

INSTA MECH

Mini Project Report Submitted by

PRINTO MATHEW

Register No: 220021087299

In partial fulfilment of the requirements for the award of the degree

Of

BACHELOR OF COMPUTER APPLICATIONS (BCA)



P.G.M COLLEGE

KANGAZHA

MAHATMA GANDHI UNIVERSITY

KOTTAYAM

2022 – 2025

DEPARTMENT OF COMPUTER APPLICATIONS

P.G.M COLLEGE

KANGAZHA



CERTIFICATE

*Certified that this project report titled **INSTA MECH** is the bonafide work of **Mr. PRINTO MATHEW** with Register Number: **220021087299** who carried out the work under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.*

Head of the Department

Mrs. Bhagya M.U

Project Co-ordinator

Mrs. Bhagya M.U

Internal Guide

Mrs. Chithra B

Submitted for the viva-voce held on.....

External Examiner

DECLARATION

I hereby declare that the mini project report “**INSTA MECH**” is a bonafide work done at PGM College Devagiri, Kangazha towards the partial fulfilment of the requirements for the award of the Degree of Bachelor of Computer Applications (BCA) from Mahatma Gandhi University, Kottayam, during the academic year 2022-2025.

Date :.....

Place: Kangazha

PRINTO MATHEW

Reg No:220021087299

ACKNOWLEDGEMENT

First and foremost, I thank God Almighty for His blessings for this miniproject. I take this opportunity to express my gratitude to all those who have helped me in completing the project successfully. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to our Principal **Prof. JEEN ABRAHAM** for providing good faculty for guidance.

I owe a great depth of gratitude towards our head of the department **Mrs. BHAGYA M.U** for helping us. I am thankful to our project guide **Mrs. CHITHRA B** for her proper guidance throughout the project. I also express my gratitude towards the supporting staffs for technical assistance.

I am indebted to my beloved teachers whose cooperation and suggestions throughout the project which helped me a lot. I also thank all my friends and classmates for their interest, dedication and encouragement shown towards the project.

I convey hearty thanks to parents for the moral support, suggestion and encouragement to make this venture a success.

PRINTO MATHEW

ABSTRACT

The **Insta Mech** project is a web application designed to offer immediate access to mechanic, crane, and carriage services for users experiencing vehicle breakdowns or other related issues while traveling. This application, provides a modern and responsive platform that ensures users can easily locate and contact service providers in their vicinity at any time, day or night.

This system addresses the current gap in roadside assistance services, where travelers often struggle to find help, especially in unfamiliar areas. This application helps to contact with verified service providers, and the application offers a reliable solution that enhances user convenience and safety.

The project consists of multiple user roles, including admins, service providers, and general users. Admins oversee the platform's operations, ensuring all registered service providers are authenticated and approved. Service providers can manage their profiles, view service requests, and communicate directly with users. General users can access detailed information about nearby services, send requests, and provide feedback.

In summary, the "Insta Mech" project aims to revolutionize roadside assistance by providing an efficient, user-friendly, and secure platform that ensures help is just a click away, no matter where the user is located.

CONTENTS

1. INTRODUCTION.....	10
1.1 PROJECT OVERVIEW.....	11
1.2 PROJECT SPECIFICATION.....	11
2. SYSTEM STUDY.....	12
2.1 EXISTING SYSTEM	13
2.2 DRAWBACKS OF EXISTING SYSTEM.....	13
2.3 PROPOSED SYSTEM.....	13
2.4 ADVANTAGES OF PROPOSED SYSTEM.....	13
3. REQUIREMENT ANALYSIS.....	14
3.1 FEASIBILITY STUDY.....	15
3.1.1 ECONOMICAL FEASIBILITY.....	15
3.1.2 TECHNICAL FEASIBILITY.....	15
3.1.3 BEHAVIOURAL/OPERATIONAL FEASIBILITY.....	16
3.2 SYSTEM SPECIFICATION.....	16
3.2.1 HARDWARE SPECIFICATION.....	16
3.2.2 SOFTWARE SPECIFICATION.....	17
3.3 SOFTWARE DESCRIPTION.....	17
3.3.1 OVERVIEW OF THE LANGUAGE USED.....	17
3.3.2 OVERVIEW OF THE DATABASE USED.....	19
4. SYSTEM DESIGN.....	21
4.1 INTRODUCTION.....	22
4.1.1 DEFINITION	22
4.1.2 OUTPUT DESIGN.....	22
4.1.3 INPUT DESIGN.....	22
4.2 MODULE DESCRIPTION.....	23
4.2.1 ADMINISTRATOR.....	23
4.2.2 MECHANIC.....	24
4.2.3 CRANE OPERATOR.....	25
4.2.4 CARRIAGE.....	25
4.2.3 USER.....	26
4.3 DATAFLOW DIAGRAM.....	27

4.4 DATABASE DESIGN.....	36
5. SYSTEM TESTING.....	43
5.1 TESTING.....	44
5.1.1 BLACK BOX TESTING.....	44
5.1.2 WHITE BOX TESTING.....	44
5.1.3 SOFTWARE TESTING STRATEGIES.....	44
6. IMPLEMENTATION.....	46
6.1 IMPLEMENTATION.....	47
7. CONCLUSION.....	48
7.1 CONCLUSION.....	49
7.2 FUTURE SCOPE.....	49
8. BIBLIOGRAPHY.....	50
9. SOURCE CODE.....	52
9.1 SAMPLE CODE.....	53
10. APPENDIX.....	70

LIST OF TABLES

1. Login.....	38
2. vehiclecategory.....	38
3. registration.....	39
4. vechiclecompany.....	39
5. servicedetails.....	40
6. servicerequest.....	40
7. forget_password.....	41
8. tbl_msg.....	41
9. bank.....	42

LIST OF FIGURES

1. LEVEL 0 DFD for INSTA MECH.....	29
2. LEVEL 1 DFD for ADMIN.....	30
3. LEVEL 1 DFD for MECHANIC.....	31
4. LEVEL 1 DFD for CRANE.....	32
5. LEVEL 1 DFD for CARRIAGE.....	33
6. LEVEL 1 DFD for USER.....	34
7. ER DIAGRAM.....	35

CHAPTER -1

INTRODUCTION

1.1 PROJECT OVERVIEW

The **Insta Mech** is a web application designed to provide immediate access to mechanic, crane, and carriage services for users facing vehicle issues while traveling. This platform allows users to quickly find and contact nearby service providers, ensuring 24/7 assistance. The system includes features for secure user registration, service provider verification, and efficient management of service requests, making it a reliable solution for roadside emergencies.

Currently, there is no existing system that provides instant access to mechanic, crane, or carriage services. Travelers experiencing vehicle problems often struggle to find assistance, especially in unfamiliar areas, relying on local knowledge or costly alternatives like hiring a taxi.

However, the project could benefit from further details on user experience design and scalability planning to accommodate a growing user base. Overall, it promises to be a valuable tool for enhancing roadside safety and convenience.

1.2 PROJECT SPECIFICATION

Insta Mech is a user-friendly, responsive web application designed to provide immediate access to mechanic, crane, and carriage services for users facing vehicle-related issues. The system offers a comprehensive list of available service providers within a specific locality, allowing users to quickly find help when needed.

This project comprised of three main modules:

The **Admin** module manages the overall functionality of the system, including verifying and approving service provider registrations, categorizing them, and handling user feedback.

The **Service Providers** module allows mechanics, crane operators, and carriage providers to register, update their profiles, and respond to service requests after admin approval.

The **User** module enables registered users to log in, access details of local service providers, and send service requests or feedback.

This system enhances roadside assistance by providing a reliable and efficient platform for both users in need and service providers looking to offer their services.

CHAPTER-2

SYSTEM STUDY

2.1 EXISTING SYSTEM

In the existing system, there is no dedicated platform that provides instant access to mechanic, crane, or carriage services for travelers facing vehicle breakdowns or other issues. When a vehicle problem occurs, especially in an unfamiliar area, the only way to get help is by trying to contact the nearest mechanic, which can be difficult if the location is unknown or if it's outside normal business hours. This lack of immediate assistance forces travelers to rely on costly alternatives like hiring a taxi or waiting for help, which can be time-consuming and inconvenient.

2.2 DRAWBACKS OF EXISTING SYSTEM

- Lack of Immediate Access
- Dependency on Local Knowledge
- Limited Availability
- Inconvenience
- Costly Alternatives

2.3 PROPOSED SYSTEM

The proposed system , **Insta Mech**, is designed to address the challenges of the existing system by providing a user-friendly, responsive web application that offers immediate access to mechanic, crane, and carriage services. This platform allows users to quickly locate and contact nearby service providers in any locality, ensuring assistance is available 24/7, regardless of the time or location. By centralizing a verified list of service providers, the system eliminates the need for users to rely on local knowledge or search extensively for help. **Insta Mech** also includes secure registration and verification processes, ensuring that all service providers are trustworthy and reliable. Users can easily log in, view detailed profiles of service providers, and send service requests or feedback, making it a convenient and efficient solution for roadside emergencies.

2.4 ADVANTAGES OF PROPOSED SYSTEM

- Instant Access to Services
- 24/7 Availability
- User-Friendly Interface
- Reduced Dependency on Local Knowledge
- Enhanced Security

CHAPTER-3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For the feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are:

- **ECONOMICAL FEASIBILITY**
- **TECHNICAL FEASIBILITY**
- **BEHAVIOURAL / OPERATIONAL FEASIBILITY**

3.1.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also, all the resources are already available, it gives an indication of the system is economically possible for development

3.1.2 TECHNICAL FEASIBILITY

Technical Feasibility deals with the hardware as well as software requirements. Technology is not a constraint to type system development. We have to find out whether the necessary technology, the proposed equipment has the capacity to hold the data, which is used in the project, should be checked to carry out this technical feasibility.

The technical feasibility issues usually raised during the feasibility stage of investigation includes these:

- This web application can run in any type of Operating System.

- The hardware required is the minimum hardware for running the applications.
- The application can be expanded easily to include more functionalities.

3.1.3 BEHAVIOURAL/OPERATIONAL FEASIBILITY

This analysis involves how it will work when the system runs in the environment in which it is implemented. The new proposed system is made with user friendly interfaces that the user can easily understand and use. The proposed system is very much useful to the users and there for it will accept broad audience from the department.

3.2 SYSTEM SPECIFICATION

The **Insta Mech** project presents a well-conceived solution to the common problem of finding immediate roadside assistance during vehicle breakdowns. By incorporating secure user registration and verification processes, the system addresses security concerns, making it reliable for both users and service providers. The application's ability to provide 24/7 access to mechanics, crane services, and carriage providers makes it highly practical for travelers. Overall, it promises to be a valuable tool for enhancing roadside safety and convenience.

The **Insta Mech** supposes to have the following features:

- It will feature a intuitive and user-friendly interface accessible to both user and services provider.
- Admins monitor system activity, handle user feedback, and manage the overall system operations.
- Service providers can update their profiles with relevant details such as contact information and service offerings.
- Users can send service requests to providers and view the status of their request and can provide feedback and file complaints regarding service providers.

