

A MASTER'S PROJECT

ON

WEB BASED TRANSPORT MONITORING AND TRAFFIC SYSTEM

FOR ADAMA TRANSPORT AGENCY

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE DEGREE

OF

M.Sc. IN TECHNICIAN TEACHERS EDUCATION IN INFORMATION TECHNOLOGY MANAGEMENT

BY:

MISRAK DAGNE

ID:GSR/5307/06

BRUK T/BERHAN

ID:GSR/5295/06

MAHAMMED MAHMMUD

ID:GSR/5308/06

Advisors:

DR. MOHAMMED IQBAL

ATO: NEMERA J.



ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF COMPUTING

JUNE, 2015

Adama ,Ethiopia

Project Title: WEB BASED TRANSPORT MONITORING AND TRAFFIC SYSTEM

Submitted To: ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF COMPUTING

Group Members

<u>Name</u>	<u>Signature</u>	<u>Date</u>
1. Bruk T/Berhan	_____	_____
2. Misrak Dagne	_____	_____
3. Mahammed Mahmmmed	_____	_____

Approval sheet

A. Candidate

_____	_____	_____
1. Student Name	sign.	Date
_____	_____	_____
2. Student Name	sign.	Date
_____	_____	_____
3. Student Name	sign.	Date

B. Approved By

1.		
_____	_____	_____
Advisor Name	sign.	Date
2.		
_____	_____	_____
Co Advisor Name	sign.	Date
3.		
_____	_____	_____
Internal Examiner Name	sign.	Date
4.		
_____	_____	_____
External Examiner Name	sign.	Date
5. _____	_____	_____
Chairperson Name	sign.	Date
6.		
_____	_____	_____
Department Head	sign.	Date.
7.		
_____	_____	_____
School Dean	sign.	Date

CERTIFICATE

The project design and implementation of Web **based transport monitoring and traffic system** is by Bruk T/birhan , Misrak Dagne and Mahammed Mahmmmed has been carried out under my supervision and this work has not been submitted elsewhere for a degree. This project has been submitted for examination with my approval as a University advisor.

Advisor Name: DR.mohammed Iqbal Signature _____ date: _____

DECLARATION

We Bruk T/Berhan, Misrak Dagne, and Mahammed Mahmmud hereby declare that this project entitled "WEB BASED TRANSPORT MONITORING AND TRAFFIC SYSTEM" is entirely our own work and has not been done or presented by other person. The result of our own work carried out under the supervision of Dr Mohammed Iqbal.

We verify that all the material used has been properly referenced.

Name

Signature

Date

1. Bruk Telebirhan

2. Misrak Dagne

3. Mahammed Mahmmud

ABSTRACT

A transport sector is the main back bone of Ethiopian economy and indeed of any economy. This is due to the fact that transports influences and influences by other sectors that make up. With growing demand of on the road as a major mode of transport, vehicle registration and licensing, driver licensing, vehicle control has a greater impact. Vehicle monitoring system on the other hand is a phenomenon that is still estranged to our country.

The obvious need of web based vehicle monitoring and traffic system prompted this project to see ways of integrating our vehicle licensing to the internet for speed, efficiency, reliability and customer satisfaction.

The web based Adama transport monitoring and traffic control system integrate, vehicle registration and licensing, driver registration and licensing and monitoring.

ACKNOWLEDGMENT

In the every begging admire to the Almighty God who gave us endurance and courage from the beginning up to this phase.

With deep sense of gratitude we acknowledge the guidance of our project advisor **Dr Mohammed Iqbal**. His assistance and encouragement has played vital role in preparing and our completing the project.

We acknowledge the staff and management of Adama Transport Agency for providing us with useful information and materials.

We acknowledge our co-advisor **Ato Nemera J.**

Finally, we acknowledge some of my fellow class mates who have been a source of inspiration to us; to our friends and well-wishers who were invaluable to us during the course of this project.

TABLE OF CONTENTS

CERTIFICATE	i
DECLARATION	ii
ABSTRACT	iii
ACKNOWLEDGMENT	iv
List of abbreviations.....	x
1. Introduction.....	1
1.1 Background of the study	2
1.2 Statement of the problem	3
1.3 Objectives of the project	4
1.3.1 General objective.....	4
1.3.2 Specific objective	4
1.4 Scope and limitations of the project	5
1.5 Significance of the project	5
1.6 Methodologies of the project.....	6
1.6.1. Data Collection Methodology.....	7
1.6.2 System analysis design Methodology	8
1.6.3 System development tools	8
1.7 Feasibility study.....	8
1.7.1 Operational feasibility	8
1.7.2 Technician feasibility	8
1.7.3 Economic feasibility.....	8
1.7.4 Schedule feasibility	9
2. Review of Related Literature	10
2.1 Transport and Development	10
2.1.1 Transport Demand.....	11
2.2 Theoretical Background.....	11
2.3 Related Literature	12
2.3.1 System Architecture survey	12
2.3.2 Available Technologies	14
2.3.3 Web application Model	16
2.4 Related Studies	18

3. Requirement analysis	20
3.1 Current System	20
3.2 Proposed System.....	22
3.2.1 Overview.....	22
3.2.2 Functional Requirements	22
3.2.3 Non Functional Requirements	23
3.3 Hardware And Software Requirements	25
3.4 Software Engineering And Software Requirement	25
3.5 Analysis Activities.....	25
3.5.1 Use Case Diagrams	26
3.5.2 Class Diagrams	42
3.5.3 Sequence Diagram.....	44
3.5.4 Activity Diagram.....	53
3.5.5 State Chart Diagram	57
3.5.6 Component Diagram	60
3.5.7 Deployment Diagram	61
4. System Design	62
4.1. Introduction	62
4.1.1 Purpose Of The System	62
4.1.2 Design Goals	62
4.1.3 Overview.....	63
4. 2. Current Software Architecture	63
4. 3. Proposed Software Architecture.....	63
4.3.1 Overview.....	64
4.3.2 Subsystem Decomposition.....	64
4.3.3 Hardware/Software Mapping.....	66
4.3.4 Persistent Data Management.....	66
4.3.5 Access Control and Security.....	68
4.3.6 Global Software Control.....	69
4.3.7 Boundary Conditions.....	70
4.4. Subsystem services	71
5. System Implementation and result	73

5.1 Introduction	73
5.2 Algorithms Used.....	73
5.3 Methods Used.....	74
5.4 Result	74
6. System Testing.....	75
6.1 Test Plan.....	75
6.1.1 Introduction.....	75
6.1.2. Test Plan Overview	75
6. 1.3 Features to be tested /not to be tested.....	77
6.1.4 Approach	78
6.1.5 Testing Material	79
6.2 Test Case Specification.....	79
6.2.1 Unit Test Cases	79
6.2.2 Functional Test Cases.....	82
6.2.3 Sample Login Test Case Result	85
7. Conclusion	87
7.1 Conclusion.....	87
7.2 Results Achieved	88
1.3 Limitation Of The Project	88
7.4 Future Works.....	88
ANNEX : A	90
ANNEX : B	91
APPENDIX : A.....	106
Appendix B.....	108
Appendix : C.....	109
Appendix : D.....	111
Appendix: F	112
Bibliography	114

List Of Table

TABLE 1-1: INTANGIBLE COST BENEFITS WORKSHEET TABLE	9
TABLE 3- 1: USE CASE FOR MANAGE USER	29
TABLE 3- 2: USE CASE FOR LOGIN	30
TABLE 3-3: USE CASE FOR REGISTER VEHICLE	31
TABLE 3-4: USE CASE FOR REGISTER VEHICLE'S OWNERSHIP	32
TABLE 3-5: USE CASE FOR REGISTER TRAINEE DRIVER	33
TABLE 3-6: USE CASE FOR ISSUE DRIVER LICENSE	34
TABLE3-7: USE CASE FOR ISSUE LIBRE	35
TABLE 3-8: USE CASE FOR ISSUE BOLO	36
TABLE 3-9: USE CASE FOR TRANSFER OR UPDATE VEHICLE AND OWNER	37
TABLE 3-10: USE CASE RENEWAL BOLO	38
TABLE 3-11: USE CASE FOR RENEWAL DRIVER LICENSE	39
TABLE 3-12: USE CASE FOR GENERATE REPORT	40
TABLE 3-13: USE CASE FOR CHARGING	41
TABLE 3-14: USE CASE FOR PAYMENT	42
TABLE 4- 1: ACCESS CONTROL MATRIX	68
TABLE 6- 1 FUNCTIONAL TEST PLAN	76
TABLE 6- 2: INTEGRATION TEST CASE PLAN	76
TABLE 6- 3: UNIT TEST CASE PLAN	77
TABLE 6- 4: LIBRE UNIT TEST CASE	79
TABLE 6- 5: LIBRE UNIT TEST CASE	79
TABLE 6- 6: LIBRE UNIT TEST CASE	80
TABLE 6- 7: LIBRE UNIT TEST CASE	80
TABLE 6- 8: DRIVERLICENSE CLASS UNIT TEST	80
TABLE 6- 9: DRIVERLICENSE CLASS UNIT TEST	81
TABLE 6- 10: DRIVERLICENSE CLASS UNIT TEST	81
TABLE 6- 11: TRANSFER LIBRE CLASS UNIT TEST	81
TABLE 6- 12: ISSUE BOLO CLASS UNIT TEST	82
TABLE 6- 13: VALIDATE BOLO CLASS UNIT TEST	82
TABLE 6- 14: MANAGE USER TEST CASE	82
TABLE 6- 15: LOGIN TEST CASE	83
TABLE 6- 16: ISSUE BOLO TEST CASE	83
TABLE 6- 17: REGISTER TRAINEE DRIVER TEST CASE	83
TABLE 6- 18: ISSUE DRIVER'S LICENSE TEST CASE	84
TABLE 6- 19: DRIVER CHARGE CASE TEST	84
TABLE 6- 20: ISSUE LIBRE	84

LIST OF FIGURES

FIGURE 2: 1 J2EE MODEL	17
FIGURE 2: 2 MVC MODEL	18
FIGURE 3-1 USE CASE DIAGRAM	27
FIGURE 3-2 : CLASS DIAGRAM	43
FIGURE 3- 2 :LOGIN SEQUENCE DIAGRAM	44
FIGURE 3- 3:SEQUENCE DIAGRAM FOR REGISTER USER	45
FIGURE 3- 4: REGISTER VEHICLE	46
FIGURE3.5:OWNER REGISTER	47
FIGURE 3- 7:PAYMENT	49
FIGURE 3- 8: ISSUE DRIVER LICENSE	50
FIGURE 3- 9: RENEW BOLO	51
FIGURE 3- 10: ISSUE LIBRE	52
FIGURE 3- 11:LOGIN ACTIVITY	53
FIGURE 3- 12:REGISTRATION ACTIVITIES	54
FIGURE 3- 13: ACTIVITY DIAGRAM FOR CHARGING	55
FIGURE 3- 13: PAYMENT BILL	56
FIGURE 3-14: LOGIN STATE CHART	57
FIGURE 3- 15: STATE CHART DIAGRAM FOR REGISTER VEHICLE	58
FIGURE 3- 16: STATE CHART DIAGRAM FOR REGISTER VEHICLE	59
FIGURE 8- 17: COMPONENT DIAGRAMS	60
FIGURE 3- 18:DEPLOYMENT DIAGRAM	61
FIGURE 4- 1: SUBSYSTEM DECOMPOSITION	65
FIGURE 4- 2: HARDWARE AND SOFTWARE MAPPING	66
FIGURE 4- 3: PERSISTENT DATA MANAGEMENT	67

List of abbreviations

Bolo	Inspection certificate
DB	Database
FT	Functional test
GUI	Graphic User Interface
HTML	Hyper Text Markup Language
IT	Integration test
JDBC	Java database connector
JSP	Java server page
Libre	Ownership certification
MVC	Model view control
P2P	Peer to Peer
PC	Personal Computer
SDD	Software Design Document
SRS	Software Requirements Specification
SSL	Secure socket layer
TC	Test Case
TM&TS	Transport monitoring and traffic system
UC	Use Case
UI	User Interface
URL	Uniform Resource Location
UT	Unit test

CHAPTER ONE

1. INTRODUCTION

Ethiopia has experienced high rates of road traffic accidents, as the road is the major means of transportation. From 2001/02-2004/5, the traffic accident death rate was in the range of 129 and 145 per ten thousand motor vehicles. Furthermore, Ethiopia has a relatively high accident records despite having low road network density and vehicle ownerships. From this, over 90% of the world's fatalities on the roads occur in low-income and middle income countries, which have only 48% of the world's registered vehicles. The highest burden of injuries and fatalities in developing countries are shared among pedestrians and passengers of buses and minibuses. For those in their productive years, or whose ages are between 15-29 years, road traffic injury is expected to be the fifth leading cause of all deaths by 2030[1] .

The major causes of the road traffic accidents are inappropriate behavior of driver, incompetent drivers, too old vehicles, and poor control and monitoring system. The current transport controlling system is not efficient and poor as it has to be coordinated with field traffic police.

The system we proposed will fill this gap and enhance previous monitoring system. The proposed system will enable Adama Transport Agency to easily monitor, all the process. Moreover; it is designed to collaborate field traffic police.

Adama Transport Agency carries out the monitoring and controlling process in coordination with traffic police. However; the existing system does not have integration and mechanism to share information between them. The gap will result in increase the number of incompetent, fake driving license user, allow uninspected vehicles and it makes charging case complicate.

The main reason that forces us to select this title is to adapt web based system that improve the customer service, transport traffic process, administration, manage license issuing process and regulate traffic. The proposed system is web based transport monitoring and road traffic system with intuitive user interface which will enable them to browse, search, get information and help decision making. It can also be accessed by intended field officer including traffic police.

1.1 Background of the study

Adama Transport Agency was established in 2003 E.C as a result of proclamation 163/2003. Since then, Adama Transport has been working to provide safe, quality transportation service. In 2006 E.C, the number of traveler that benefit from service increased to 8,393,784.

One of the primary responsibilities of Adam Transport Agency is to facilitate safe road traffic. To assure this it has been delivering workshops for different group of the community. As result the number of Road Traffic Accident has dropped by 35% in 2006E.C when compared with 2005E.C.

In response to generate the government revenue, Adama Transport Agency is working to indentify and collect income source in a responsible and legal manner. Hence at the end of 2006.EC the agency has generated income of 14,767,371.18 birr.

In Adam city, there two government agency that are responsible for vehicle registration and control. They are Adama Transport Agency and internal revenue. The usual practice is for an owner to visit these two for the necessary payment, data collection and issuance of necessary documents and materials such as plate numbers.

Internal revenue collects fees for new automobile licenses and registration from owners. They may request more documents such as custom papers, purchase receipt, or a change of owner certificate on automobile. Every automobile within the country must be registered under a state and a local government before a license plate is issued.

In this research work, interaction will be created on the computerization and implementation of vehicle licensing authority.

1.2 Statement of the problem

The need to provide efficient traffic penalty or charging services, especially in Adama towns, has become a real challenge for city administrators and traffic officers. The problem needs attention as the number of traffic is increasing rapidly and also the town is becoming hub to different cities.

Adama transport agency and traffic police are working together to provide and facilitate transport for Adama town residence and travelers. And they assure the safe transport system as the road accident is one of the national threats.

Based on our analysis on selected transport agency is using MS-access database for recording only. The current system does not have centralized or server based database. They use this database for keeping records and use manual system for giving service for intended users. Relaying on such service system makes very difficult to achieve the National Road Safety Strategy and Program of Vision 2020

The existing system is based on Ms- access 2007 system used for recording charge cases . The current system record status of driver whenever penalized by traffic police. When a drive is charged , they come to agency to pay for charge case. The driver register his is payment case after payment.

They should pay fee in given time that is 48 hours. If the driver do not pay in a given time , she/he should be penalized for delay. So this operation have many problem for instance the driver, sometime wait for long queue to pay.

If the traffic police want to check the driver license validity, it will be difficult to check whenever and wherever. Even though , the license Id has barcode, the system is not accessible by field office and traffic police.

The basic problems facing Adama transport are the ever increasing number of customers coming for one service or the other, the agency can no longer handle with the demands of their customers without much delay with their present manual mode of operation since application forms have to pass from agency officers to customers before registration, issuance or renewal of driving license, libre and bolo. Currently customer have to wait for three days before their demands are met.

The current operation is slow to find and process driver's, owner's and vehicle's files, and difficult to provide the necessary report for manager, stakeholder as needed.

The current problems are summarized as follow:

- It is error prone as it mainly depends on the police traffic. For instance, the police traffic may forget to return the manually filled form on time.
- It lacks flexibility in data retrieval and report generation. The searching of information about the previously faulty vehicles and drivers is not possible as the police traffic don't access the pre-history of the vehicles and driver. Besides the report generation like how many accidents took place in this month, for instance, is difficult.
- A lot of information is filled redundantly for different organization which is time wasting and error prone.
- There is no centralized system available; it is difficult to keep track of repeated offenders.
- Lack of integrated system to deliver the service
- Huge physical document storage area needed makes the retrieval process bulky
- Redundant storage of data
- Lack of proper, accurate and concise information about the vehicle owner
Poor performance of the system during information retrieval due to inefficient storage of data.

1.3 Objectives of the project

1.3.1 General objective

The main objective of the project is to develop web based transport monitoring and road traffic system for Adam transport agency.

1.3.2 Specific objective

To achieve the above general objective the following are specific objectives:-

- To convert manual paper based forms and reports to computerize.
- To design system that a way for web based register driver ,owner and vehicle detail information , monitor charge , payment and issued license certificate from the different daily activities

- To design a database that will store all data and uses stored information to make some desired decision and Ensure data accuracy and Reduce redundancy
- To design user friendliness and interactive that is more attractive and each for users
- To develop web application that can display or view information from database, issue bolo, issue liber and generate report, that minimize time needed for the various processing and Provide update information for user who are uses this system.
- Guarantee security of the system by basing on the users' privileges. Security of data, Increases accountability

1.4 Scope and limitations of the project

The scope of this project is to develop the web based transport monitoring and traffic system that perform all activities of Adam Transport Agency. The aim of the project is to design web based system which perform the following tasks:

- Record drivers information in detail
- Record vehicles information in detail.
- Record vehicle's owner information in detail
- Provide updated information for intended users
- Search, delete ,update and generate reports on relevant data
- Restrict an authorized access
- Can be browsed through mobile phone
- Handle charging process
- Process and issue drivers license
- Easily collaborate with traffic police

It didn't consider the part that connects it with police commission database constraints. The time constraints can hinder the successful completion of the project as the project needs more time to analyze.

1.5 Significance of the project

This project has significance in different perspective. From the user point which includes Adama transport agency, traffic police station, Drivers, vehicles ownerships and vehicles. This project is to change current system to the web based system which is aimed to reduce information retrieve time, to make easy charging process, to access traffic information, update

vehicle information and process driver's license. Moreover its make the Traffic penalty system more efficient.

The system is significant for the user because of the below mentioned reasons:

- More effective and efficient
- Less cost
- Easy to use or operate
- Feasible , technician and economical
- Reduce the time to perform provide services.
- Reduce errors made both client side and user.
- Different documents in different functional department are being stored in integrated database.
- Increases the effectiveness of communication with stakeholders
- It Increases the job satisfaction of the employees who works on the system.
- It improves speed, efficiency, flexibility, reliability, scalability and secured service delivery.

1.6 Methodologies of the project

We used Object Oriented methodology to develop our system because it is a popular technician approach to analyzing, designing and implement an application, system, or business by applying the object-oriented paradigm and visual modeling throughout the development life cycles to foster better stakeholder communication and product quality. We have selected Object Oriented Analysis since it has so many advantages and which can make our system more effective. The advantages of Object Oriented Analysis are:

- The ability to tackle more achieving problem domains.
- Increased consistency among analysis, design and programming activities.
- Common design and coding methodologies, including Object-Oriented Design (OOD), Design Patterns, Refactoring, and the Unified Modeling Language (UML)
- Explicit representation of commonality among system components.
- Reusability of analysis, design and programming results.

- Increased consistency among all the models developed during Object Oriented Analysis, design and implementation.

1.6.1. Data Collection Methodology

To get the necessary data for the effective development of the system we used different data collection methodologies in different context. Among the different methodologies we used the following methods-

- Interviews
- Observation
- Document analysis

I. interviews

An interview is a planned meeting during which you obtain information from another person. The personal interview is often the preferred information gathering technique when developing business and user requirements.

Interviews were carried out with officers of Adama Transport Agency and Traffic police.

II. Observation

Observation is a technique that enables the analyst to view how processes and activities are being done in the context of the business. This additional perspective can give a better understanding of system procedures. It is sometimes worthwhile to read procedure manuals to find out how things should be done.

We tried to observe the actual operations of the existing system in Adama transport agency and interaction between different departments within the agency.

III. Document analysis

Reviewing documents is the process of searching, finding and extracting information from documents which have been created and used by the organization you are trying to automate.

1.6.2 System analysis design Methodology

1.6.3 System development tools

In this project the following development tools are used.

- Mysql 5.1 to design data base and the used as data base server
- Microsoft vision 2007: for designing UML diagram associated with the project
- Microsoft Office 2007: for documentation
- NetBeans 8.0.2 for designing web interface
- Enterprise Architecture 7.1 for designing UML diagram associated with the project
- Glassfish server 4.1 for hosted application

1.7 Feasibility study

There are different feasibility requirements that are expected from any project so this project is feasible in terms of the following

1.7.1 Operational feasibility

Operational feasibility is the measure of how well the proposed solution of the problems will work in the organization context. The proposed system which we will develop is operationally feasible in that it improve the current operation of the Adama Transport Agency.

1.7.2 Technician feasibility

Technician feasibility is the measure of practicality of the specific technician solution and the availability of technician resources and expertise. This system is to be utilized by any users of the organization as they are authenticated to use. The developer teams have better knowledge on developing system and give the necessary support to user starting from the development to the maintenance of the system if deployed.

1.7.3 Economic feasibility

Economic feasibility is the measure of cost effectiveness of the project. As per our analysis of the cost benefit analysis the existing system is not feasible with different perspective so we develop a system which reduces the current cost and improve profitability.

Intangible benefits: The following worksheet lists the intangible benefits associated with the project.

Table 1-1: Intangible cost benefits worksheet table

INTANGIBLE BENEFITS WORKSHEET
1.Increase members Morale
2.Reduce Resource and labor Consumption
3.Increase Management flexibility
4. Provides More timely information.
5. Provides more secure environment.

1.7.4 Schedule feasibility

Schedule feasibility is making sure that whether time schedule of the project met with the completion day of the project. We the developer team tries to complete the project based on the project schedule.

CHAPTER TWO

2. REVIEW OF RELATED LITERATURE

2.1 Transport and Development

A major concern in the past three or four decades has been the question of the development of nations with low levels of production and income. As a consequence, academics and governments have given much attention to the factors that enhance or retard development. Many people would agree that improved transportation plays a key role in economic development. This indeed is one of the few general truths which it is possible to derive from the study of economic history.

Transport is one of the basic ingredients in the operation of an economic system, as it is interwoven with numerous activities and facilitates exchange and interaction. As far back the eighteenth century, Adam Smith noted the beneficial impact of transport on the economic performance and growth of nations. Transport stimulating agricultural production, low levels of regional inequality by opening up remote areas, and as a result induces better territorial integration.

In the context of an agricultural economy, such as Ethiopia's the provision of better and more extensive transport system should, therefore, bring about higher marketed out put levels, lower crop losses from storage problems and the farm collection and distribution of farm produce.

