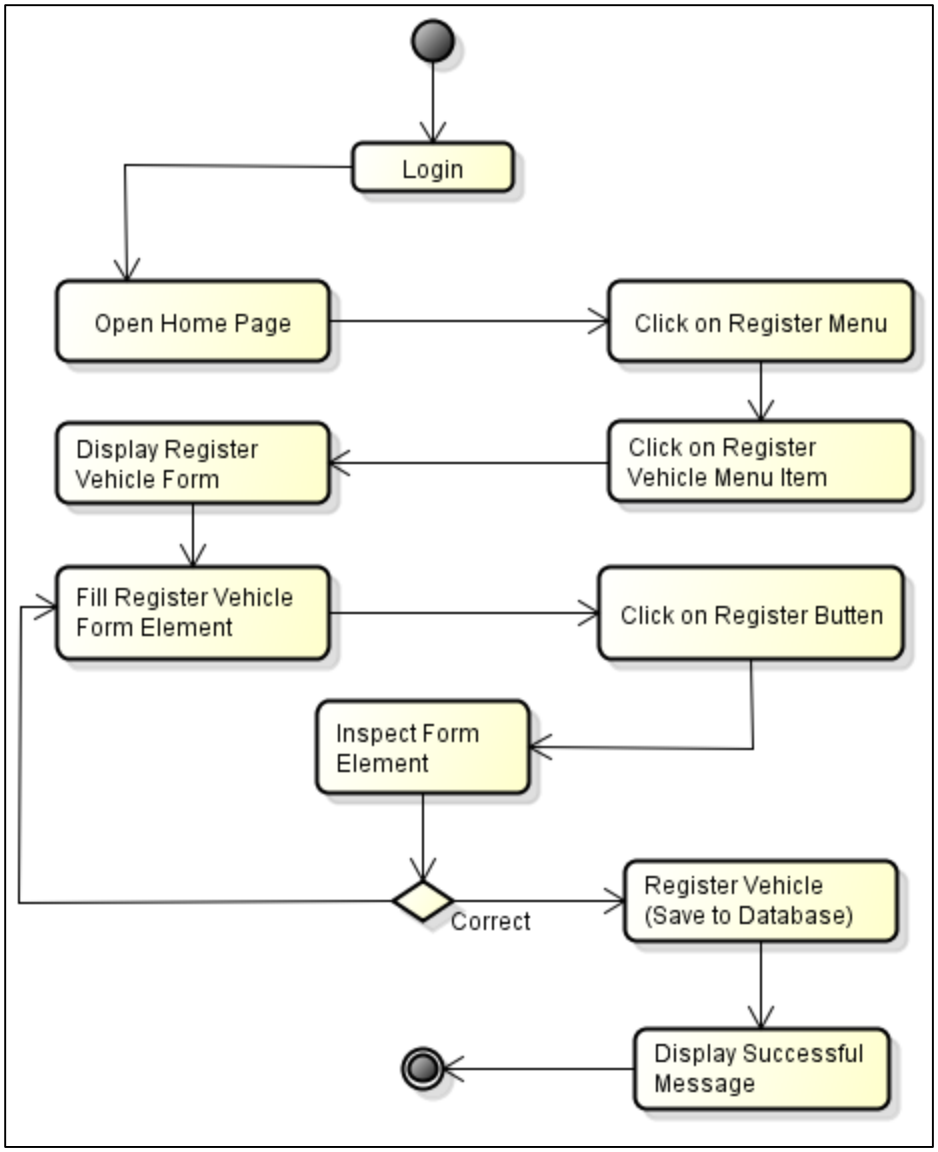


Figure 24: Register Vehicle - Activity Diagram

View Employee Profile

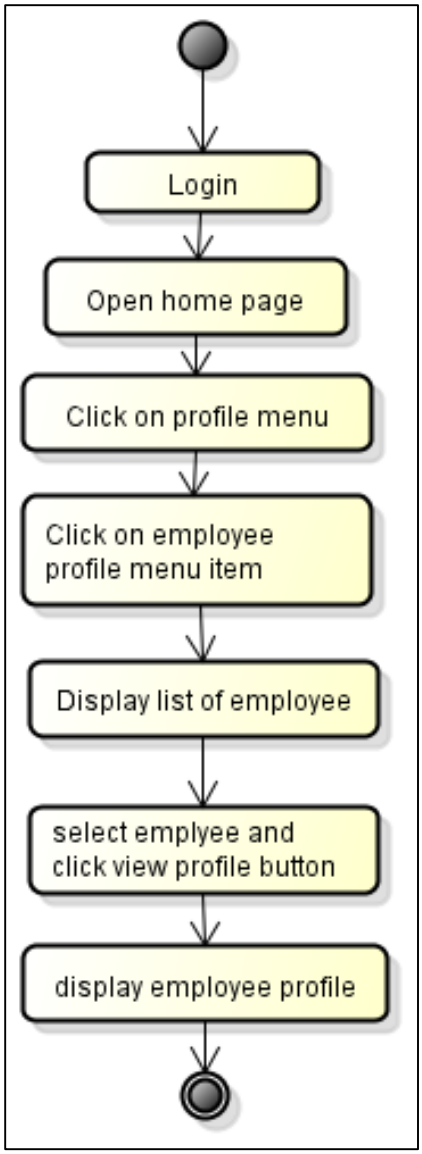


Figure 25: View Employee Profile - Activity Diagram

View Driver Profile

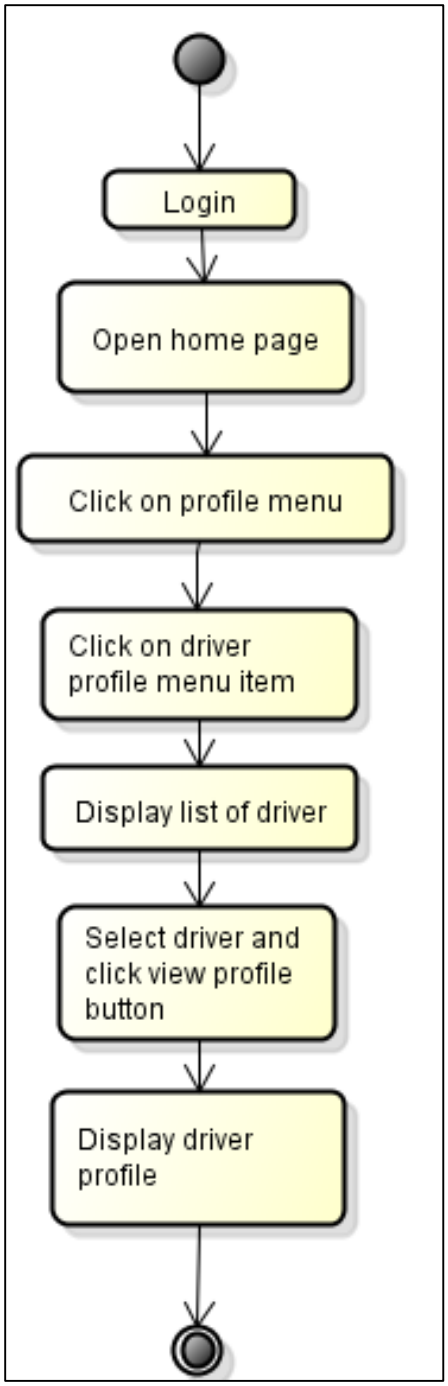


Figure 26: View Driver Profile - Activity Diagram

View Traffic police profile

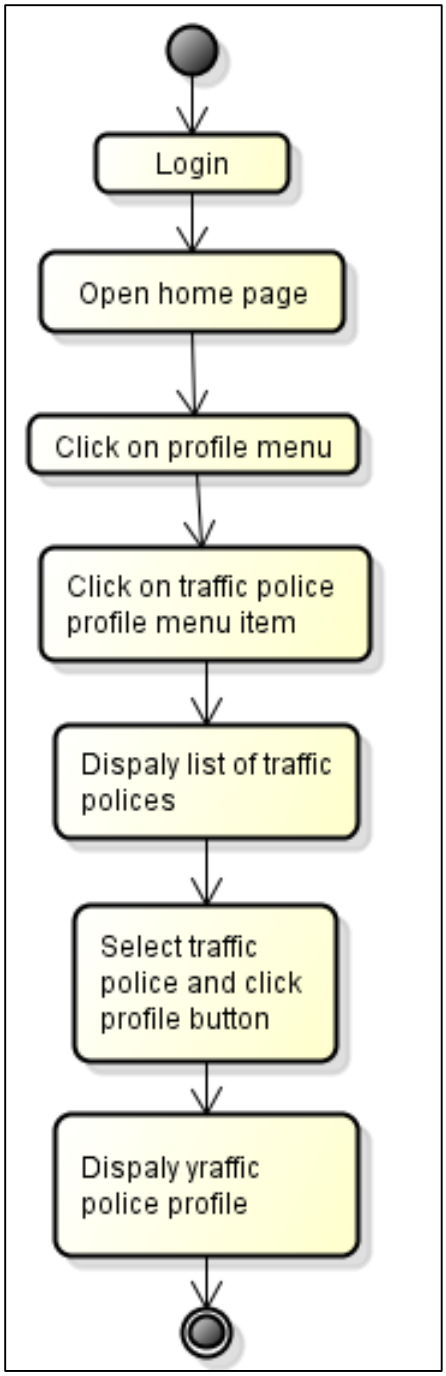


Figure 27: View Traffic police profile - Activity Diagram

View Vehicle info

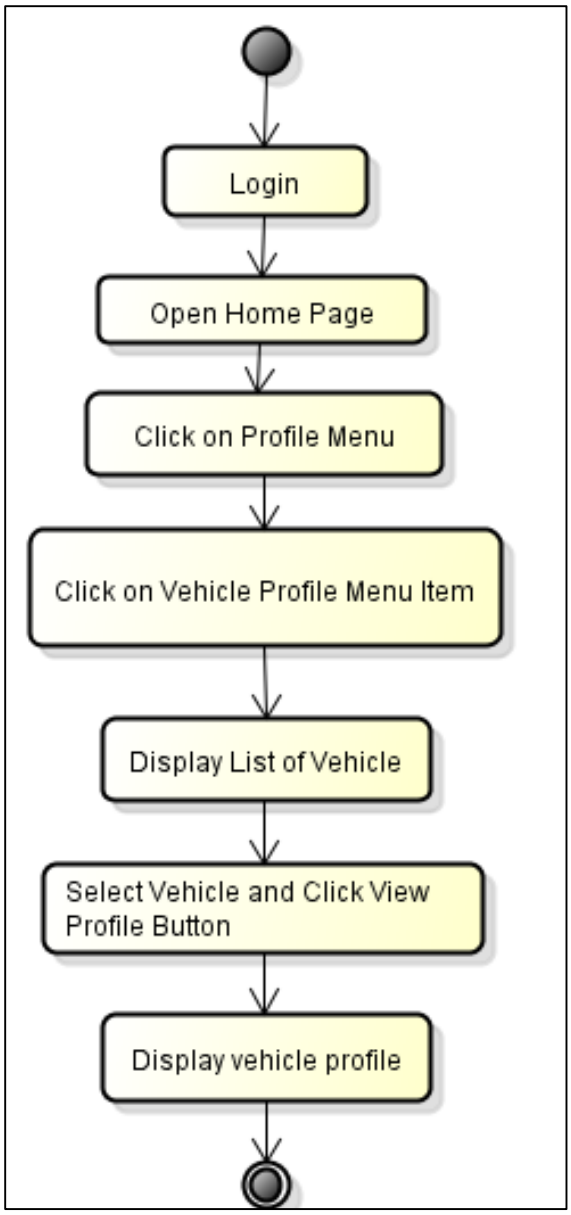


Figure 28: View Vehicle information - Activity Diagram

Punish Driver

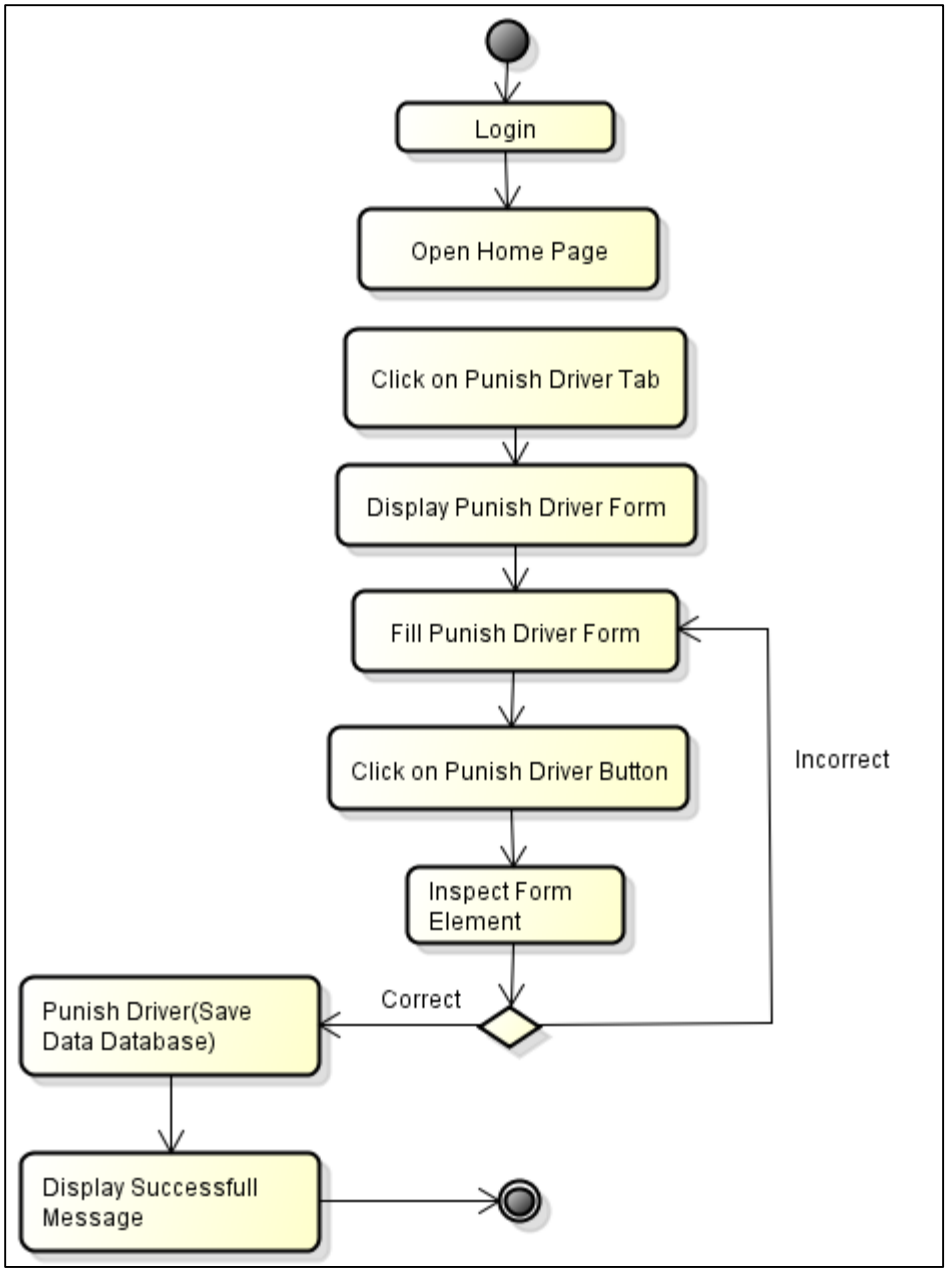


Figure 29: Punish Driver - Activity Diagram

3.4. State Chart Diagram

UML state machine diagrams depict the various states that an object may be in and the transitions between those states. In fact, in other modeling languages, it is common for this type of a diagram to be called a state-transition diagram or even simply a state diagram. A state represents a stage in the behavior pattern of an object, and like UML activity diagrams it is possible to have initial states and final states. An initial state, also called a creation state, is the one that an object is in when it is first created, whereas a final state is one in which no transitions lead out of. A transition is a progression from one state to another and will be triggered by an event that is either internal or external to the object.