3.2.1 HARDWARE SPECIFICATION

The section of the hardware is very important in the existence and proper working of any application as the proposed system is an online application it can run in any system. The basic hardware requirements are:

- **Processor:** Intel Pentium Dual Core or equivalent
- **RAM:** 1 GB
- **Hard Disk Drive:** 40 GB
- **Monitor:** Display Panel (1024 X 764)
- **Key Board:** Normal or multimedia keyboard
- **Mouse:** Logitech Serial Mouse or Compatible mouse

3.2.2 SOFTWARE SPECIFICATION

The Software Specification details the software components and tools required to develop, deploy, and maintain the **Insta Mech** web application. The basic software requirements are:

- **Front-end** : HTML, CSS
- **DataBase** : MySQL
- **Middleware** : PHP
- **Client on PC** : Any Operating System

3.3 SOFTWARE DESCRIPTION

3.3.1 FRONTEND DESCRIPTION & OVERVIEW OF THE LANGUAGE USED

HYPER TEXT MARKUP LANGUAGE (HTML)

To public information for global distribution, one needs a universally understood language, a kind of publishing mother tongue that all computers may potentially understand. The publishing language used by the World Wide Web is HTML. HTML is short for hypertext markup language. It defines the structure and layout of a web document by using a variety of tags and attributes. HTML pages contain a set of markup symbols or codes intended for display on a browser. The markup tells the browser how to display a webpage's words and images for the user. Each individual markup code is referred to as an element or tag. Some elements come in pairs (container tags) that indicate when some display effect is to begin and when it is to end. Some tags enable the document to display formatted text, color, a variety of fonts, graphic images, special effects, hypertext jumps to other.

Internet locations and information forms...HTML 5 is an advancement over the standard specification of HTML. It extends HTML with mechanisms for style sheets, scripting, frames, embedding objects, improved support for right to left and mixed direction text, richer tables and enhancements to forms offering improved accessibility for people with disabilities. HTML 5 also takes great strides towards the internationalization of documents, with the goal of making the Web truly World Wide. Early versions of HTML were defined with loose syntactical rules, which helped its adoption by those unfamiliar with web publishing. Overtime, the trend has been to create increasingly strict language syntax. HTML 5 is the current version of HTML specification. This project makes use of HTML 5 specification.

CSS

CSS, or Cascading Style Sheets, is a stylesheet language used for describing the presentation and formatting of a document written in HTML or XML. It enables web developers to control the layout, appearance, and style of multiple web pages simultaneously. CSS separates the structure of a document from its presentation, allowing for greater flexibility and consistency in the design of websites.

By using CSS, web developers can create visually appealing and consistent designs across various web pages while maintaining a clear separation between content and presentation. This separation enhances maintainability and makes it easier to update the look and feel of a website without altering its underlying structure.

WAMP

Wamp Server refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language. Wamp Server will use, by default, the latest versions of Apache, MySQL and PHP. Once Wamp Server is installed, it is possible to add as many Apache, MySQL and PHP releases as one wants. ... You can create a subfolder in which to place all your PHP files.

Wamp Server is a collection of programs you can use to turn your regular desktop PC to a fully compatible web server with HTTP, PHP, MySQL, PHP My Admin, SQL Buddy applications. The advantage is that it is easy configurable with the built-in tools. Also it is structured in the way that you have everything you need at a click distance. The configuration screens provide extra information how settings should look like. Another thing would be that the packed applications are configured to be compatible between them, all being automatic configured, you just unpack your script, database, application, run the installer (if it has any) and that's it.

MICROSOFT VISUAL STUDIO

Visual Studio is an Integrated Development Environment (IDE) developed by Microsoft to develop GUI (Graphical User Interface), console, web applications, web apps, mobile apps, cloud, and web services, etc. With the help of this IDE, can create managed code as well as native code. It uses the various platforms of Microsoft software development software like Windows store, Microsoft Silverlight, and Windows API, etc. It is not a language-specific IDE as you can use this to write code in C#, C++, VB (Visual Basic), Python, JavaScript, and many more languages. It provides support for 36 different programming languages. It is available for windows as well as for macOS. In Visual Studio

the tool windows are customizable as a user can add more windows, remove the existing open one or can move windows around to best suit.

3.3.1 BACKEND DESCRIPTION & OVERVIEW OF THE DATABASE USED

PHP

PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Preprocessor.

PHP code may be embedded into HTML or HTML5 code, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications. The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

MYSQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Wideniu's daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

FEATURES :

- Data security
- On demand scalability
- High performance
- Comprehensive Transactional Support
- Complete workflow control
- Reduced total cost of ownership
- Backup and recovery
- Security and authorization

CHAPTER- 4

SYSTEM DESIGN

4.1 INTRODUCTION

4.1.1 DEFINITION

The most creative and challenging face of the system development is System Design. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Design goes through the logical and physical stages of development. In designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfil. The first step is to determine how the output is to be produced and in what format. Second, input data and master files have to be designed to meet the requirements of the proposed output. The operational phases are handled through program construction and testing. Design of a system can be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Thus, system design is a solution to “how to” approach to the creation of a new system. Thus, important phase provides the understanding and the procedural details necessary for implementing the system recommended in the feasibility study. The design step provides a data design, architectural design, and a procedural design.

4.1.2 OUTPUT DESIGN

In the output design, the emphasis is on producing a hard copy of the information requested or displaying the output on the CRT screen in a predetermined format. Two of the most output media today are printers and the screen. Most users now access their reports from a hard copy or screen display. Computer's output is the most important and direct source of information to the user, efficient, logical, output design should improve the systems relations with the user and help in decision-making. As the outputs are the most important source of information to the user, better design should improve the system's relation and also should help in decision-making. The output device's capability, print capability, print capability, response time requirements etc. should also be considered form design elaborates the way output is presented and layout available for capturing information. It's very helpful to produce the clear, accurate and speedy information for end users.

4.1.3 INPUT DESIGN

In the input design, user-oriented inputs are converted into a computer-based system format. It also includes determining the record media, method of input, speed of capture and entry on to the screen. Data entry accepts commands and data through a keyboard. The major approach to input design is the menu and the prompt design. In each alternative, the user's options are predefined. The data flow diagram indicates logical data flow, data stores, source and destination. Input data are collected and organized into a group of similar data. Once identified input media are selected for processing.

In this online application, importance is given to develop Graphical User Interface (GUI), which is an important factor in developing efficient and user-friendly application. For inputting user data, attractive forms are designed. User can also select desired options from the menu, which provides all possible facilities.

Also, the important input format is designed in such a way that accidental errors are avoided. The user has to input only just the minimum data required, which also helps in avoiding the errors that the users may make. Accurate designing of the input format is very important in developing efficient application. The goal of input design is to make entry as easy, logical and free from errors.

4.2 MODULE DESCRIPTION

INSTA MECH includes following main modules:

- ADMINISTRATOR
- MECHANIC
- CRANE OPERATOR
- CARRIAGE
- USER

4.2.1 ADMINISTRATOR

The admin acts as a central authority responsible for overseeing, verifying, and managing the interactions between users, service providers, and the system. The main functionalities are:

- **Login:**

Admin needs to login with username and password.

- **Add/Delete/Update Vehicle Category:**

Admin can add vehicle category and he can update or delete.

- **Add/Delete/Update Vehicle Company:**

Admin can add vehicle company and he can update or delete.

- **Approve/Reject Service Provider Request:**

Admin can view, approve or reject service providers (mechanic, crane, carriage) requests by verifying their details like identity, qualifications, experience.

- **View all Approved Request:**

Admin can view all the approved request.

- **View all Rejected Request:**

Admin can view all the Rejected request.

- **View Services provided by Service providers:**

Admin can view all the services provided by mechanic, crane, carriage.

- **View Service Request:**

Admin can view all the services request from users.

- **View Service Request Paid:**

Admin can check whether a user had paid for the service.

- **View Feedback & Send response:**

Admin can view feedback send by the users and send response to them.

4.2.2 MECHANIC

Mechanic can register and provide their services to the user. The mechanic plays an essential role by offering vehicle repair and maintenance services to users who face breakdowns or other automotive issues while on the road.

- **Registration:**

Mechanic can register by giving his/her necessary details like name ,phone number, address, state, district, city, email, profile, etc..

- **Login:**

After getting admin approval the mechanic can login by username and password.

- **Change Password:**

Mechanic can change their password.

- **View/Update Profile:**

Mechanic can view and update his/her account details.

- **View/Add/Delete Service Details:**

Mechanic can view, add, and delete service details.

- **Premium Account:**

Mechanic can pay 100 rupees to make their account to premium; after that their service will be displayed on the home page.

4.2.3 CRANE OPERATOR

Crane can register and provide their services to the user. The mechanic plays an essential role by offering vehicle repair and maintenance services to users who face breakdowns or other automotive issues while on the road.

- **Registration:**

Crane can register by giving his/her necessary details like name ,phone number, address, state, district, city, email, profile, etc..

- **Login:**

After getting admin approval the crane can login by username and password.

- **Change Password:**

Crane can change their password.

- **View/Update Profile:**

Crane can view and update his/her account details.

- **View/Add/Delete Service Details:**

Crane can view, add, and delete service details.

- **Premium Account:**

Crane operators can pay 100 rupees to make their account to premium; after that their service will be displayed on the home page.

4.2.4 CARRIAGE

Carriage providers can register and provide their services to the user. The mechanic plays an essential role by offering vehicle repair and maintenance services to users who face breakdowns or other automotive issues while on the road.

- **Registration:**

Crane can register by giving his/her necessary details like name ,phone number, address, state, district, city, email, profile, etc..

- **Login:**

After getting admin approval the crane can login by username and password.

- **Change Password:**

Carriage operators can change their password.

- **View/Update Profile:**

Carriage operators can view and update his/her account details.

- **View/Add/Delete Service Details:**

Carriage operators can view, add, and delete service details.

- **Premium Account:**

Carriage providers can pay 100 rupees to make their account to premium; after that their service will be displayed on the home page.

4.2.5 USER

User can register, login and specify their requirements and send feedback to admin.

- **Registration:**

Users can register by giving his/her necessary details.

- **Login:**

After getting admin approval the users can login by username and password.

- **Change Password:**

Users can change password.

- **View/Update Profile:**

User can view and update his/her profile.

- **Payment:**

User can pay for the service they have been provided.

- **Send Feedback/ View Response:**

User can send feedback to admin detailing about the service provided/about the service provider and can view the response of the admin regarding the feedback.

- **Search Mechanic/Crane/Carriage operators:**

User can search mechanic, crane operator, carriage providers.

- **Send service requests:**

Users can send service request to the service providers.

4.3 DATA FLOW DIAGRAM

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. It's a structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process-oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc.

The purpose of the design is to create architecture for the evolving implementation and to establish the common tactical policies that must be used by desperate elements of the system. We begin the design process as soon as we have some reasonably completed model of the behaviour of the system. It is important to avoid premature designs, wherein develop designs before analysis reaches closer. It is important to avoid delayed designing where in the organization crashes while trying to complete an unachievable analysis model.