Transport has a necessary role for economic growth and social development of a country especially in the early and middle stages of development. In this period demand for transport tends to grow faster than GDP growth and calls for huge investments before the country is able to provide all the necessary resources of capital. The transport sector accounts for an average of 5-10% of GDP in developing countries and usually calls for a larger share (15-20%) of the total annual investment. In develop countries where transport infrastructure and systems are largely in place, a much more favorable situation exists-the contribution of transport to GDP is higher (7-15%) while the share of transport in total investment is considerably lower(8-10%). [2]

2.1.1 Transport Demand

The need for transportation stems from the interaction among social and economic activities dispersed in space. The diversity of these activities and the complexity of their patterns of interaction result in numerous determinants of transportation needs. Commodities are shipped from place to place for a myriad of reasons stemming from the economic necessities of production and consumption and from the pursuit of economic advantage and gain.

There has been a close link between growth in road freight transport demand and economic growth. Assumption for economic growth has been the basis for many forecasts of freight traffic. For freight transportation the demand is a function of the nature and importance of economic activities (GDP, Commercial surface number of tons) and of modal preferences. The demand for freight transport increased as a result of both growing export/import and the related growth in the internal economy.

2.2 Theoretical Background

Web applications evolved from Web sites or Web systems. [3] The first Web sites, created by Tim Berners-Lee while at CERN (the European Laboratory for Particle Physics), formed a distributed hypermedia system that enabled researchers to have access to documents and information published by fellow researchers, directly from their computers. Documents were accessed and viewed with a piece of software called a browser, a software application that runs on a client computer. With a browser, the user can request documents from other computers on the network and render those documents on the user's display. To view a document, the user must start the browser and enter the name of the document and the name of the host computer where it can be found. The browser sends a request for the document to the host computer. The request is handled by a software application called a Web server, an application usually run as a service, or daemon that monitors network activity on a special port, usually port 80. The browser sends a specially formatted request for a document (Web page) to the Web server through this network port. The Web server receives the request, locates the document on its local file system, and sends it back to the browser.

This Web system is a hypermedia system because the resources in the system are linked to one another. The term Web comes from looking at the system as a set of nodes with interconnecting links. From one viewpoint, it looks like a spider's web. The links provide a means to navigate the resources of the system. Most of the links connect textual documents, but the system can be used to distribute audio, video, and custom data as well. Links make navigation to other documents easy. The user simply clicks a link in the document, and the browser interprets that as a request to load the referenced document or resource in its place.

A Web application builds on and extends a Web system to add business functionality. In its simplest terms, a Web application is a Web system that allows its users to execute business logic with a Web browser. This is not a very precise definition, but most people's conception of a Web application is not, either. There is a subtle distinction between a Web application and a Web site. For the purpose of this book, a Web application is a Web site where user input navigation through the site and data entry affects the state of the business: beyond, of course, access logs and hit counters. In essence, a Web application uses a Web site as the front end to a business application.

2.3 Related Literature

2.3.1 System Architecture survey

2.3.1.1 Tired architecture

Having a review of current web site systems, most of them are based on a tiered architecture, especially the ones used in e-commerce and large web-based applications. The tiered architecture is also called distributed architecture which means that systems are composed of programs running on multiple hosts, such as a web server and a database server. A tier above is one of those hosts but it can be virtual distributed applications running on a single host. So a tier may be a logical, as well as a physical layer of a system. In this project, the word 'tier' refers to logical layers. Generally, there are several tiers that represent different logics such as the presentation logic which shows how information is presented to the clients and the business logic which means the collection of objects and methods that are different from business to business. The advantage of such a design is to allow separation of concerns, which means coding paradigms and required skill sets are both different. Typically, there are following kinds of tired system:

- A) **One-tier systems** can be either applications running on a physical computer or isolated systems which disconnect to a network but can function correctly from logical view. Applications like word processors and image viewers belong to this category. The major advantages of one-tier systems are their simplicity and high performance. Since there is no need to communicate with remote services through a network, the application can have the maximum performance allowed by its hardware.
- B) **Two-tier systems** run on two computers, composed of a client application and a server application. For example, a typical web application developed by CGI runs on a client's web browser and a web server. But as mentioned before, a tier can be a logical layer of a system, so it is possible to have a two-tier application running on one machine with a division of two layers. Thus, there are usually three parts in a two-tier system: a client, a server and a protocol. The role of the protocol, which in most cases is the HTTP protocol, is used to send and request data between the client and the server. The advantage of this architecture is to separate presentation logic from business logic, which means the presentation logic is on the client side, while the business logic is on the server side. So the majority of the data processing can be done on the server and the client just needs to receive and present the data derived from the server side. However, this kind of architecture has little potential for resource sharing, which is a big problem for large database-driven system.
- C) **Three-tier systems** have one more layer and usually represent presentation logic, business logic and data logic which means how to ensure the persistence, security and data respectively. This architecture allows concurrent data access via transactions. It is common to have more than one client to retrieve data simultaneously, so it can avoid the data corruption. For instance, a server requests or stores data in the database server after it receives a request from a client and then it gives information returned from the database back to the client.
- D) **N-tier systems** are those systems having more than three tiers. This type provides great performance capabilities, enhancing the amount of services it can provide and allowing greater maintainability. The disadvantage of an n-tier system is that it is inefficient in most cases, expensive and more complex.

In short, the most widely used architecture is a three-tier type, which offers a separation of presentation logic, business logic and data logic. This architecture is argued and proved as the best solution for a web-based database-driven system. So it is necessary to discuss the available technologies to employ next.

2.3.2 Available Technologies

Since three-tier architecture enhances the separation of presentation logic, business logic and data logic, we can discuss the technologies that could be employed from the view of each tier.

A) Presentation Tier

Presentation tier means how the web pages interact with the clients. Since most of the content rendered to the end users is generated from the database, this project is a dynamic web system. The current available and popular technologies to produce dynamic content in a web page are Active Server Pages, Java Server Pages and Velocity Template Language.

Active Server Pages (ASPs) are the contribution to the development of dynamic pages from Microsoft. This technology has experienced two generations, classic ASPs and recent ASP.NET. Classic ASPs use scripting languages on the server side and have support for VBScript and Jscript which are built in the web pages. New ASP.NET supports any.NET compliant language, such as C#, Java. Since code in ASP.NET is compiled rather than interpreted by the ISA server, it runs faster than classic ASPs. Though both of these technologies are widely used and easy to learn, they are heavily dependent on the Microsoft's server and framework. Because this project is being developed under an open source environment, they are not the proper solutions.

Java Server Pages (JSPs) are another technology provided by Sun for developing web pages that include dynamic content. They enable web developers and designers to rapidly develop and easily maintain, information-rich, dynamic Web pages. Unlike a plain HTML page, a JSP page can change its content based on any number of variable items, including the identity of the user, the user's browser type, information provided by the user, and selections made by the user. [6] A JSP file has a ".jsp" extension and contains both standard markup language elements (e.g. HTML tags) and special JSP elements that allow the server to insert dynamic content in the page

Velocity is one of the Apache Jakarta Projects, which is a Java-based template engine. It permits anyone to use a simple yet powerful template language to reference objects defined in Java code. Velocity pages are actually written in the form of HTML with embedded Velocity code and end with a “.vm” extension.

The most attraction Velocity has is that it fully enhances the Model-View-Controller (MVC) pattern to organize web sites, which is regarded as a clear way for web designers and Java programmers to work in parallel. This means that web page designers can focus solely on creating a nice site, and programmers can focus solely on writing top-notch code. Since Java code is separated by Velocity from the web pages, it is easier to maintain the web site over its lifespan. So Velocity is a valid alternative to JSPs. However, JSPs plus JavaBeans can also apply MVC design pattern to web sites.

Considering the valuable experiences we have in JSPs and Java technologies and the limited time on the development, JSPs are enough to carry out the presentation tier with the proper cooperation with Java Beans and Servlets. So JSP is preferred technology to carry out the presentation tier in this project.

B) Business Tier

The Business Tier shows us the collection of objects and methods that are different from business to business, which means how to represent your business logic to be performed in a certain way in your project. In this tier, the available technologies are JavaBeans and Java Servlets.

- JavaBeans

JavaBeans (Enterprise Java Beans) are a platform-independent architecture for Java, which is one of the most interesting and powerful ways to cooperate with JSPs. JavaBeans are reusable Java classes whose methods and variables follow specific naming conventions to give them added abilities.

- Java Servlets

The Java Servlet API allows a software developer to add dynamic content to a web server using the Java platform. It has the ability to maintain state after many server transactions, which is done using HTTP Cookies, session variables or URL rewriting. Java Servlets is a generic server extension, which is actually a Java class that can be loaded dynamically to expand the functionality of a server. Since servlets are written in the highly portable Java language and follow a standard framework, they provide a means to create sophisticated server extensions in a server and operating system independent way.

C) Data Tier

Today, the most popular and efficient way to store data in a web application is using database or XML.

2.3.3 Web application Model

Since this project is actually a web-based application, it is important to review some popular and useful web application models so that an appropriate model can be chosen.

1. The J2EE Model

The Java 2 Enterprise Edition (J2EE) Model is actually a three tiered web application model, which includes Client Tier, Middle Tier and Enterprise Information System (EIS) Tier. J2EE is a compilation of various Java APIs, including JSP, Java Servlets, Enterprise JavaBeans (EJB) and JDBC. These APIs stand for different component technologies, and are managed by what the J2EE documents name containers.

A web container provides the runtime environment for servlets and JSP components, translating requests and responses into standard Java objects. Similarly, an EJB container manages EJB components. One important thing we should keep in mind is that an EJB is quite different from a Java Bean. On the one hand, a JavaBeans component is a regular Java class, which follows a few simple naming conventions and can be used by any other Java class. On the other hand, an Enterprise JavaBean must be developed in compliance with a whole set of strict rules and works only in the environment provided by an EJB container. Furthermore, components in the two types of containers can use other APIs (e.g. JDBC) to access databases.

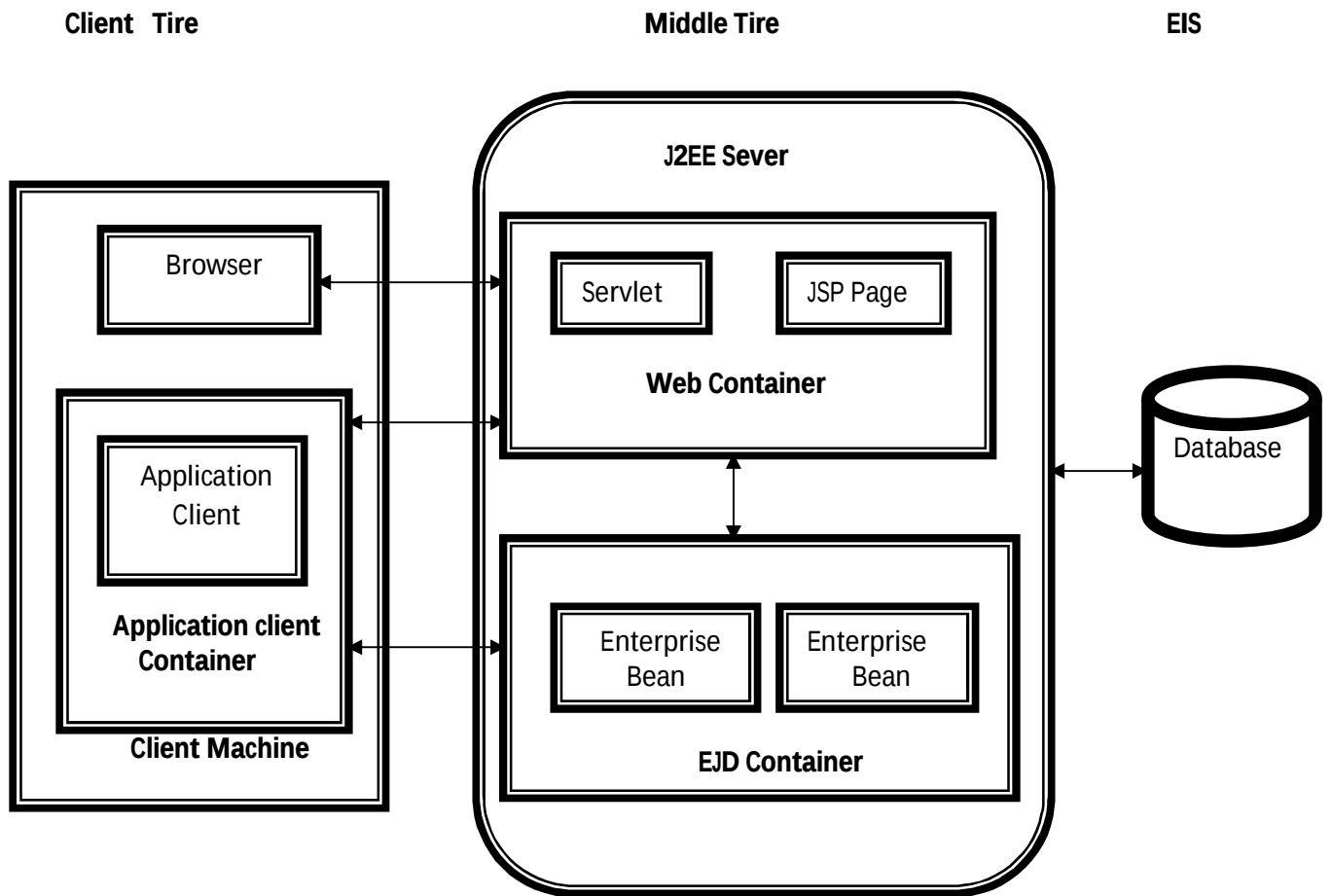


FIGURE 2: 1 J2EE MODEL

Figure 2.1 Shows an overview of all the components and their relationship, which was introduced by Pawlan (2001) [4].

2. The MVC Design Model

The Model-View-Controller (MVC) design model was first introduced by Xerox in a number of papers published in the late 1980s in conjunction with the Smalltalk language. And since then it has been widely used in all popular programming languages because of its advantages on reducing effort needed in software development. As its name states, this design pattern is based on the three entities, the Model, the View, and the Controller. “The Model represents pure business data and the rules for how the data is displayed or the user interface for modifying the data. The View, on the other hand, knows all about the user interface details. It also knows about

the public Model interface for reading its data, so that it can render it correctly, and it knows about the Controller interface, so it can ask the Controller to modify the Model” [5]. Before MVC, user interface designs tended to mix these entities together. MVC decouples them to increase flexibility and reuse and enforces the separation of responsibilities into different tiers. The figure 2.2 shows a brief view on how the different pieces communicate in a MVC model.

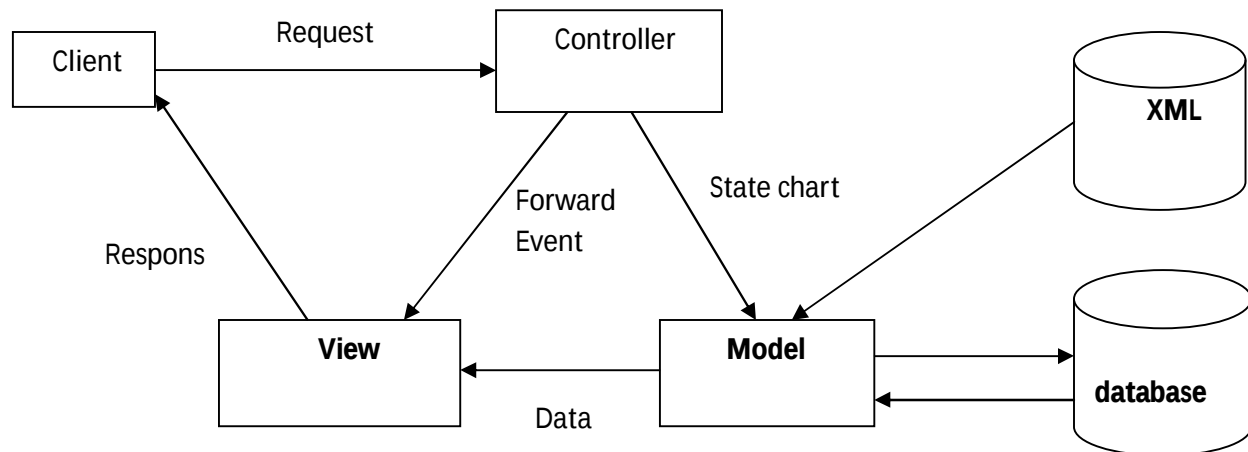


FIGURE 2: 2 MVC MODEL

In practice, there are several ways to assign MVC model to the different types of J2EE components, which depends on the scenario, the types of clients supported, and whether or not EJB is used. The next subsection describes possible architecture scenarios where JSP plays an important role.

2.4 Related Studies

In [6] A web- based online automated motor vehicle registration system is designed to automate the way motor vehicles are been registered. This system capture data on motor vehicles and their owners, facilitate the production of motor vehicle titles, provide motor vehicle related information as needed, and facilitate easy identification by owners name, plate number and chassis number.

An SMS and web based traffic management system [7], has a web base and online system. The system primarily intended to aid traffic police as it enables the traffic police to record case and update cases. The system is also interface with mobile so that a traffic police is updated with SMS. This System has a web bases software and central database. Each traffic box has at least one compute connected to a central database. If any traffic wants to entry case against any vehicle then they can easily entry it from the police box. They do not need to record it manually. If the traffic wants to check the vehicle information then they can easily search it through the vehicle number. The traffic police can also see the previous case records from this software. They can see the driver information through the driving license number also.

In this project a mobile SMS is integrated with database, in such a way that a traffic police can access a central database by sending SMS through mobile.

An SMS and Web based traffic Management system also has offline application. This offline application is a backup of web based application that set side if in case the remote system is not accessible. This project is developed using Ms Access for data base and PHP for design server scripting. The offline system implemented with Visual Basic.

CHAPTER THREE

3. REQUIREMENT ANALYSIS

3.1 Current System

In the current system more operations are performed manually which results working through large document. And some activities are recorded on desktop based database. Currently Adama Transport Agency use MS-Access Data base with is store in a single computer and shared in peer to peer architecture. The other mean to provide and give their services to the customers is through manual paper work such as forms and telephone calls

The existing system provide services and information about vehicles , drivers, ownership of vehicles ,transport traffic status, issuing license, penalty charging of drivers, assigning public transport(vehicles) to different station, informing the fee of public transport and regulation service through the interaction of the following main component:

- Transport service
- Customers(drivers, ownership's vehicles)
- Traffic police

The existing system has been performing its activity in the following:

1. Assuring transport traffic process
2. Assuring transport service process
3. Issuance and renewal of
 - Motor vehicle licenses
 - Local driving license
 - Learners permit
4. Collection of fees emanation from its operation and payment of some into the appropriate subhead of the state government's account
5. Preparing and taking care of statutory Registers of all licenses issued.

Major functions/activities in the existing system like inputs, processes & outputs

- For issue libre and bolo in existing system with their role players.

Input

Activity 1: Ownership information

Activity 2: Vehicle Information

Activity 3: Check Custom clearance

Process

Activity 5: Agency Technician check or information about vehicle history and inspection.

Activity 6: Agency technician check service payment.

Activity 7: Agency technician registers owner and vehicle information file it.

Activity 8: Open and organize file for each owner

Output

Activity 9: Issue bolo and libre for the owners

For issue driver license in existing system with their role players.

Input

Activity 1: Collect trainee driver information

Activity 2: Trainee takes exam

Activity 3: Trainee Pay for service

Process

Activity 4: Agency technician screen exam result.

Activity 5: Agency technician registers detail driver information for those who pass exam.

Activity 6: Open and organize file for each driver who pass exam

Output

Activity 7: Issue driver license with attached photo

Problems with existing system

- Lack of security of data
- Needs more man power
- Time consuming
- Consumes large volume of paper work
- Duplicate data on different computer

- Exposed to errors
- Difficult to identify fake license
- Difficult to access information anywhere easily and quickly.

3.2 Proposed System

3.2.1 Overview

To eliminate those problems of the existing system the proposed system promises to reduce problems by developing web base system that providing updated service with necessary security level.

Advantages of the Proposed System

The system will be designed to provide simple and user friendly environment with minimum cost. Furthermore; the system will offer the following advantage:

- Security of data
- Ensure data accuracy
- Provide real time information for traffic police
- Minimize time needed for the various processing
- Reduce redundancy
- Reduce costs
- User friendliness and interactive.
- Increases accountability

3.2.2 Functional Requirements

The system should provide the following functions:

- Register and Keep record of data in a central database.
- Maintain and record complete information about drivers, ownerships, vehicles and vehicles' associations and charging.
- Issue driver license , libe and Bolo
- Renew license and bolo
- Suspend driver license, vehicle service and return plat no, and LIBRE
- charge cases
- Service and charging fee payment
- Allow to update , delete and backup data

- Search information.
- Generate report

3.2.3 Non Functional Requirements

Non functional requirement describes user visible aspect of the system that is not designed to the functional behavior of the system. They span on a number of issues from user interface to security issue.

In Short non-functional requirements are describe

- ✓ The user interface shall be simple for user on how to use the system.
- ✓ The system shall be attractive and easy to deal with.
- ✓ The system response time for request is not more than 3 second
- ✓ The system should be protected through password. It is only an administrator who has the authority to create user account, change password and delete account.
- ✓ The system must provide backup mechanism.

3.2.3.1 User Interface

This nonfunctional requirement is basically concerned on what kind of web page interface the system should provide or what is the level of expertise of the user. The system may be divided into the following user groups, administrator, technician, field offices, traffic police and finance clerk.

- The system will have graphical user interface
- The system can be easily accessed by user
- Each record stored in the system in a single database server and can access by multiple users.
- The interface should not appear complicated or require extensive training.
- The dialogue with the system should be easy to follow and it should be obvious to users how to navigate their way through various screens.

3.2.3.2 Hardware Consideration Computer

The software designed needed the following hardware for an effective operation of the newly designed system.

1. Pentium IV system

2. At least 256MB of RAM
3. Enhanced keyboard
4. 20GB hard disk space
5. SVGA monitor
6. Mouse

3.2.3.3 Software Consideration

In the non functional requirement we can discuss a lot of software's that are necessary for the system should have.

- Operating system Windows 7 Ultimate edition ,
- MySQL 5.5.8 database server.
- Mozilla Firefox 26 to run the system in the local host
- Window server 2003/2008

3.2.3.4 Performance Characteristics

The system is developed to run in different network level and different protocols. This means that the system will support multi user access the system at the same time. So it should be noted that the performance is kept continuous disregarding the tolerance that will occur on it. Beside that the system should respond the request as quick as possible. It is better to have a good network speed.

- The system is fully operational for many hours per day as needed.
- Efficiency (resource consumption for given load): consumes less resource like time, manpower needed to perform the task.
- Effectiveness: more effective when compared to the manual approach, resulting in fast accomplishment of task with in some amount of time.

3.2.3.5 Error Handling And Extreme Conditions

Since the system exceptions are well handled for invalid system inputs the system operations, the system gives proper warning and information messages.

Reliability is a major characteristic of high-quality software. Software should behave well in nearly every situation. During the development process we try to avoid, detect and remove as many as possible.

Those mechanisms become a part of the basic infrastructure, which is used by nearly every part of the software. Thus the structure of the whole software is influenced by the error-handling mechanism and therefore has to be designed carefully.

3.2.3.6 Security Issue

A system will be dealing with delicate data; the system should be stored in a highly secure manner and should be immune from any hacking attempt.