Generate Report

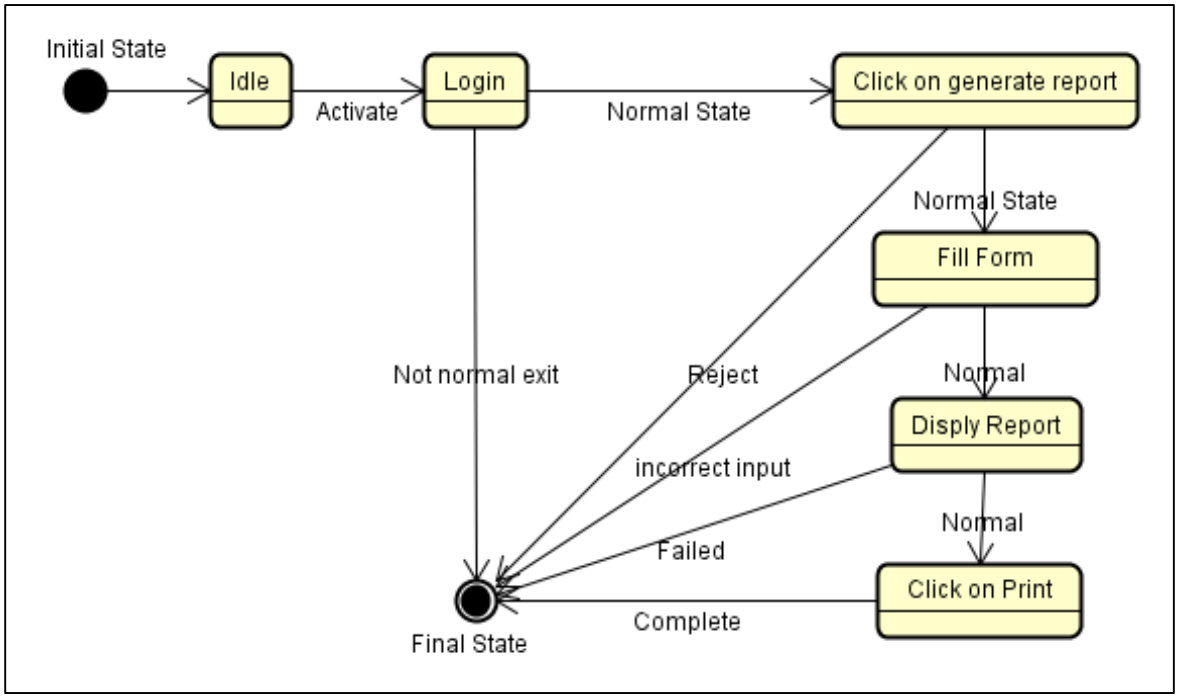


Figure 30: Generate Report - State chart Diagram

Register Traffic Police

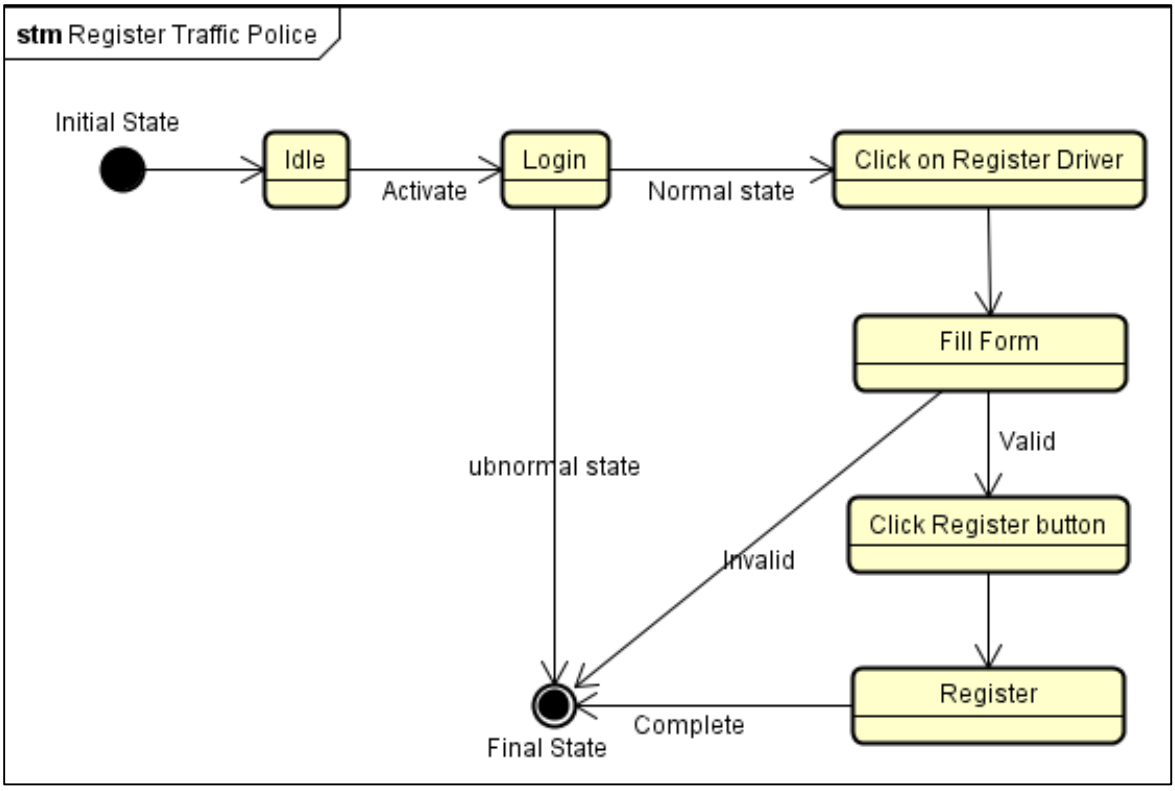


Figure 31: Register Traffic Police - State chart Diagram

View Driver Profile

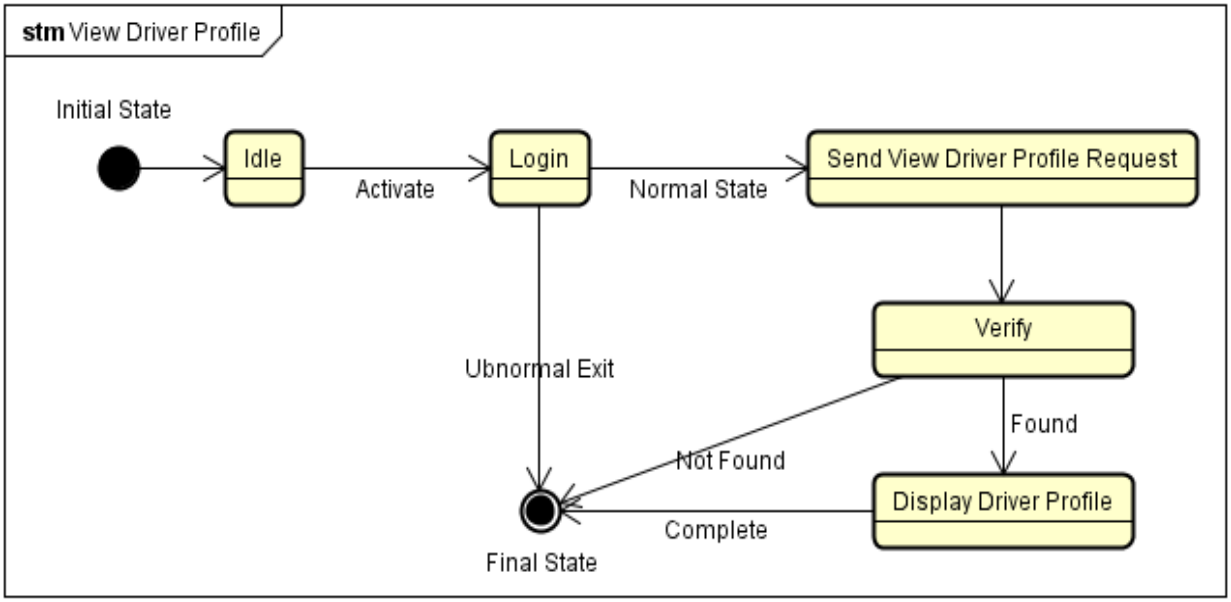


Figure 32: View Driver Profile - State chart Diagram

3.5. Class Diagram

A software system is a set of communicating entities that collaborates to perform task. The architectural Design is a top level design which shows these entities, their relationships.
And we use to describe or show architectural design using class diagram.

Class diagrams show the static structure of the model, in particular, the things that exist (such as classes and types), their internal structure, and their relationships to other things.

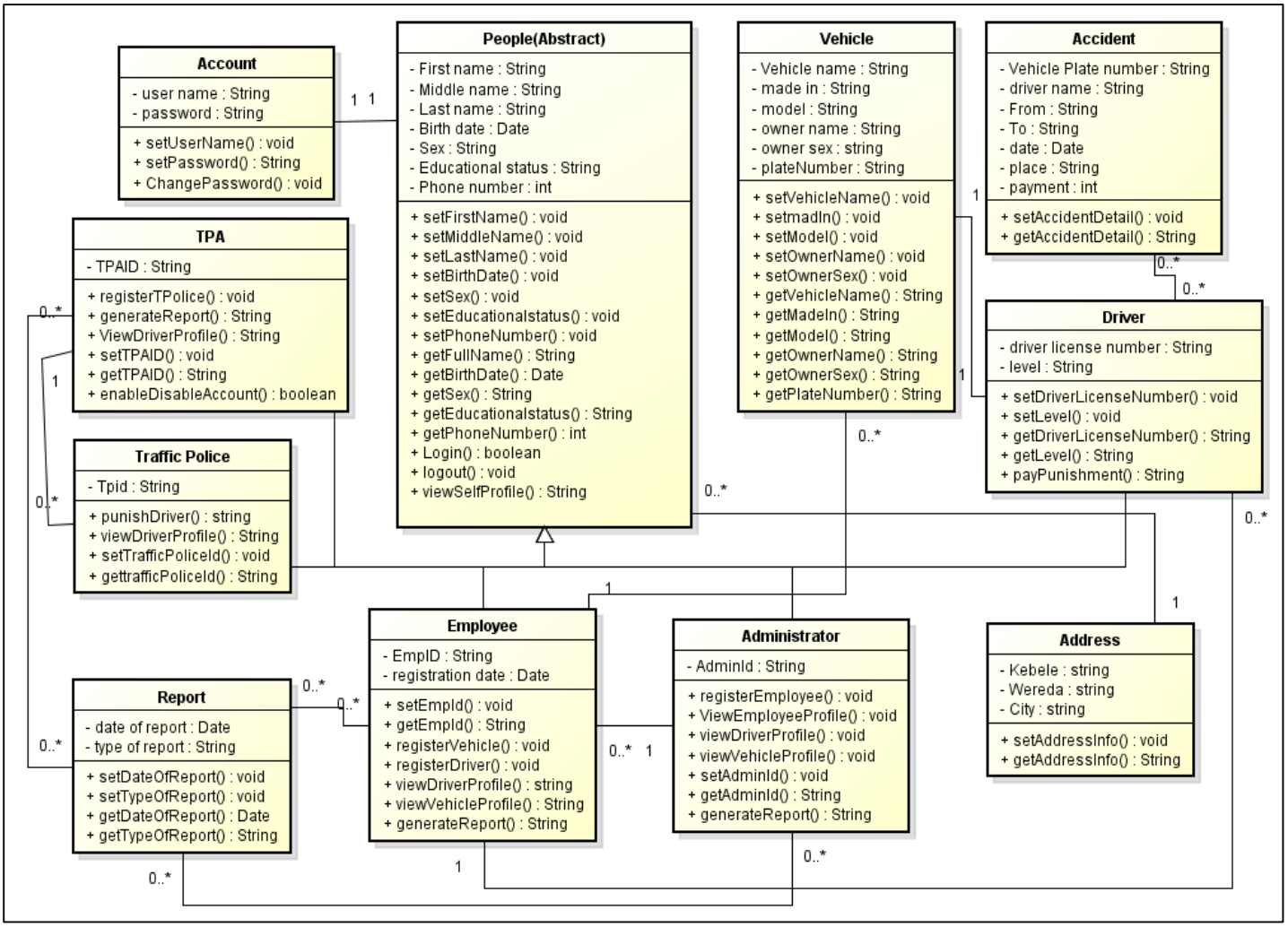


Figure 33: Class Diagram

Chapter Four

4. System Design

System design is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

4.1. purpose and goal of design

The purpose and design goal of the system is to provide a solution for problems that are listed on the analysis phase. Design goals are specification which the system needs to achieve.

Performance: the system need to have maximum throughput at minimum time to allow many users to access concurrently, and it should not take much space to be installed on users' cell phones.

Maintainability: the application should be easily scalable and extensible to add new features. And also it should be easily modifiable to make changes on the functionalities. To achieve this good and open documentation of the system is necessary. The maintainability of the system includes:

Perfective: adding new functionalities or changes to the system.
Adaptive: adapting changes to the external environment, changes to the government policy.
