CHAPTER 1

INTRODUCTION

1.1. Purpose of this Project

This document is aimed at:

- Providing the necessary inputs to the detailed requirements gathering phase and further on for the SDLC processes.
- This document also serves to establish the traceability between the Business Objectives and the requirements identified in the proposed solution and how they satisfy the stated objectives.
- Provide expectation traceability in terms of the requirements and the user expectation
- Serves as a formal template for documenting the Business Requirements which also includes statutory and regulatory requirements.

The purpose of this document is to systematically capture the requirements of the project and the system to be developed. The document also

1.2. Overview

1.2.1. Objective

- Below are the objectives that shall be fulfilled post the execution of this project:
- User registration & credential authentication.
- Admin registration & credential authentication.
- Addition of services, it's price by the Admin.
- Addition of workers and setting their availability by the Admin.
- Viewing and Availing a service by User

1.3. Intended Audience

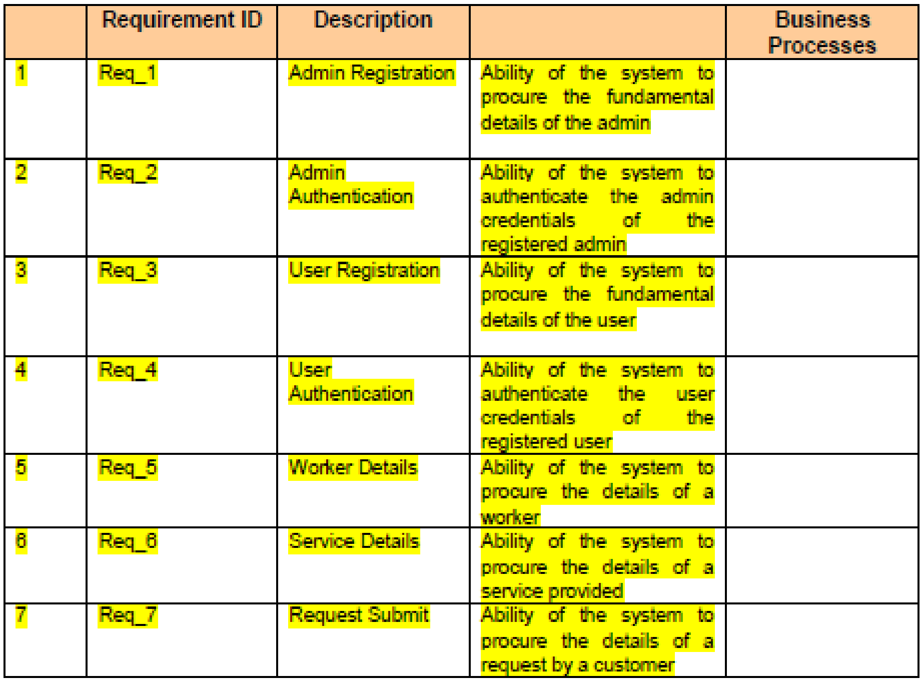
- Interns
- Mentors and SME's
- Delivery assurance/excellence group

1.4. Business Case

With increased working population globally, there have been an increased necessity to bridge their need of home services like plumbing, electricity, minor civil tasks – which are typically an unorganized sector. A one-stop portal for all domestic services is the need of the public wherein a user is able to perform a location-based search, by service, by price quoted by the service provider.

The solution developed will address the objective in a holistic manner and will have all the features and functionalities which shall let the customer to perform a location-based search, browse by the nature of service offered, contact info and sort by the price quoted. The portal will also provide admin features.

1.5. High Level Business Requirement



	Requirement ID	Description		Business Processes
1	Req_1	Admin Registration	Ability of the system to procure the fundamental details of the admin	
2	Req_2	Admin Authentication	Ability of the system to authenticate the admin credentials of the registered admin	
3	Req_3	User Registration	Ability of the system to procure the fundamental details of the user	
4	Req_4	User Authentication	Ability of the system to authenticate the user credentials of the registered user	
5	Req_5	Worker Details	Ability of the system to procure the details of a worker	
6	Req_6	Service Details	Ability of the system to procure the details of a service provided	
7	Req_7	Request Submit	Ability of the system to procure the details of a request by a customer	

Table 1.5 High Level Business Requirement

CHAPTER 2

THE PROBLEM STATEMENT

The “Home Service” is an independent stand–alone product. Its’ idea originated to provide a common platform for workers as well as users. With the use of this platform user can request a work from the admin and admin can allot a worker to that user on the basis of their requirement. It is a new and self-contained product. It is a Web based Application. This application provides simple mechanism for the Users to provide best possible worker near to their location.