3.3 Hardware And Software Requirements

To develop this online system for used different system development environments. These are:

- NetBeans IDE 8.0.2
- JSP 2.1
- JavaDevelopment Kit (jdk-8u11-wondows-x64)
- MySQL 5.5.8 for the database.
- GlassFish 4.1 sever
- Java script for some security control activates
- Mozilla Firefox 26 to run the system in the local host
- Enterprise Architecture tool for Design

3.4 Software Engineering And Software Requirement

For the analysis and design of the project we followed, Iterative, Incremental Framework which is a logical extension to spiral model but more formal and rigorous for our project. The framework is divided into four major phases: Inception; Elaboration; Construction and Transition.

3.5 Analysis Activities

We used Unified Modeling Language (UML), which is used in object oriented software engineering. Although typically used in software engineering it is a rich language that can be used to model an application structures, behavior and even business processes. There are 14 UML diagram types to help us model the system. All type of UML state as following Class Diagram, Component Diagram, Deployment Diagram, Object Diagram, Package Diagram, Profile Diagram, Composite Structure Diagram, Use Case Diagram, Activity Diagram, State

Machine Diagram, Sequence Diagram, Communication Diagram, Interaction Overview Diagram and Timing Diagram.

We selected the most necessary UML diagrams such as Use Case diagram, Class Diagram, Component Diagram, Deployment Diagram, State chart Diagram, Sequence Diagram and Activity diagram.

Use Case diagram, Class Diagram, Component Diagram UML, it show the things in a system being modeled. In a more technical term they show different objects in a system. And shows what should happen in a system. They describe how the objects interact with each other to create a functioning system.

3.5.1 Use Case Diagrams

Use Case modeling identifies and describes the system functions by using a tool called Use Cases. Use Cases describe the system functions from the perspective of external users and in a manner and terminology they understand. To accurately and thoroughly accomplish this demands a high level of user involvement and a subject matter expert who is knowledgeable about the business process or event.

Actors

Use Cases are external entities that interact with the system. Use Cases are initiated or triggered by actors. An actor initiates system activities for the purpose of completing some task. An actor is a person, or external system that place a role in one or more interactions with yours system. A relationship between actors and classes are indicated in Use Case diagram, a relationship exists where ever an actor is involved with an interaction described by a Use Case a relationship can also exist between Use Cases.

In the diagram shown below Use Case are derived from functional requirement. During the interview and discussion with Adama transport agency officers, observation of BPR and operational manual. We come to identify the following actors and Use Cases. According to our analysis there are sixteen Use Case and seven primary actors. This Use Case and actors does not describe all function within the agency however it is sufficient for this project.

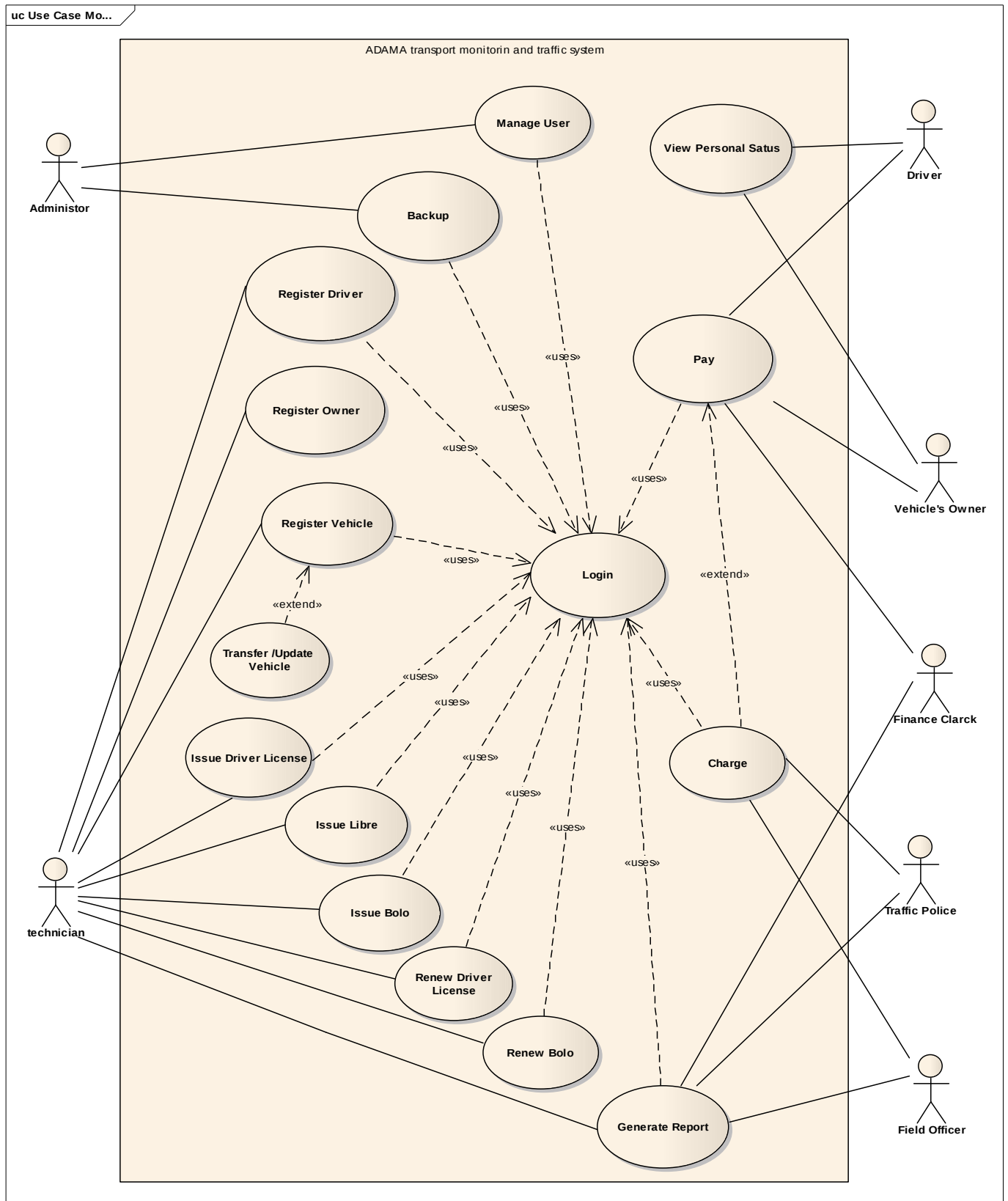


FIGURE 3-1 USE CASE DIAGRAM

3.5.1.1 Descriptions of Actors

Administrator: A person who is responsible for system to manage user and backup.

Field officer: A person who is responsible for assure transport traffic processing. She/he can make offence and charge those who drivers and ownership pass the role of traffic and generates different kinds of report.

Technician: A person who is responsible to issue or give driver's license, Libre and Bolo for Adama transport agency customer.

Traffic Police: Traffic polices are in charge of controlling the road traffic and also responsible for reporting of accidents to the central base station.

Financial Clerk: A person who is responsible to receive fee from those who charged and give receipt to them .he/she open account.

Vehicle owner: A People who have vehicle. He/she need Libre and bolo from Adama transport agency.

Driver: A person who can drive a vehicle. He/ She need driver license from Adama transport agency.

Description Of Use Cases

Table 3- 1: Use Case for Manage User

Name	Manage user
User Case Id	UC-1
Actor	Administrator
Purpose	To create, update, delete , search and control user
Precondition	The administrator must be registered
Basic course of action	1. The Administrator initiates login 2. The system display login form 3. The Administrator enters user name and password and login 4. The system validate the user name and password 5. The system displays the messages whether successful or not. 6. The system displays the administrator manage page 7. The administrator enter the required input to create, update , delete and search and control user role 8. The system checks the validate of the input data 9. The system displays the messages whether successful or not. 10. The system create ,update, delete search to/from database
Post condition	New users are created and registered to the Database.
Alternative action	A1: Step5, If the administrator enters wrong password and user name, The system goes back to Step1,2,3 and 4 A2: Step9, The system may display an error message in case of any database transaction error or invalid input and the administrator may repeat step 7, step 8 or may exit from the system.
Exit condition	- User is registered , updated, deleted or searched - The administrator log out

Table 3- 2: Use Case for login

Name	Login
Use Case Id	UC-2
Actor	Administrator, Technician, Financial clerk, Traffic police, Field officer
Purpose	A user logs into the system
Precondition	First registered and having account name and password.
Basic course of action	1. The all users initiates login. 2. The system display login form 3. The all User enter user name and password 4. The system checks the validate of the user name and password 5. The system displays the messages whether successful or not. 6. The system displays the users of their own page
Post condition	Users are Logged
Alternative action	A1: Step5,if user enter invalid user name and password, the administrator go to back step 3
Exit condition	The users log out

Table 3-3: Use Case for Register Vehicle

Name	Register vehicle
Use Case ID	UC-3
Actor	Technician
Purpose	To register new vehicle
Precondition	new arrival vehicle and does not exist in a system
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password. 4. The system checks the validate of the user name and password 5. The system displays the messages whether correct or not. 6. The system display a vehicle registration page 7. The technician enters the details of the vehicle information (fill the forms). 8. The system verifies, and store the vehicle information in to the database
Post condition	New vehicle is recorded and registered to the central base.
Alternative action	A1: Step5, if user enter invalid user name and password, the technician go to back step 3 A2: step 8, if the technician may enter invalid input, technician repeat step 7
Exit condition	- Vehicle registered and log out

Table 3-4: Use Case for Register vehicle's ownership

Name	Register vehicle's owner
User Case ID	UC-4
Actor	Technician
Purpose	To register a ownership information who have vehicle
Precondition	The owner must be new or do not exist in a system
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password. 4. The system checks the validate of the user name and password 5. The system displays the messages whether correct or not. 6. The system display a owner registration page 7. The technician enters the details of the owner information 8. The system verifies, and store the register owner information in to the database
Post condition	New owner are recorded and registered to the central base.
Alternative action	A1: Step5, if user enter invalid user name and password, the technician go to back step 3 A2: step8, if the technician may enter invalid input, technician repeat step 7
Exit condition	- Owner registered and log out

Table 3-5: Use Case for Register trainee driver

Name	Register driver
Use Case ID	UC-5
Actor	Technician
Purpose	To register an driver to give their license
Precondition	The drive must be passed the driver exam.
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password. 4. The system checks the validate of the user name and password 5. The system displays the messages whether correct or not. 6. The system display a driver registration page 7. The technician enters the details of the driver information 8. The system verifies, and store the register driver information in to the database
Post condition:	New driver are recorded and registered to the central base.
Alternative action	A1: Step5, if user enter invalid user name and password, the technician go to back step 3 A2: Step 8, if the technician may enter invalid input, technician repeat step 7
Exit condition	The driver registered and log out

Table 3-6: Use Case for Issue Driver License

Name	Issue Driver License
Use Case ID	UC-6
Actor	Technician
Purpose	To issue license for trainees
Precondition	The trainees must pass the exam to be driver and pay service fee
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password. 4. The system checks the validate of the user name and password 5. The system displays the messages whether correct or not. 6. The system display an issue driver license page 7. The system check the exam result greater than 85 8. The system Issue driver license information
Post condition	New license is stored and printed.
Alternative action	A1: Step5, If the technician enter invalid input user name and password technician may repeat step 3
Exit condition	- driver license is issued - Log out

Table3-7: Use Case for Issue Libre

Name	Issue Libre
Use Case ID	UC-7
Actor	Technician
Purpose	To issue assuring vehicle ownership license
Precondition	The owner and vehicle must registered in a system and pay service fee
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password 4. The system checks the validate of the user name and password 5. The system displays the messages whether correct or not. 6. The system display a Libre page 7. The system display the Libre information
Post condition	Libre is stored and printed
Alternative action	A1: Step5, If the technician enter invalid input user name and password technician may repeat step 3
Exit condition	- Issued Libre - Log out

Table 3-8: Use Case for Issue Bolo

Name	Issue Bolo
Use Case ID	UC-8
Actor	Technician
Purpose	To issue Bolo for vehicle
Precondition	The vehicle must be inspected by technician and pay service fee
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password 4. The system checks the validate of the user name and password 5. The system displays the messages whether correct or not. 6. The system display an issue Bolo page 7. The system display the Bolo information
Post condition	New Bolo is issued and printed
Alternative action	A1: Step 5, If the technician enter invalid input user name and password technician may repeat step 3
Exit condition	- Issued bolo and - log out

Table 3-9: Use Case for Transfer or update vehicle and owner

Name	Transfer or update
Use Case	UC-9
Actor	Technician
Purpose	To transfer the ownership, update service, update body and update location
Precondition	Owner and vehicle must registered and Pay service fee
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password 4. The system checks the validate of the user name and password validate 5. The system displays the messages whether correct or not. 6. The system display a transfer page 7. The technician select type of transfer like as change service, body , location work place and transfer ownership 8. The system check the registration statue of owners and vehicles, and verify services fee 9. The system display update information
Post condition	Vehicle and owner are transferred and print report
Alternative action	A1: Step 5, If the technician enter invalid input user name and password technician may repeat step 3 A2: Step 8, If the system check owner pay service fee or occur in case of any database transaction error or invalid input and technician may repeat step 7.
Exit condition	- Vehicle is transferred or updated - Log out

Table 3-10: Use Case Renewal Bolo

Name	Renew Bolo
Use Case ID	UC-10
Actor	Technician
Purpose	To renew bolo annually
Precondition	Bolo must issued previous and pay for renew bolo fee
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password 4. The system checks the validate of the user name and password validate 5. The system displays the messages whether correct or not 6. The system display a renewal page 7. The technician enters the details information bolo (fill the forms). 8. The system verifies and check the input data 9. The system display renewal bolo information and generate report.
Post condition	Issued renewal bolo
Alternative action	A1: Step 5, If the technician enter invalid input user name and password technician may repeat step 3 A2: Step 8, If the system check service pay fee or occur in case of any database transaction error or invalid input and technician repeat step 7.
Exit condition	- Renewed - Log out

Table 3-11: Use Case for Renewal Driver License

Name	Renew Driver license
Use Case ID	UC-11
Actor	Technician
Purpose	To renew driver license
Precondition	Driver license must registered and pay fee for renew
Basic course of action	1. The Technician initiates login 2. The system display a login form 3. Technician enters user name and password 4. The system checks the validate of the user name and password validate 5. The system displays the messages whether correct or not 6. The system display a renewal page 7. The technician enters the details information driver license (fill the forms). 8. The system verifies and check the renewal input data 9. The system display renewal driver license information and generate report.
Post condition	Issued renewal driver license
Alternative action	A1: Step 5, If the technician enter invalid input user name and password technician may repeat step 3 A2: Step 8, If the system check a service pay fee or an occur in case of any database transaction error or invalid input and technician repeat step 7.
Exit condition	- Renewed - Log out

Table 3-12: Use Case for Generate Report

Name	Generate report
Use Case ID	UC-12
Actor	Technician ,Finance clerk, Field officer ,traffic police
Purpose	To summarize or revise all information
Precondition	When required or needed information
Basic course of action	1. The technician login. 2. The system display a search , filter report page 3. The technician enters value on search by ID the details information 4. The system check report input data or the ID found or not 5. The system generates report.
Post condition	Print report
Alternative action	A1: Step4, if the technician enter ID invalid input, he go back to step3
Exit condition	- Report is generated - Log out

Table 3-13: Use Case for Charging

Name	Charge
Use Case ID	UC-13
Actor	Traffic police , field officer
Purpose	To offense charge
Precondition	Driver must make mistake or offense
Basic course of action	1. Traffic police and field officer login 2. The system display a charge page 3. The Traffic police and field officer enter the Driver ID to search to charge 4. The system display a driver charge information 5. The Traffic police and field officer select the charge code 6. The system generates charge.
Post condition	The charge are stored , view and print
Exit condition	- Charged - Log out

Table 3-14: Use Case for payment

Name	Payment
Use Case ID	UC-14
Actor	Driver , owner , finance clerk
Purpose	To pay service and charge fees
Precondition	Renew, charge case, and new service
Basic course of action	1. The finance clerk login 2. The system display a payment page 3. The finance clerk enters the owner ID or driver ID to search for payment. 4. The system displays the owner and driver charged information. 5. The finance clerk select paid option button after receive fee 6. The system update the driver charge statue and display message the driver paid and now he/she is free. 7. The driver is not pay in given two days 8. The system add penalty 10 % each 5 days and after 5 days the system suspend licenses.
Post condition	The charges are paid , view and print bill
Alternative action	
Exit condition	Log out

3.5.2 Class Diagrams

A class diagram depicts classes and their interrelationships. Class diagrams describe the structure of the system in terms of classes and objects. Classes are abstractions that specify the attributes and behavior of a set of objects. Objects are entities that encapsulate state and behavior.

We use interviews to collect all the concepts and idea that agency recognize. The project team also used brainstorming suggestion for the concept. Once brainstorming is completed we

discussed and justified on each suggestion. Some concepts are extracted from physical or tangible objects, transaction, and role of people, which are specified on requirement.

Class diagram has boxes including three compartments because it is a standard convention depicts classes and objects. The top compartment displays the name of the class or object. The center compartment displays its attributes, and the bottom compartment displays its operations. The attribute and operation compartment can be omitted for clarity. Object names are underlined to indicate that they are instances.

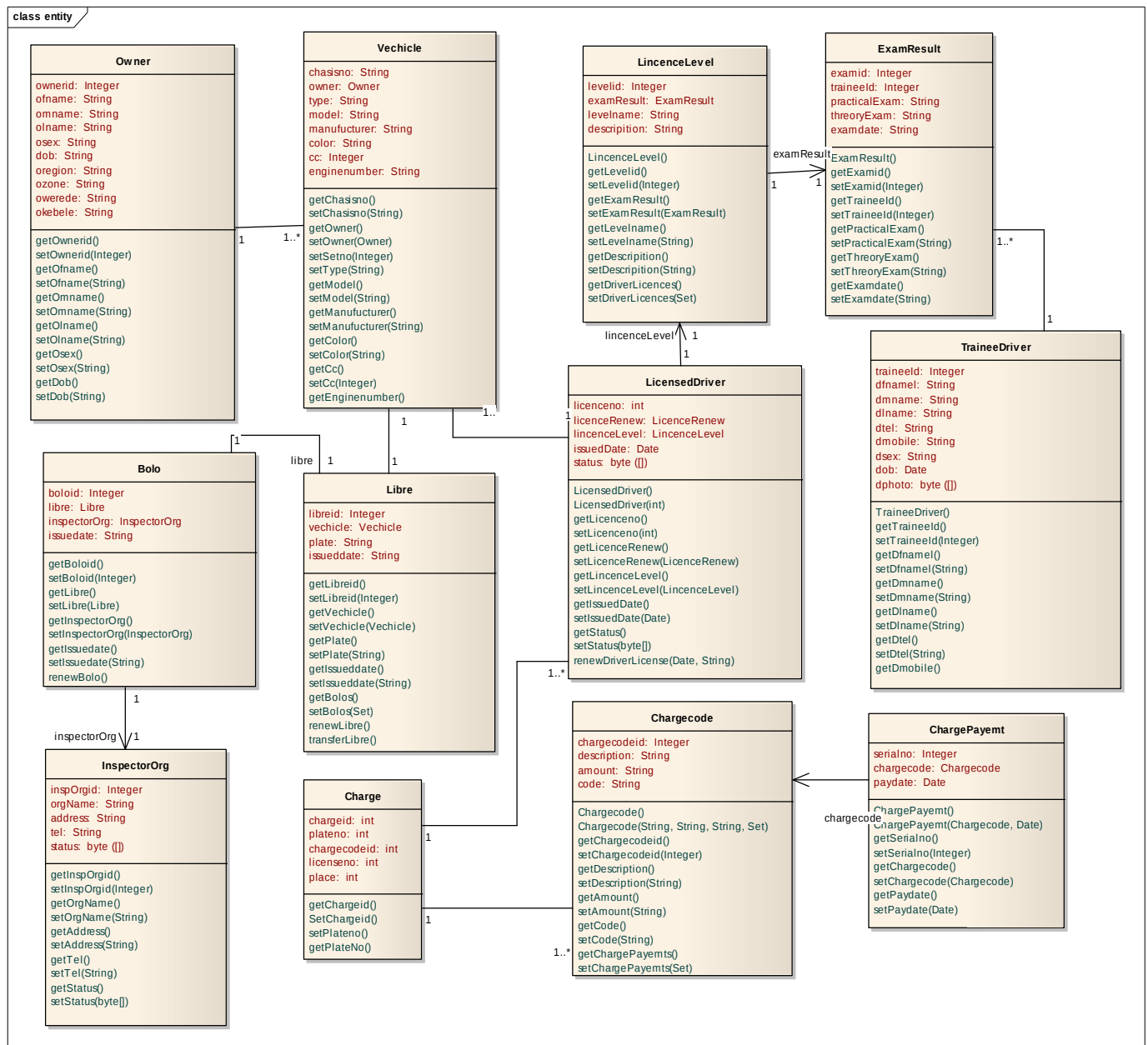


FIGURE 3-2: CLASS DIAGRAM

3.5.3 Sequence Diagram

A sequence diagram describes behavior as a sequence of messages exchanged among a set of objects. A sequence ties a Use Case with objects it shows how the behavior of a Use Case are distributing among its participating object.

A Use Case only shows how requests from the actors are accepted and return result to actor and not concerned with how it fulfills the request. So the sequence at analysis used to helps to show how requests are fulfilled.

The number of sequence diagram determined b Use Case of primary actors. So we have 9 sequence diagrams.