Corrective: fixing errors on the system.
Preventive: increase systems maintainability and reliability.

Usability: can be seen in two ways.

Effectiveness: the system should provide services which help the user to achieve their goal.
Efficiency: the users should spend minimum effort and resource to achieve their goal.

Reliability: the system should maintain and perform its functionalities under any condition. It should have minimum frequency of failure and adaptable for failures.

Security: the system should have authentication to allow only authorized users.

User-friendly: the system should be easy to learn, understand and operate. It should provide interactive and easy interfaces which can allow unprofessional users to deal with it.

4.2. Current system architecture

Currently there is no any software used, the activity's in the organization is done manually.

4.3. proposed software architecture

The proposed software has three- tier architecture.

- The presentation tier: is the top most level of the application. It is the one the clients directly interacted. It provides GUI to allow the client gaining access of the system.
- Logical tier/ middle tier: It accepts inputs from the client and performs detailed processing. It is a bridge between data access tier and presentation tier.
- Data access tier: provides data persistence mechanism and storage to the data. It consists of a mechanism to access the database without installing data base dependent drivers and libraries on the client device

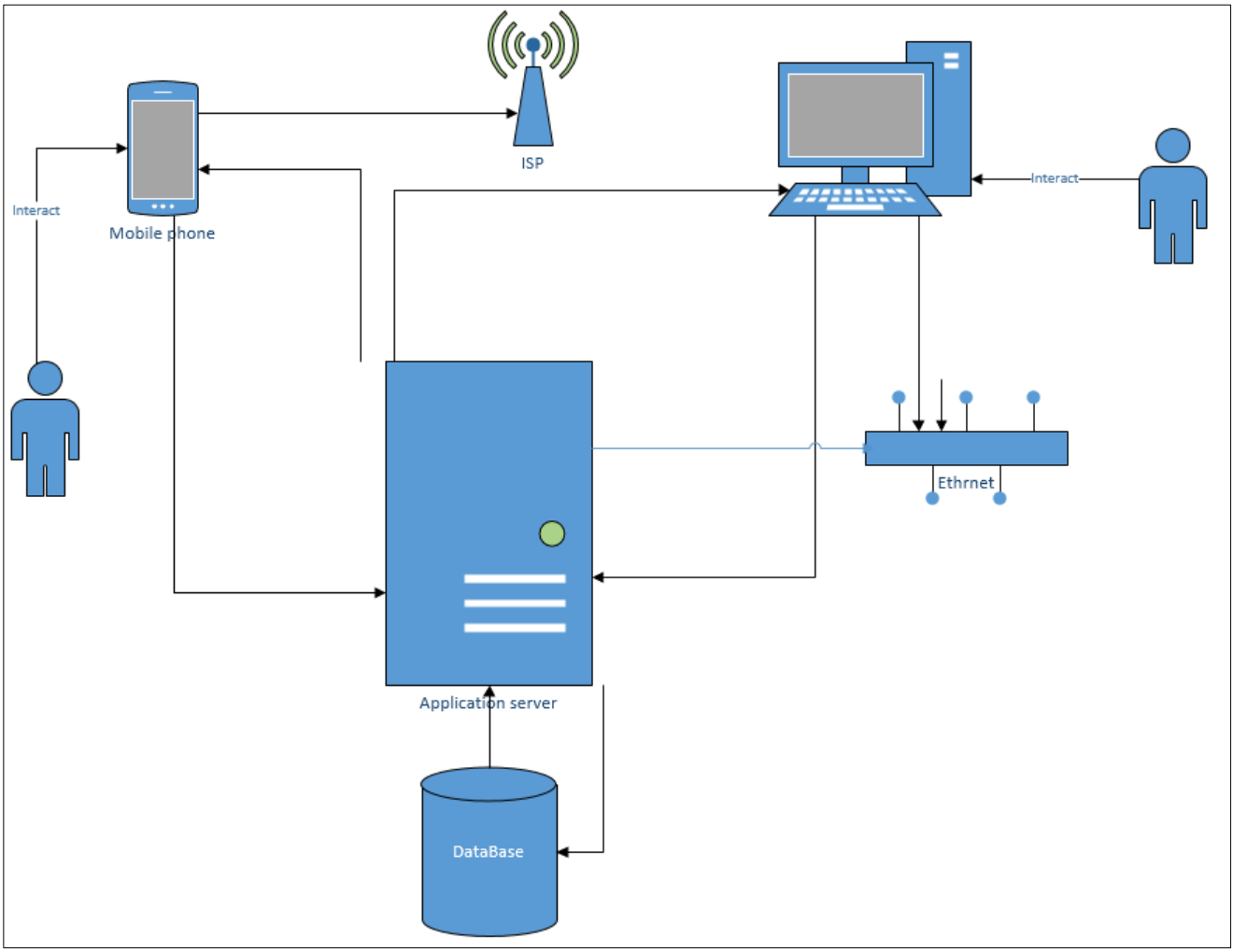


Figure 34: System Architecture

4.4. Deployment Diagram

Deployment diagrams show the configuration of run-time processing elements and the software components, processes, and objects that live on them. Software component instances represent run-time manifestations of code units.