Throughout the project, the context flow diagrams, data flow diagrams and flow charts have been extensively used to achieve the successful design of the system. In our opinion, "efficient design of the data flow and context flow diagrams helps to design the system successfully without much major flaws within the scheduled time". This is the most complicated part in a project. In the designing process, our project took more than the activities in the software life cycle. If we design a system efficiently with all the future enhancements, the project will never become junk and it will be operational.

The data flow diagrams were first developed by Larry Constantine as way for expressing system requirements in graphical form. A data flow diagram also known as "bubble chart" has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. It functionally decomposes the requirement specification down to the lowest level. DFD depicts the information flow, the transformation flow and the transformations that are applied as data move from input to output. Thus DFD describes what data flows rather than how they are processed. Data Flow Diagram is quite effective, especially when the required design is unclear and the user and analyst need a notational language for communication. It is one of the most important tools used during system analysis. It is used to model the system components such as the system process, the data used by the process, any external entities that interact with the system and information flows in the system. Four basic symbols are used to construct data flow diagrams. They are symbols that represent data

source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

Steps to Construct Data Flow Diagrams: -

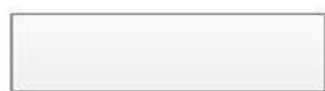
Four steps are commonly used to construct a DFD

- Process should be named and numbered for easy reference. Each name should be representative of the process.
- The destination of flow is from top to bottom and from left to right.
- When a process is exploded in to lower level details they are numbered.
- The names of data stores, sources and destinations are written in capital letters.

Rules for constructing a Data Flow Diagram

- Arrows should not cross each other.
- Squares, circles and files must bear names.
- Decomposed data flow squares and circles can have same names.
- Draw all data flow around the outside of the diagram.

Data Flow Diagram Symbols:



Source or Destination of data



Data Flow

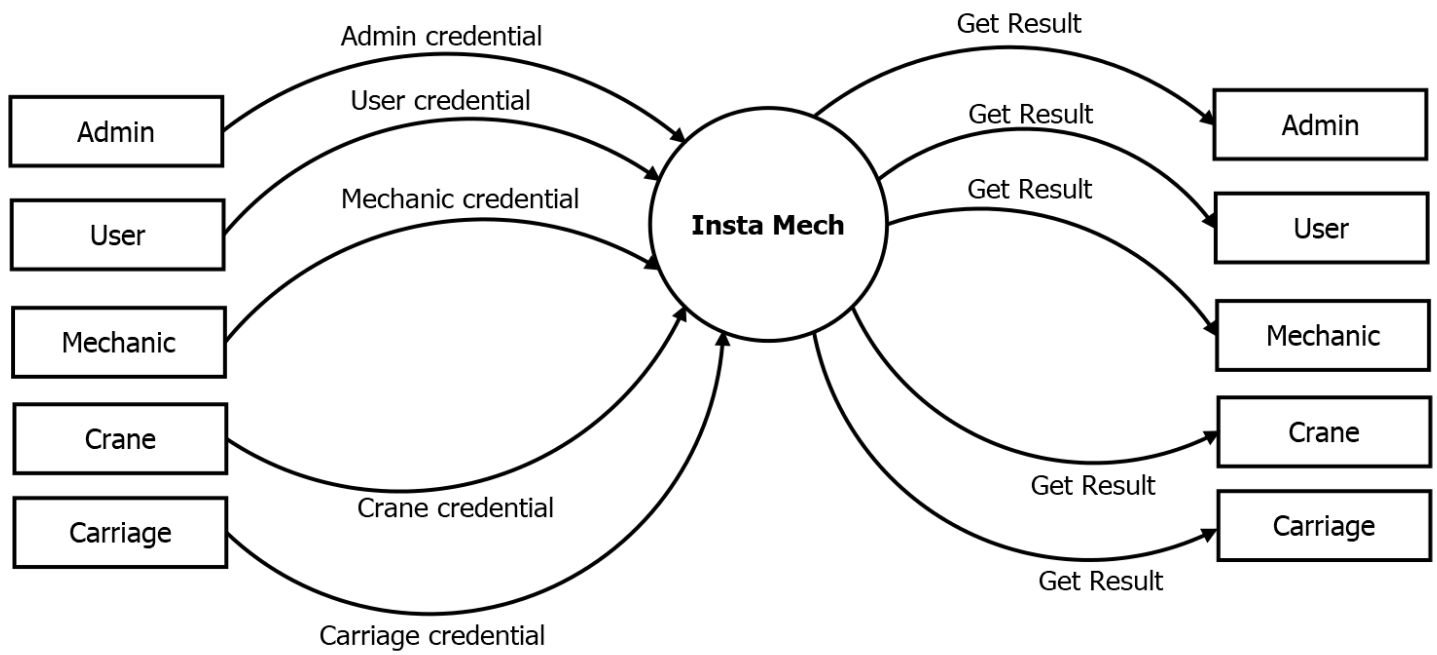


Process

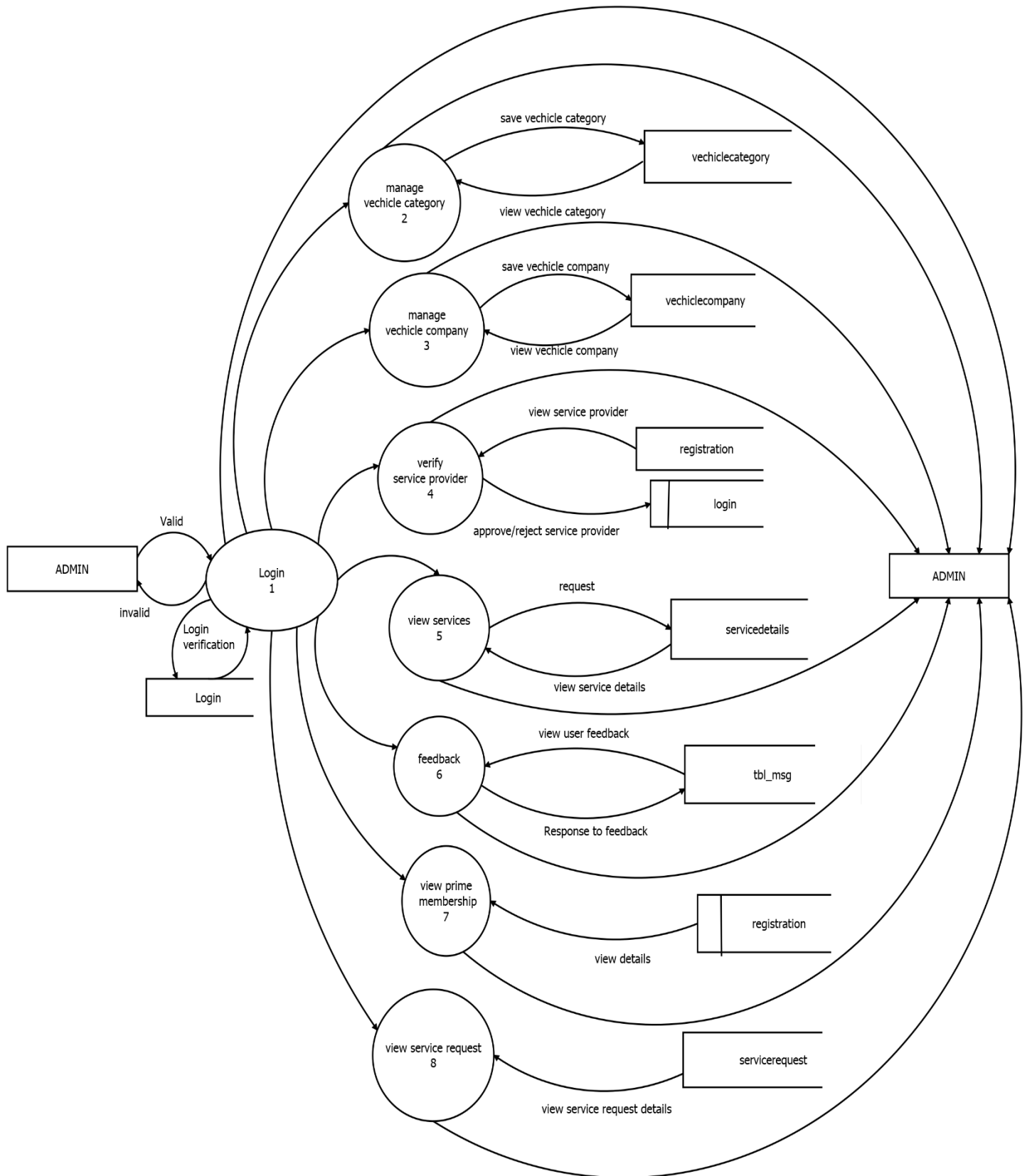


Database

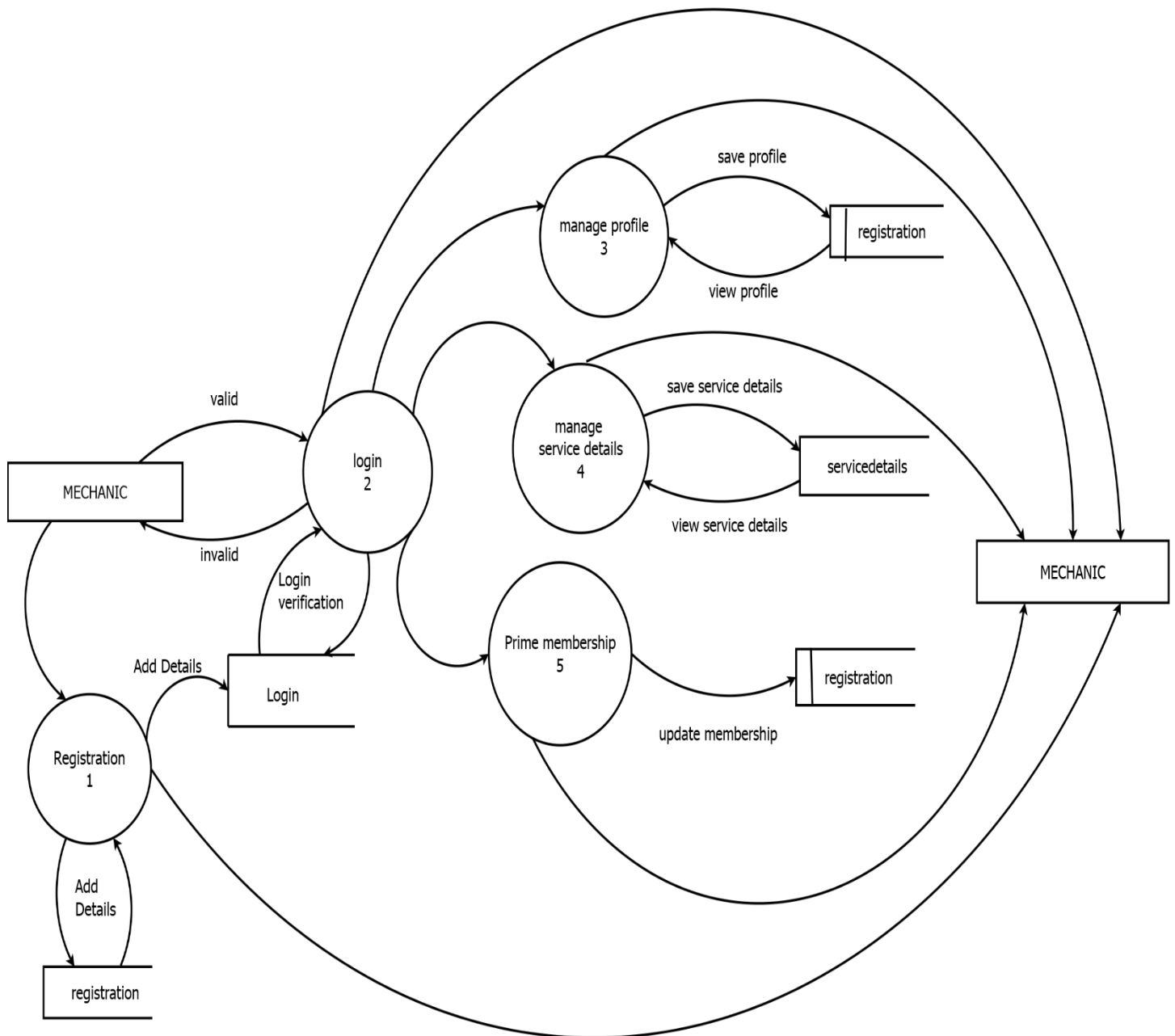
LEVEL – 0 DFD for INSTA MECH



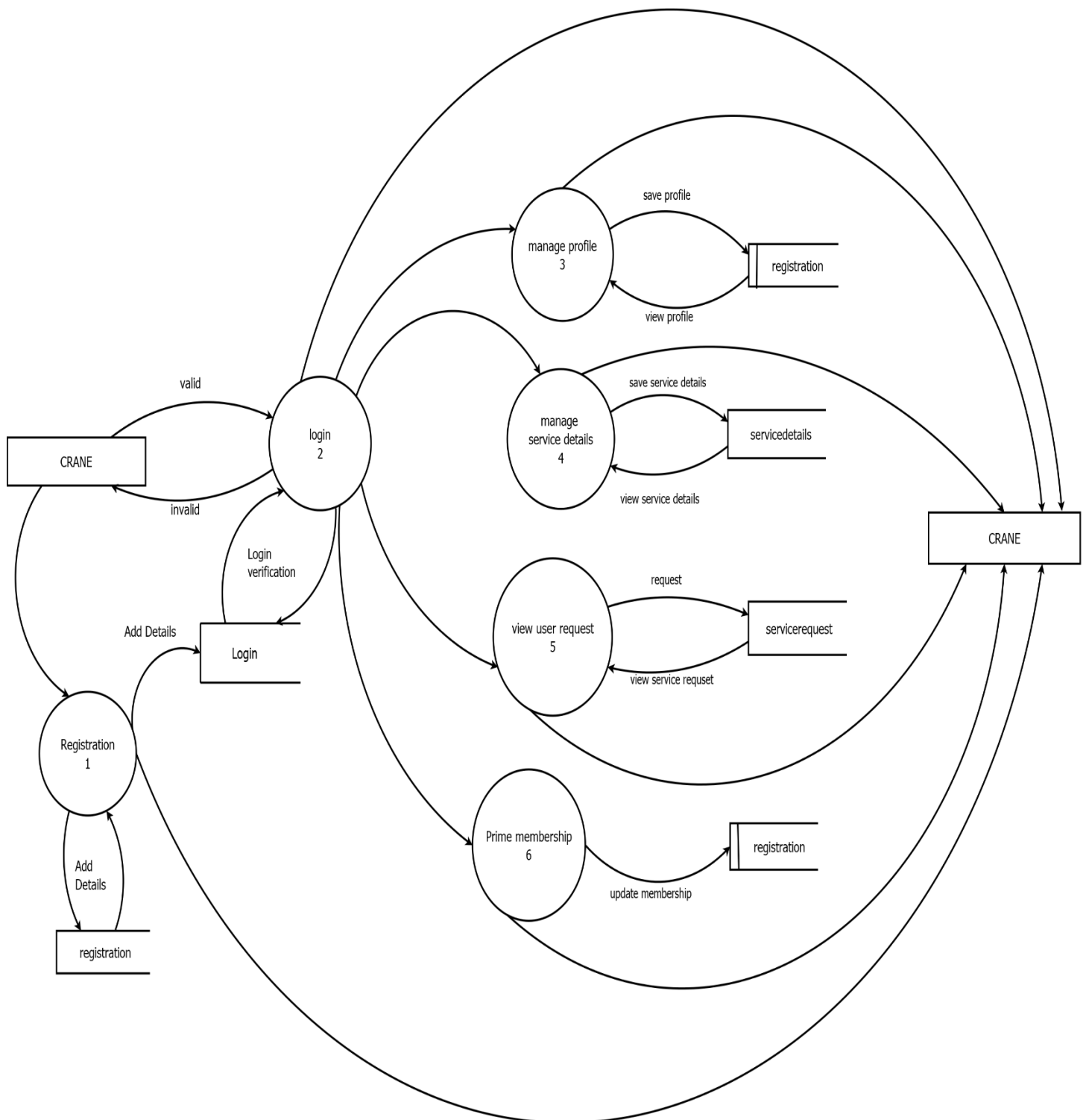
LEVEL – 1 DFD for ADMIN



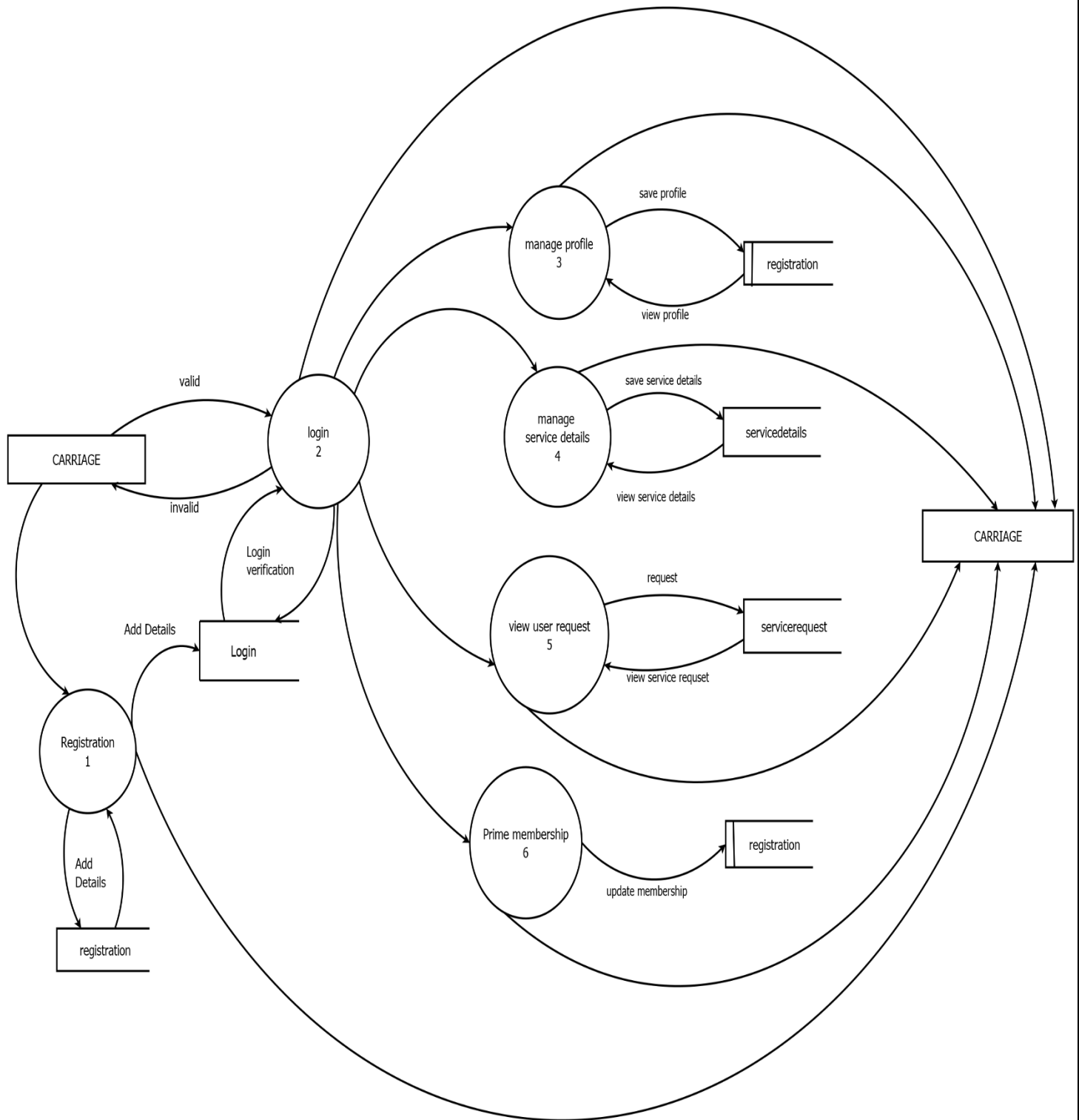
LEVEL – 1 DFD for MECHANIC



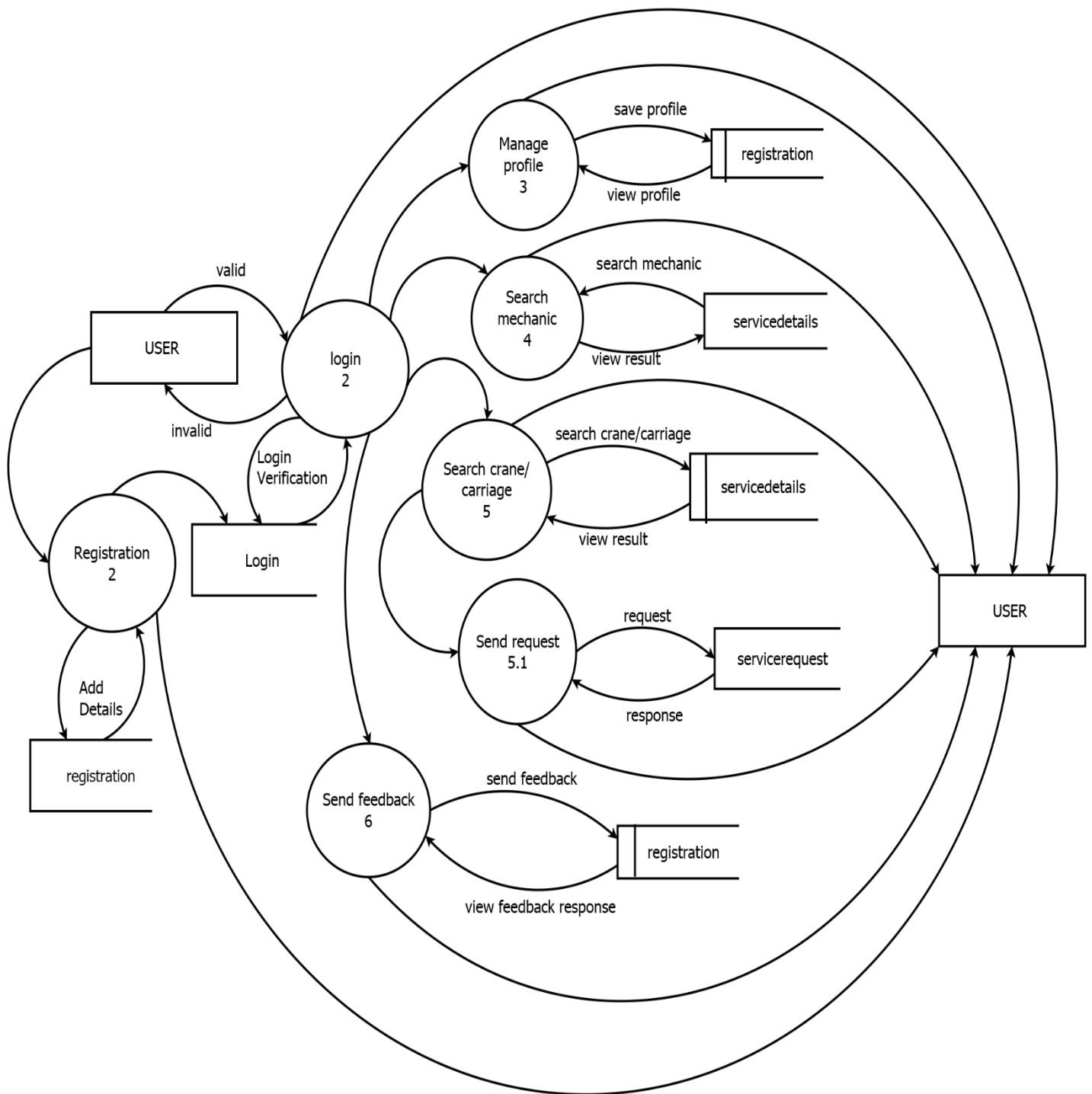
LEVEL – 1 DFD for CRANE



LEVEL – 1 DFD for CARRIAGE

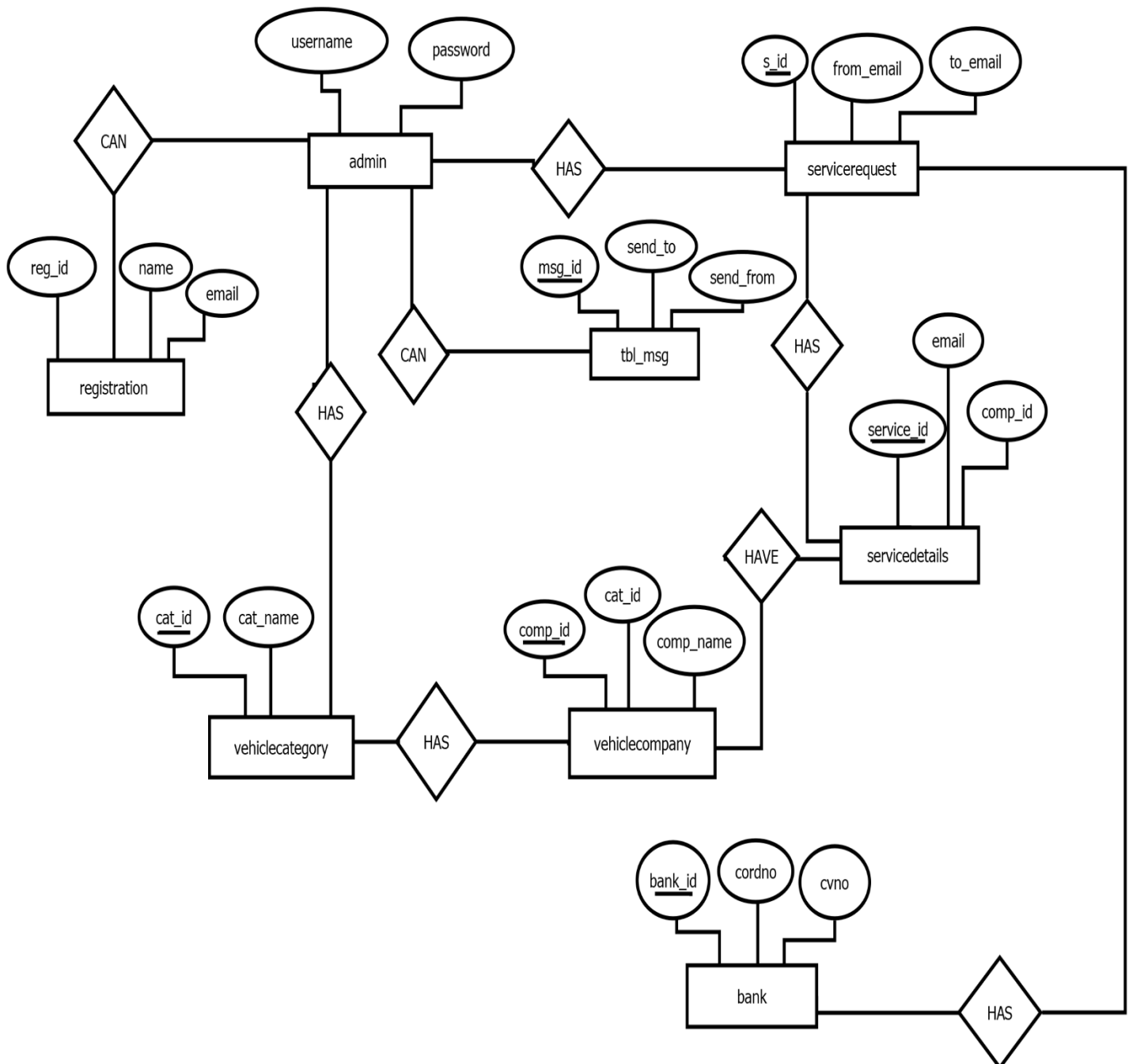


LEVEL – 1 DFD for USER



4.4 ERD -ENTITY RELATIONSHIP DIAGRAM

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology(IT) system. An ERD uses data modeling techniques that can help define business processes and serve as the foundation for a relational database.



4.5 DATABASE DESIGN

Database

A database is a collection of data. By data, we mean known facts that can be recorded and that have implicit meaning. For example, consider the names, telephone numbers and addresses of people you know. You may have recorded this data on an indexed address book, or you may have stored it on a hard drive using a personal computer and software such as Microsoft Access or Excel. This is a collection of related data with an implicit meaning and hence is a database. A database has the following implicit properties:

- A database represents some aspect of the real world, sometimes called the mini world or the Universe of Discourse (UOD). Changes to the mini world are reflected in the database.
- A database is a logically coherent collection of data with some inherent meaning. A random assortment of data cannot correctly be referred to as a database. A database is designed, built and populated with data for specific purpose. It has intended group of users and some preconceived applications in which these users are interested.

Data Base Management System

A database management system (DBMS) is a collection of programs that enable users to create and maintain a database. The DBMS is hence general-purpose software system that facilitates processes of defining, constructing, manipulating, and sharing databases among various users and applications. Defining a database involves specifying the data types, structure and constraints for the data to be stored in the database. Constructing the database is the process of storing the data itself on some storage medium. Manipulating a database includes such functions as querying the database to retrieve specific data, updating the database to reflect changes in the mini world, and generating reports from the data. Sharing a database allows multiple users and programs to access the database concurrently. This section describes the structure of data and how the data are stored, processed and organized. RDBMSs have become a predominant choice for the storage of information in new databases. Here used the EAV model in the data base design. This model is an optimal approach to data modelling for a problem domain. In an EAV data model, each attribute-value pair is a fact describing an entity, and a row in an EAV table stores a single fact. EAV model helps to create "long and skinny" databases. By "long" it refers to the number of rows, by "skinny" it refers to the fewer columns.

DATA NORMALIZATION