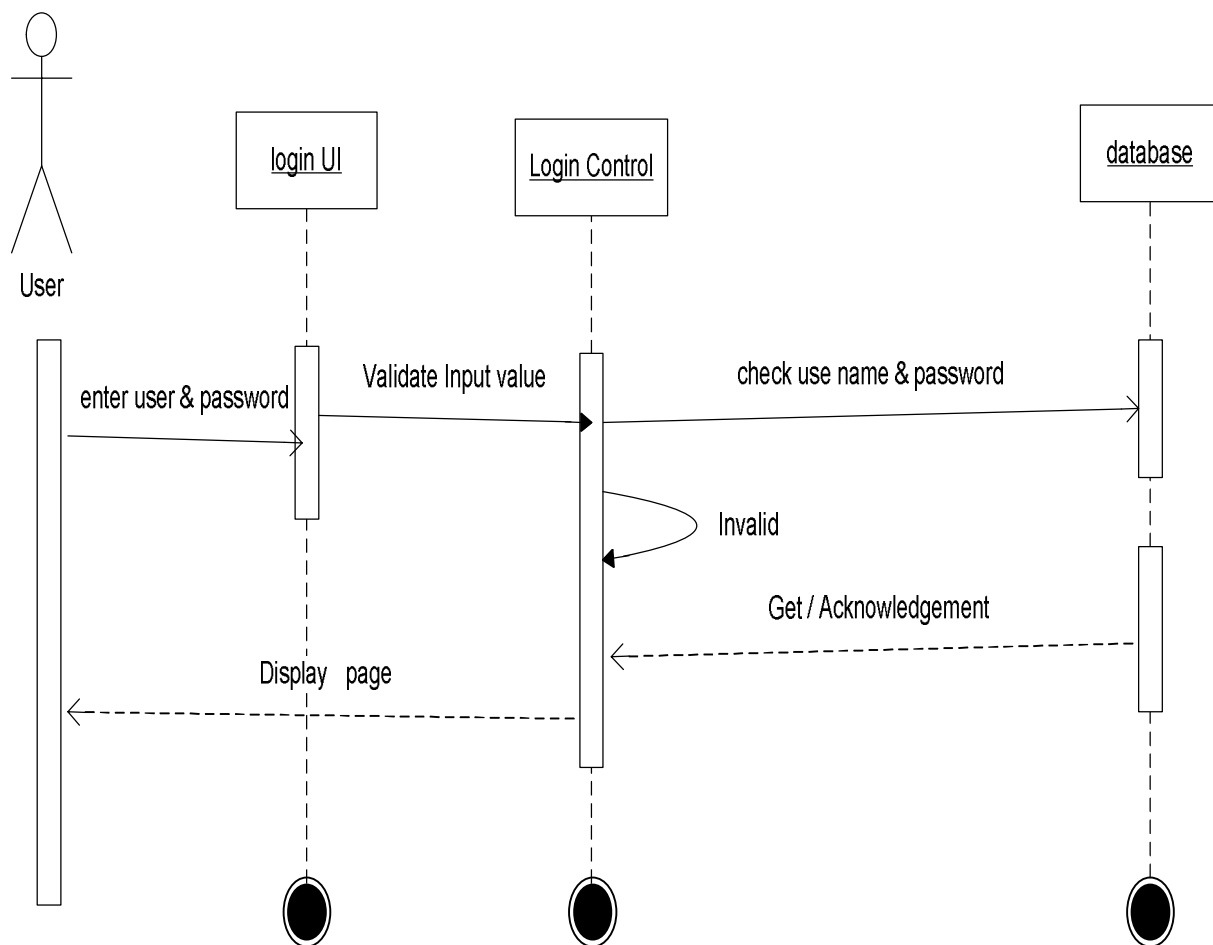


FIGURE 3- 2: LOGIN SEQUENCE DIAGRAM

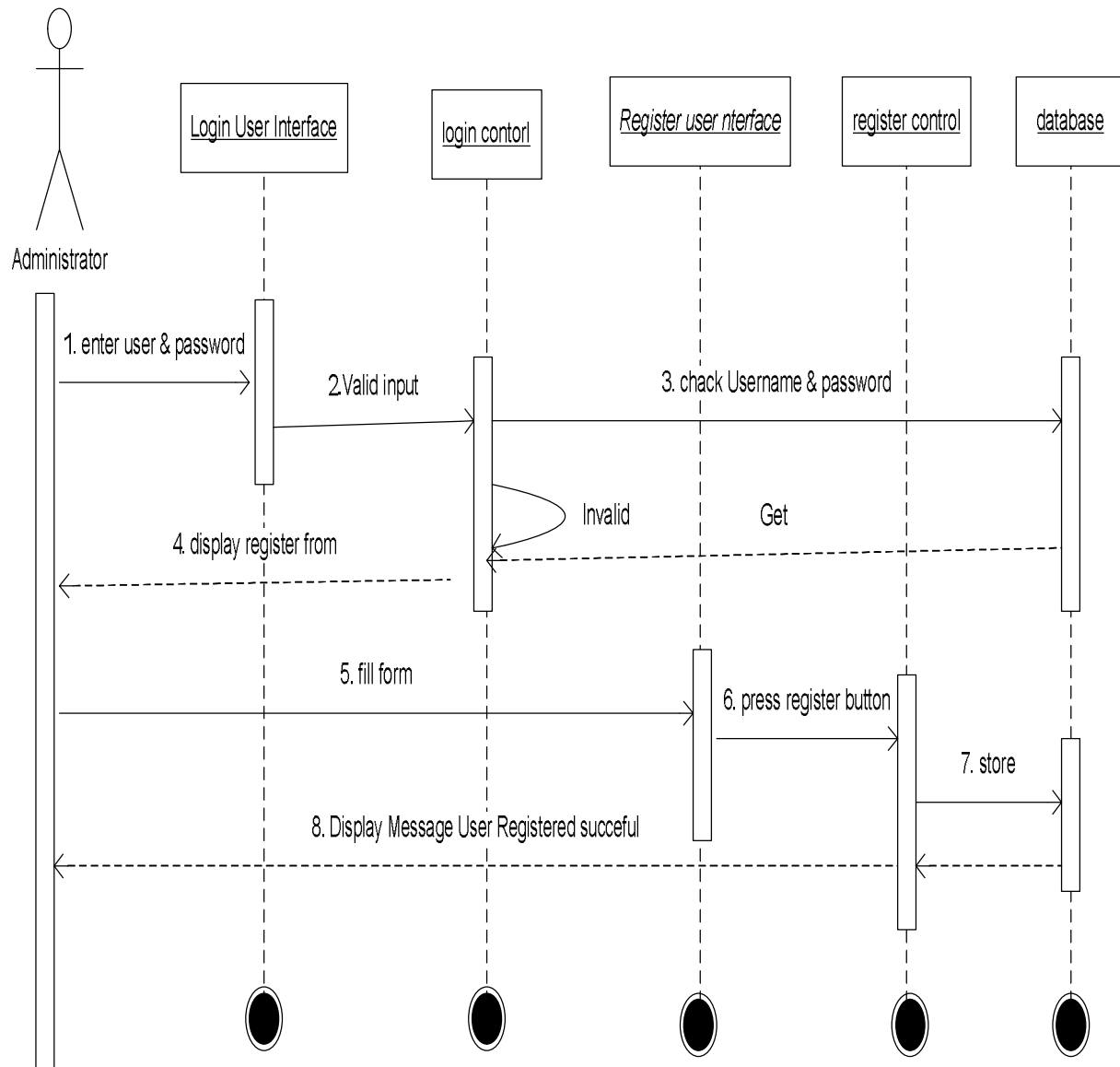


FIGURE 3- 3:SEQUENCE DIAGRAM FOR REGISTER USER

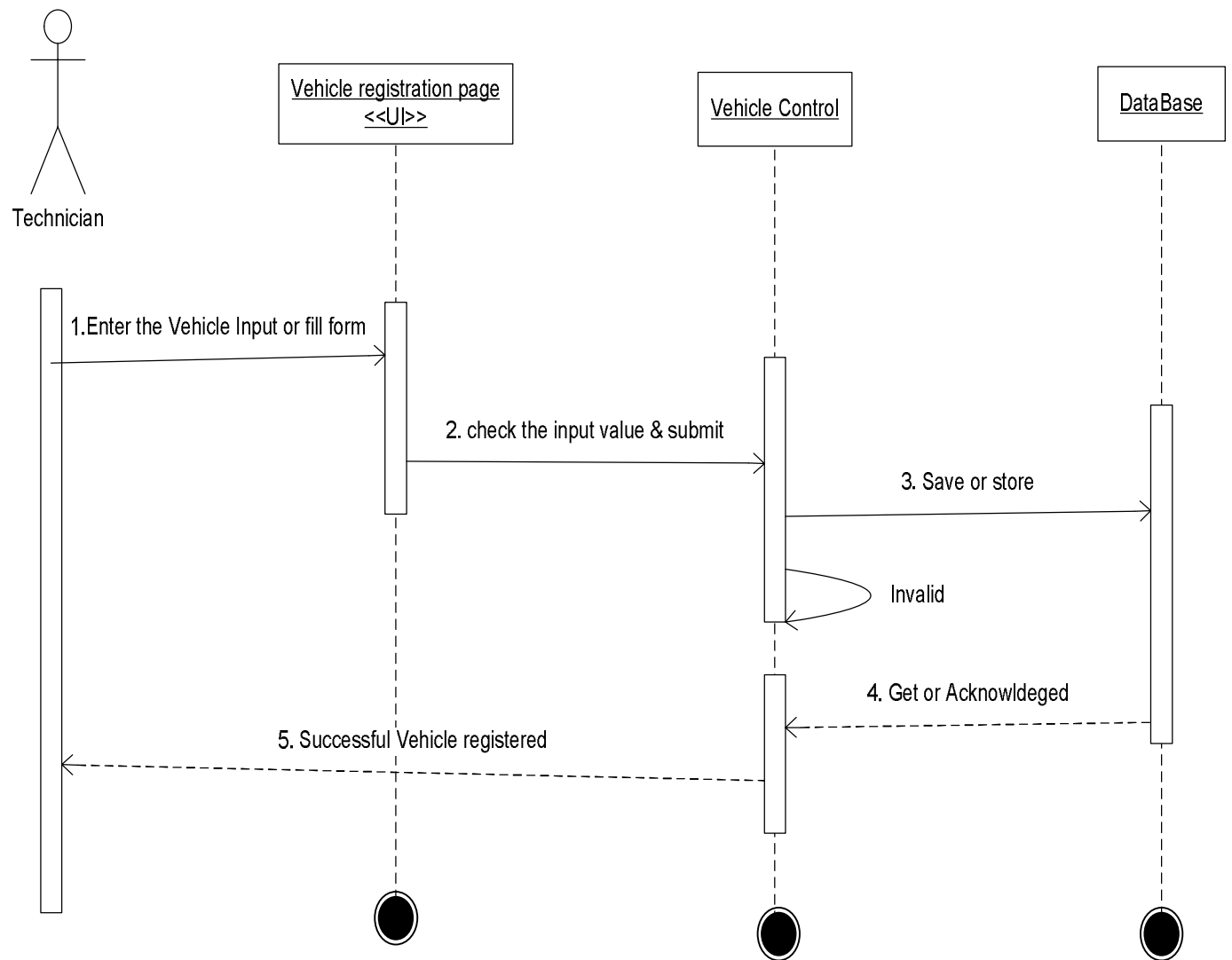


FIGURE 3- 4: REGISTER VEHICLE

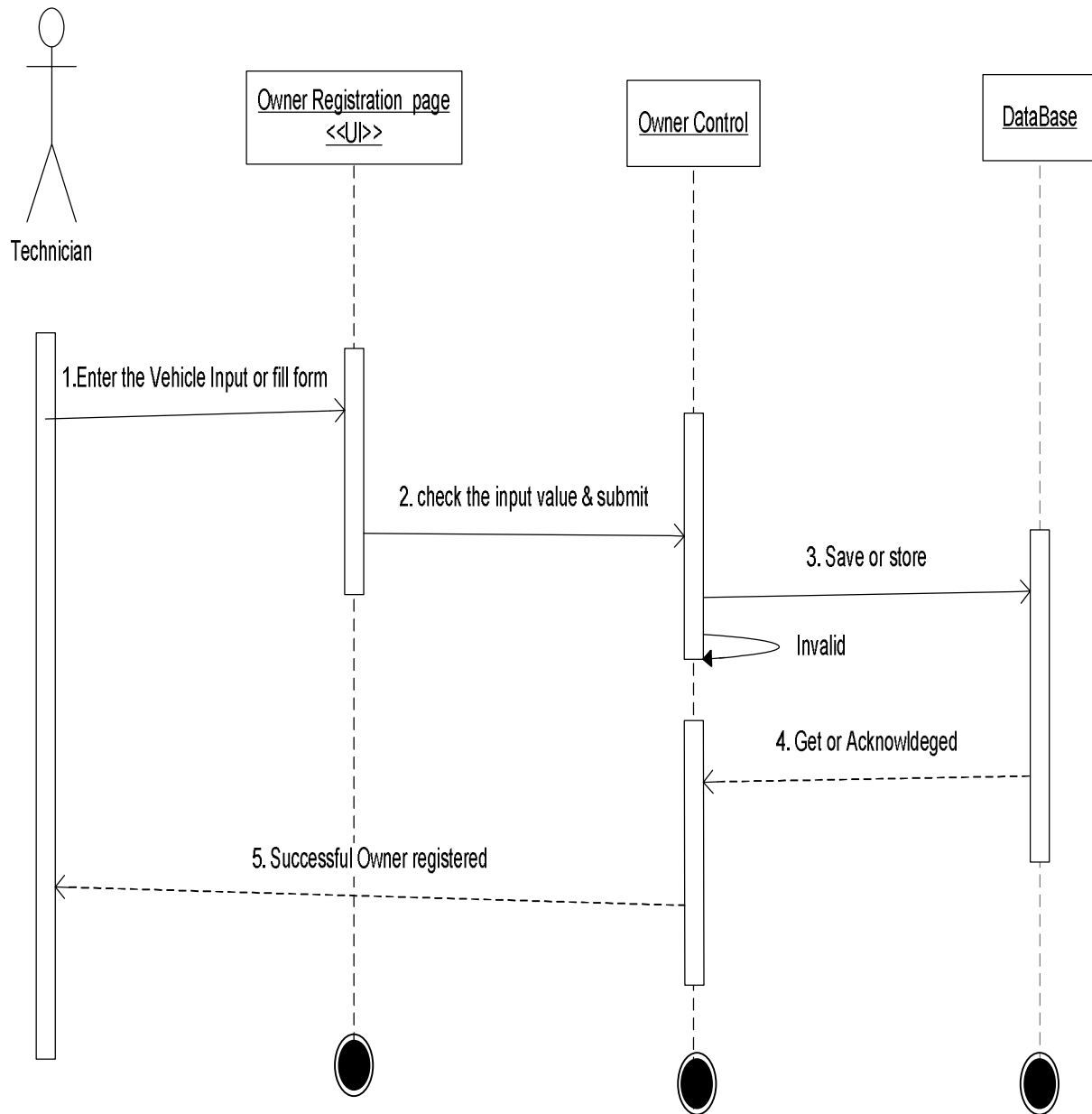


FIGURE3.5:OWNER REGISTER

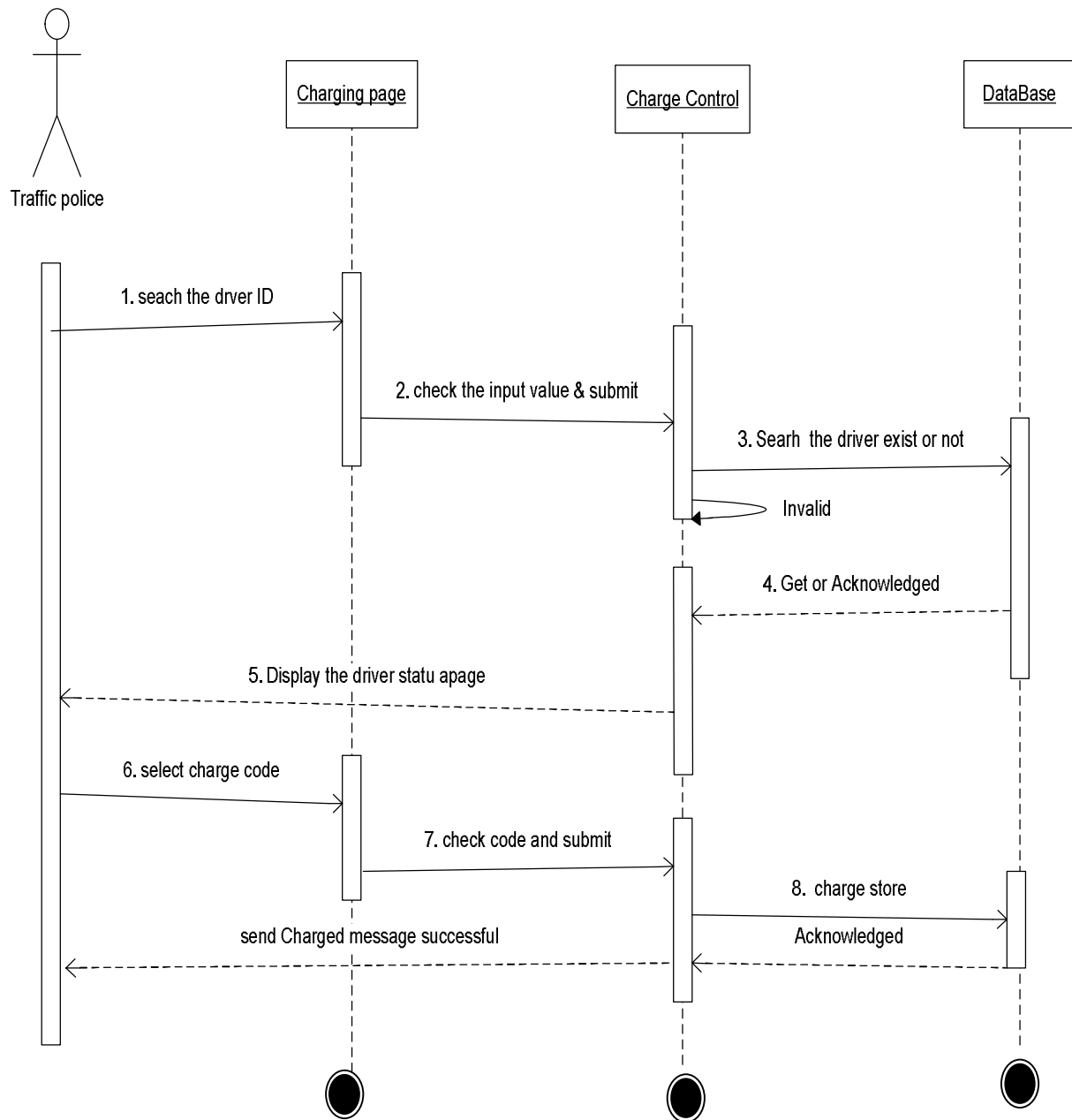


Figure 3- 6: Charging sequence

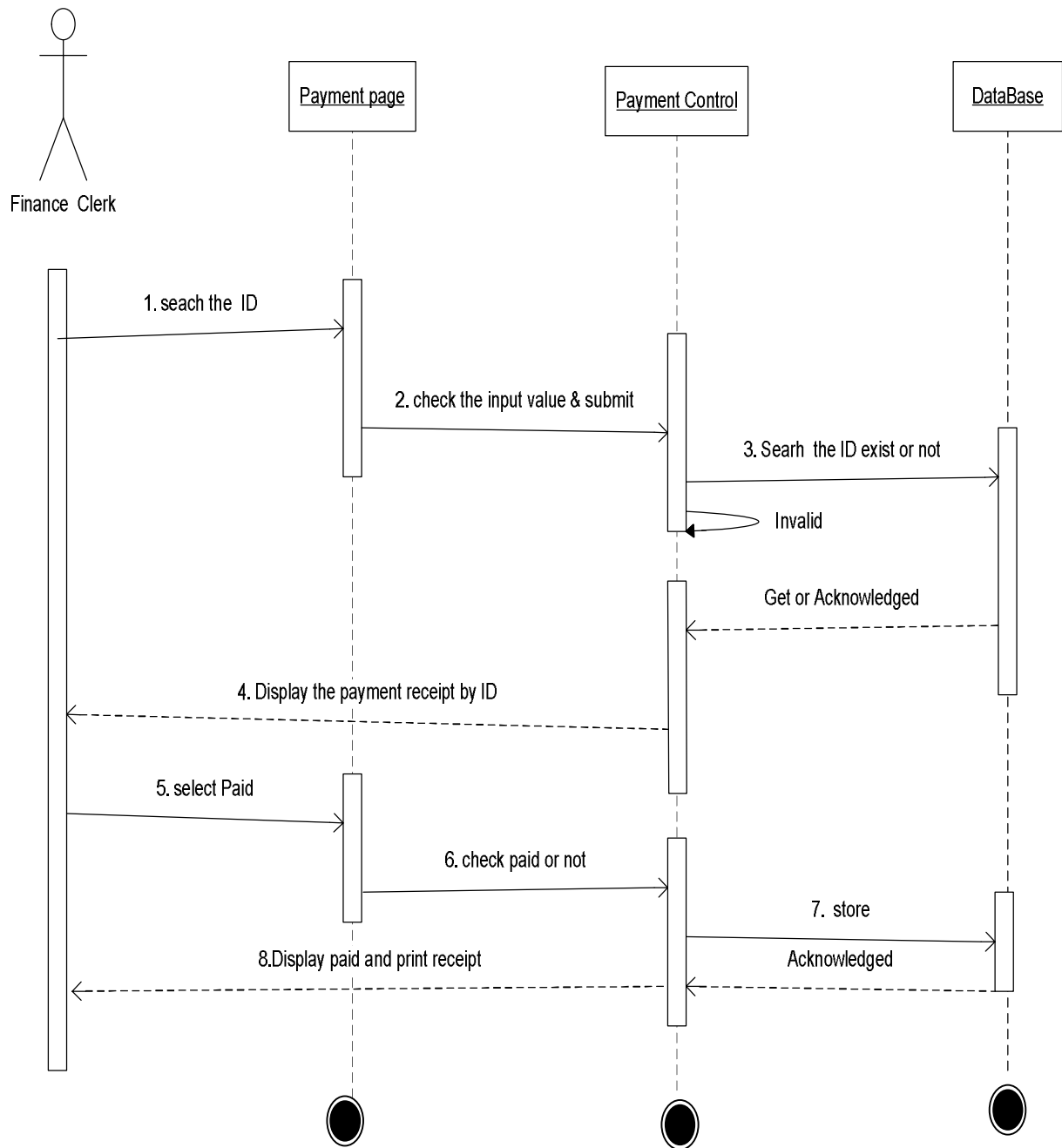


FIGURE 3- 7: PAYMENT

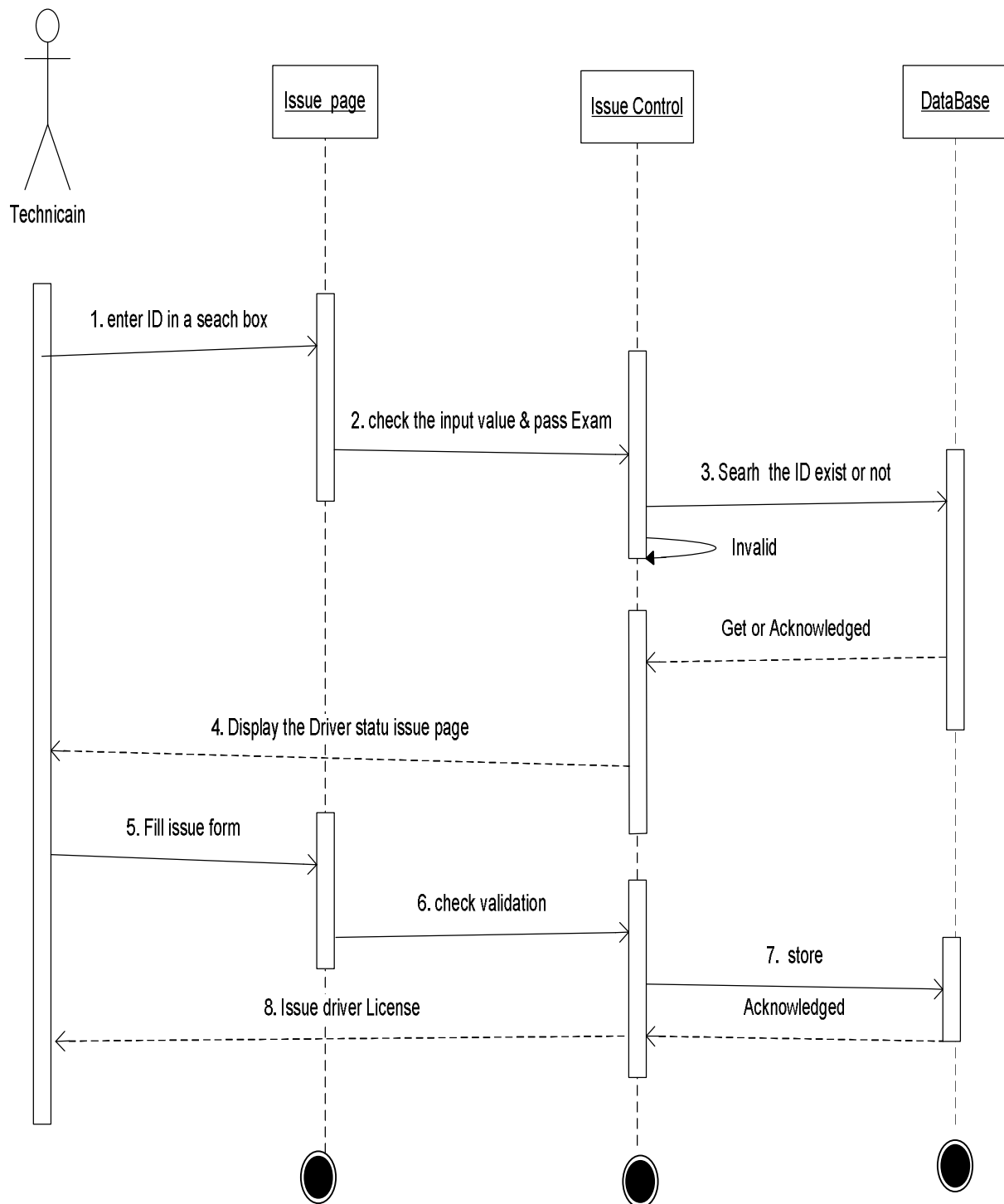


FIGURE 3- 8: ISSUE DRIVER LICENSE

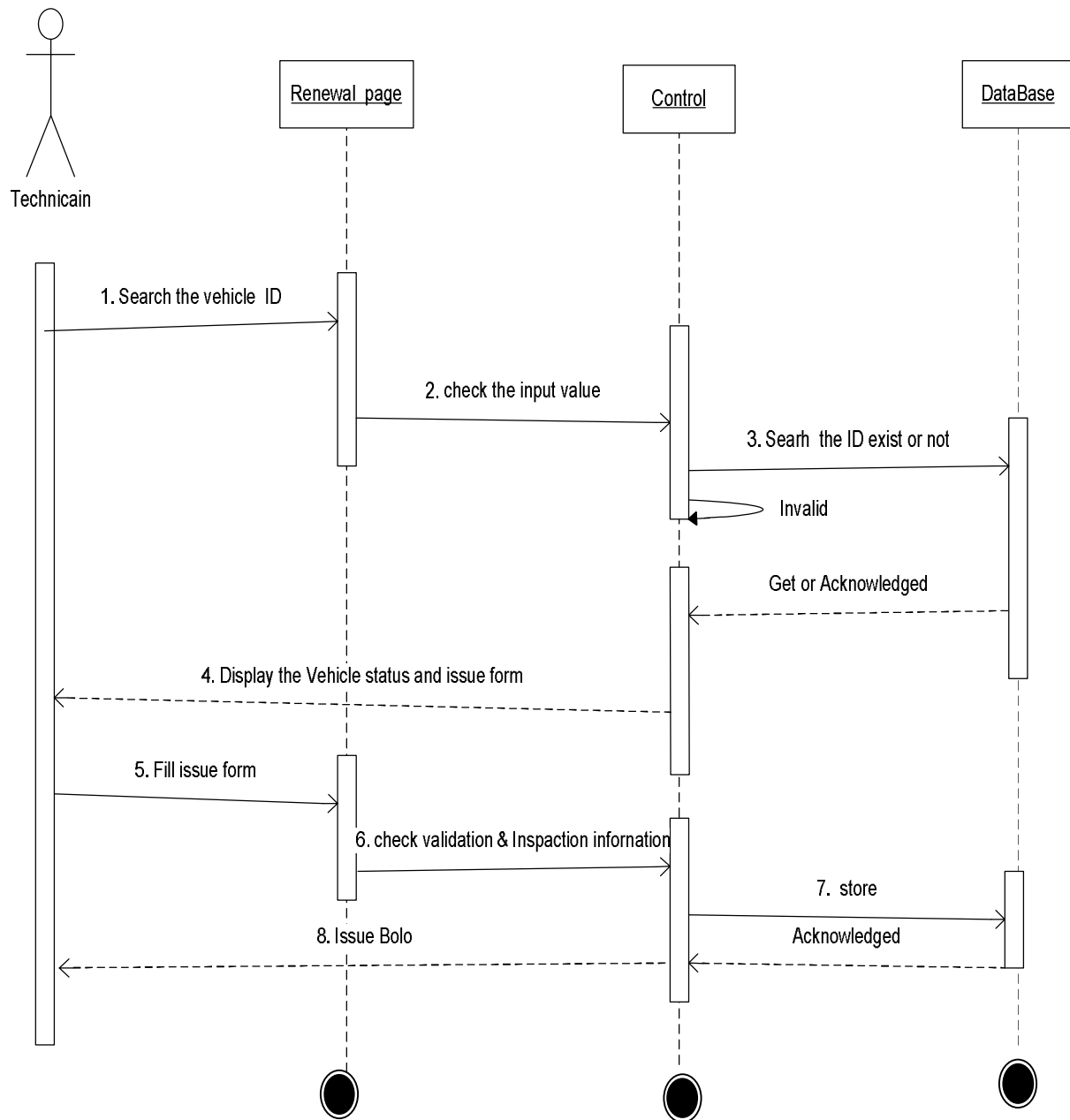


FIGURE 3- 9: RENEW BOLO

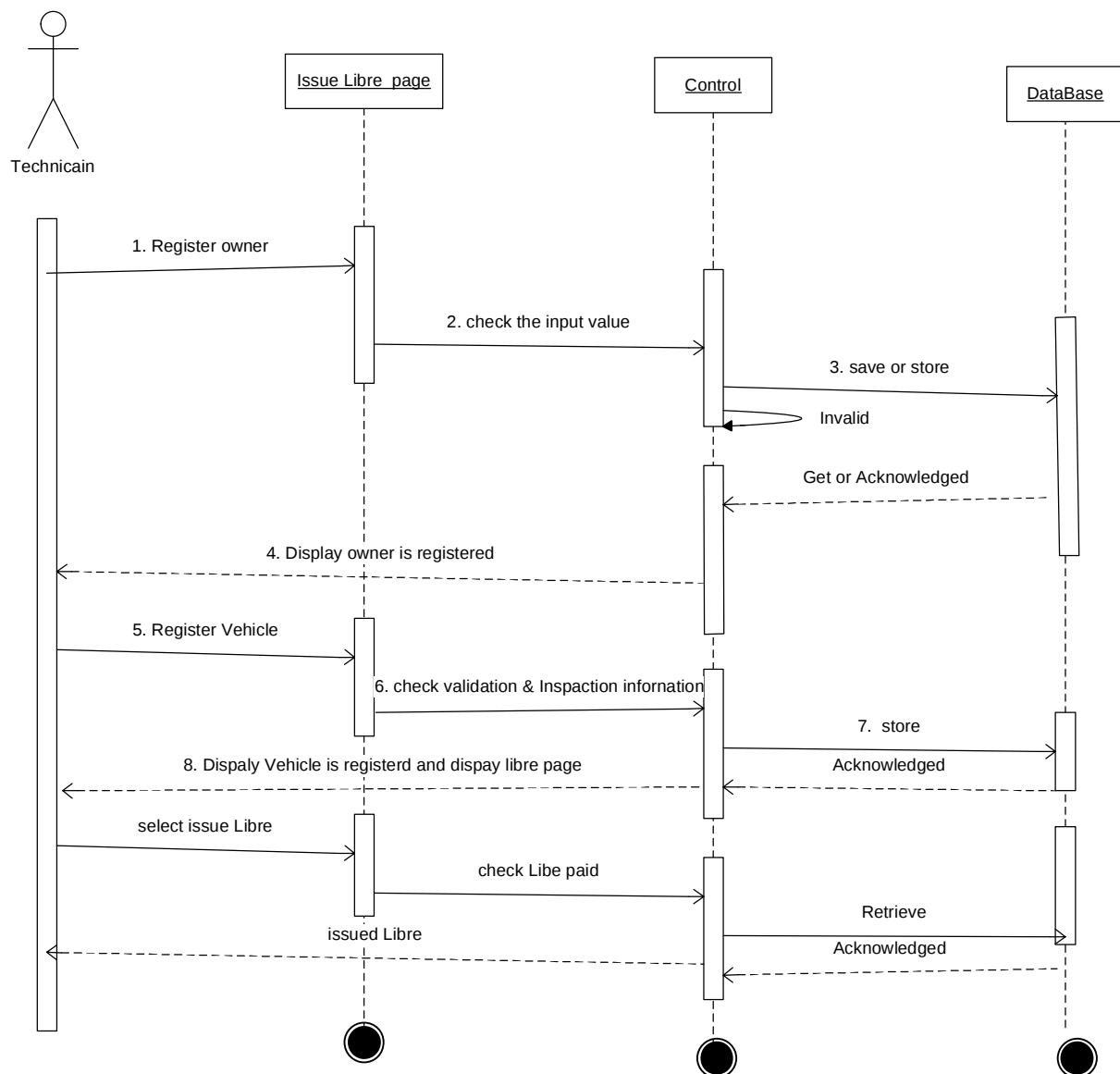


FIGURE 3- 10: ISSUE LIBRE

3.5.4 Activity Diagram

Activity diagrams show the procedural flow of control between two or more class objects while processing an activity. Activity diagrams can be used to model higher-level