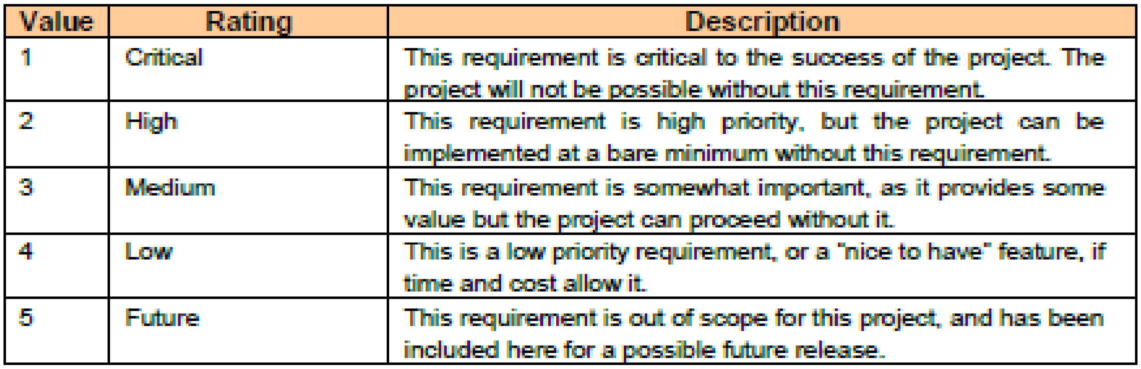
2.1. Assumption and Dependencies

- User must be trained for basic computer functionalities.
- User must have the basic knowledge of English
- The system must be able to respond to database software within reasonable time.
- Front-end (user interaction): - The product will require a mobile with an application program or with any other application program and a communication channel.
- The admin and user must have a laptop to enter the details and the interface must be user friendly.
- The speed of the communication channel (if any) must be, at a minimum 28.8 kbps to support message transfer in reasonable time.

