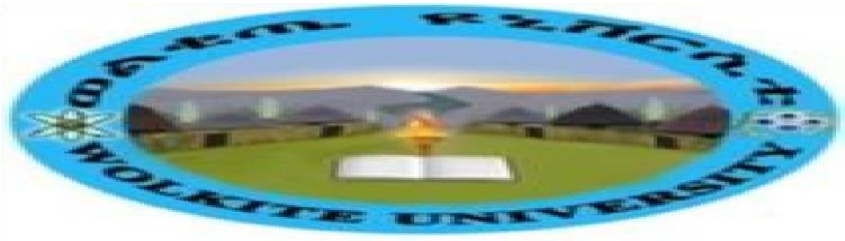


WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF INFORMATION SYSTEM

Industrial Project

On

WEB BASED GARAGE MANAGEMENT SYSTEM FOR AUTOCARE GARAGE P.L.C.

Submitted by:

- Moata Girma CIR/168/11
- Muse Asrat CIR/171/11
- Beamlaktibeb Getachew CIR/121/11

Project advisor: Mr. Abdu Amde

June 14, 2022

Wolkite University, Wolkite, Ethiopia

Wolkite University
College of Computing and Informatics
Department of Information System

Web Based Garage Management System

For

AutoCare Garage Plc.

Submitted by:

- Moata Girma CIR/168/11
- Muse Asrat CIR/171/11
- Beamlaktibeb Getachew CIR/121/11

Submitted to Department of Information System in Partial Fulfilment of the Requirement for the Degree of Bachelor of Science in Information System.

Project advisor: Mr. Abdu Amde.

June 14, 2022

Wolkite University, Wolkite, Ethiopia

Declaration

This is to declare that this project work which is done under the supervision of Mr. Abdu Amde and having the title Web Based Garage Management System for AutoCare Garage Plc. is the sole contribution of:

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

Group Members:

Full Name	Signature
_____	_____
_____	_____
_____	_____

Approval Form

This is to confirm that the project report entitled Web Based Garage Management System for AutoCare Garage Plc. Submitted to Wolkite University, College of Computing and Informatics Department of Information System.

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

Acknowledgments

First, we would like to thank almighty God who made it all possible. Also we would like to forward the deepest of our appreciation and gratitude to our advisor **Mr. Abdu Amde** for his patience and constructive advice.

Table of Contents

List of Table

List of Figure

List of Acronyms and Abbreviation

Admin	Administrator
BR	Business rule
CSS	Cascade style Sheet
DB	Database
GMS	Garage Management System
GUI	Graphical User Interface
LAN	Local Area Network
HTTP	Hypertext Transfer Protocol
OOAD	object-oriented analysis and design
PIECE	Performance, Information, Economical, Control, Efficiency
PHP	Hypertext Preprocessor
PLC	Public Limited Public
SDLC	Software Development Life Cycle
SQL	Structural Query Language
UML	Unified Modeling Language.
UC	Use case
UI	User Interface

Abstraction

Web based management system provides integration and optimization of various business processes that leads to improve the functional processes of an organization. The main purpose of this project was to develop web-based management system in order to solve the existing challenges of the manual garage management system of the AutoCare Garage Plc. For this purpose, the required data was collected using unstructured interview, direct observation and document reviews. In this project, the implementation of the web-based garage management system carried out using PHP, JAVASCRIPT, CSS and MYSQL for the database.

- **Introduction**
- **Objectives**
- **Methodology**
- **System analysis**
- **Implementation and testing**
- **Conclusion and recommendation.**

.

Chapter One

Introduction

1.1. Introduction

In the current technological era, information is so vital for the day to day decision of businesses and alike. Mobile, cloud, stock exchanges, commodity exchanges, financial institution are all connected through networks that are the integral part of message communication in the world now and in the future. The World Wide Web (WWW) is a techno-social system to interact humans based on technological networks. The use of web is growing at a phenomenal rate across the globe. Organizations increase their investment in and usage of web-based technology. The scope of web-based application has grown enormously and has moved to become a platform that can support all facets of organizational work[1]. The reason for the Web's success is largely due to its simplicity for use and information retrieval. Its nature of simplicity and usability makes web to be preferably used in any sector for management and transaction of information. As a basis for designing complex information systems the Web-technology has matured a lot over the last few years and are far more compatible across platforms[2].

1.2. Background of the Organization

AutoCare Garage Plc.is a licensed general automotive garage established to handle vehicle maintenance, repair and give consultancy services that is located in Addis Ababa City, Yeka Sub-city, Woreda 11.

1.2.1. Vision, Mission, and Objectives AutoCare Garage Plc.

1.2.1.1. Vision

AutoCare Garage P.L.C. aspires to be the leading Center of Excellence in modern automotive technology in Ethiopia by 2025.

1.2.1.1. Mission

The mission of AutoCare Garage P.L.C. is to provide standard modern automotive maintenance in order to solve technical problems of vehicles through technical support,

training, research and consultancy and make Ethiopia fully benefited from the auto industry.

1.2.1.3. Objectives

a) To provide standard maintenance for modern vehicles with skilled manpower and up-to-date equipment
b) To satisfy the need of customers by solving technical problems and provide technical advice.
c) To train auto technicians who can work in modern vehicle maintenance activities
d) To support drivers/ vehicle owners on modern vehicle handling through training
e) To prepare standard teaching materials, books and magazines to support auto related education and trainings in the country.
f) To provide consultancy services related to automotive technology to governmental and non-governmental organizations.

1.3. Statement of the Problem

AutoCare Garage Plc. has a large size garage which is giving full garage operational services for the vehicles of his customers including different organization. The existing function for handling the information of the garage is totally manual operation. This is time consuming and paper-based information handling. AutoCare Garage Plc. currently operate all its works using handwritten forms or slips stored in drover (files). The handwritten information cause some clashes in the records such as missing a particular slip or maintaining the records in those Garage. So, to avoid the problem arises, we need an automated system that keeps a track of all the records & related information.

The problems of the existing manual system can be listed as follows:

- It lacks flexibility in form and report generation.
- Insufficient record system, to manage vehicle service details and vehicle maintenance details.
- Garages are complicated to handle, and most of the time, they are understaffed with numerous tasks to complete in limited time. It is not simple to manage so many work requests at a time so there is need for proper mechanism for booking appointments. For example, there is one inspection machine in those garage, which is service that take around 2-3 hours for inspections one vehicles.

- There is no flexible mechanism for vehicle assigning.
- Inefficiency in store issue i.e. release of material for technician.
- Customer follow their vehicle status by coming physically which time and resource consuming.
- Garage cannot make their services fully more available to the public through the internet except by paying the cost of advertisement for a website that make online Advertising and Promotion and they rather make use of posters to advertise their services to the public, which is limited to few features as compare to a web base system.

This project is therefore to develop a web-based garage management system which can solve the above problems which in turn enhance the overall garage management system of the AutoCare Garage Plc.

1.4. Objective of the Project

1.4.1. General Objective

The General objective of this project is to design, develop and test web-based garage management system for AutoCare Garage Plc.

1.4.2. Specific Objectives

In addition to the general objectives, we provide some specific objectives in the following manner

- To develop web based system that can provide online booking and create repair estimate.
- To develop system that generate report from the different daily activities.
- To design user friendliness and interactive interface that is more attractive and promote AutoCare Garage Plc. services to the public.
- To enable facilitated vehicle assignment and store issues.
- To develop system that record all details of service and maintenance.
- To enable online payment.

1.5. Scope and Limitation of the Project

1.5.1. Project Scope

The project mainly focused on technical management aspects of AutoCare Garage Plc., which include service and maintenance management of the garage and our final scope is designs and develop a web-based system for the organization.

1.5.2. Limitation of the Project

The system we proposed doesn't covered different department that are found in AutoCare Garage Plc. related to technical aspects of the Garage due to time constraint, shortage of enough finance and technologies to cover the activity such as

- Among the service given by the company it doesn't include consultant and training service on automotive technology.

1.6. Feasibility Study

A feasibility study is an analysis of how successfully a project can be completed, accounting for factors that affect it such as economic, technical, and legal (sometimes political), operational and scheduling factors. Project managers use feasibility studies to determine potential positive and negative outcomes of a project before investing a considerable amount of time and money into it[3].

1.6.1. Technical Feasibility

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. The technology we need to build the system is available and we can capable to work with those technology, due this the project is feasible.

1.6.2. Operational Feasibility

This assessment involves undertaking a study to analyze and determine whether and how well the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development [3].

The proposed system is operationally feasible because

- The system satisfies the user need or requirement.
- Provide adequate throughput response time.
- Very easy to use.

1.6.3. Economic Feasibility

This examined typically to determine cost/benefits analysis of the project, this will helping the organization to determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

Economic feasibility can be tangible and intangible benefits perspective.

Tangible benefits

Since this project is going to web based system, there is:

- Reduction of cost for material that used for manual operation.
- Reduce time consumption by manual activities.
- Reduce space to store file cabinet not manual.

Intangible benefits

The intangible benefits we have pointed out the system development are the following

- Easily access information.
- Increase efficiency.
- Ensure data accuracy.

1.6.4. Legal Feasibility

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like county laws, data protection acts or social media laws.

Our project don’t contradict the above laws so, it is legally feasible.

1.6.5. Schedule Feasibility

Schedule feasibility is the process of assessing the degree to which the potential time frame and completion dates for all major activities within a project meet organizational deadlines and constraints for effecting change[3].

The plan is to finish our project in at the end of the academic year. Considering all the above statements we can say our project is feasible in terms of schedule.

1.7. Significance of the Project

This project help the information handling of the organizational information and create aware of the importance of web based system including the following.

Effective customer management

The system will help manage the customer in an effective way by recording its data and transactions in a structured way. This will also give an opportunity to garages to understand the customer and its needs in a better.

Effective data management

Physical data management through files and papers are things of the past. The system help Garage to keep and organize data in an efficient way so that search and retrieval becomes easy.

Effective time management

Many time consuming processes like generating various types of reports and form are just available at the click of a button, saving lot more time for the garages.

1.8. Beneficiary of the Project

Those who will primarily benefit from the new system and those who will be affected by the new system include:

To AutoCare Garage Plc.

- Promotes business.
- The system will improve information processing and retrieval as well as improve records management.
- Reduce space to store file cabinet not manual.

- Make flexible the working assign and Save time.

To customer:

- The customers can they save their time and resource by view their vehicle status through this system rather than come and ask for each ongoing repair and booking an appointment.
- The system is therefore be able to reduce the time consumption by offering online enquiries through displaying the detailed products and services offered by the Garage.

To store keeper:

- Can easily register, update & delete item.
- Easily manage whom accept what items in easily way.
- Make store issues with less/no error.

To service advisor:

- Help him to easily retrieve appointment that is booked before certain time.
- Easily assign vehicle to technician team.

To manager:

- Generate different form easily.
- View report easily.
- Easily assign vehicle to technician team.

To technical head:

- Record repairing/maintenance under his team in proper manner.
- Generate reports easily.

Project team members

- Help us complete our degree as it is a partial requirement for our degree.
- Increase our knowledge on the web development.
- Gives us experience regarding project development.

1.9. Methodology of the Project

The methodology is the analysis of the principles or procedures of inquiry in a particular field of study. When we relate it with our project it is the system on how we will perform information gathering, analyze and design the system, implement, test and evaluate it using different ways.

1.9.1. System Analysis and Design Methodology

Among the different methodologies available, we decided to use object-oriented analysis and design (OOAD) methodology for this project. We chose object-oriented analysis and design (OOAD) methodology because of:

- Increased extensibility,
- Increased reusability,
- Improved quality,
- Lesser maintenance cost,
- Less complexity
- Increased chance of project success.

1.9.2. Data Collection Method

Data collection is a systematic approach to gathering and measuring information from a variety of sources to get a complete and accurate picture of an area of interest. We have chosen the following data collection methods to conduct our project.

Observations

We used this method to obtain information about current functional process of the AutoCare Garage Plc.

Interview

For our project to be successful we used unstructured interview. The interviewees are the AutoCare Garage plc. Service advisor, mechanics, and store keepers.

Document analysis

We analysis different documents that are related to our project such as work related to E-Garage, Garage management system and some AutoCare Garage Plc. document

1.9.3. System Developmental Model

The software development life cycle **(**SDLC) we are going to use is Iterative model which focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader features set until the final system completes[4].

It has also the benefits or advantages of:

- It is easily adaptable to the ever changing needs of the project as well as the client.
- It is more cost effective to change the scope or requirements in Iterative model.
- Risks are identified and resolved during iteration; and each iteration is an easily managed.

1.9.4. System Testing Methodology

Testing is the process by which a system or components are compared against requirements and specifications according to the user requirement. There are different types of testing methods; among them for this project three levels of testing conducted.

Unit testing

Each unit test is tested separately before integrating them into modules to test the interfaces between modules. While implementing the code all small components such as functions, conditional statements, loops, and others were tested.

Integration Testing

It is the activity of finding faults when testing the individually tested components together, all or some of the components of the project developed individually would be combined together and tested as a whole system to test the functional and performance requirements on the major items of the system. The integration testing generally follows unit testing where each module is tested as a separate unit.

System Testing

System testing tests all the components together, seen as a single system to identify errors with respect to the scenarios from the problem statement and the requirements and design goals identified in the analysis and system design, respectively.

The goal of system testing is to detect faults that can only be exposed by testing the entire integrated system or some major part of it.

1.9.4.1. Testing Tools and Environment

XAMPP

Open-source software developed by Apache friends XAMPP software package contains Apache distributions for Apache server, Maria DB, PHP, and Perl. And it is basically a local host or a local server. This local server works on your own desktop or laptop computer. The use of XAMPP is to test the clients or your website before uploading it to the remote web server. This XAMPP server software gives you the suitable environment for testing MYSQL, PHP, Apache and Perl projects on the local computer [5].So we use XAMPP to test our proposed system.

1.9.5. Development Tool and Technologies (Development Environment)

1.9.5 .1. Frontend Technologies

We use the following frontend technologies to develop our proposed system:

JavaScript

JavaScript a lightweight, cross-platform, and interpreted scripting language. It is well-known for the development of web pages, many non-browser environments also use it. JavaScript can be used for Client-side developments as well as Server-side developments.

HTML

It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages.

CSS

Cascading style sheets (CSS) is used to optimize pages for responsive web design and to generate more advanced graphics such as hover effects, in addition to adjusting simple things like colors, fonts, and spacing. A style sheet instructs the web browser on how to display the content in question.

1.9.5 .2. Backend Technologies

We use the following backend technologies to develop our proposed system

PHP

PHP is a server-side, open-source, object-oriented scripting language. PHP is a popular web development language. Also programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web based software applications.

MySQL database

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software.

1.9.5.3. Development Tool

- Visual studio.
- Apache Server.

1.9.5 .4. Documentation and Modeling Tools

Documentation Tools

- MS office word 2013
    - We used for documentation of the project.
- MS office PowerPoint 2013
    - Used for preparation of the project presentation.
- Mendeley software.
    - For reference

Modeling Tools

- Visual paradigm 16.3.
- Draw.io.
- Adobe Photoshop.

1.9.5 .4. Deployment Environment

Our system is compatible with every operating system platform. It will support all operating systems and can be deployed on them.

1.10. Organization of the Project

Generally, the paper is organized into seven chapters. The first chapter starts with general information about the project, back ground of the organization followed by statement of the problem and continued with the objective and scope of the project, feasibility study, significance and beneficiary of the project, and methodology of the project. The second chapter is looking into existing system, user of existing system, major function of existing system, weakness and business rule followed by the organization. The third chapter is about proposed system which presents the functional & non-functional requirements of the project. The fourth chapter deals with system analysis using Unified Modeling Language (UML) diagram such as use case diagram, sequence diagram, activity diagram, state chart diagram and class diagram. The five chapter deals brief overview of the design goals, current and proposed software architecture, subsystem decomposition, and package and user interface design. The six chapter concerned with implementation and testing and finally highlight the conclusion and recommendation.

Chapter Two

Description of the Existing System

2.1. Introduction

Investigations of the existing system in use by various means provide the basic and necessary inputs for the system to be designed. This chapter will be devoted to understanding about the existing system. It broadly identifies user of existing system. And overview the existing system functionality, weakness and finally states business rule that followed by the existing system.

2.2. Introduction of Existing System

This portion highlight about understanding how the existing manual system performs its task in step by steps, In order to fully understand the existing system, we examine using unstructured interview, document analysis, and observed business process as well.

The existing system workflow process order/flow start as performs its tasks starting from the requisition come from the customer, then the Garage accept the work after they agreed. If customer need only service they simply registered and vehicle assigned to technician team; but if customer need a maintenance based on his/her request, vehicle will inspected by the technical personnel and then format is filled based on the information obtained from the customer or technical personnel i.e. a person who inspect the vehicle, then customer are registered for full maintenance by manager include customer job order is recorded, accordingly the work is assigned to different groups of technician according to their role, then technician request Store keeper items to repair the vehicle which is will be approved by their technician head, then the technician head record repairing and maintenance task, including detail problem, maintained part, used spare part and labor, then prepare report to the manager about ongoing task, after work is completed manager register ended maintained vehicle on vehicle delivery form, then customer is called and receive his/her vehicle by signing, that his order job has done and check the vehicle in a good condition and maintained. The various type of information is tracked with above work process.