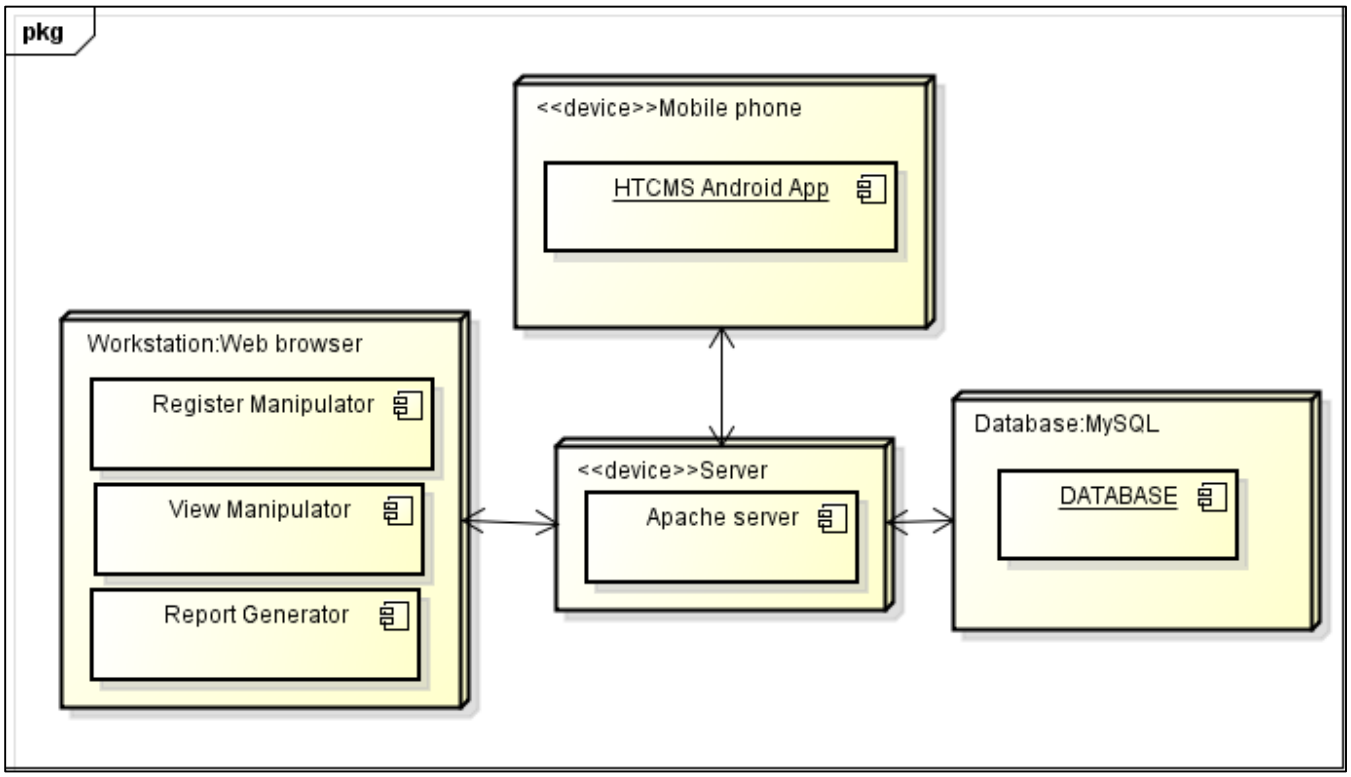


Figure 35: Deployment Diagram

4.5. User Interface Prototyping

User interface is the external part of the system which is used to access and interact with the system easily. Our system has two interfaces, the mobile application user interface and web application user interface

4.5.1. Navigation Tree for Web Application

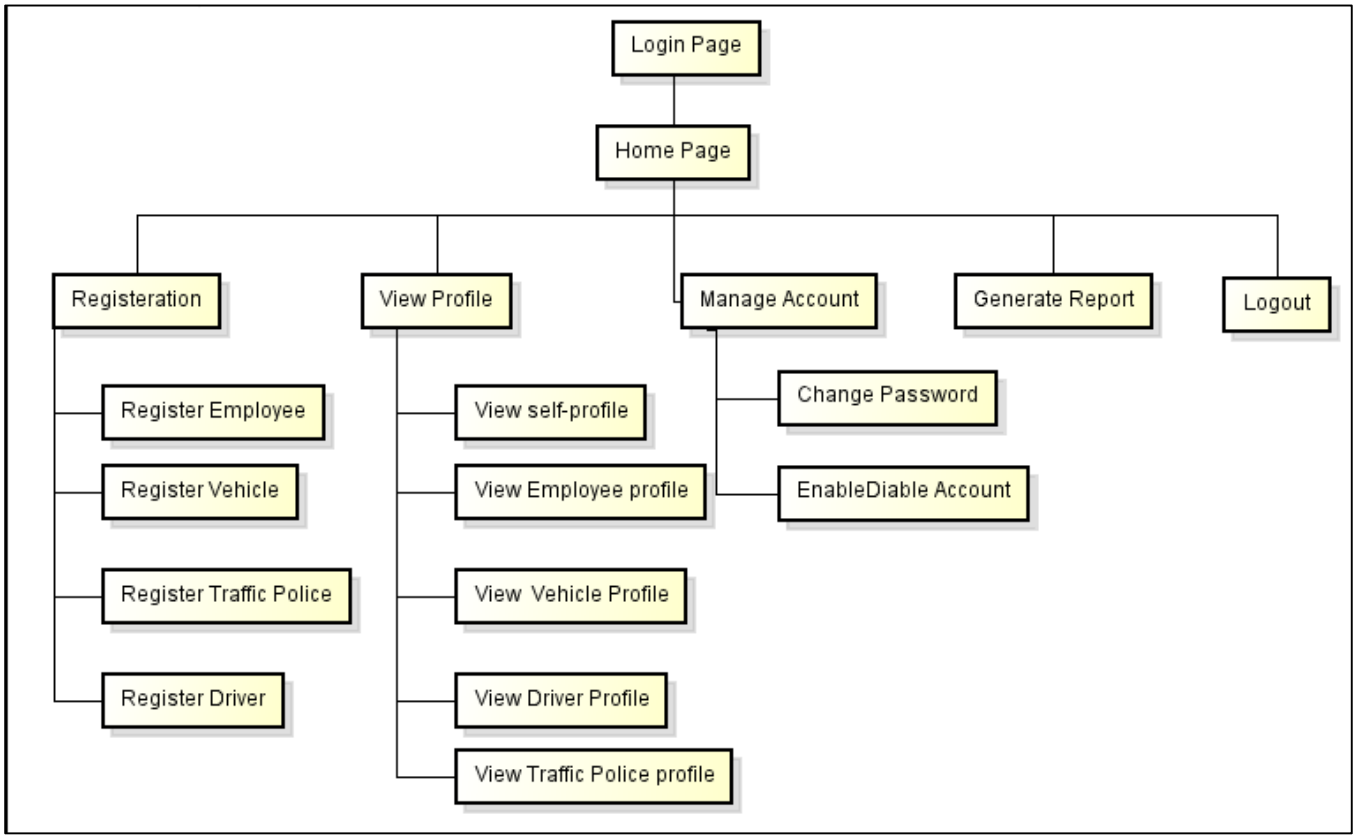


Figure 36: Navigation Tree Diagram for HTCMS Web application

4.5.2. Web Application User Interface

Below are some of the major interfaces that the Web application user faces while using our system.