business process at the business unit level, or to model low-level internal class actions. Activity diagrams are "fewer technicians" in appearance, compared to sequence diagrams, and business-minded people tend to understand them more quickly.

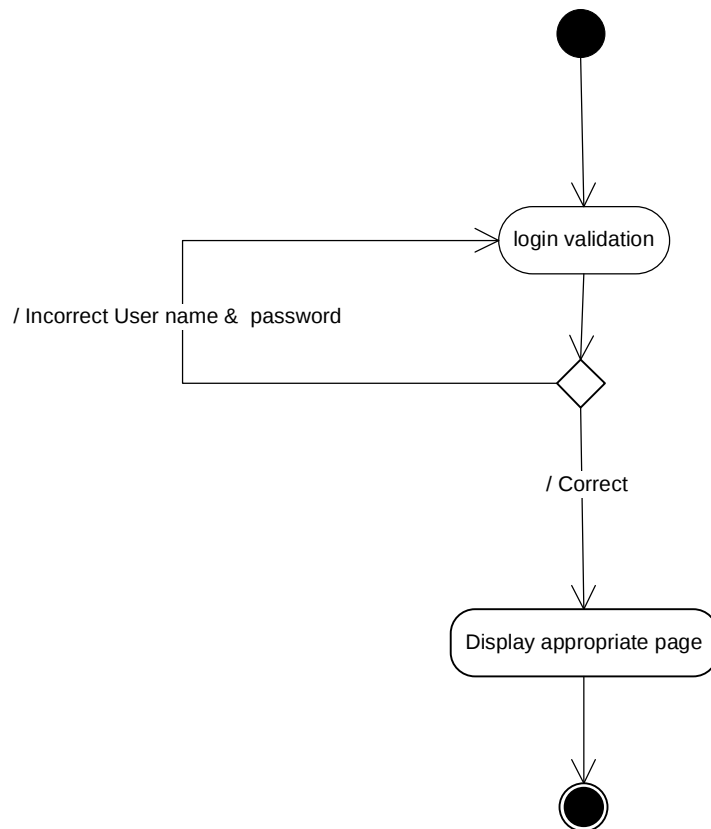


FIGURE 3- 11: LOGIN ACTIVITY

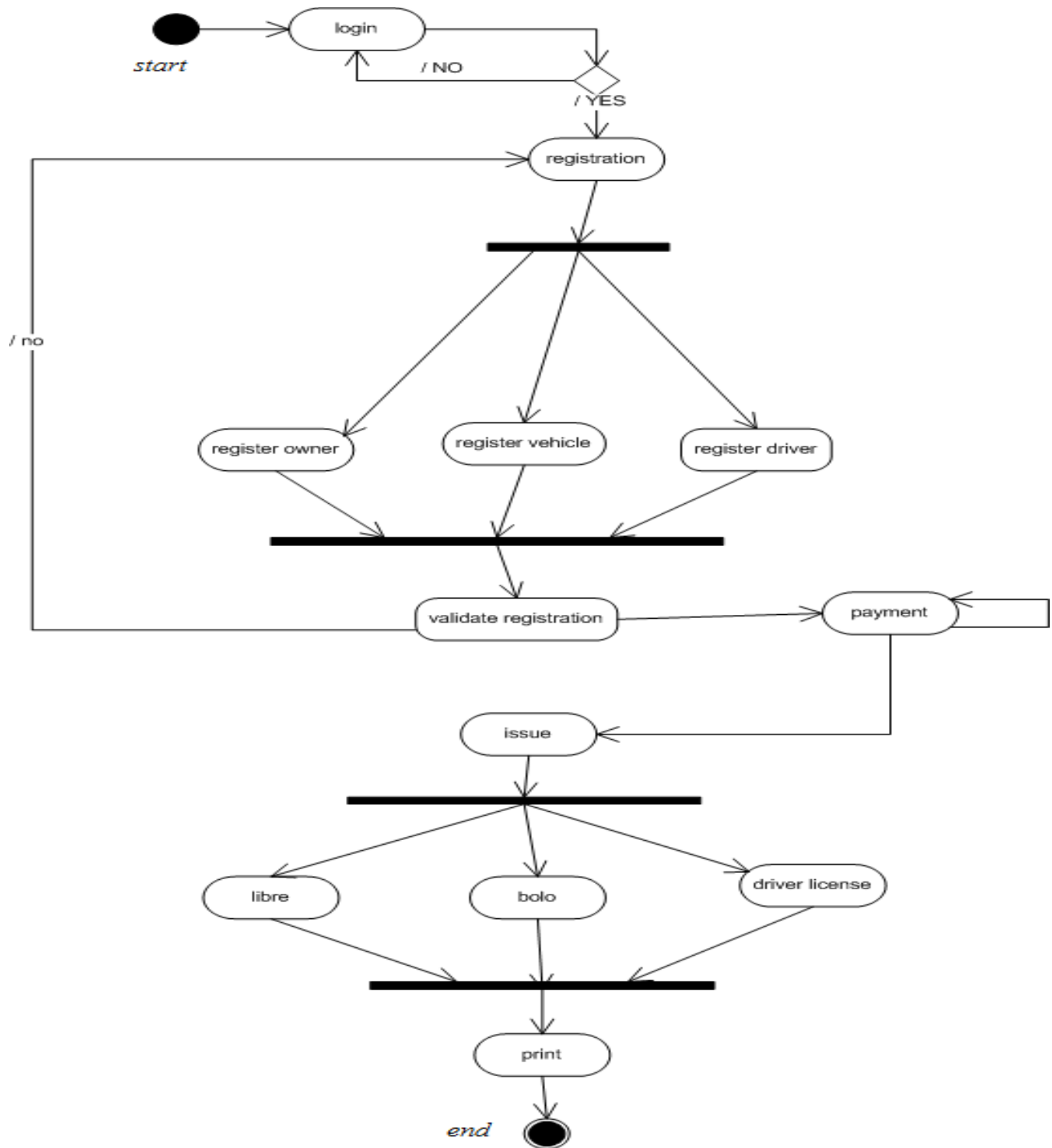


FIGURE 3- 12: REGISTRATION ACTIVITIES

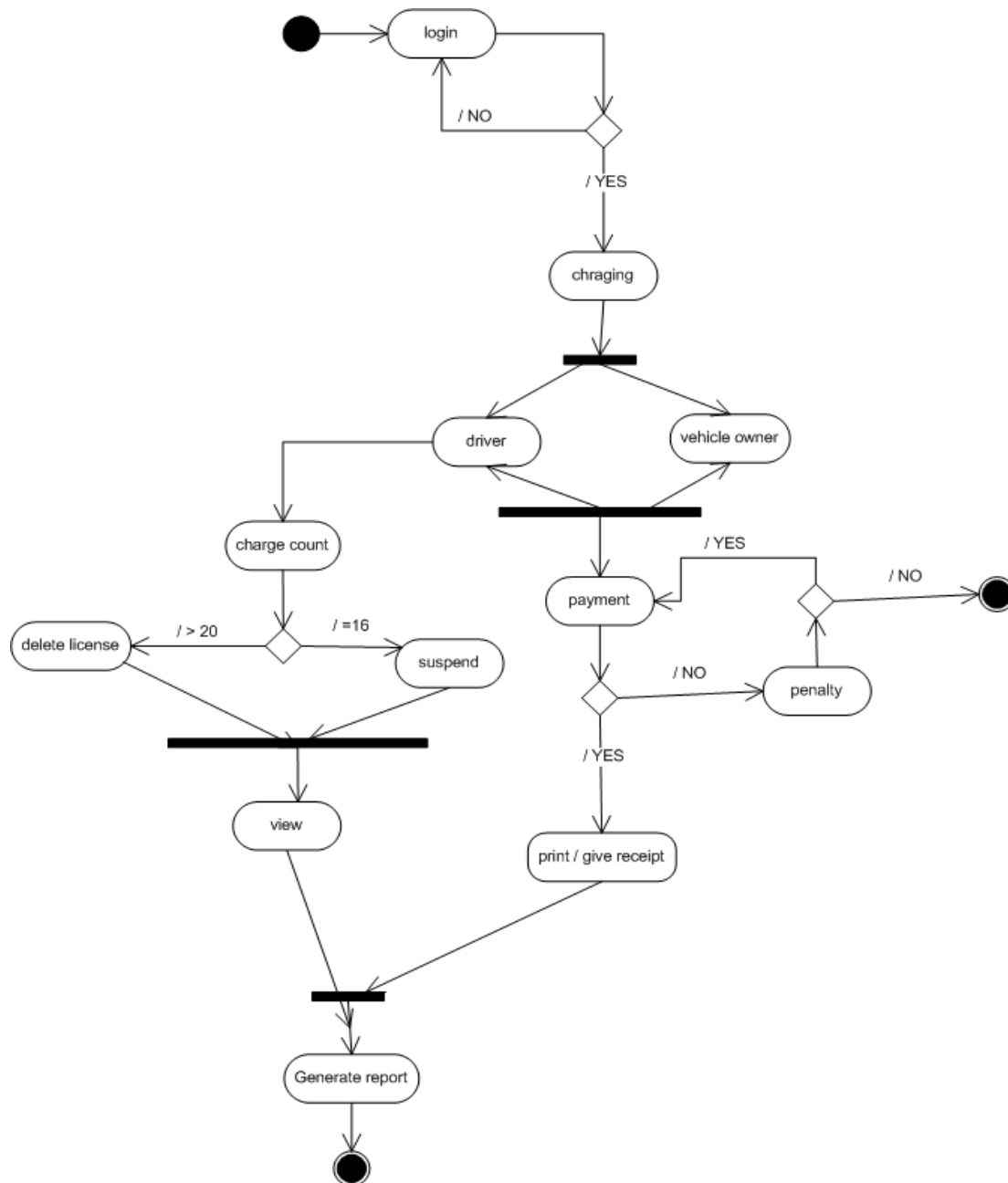


FIGURE 3- 13: ACTIVITY DIAGRAM FOR CHARGING

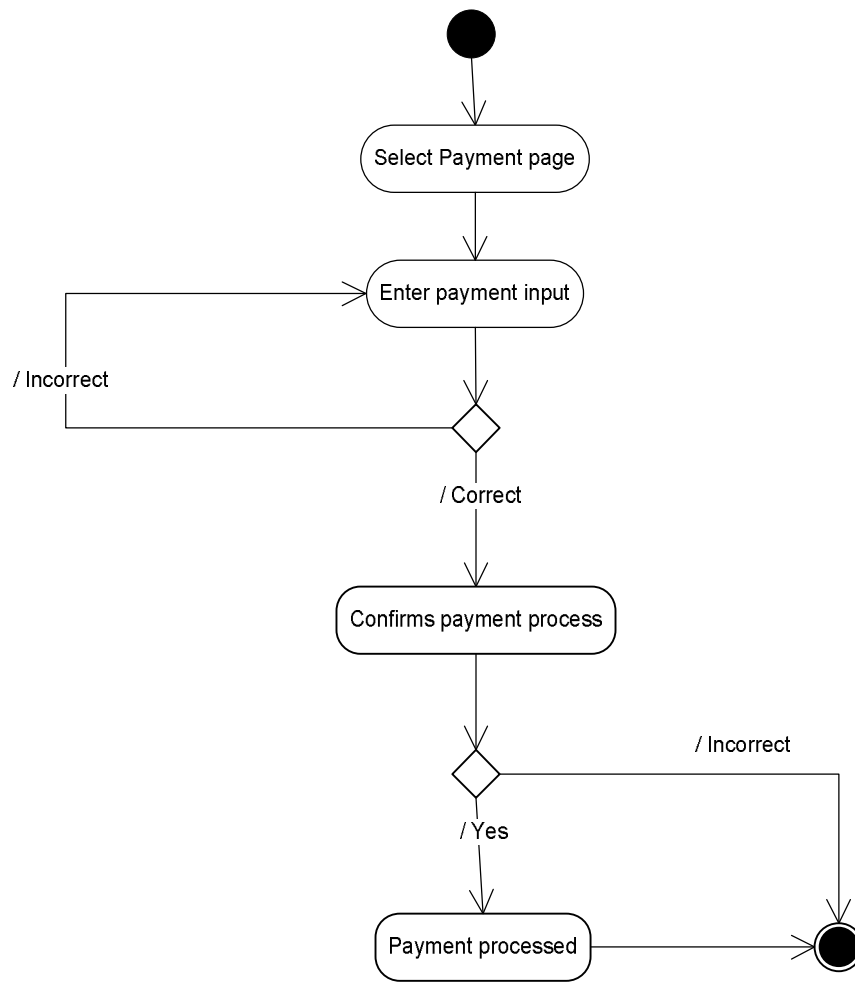


FIGURE 3- 11: PAYMENT BILL

3.5.5 State Chart Diagram

In this part the team used to model the behaviors of the objects by drawing the state diagram. That shows the sequence of states that an object goes through, the events that cause the transition from one state to the other and the actions that result from a state change.