2.3. Users of Existing System

In the existing system there are different users and listed as follow with their roles:

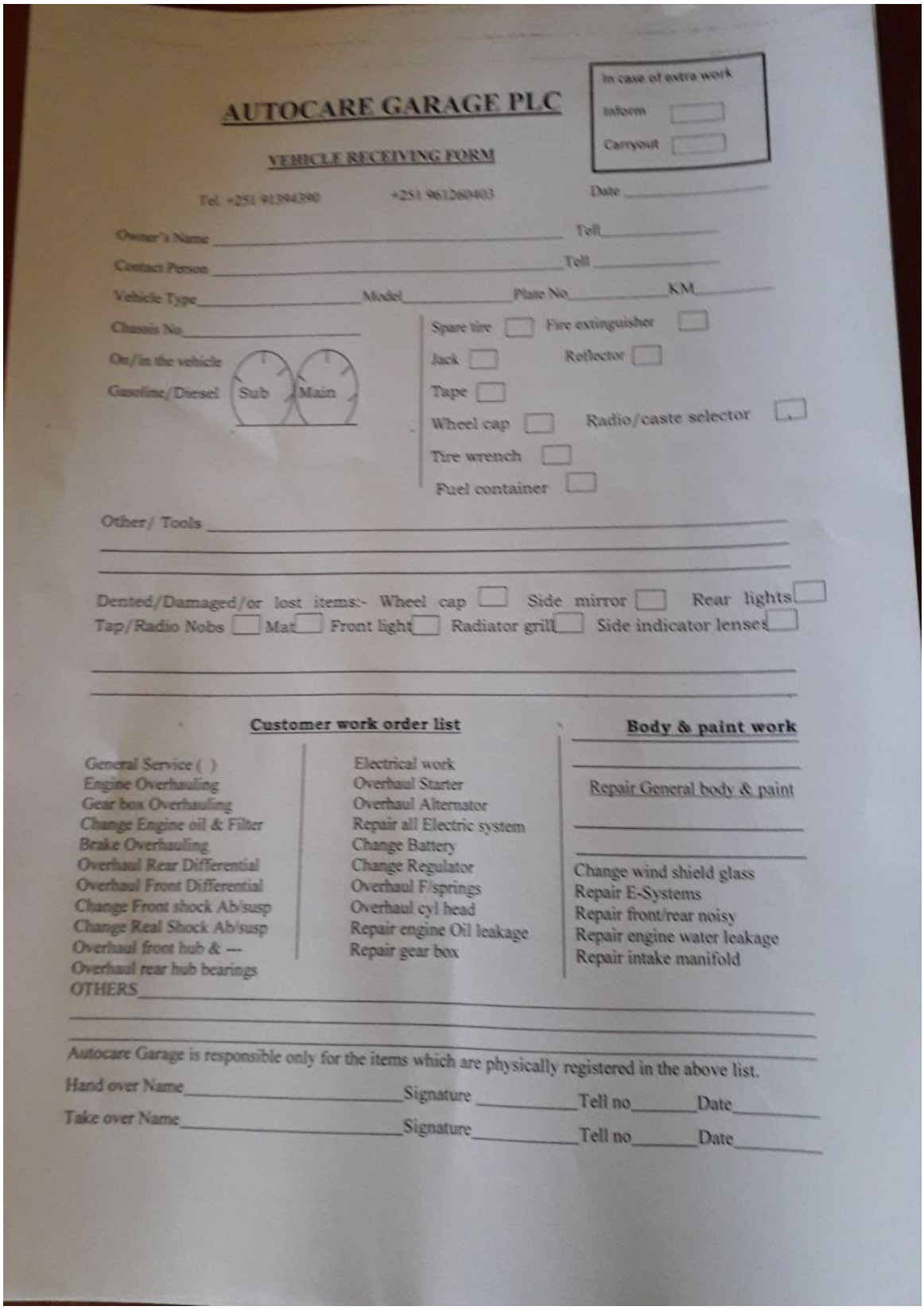
1. **Customer:** The customers role in the existing system:
    - Book an appointment physically or through call.
    - Make order for service or maintenance.
    - View and fellow up Vehicle status and make payment.
2. **Manager:** The roll of manager is he receive vehicle from customers if it's for heavy maintenance including crushed vehicle and assigned to technician team. In addition, he follow up over flow of works such as management overall activities of garage include
    - Receiving report from technician head.
    - Register delivered vehicle.
3. **Service Advisor:** The role of service advisor is to welcome customers & include the following role.
    - Register customer for services.
    - Scheduling service appointments.
    - Deal related to customers inquiries & complain.
    - Assign vehicle to different team.
4. **Technician head:** The roll of the Technician head he receive the vehicle from the service advisor or manager and his team trouble shoot & maintain the assigned vehicle, then after the work is finished or on progress he add detail the problem and how it solved also include the following activities.
    - Approve items requisition asked by technician under his team.
    - Record daily repairing/maintenance under his team.
    - Prepare report.
5. **Store keeper:** The role of store keeper:
    - To record item.
    - Check availability of the items.
    - To store issues to the technicians approved by his technician head
    - And finally generate Performa.

2.4. Major Functions of the Existing System

The major function of the existing system are:

- Customer information details.
- Record of vehicle Information with detail problem.
- Item records.
- Record maintenance status.
- Reports generation.
- Assign of vehicle.
- Store issues,
- Record accepted spare parts by technician.
- Register customer appointment.

2.5. Forms of the Existing Systems



AUTOCARE GARAGE PLC

VEHICLE RECEIVING FORM

In case of extra work
Inform
Carryout

Tel. +251 913394390 +251 9612604O3

Date _____

Owner's Name _____ Tell_____

Contact Person _____ Tell _____

Vehicle Type_____ Model_____ Plate No_____ KM_____

Chassis No_____

On/in the vehicle

Gasoline/Diesel Sub Main

Spare tire Fire extinguisher

Jack Reflector

Tape

Wheel cap Radio/caste selector

Tire wrench

Fuel container

Other/ Tools _____

Dented/Damaged/or lost items:- Wheel cap Side mirror Rear lights

Tap/Radio Nobs Mat Front light Radiator grill Side indicator lenses

Customer work order list

General Service ( )
Engine Overhauling
Gear box Overhauling
Change Engine oil & Filter
Brake Overhauling
Overhaul Rear Differential
Overhaul Front Differential
Change Front shock Ab/susp
Change Real Shock Ab/susp
Overhaul front hub & ---
Overhaul rear hub bearings

Electrical work
Overhaul Starter
Overhaul Alternator
Repair all Electric system
Change Battery
Change Regulator
Overhaul F/springs
Overhaul cyl head
Repair engine Oil leakage
Repair gear box

Body & paint work

Repair General body & paint

Change wind shield glass
Repair E-Systems
Repair front/rear noisy
Repair engine water leakage
Repair intake manifold

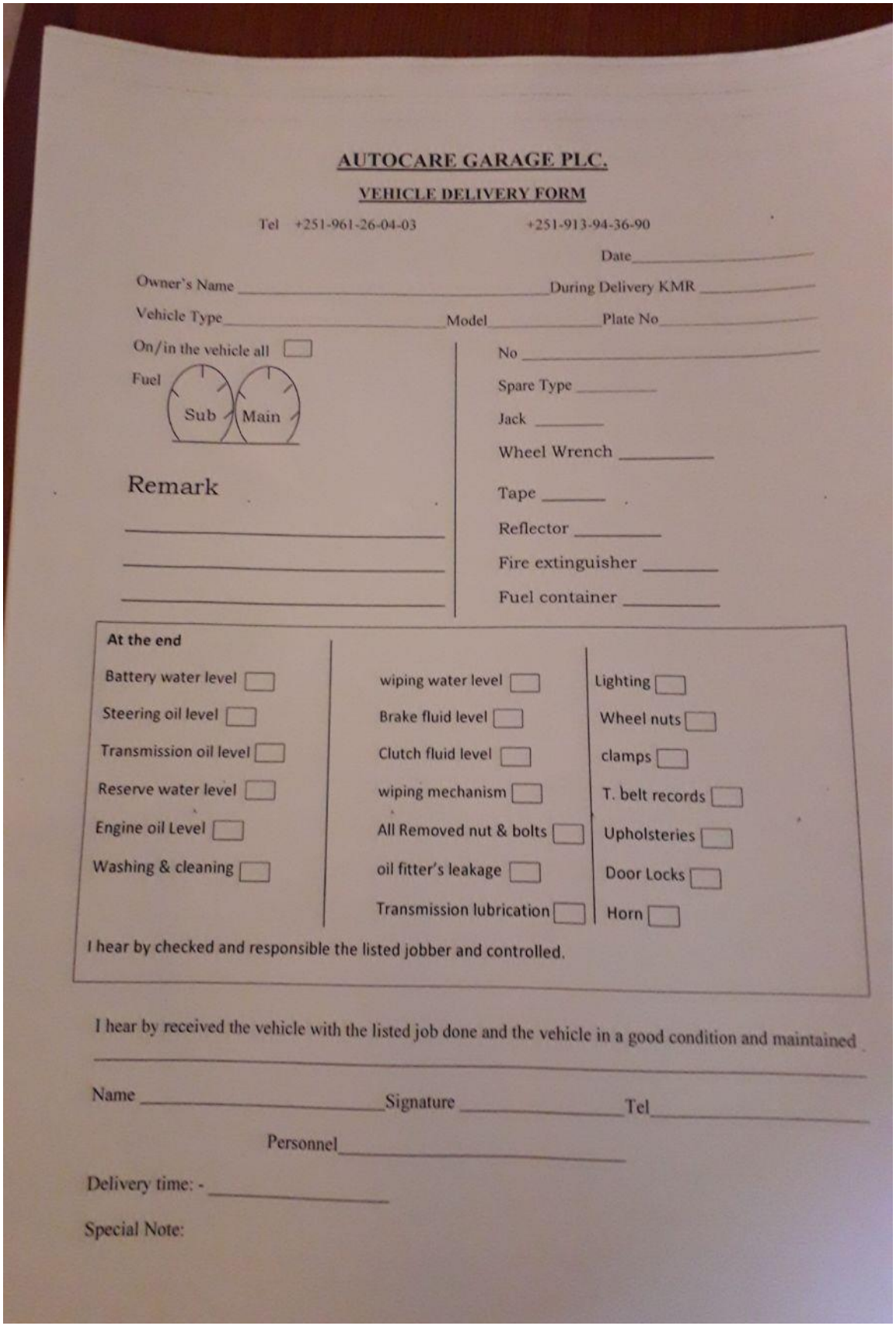
OTHERS_____

Autocare Garage is responsible only for the items which are physically registered in the above list.

Hand over Name_____ Signature _____ Tell no_____ Date_____

Take over Name_____ Signature_____ Tell no_____ Date_____

Fig.2. 1 Vehicle Receiving Form



AUTOCARE GARAGE PLC.

VEHICLE DELIVERY FORM

Tel +251-961-26-04-03 +251-913-94-36-90

Date_____

Owner's Name _____ During Delivery KMR _____

Vehicle Type_____ Model_____ Plate No_____

On/in the vehicle all ☐

Fuel Sub Main

Remark

_____

_____

_____

No _____

Spare Type _____

Jack _____

Wheel Wrench _____

Tape _____

Reflector _____

Fire extinguisher _____

Fuel container _____

At the end

Battery water level ☐	wiping water level ☐	Lighting ☐
Steering oil level ☐	Brake fluid level ☐	Wheel nuts ☐
Transmission oil level ☐	Clutch fluid level ☐	clamps ☐
Reserve water level ☐	wiping mechanism ☐	T. belt records ☐
Engine oil Level ☐	All Removed nut & bolts ☐	Upholsteries ☐
Washing & cleaning ☐	oil fitter's leakage ☐	Door Locks ☐
	Transmission lubrication ☐	Horn ☐

I hear by checked and responsible the listed jobber and controlled.

I hear by received the vehicle with the listed job done and the vehicle in a good condition and maintained

_____

Name _____ Signature _____ Tel_____

Personnel_____

Delivery time: - _____

Special Note:

Fig.2. 2 Vehicle Delivery Form

2.6. Weakness of the existing system

We will describe the problem of the existing system using the PIECE framework. This framework is used to identify the problems within an existing information system. The manual information management is disposed to various problems. These problems can be seen from the following perspectives like performance, information, economic, control, and efficiency, given by the existing system.

- **Performance:** The performance of any system is required to show to meet the needs of users of that system. The current system's performance is weak, time required from initiation to completion of a particular task is relatively high.
- **Information:** - In the manual data management system, there is data redundancy, incomplete and inaccurate data records, data is not organized well and as well as Information is provided to the public, manually through poster which at sometimes is not timely and updated information.
- **Economical**: - AutoCare Garage Plc. is not a digitized organization. The various activity performs all of its tasks manually which requires human power to do that task, plenty of papers to print out different format of paper and time. Also in manual operation, most of the activities are prone to wastage of resources.
- **Controlling:** - Since all the information associated with the manual system are stored manually the security and control of this information are poor and In addition there has been loss of filled formats from different files and vulnerable to cheating paper. Another control related problem is the files are stored physically so if they are damaged by different causes there will not be a backup.
- **Efficiency:** - The existing garage management system has well defined organizational structure but report / communication / information exchange among the employees is totally manual, paper based or using telephone activities for the whole activity. Due to the mentioned cases, within the different steps there is a delay, that is reflects less efficiency. So that it not unable to perform function with minimum:
    - Papers.
    - Human power.
    - Time etc.

2.7. Business Rules of the Existing System

A business rule defines or constrains one aspect of your business that is intended to assert business structure or influence the behavior of your business [6]. The following business rule are identified with existing system.

BR1: AutoCare Garage Plc. is responsible only for the items which are physically registered.

BR2: Technician head must approve the inventory asked by technician.

BR3: Store keeper must have to checks signatures when issues the items.

BR4: The customer have to receive the vehicle by signing, that his order job has done and check the vehicle in a good condition and maintained.

Chapter Three

Proposed System

3.1. Introduction

This chapter is the main and soul of our proposed system. It deals finds out the functional and non-functional requirements of the proposed system.

3.2. Functional Requirements

Functional requirements are those that refer to the functionality and the intended behaviors of the system, i.e. what services it will provide to the user. This behavior and functionality may be expressed as services, tasks or functions that the system is required to perform. The following are the functional requirements of the proposed system.

The system should allow:

- **Administrator:**
  - To add/view/update/delete/ accounts.
  - To manage backup.
- **Customer:**
  - To book an appointment for services.
  - To view estimate history.
  - To make payment.
- **Service advisor:**
  - To manage appointment (confirm, reject).
  - To register customer for service.
  - To assign vehicles.
- **Manager:**
  - To receive vehicle/register customer if for heavy maintenance.
  - To assign vehicle.
  - To view report.
  - To create repair estimate.
  - To add delivered car.
- **Technician head:**
  - To view assigned vehicle.

- To request items.
- To record status of repair and maintenance.
- To generate report.

- **Store keeper:**
  - To record items.
  - To view requisition.
  - To store issues.
  - And etc.

3.3. Non-functional Requirements

The non-functional requirements describe user visible aspects of the system that are not designated to the functional behavior of the system. It is the overall characters that make the product property attractive, usable, fast or reliable to the users. They cover on a number of issues from user interface to security issues and the constraints of the system to improve quality service.

3.3.1. User Interface and Human Factors

Our system intends to provide user friendly GUI using consistent theme throughout all pages to avoid user confusion, with intuitively understandable buttons, wizards, etc. Moreover the GUI will consider different monitor resolutions of user workstation.

Different users ranging from layman to professional visit this system; as a result our system will consider these gaps and provide ease of use by avoiding complexity, unnecessary functionality and by giving descriptive name to different components.

3.3.2. Hardware Consideration

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

- Desktop /laptop computers.
- Uninterrupted power supply.
- Internet connection
- Database server computers and web server computers. , etc.

3.3.3. Security Issues

The system is protected with authentication and authorization security mechanism. Any user who wants to login to the system must have a user name and password to make the system secured from unauthorized access. To handle this, users are required to give their unique user name together with their password at login. It is always important to ensure that only valid data is entered and only valid operations are performed on the system.

A user with guest role is not authenticated to enter into the system but can only view the home page, book appointment & make contact while surfing the web since they are allowed to access the web page which contains the homepage. In this way we can grant that every page on the website is accessed by the authorized person only.

Authorization: Which resources the user can access and which operations the user is allowed to perform. The users that login the systems and performs their activities (add, view, delete, and manage) depend on their authorization of resource access. Example Technician head not accessing user account creation and Store keeper do not register customer.

3.3.4. Performance Consideration

We will use different method in order for the proposed system to have a good response time. Such as using PHP functions, upgrading our PHP to the latest PHP version, code optimization (shorten codes), remove unnecessary bugs and develop easily loadable interface components.

3.3.5. Error Handling and Validation

In our system we use message to handle errors, if user make a mistake when the doing some operations display information about the errors by using message box. Example: when the expected input is string the user, will not be able to provide numerical values and if user insert incorrect password the message says “Please Insert the correct Password”. It used to inform the user made a mistake then the user retype the password.

3.3.6. Quality Issues

The system must be available, robust and maintainable. As well as the system need to be reliable, i.e. users can trust or rely on due to the accuracy of the information or data it provides.

3.3.7. Backup and Recovery

We have to come up with a recovery plan in order to deal with risks happen to our server. This could include hardware failure, server software failure, or a disaster such as a fire or a flood at our hosting facility. So we will be taking the following measures in order to make sure that we are saved if something happens to our host server.

Get the web/database backup on a regular cron job: is an approach used to set up a cron job on web server to automatically pull backups on a schedule (e.g., daily, weekly, monthly) as best fits site's needs.

Store all backup files in multiple places: - We are going to store the backups in multiple places like on the cloud, in the local server, and external hard drives so as if one fails to make sure we have another copy too.

3.3.8. Physical Environment

This system will be deployed in the organization where there is a network infrastructure with internet connection. The data center should be air conditioned and free of any interferences.

3.3.9. Resource Issues

The system needs resources that have high processor speed and memory for server. Additionally, the proposed system will require network devices.

3.3.10. Documentation

At each stage of this project all the relevant information's are properly documented and at the completion of the project, every activity in the development process will be documented for future reference, easily access and maintenance of the system at the end of the documentation will include user manual and executable file.

Chapter Four

System Analysis

4.1. Introduction

Analysis is the detailed study of the various operation performed to the system and their relationship to outside the system. This chapter focuses on system analysis of the new system using Unified Modeling Language (UML).

4.2. System Model

System modeling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. It is about representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML) [7].

Five types of UML diagrams that are the most useful for system modeling:

- **Activity** diagrams, which show the activities involved in a process or in data processing.
- **Use case** diagrams, which show the interactions between a system and its environment.
- **Sequence** diagrams, which show interactions between actors and the system and between system components.
- **Class** diagrams, which show the object classes in the system and the associations between these classes.
- **State** diagrams, which show how the system reacts to internal and external events.