- After the web application is browsed the login page displayed

Login

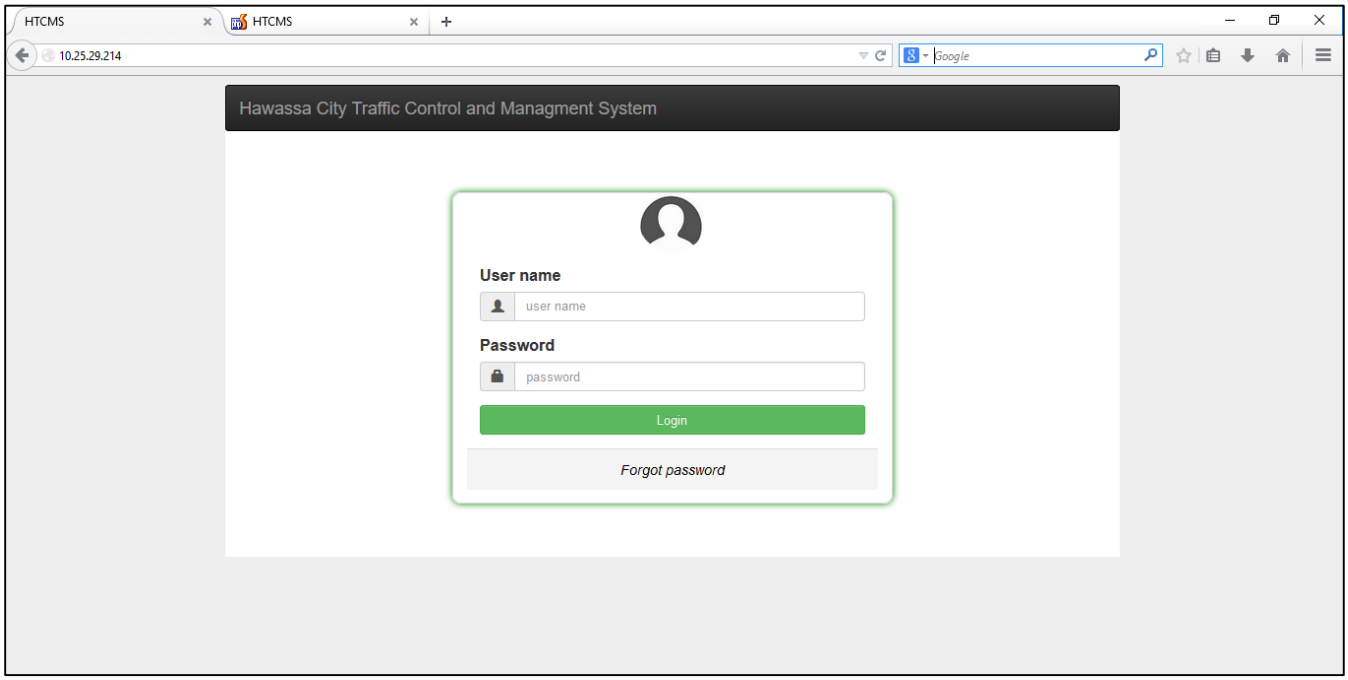


Figure 37:Login interface

- After the user login the Home page is displayed

Homepage

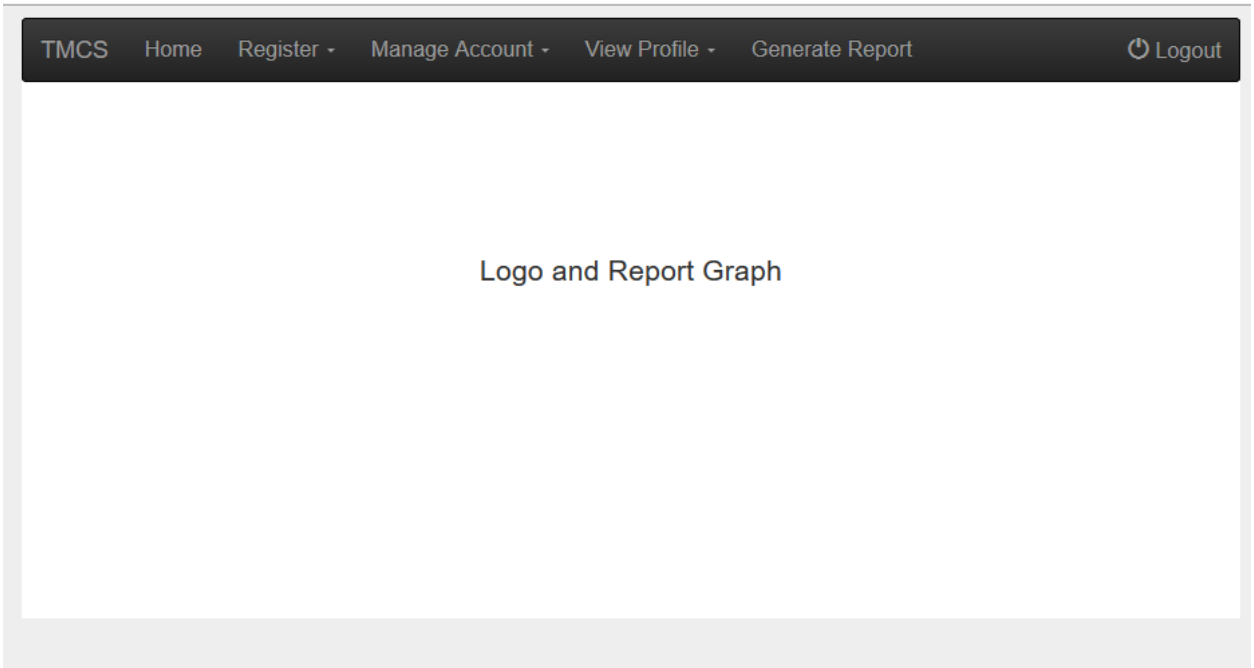


Figure 38: Homepage UI

- After the user clicks change password menu item under Manage Account, The change password form displayed

Change Password

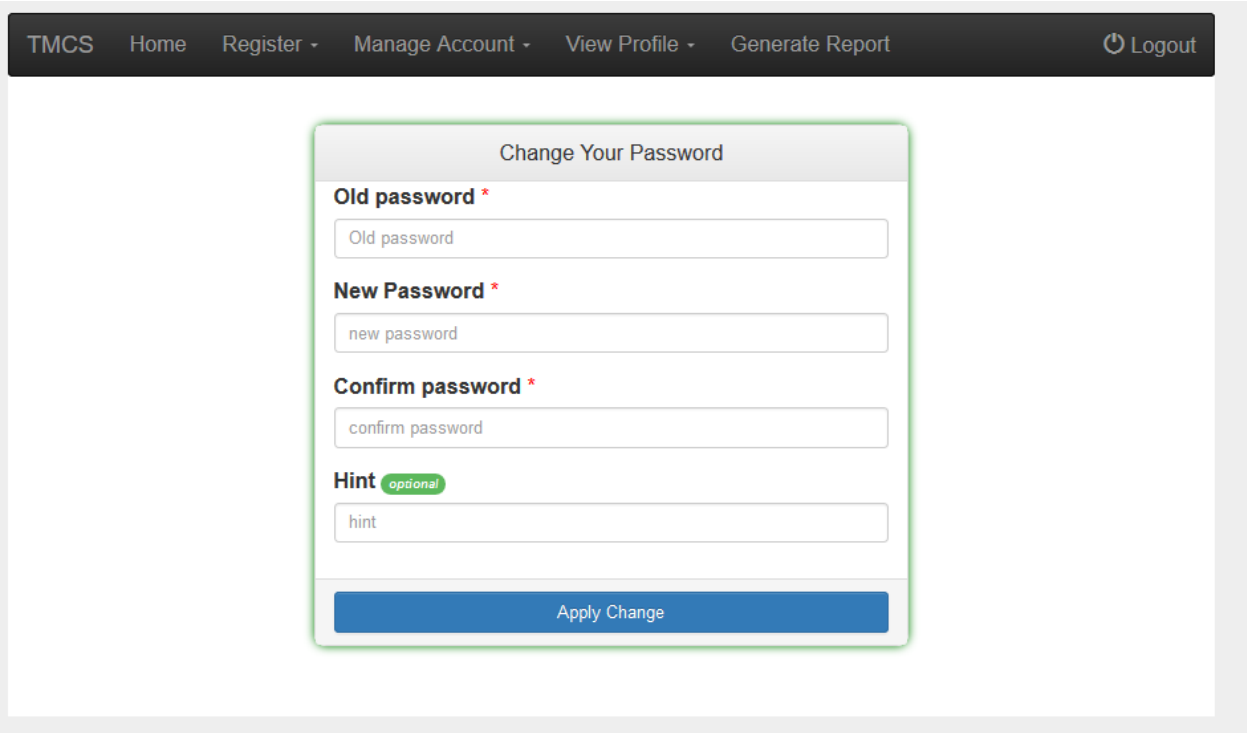


Figure 39: Change Password UI

- After Administrator click the Register Employee menu item under Register menu, The Register Employee form displayed

Register Employee

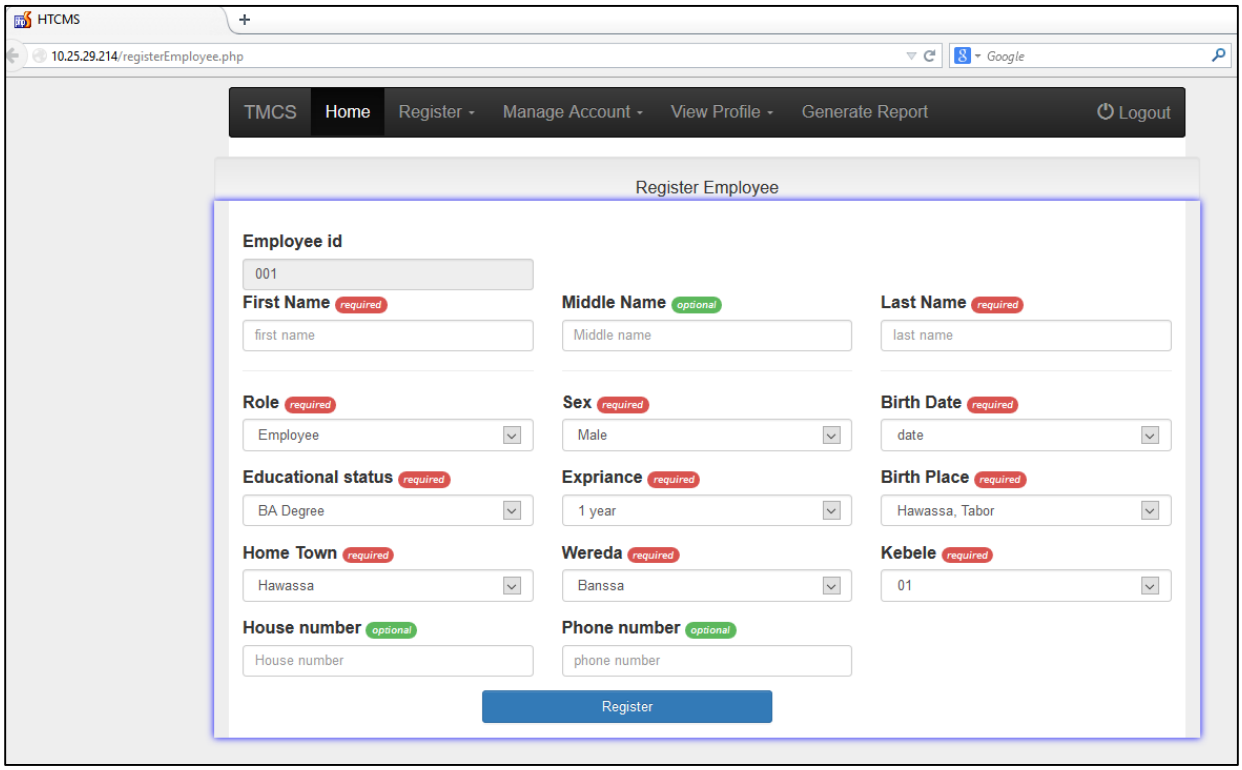


Figure 40: Register Employee interface

- After user click or hover on View profile menu item, menu item under View profile display

View Employee Profile menu

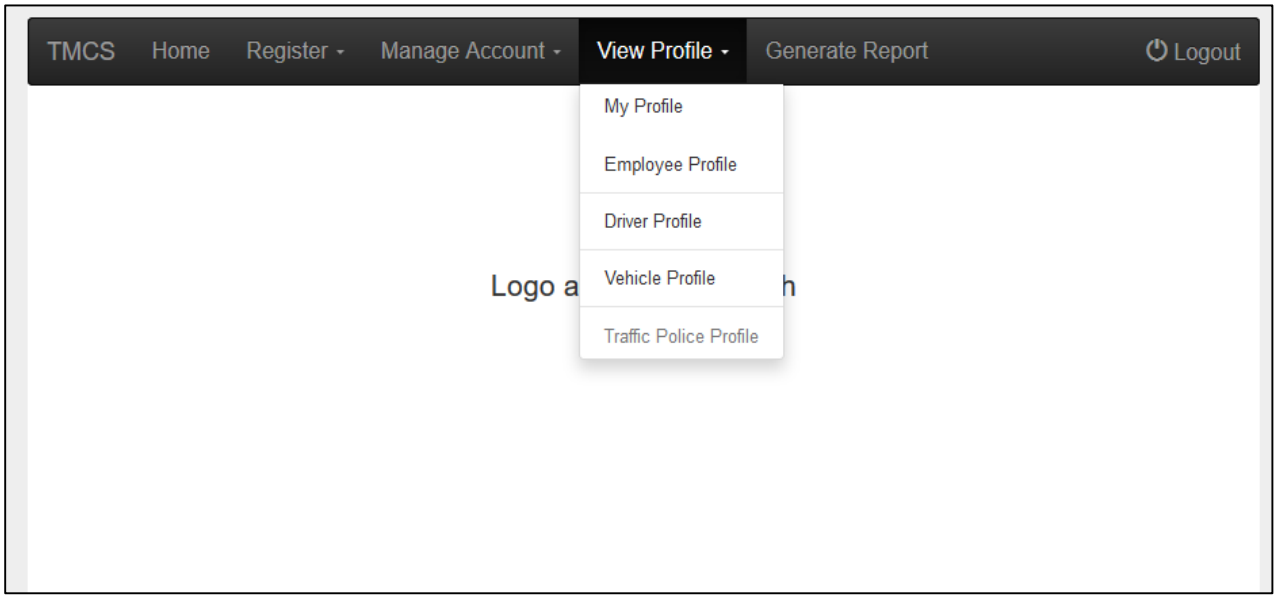


Figure 41: View Employee profile menu

- After user click on View Employee profile menu item List of Employee Displayed.