1. The login state chart

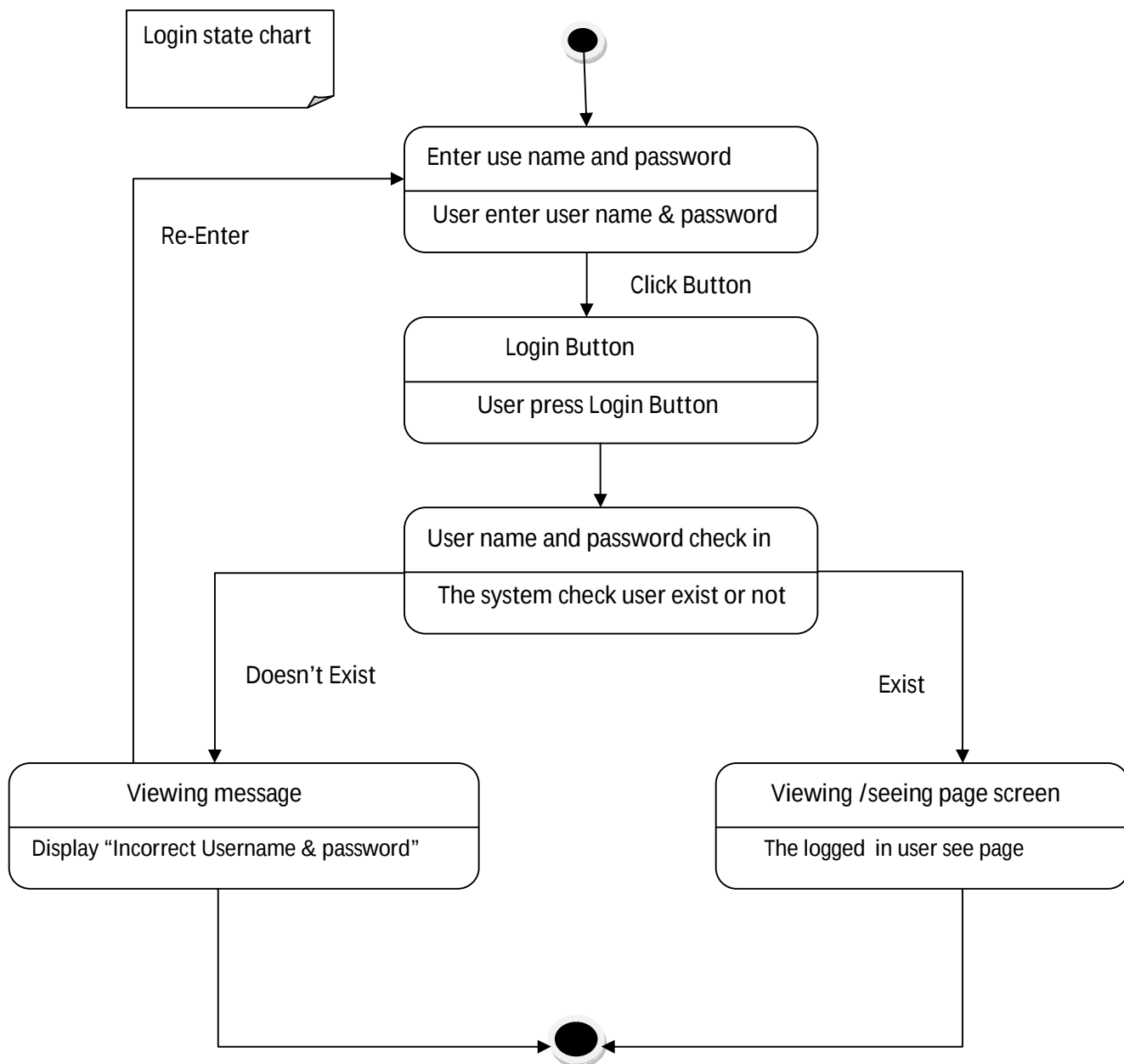


FIGURE 3-14: LOGIN STATE CHART

2. State chart diagram for register vehicle

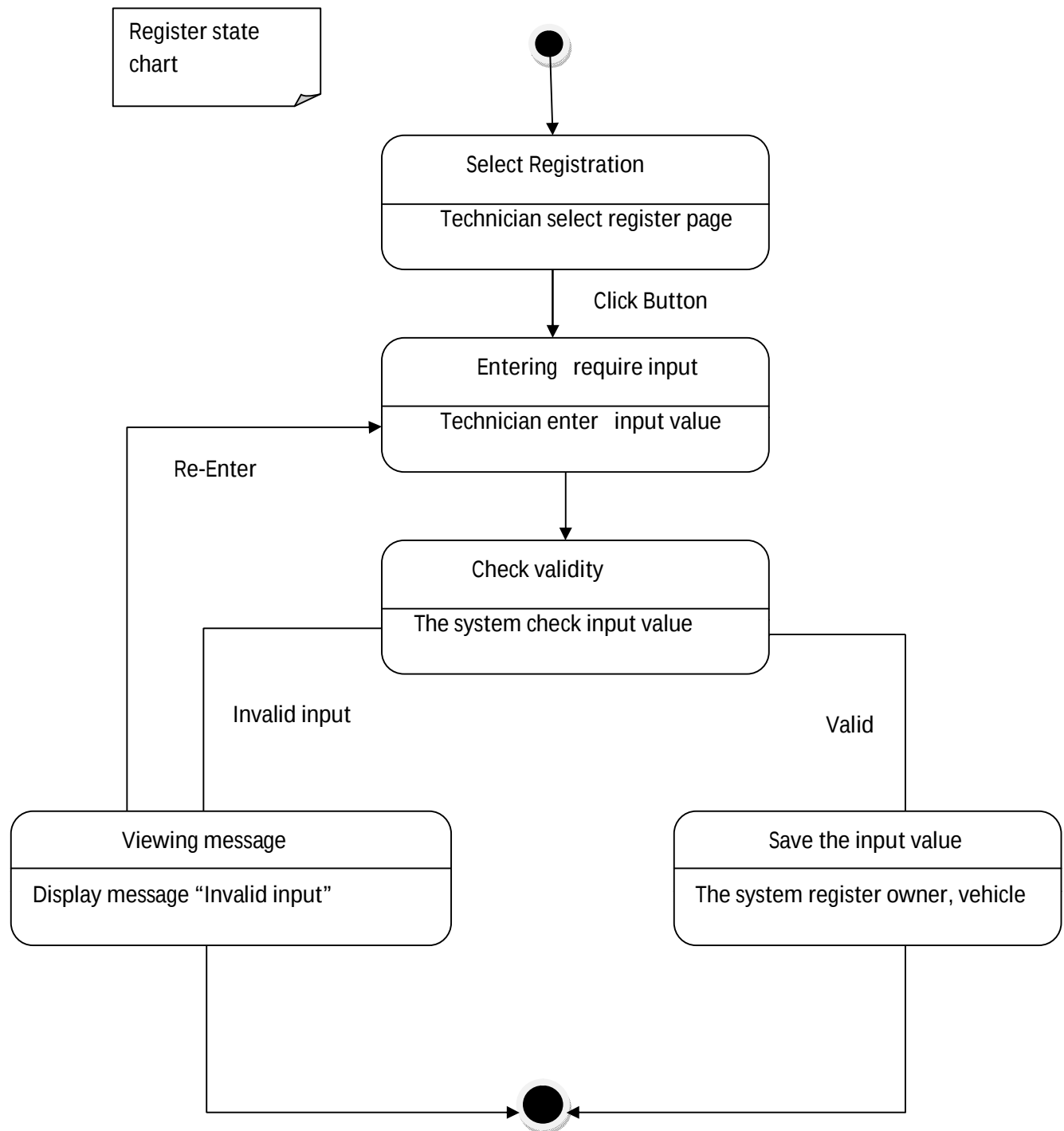


FIGURE 3- 15: STATE CHART DIAGRAM FOR REGISTER VEHICLE

3. State chart Diagram for charge

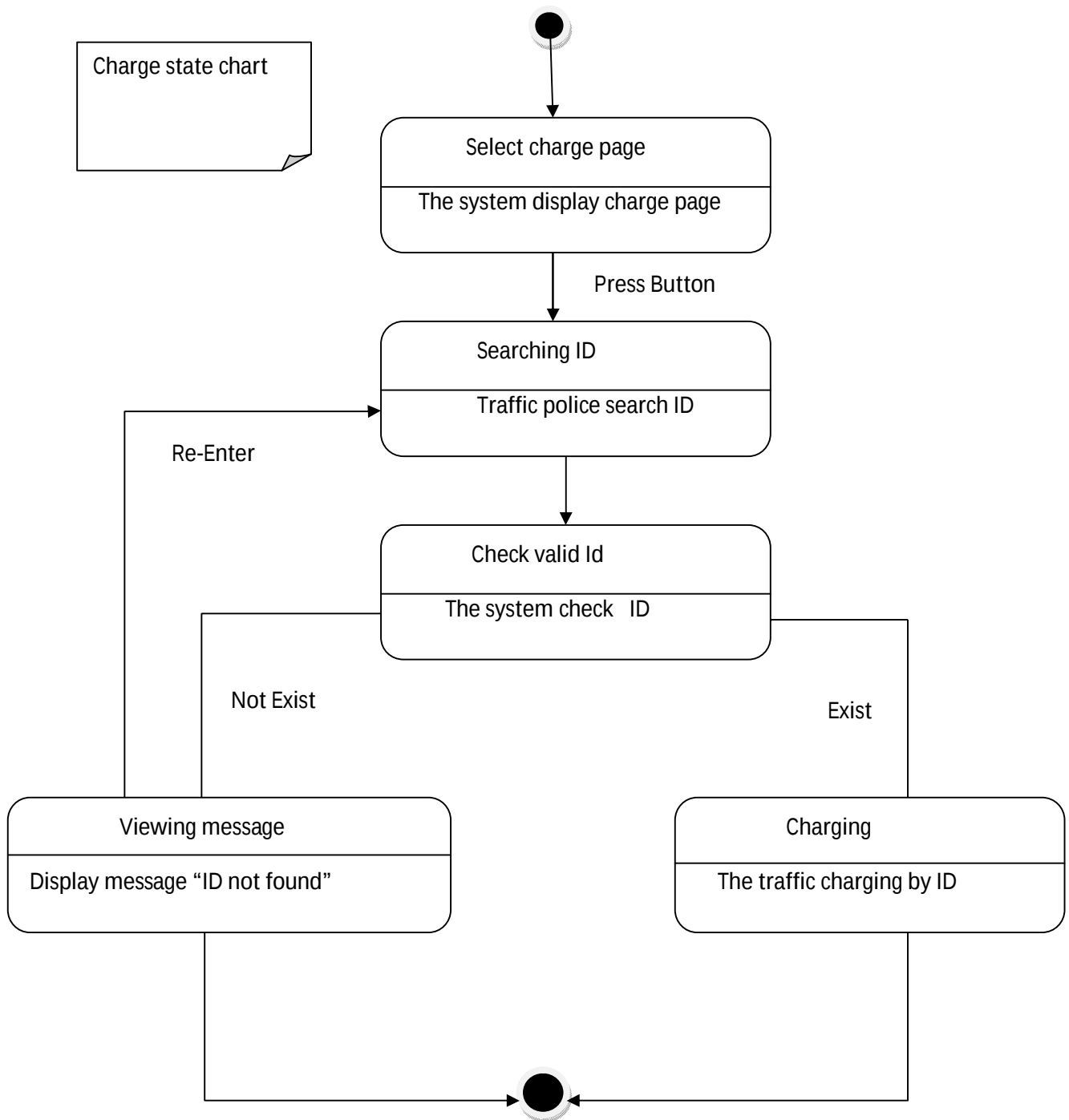


FIGURE 3- 16: STATE CHART DIAGRAM FOR REGISTER VEHICLE

3.5.6 Component Diagram

Component diagrams describe the organization and dependencies among physical software components, such as source code, runtime (binary) code, and executables. For example, a component diagram can illustrate the dependency between source files and executable files, similar to the information within make files, which describe source code dependencies and can be used to compile and link an application. A component is represented by a rectangle with two tabs overlapping the left edge. A dependency is denoted by a dotted arrow going from a component to the component it depends on.

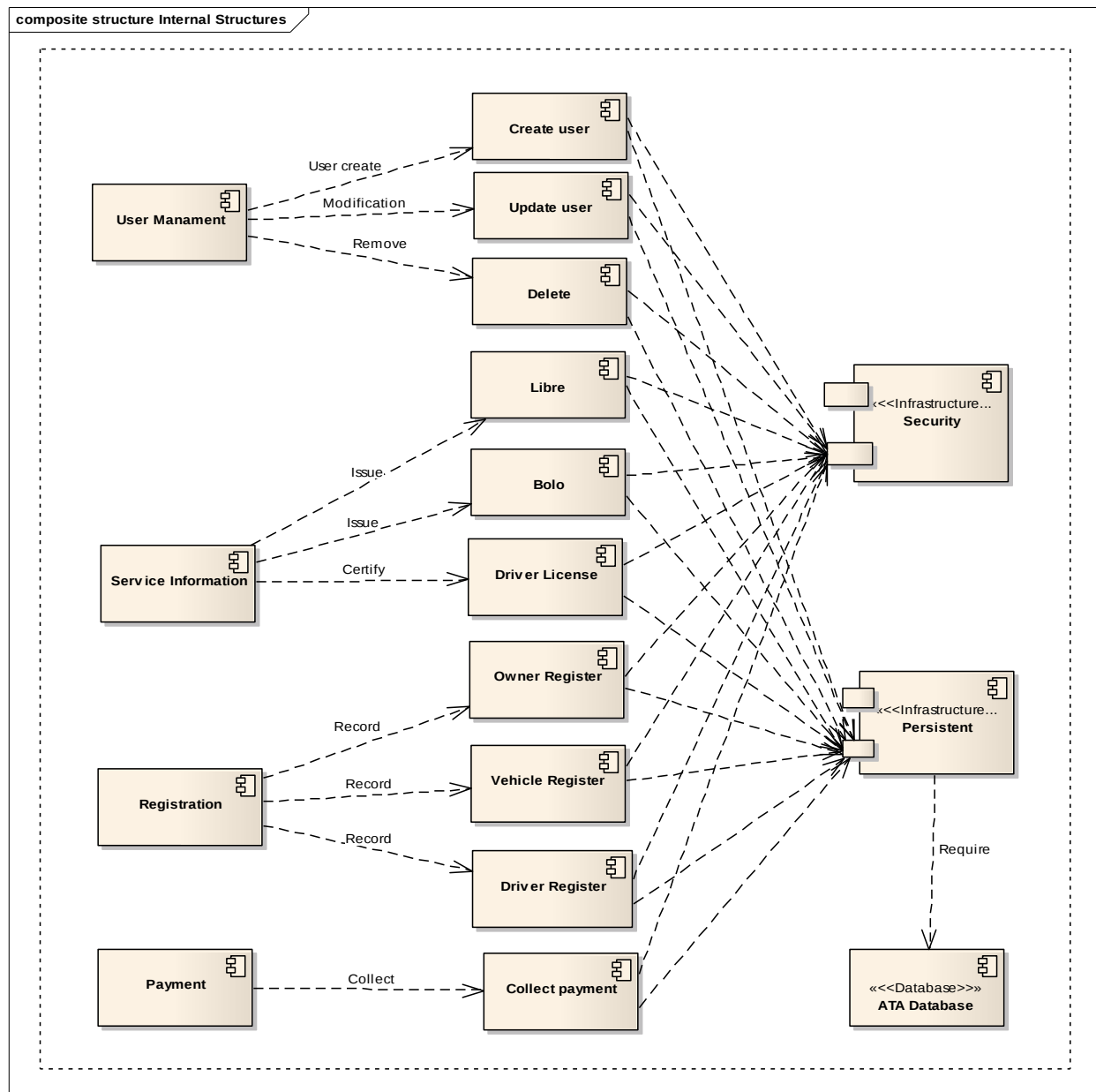


FIGURE 8- 17: COMPONENT DIAGRAMS

3.5.7 Deployment Diagram

Deployment diagrams depict the configuration of the runtime system, showing the hardware nodes, the components that run on these nodes and the connections between nodes. A node is represented by a three-dimensional cube.

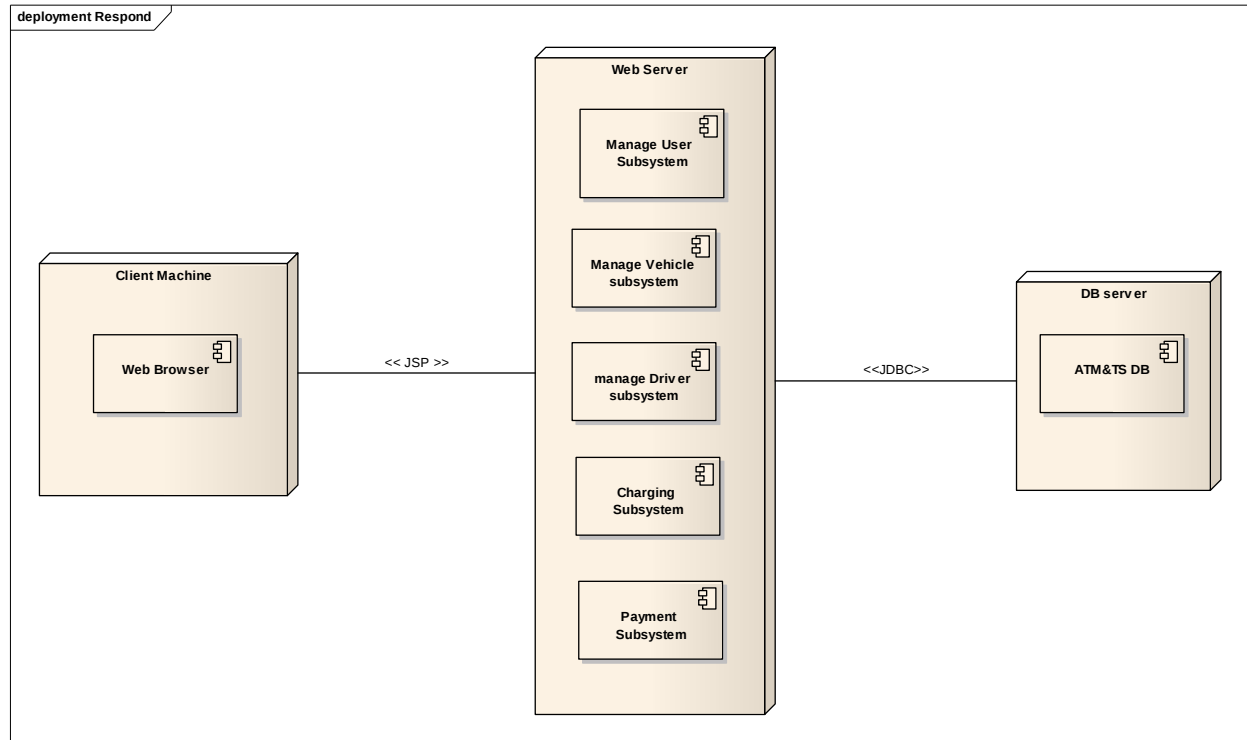


FIGURE 3- 18: DEPLOYMENT DIAGRAM

The above diagram shows deployment view of Adama transport monitoring and traffic police web base system. There is physical allocation of components to computational units such as node web browser connect to web server using JSP and web server node connect to database using JDBC,

CHAPTER FOUR

4. SYSTEM DESIGN

4.1. Introduction

This chapter describes the result of the design of the system design carried out during design phase of the Adama transport monitoring and traffic system. The design goal is stated clearly in such a way that making design decisions will be in light of these goals. The second section which is the most important one describes the system under development in terms of system decomposition, Current software architecture, Hardware/software mapping, Persistent data management, and database system and access control. Finally, the last section provides a detailed design of Global software control, Boundary conditions, and Subsystem services.

4.1.1 Purpose Of The System

The web based transport monitoring and traffic system web based application under consideration will facilitate the orderly and timely issuance bolo, driver license and libre and charge and pay with web browser minimize the cost and time and interface of human and reduce corruption.

4.1.2 Design Goals

The design goals represent the desired qualities that the system should have and provide a consistent set of criteria that would be taken into consideration when making design decisions.

The following are also outlined to be the design goals of the project.

- **Availability:**-The system should be available 100% of the time during all working hours.
- **Security:**-The system should be designed in such a way that every user is prompted for username and password to access the system's database or some functionality.
- **Extensibility:**-The system should be able to entertain the addition of any new services related to reporting system of the accident while at the same time preserving the general frame work of the system.
- **Usability:** -The system should have a simple and easy graphical interface that users can understand quickly.

4.1.3 Overview

The purpose of this design document is to provide the design models and methodologies that are developed and used to satisfy the requirements. This document presents detailed diagrams that represent the functionality of the system from the system's point of view to provide the necessary interfaces to the users.

4. 2. Current Software Architecture

The current system uses Peer-to-peer (P2P) architecture. This method is delivering computer services amongst a set of participants. Each participant shares a portion of their own resources, such as processing power, disk storage, network bandwidth, printing facilities. These resources are provided directly to other participants without intermediary network hosts or servers. Thus P2P participants are both providers and consumers of services simultaneously in contrast with other service models.

In Adama transport agency use database designed and developed with Microsoft access 2007. This database is shared among computer in peer to peer architecture. The current system processes all activity by hand and results are stored in database.

4. 3. Proposed Software Architecture

The web based transport monitoring and traffic system use a client-server model with an MVC architecture to maximize achievement of the design goals. MVC allows the system to achieve the goals of high fault tolerance, flexibility, modifiability, low cost, and usability.

The client-server model works particularly well for systems requiring high fault tolerance since the architecture supports backup servers. Client-server architecture supports the seamless connection between many clients and with an arbitrary number of servers.

MVC is a well defined design that allows for easy modifications as long as the predefined interfaces remain the same. Thus MVC naturally meets the design goals of flexibility and modifiability.

Finally, the client-server and MVC models support client-side GUIs and server-side databases. The clients can support appealing GUIs which increases the usability of the system (thus meeting the user friendliness design goal). A high level summary of the web based transport monitoring and traffic system MVC model is as follows:

- The view runs on the client and is displayed in a web browser.

- The view makes control requests to the Controller via the provided interface.
- The controller then gets and sets attributes of the model which is the third layer.
- The model is an active record running off of a MySQL database.

4.3.1 Overview

In this section, the detailed description of the proposed software architecture will be given through different topics which are subsystem decomposition, hardware-software mapping, persistent data management, access control and security, global software control and boundary conditions.

4.3.2 Subsystem Decomposition

A subsystem is characterized by the services it provides to other sub system. Services are a set of related operations that has a common purpose. In order to make the implementation easy and manageable, the newly developed system for the transport service is decomposed in to the following smaller subsystems.

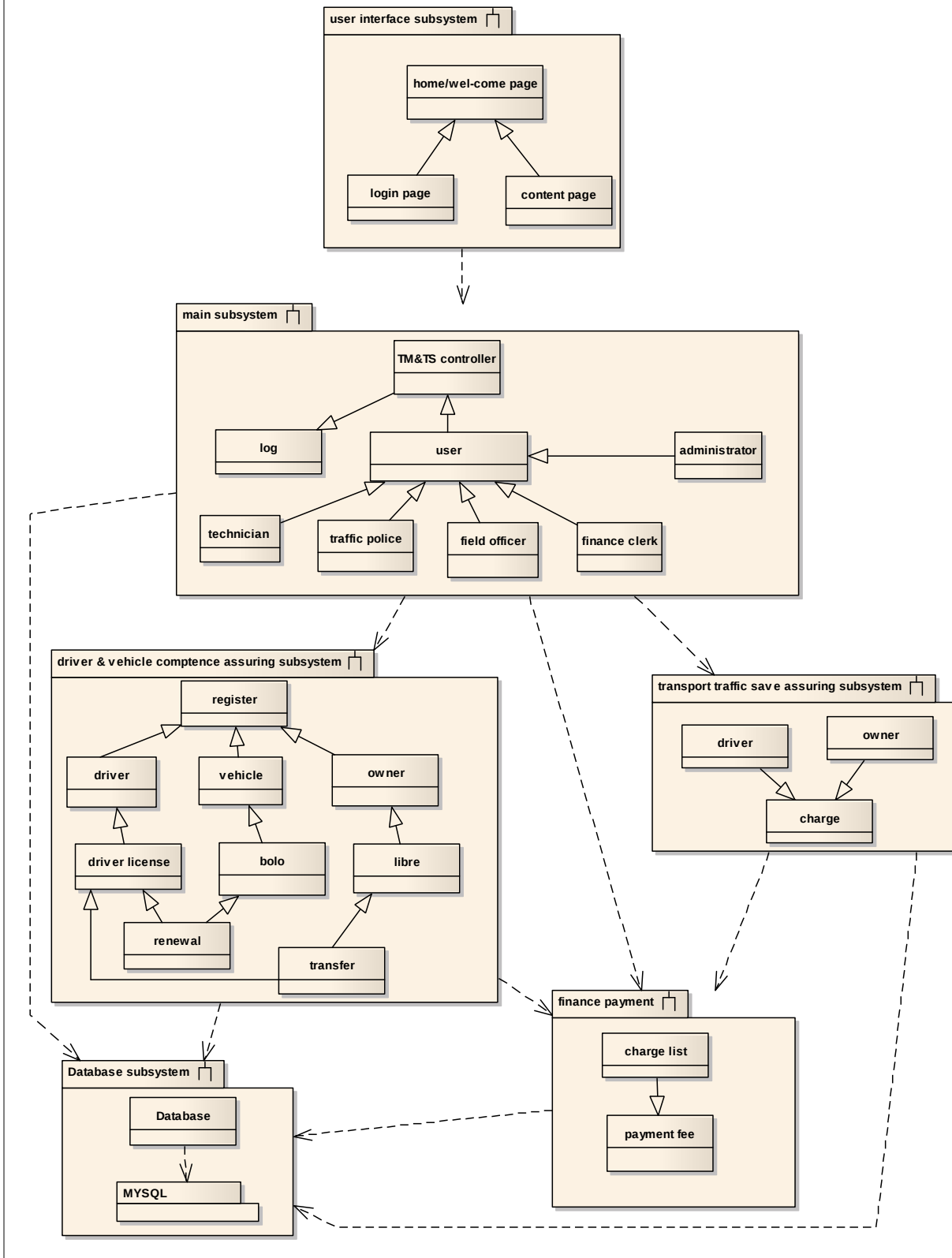


FIGURE 4- 1: SUBSYSTEM DECOMPOSITION

4.3.3 Hardware/Software Mapping

The client only requires a basic computer running a web browser with a connection to the server. The server requires access to the database and sufficient computing power to handle the load of users expected.