4.2.1. Use Case Model

Use Case modeling is the simplest and most effective technique for modeling system requirements from a user's viewpoint. Use cases are used to model how a system works. The use case model consists of a number of elements that are use case, actor and relationship between them.

An Actor: is a person or an external system that plays a role in one or more interaction with the system.

Use case: A **use case** is a description of a function of a system, intend to describe (sometimes formally) how a system should work.

Here are possible use cases and actor identified for the system:

Use Case ID	Use Case name	Actor
UC1	Login	All Actor
UC2	Manage account	Administrator
UC3	Manage backup	Administrator
UC4	Make payment	Customer
UC5	Book an appointment	Customer
UC6	View estimate	Customer
UC8	Manage an appointment	Service advisor
UC9	Register customer	Service advisor , manager
UC10	Assign vehicle	Service advisor, manager
UC11	View report	Manager
UC12	Create estimate	Manager
UC13	Add delivered car	Manager
UC14	View assigned vehicle	Technician head
UC15	Request items	Technician head
UC16	Record repair status	Technician head
UC17	Generate report	Technician head
UC18	Records Items	Store Keeper
UC19	View request	Store Keeper
UC20	Store issues	Store Keeper

Table 4. 1 Use Case Identification

4.2.1.1. Use Case Diagram

A use case diagram illustrates a set of use cases for a system, the actors of these use cases, the relations between the actors and these use cases, and the relations among the use cases. It also describes a function provided by the system as a set of events that yield a visible result for the actors.

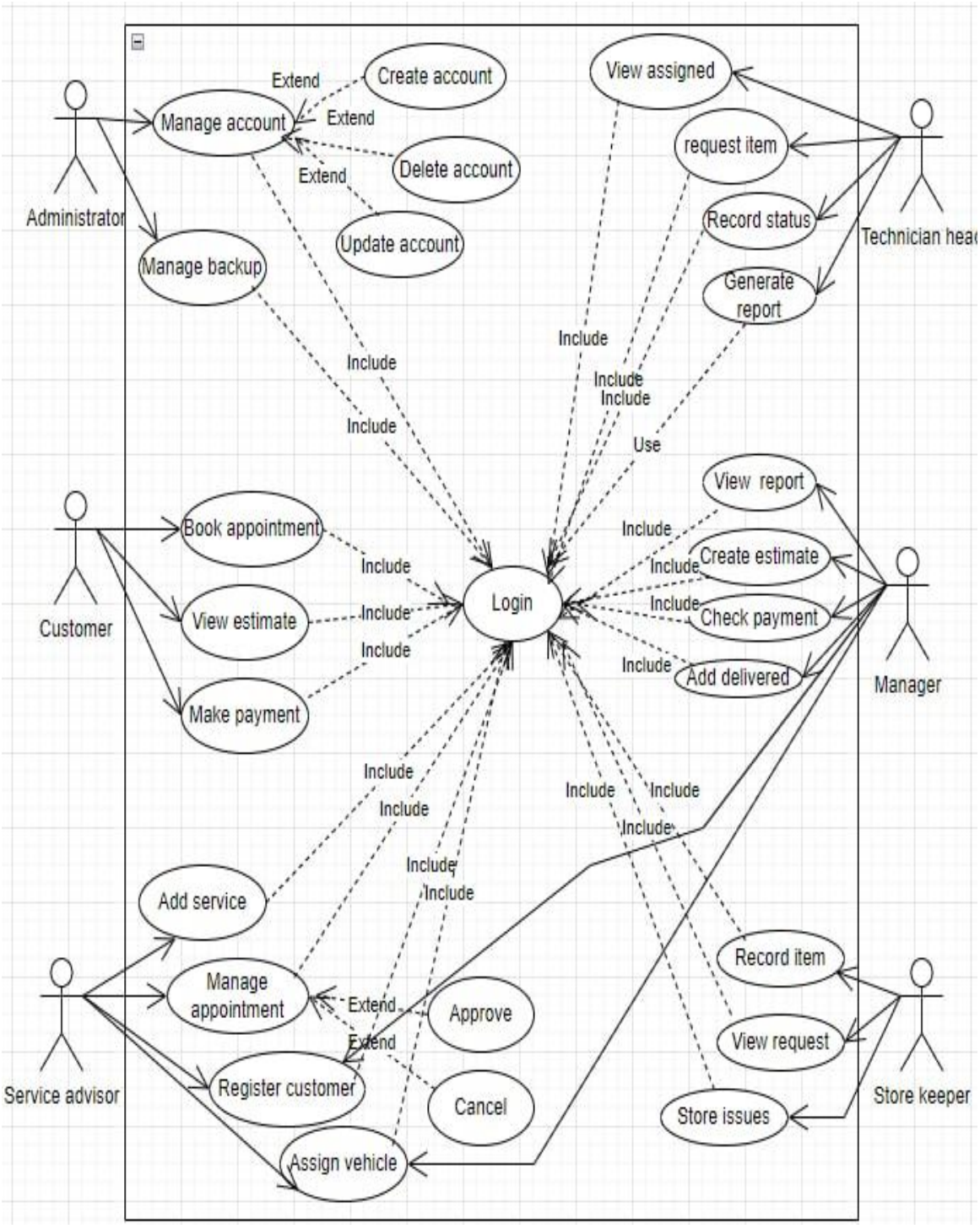


Fig.4. 1 Use Case Diagram

4.2.1.2. Use Case Description

Use case description (also sometimes called narrative of the use case), which is the text-based, detailed, step-by-step interactions and dialogue between the actor and the system.

1. Use case description -login

Use case name	Login
Use case ID	UC -1
Actors	Administrator, Service advisor , Technician head, manager & Store keeper
Description	This use case describes how the system allows the actor to login into the system to access the relevant functions according to the user role.
Pre-condition	The actor must have authorized user-name and password.
Basic course of action	1. The Actor initiates login system. 2. Then the system displays the login interface. 3. The Actor inserts the username and password on the space provided and then click the login button. 4. The system validates the correctness of the user name and password. 5. The system displays the users of their own page. 6. Use case ends.
Post-condition	Actors are able to login.
Alternative action	A1:Step4, if user enter invalid user name and password, The system displays an error message “Incorrect user-name or Password” and ask to Re-enter user-name and Password again. and use case resume to go to back step 3
Exit	- Logout

Table 4. 2 Log in – Use Case Description

2. Use case description -manage user

Name	Manage user account
User case ID	UC-2
Actor	Administrator
Description	This use case describes how the system allows the Administrator to create, update, delete, search and control user.
Precondition	The administrator must be logged in.
Basic course of action	1. After the administrator successfully login into the system click on the manage page. 2. The system displays the administrator manage page 3. The administrator enter the required input to create, update, delete and search and control user role. 4. The system checks validate of the input data. 5. The system displays the messages whether successful or not. 6. The system create ,update, delete search to/from database 7. Use case ends.
Post condition	The administrator is able to manage (view, update, delete, create) users account.
Alternative action	A1: Step4, The system may display an error message in case of any database transaction error or invalid input and the administrator may repeat step 3 or may exit from the system.
Exit condition	- User account is registered , updated, deleted or searched - The administrator log out

Table 4. 3 Manage User Account - Use Case Description

3. Use case description -book an appointment

Use case name	Book an appointment
Use case ID	UC -5
Actors	Customer
Description	This use case describes how the system allows customer to book an appointment for services.
Pre-condition	The customer must access the GMS website & fill a valid form.
Basic course of action	1. The customer access the GMS website. 2. Then the system displays the GMS website interface. 3. The customer click on book an appointment button. 4. The system display book an appointment form. 5. The customer fill the form and click submit menu. 6. The system validates the entered information. 7. The use case ends.
Post-condition	The appointment is successfully submitted.
Alternative action	A1: step6 If the filled form is invalid the system ask to enter a valid form. The use case resume at Basic course of action step 4
Exit condition	- Appointment is booked - The customer leaves website

Table 4. 4 Book an Appointment - Use Case Description

4. Use case description - View estimate

User Case Name	View estimate
User Case ID	UC-6
Actor	Customer
Description	This use case describes how the system allows customer to view estimate history.
Precondition	The Customer must know their repair id.
Basic course of action	1. In order to view, the Customer must have to login to the system. 2. Then system displays the main customer page. 3. The Customer clicks on the view estimate history button. 4. Then the system will display the repair estimate interface. 5. The customer enter repair id. 6. The system check the entered repair Id, and display the result. 7. Use case ends.
Post condition	The estimate history is viewed.
Alternative action	A1: Step6, If repair car id is not found, The system display no car found based on your repair ID message, The use case resume at Basic course of action step 4
Exit condition	- view repair estimate - log out

Table 4. 5 View Estimate- Use Case Description

5. Use case description - Register customer

User Case Name	Register customer
User Case ID	UC-10
Actor	Service advisor , manager
Description	This use case describes how the system allows service advisor or manager to register customers.
Precondition	The Actor must be logged in.
Basic course of action	1. After Actor login into the system click on register customer button 2. The system display a registration form. 3. They enters the details of the customer information 4. The system validates, and store the register customer information in to the database. 5. The systems display, notify successful registration. 6. Use case ends.
Post condition	New customer are recorded and registered to the database.
Alternative action	A1: step4, if they may enter invalid input, The use case resume at Basic course of action step 2
Exit condition	- Customer registered - Log out

Table 4. 6 Register Customer - Use Case Description

6. Use case description - Generate report.

Name	Generate report
Use Case ID	UC-18
Actor	Technician head
Description	This use case describes how the system allows technician head to generate report.
Precondition	Whenever the report time is arrived or the report requested or needed by manager.
Basic course of action	1. After successfully login ,The technician head clicks “generate report” button 2. Then, the system displays the report form. 3. The user fills the necessary input and click the “generate” button. 4. The system validates input data. 5. The System generates the report. 6. Use case ends
Post condition	Report is generated
Alternative action	A1: Step4, if the technician head enter invalid input, he go back to step2,3
Exit condition	- Report is generated - Log out

Table 4. 7 Generate Report- Use Case Description

4.2.1.3. Use Case Scenario

Scenarios are concrete examples of the future system in use. Developers use these scenarios to communicate with the user and deepen their understanding of the application domain. Scenario is a concrete, focused, informal description of a single feature of the system from the viewpoint of a single actor.

Scenario name: login

Actor: all user

All the actors' wants to login into the system. They should initiates login page, Then system displays the login form. After that the user fill the form with the appropriate username and password. Then the system checks the input data and if the user fills the correct information the system allows the user to enter into the system and perform their activity. If the user enters invalid input the system generates the error message else user page is display next perform his authenticated operation and logout.

I. **Scenario name: Book an appointment.**

Actor: Customer

If the Mr. Daniel wants to book an appointment for service of his car, he must access GMS website. After that Mr. Daniel click on book appointment button. Then the system displays the book appointment form, then the Mr. Daniel fill the form and click submit. Then the system display success message else ask to fill the form correctly.

II. **Scenario name: Customer registration.**

Actor: Service advisor and manager

If the Service advisor wants to register the customer/vehicle, he/she browse the registration form from the dashboard. After visit the page he/she click the add customer. During these the system displays the form. After that the actor fill the form and click the save button. During these the system checks the user information. If they are correctly fill the form registration is added to the database. If they are not correctly fill the form system generates error message else the registration page is displayed and the user perform their activity they are allowed by the system. Finally, the user leaves the page.

4.3. Object Model

The object model is a system or interface which is basically used to visualize elements in terms of objects in a software application. It is modeled using object-oriented techniques and before any programming or development is done, the object model is used to create a system model or an architecture[8].

4.3.1. Analysis Level Class Diagram

Class diagram is representation of a collection of objects with common structure, behaviors and relationships [6].

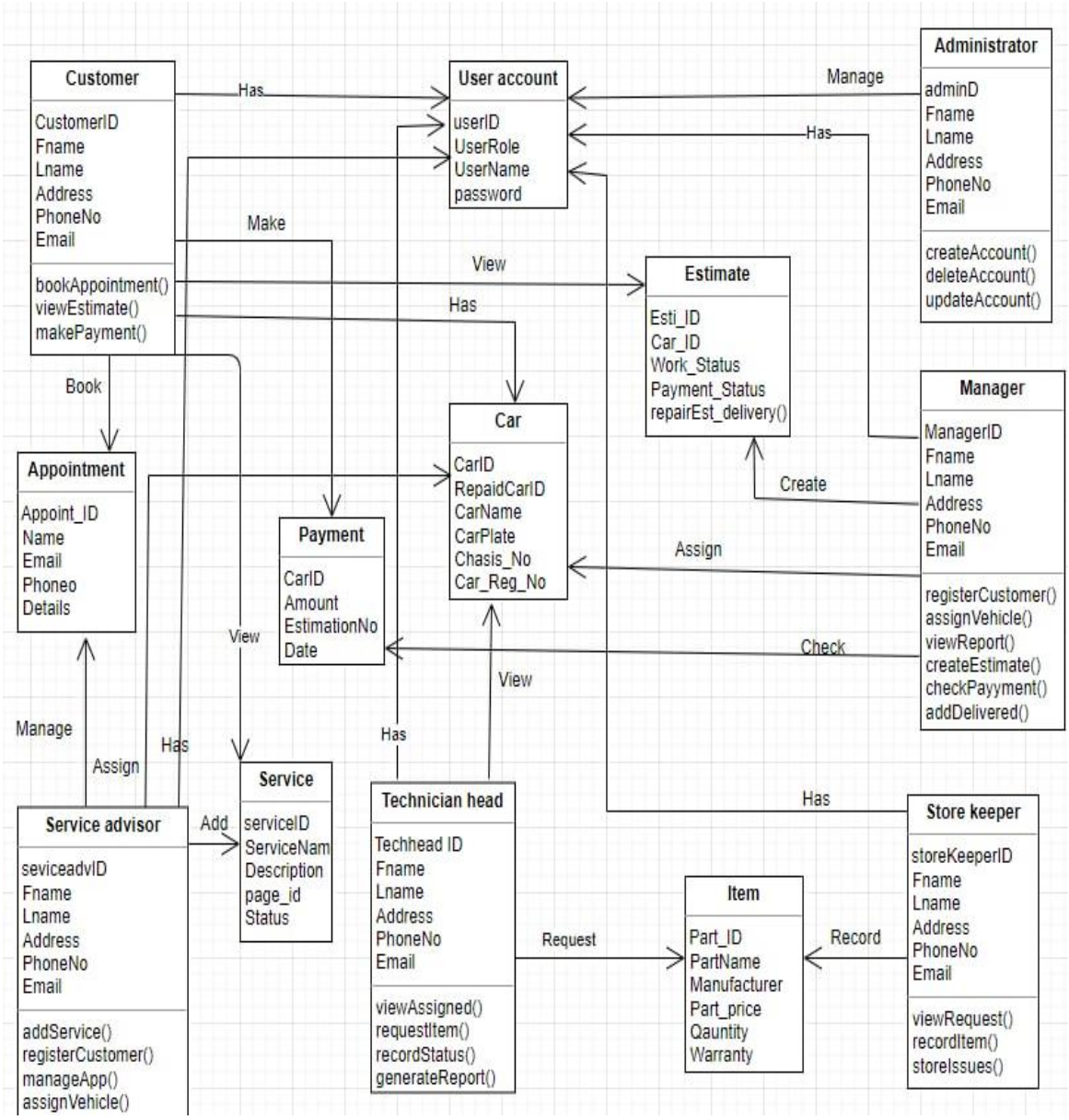


Fig.4. 2 Analysis Level Class Diagram

4.3.2. Data Dictionary

1. Data Dictionary for manager

Attributes	Caption	Example	Data Type	Attributes size	Constraint
managerID	ID	Mgr/15	varchar	10	Primary Key
firstName	First Name	Kebede	varchar	15	Not Null
middleName	Middle Name	Abebe	varchar	15	Not Null
lastName	Last Name	Biniam	varchar	15	Not Null
gender	Gender	Male	varchar	6	Not Null
birthDate	Birth Date	25/04/1992	datetime	10	Not null
email	Email	kebe@gmail.com	varchar	30	Unique
phoneNumber	Phone Number	+251910181312	varchar	20	Unique

Table 4. 8 Data Dictionary for manager

2. Data dictionary for vehicle

Attributes	Caption	Example	Data Type	Attributes size	Constraint
chassisNo	Chassis No	rav 4 – 0D502276	varchar	30	Primary key
vehicleType	Vehicle Type	Automobile	varchar	15	Not Null
model	Model	TOYATA2022	varchar	15	Not Null
plateNo	Plate No	AA:23922	varchar	9	Unique
motorNo	Motor No	127	Int	6	Not Null

Table 4. 9 Data Dictionary for Vehicle

3. Data dictionary for item

Attributes	Caption	Example	Data Type	Attributes size	Constraint
partNo	Part no	W11	varchar	22	Primary Key
partName	Item Name	Glass	varchar	20	Not Null
category	Category	Wheel	varchar	20	Not Null
price	Price	19900.34	Float	6	Not Null
quantity	Quantity	200	Int	5	Not Null
Supplier	Supplier	Great wall	Varchar	20	Not null

Table 4. 10 Data Dictionary for Item

4. Data dictionary for report

Attributes	Caption	Example	Data Type	Attributes size	Constraint
reportID	Report ID	Rep/35	varchar	20	Primary Key
reportDate	Report Date	21/10/2022	datetime	12	Not null
reportTitle	Report Title	Written text	varchar	15	Not null
userRole	User Role	Head	varchar	20	Not null
reportContent	Report Content	Written text	varchar	500	Not null

Table 4. 11 Data Dictionary for Report

5. Data Dictionary for User Account

Attributes	Caption	Example	Data Type	Attributes size	Constraint
userID	User ID	MANAGER/15	varchar	15	Foreign Key
userName	User Name	Yonas.	varchar	15	Not null
password	Password	Password123.	varchar	10	Not null
userRole	User Role	Manager	varchar	14	Not null

Table 4. 12 Data Dictionary for User Account

6. Data Dictionary for record status

Attributes	Caption	Example	Data Type	Attributes size	Constraint
statusID	Account ID	repairstatus/35	varchar	15	Primary Key
serviceType	Service Type	Inspection	varchar	15	Not null
detailProblem	Detail problem	motor failure	varchar	15	Not null
sparePartUsed	Spare part used	motor	varchar	10	Not null
labour	labour	Consume 3 hr and three person	varchar	20	Not null

Table 4. 13 Data Dictionary for Record Status

4.4. Dynamic Model

Dynamic Modeling represents the temporal aspects of a system, capturing the control elements through which the behavior of objects can be understood over time. The Dynamic Model describes those aspects of a system concerned with time and the sequencing of operations - events that mark changes, sequences of events, and the organizing of events and states [9].