View Employee List

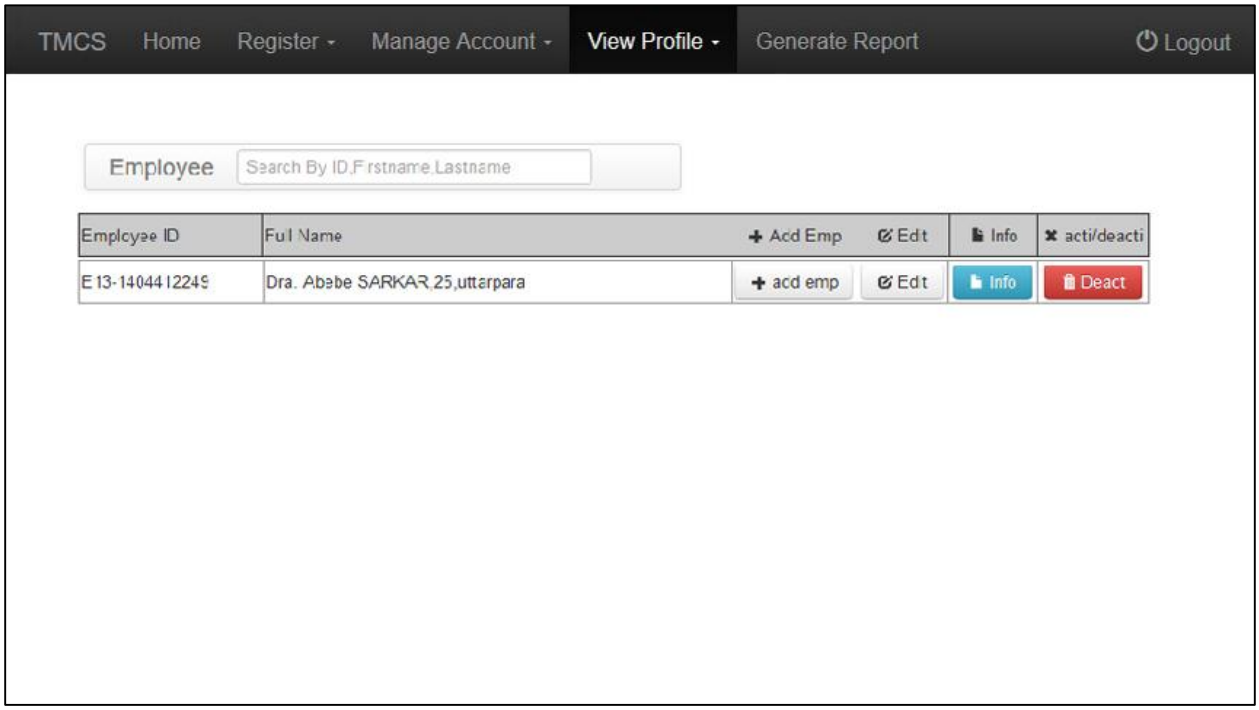


Figure 42: View Employee List

- After user select Employee and click on Info button, Employee profile displayed.

View Employee Full Profile

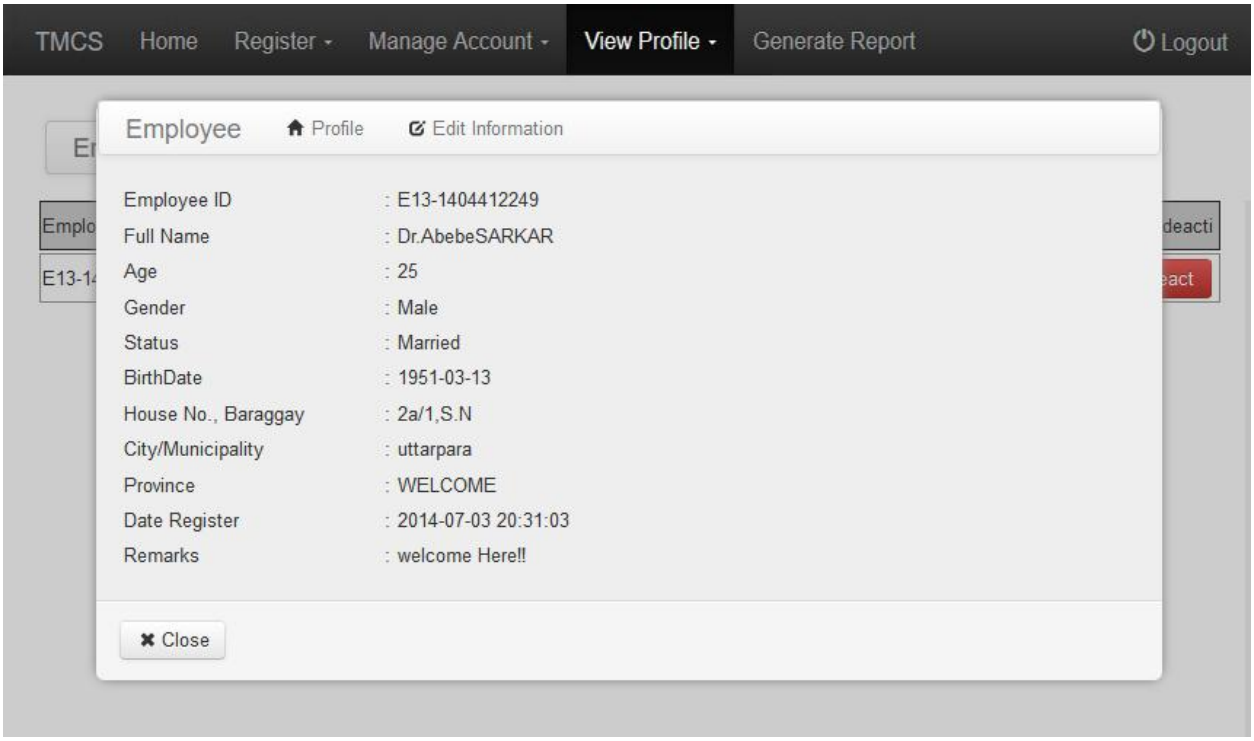
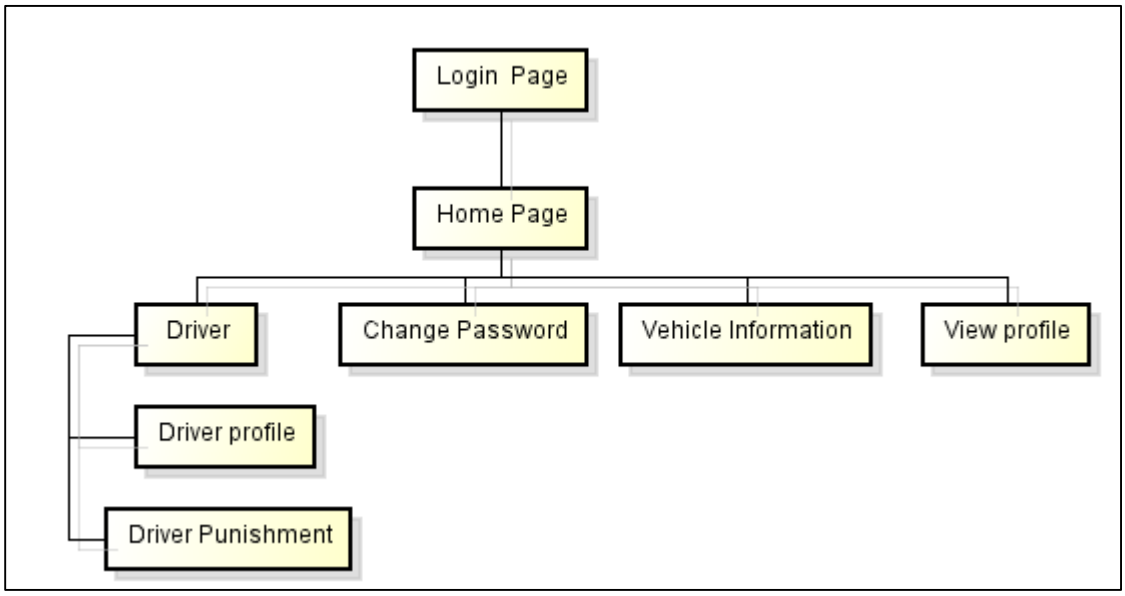


Figure 43: View Employee Full Profile

4.5.3. **Navigation Tree for Mobile Application**



4.5.4. **Mobile Application User interface**

After We launch HTCMS Android application the following login page will be displayed

Login

Figure 44: Login android interface

- After the user enters password and password correctly homepage will be displayed

Homepage

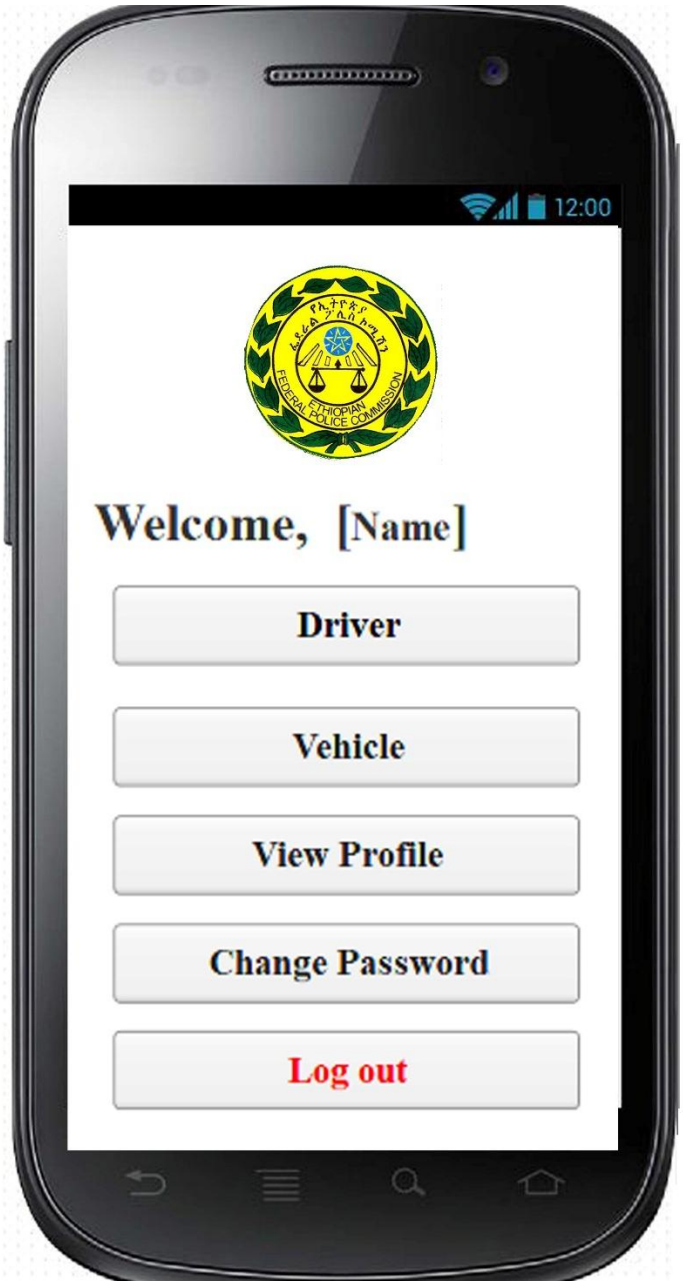


Figure 45: Homepage android interface

- When the user tabs on Driver button search driver dialog box displays, after the user enter License number and tab view button Driver profile will be displayed