Normalization is a systematic approach of decomposing tables to eliminate data redundancy and undesirable characteristics like insertion, update and deletion anomalies. It is two-step process that put data into tabular form by removing duplicate data from the relation tables.

Normalization used for mainly two purpose:

- Eliminating redundant data.
- Ensuring data dependencies make sense, i.e., data is logically stored.

NORMALIZATION RULE

Normalization rules are divided into following normal forms

- First normal form
- Second normal form
- Third normal form
- Boyce codd normal form

FIRST NORMAL FORM (1NF)

A relational R is in 1NF if all underlying domains contains atomic values.

First Normal Form put two restrictions,

- 1) Fields of an n-set should simple, atomic values.
- 2) N-set should have no repeating groups.

SECOND NORMAL FORM (2NF)

A relational R is in 2NF if and only if it is 1NF and every non-key attribute is fully functionally depending on the primary key. An attribute is a non-key if it does not particularly in the primary key.

THIRD NORMAL FORM (3NF)

A relation R is in 3NF if and only if it is 2NF and every non-key. Attribute is non-transitively dependent on the primary key.

BOYCE CODD NORMAL FORM (BCNF)

A table is in Boyce-Codd Normal Form (BCNF) if:

1. It is in the Third Normal Form (3NF).
2. For every non-trivial functional dependency ($X \rightarrow Y$), X is a superkey.

TABLE STRUCTURE

The main tables of **INSTA MECH** includes the following:

Table No: 1

Table Name: login

Primary Key: username

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
username	Varchar	Primary Key	50	Unique identifier for the login table.
password	Varchar		50	Password
user_type	Varchar		50	User type
status	Varchar		50	Status

Table No: 2

Table Name: vechiclecategory

Primary Key: cat_id

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
cat_id	int	Primary Key	12	Unique Identifier for the vechicle categeory table.
cat_name	varchar		50	Specify the category name

Table No: 3**Table Name: registration****Primary Key: reg_id**

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
reg_id	int	Primary Key	12	Unique Identifier for the registration table
Name	varchar		50	Specify the name
Email	varchar		50	Specify the email
Phone	varchar		50	Specify the phone
Address	varchar		100	Specify the address
reg_type	varchar		50	Specify the registration type
State	varchar		50	Specify the state
City	varchar		50	Specify the city
District	varchar		50	Specify the district
Paystatus	varchar		50	Specify the pay status
Picture	varchar		100	Upload the picture

Table No: 4**Table Name: vehiclecompany****Primary Key: comp_id****Foregin Key: cat_id**

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
comp_id	int	Primary Key	12	Unique Identifier for the vehicle company table
cat_id	int	Foreign Key	12	Unique Identifier for the category table
comp_name	varchar		50	Specify company name

Table No: 5**Table Name: servicedetails****Primary Key : service_id**

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
service_id	int	Primary Key	12	Unique Identifier for the service details table
s_name	varchar		50	Specify users name
s_description	text			Specify the name
s_rate	int		12	Specify the description
email	varchar		50	Specify the email
comp_id	int	Foreign Key	12	Specify the company id
picture	varchar		200	Upload the picture
status	varchar		200	Specify the status