4.4.1. Sequence Diagram

Sequence diagrams show a detailed flow for a specific use case or even just part of a specific use case. It is almost self-explanatory that shows the calls between the different objects in their sequence and can show different calls to different objects.

Sequence Diagram

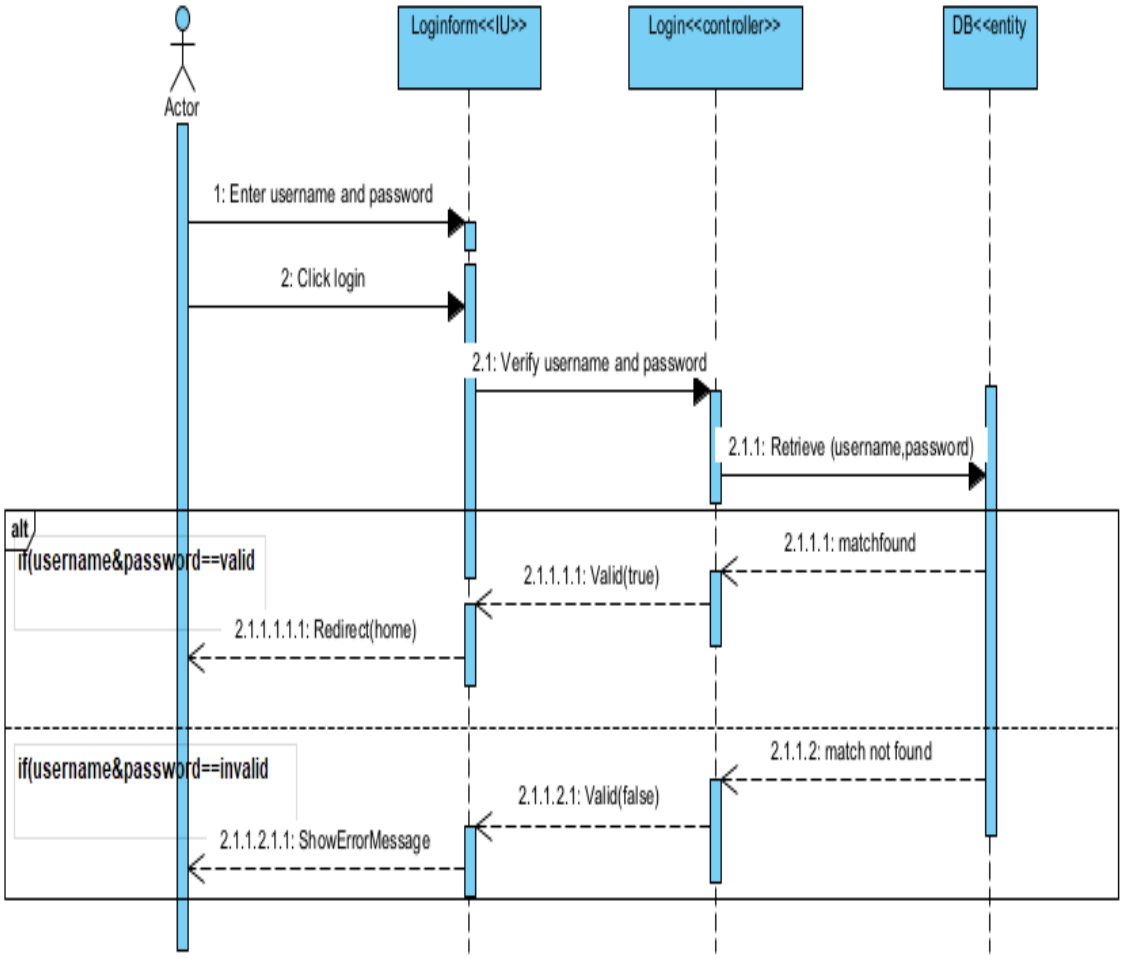


Fig.4. 3 Login - Sequence Diagram

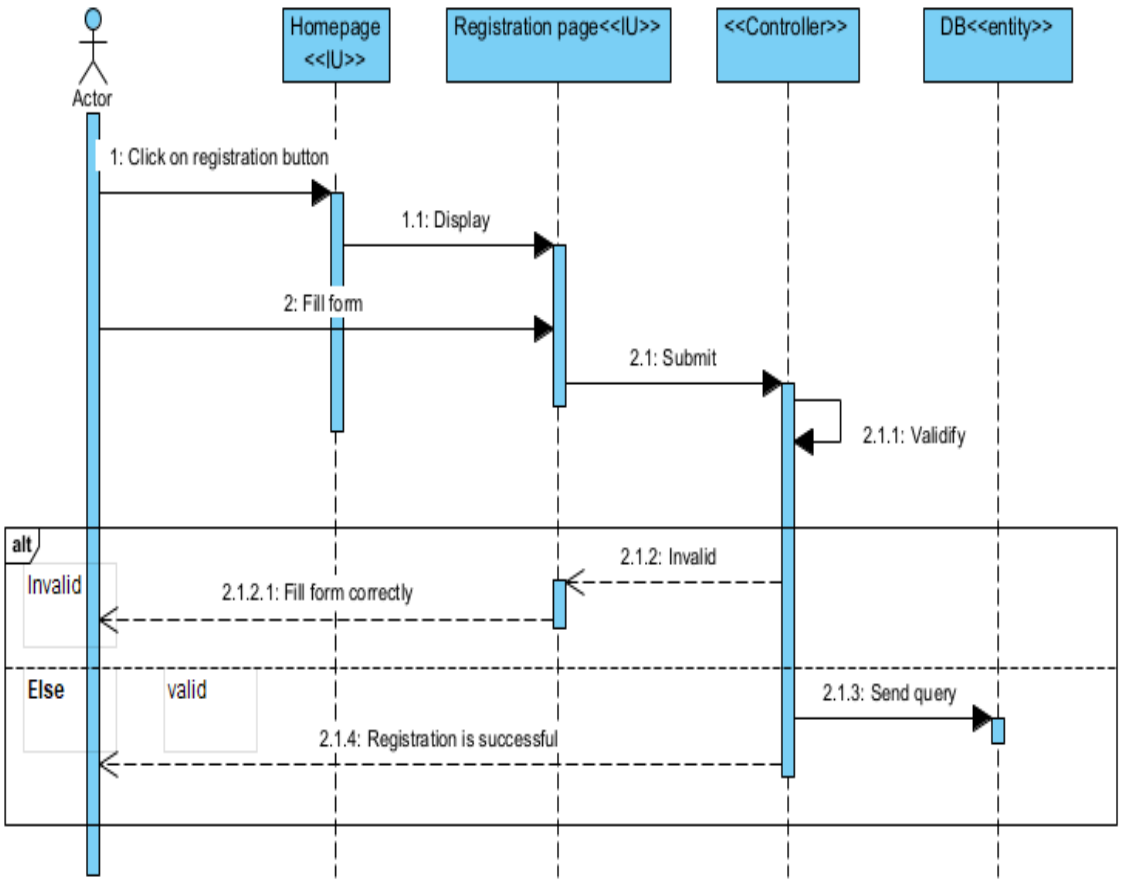


Fig.4. 4 Customer Registration- Sequence Diagram

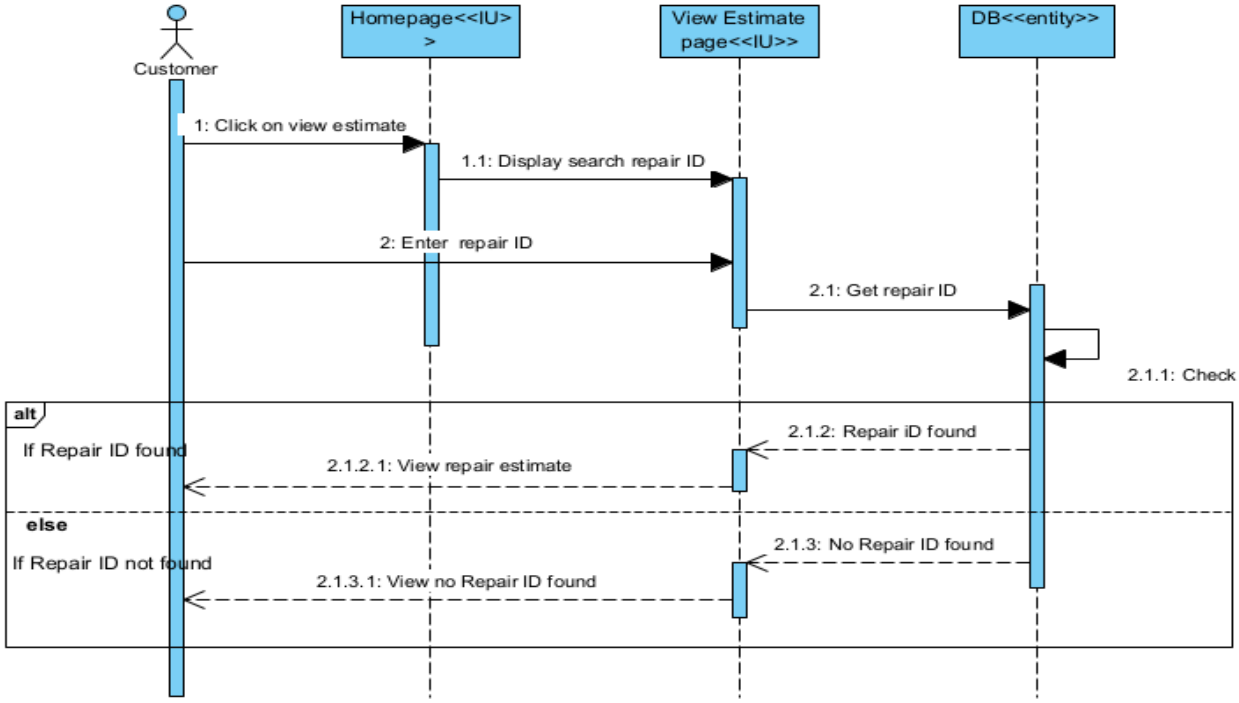


Fig.4. 5 View Estimate- Sequence Diagram

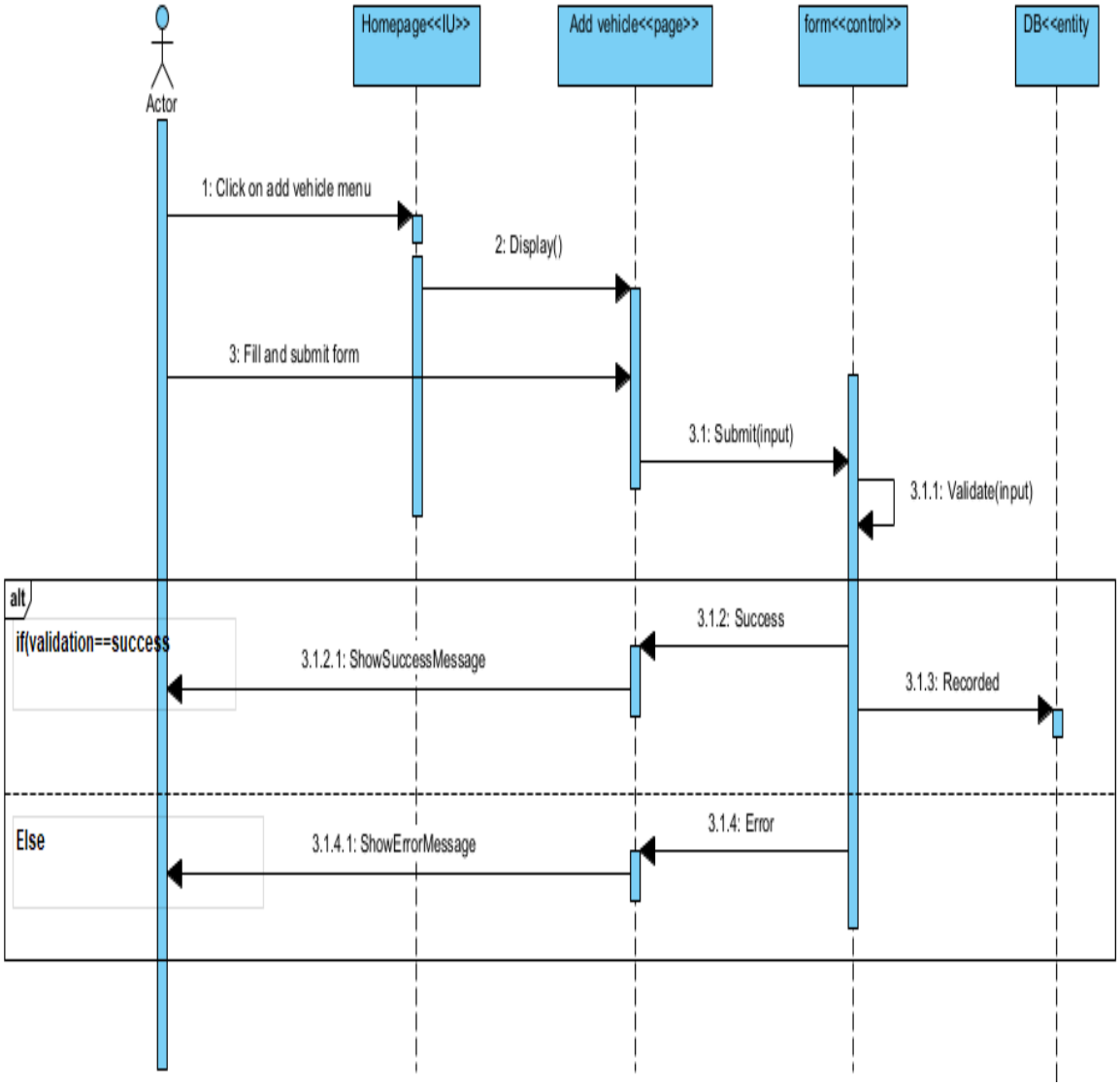


Fig.4. 6 Add Vehicle - Sequence Diagram

4.4.2. Activity Diagram

Activity diagram shows the conditional logic for the sequence of system activities needed to accomplish a business process. It clearly shows parallel and alternative behaviors that can be used to show the logic of a use case.

The activity diagram starts with a solid circle connected to the initial activity. The activity is modeled by drawing a rectangle with rounded edges, enclosing the activity's name. Activities can be connected to other activities through transition lines, or to decision points that connect to different activities guarded by conditions of the decision point. Activities that terminate the modeled process are connected to a termination point.