Driver Profile

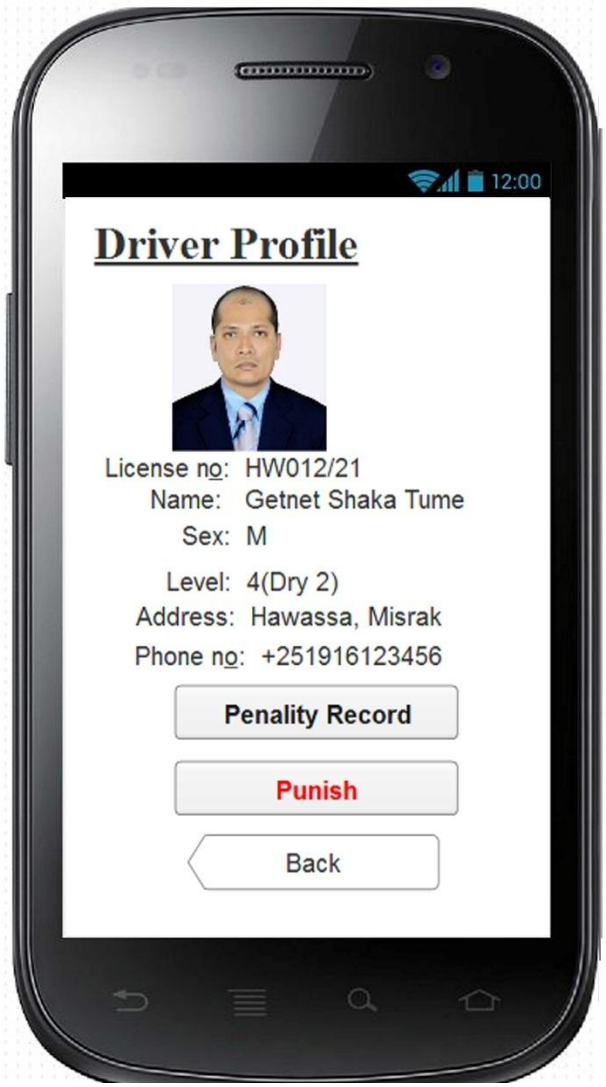


Figure 46: Driver Profile Android interface

- When the user tab Vehicle button and search Vehicle by entering plate number Vehicle profile displays

Vehicle Profile

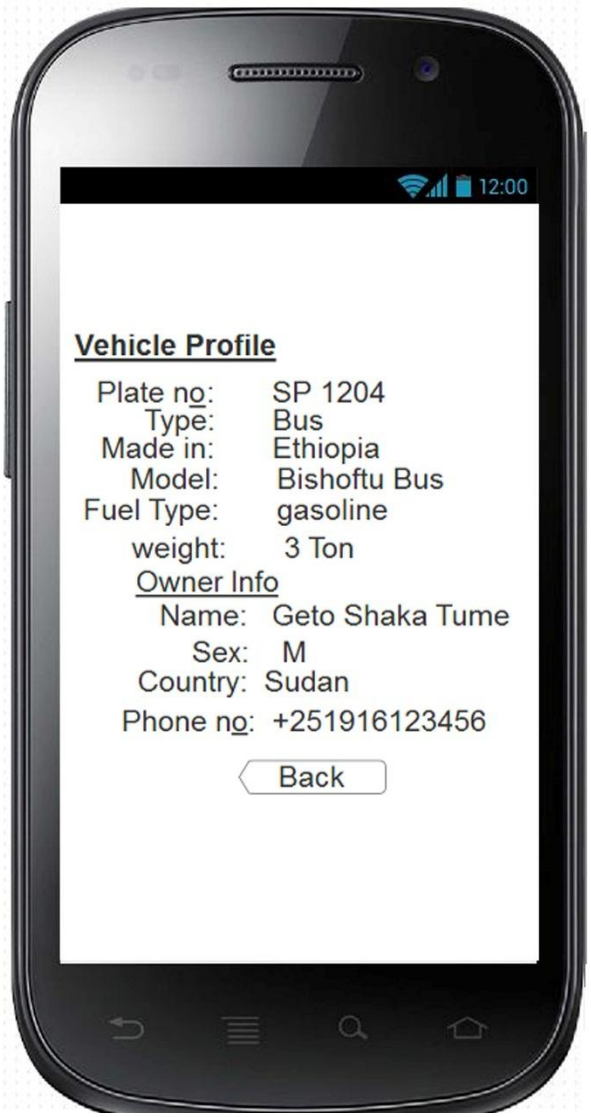


Figure 47: Vehicle Profile Android interface

- When the user taps View profile button user detail page displays

My Profile

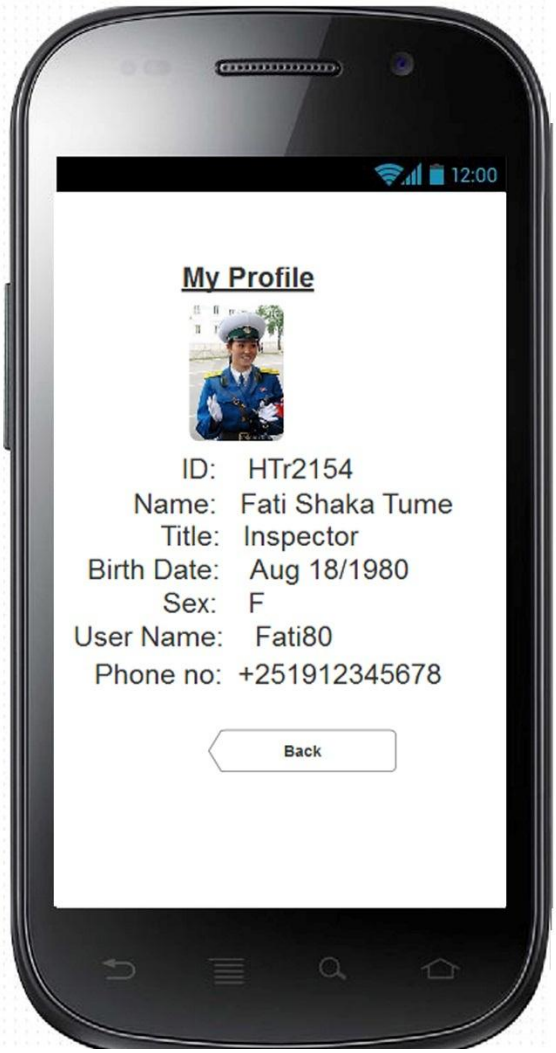


Figure 48: Self Profile Android interface

- When the user taps change password button change password form displays

Change Password

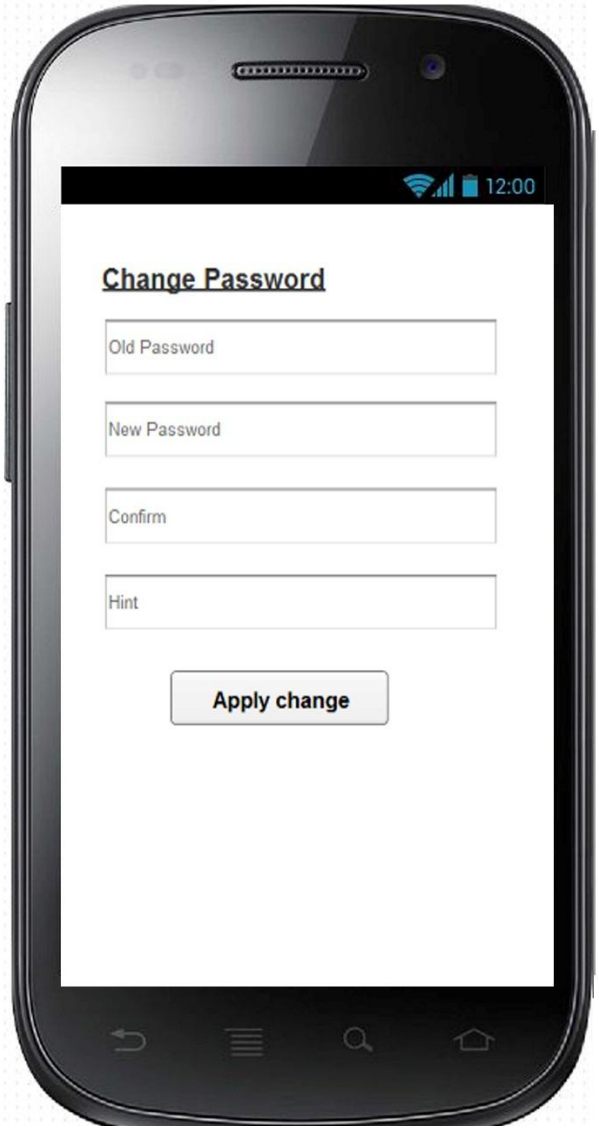


Figure 49: Change Password Android Interface

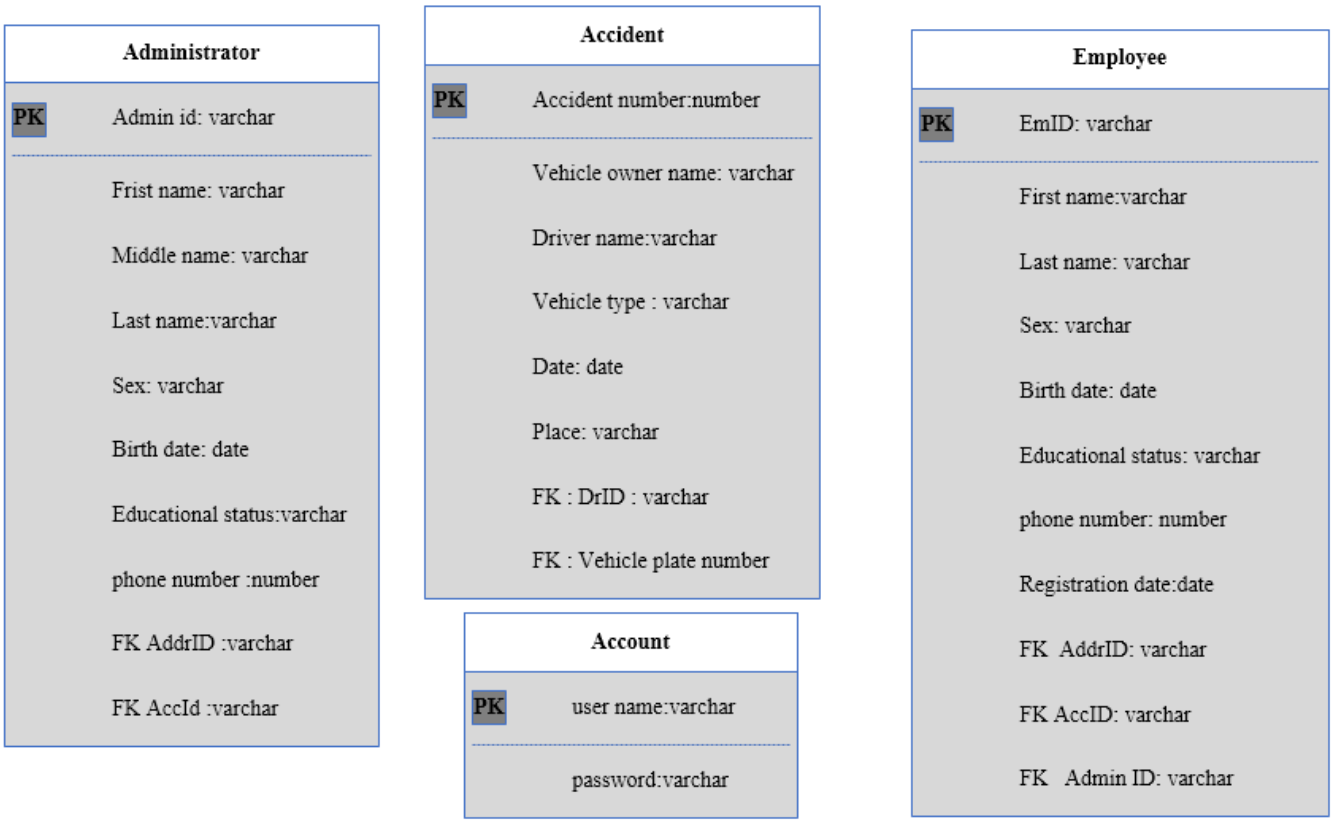
- When the user taps punish button from driver profile driver punishment form displays