2.2. Functional Requirement

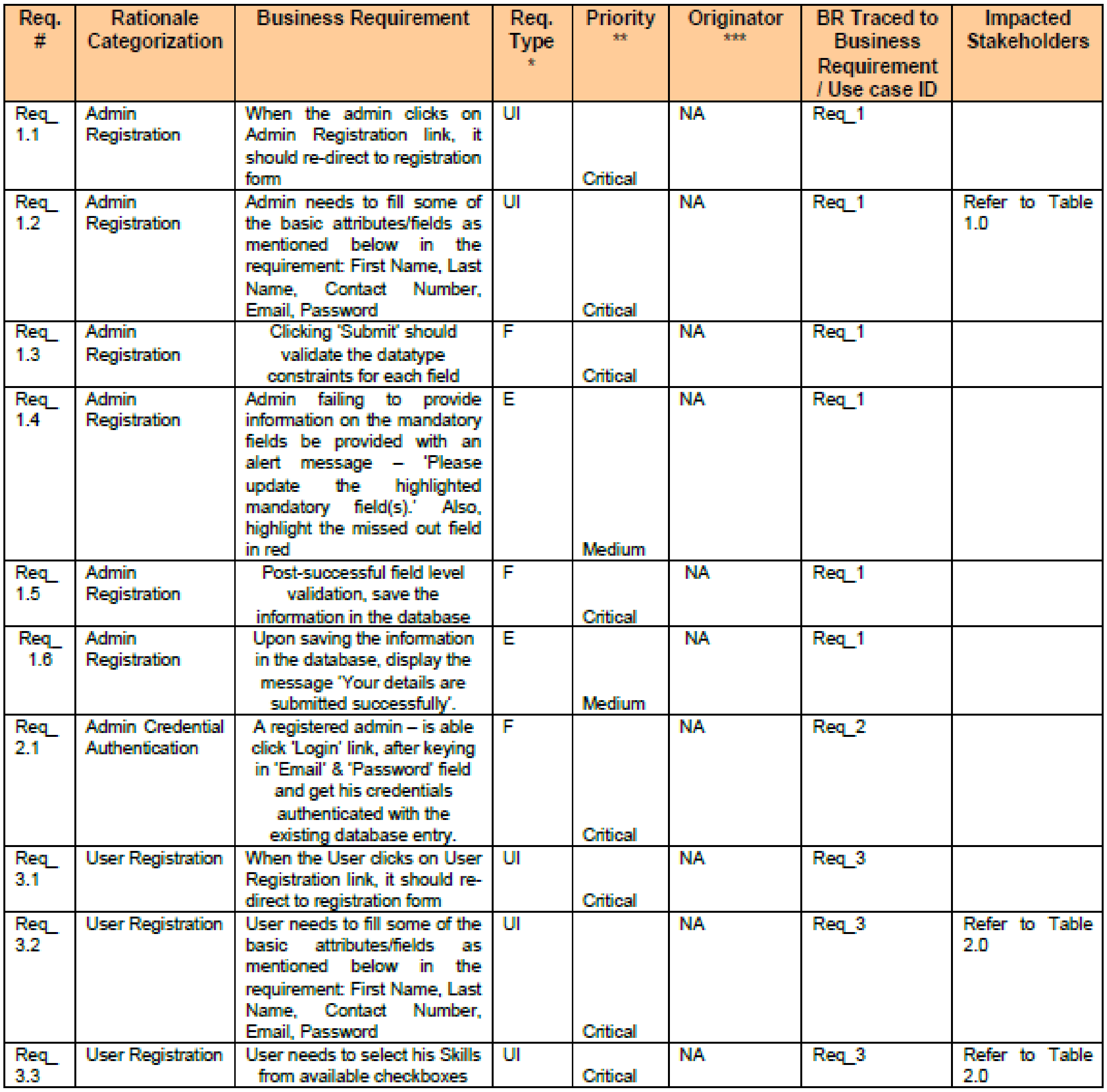
The functional requirements are charted for each of the high-level requirements called out in the earlier section:
Additionally, the following elements are captured for each business requirement in the table provided below: -

- Req. Type = (F Core Functionality, E Exception, UI User Interface, R Reporting)
- Priority of Requirement = (1=Base Functionality, 2=Advanced Functionality,
    - 3=Additional Opportunities)
- Originator = (Name of the business process of the system/ department or function name in the customer organization)