Table No: 6**Table Name: servicerequest****Primary Key: s_id**

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
s_id	int	Primary Key	12	Unique Identifier for the service request table
from_email	varchar		50	Specify the email
to_email	varchar		50	Specify the email
s_date	date			Specify the date
description	text			Specify the description
date	date			Specify the date
status	varchar		10	Specify the status

Table No: 7**Table Name: forget_password****Primary Key: forgetpassword_id**

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
forgetpassword_id	int	Primary Key	12	Unique Identifier for the forget password table
email_id	varchar		70	Specify the email
random_number	int		12	Specify the random number

Table No: 8**Table Name: tbl_msg****Primary Key: msgid**

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
msgid	int	Primary Key	12	Unique Identifier for the tbl_msg table
send_from	varchar		50	Specify the email
send_to	varchar		50	Specify the email
date	date			Specify the date
description	varchar		100	Specify the description
response	varchar		20	Specify the response
status	varchar		50	Specify the status
type	varchar		15	Specify the type

Table No: 9**Table Name: bank****Primary Key: bank_id**

FIELD NAME	DATA TYPE	CONSTRAINTS	SIZE	DESCRIPTION
bank_id	int	Primary Key	12	Unique Identifier for the tbl_msg table
cvvno	int		12	Specify the
expirmonth	varchar		50	Specify the expire month
expiryear	varchar		50	Specify the expire year
amount	int		25	Specify the amount
accnumber	varchar		30	Specify the account number
cardno	varchar		30	Specify the card number
user_type	varchar		20	Specify the user type

CHAPTER – 5

SYSTEM TESTING

5.1 TESTING

Testing is a process to show the correctness of the program. Testing is needed to show completeness, it improves the quality of the software and to provide the maintenance aid. Some testing standards are therefore necessary reduce the testing costs and operation time. Testing software extends throughout the coding phase and it represents the ultimate review of configurations, design and coding. Based on the way the software reacts to these testing, we can decide whether the configuration that has been built is study or not. All components of an application are tested, as the failure to do so many results in a series of bugs after the software is put to use.

5.1.1 BLACK BOX TESTING

Black box testing, also called behavioural testing, focuses on the functional requirements of software. This testing approach enables the software engineer to derive the input conditions that will fully exercise all requirements for a program. Black box testing attempts to find the errors like,

- Incorrect or missing functions
- Interface errors
- Errors in data structures or external database access
- Behaviour or performance error
- Initialization and termination errors
- In Black box testing software is exercised over a full range of inputs and outputs are observed for correctness

5.1.2 WHITEBOX TESTING

White box testing is also called Glass box testing is a test case design control; structure of the procedural design to derive test cases using White box testing method, the software engineer can derive the test cases that guarantee that all independent paths within the module have been exercised at least once. Exercise all logic decisions on their true or false sides. Execute all loops at their boundaries and within their operational bounds. Exercise internal data structure to ensure their validity.

5.1.3 SOFTWARE TESTING STRATEGIES

Software Testing involves,

- Unit testing
- Integration testing
- System testing

- Acceptance Testing

The first level of test is unit testing. The purpose of unit testing is to ensure that each program is fully tested.

The second step is integration testing. In this individual program units or programs are integrated and tested as a complete system to ensure that the software requirements are met.

System testing involves a series of tests to verify that all system elements have been properly integrated it also attempts to verify the software's protection mechanisms. It also tests the run time performance of a system within the context of an integrated system. It also includes stress testing in which the system is executed in a manner that demands resources in abnormal quantity, frequency or volume.

Acceptance Testing involves planning and the execution of various types of tests in order to demonstrate that the implemented software system satisfies the requirements. Finally our project meets the requirements after going through all the levels of testing.

CHAPTER - 6

IMPLEMENTATION

6.1 IMPLEMENTATION

Implementation is the process of converting the system design into code, testing the system and giving the user training. Implementation of a new system design is a crucial phase in the system development life cycle.

Implementation simply means converting a new system design into operation. This involves creating computer compatible files, training the operating staff and installing hardware, terminals and telecommunications network before the system is up and running. A critical factor in conversion is not disrupting the functioning of the organization. There are three types of implementation.

- Implementation of computer system to replace manual system.
- Implementation of new computer system to replace existing system.
- Implementation of modified application to replace an existing one.

Implementation walkthroughs ensure that the completed system actually solves the original problem. This walkthrough occurs just before the system goes into use, and it should include careful review of all manuals, training materials and system documentation. Again, users, the analyst and the members of the computer services staff may attend this meeting. Required user training should be provided for the company to use the panel, to create database, database user, assigning user to database and for uploading to the appropriate directory.

By following this implementation plan, the **INSTA MECH** project will be developed systematically, ensuring all features are integrated, tested, and deployed effectively. The focus on security, user experience, and scalability will help the application serve its intended purpose efficiently and reliably.

CHAPTER - 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

The **InstaMech** project successfully addresses the critical need for immediate roadside assistance by providing a user-friendly platform that connects individuals with nearby mechanics, crane services, and carriage providers. By leveraging modern web technologies such as HTML5, CSS3, Bootstrap, JQuery, PHP, and MySQL, InstaMech offers a reliable and scalable solution that works seamlessly across different devices. The platform's core features, including real-time service requests, secure user authentication, and an admin-managed verification system, ensure that users can receive prompt and trustworthy assistance, regardless of location or time of day.

7.2 FUTURE SCOPE

The **Insta Mech** platform has great potential for growth and development in the future, with several enhancements and expansions that could be implemented to improve functionality and user experience. With a well-structured implementation, focusing on performance, advanced searching, availability of the service provider, user-request, security, and user experience. By continuously gathering user feedback and iterating on the system, **Insta Mech** can evolve and expand its services, ensuring long-term value and success in the market.

BIBLIOGRAPHY

BOOKS REFERED**AUTHOR**

Analysis and Design of Information System	:	James A Senn
Introduction to Data Base Management System	:	Naveen Prakash
Ideas to develop a literature review	:	Cooper. M

www.w3schools.com

www.geeksforgeeks.org

CHAPTER - 9

SOURCE CODE

9.1 SAMPLE CODE

Code for connection

```
<?php
class Connectionclass
{
public $db = null;
public function open()
{
$dbname ="insta";
$this->db = mysqli_connect("localhost","root","", "$dbname") or die('Error connecting to MySQL
server.');
```

```
}
public function Manipulation($qry)
{
$this->open();
$response =array();
try
{
$result=mysqli_query($this->db,$qry);
if($result)
$response['status']="true";
else
throw new Exception(mysqli_error($this->db));
}
catch(Exception $e)
{
$response['status']="false";
$response['Message']= $e->getMessage();
}
mysqli_close($this->db);
return ($response);
```

```
}  
public function GetTable($qry)  
{  
    $this->open();  
    try  
    {  
        $result=mysqli_query($this->db,$qry);// or throw new Exception("Error in Query");  
        if($result)  
        {  
            $data = array(); // create a variable to hold the information  
            while ($row = mysqli_fetch_assoc($result))  
            {  
                $data[] = $row;  
            }  
            return $data;  
        }  
        else  
        throw new Exception(mysqli_error($this->db));  
    }  
    catch(Exception $e)  
    {  
        return $e->getMessage();  
    }  
    mysqli_close();  
}  
public function GetSingleData($qry)  
{  
    $this->open();  
    try  
    {  
        $result=mysqli_query($this->db,$qry);// or throw new Exception("Error in Query");  
        if($result)
```

```
{
$row = mysqli_fetch_row($result) ;
return ($row)!=null ?$row[0]:"";
}
else
{
throw new Exception(mysqli_error($this->db));
}
}
catch(Exception $e)
{
return $e->getMessage();
}
mysqli_close($con);
}

public function GetSingleRow($qry)
{
$this->open();
try
{
$result=mysqli_query($this->db,$qry);// or throw new Exception("Error in Query");
if($result)
{
$row = mysqli_fetch_array($result) ;
return $row;
}
else
throw new Exception(mysqli_error($this->db));
}
catch(Exception $e)
{
return $e->getMessage();
}
```

```
}  
mysqli_close($con);  
}  
public function GenerateID($qry,$num)  
{  
    $this->open();  
    try  
    {  
        $result=mysqli_query($qry);// or throw new Exception("Error in Query");  
        if($result)  
        {  
            $row = mysqli_fetch_row($result) ;  
            if(empty($row[0]))  
            {  
                $maxid=($num+1);  
            }  
            else  
            {  
                $maxid=$row[0]+1;  
            }  
            return $maxid;  
        }  
        else  
        throw new Exception(mysqli_error($this->db));  
    }  
    catch(Exception $e)  
    {  
        return $e->getMessage();  
    }  
    mysqli_close();  
}  
function alert($msg, $url=null)
```



```
{
echo $msg= str_replace("", "", $msg);
echo "
<script type='text/javascript'>
alert('".$msg."');
location.href='".$url."";
</script>";
}
public function getpostedby()
{
session_start();
$username=$_SESSION['username'];
return $username;
}
public function getcurtime()
{
$date= date('Y-m-d');
return $date;
}
}
?>
```

register.php

```
<?php

require_once("header.php");

require_once('../Connectionclass.php');

?>

<html >

<head>

<title> InstaMech Online Register Form </title>

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta charset="utf-8" />

<script>

addEventListener("load", function() { setTimeout(hideURLbar, 0); }, false); function hideURLbar(){
window.scrollTo(0,1); }

</script>

<link href="register_templates/css/style.css" rel='stylesheet' type='text/css' media="all">

<link href="//fonts.googleapis.com/css?family=Saira+Condensed:400,500,600,700" rel="stylesheet">

</head>

<body>

<h1 class="header-w3ls"> Instamech Online Register </h1>

<div class="online-w3l-form">

<div class="appointment-w3">

<form action="codes/register_exe.php" method="post" enctype="multipart/form-data">

<div class="main">

<div class="form-left-w3l">
```

```
<input type="text" class="top-up" name="name" placeholder="Name" required="" pattern="[a-zA-Z\s]+" title="Alphabets and Space ">
```

```
</div>
```

```
<div class="form-right-w3ls ">
```

```
<input class="button" type="text" name="phone_number" placeholder="Phone Number" required="" pattern="[9876][0-9]{9}" title="Enter a valid mobile number" maxlength="10" minlength="10">
```

```
<div class="clearfix"></div>
```

```
</div>
```

```
</div>
```

```
<div class="form-add-w3ls">
```

```
<input type="text" name="address" placeholder="Address" required="" >
```

```
</div>
```

```
<div class="main">
```

```
<div class="form-left-w3l">
```

```
<input type="text" name="state" placeholder="State" required="" pattern="[a-zA-Z\s]+" title="Characters only">
```

```
</div>
```

```
<div class="form-right-w3ls ">
```

```
<input type="text" name="district" placeholder="District" required="" pattern="[a-zA-Z]+" title="Characters only">
```

```
<div class="clearfix"></div>
```

```
</div>
```

```
</div>
```

```
<div class="main">
```

```
<div class="form-left-w3l">
```

```
<input type="text" name="city" placeholder="City" required="" pattern="[a-zA-Z]+" title="Characters only">
```

```
</div>
```

```
<div class="form-left-w3l">
```

```
<input type="email" name="email" placeholder="Email" required="" pattern="^[^s@]+@[^s@]+\.[^s@]+" title="Invalid Email Format">
```

```
<div class="clearfix"></div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="main">
```

```
<div class="form-left-w3l">
```

```
<label style="color: white;">Profile photo</label>
```

```
<br>
```

```
<input type="file" name="photo" placeholder="Photo">
```

```
</div>
```

```
<div class="form-right-w3ls">
```

```
<select class="form-control buttom" name="register_type">
```

```
<option value="">Register Type</option>
```

```
<option value="mechanic">Mechanic</option>
```

```
<option value="crane">Crane</option>
```

```
<option value="carriage">Carriage</option>
```

```
<option value="customer">Customer</option>
```

```
</select>
```

```
<div class="clearfix"></div>
```

```
</div>

</div>

<div class="main">

<div class="form-left-w3l">

<input type="password" name="password" placeholder="Password" required="">

</div>

</div>

<div class="btnn">

<input type="submit" value="Register">

</div>

</form>

</div>

</div>

<div class="copy">

<p>&copy;2024 Instamech Online Register. All Rights Reserved </p>

</div>

</body>

</html>

<?php

require_once("footer.php");

?>
```