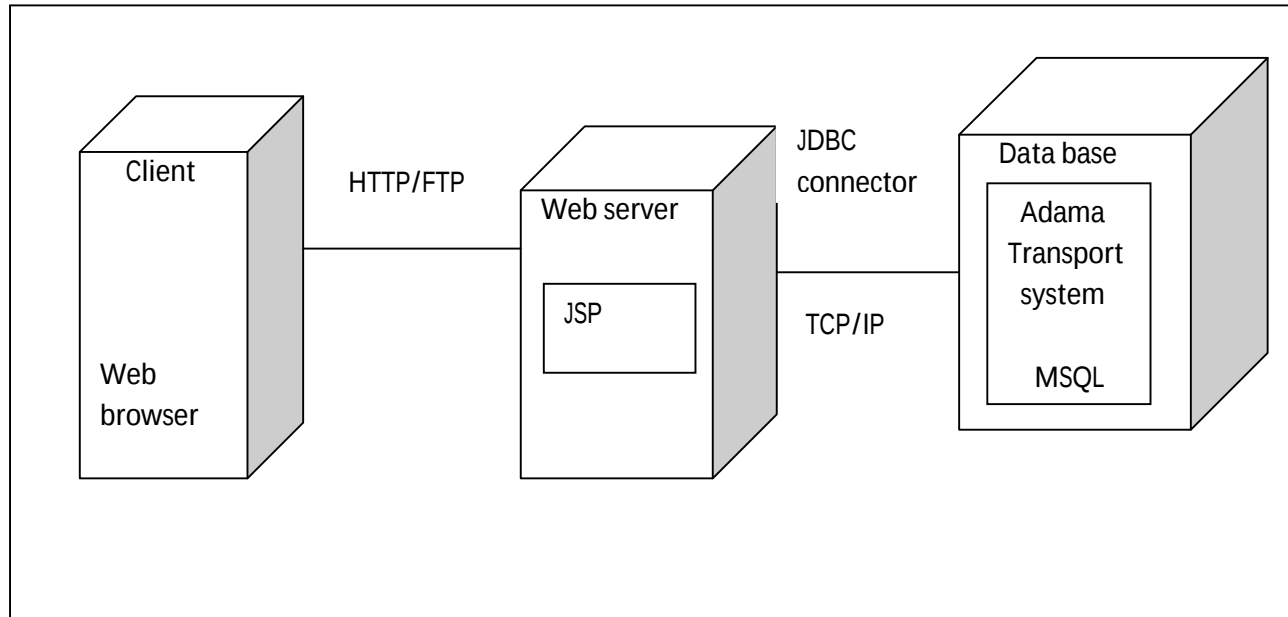


FIGURE 4- 2: HARDWARE AND SOFTWARE MAPPING

4.3.4 Persistent Data Management

Web based transport monitoring and traffic system will largely depend on a relational database to perform day-to-day operations and storing log data. Data will be stored in a MySQL 5.7 DBMS and manipulated through the Database Subsystem, which will ensure data integrity and consistency. Database Subsystem will contain all necessary SQL queries that will be accessible by the rest of the Subsystems.

Database will be hosted on Adama Transport Agency Server and will be accessible 24/7 via a local area network. The data stored in the database will include:

- Owners
- Vehicle
- Trainee drivers
- Driver license
- Charge cases
- Inspector organization
- Users
- Wanted list

The database will be backed up on a daily basis to ensure data security.

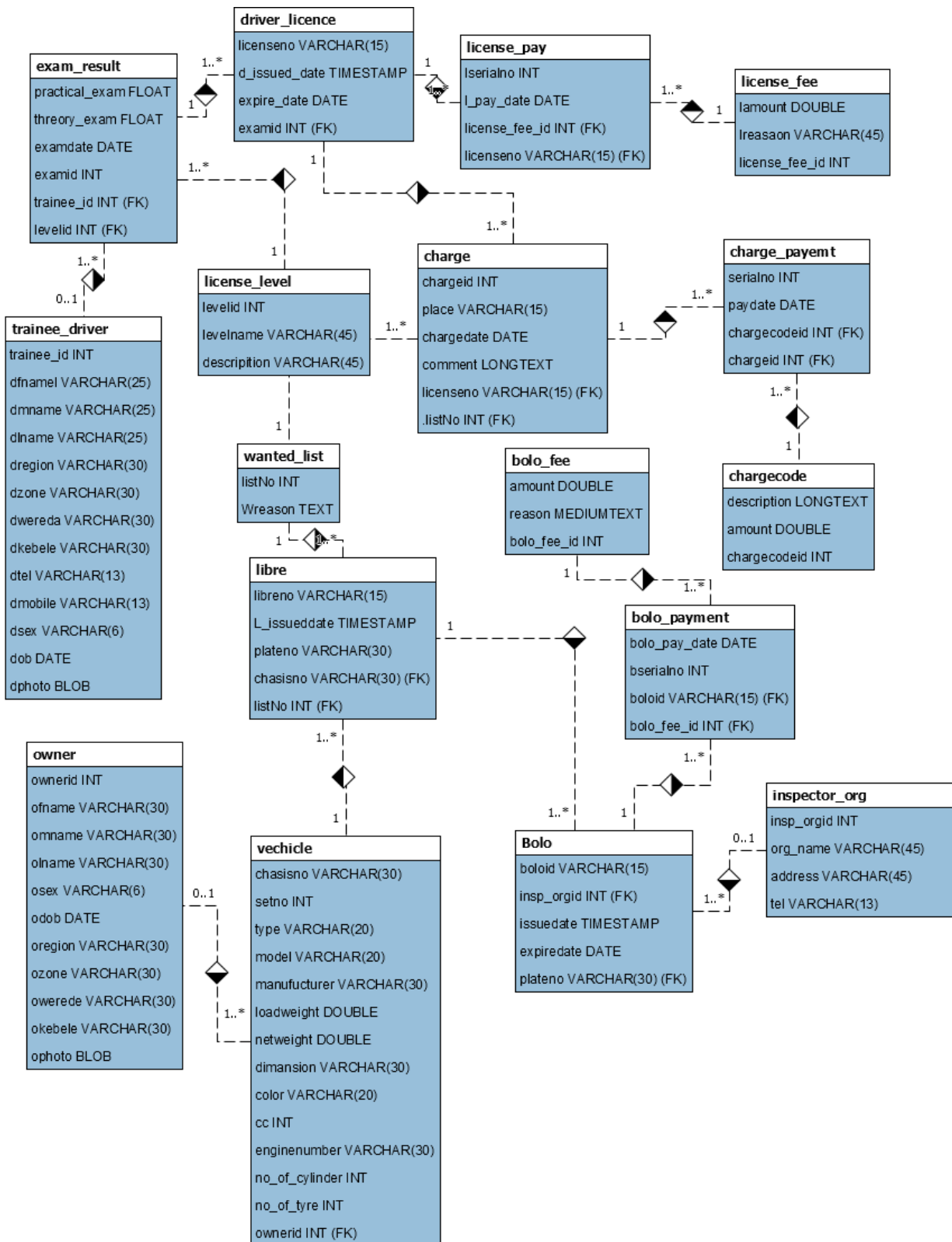


FIGURE 4- 3: PERSISTENT DATA MANAGEMENT

4.3.5 Access Control and Security

Access Control

All users need to login before taking any action. After login, each user is going to be redirected to different landing pages depending on their user role. Every user's access will be limited. A driver can only view her/ his data, a owner can only view her/ her data , a technician can only access the owner, vehicle and driver data ,a traffic police and field officer can access only charge data and a clerk can access the finance payment . Since all the communication between client and server is going to be handled by internet, the connection protocol is going to be HTTPS.

Access Control Matrix:

The access control matrix defines access rights of primary system actors .The actions available are Create (C), Read (R), Update (U) and Delete (D).

Table 4- 1: Access Control Matrix

Activity	Actors				
	Administrator	Technicians	Field officer	Traffic police	Finance clerks
User account	C R U D	R	R	R	R
Bolo	-	C R U D	R	R	R
Libre	-	C R U D	R	R	-
Driver License	-	C R U D	R	R	R
Renewal bolo	-	C R U D	R	R	R
Renewal driver license	-	C R U D	R	R	R
Transfer	-	C R U D	-	-	R
Payment	-	C R U D	R	R	R

Security:

- Authentication mechanism: Transport monitoring and traffic system will be used for authenticating the user login.
- Encryption: SSL will be used for data encryption that will be sent over the internet.
- Database Security: Varying level of access control will be provided to different users of the system to maintain the database security. Access control will be differentiated based on the access privilege provided to each user for each table. Administrator will have update privileges on all the tables.

4.3.6 Global Software Control

The Transport monitoring and traffic architecture is to have an explicit, decentralized software control. The system's dynamic control is distributed among different controllers such that each object delegates some responsibility to other objects. The request initiations are event-driven:

- The technician initiates the UI Subsystem by logging into the system. The Transport monitoring and traffic Main Subsystem is initiated the driver and vehicle competency assuring.
- The traffic police, field officer initiate the UI Subsystem by logging into the system. The Transport monitoring and traffic Main Subsystem is initiated the transport traffic save assuring.
- The finance clerk initiates the UI Subsystem by logging into the system. The Transport monitoring and traffic Main Subsystem is initiated the payment.
- The Database Subsystem can be initiated at any point by all Subsystems except the UI Subsystem. Any request (add/update/delete/retrieve) to the repository initiates the Database Subsystem. The system stores its contents in the database between executions. When the application is run again, it retrieves the contents from the previous execution. Any change in the contents during this execution updates the database.

Global control of this website client will be processed request a server. When a user makes a request from the client software, the first thing done in the server is to check if the user is authenticated or not. After authentication is approved, the data is processed according to the request. Client receives the JSP and parses it with the appropriate parsing template depending on the request. After parsing, necessary data is presented to user.

4.3.7 Boundary Conditions

We considered server and client with three boundary condition.

- Initialization: The server must be started prior to the connection of any clients. This shall be done by the Administrator.
- Termination: If a client disconnects, either intentionally (i.e. by logging out of the system) or unintentionally (i.e. unresponsive computer), this shall have no effect to the server. If the server shuts down, then each of the client's requests shall time out and an error shall be reported in the client's GUI.
- Failure: Failure of either the server or clients shall be treated the same as in the termination.

We consider access log in validation system with three boundary condition.

Initialization

- A user enters the web interface of Transport monitoring and traffic application using an internet browser.
- User needs to be authenticated in order to take any action. After authentication the browser sends a cookie which presents unique UserID. If already authenticated, user is provided to continue.
- If authentication is successfully completed, other pages will be available for him according to his user level.

Termination

- At any instant of a time, a user can terminate his session by going to logout page.
- Termination is also done through time checks. Each cookie that is assigned to a user has expiry time of an hour. After an hour, cookie expires and member needs to authenticate again.

Failure

- Upon a connection lost with database an error message will be presented to user and no further action will be allowed.
- If user enters an unexpected input, execution will be terminated and user will be prompted to enter correct information.

We consider three boundary condition such as startup, shut down and error condition.

Startup: go to system URL and login

Shut Down: click log out and close browser

Error Conditions:

- Logging in:
 - o Username or password field are blank.
 - o Username is not a decimal number.
 - o Password and username don't match.
 - o Username is wrong or does not exist.
 - o The welcome screen does not appear after logging in.
- User settings
 - o User is unable to change certain settings or changes don't reflect.
 - o Between the time of editing and updating, the system crashes.

4.4. Subsystem services

This section is intended to describe the services provided by each subsystem.

The User Interface (UI) Subsystem

- Takes user input
- Passes user input to controlling layer (server)
- Displays system messages

The Main Subsystem

- Receives user queries through the UI subsystem
- Sends system responses to the UI subsystem
- Handles user, creation and management
- Sends DB update requests for the User and Map objects to DB subsystem

The driver and vehicle competency assuring Subsystem

- Provides driver and vehicle information
- Issue bolo for vehicle

- Issue titre for owner of vehicle
- Issue driver license
- Renew titre and driver license

The payment Subsystem

- View charge information
- Receives payment

CHAPTER FIVE

5. SYSTEM IMPLEMENTATION AND RESULT

5.1 Introduction

Implementation of the system involves writing a computer program, testing and eventually change over to the new system. It also spells out how the entire system or program is and how it works both the arrangement of the program flowchart and pseudo code and equally the program design itself. The source program is also attached as well as some print outs from the program.

5.2 Algorithms Used

Pseudo code is a kind of structured English for describing algorithms. It allows the designer to focus on the logic of the algorithm without being distracted by details of language syntax. At the same time, the pseudo code needs to be complete. It describes the entire logic of the algorithm so that implementation becomes a rote mechanical task of translating logic into source code. A line of pseudo code translates to one or many lines of actual code.

In general the vocabulary used in the pseudo code should be the vocabulary of the problem domain, not of the implementation domain. The pseudo code is a narrative for someone who knows the requirements (problem domain) and is trying to learn how the solution is organized.

1. Login algorithm

- 1. Get username*
- 2. get password*
- 3. if (username == username && password == password) then*
- 4. login successful*
- 5. Verify user role*
- 6. if (user role == admin) then*
- 7. display admin page*
- 8. elseif (user role == technician) then*

9. *display technician pagae*
10. *elseif(user role== traffic police v use role== field officer) then*
11. *display charge page*
12. *elseif(user role === finanncce clerk)then*
13. *display finance page*
- else*
14. *Login failed.*
15. *End if*
16. *Got 1*
17. *Stop*

5.3 Methods Used

We used an Object Oriented methodology to divide the system in to different modules and the system has three tire architectures. The architecture divide into three layers/tires Model, View and controller. Or presentation tier, UI, and business logic.

The original request is handled by a servlet

- **Model:** The servlet invokes the business-logic and data-access code and creates beans to represent the results. That's the model.
- **View:** The servlet decides which JSP page is appropriate to present those particular results and forwards the request there (the JSP page is the view)
- **Controller:** The servlet decides what business logic code applies and which JSP page should present the results (the servlet is the controller)

5.4 Result

After the project implementation process a number of academic knowledge learned from the Java Ecommerce Model was found very useful in the real life practice. In detail, both the three tiered web architecture and the MVC design model were successfully applied to the system. The technologies suitable for each tier were also practically used to implement the required functions in the system. For instance, the database pooling technology was found to be the best solution on the database connection and JSP plus JavaBeans played a good role in the task of data rendering. All these findings are valuable experience for our future study.

6. SYSTEM TESTING

6.1 Test Plan

6.1.1 Introduction

Testing is a program consists of providing the program with a set of test input (or test cases) and observing if the program behaves as aspect, then the conditions under which a failure occurs are noted for debugging and the correction.

Test is to determine whether it meets the uses requirement. This is the last chance to detect and correct errors before the system is installed for user acceptant testing. The purpose of the system testing is to consider all the likely variations to which it will be subjected and then push the system to its limits.

Testing is vital to the process of the system. System testing makes a logical assumption that if all the parts of the systems are correct, the goal will be successfully achieved. Inadequate testing leads to error, which may not appear until the product is implemented at the real world.

6.1.2. Test Plan Overview

Test plan is a general document for entire project, which defines the scope, approach to be taken and the personal responsible for different activities of testing. The inputs for forming test plane are

- Requirements document
- System design

A) Relationship Between Requirements Analysis Document and Test Plan

The functional requirements (Use Cases) recognized in Requirements Analysis are mapped to Functional Test Cases listed in Test Cases as follows:

Table 6- 1 Functional Test Plan

Functional Requirement (Use Cases Name)	Test Case ID
Manage User	FT_01,FT_2,
issue libre	FT_03
Register trainee drivers	FT_04
Issue Bolo	FT_05
Issue driver license	FT_06
Charging	FT_07
Transfer Owner	FT_08

B) Relationship between Software Architecture Document and Test Plan

The sub-systems recognized in Software Architecture Document are mapped to Integration Test Cases listed in Test Cases as follows:

Table 6- 2: Integration Test case Plan

Sub-Systems	Test Case ID
UI Interface Subsystem	IT_01
User Management Subsystem	IT_02
Admin Operations Subsystem	IT_03
Persistent Data Storage Subsystem	IT_04

C) Relationship between Object Specification Document and Test Plan

The packages recognized in Software Architecture Document are mapped to Unit Test Cases listed in Test Cases as follows:

Table 6- 3: Unit Test case Plan

Packages(Class)	Test Case ID
Libre	UT_01, UT_02,UT_03
DriverLicense	UT_04,UT_5,UT_6,UT_7
Bolo	UT_08, UT_09, UT_10
Charge	UT_11, UT_12, UT_13, UT_14

6. 1.3 Features to be tested /not to be tested

6.1.3.1 Features to be tested

This project is tested by breaking in to different modules. This module is tested separately. The module to be tested are creating user account, owner registration, vehicle registration, driver registration, and issuing libre and issuing Bolo.

6.1.3.2. Features not to be tested

A) Time the system came up after goes down.

This feature cannot be tested as we cannot simulate an actual server crash and try and recover it.

B) How does the system performance vary during high loads?

This non-functional feature of performance under higher loads could not be tested due to lack of time and resources to generate large volumes of incident report requests

C) Does the system perform optimally under low/normal load?

It is hard to determine in short time as to what should equate to optimal performance. Hence this feature cannot be tested. Also it is hard to saw what low and normal load is.

D) Is the system portable across different platforms?

This could be tested, as the system could not be deployed across different platforms for testing due to lack of time.

6.1.4 Approach

The project members use different testing methodology to testing the system. Those are black box testing and white box testing. White box testing is a testing technique that uses to check the internal mechanism of the system. Black box testing is ignores the internal mechanism of the system and it focuses on the output generated against any input and execution of the system.

Once code has begun, the testing process can begin and proceed in parallel. Each program module is produced, it can be tested individually, as part of the larger program, and then as part of larger system. The following are different testing strategies.

❖ Unit Testing

It will be done by the developer and will be approved by the development team lead. Proof of unit testing (test case list, sample output, data printouts, and defect information) is provided by the development team. This must be provided by the programmers to the development team lead before unit testing will be accepted and passed on to the testing team. All unit test information will also be provided to the test team.

❖ System/Integration Testing

This will be performed by the test team and with the knowledge from the individual developers who were also involved with developing the system is required. No specific test tools are available for this project. Programs will enter into System/Integration test after all critical defects have been corrected (In the unit testing). A program may have errors only in the interfacing level and should not have any errors in the internal functionality.

6.1.5 Testing Material

Hardware requirement

- 3 Pentium machines, each having 1GB RAM, 40GB HDD
- Switch for creating network

Software requirements

- Windows XP, windows 7 and 8
- NetBean was used to test java codes
- No special software was needed to test the system

6.2 Test Case Specification

Although there is one test plan for entire project test cases have to be specified separately for each test case. Test case specification gives for each item to be tested. All test cases and outputs expected for those test cases.