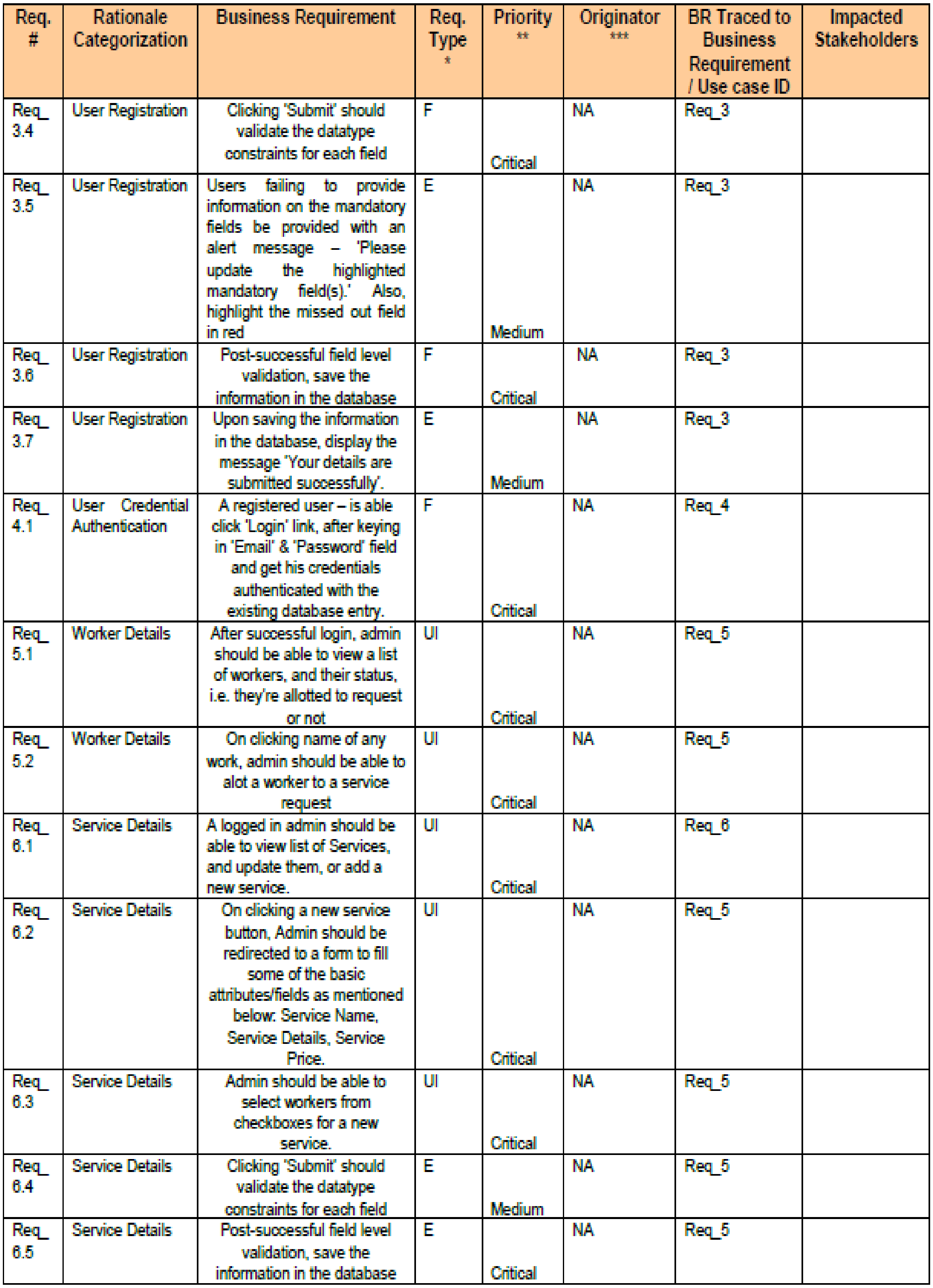
Value	Rating	Description
1	Critical	This requirement is critical to the success of the project. The project will not be possible without this requirement.
2	High	This requirement is high priority, but the project can be implemented at a bare minimum without this requirement.
3	Medium	This requirement is somewhat important, as it provides some value but the project can proceed without it.
4	Low	This is a low priority requirement, or a "nice to have" feature, if time and cost allow it.
5	Future	This requirement is out of scope for this project, and has been included here for a possible future release.