APPROVAL

mechanic_list.php

```
<?php

require_once("header.php");

require_once("sidebar.php");

require_once('../Connectionclass.php');

$obj = new Connectionclass();

$type=$_REQUEST['type'];

if($type=='approve')

{

    $qry="select * from registration INNER JOIN login ON registration.email=login.username where
    registration.reg_type='mechanic' and login.status='active'";

    $exe=$obj->GetTable($qry);}

elseif($type=='reject')

{

    $qry="select * from registration INNER JOIN login ON registration.email=login.username where
    registration.reg_type='mechanic' and login.status='rejected'";

    $exe=$obj->GetTable($qry);

}

?>

<div class="main-panel">

<div class="content">

<div class="row">

<div class="col-lg-12">

<div class="card">
```

```

<div class="card-body">
<section class="tables-section">
<div class="outer-w3-agile mt-3">
<h4 class="tittle-w3-agileits mb-4">mechanic list</h4>
<table class="table">
<thead>
<tr>
<th scope="col">#</th>
<th scope="col">Name</th>
<th scope="col">Email</th>
<th scope="col">Phone</th>
<th scope="col">Address</th>
<th scope="col">State</th>
<th scope="col">City</th>
<th scope="col">District</th>
<th scope="col" colspan="2">Actions</th>
</tr>
</thead>
<tbody>
<?php
$i=1;
foreach($exe as $m)
{
?>

```

```

<tr>

<th scope="row"><?php echo $i++; ?></th>

<td><?php echo $m['name']; ?></td>

<td><?php echo $m['email']; ?></td>

<td><?php echo $m['phone']; ?></td>

<td><?php echo $m['address']; ?></td>

<td><?php echo $m['state']; ?></td>

<td><?php echo $m['city']; ?></td>

<td><?php echo $m['district']; ?></td>

<?php
if($m['status']=='active')
{
?>

<td><a onclick="return confirm('Are you sure want to Reject?');"
href="codes/mechanic_reject.php?action=reject&id=<?php echo $m['reg_id']; ?>" class="btn btn-
danger btn-sm">Reject</a></td>

<?php
}

else
{
?>

<td><a href="codes/mechanic_approve.php?id=<?php echo $m['reg_id']; ?>" class="btn btn-success
btn-sm">Approve</a>

<a onclick="return confirm('Are you sure want to Delete?');"
href="codes/mechanic_reject.php?action=delete&id=<?php echo $m['reg_id']; ?>" class="btn btn-
danger btn-sm">Delete</a>

```



```
</td>
```

```
<?php
```

```
}
```

```
?>
```

```
</tr>
```

```
<?php
```

```
}
```

```
?>
```

```
</tbody>
```

```
</table>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<?php
```

```
require_once("footer.php");
```

```
?>
```

payment.php

```
<?php

require_once("header.php");

require_once("sidebar.php");

require_once('../Connectionclass.php');

$obj = new Connectionclass();

$username=$_SESSION['username'];

?>

<div class="main-panel">

<div class="content">

<div class="outer-w3-agile mt-3">

<div class="card">

<div class="card-header">PAYMENT </div>

<div class="card-body">

<div class="position-center">

<form role="form" method="post" action="codes/payment_exe.php" >

<div class="form-group">

<label for="exampleInputEmail1">Card Number</label>

<input type="text" class="form-control" id="exampleInputEmail1" name="cardno" required=""
pattern="\d*" title="Please enter numbers only" maxlength="16" minlength="16">

</div>

<div class="form-group">

<label for="exampleInputPassword1">Expire Month</label>

<select size="1" name="expmonth" class="form-control">
```

```
<option value="January">Select</option>
<option value="January">January</option>
<option value="February">February</option>
<option value="March">March</option>
<option value="April">April</option>
<option value="May">May</option>
<option value="June">June</option>
<option value="July">July</option>
<option value="August">August</option>
<option value="September">September</option>
<option value="October">October</option>
<option value="November">November</option>
<option value="December">December</option>
</select>
</div>

<div class="form-group">
<label for="exampleInputPassword1">Expire Year</label>
<?php
$Startyear=date('Y');
$endYear=$Startyear+10;
$yearArray = range($Startyear,$endYear);
?>
<select name="expyear" class="form-control">
<option value="">Select Year</option>
```

```
<?php
foreach ($yearArray as $year) {
    $selected = ($year == $Startyear) ? " : ";
    echo '<option '.$selected.' value="'.$year.'">'.$year.'</option>';
}
?>
</select>

</div>

<div class="form-group">
<label for="exampleInputPassword1">Cvv</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="cvv" required=""
pattern="\d*" title="Please enter numbers only" maxlength="3" minlength="3">

</div>

<div class="form-group">
<label for="exampleInputPassword1">Amount</label>

<input type="text" class="form-control" id="exampleInputPassword1" name="amount" value="<?php
echo 100 ?>" readonly >

</div>

<div>

<button type="submit" class="btn btn-info">Pay Now</button>

</form>

</div>

</div>

</section>

</div>
```

```
</div>
```

```
<?php
```

```
require_once('footer.php');
```

```
?>
```

CHAPTER-10

APPENDIX

INSTA MECH



Figure 1 :Home page

Tomcy George	9875643217
Tomcy villa	
Kerala	Alappuzha
Edathuva	tomcy@gmail.com
Profile photo Choose File 30a8b3cc96_ 5b251f4.jpg	Carriage
.....	