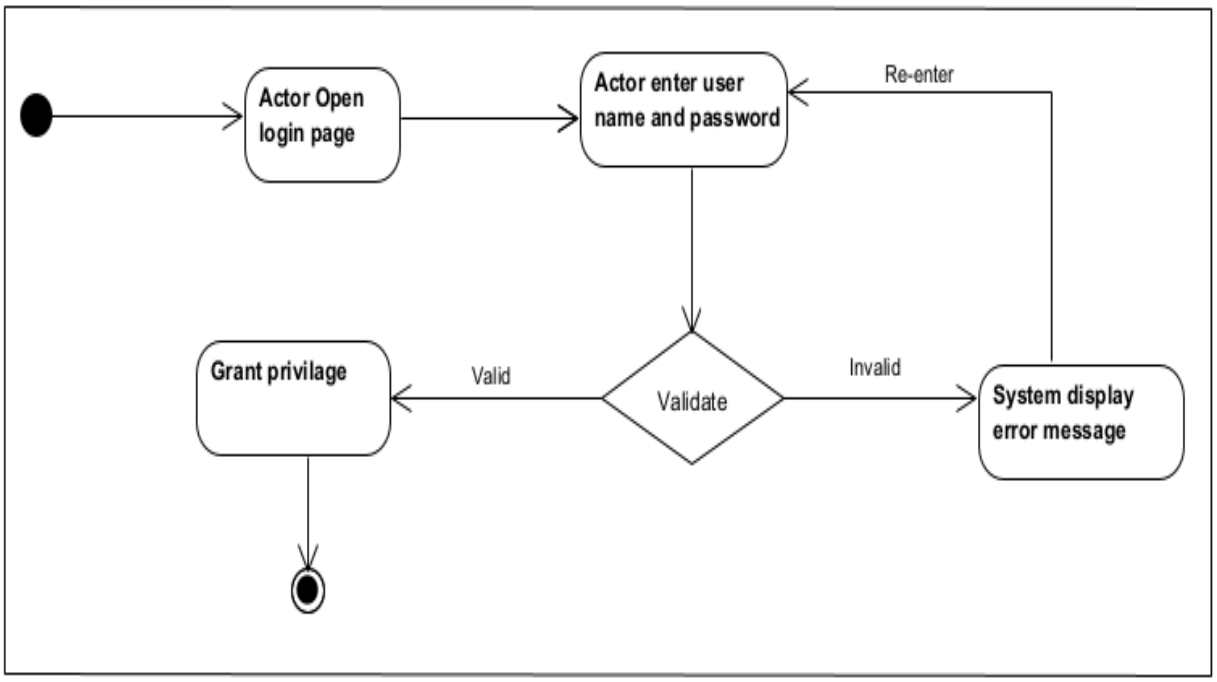


Fig.4. 7 Login - Activity Diagram

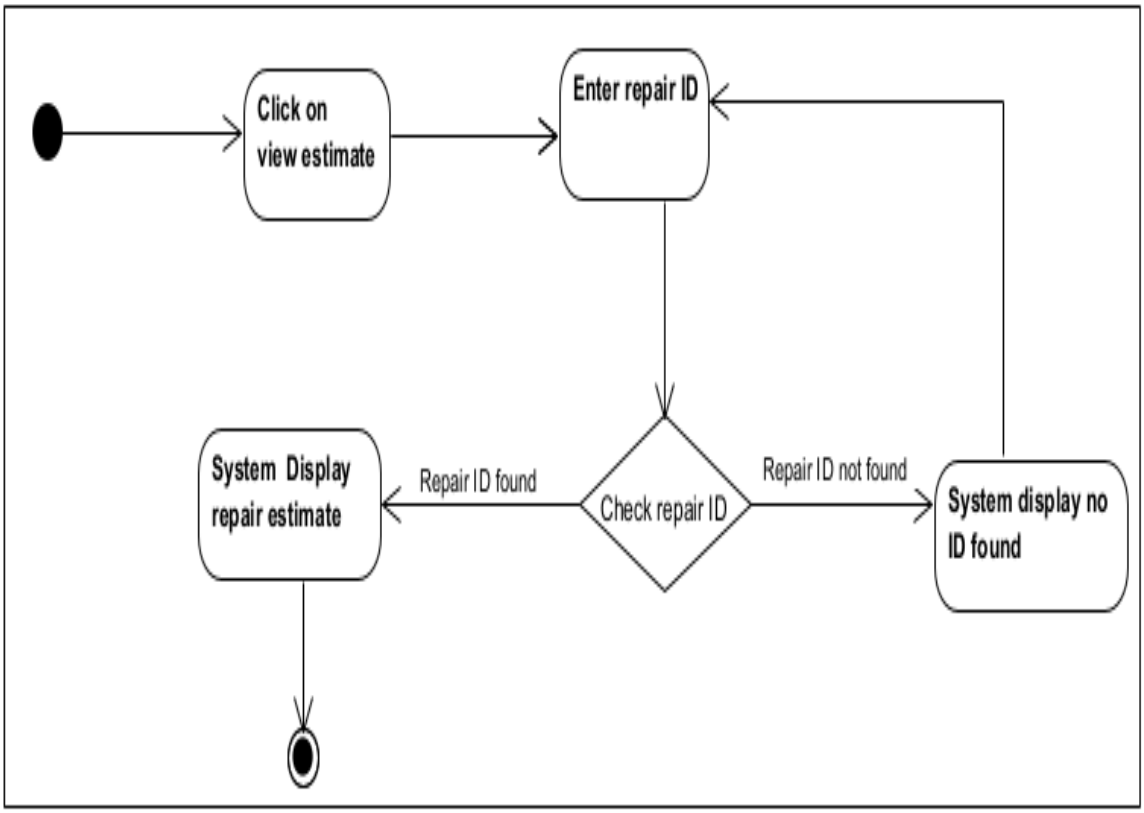


Fig.4. 8 View Estimate - Activity Diagram

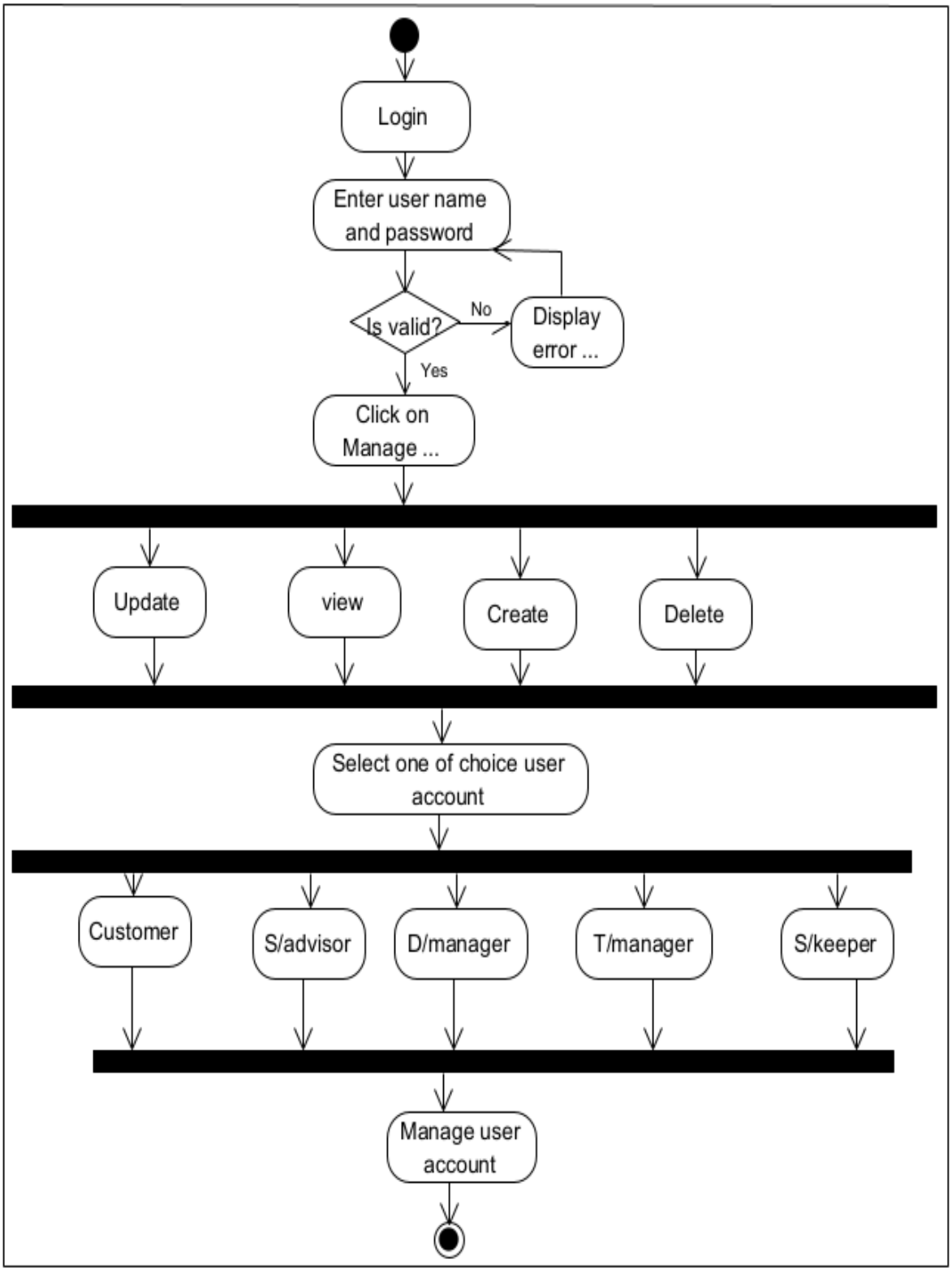


Fig.4. 8 Manage Account - Activity Diagram

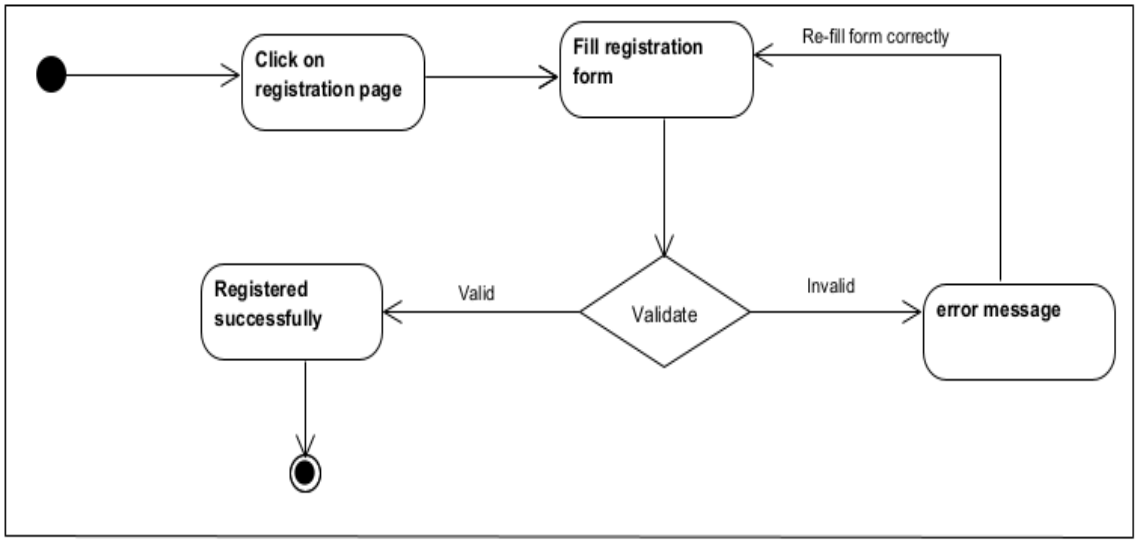


Fig.4. 10 Register Customer - Activity Diagram

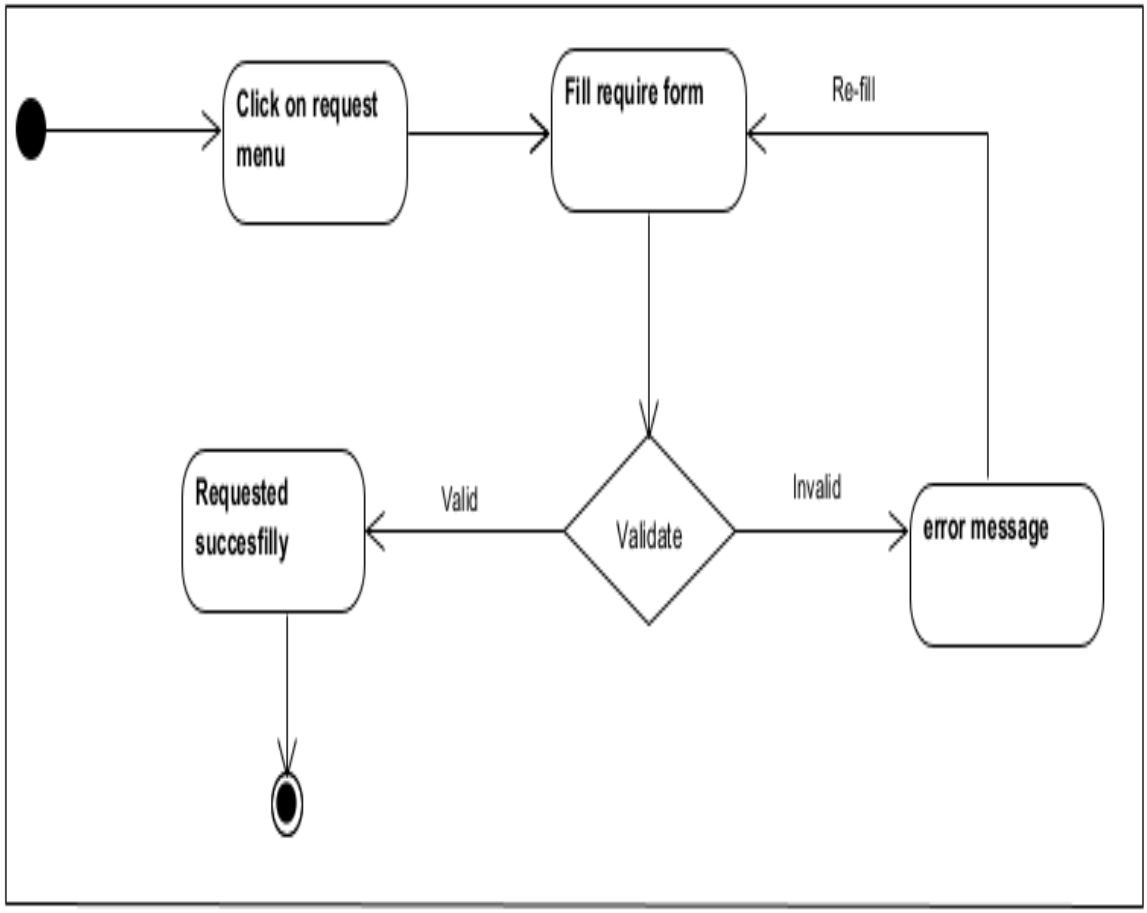


Fig.4. 11 Request Item - Activity Diagram

4.4.3. State Chart Diagram

State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered.