6.2.1 Unit Test Cases

Table 6- 4: Libre Unit Test Case

Test case specification identifier	UT_1
Test items	createLibre()
Input specifications	ownerId,vechicleId
Output specifications	Generate libre correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Inter case dependencies	

Table 6- 5: Libre Unit Test Case

Test case specification identifier	UT_2
Test items	transferLibre()
Input specifications	ownerId,vechicleId and newOwnerId
Output specifications	Transfer libre to new owner correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	
Inter case dependencies	

Table 6- 6: Libre Unit Test Case

Test case specification identifier	UT_3
Test items	changeLibre()
Input specifications	vechicleId
Output specifications	Create a new libre correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	
Intercase dependencies	

Table 6- 7: Libre Unit Test Case

Test case specification identifier	UT_4
Test items	transferLibre()
Input specifications	ownerId,vechicleId and newOwnerId
Output specifications	Transfer libre to new owner correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Intercase dependencies	

Table 6- 8: DriverLicense Class Unit Test

Test case specification identifier	UT_5
Test items	issueDriveLicense()
Input specifications	driveId,examId,levelId
Output specifications	Generate driver license correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Intercase dependencies	

Table 6- 9: DriverLicense Class Unit Test

Test case specification identifier	UT_6
Test items	renewDriveLicense()
Input specifications	DriverId
Output specifications	Activate license correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Intercase dependencies	

Table 6- 10: DriverLicense Class Unit Test

Test case specification identifier	UT_7
Test items	validateDriverLicense()
Input specifications	DriverId
Output specifications	Identify valid driver license
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Intercase dependencies	

Table 6- 11: Transfer Libre Class Unit Test

Test case specification identifier	UT_8
Test items	transferLibre()
Input specifications	ownerId,vehicleId and newOwnerId
Output specifications	Transfer libre to new owner correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Intercase dependencies	

Table 6- 12: Issue Bolo Class Unit Test

Test case specification identifier	UT_9
Test items	issueBolo()
Input specifications	vechicleId, insp_orgid,
Output specifications	Generate Bolo correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Intercase dependencies	-

Table 6- 13: Validate Bolo Class Unit Test

Test case specification identifier	UT_10
Test items	validateBolo()
Input specifications	BoleId
Output specifications	Generate new Bolo and activate correctly
Environmental needs	Windows XP, Netbean, Glassfish
Special procedural requirements	None
Intercase dependencies	

6.2.2 Functional Test Cases

Table 6- 14: Manage User Test Case

Test case specification identifier	FT_01
Test items	User Management, Persistent Data Storage
Input specification	New user joining the system
Output specification	Successful addition
Environmental needs	window XP, 7/8 Mozilla firefox
Special procedural requirements	

Table 6- 15: Login Test Case

Test case specification identifier	FT_2
Test item	User Management, Persistent Data Storage
Input specification	Staff provides a valid user name and password
Output specification	If information correct, homepage is displayed
Environment needs	Windows xp,7, and 8, Glassfish
Inter case dependencies	Active status
Special procedural required	

Table 6- 16: Issue Bolo Test Case

Test case specification identifier	FT_3
Test item	Bolo Management , Persistent Data Storage
Input specification	Owner , vehicle
Output specification	Issue bolo
Environment needs	Windows xp, 7/8 mozilla firefox
Inter case dependencies	
Special procedural requirements	

Table 6- 17: Register Trainee Driver Test Case

Test case specification identifier	FT_4
Test item	Driver license , Persistent Data Storage
Input specification	New trainee driver
Output specification	Successful registration
Environmental needs	Windows xp, 7/8 mozilla firefox
Special procedural requirements	
Inter case dependencies	

Table 6- 18: Issue driver's license test case

Test case specification identification	FT_06
Test item	driver Management , Persistent Data Storage
Input specification	Trainee driver with pass result
Output specification	Issue driver license
Environmental needs	Windows xp, 7/8 mozilla firefox
Special procedural requirements	
Inter case dependencies	

Table 6- 19: Driver Charge case test

Test case specification identifier	FT_07
Test item	Charging management , Persistent Data Storage
Input specification	Charge, driver
Output specification	Charge case payment is generated
Environmental needs	Windows xp, 7/8 mozilla firefox
Special procedural requirements	
Inter case dependencies	

Table 6- 20: Issue libre

Test case specification identifier	FT_08
Test item	Libre management, Persistent Data Storage
Input specification	New owner, vehicle , old owner
Output specification	Ownership is transferred
Environmental needs	Windows xp, 7/8 mozilla firefox
Special procedural requirements	
Inter case dependencies	

6.2.3 Sample Login Test Case Result

In this section we explain the Logon test cases that are used during testing

- *Test Case Specification Identifier:*
 - Login
- *Test Items: FT-2*
- *Components under Test:*
 - o Logon.jsp
 - o Ownershipregistration.jsp
 - o Controller class/object
 - o User class/object
 - o Database
- *Features Being Exercised:*
 - o *Valid User Login* - correctly logged in with correct username and password
 - o *Valid User Name* - unable to log in with correct username and invalid password
 - o *Valid User Password* - unable to login with invalid username and correct password
 - o *Invalid User Login* - unable to login with invalid username and invalid password
 - o *Invalid resource* - correctly logged with correct username and password, not cannot access the page
 - o *Valid resource* - access page with correct username and password
- *Input Specifications:*
 - o Use Identification
 - o Password
- *Output Specifications:*
 - o Upon valid login, user will be sent the system screen
 - o Upon invalid login, messages boxes with appropriate messages denoting the reason(s) for invalid login
- *Environmental Needs:*
 - o MySQL Database connection to a MySQL database with a table set up for users with users already entered
- *Special Procedure Requirements:*
 - o ATM&TS System must be running

Example 1: For result of the user login test case .We enter user name and password

Test Case # FT-2

Test Case Name - User Logi

Test Items – Functionality of the User Login screen

Input Specifications –

- Valid UserName - admin
- Valid Password – adminadmin
- Invalid UserName – dak
- Invalid Password - 233

Output Specifications –

Input	Expected Output
Valid UserName + Valid Password + Valid resource	Display ownershipregistration .jsp
Invalid UserName + Valid Password	Display Error Message
Valid UserName + Invalid Password	Display Error Message
Invalid UserName + Invalid Password	Display Error Message

Chapter Seven

7. CONCLUSION

7.1 Conclusion

This project is aimed to develop web-based application that named Adama Transport monitoring and traffic Web-based system. It enables user's access this system from anywhere through internet by pc or mobile.

In the previous chapters, we were intended in analyzing the manual system of the transport agency in Adama Town only up to proposing new system (web-base system) that solves the difficulties related to the manual system that was briefly described in the previous chapters. After a detail study of the manual and desktop based system of Adama transport agency, models have been designed to reflect the web-based system that is supposed to solve the problems identified.

By having this over the manual and desktop based system our aim was to build a new web-based application system that have greater functionality that enhance effectiveness and efficiency related parameters on the system.

The new system, it facilitates activities such as registration, issue bolo, libre and driver's license and traffic police or, retrieval of charging, owners, drivers and vehicle like information. Also the system avoids loss of files, reduces high storage space requirement and generally it increases the effectiveness and efficiency of the system as a whole.

This application stores all the information in the database which can be retrieved whenever needed and all the validations are performed during the entry of the data by the user thus ensuring that the user can not enter any wrong data which could cause problem later. In developing this project all group members contributed their full capability with maximum interest and all group members get ways toward developing a project.

The team believe that the different tools and techniques has helped a lot in capturing real user requirements and model and design the right system for the users for their day to day activities.

To achieve our goal to design new system the project team has spent all of its time on the project on performing the tasks individually and in group based on the schedule given by the department.

7.2 Results Achieved

This new system has high security, very fast and cost effective and reliable .It can be applied on different operating system due to the java features technology .it function or operation to be apply on transport and traffic service such as driver and vehicle competence assuring and transport traffic process assuring.

As the result of this newly developed system owner and driver of information will be secured.

1.3 Limitation Of The Project

The Adama transport monitoring and traffic web based system is limited to driver, owner and vehicle registered in Adama city that concerning mainly issue bolo and libre for owner and issue driver's license for driver which having high security but for registered driver and vehicle pass or use the Adama city road those who are not registered in Adama. The system could not find out information such vehicle and driver but the system charge driver by traffic police neither driver data nor vehicle data.

This system does not print plate number on metal .The software does not inspect vehicle. Inspection of vehicles is done by inspection officers who then issue the owner road worthy certificate. This software does not process bank transactions.

7.4 Future Works

According to scope of our project the team develops web application .Because of the time constraint we could not do beyond to our scopes, but in the future the team believes that this system can be fully operational by having enough time and fully information.

The team would recommend that further work should done on the system in order to make the system perform better

The Adama transport monitoring and traffic web based system is newly developed system. For this reason many thing are expected to be arranged and fulfilled for successful implementations of it. The project team recommendations are stated as follows.

1. Adama Transport agency to implement the project by providing server, computer, printer, other importance device and internet infrastructure.
2. Adama Transport agency, to staff technician person for system administrator who can maintain the system frequently.

3. Adama Transport web based system can upgrade to connect driver and vehicle competency assuring process and transport traffic assuring process to store the day to day activity or service of drivers and vehicles.
4. The project members believe that development of this project is the motivation for the person that needs to develop this like web base system. We allow or want to recommend any person can update, and modify this project.
5. The next developer fully integrated Adama transport monitoring and traffic web based system with city vehicle association and with federal, region, with other city and wereda transport agency or office can work and upgrade it functionality.

ANNEX: A

Sample Input/ Output Screens

File Edit View History Bookmarks Tools Help

ADAMA TRANSPORT AGENCY x +

http://localhost:8080/ACTAT/newowner.jsp

ADAMA TRANSPORT AGENCY

Register Bolo Issue lebre Driver License Transfer

Owner Registration

Owner Id:

First Name

Middle Name

Last Name

Sex

DOB

Region

Zone

wereda

kebele

Next

1:36 PM 6/8/2015

ANNEX: B

Sample Codes

Newowner.jsp

```
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
```

```
<%@taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql"%>
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<% String path=request.getContextPath();
```

```
String
```

```
basePath=request.getScheme()+"://"+request.getServerName()+":"+request.getServerPort()+"/";
```

```
%>
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
```

```
"http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
<link rel="stylesheet" type="text/css" href="css/acta.css">
```

```
<base href="<%=basePath%>"
```

```
<title>ADAMA TRANSPORT AGENCY</title>
```

```
</head>
```

```
<body>
```

```
<div id="main">
```

```
<div id="adminMenu">

  <ul id="nav">

    <li>

      <a href="#"> Register</a>

    <ul>

      <li>

        <a href="#"> Vehicle</a>

      </li>

      <li>

        <a href="#"> Owner</a>

      </li>

      <li>

        <a href="#"> Driver</a>

      </li>

    </ul>

  </li>

  <li>

    <a href="#"> Bolo</a>

    <ul>

      <li>

        <a href="#"> Issue Bolo</a>

      </li>

      <li>
```

 Renew Bolo

Issue libre

Driver License

 Issue License

 Renew License

Transfer

 Ownership Change


```

        <a href="#"> Vehicle Body change</a>

    </li>

    <li>

        <a href="#"> Vehicle Service Type Change</a>

    </li>

</ul>

</li>

</ul>

</div>

<div class="clear"></div>

<div id="indexLeftColumn">

    <h3> Owner Registration </h3>

    <form action="http://localhost:8080/ACTAT/NewOwner" method="post">

        <table border="0">

            <tr></tr>

            <tr>

                <td>Owner Id:</td>

                <td><input type="text" name="ownerid" value="" /></td>

            </tr>

            <tr>

                <td>First Name</td>

                <td><input type="text" name="ofname" value="" /></td>

            </tr>


```

```

<tr>

<td>Middle Name</td>

<td><input type="text" name="omname" value="" /></td>

</tr>

<tr>

<td>Last Name</td>

<td><input type="text" name="olname" value=""></td>

</tr>

<tr>

<td>Sex </td>

<td><input type="text" name="osex" value=""></td>

</tr>

<tr>

<td>DOB</td>

<td><input type="Date" name="odob" value="MM/DD/YYYY" ></td>

</tr>

<tr>

<td>Region</td>

<td><input type="text" name="oregion" value="" ></td>

</tr>

<tr>

<td>Zone</td>

<td><input type="text" name="owereda" value="" ></td>

```

```

        </tr>

        <tr>

            <td>wereda</td>

            <td><input type="text" name="owereda" value="" ></td>

        </tr>

    <tr>

        <td>kebele</td>

        <td><input type="text" name="owereda" value="" ></td>

    </tr>

    <tr>

        <td>Tel</td>

        <td><input type="text" name="okebele" value="" ></td>

    </tr>

    <tr>

        <td></td>

        <td><input type="submit" value="Next " ></td>

    </tr>

</table>

</form >

</div>

<div id="tecIndexRightColumn">

    </div>

</div>

```

</body>

</html>

Newowner.jsp servlet

```
package controller;
```

```
import java.io.IOException;
```

```
import java.io.PrintWriter;
```

```
import java.util.Date;
```

```
import javax.servlet.RequestDispatcher;
```

```
import javax.servlet.ServletConfig;
```

```
import javax.servlet.ServletException;
```

```
import javax.servlet.annotation.WebServlet;
```

```
import javax.servlet.http.HttpServlet;
```

```
import javax.servlet.http.HttpServletRequest;
```

```
import javax.servlet.http.HttpServletResponse;
```

```
import model.OwnerManager;
```

```
import bean.Owner;
```

```
import java.text.ParseException;
```

```
import java.text.SimpleDateFormat;
```

```
import java.util.logging.Level;
```

```
import java.util.logging.Logger;
```

```
import javax.ejb.EJB;
```

```

@WebServlet(name="NewOwner",urlPatterns = {"/NewOwner"})

public class NewOwner extends HttpServlet {

    public NewOwner() {

        System.out.println("object created new owner....");

    }

    OwnerManager mgr;

    @Override

        public void init(ServletConfig c){

            mgr=new OwnerManager();

        }

    @Override

        protected void doGet(HttpServletRequest request, HttpServletResponse response)

            throws ServletException, IOException {

                doPost(request, response);

            }

    @Override

        protected void doPost(HttpServletRequest request, HttpServletResponse response)

            throws ServletException, IOException {

                Owner owner=new Owner();

                RequestDispatcher rd;

                owner.setOwnerid(request.getParameter("ownerid"));

```

```

        owner.setOfname(request.getParameter("ofname"));

        owner.setOmname(request.getParameter("omname"));

        owner.setOlname(request.getParameter("olname"));

        owner.setOsex(request.getParameter("osex"));

    try {

        Date dob = new SimpleDateFormat("MM/dd/yyyy").parse(request.getParameter("odob"));

        owner.setOdob(dob);

    } catch (ParseException ex) {

    }

    owner.setOregion(request.getParameter("oregion"));

    owner.setOzone(request.getParameter("ozone"));

    owner.setOwerede(request.getParameter("owereda"));

    owner.setOregion(request.getParameter("oregion"));

    if (mgr.insertData(owner))

    {

        rd=request.getRequestDispatcher("/index.jsp");

        rd.forward(request, response);

    }

}

}

```

Owner.java

```
package bean;

import java.io.Serializable;

import java.util.Collection;

import java.util.Date;

import javax.persistence.Basic;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.NamedQueries;

import javax.persistence.NamedQuery;

import javax.persistence.OneToOne;

import javax.persistence.Table;

import javax.persistence.Temporal;

import javax.persistence.TemporalType;

import javax.validation.constraints.NotNull;

import javax.validation.constraints.Size;

import javax.xml.bind.annotation.XmlRootElement;

import javax.xml.bind.annotation.XmlTransient;

public class Owner implements Serializable {

    private String ownerid;
```

```
private String ofname;

private String omname;

private String olname;

private String osex;

private Date odob;

private String oregion;

private String ozone;

private String owereide;

private String okebele;

private Collection<Vehicle> vehicleCollection;

public Owner() {

}

public Owner(String ownerid) {

    this.ownerid = ownerid;

}

public String getOwnerid() {

    return ownerid;

}

public void setOwnerid(String ownerid) {

    this.ownerid = ownerid;

}

public String getOfname() {

    return ofname;
```

```
}

public void setOfname(String ofname) {

    this.ofname = ofname;

}

public String getOmname() {

    return omname;

}

public void setOmname(String omname) {

    this.omname = omname;

}

public String getOlname() {

    return olname;

}

public void setOlname(String olname) {

    this.olname = olname;

}

public String getOsex() {

    return osex;

}

public void setOsex(String osex) {

    this.osex = osex;

}

public Date getOdob() {
```

```
        return odob;
    }

    public void setOdob(Date odob) {

        this.odob = odob;
    }

    public String getOregion() {

        return oregion;
    }

    public void setOregion(String oregion) {

        this.oregion = oregion;
    }

    public String getOzone() {

        return ozone;
    }

    public void setOzone(String ozone) {

        this.ozone = ozone;
    }

    public String getOwerede() {

        return owerede;
    }

    public void setOwerede(String owerede) {

        this.owerede = owerede;
    }
}
```

```

public String getOkebele() {

    return okebele;

}

public void setOkebele(String okebele) {

    this.okebele = okebele;

}

@XmlTransient

public Collection<Vechicle> getVechicleCollection() {

    return vechicleCollection;

}

public void setVechicleCollection(Collection<Vechicle> vechicleCollection) {

    this.vechicleCollection = vechicleCollection;

}

```

@Override

```

public int hashCode() {

    int hash = 0;

    hash += (ownerid != null ? ownerid.hashCode() : 0);

    return hash;

}

```

@Override

```

public boolean equals(Object object) {

    // TODO: Warning - this method won't work in the case the id fields are not set

    if (!(object instanceof Owner)) {

```

```

        return false;

    }

    Owner other = (Owner) object;

    if ((this.ownerid == null && other.ownerid != null) || (this.ownerid != null &&
!this.ownerid.equals(other.ownerid))) {

        return false;

    }

    return true;

}

@Override

public String toString() {

    return "bean.Owner[ ownerid=" + ownerid + " ]";

}}

```

Bolo Form

6054 88582

በኢትዮጵያ ፌዴራላዊ ዲሞክራሲያዊ ሪፑብሊክ
FEDERAL DEMOCRATIC REPUBLIC OF ETHIOPIA
የመሠረት ልማት ሚኒስቴር
MINISTRY OF INFRASTRUCTURE
የመገናኛ ተራጓዢ ጉዞ ባለሥልጣን
ROAD TRANSPORT AUTHORITY
የተሽከርካሪዎች ዓመታዊ የቴክኒክ ምርመራ ስርተፈካት
VEHICLES ANNUAL TECHNICAL INSPECTION CERTIFICATE

- 1.1 የምርመራ ቦታ _____
PLACE OF INSPECTION
- 1.1.1 የባለንበረት ስም _____
OWNER'S NAME
- 1.1.2 የሠላዳ ቁጥር _____
PLATE NUMBER
- 1.1.3 የሻሲ ቁጥር _____
CHASSIS NUMBER
- 1.1.4 የተሽከርካሪው ዓይነት _____
VEHICLE TYPE
- 1.1.5 የተሽከርካሪው ሞዴል _____
VEHICLE MODEL
- 1.1.6 የባለቤትነት መታወቂያ ደብተር ቁጥር _____
TITLE CERTIFICATE NUMBER
- 1.1.7 የተፈቀደለት የጭነት መጠን _____ በሰው
LICENSED CAPACITY SEAT
_____ ኪ.ግ
_____ ሊትር
_____ LITRE

1.2 የምርመራ ሠርተፈካት
CERTIFICATE OF INSPECTION

ተሽከርካሪው ተመርምሮ ጉድለት _____ በመሆኑ የቴክኒክ ምርመራ _____
 ተደርጓል።
 THE VEHICLE HAS BEEN INSPECTED AND FOUND / NOT FOUND _____
 AND ACCORDINGLY THE VEHICLE IS JUDGED TO _____ THE INSPECTION

የመርማሪው ስምና ፊርማ _____ የኃላፊው ስምና ፊርማ _____
 NAME AND INITIAL OF TECHNICIAN NAME AND INITIAL OF HEAD

ቀን _____ ቀን _____
 DATE DATE

በደረሰኝ ቁጥር _____ ተከፍሎ ተሰጣፊ ምልክት ቦሎ ቁጥር _____ ተሰጥቷል።

APPENDIX B

Libre Form

	Lak. Shaarsu/የጃንሰው ቁጥር <u>RN46-026940</u> Lak. Motorra/የተር ቁጥር <u>18R-2041363</u>
Maqaa Abbaa-Qabeenya Owner's Name የባለቤት ስም <u>ፖሊስ ጽ/ቤት ሰራተኛ</u>	GOSA KONKOLAATAA <u>PA7</u> የተሰርዘረው ዓይነት <u>553</u> BIYYA ITTI HOJJATAME የተሰራበት አገር <u>የጥ</u> KUBBAANTIYVAA OMIISHE አጥሪ ከገንዘብ MOODEELA KONKOLAATAA <u>RN4614</u> የተሰርዘረው ጥያቄ
Lak. Gabatee Plate No. የባለቤት ቁጥር <u>2-0396070</u>	BAHA HOJJATAME የተሰራበት ዘመን BAL'INA QAAMAA (m) የአካል ስፋት (ሜ) HOJJAA QAAMAA (m) የአካል ቀዝቃዛ (ሜ) DHEERINA QAAMAA (m) የአካል ርዝመት (ሜ) GUSA QAAMAA <u>PA7</u> የአካል ዓይነት QABIYYEE MOTORRA (cc) የተር ፍላጎት (ሊ)
Tajjajjila Itti Galmaa'ee Services አገልግሎት <u>የጥ ጽ/ቤት</u>	

Inspection Form

[illegible]

በመሠረተ ልማት ሚኒስቴር
Ministry of Infrastructure
የመንገድ ትራንስፖርት ባለሥልጣን
Road Transport Authority
የአዲስ ተሽከርካሪ ምርመራ ስርተፊኬት
New Vehicle Inspection Certificate

ስለ ተሽከርካሪው መግለጫ
VEHICLE DESCRIPTION

1. የተሽከርካሪው ዓይነት
Type of Vehicle _____
2. ተሽከርካሪው የተሠራበት
Vehicle Make _____
3. የተሽከርካሪው ሞዴል
Vehicle Model _____
4. የተሠራበት ዘመን _____ የተመረተበት አገር _____ ሻሲ _____ አካል _____
Years of Manufacture (G.C) _____ Country of Manufacture, Chassis _____ Body _____
5. የአካል ዓይነት
Body Type _____
6. የሰሌዳው መጠን /የፊትና የኋላ/
Size of Plate /front & rear/ _____
7. የሻሲ ቁጥር
Chassis Number _____
8. የአካሉ ስፋት /ሜትር/
Body Length (M) _____
9. የአካል ርዝመት /ሜትር/
Body Length /M/ _____
10. የአካሉ ቁመት /ሜትር/
Body Height /M/ _____
11. የአካሉ ቀለም
Body Colour _____
12. የሞተር ቁጥር _____ የሲሊንደር ብዛት _____
Engine Number _____ No. of Cylinders _____
13. የሞተር ችሎታ /ሲ.ሲ./
Engine Capacity (C.C) _____
14. የነዳጅ ዓይነት
Type of Fuel _____
15. የሞተር የፈረስ ገጽቦት
Engine Horse Power _____
16. የጎማው መጠን የፊት አክስል _____ የኋላ አክስሎች _____
Tyre Size Front Axle _____ Rear Axles _____
17. የአክስሎች ብዛት _____ የአክስል የአነዳጅ ዓይነት _____ የጎማ ብዛት _____
Number of Axles _____ Axle Drive Type _____ No of Tyres _____
18. የፊት አክስል ከፍተኛው ክብደት ኪ 1 _____ 2 _____ 3 _____ 4 _____
Front Axle Max. Load _____
19. የኋላ አክስሎች ከፍተኛው ክብደት ስ- _____
Rear Axle Max. Load _____
20. የተሽከርካሪው ጠቅላላ ክብደት ግ _____
Gross Vehicle Weight _____
21. ነጠላ ክብደት ም _____ ሰው (Person) _____
Tare Weight _____

APPENDIX : D

Driver license



APPENDIX: F

Interview questions

1. What is your job description?

2. What is the primary purpose of the agency ?

3. What type of tasks do you perform in a typical day?

4. What is the purpose of vehicle registration ?

5. How do register vehicle ?

6. What is the purpose of issuing bolo?

7. What type of things do you need to keep track of ?

8. What service do your agency provide to your customer?

9. Do you feel you need web based system?

10. What type of security does want for the systems?

11. What type of Staff should only see the data necessary to do their job in a form that suits what they're doing?

.....
.....
.....

For example: A technician sees owners, vehicles, drivers detail and should only be displayed one at a time not as report.

- I. What historical data do you want hold?
- II. What we want to keep detail of owners and vehicles?

BIBLIOGRAPHY

- [1]. Segni G cause of road traffic accident and possible counter measures on Addis Ababa– Shashemne road [Ms thesise]; Addis Ababa university 2014
- [2]. Road freight transport in ethiopia With special emphasis On addis ababa – djibouti corridor
By Asnake tadesse
- [3]. Building Web Applications with UML, 2nd Edition by Jim Conallen,
<https://books.google.com.et/books?isbn=0201730383>, and
<http://www.informit.com/articles/article.aspx?p=30610>
- [4]. Cooke, M.P. (2003), E-Commerce lecture notes, Department of Computer, Sheffield University.
- [5]. Bergsten, H. (2002) Java Server Pages (2nd Edition). O'REILLY
- [6]. International Conference on Science, Technology, Education, Arts, Management and Social Sciences iSTEAMS Research Nexus Conference, Afe Babalola University, Ado-Ekiti, Nigeria, May, 2014.
- [7]. An SMS and web-based traffic case management system in Bangladesh by Mahmud, mran; Roy, Premankur; Saha, Sangeet .
- [8]. The complete reference of HTML & CSS 5Th Edition, Thomas A, Powell
- [9]. Java how to program 9TH Edition, Paul Deitel, Harvey Deitel
- [10]. Professor Lonnie D. Bentley "System Analysis and Design methods" Simon Bennet, Steve McRobb and Ray Farmer: Object oriented system analysis and design