Punish Driver

Figure 50: Punish Driver Android Interface

4.6. Database Design

4.6.1. Table Design



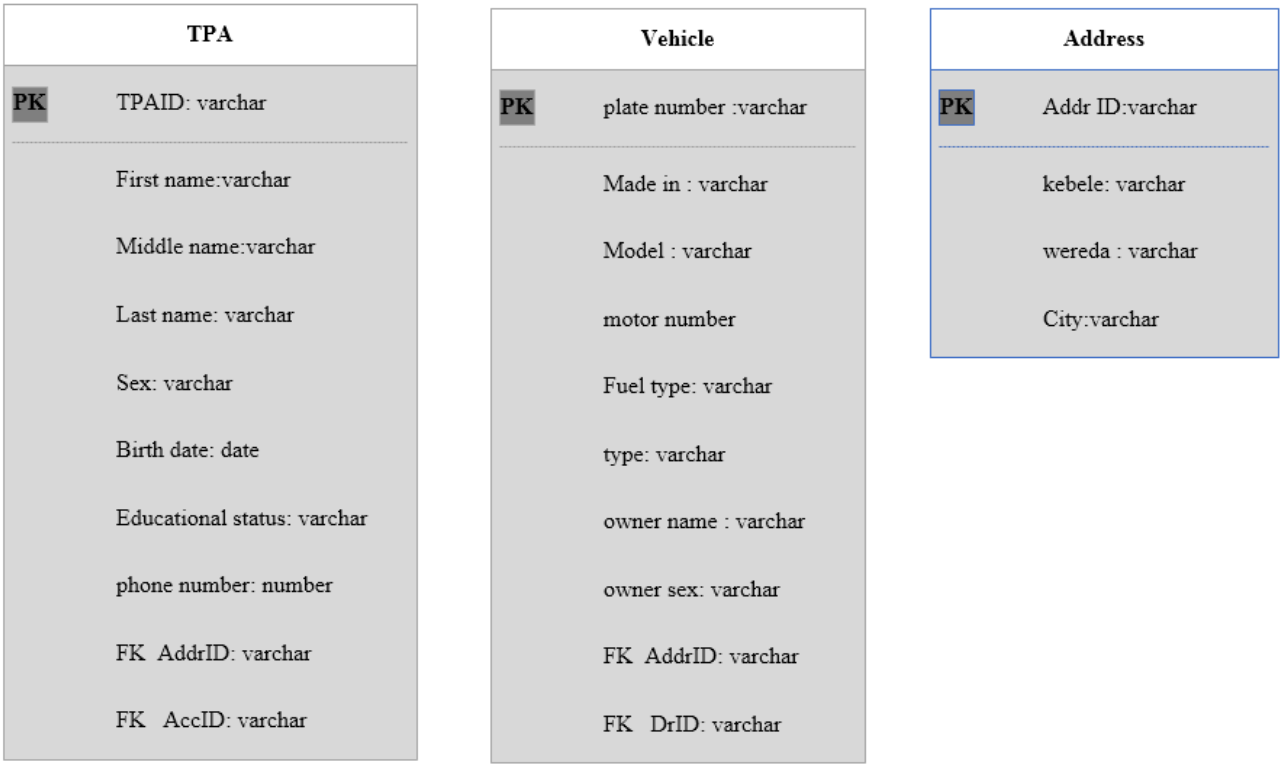


Figure 51: Database Table Design

4.6.2. **Normalization and Schema Design**

Normalization

Normalization is the process of organizing data into tables in such a way that the results of using the database are always unambiguous and as intended. Normalization may have the effect of duplicating data within the database and often results in the creation of additional tables. (While normalization tends to increase the duplication of data, it does not introduce redundancy, which is unnecessary duplication.) Normalization is typically a refinement process after the initial exercise of identifying the data objects that should be in the database, identifying their relationships, and defining the tables required and the columns within each table.

First Normal Form (1NF)

First normal form (1NF) sets the very basic rules for an organized database:

- ✓ Eliminate duplicative columns from the same table.
- ✓ Create separate tables for each group of related data and identify each row with a unique column or set of columns (the primary key).

Second Normal Form (2NF)

Second normal form (2NF) further addresses the concept of removing duplicative data:

- ✓ Meet all the requirements of the first normal form.

- ✓ Remove subsets of data that apply to multiple rows of a table and place them in separate tables.
- ✓ Create relationships between these new tables and their predecessors through the use of foreign keys.

Third Normal Form (3NF)

Third normal form (3NF) goes one large step further:

- ✓ Meet all the requirements of the second normal form.
- ✓ Remove columns that are not dependent upon the primary key.

1. **Mapping Administrator entity**

AdID	First name	Middle name	Last name	Sex	Birth date	Educational status	Phone number	AddrID	AccID

- ❖ **AccID**: foreign key references the primary key of Account entity
- ❖ **AddrID**: foreign key references the primary key of Address entity
- ❖ Full fill First, Second and Third normalization

2. **Mapping Address Entity**

AddrID	Kebele	wereda	city

- ❖ Full fill first, second and third normalization

3. **Mapping Account entity**

User name	password

- ❖ Full fill First, Second and Third normalization.

4. **Mapping Employee entity**

EmId	First name	Last name	Age	Sex	Birth date	Educational status	Registration date	AdID	AccID	AddrID

- ❖ **AdID**: foreign key references the primary key of Administrator entity
- ❖ **AddrID**: foreign key references the primary key of Address Entity
- ❖ **AccID**: foreign key references the primary key of Account entity
- ❖ Full fill First, Second and Third normalization.

5. **Mapping Vehicle entity**

plate number	Model	Made in	Motor number	Fuel type	Type	Owner name	Owner sex	Owner address	EmpID	DrID

- ❖ **EmpID:** foreign key references the primary key of Employee entity
- ❖ **DrID:** foreign key references the primary key of Driver entity
- ❖ Full fill First, second and third normalization.

6. **Mapping Driver entity**

Driver License number	First name	Last Name	sex	Region	Level	Registration date	Phone number	AddrID	EmpID	AccID

- **EmpID:** foreign key references the primary key of Employee entity
- **AddrID:** foreign key references the primary key of Address entity
- **AccID**: foreign key references the primary key of Account entity
- Full fill First, Second and Third normalization.

7. Mapping TPA entity

TPAID	First Name	Last Name	Sex	Birth date	Phone number	Educational status	Phone number	AddrID	AccID

- **AccID**: foreign key references the primary key of Account entity
- **AddrID**: foreign key references the primary key of Address entity
- Full fill First, Second and Third normalization.

8. Traffic police entity

TpID	First name	Last name	Age	Sex	Birth date	Address	TPAID	AccID	AddrID

- **AccID**: foreign key references the primary key of Account entity
- **AddrID**: foreign key references the primary key of Address entity
- **TPAID:** foreign key references the primary key of TPA
- Full fill First, Second and Third normalization.

9. Mapping Punishment relation ship

TpID	DLnumber	Date

- **TpID**: foreign key references the primary key of Traffic Police.
- **DLnumber**: foreign key references the primary key of Driver entity.
- Full fill First, Second and Third normalization

10. Mapping Report entity

ReID	Report date	Report type

- Full fill First, Second and Third normal form

11. Mapping Administrator-Report Relation ship

AdminID	ReID

- **AdminID**: foreign key references the primary key of administrator
- **ReID**: foreign key references the primary key of Report entity
- Full fill First, Second and Third normal form

12. Mapping TPA-Report relation ship

TPAID	ReID

- **TPAID**: foreign key references the primary key of TPA

- **ReID**: foreign key references the primary key of Report entity
- Full fill First, Second and Third normal form

13. Mapping Employee-Report Relation ship

EmID	ReID

- **EmID**: foreign key references the primary key of Employee
- **ReID**: foreign key references the primary key of Report entity
- Full fill First, Second and Third normal form

14. Mapping Accident/Crime entity

Accident number	Vehicle owner name	Driver name	From	To	Vehicle type	Date	Place	DrID	Vehicle Plate Number

- **DrID:** foreign key references the primary key of Driver entity
- **Vehicle Plate number:** foreign key references the primary key of Vehicle
- Full fill First Normal Form
- Requires Second Normal Form because they do not fully functional dependent

2NF

14. 1 Mapping Accident/Crime entity

DrID	Vehicle Plate number	From	To	Date	Place

14.2 Mapping Accident/Crime entity

Vehicle plate number	Vehicle owner name	Vehicle type

14.3 **Mapping Accident/Crime entity**

DrID	Driver Name

- Full fill Third normal form.

4.6.3. Access control and security Access control

Here we will describe the privileges or authorities of actors over the functionalities. In this system there are three actors Administrator, Employee, Traffic Police, Traffic police Administrator and Driver. Each has their own privileges to gain access of the system.

Below is the description of access control using access control matrix.

	Administrator	Employee	Driver	TPA	Traffic police
Login	✓	✓	✓	✓	✓
Register Employee	✓				

Register Driver		✓			
Register Traffic police				✓	
Register Vehicle		✓			
View self-profile	✓	✓	✓	✓	✓
View Employee profile	✓				
View Driver profile	✓	✓		✓	✓
View Vehicle Profile	✓	✓		✓	
View Traffic police profile				✓	
Change password	✓	✓	✓	✓	✓
Enable/Disable Account	✓			✓	
Generate report	✓	✓		✓	
Punish Driver					✓
Logout	✓	✓	✓	✓	✓

Table 18: Access control table

Security

Here are some security issues taken in the system

- All users of the system must first login to perform operation and get services.
- When Employee, Driver and Traffic Police Registered, the will get user name and password.
- The user name and password of the users are encrypted and store in the database.

5. Conclusion

In this project, we develop a web application for Hawassa city traffic control and management system with Driver license control mobile application that minimizes time and money consumption for doing those activities.

The system consists of mobile and web application. These are two different applications on the same database. The mobile application that is phone based android application that enables the user to do activities like Punish Driver, register crime or car accident, control fake Driver license and View profile. The web application enables to create Register Employee, Vehicle, Driver, Traffic police, view Employee profile, Vehicle profile, Driver profile, and Traffic police profile and generate report.

To collect the data which is required for our project we use Open Interview, analysis of existing manual system. Our model contains analysis model which contains the functional and non-functional requirements, use case, sequence, state chart, activity diagram, conceptual modeling of classes and user interface prototypes. And also contains design model which consists deployment diagram, system architecture design, Database table design, Schema design and normalization and access control.

The problem, which is existed in the earlier manual system, will be removed to a large extent. And it is expected that this project will go a long way in satisfying requirements

Reference

Object-Oriented Software Engineering Using UML, Patterns, and Java, 2nd Edition

Elmasri-Navathe-Fundamentals of Database Systems 3rd Edition

Hawassa RTA Accident Registration booklet

www.wikipedia.com

Appendix 1

Interview Questions

1. What is the existing system your organization using now?
2. What are the existing system drawbacks?
3. What methods are used to register an employee?
4. How illegal (criminal) drivers are punished?
5. How does this office store records and information?