REGISTER

©2024 Instamech Online Register. All Rights Reserved

Figure 2: Registration page

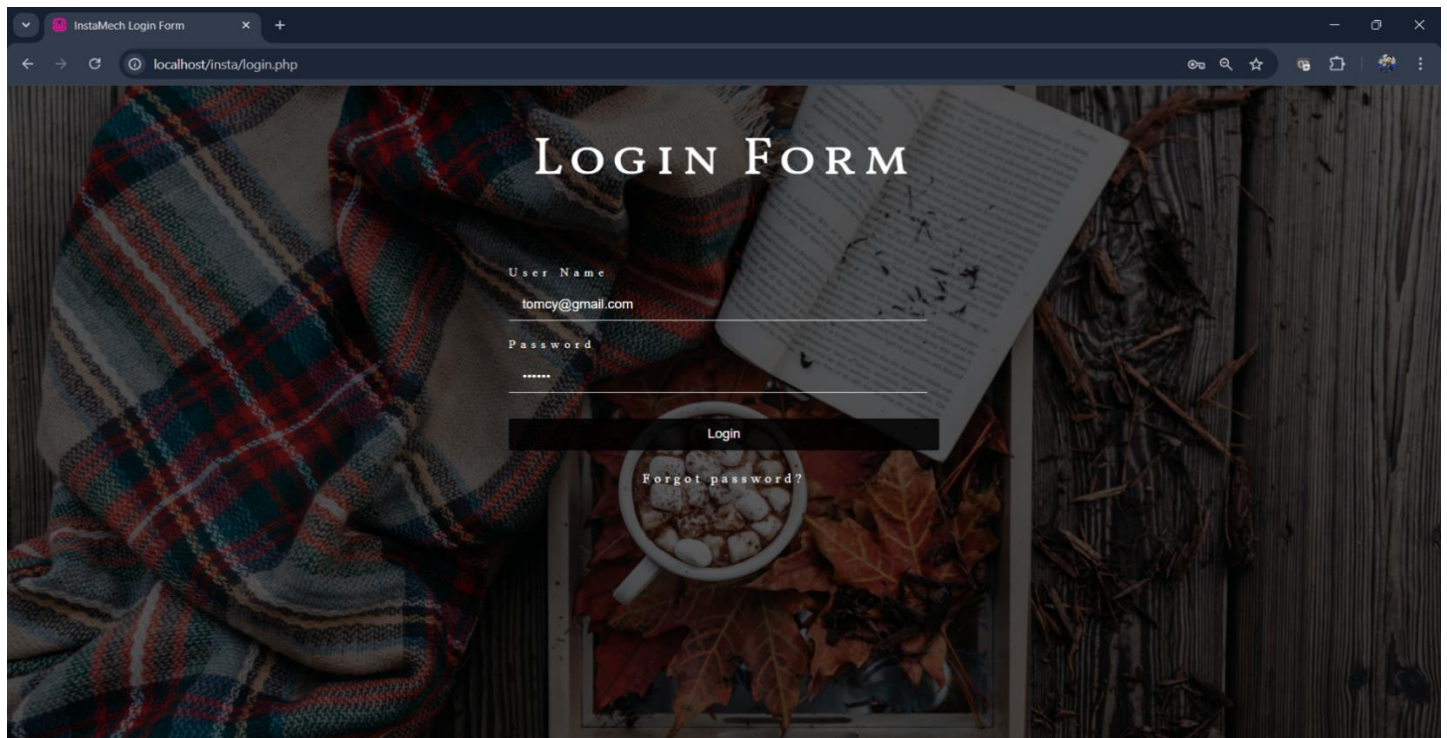


Figure 3: Login page

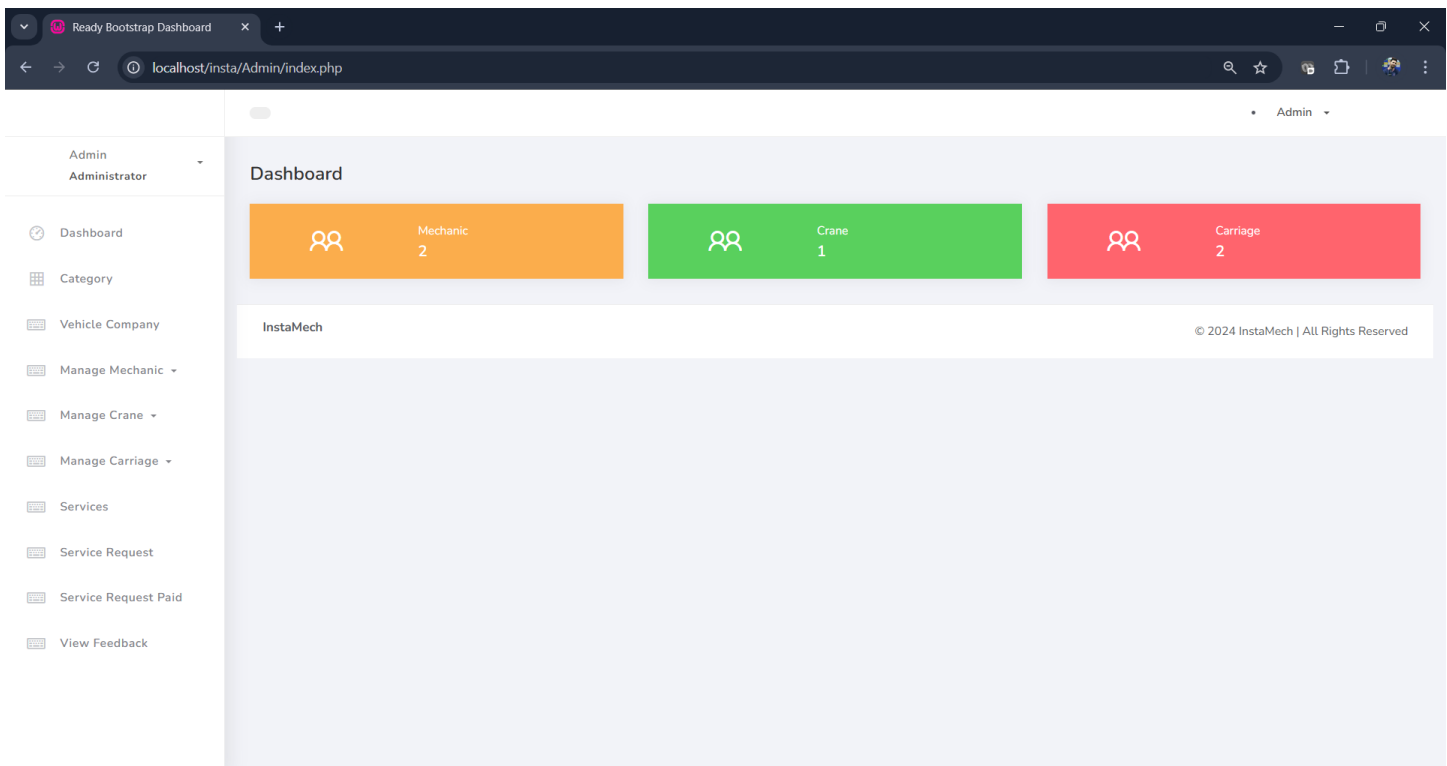


Figure 4: Admin Dashboard

The screenshot shows the 'User Dashboard' of the InstaMech application. The browser address bar indicates the URL is `localhost/insta/user/profile_view.php`. The left sidebar contains navigation links: Dashboard, Edit Profile, My Service Request, Search Mechanic, Search Crane or Carriage, and Feedback. The main content area displays the user's profile for 'Akhil CS'. It includes a profile picture with a 'Change Image' button and a table of user details. The details table lists the following information:

Email	akhilcs@gmail.com
Address	Akhil Villa
State	Kerala
District	Kottayam
City	Nadakapadam
Phone Number	8765432109

Below the table is an 'Edit Details' button. The footer of the page shows 'InstaMech' on the left and '© 2024 InstaMech | All Rights Reserved' on the right.

Figure 5: User Dashboard

The screenshot shows the 'Payment' page of the InstaMech application. The browser address bar indicates the URL is `localhost/insta/user/payment.php?s_id=88&rate=2000`. The left sidebar is identical to the previous screenshot. The main content area is titled 'Pay Now' and contains a form with the following fields:

- Card Number:
- Expire Month:
- Expire Year:
- Cvv:
- Amount:

A 'Pay Now' button is located at the bottom right of the form.

Figure 6: payment

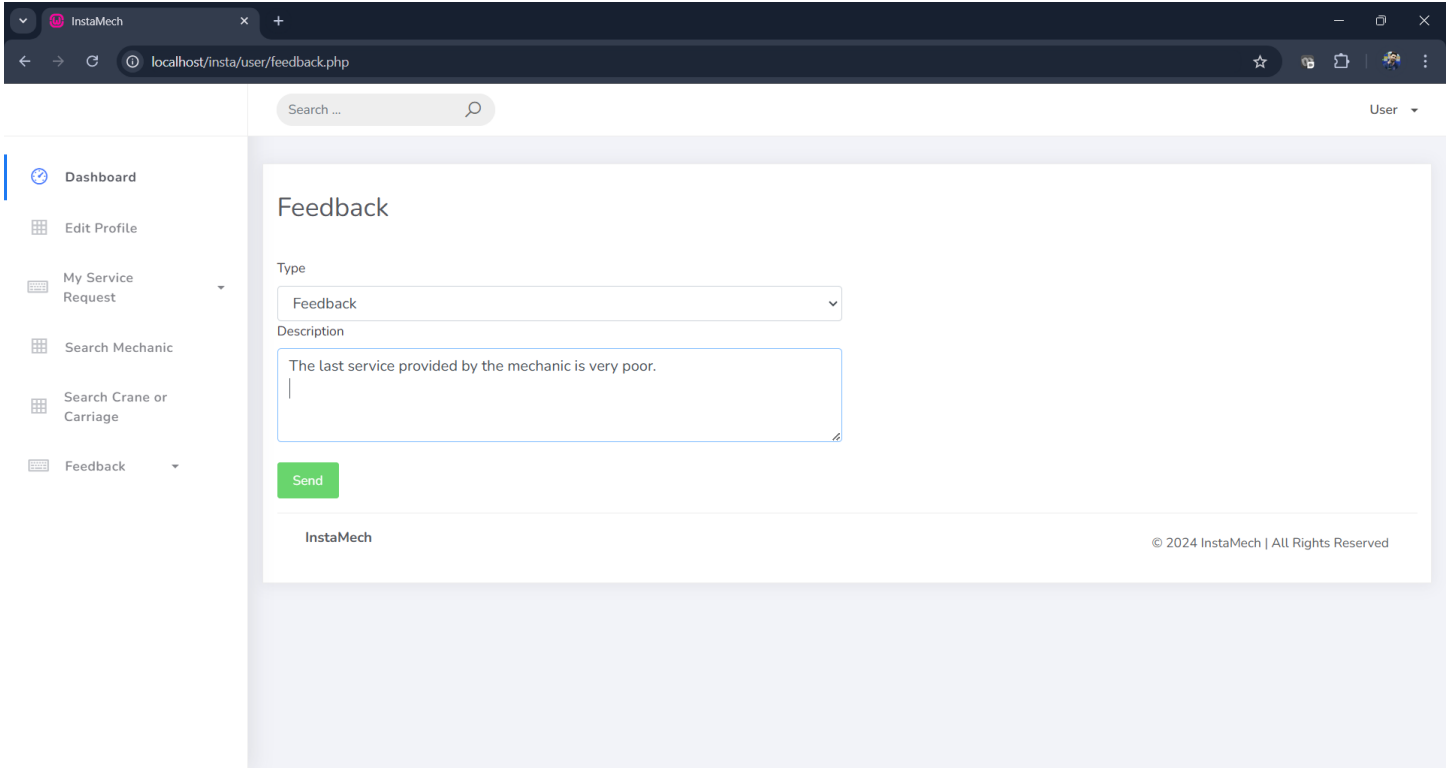


Figure 7: User Feedback

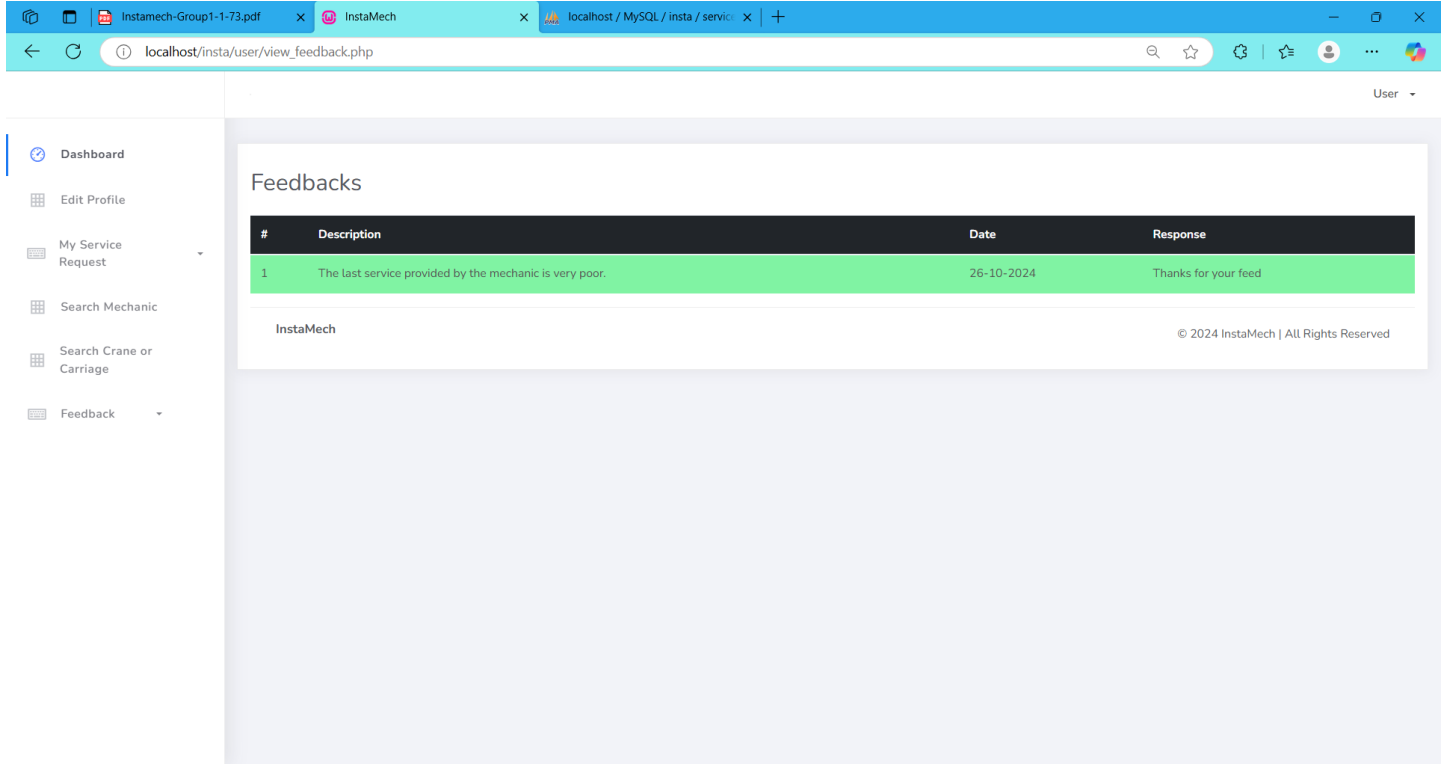


Figure 7.1: User Feedback