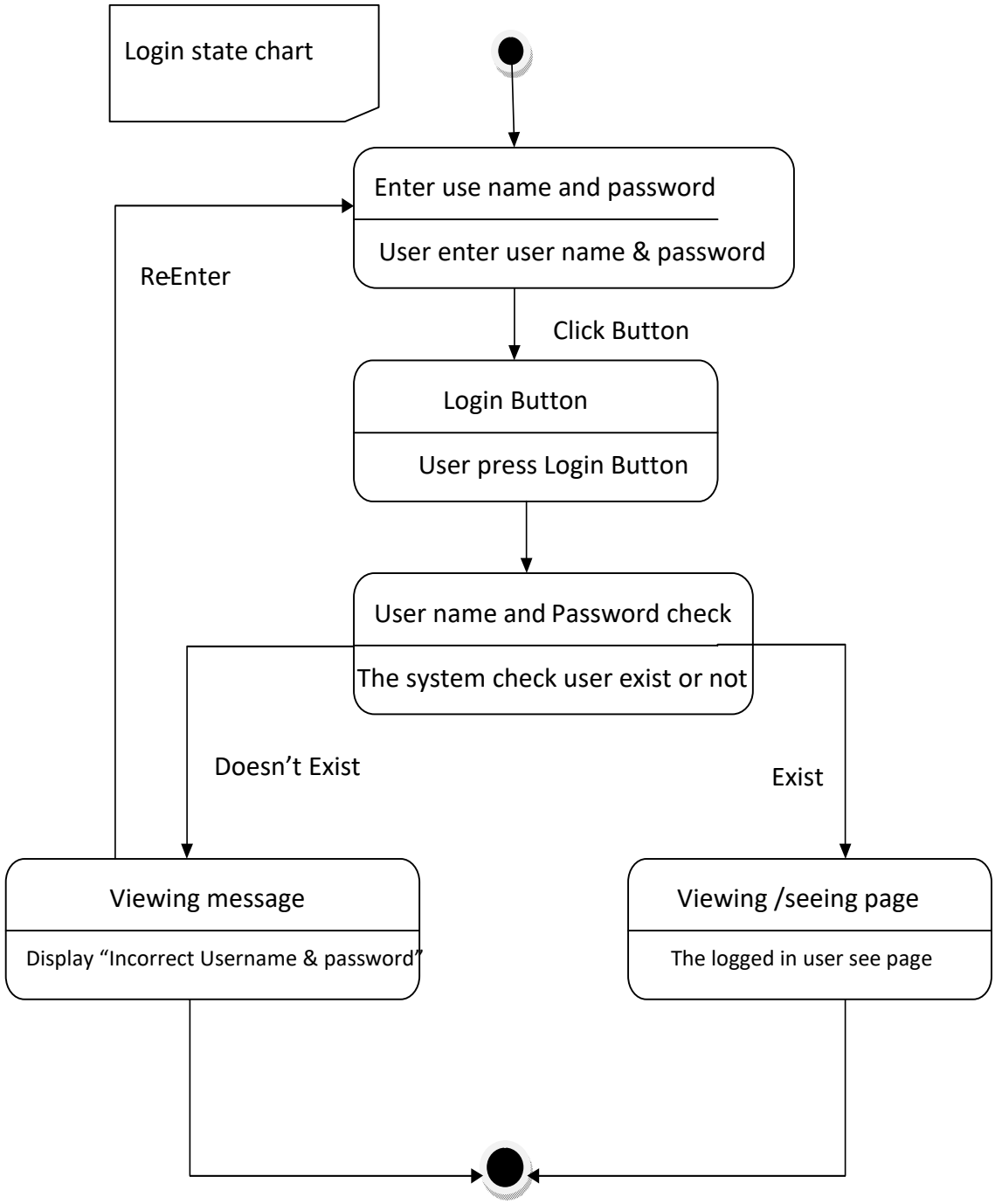


Fig.4. 12 Login – State Chart Diagram

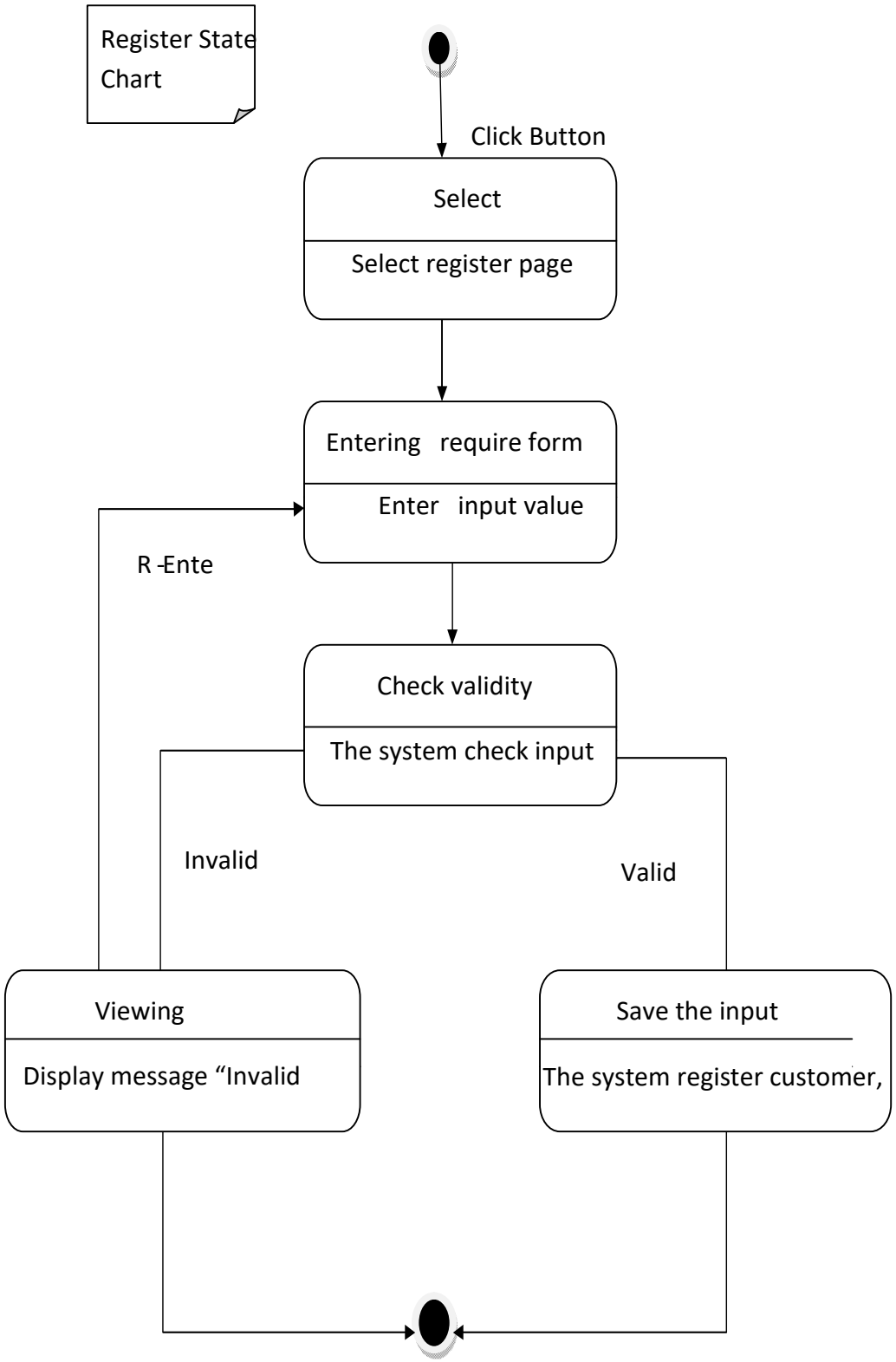


Fig.4. 13 Registration Customer - State Chart Diagram

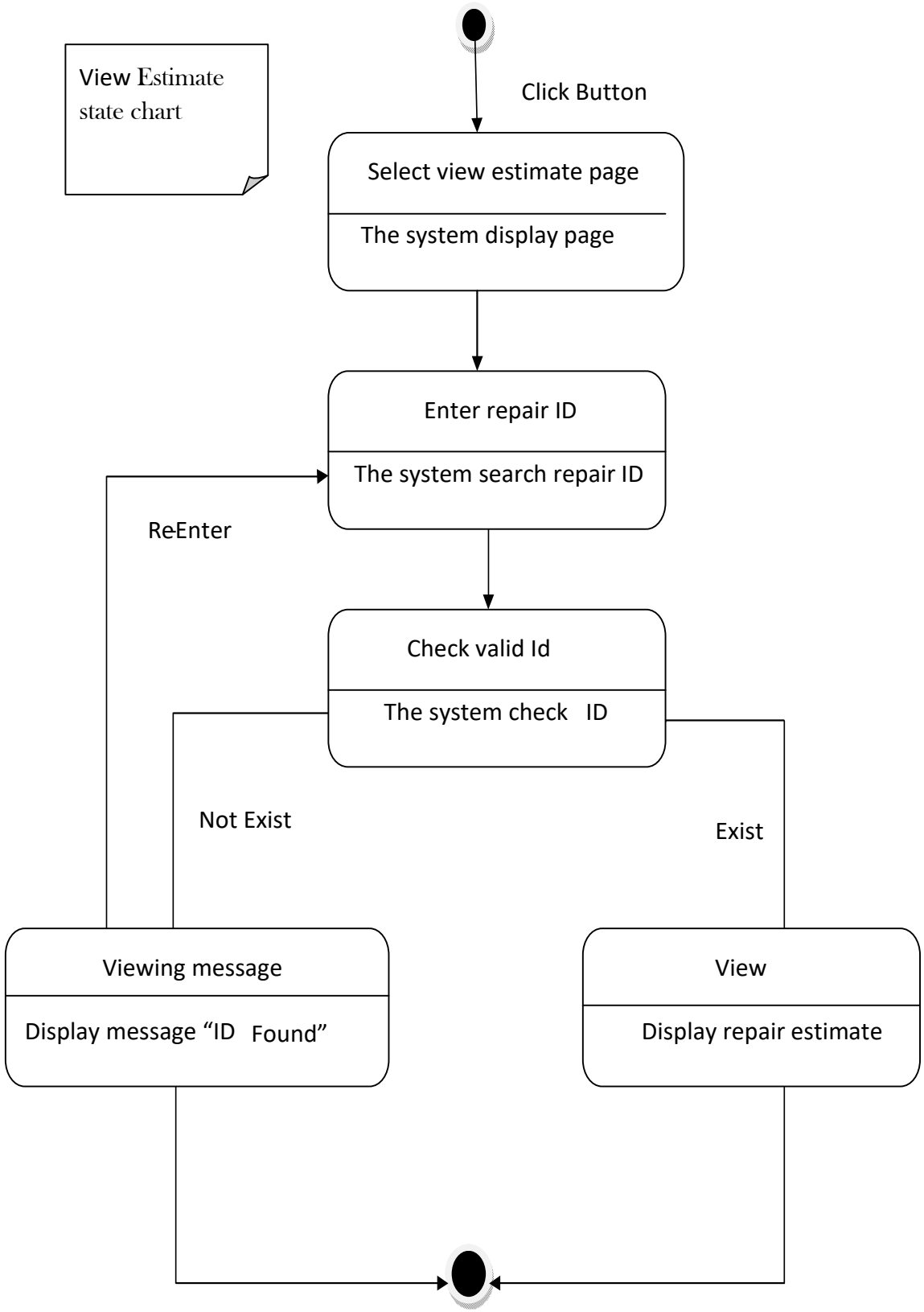


Fig.4. 14 View Estimate - State Chart Diagram

Chapter Five

System Design

5.1. Introduction

This chapter describes the design of the system and overview clearly design goal. The second section which is the most important one describes the system under development in terms of system decomposition, hardware/software mapping, persistent data management, access control and packages. Finally, the last section highlight a pseudo algorithm and user interface design.

5.2. Goal of System Design

The design goals represent the desired qualities of the system and provide a consistent set of criteria that must be considered when making design decisions.[10].

Some of the goals are listed below

Portability: The system should work in different platforms because of there could be platform shifting in the future.

Robustness: The system will handles any incorrect input from the user. There is an error handling mechanisms and also recognize the input and prompt the user to make the necessary corrections.

Availability: The system should be available for twenty-four hours per week (24/7) so that users can have access to it anytime.

Security: The system should be secured from the unauthorized.

Performance: The system will have good performance as much as possible.

End-user: The system should provide user-friendly and self-explanatory graphical user interface that eases the interaction of the user with the system.

Fault Tolerance: Whenever the user commits errors intentionally or unintentionally the system should continue performing its task by displaying user friendly error messages.

Good Documentation: The user documentation should considers the different kinds of users.

5.3. Current System Architecture

Currently the organization uses a manual system, this implies that the organization has no current software architecture.

5.4. Proposed System Architecture

The architecture plays a main role in modularizing the different components of a system. The our system will be built based on the assumption that users use their phones or computer in the environment with a wireless network or use their laptop or desktop in the environment with LAN networks. So developed system must be on client/server architecture. It have a minimum of three layers (tier); the presentation tier, logical tier, and data access tier.

- **The Presentation Tier:** is the topmost level of the application. It is the one the users directly interact with. It provides GUI to allow the client gaining access to the system.
- **Logical Tier/Middle Tier:** It accepts inputs from the client and performs detailed processing. It is a bridge between the presentation tier and data access tier.
- **Data Access Tier:** Provides data persistence mechanism and storage to the data.

The proposed logical architecture is depicted below.

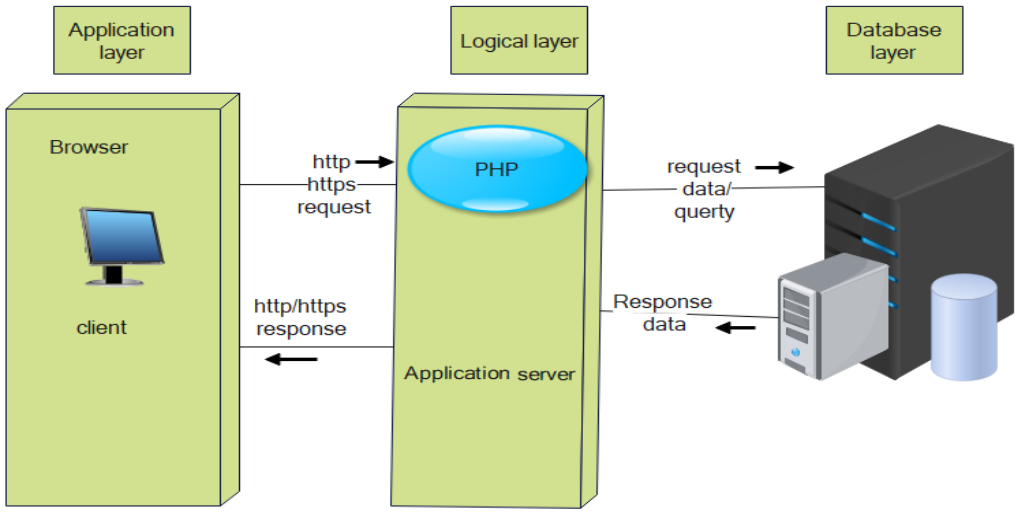


Fig.5. 1 Architectural Diagram

5.4.1. Subsystem Decomposition and Description

In order to make the implementation easy and manageable, the newly developed system for the garage management is decomposed in to the following smaller subsystems.

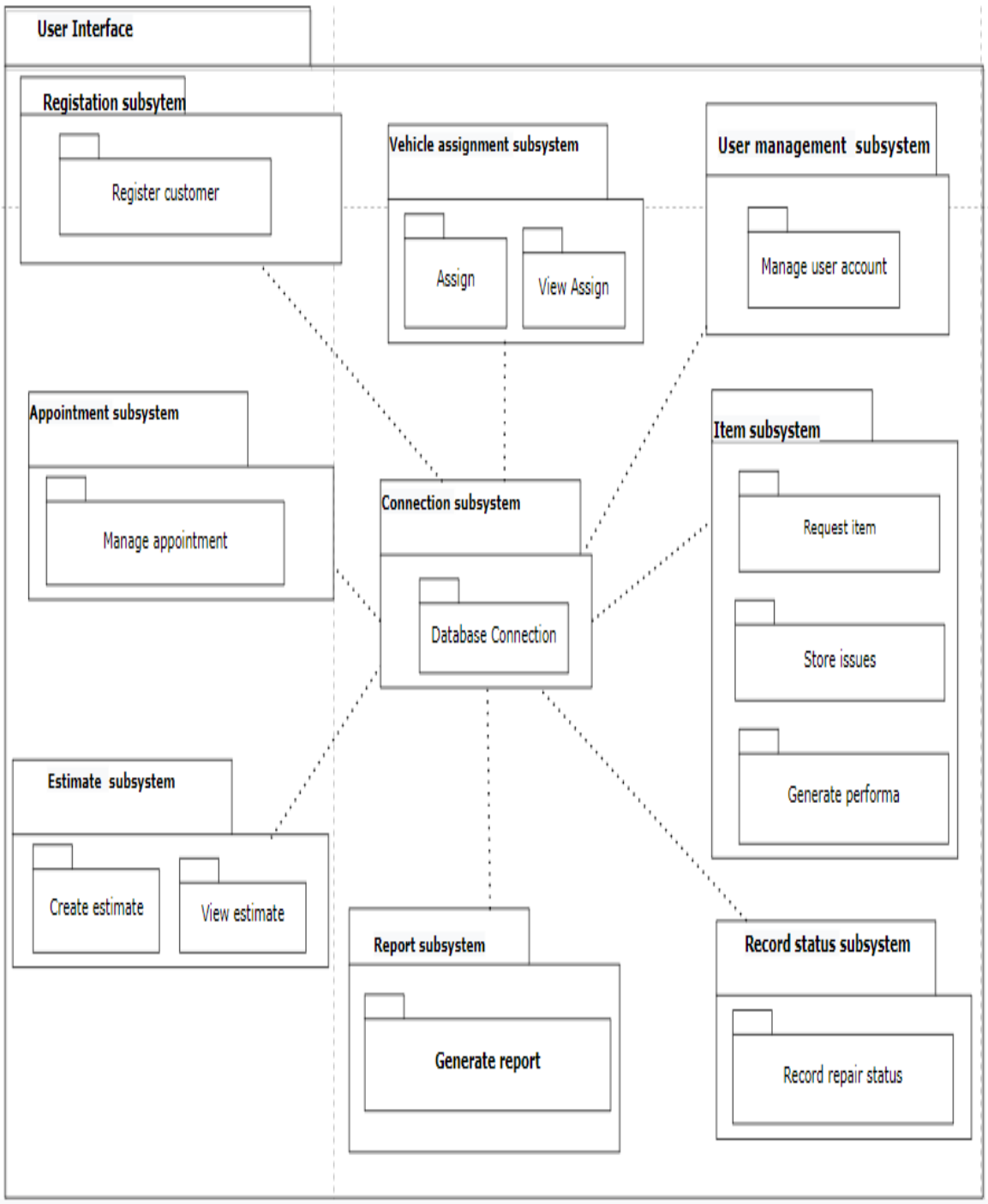


Fig.5. 2 Subsystem Decomposition Diagram

5.4.2 Hardware/Software mapping

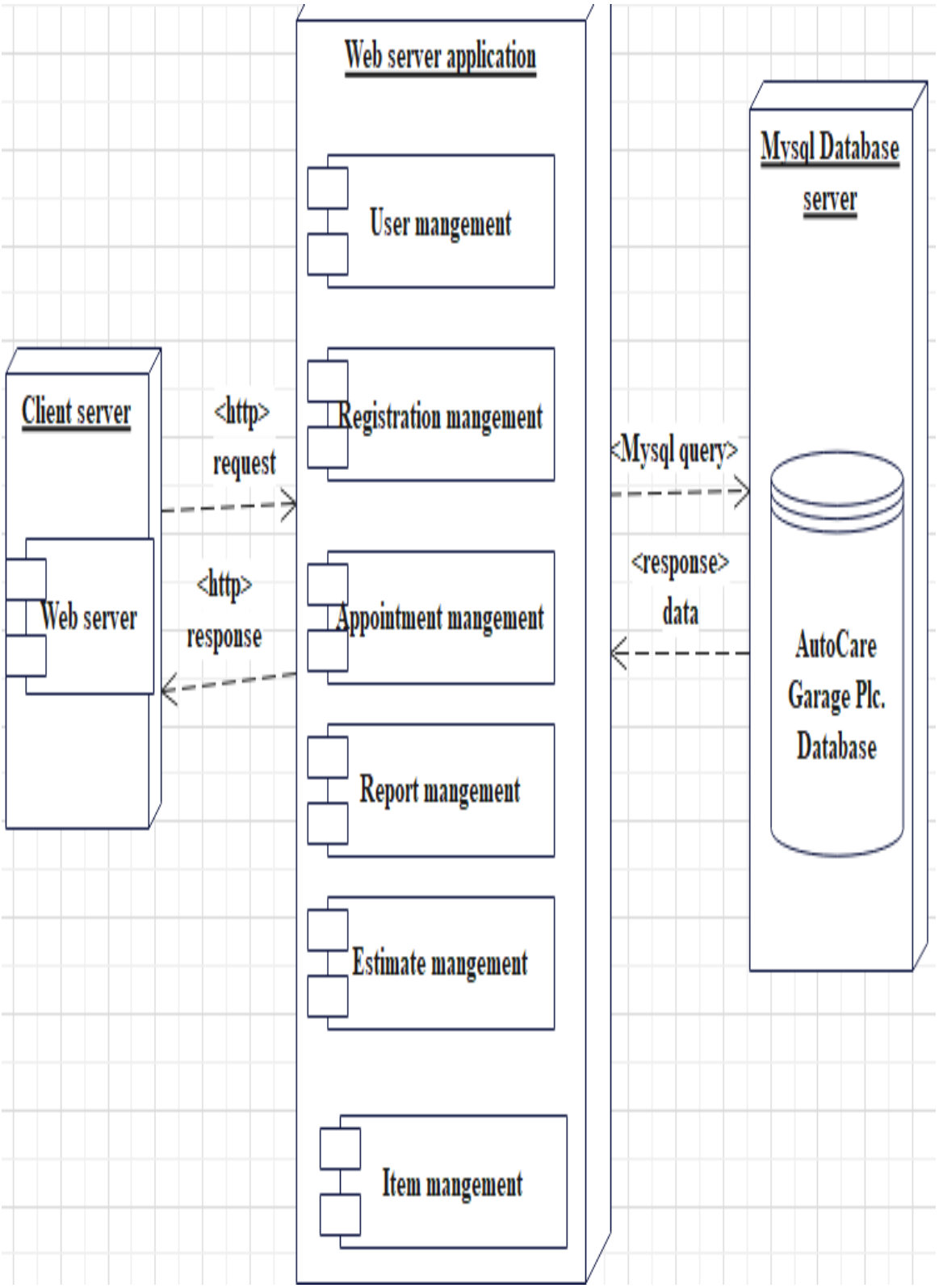


Fig.5. 3 Hardware/Software Mapping Diagram

5.4.3. Detailed Class Diagram

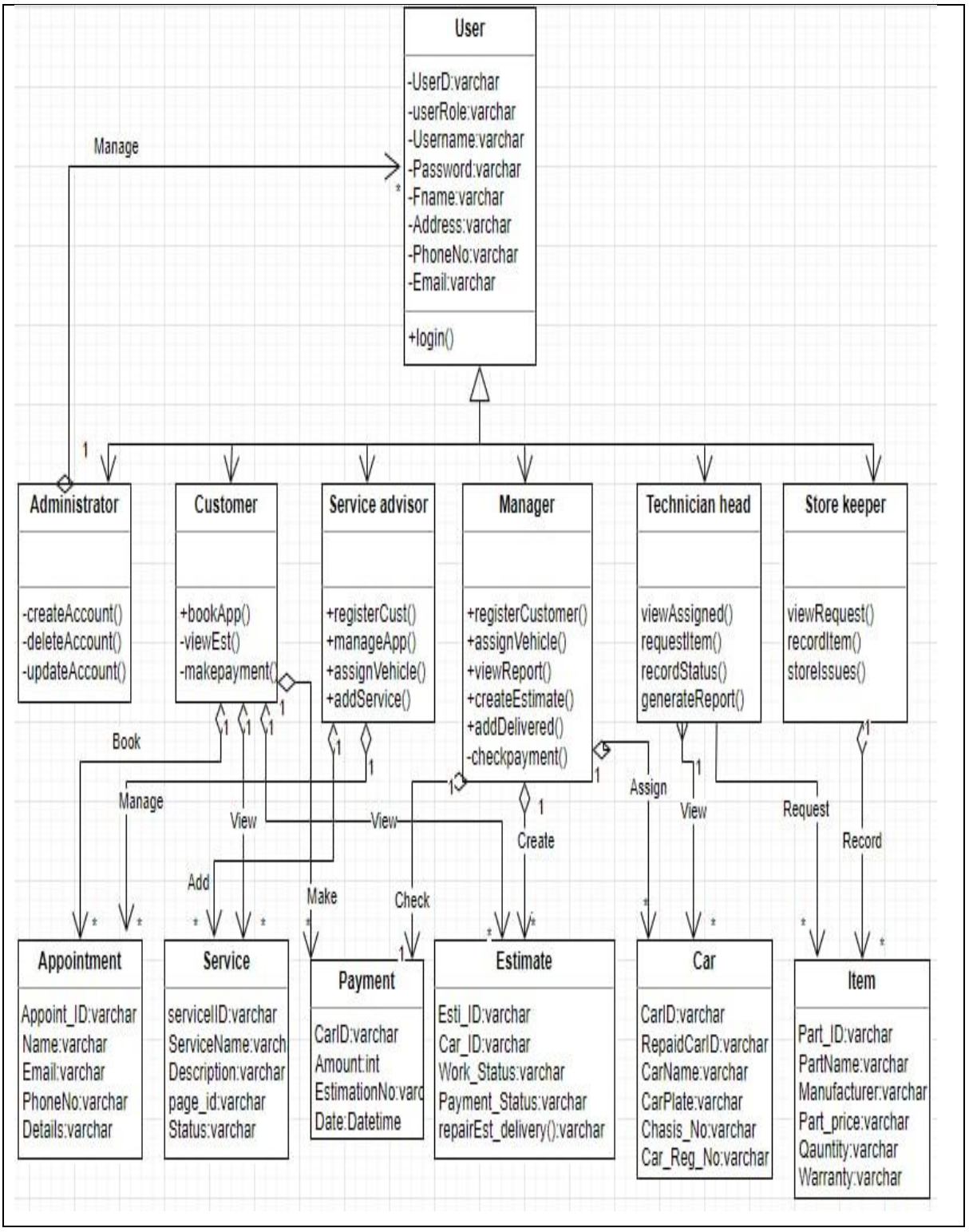


Fig.5. 4 Detailed class Diagram

5.4.4. Persistent Data Management

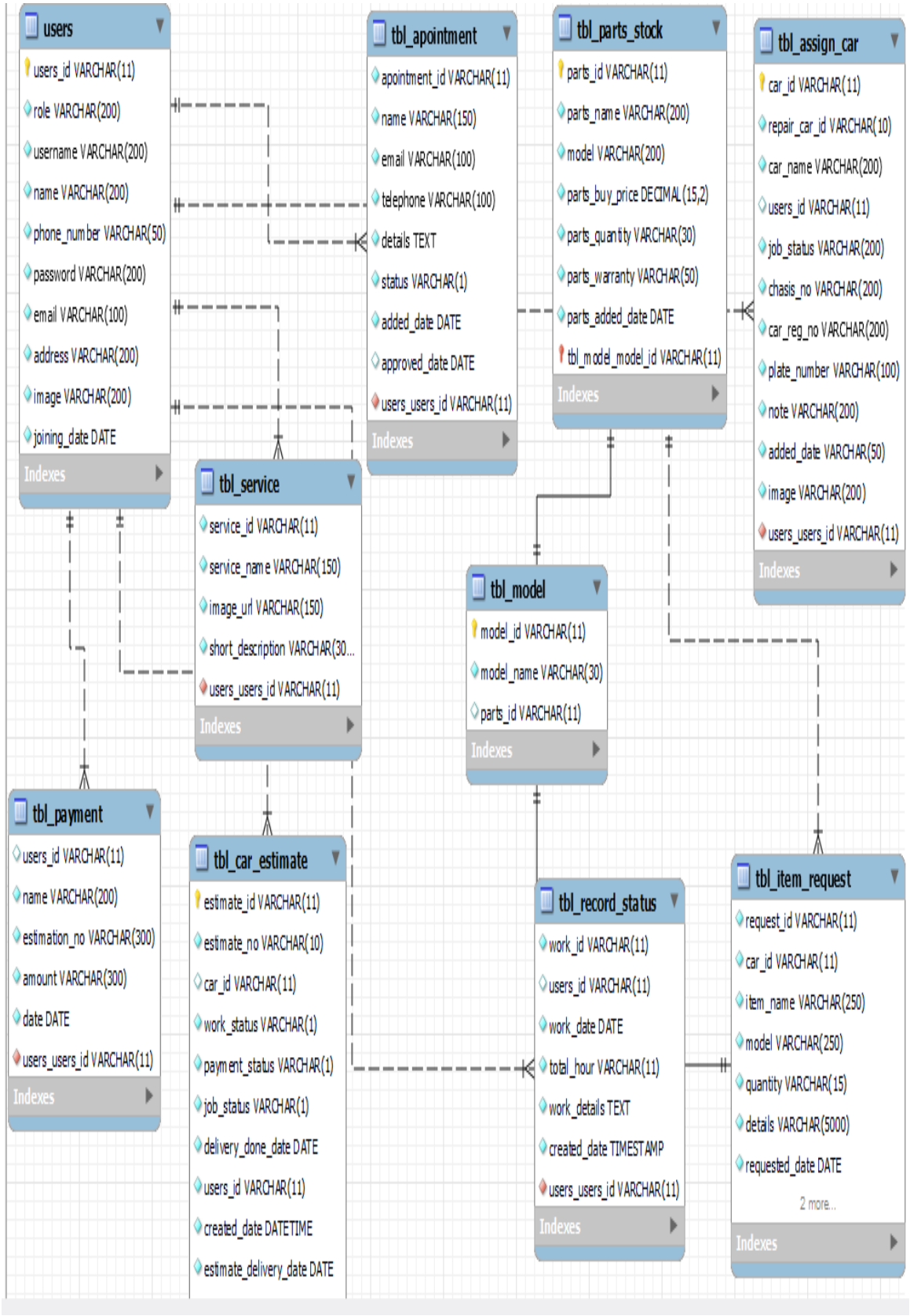


Fig.5. 5 Persistent Data Management Diagram

5.4.5. Access Control and Security

	Admin	Customer	Service advisor	Manager	Technician head	Store keeper
Login	✓	✓	✓	✓	✓	✓
Manage backup	✓					
Manage account	✓					
Book appointment		✓				
View estimate		✓				
Make payment		✓				
Manage appointment			✓			
Register customer			✓	✓		
Assign vehicle			✓	✓		
View report				✓		
Create estimate				✓		
Add delivered car				✓		
View assigned vehicle					✓	
Approve requisitions					✓	
Record repair status					✓	
Generate report					✓	
Records Items						✓
View request						✓
Store issues						✓

Table 5. 1 Access Control and Security

5.6. Packages

Package diagrams are structural diagrams used to show the organization and arrangement of various model elements in the form of packages. A package is a grouping of related UML elements, such as diagrams, documents, classes, or even other packages [11].

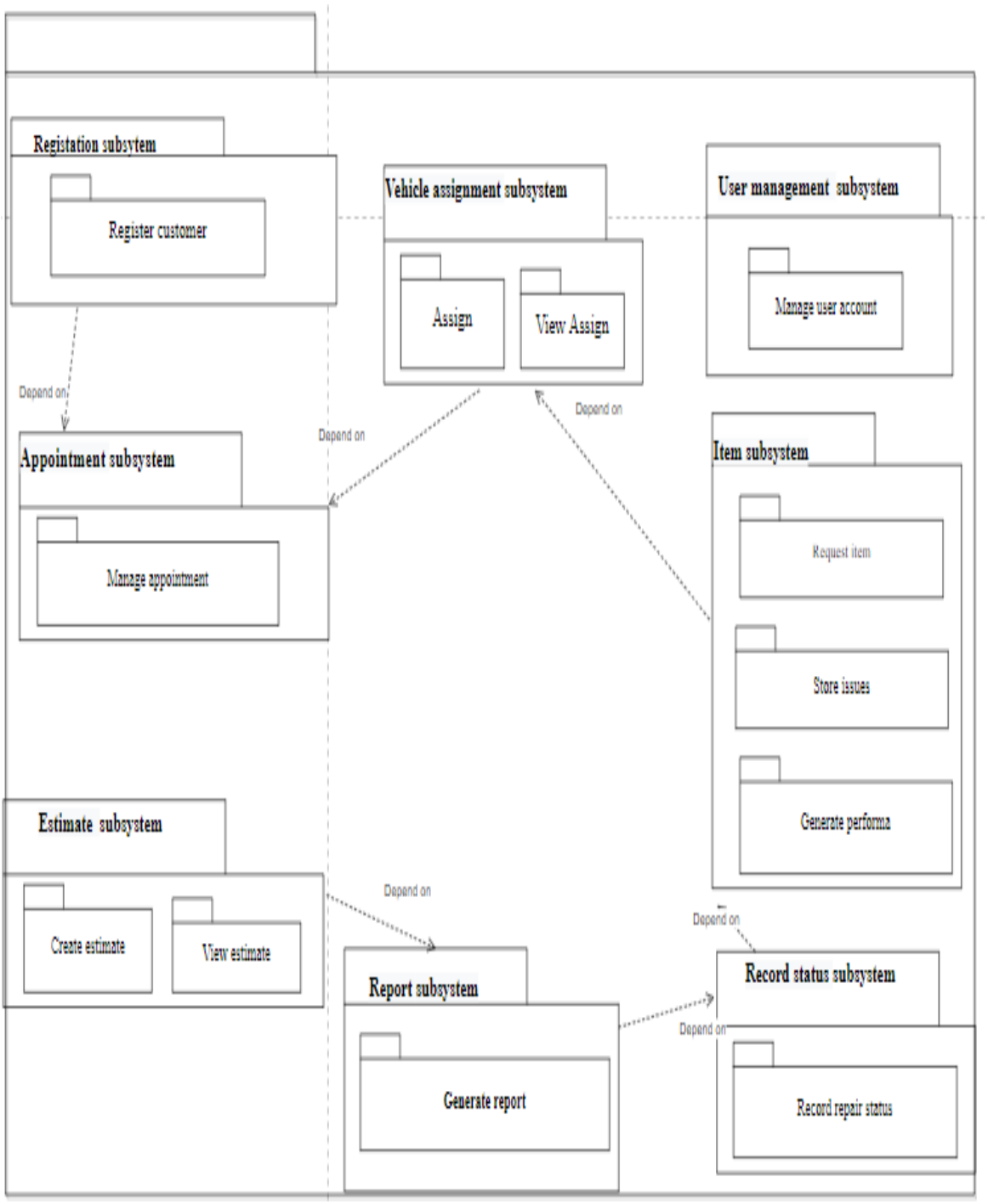


Fig.5. 6 Packages Diagram

5.7. Algorithm Design

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.

Pseudo code for Login

step1: BEGIN

step2: enter username and password on the given form.

Step3: IF username and password valid, THEN send to Database.

step4: IF username and password invalid, THEN go to step 2 and re-enter valid username and password.

Step5: Database check entered input

step5.1: IF valid, THEN logged in successfully

step5.2: IF invalid, THEN not logged in successfully and try again with correct username and password.

Pseudo code for Registration

step1: BEGIN

step2: IF user clicks registration link, THEN form displayed.

Step3: User fill required data.

Step4: If filled data valid and user with this data is not exist, then add user to Database

Step5: Display user registered successfully.

Step6: IF filled data invalid or the user with this information is preexist, THEN go to step3 and re-enter valid input, or

step7: ENDIF

5.8. User Interface Design

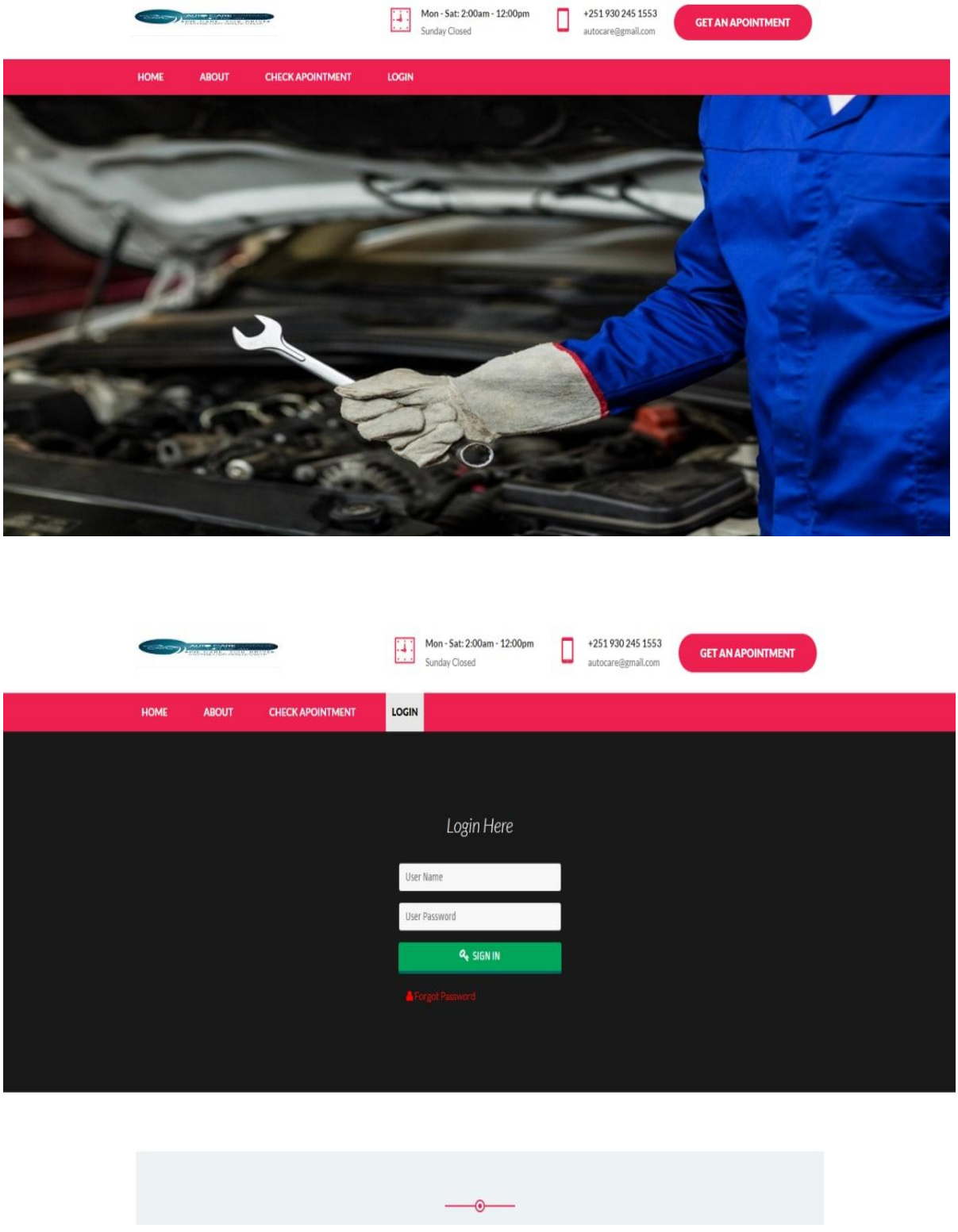


Fig 5.7 User Interface Design

Chapter 6

Implementation and Testing

6.1 Introduction

In this chapter we discuss about implementation of database, implementation of class diagram, configuration of application server and application security as well testing.

6.2 Database Implementation

We use MYSQL database to manage our system which assists to insert, update, delete and view data which is stored in the database.

We use MySQL because it can be deployed and run on any operating system. MySQL is easy, fast and can be used for any type of database whether it is relational or simple, large or small.

6.3 Implementation of the Class Diagram

We have used EdrawMax in order to draw the structure of the class diagram and we have implemented it in the code section.

6.4 Configuration of the Application Server

We use Xampp application server because Xampp is simple, lightweight Apache distribution it is extremely easy to create a local web server for testing and deployment purposes. Everything you needed is to set up a web server – server application (Apache), database (Maria DB), and scripting language (PHP). XAMPP works equally well on Linux, Mac, and Windows. Since it is suitable and have the function, we listed above we use Xampp application server.

6.5 Configuration of Application Security

From the security aspect of our system, done the following tasks.

- Implement all input validations properly, our system used to validate all the inputs by returning error message and suggesting to try again if the invalid input is given it.
- We implement encryption for user password by MD5.
- We implemented session for each user login page.
- We applied authorization: Which resources the user can access and which operations the user is allowed to perform

6.6 Implementation of User Interface

We implement the user interface by considering the users of our system and easy for use.

6.7 Testing

6.7.1 Test Case

The test cases are used to test whether there is differences between the functional requirements, means the use case models and the observed systems behavior.

<table>
<tr><td colspan="2">Test Case 1: Admin Login</td></tr>
<tr><td colspan="2">Test case objective : To login to the system</td></tr>
<tr><td colspan="2">Test case description: enter Username and Password, then presses login button. Client program contacts with server, server contacts with the database, and database checks for authentication and displays administrator page.</td></tr>
<tr><td colspan="2">Requirements Verified: Yes</td></tr>
<tr><td colspan="2">Test Environment: Apache MySQL server must be in running state, Database Should contain appropriate table and link must be established between server and client program.</td></tr>
<tr><td colspan="2">Test Setup/Pre-Conditions: Apache server should be in running state and User name and Password fields should be filled correctly.</td></tr>
<tr><td>Actions</td><td>Expected Results</td></tr>
<tr><td>The user should enter the correct user name and password to login.</td><td>Displays Admin page.</td></tr>
</table>

If user name and password are not filled correctly the system display to fill the user name and password again.

Table 6.1 Test Case: login

6.7.2 Testing Tools and Environment

Test Environment consists of elements that support test execution with software, hardware and network configured. Test environment configuration must mimic the production environment in order to uncover any environment/configuration related issues.

A typical Environmental Configuration for a web-based application is given below: For web server we use Apache server, for Database MySQL, we use widow operating system, and we use chromes browser.

6.7.3 Unit Testing

Every module of the System is separately tested. We tests every module by applying some selection mechanism. Through this mechanism every modules gets tested. If an error occurs correction will be taken without affecting another module. We has conducted this testing procedure to evaluate only the outputs generated in response to selected inputs and execution conditions. We has conducted this testing procedure during writing the code for each desired components of the system to check if the written code is working properly or not.

6.7.4 Integration Testing

In this testing part, all the modules will be combined together and tested it for its fitness with each other and with the systems functionality. If error occurs in combining them, the module with problem will be identified and recombined.

6.7.5 System Testing

In System testing we have destined garage management system to check the behavior of a complete and fully integrated system product based on the software requirements specification document.

For this type of testing do not required knowledge of internal design or structure or code. We have checks this facility system in black box type of testing. The bringing together of all the programs is that a system comprises for testing purposes. Any failed components should be migrated back to the development phase for rework, and the passed components should be migrated ahead for security testing.

Chapter Seven

Conclusion and Recommendation

7.1 Conclusion

So far we have tried to deal with the first six chapters. That mainly consists of requirement gathering, problem identification, proposing and designing new system, finally implement and test our system for the organization at hand as the fulfillment of the industrial project. The main purpose of this proposed system is to change the existing manual system to web-based garage management system. In order to gather data that used for analyzing the existing system, we used methodologies such as observation, interview, and questionnaires. The analysis of proposed system contains models such as use case diagram, sequence diagram, activity diagram, and class diagram and user interface prototype; as well as design of system based on system decomposition, finally puts implementation and testing of our system. The system security is efficient, as all the records are stored in the database can be retrieved easily by user access privilege.

Generally, after we have completed the project we are sure that the problems in the existing system would be overcome and expected that it will satisfy all users. Because the proposed system makes all the process or activities computerized to reduce human errors and to increase efficiency as well as strength relationship between customer and organization. And also the proposed user interface is made simpler and interactive to the user.

7.2 Recommendation

The system that we have developed which is web based garage management system for AutoCare Garage Plc. We recommend the following features need to be included in any further Revision and extension attempt

- ✓ May used the web based to change in to android or mobile based application.
- ✓ Include GPS to give location based service.

Who are interested to develop a new system on garage management System for AutoCare garage Plc. or other related systems can get some initial idea about the system. By Focusing on the limitation and functional areas of the system they can also develop a better garage management system.

References

[1] S. Aghaei, "Evolution of the World Wide Web : From Web 1.0 to Web 4.0," *Int. J. Web Semant. Technol.*, vol. 3, no. 1, pp. 1–10, Jan. 2012, doi: 10.5121/IJWEST.2012.3101.

[2] A. Patidar, S. Dosalwar, and T. Varshney, "An Effective Garage Management System Web Application for Customer Service," *Int. J. Comput. Appl.*, vol. 183, no. 31, pp. 18–22, 2021, doi: 10.5120/ijca2021921699.

[3] Simplilearn, "Feasibility Study And Its Importance in Project Management [Updated]." https://www.simplilearn.com/feasibility-study-article#what_is_feasibility_study_ (accessed Jan. 08, 2022).

[4] "Iterative Model: Advantages and Disadvantages |Professionalqa.com." https://www.professionalqa.com/iterative-model (accessed Feb. 26, 2022).

[5] I. Verifier and H. E. Qualifications, "Higher Nationals Internal verification of assessment decisions – BTEC (RQF)," pp. 1–47, 2016.

[6] S. W. Ambler, The Object Primer, 3rd Edition ed., Cambridge: Cambridge University press, 2004.

[7] "CS 410/510 - Software Engineering class notes." https://cs.ccsu.edu/~stan/classes/CS410/Notes16/05-SystemModeling.html (accessed Feb. 23, 2022).

[8] "An Overview Of Polymorphism, Inheritance And Encapsulation In OOP." https://www.c-sharpcorner.com/article/an-overview-of-polymorphism-inheritance-and-encapsulation-in-oop/ (accessed Jan. 08, 2022).

[9] "PTC Help Center." https://support.ptc.com/help/modeler/r9.1/en/index.html#page/Integrity_Modeler/rtsme/dynamic_modeling_overview.html (accessed Feb. 06, 2022).

[10] C. S. Faculty, "System Design : Decomposing the System," pp. 1–54, 2006.

[11] "Package Diagram Tutorial | Lucidchart." https://www.lucidchart.com/pages/uml-package-diagram (accessed Feb. 08, 2022).

Appendices

Appendix A: Unstructured Interview list

1. What are the total numbers of employees in the garage?
2. How the information is currently managed in your organizations?
3. is there any limitations regarding to available reporting formats?
4. Does your Garage have an organizational structure?
5. Does your Garage have a written or documented working processes profile?
6. Have your Garage ever identified the problems related to information management systems?
7. What is role of service advisor store keeper, technician head& manager?
8. How technician accept items?

Appendix B: Observation Check lists

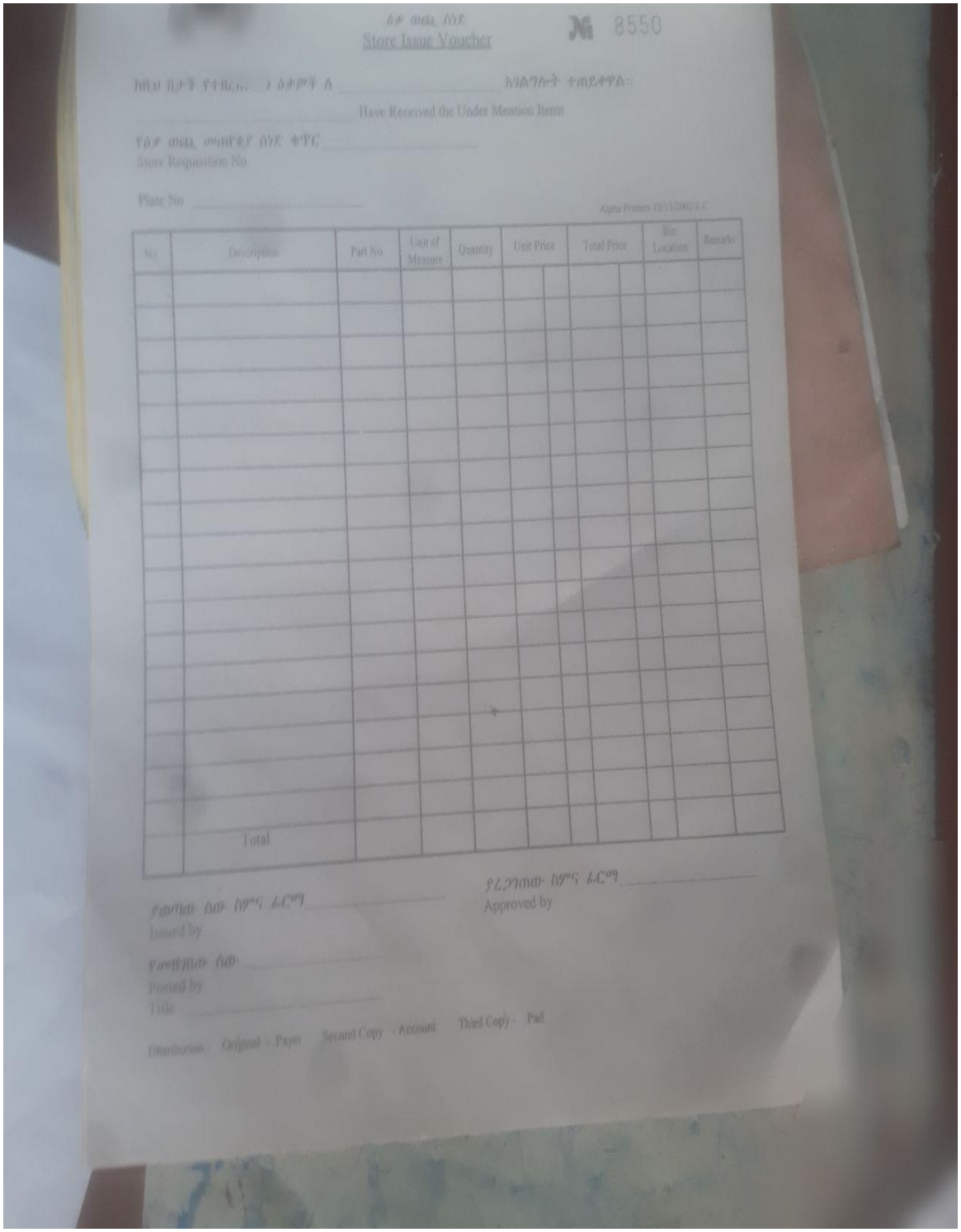
Observation Check-list of guide lines

1. Observation of the AutoCare Garage plc.
2. Organizational structure and work culture.
3. Observation of the AutoCare Garage plc. ICT Materials & items (accessories, availability, quality, etc.)
4. Observation of the AutoCare Garage's Plc. Physical environment or working condition
5. Observing the work processes activities of the whole processes of the Garage.

Appendix C Different forms

1. Current manual system for registration

2. Store issues voucher



ዕቃ ወጪ ስነድ
Store Issue Voucher
№ 8550

ከዚህ በታች የተዘረዘሩ ዕቃዎች ለ _____ አገልግሎት ተወስደዋል።
_____ Have Received the Under Mention Items

የዕቃ ወጪ መጠየቂያ ስነድ ቁጥር _____
Store Requisition No

Plate No _____

Alpha Printers 12/11/2002 E.C

No	Description	Part No	Unit of Measure	Quantity	Unit Price	Total Price	Bin Location	Remarks
	Total							

የወጣው ሰው ስምና ፊርማ _____
Issued by

የጸደቀው ሰው ስምና ፊርማ _____
Approved by

የመዘገበው ሰው _____
Posted by

Title _____

Distribution - Original - Payer Second Copy - Account Third Copy - Pad

Appendix D: Sample source Code

Sample code for registration

```
<?php
include('header.php');
$success = "none";
$user = '';
$c_name = '';
$c_email = '';
$c_address = '';
$c_mobile = '';
$c_password = '';
$title = 'Add New Customer';
$button_text="Save Information";
$successful_msg="Add Customer Successfully";
$form_url = WEB_URL . "service_advisor/addcustomer.php";
$id="";
$hdnid="0";
$image_cus = WEB_URL . 'img/no_image.jpg';
$img_track = '';
$wow = false;

/*####################################################################*/
if(isset($_POST['txtCName'])){
        if(!$wms->checkCustomerEmailAddress($link, $_POST['txtCEmail'])) {
                $image_url = uploadImage();
                if(empty($image_url)) {
                        $image_url = $_POST['img_exist'];
                }
                $wms->saveUpdateCustomerInformation($link, $_POST, $image_url);
                if((int)$_POST['customer_id'] > 0){
```

```
                $url = WEB_URL.'service_advisor/customerlist.php?m=up';
                header("Location: $url");
            } else {
                $url = WEB_URL.'service_advisor/customerlist.php?m=add';
                header("Location: $url");
            }
            exit();
        } else {
            $wow = true;
        }
}

if(isset($_GET['id']) && $_GET['id'] != ''){
        $row = $wms->getCustomerInfoByCustomerId($link, $_GET['id']);
        if(!empty($row)) {
            $user = $row['username'];
            $c_name = $row['name'];
            $c_email = $row['email'];
            $c_address = $row['address'];

            $c_mobile = $row['phone_number'];
            $c_password = $row['password'];
            if($row['image'] != ''){
                $image_cus = WEB_URL . 'img/upload/' . $row['image'];
                $img_track = $row['image'];
            }
            $hdnid = $_GET['id'];
            $title = 'Update Customer';
            $button_text="Update Information";
            $successful_msg="Update Customer Successfully";
```

```
            $form_url = WEB_URL . "service_advisor/addcustomer.php?id=".$_GET['id'];
        }
}

//for image upload
function uploadImage(){
        if((!empty($_FILES["uploaded_file"])) && ($_FILES['uploaded_file']['error'] == 0)) {
          $filename = basename($_FILES['uploaded_file']['name']);
          $ext = substr($filename, strrpos($filename, '.') + 1);
          if(($ext == "jpg" && $_FILES["uploaded_file"]["type"] == 'image/jpeg') || ($ext == "png" && $_FILES["uploaded_file"]["type"] == 'image/png') || ($ext == "gif" && $_FILES["uploaded_file"]["type"] == 'image/gif')){
                $temp = explode(".",$_FILES["uploaded_file"]["name"]);
                $newfilename = NewGuid() . '.' .end($temp);
                move_uploaded_file($_FILES["uploaded_file"]["tmp_name"], ROOT_PATH . '/img/upload/' . $newfilename);
                return $newfilename;
          }
          else{
                return '';
          }
        }
        return '';
}

?>
<!-- Content Header (Page header) -->

<section class="content-header">
  <h1><i class="fa fa-users"></i> Customer </h1>
```

```
 <ol class="breadcrumb">
   <li><a href="service_advisor_dashboard.php"><i class="fa fa-dashboard"></i> Home</a></li>
   <li><a href="customerlist.php">Customer</a></li>
   <li class="active">Add/Update Customer</li>
 </ol>
</section>
<!-- Main content -->
<form onSubmit="return validateMe();" action="<?php echo $form_url; ?>" method="post" enctype="multipart/form-data">
 <section class="content">
   <!-- Full Width boxes (Stat box) -->
   <div class="row">
     <div class="col-md-12">
     <?php if($wow) { ?>
          <div id="me" class="alert alert-warning alert-dismissable" style="display:<?php echo $delinfo; ?>">
     <button aria-hidden="true" data-dismiss="alert" class="close" type="button"><i class="fa fa-close"></i></button>
     <h4><i class="icon fa fa-ban"></i> Warning!</h4>
     Email already exist choose another one. </div>
          <?php } ?>


               <div align="right" style="margin-bottom:1%;"> <button class="btn btn-success" type="submit" data-toggle="tooltip" href="javascript:;" data-original-title="<?php echo $button_text; ?>"><i class="fa fa-save"></i></button>  <a class="btn btn-warning" title="" data-toggle="tooltip" href="<?php echo WEB_URL; ?>service_advisor/customerlist.php" data-original-title="Back"><i class="fa fa-reply"></i></a> </div>


      <div class="box box-success">
        <div class="box-header">
          <h3 class="box-title"><i class="fa fa-plus"></i> Customer Form</h3>
```

```
</div>
<div class="box-body">
 <div class="form-group">
  <label for="txtCUserName"><span style="color:red;">*</span>User Name :</label>
  <input type="text" name="txtCUserName" value="<?php echo $user;?>" id="txtCUserName" class="form-control" />
</div>
<div class="form-group">
 <label for="txtCName"><span style="color:red;">*</span> Name :</label>
 <input type="text" name="txtCName" value="<?php echo $c_name;?>" id="txtCName" class="form-control" />
</div>
<div class="form-group">
 <label for="txtCEmail"><span style="color:red;">*</span> Email :</label>
 <input type="text" name="txtCEmail" value="<?php echo $c_email;?>" id="txtCEmail" class="form-control" />
</div>
<div class="form-group">
 <label for="txtCAddress"><span style="color:red;">*</span> Address :</label>
 <textarea name="txtCAddress" id="txtCAddress" class="form-control"><?php echo $c_address;?></textarea>
</div>

<div class="form-group">
 <label for="txtCMobile"><span style="color:red;">*</span> Mobile Tel :</label>
 <input type="text" name="txtCMobile" value="<?php echo $c_mobile;?>" id="txtCMobile" class="form-control" />
</div>
<div class="form-group">
 <label for="txtCPassword"><span style="color:red;">*</span> Password :</label>
```

```
            <input type="text" name="txtCPassword" value="<?php echo $c_password;?>" id="txtCPassword" class="form-control" />
          </div>
          <div class="form-group">
            <label for="Prsnttxtarea">Priview :</label>
            <img class="form-control" src="<?php echo $image_cus; ?>" style="height:100px;width:100px;" id="output"/>
            <input type="hidden" name="img_exist" value="<?php echo $img_track; ?>" />
          </div>
          <div class="form-group"> <span class="btn btn-file btn btn-primary">Upload Image
            <input type="file" name="uploaded_file" onchange="loadFile(event)" />
            </span> </div>
        </div>
        <input type="hidden" value="<?php echo $hdnid; ?>" name="customer_id"/>
        <!-- /.box-body -->
      </div>
      <!-- /.box -->
    </div>
  </div>
 </section>
</form>
<!-- /.row -->
<script type="text/javascript">
function validateMe(){
        if($("#txtCName").val() == ''){
                alert("Customer Name is Required !!!");
                $("#txtCName").focus();
                return false;
        }
        else if($("#txtCEmail").val() == ''){
```

```
            alert("Email is Required !!!");
            $("#txtCEmail").focus();
            return false;
        }
        else if($("#txtCAddress").val() == ''){
            alert("Address is Required !!!");
            $("#txtCAddress").focus();
            return false;
        }
        else if($("#txtCHomeTel").val() == ''){
            alert("Home Tel Number is Required !!!");
            $("#txtCHomeTel").focus();
            return false;
        }
        else if($("#txtCWorkTel").val() == ''){
            alert("Work Tel Number is Required !!!");
            $("#txtCWorkTel").focus();
            return false;
        }
        else if($("#txtCMobile").val() == ''){
            alert("Mobile Tel Number is Required !!!");
            $("#txtCMobile").focus();
            return false;
        }
        else if($("#txtCPassword").val() == ''){
            alert("Password is Required !!!");
            $("#txtCPassword").focus();
            return false;
        }
        else{
```

```
                return true;
        }
}
</script>
<script type="text/javascript">
  $( document ).ready(function() {
        setTimeout(function() {
                $("#me").hide(300);
                $("#you").hide(300);
        }, 3000);
});
</script>
<?php include('footer.php'); ?>
```