Table 2.2.1



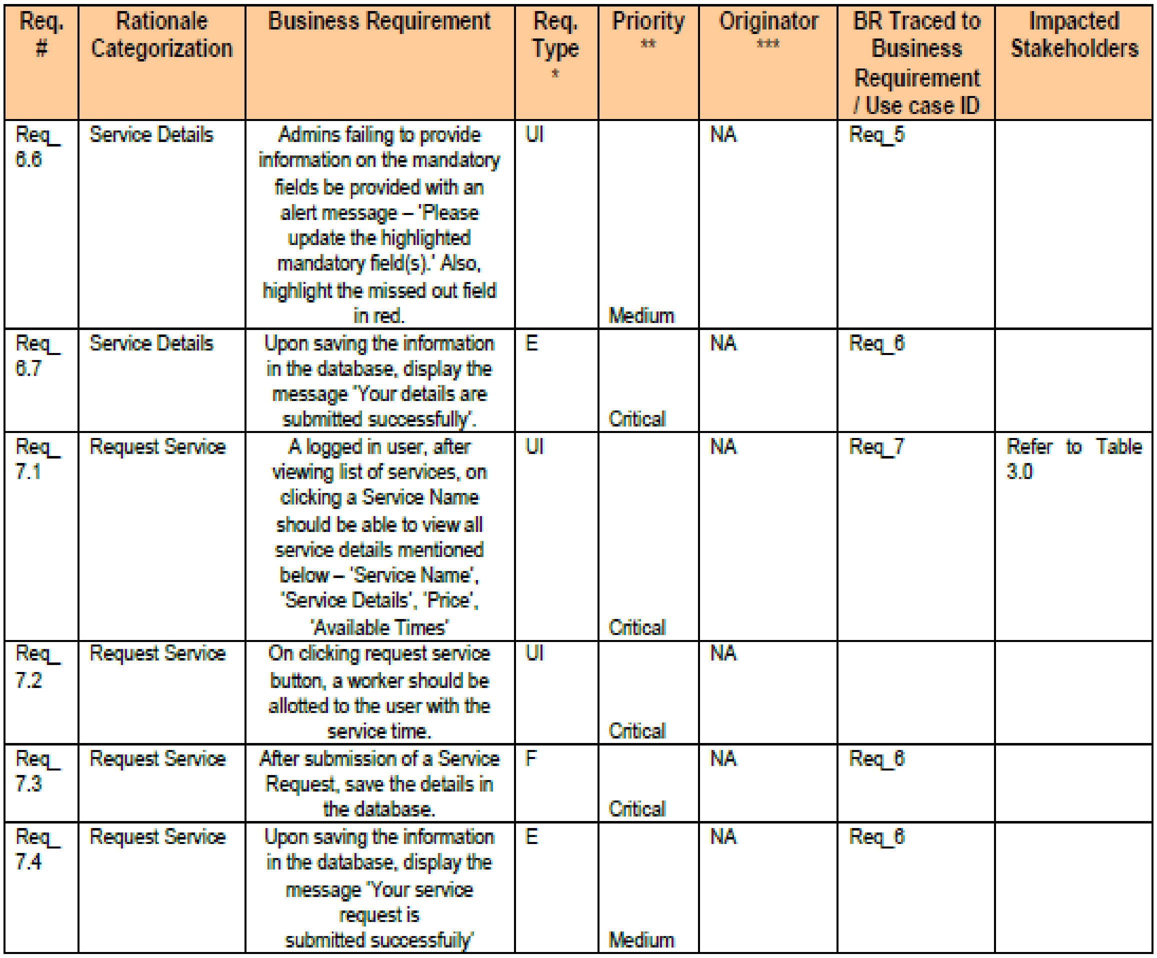
Req. #	Rationale Categorization	Business Requirement	Req. Type *	Priority **	Originator ***	BR Traced to Business Requirement / Use case ID	Impacted Stakeholders
Req_ 1.1	Admin Registration	When the admin clicks on Admin Registration link, it should re-direct to registration form	UI	Critical	NA	Req_1	
Req_ 1.2	Admin Registration	Admin needs to fill some of the basic attributes/fields as mentioned below in the requirement: First Name, Last Name, Contact Number, Email, Password	UI	Critical	NA	Req_1	Refer to Table 1.0
Req_ 1.3	Admin Registration	Clicking 'Submit' should validate the datatype constraints for each field	F	Critical	NA	Req_1	
Req_ 1.4	Admin Registration	Admin failing to provide information on the mandatory fields be provided with an alert message – 'Please update the highlighted mandatory field(s).' Also, highlight the missed out field in red	E	Medium	NA	Req_1	
Req_ 1.5	Admin Registration	Post-successful field level validation, save the information in the database	F	Critical	NA	Req_1	
Req_ 1.6	Admin Registration	Upon saving the information in the database, display the message 'Your details are submitted successfully'.	E	Medium	NA	Req_1	
Req_ 2.1	Admin Credential Authentication	A registered admin – is able click 'Login' link, after keying in 'Email' & 'Password' field and get his credentials authenticated with the existing database entry.	F	Critical	NA	Req_2	
Req_ 3.1	User Registration	When the User clicks on User Registration link, it should re-direct to registration form	UI	Critical	NA	Req_3	
Req_ 3.2	User Registration	User needs to fill some of the basic attributes/fields as mentioned below in the requirement: First Name, Last Name, Contact Number, Email, Password	UI	Critical	NA	Req_3	Refer to Table 2.0
Req_ 3.3	User Registration	User needs to select his Skills from available checkboxes	UI	Critical	NA	Req_3	Refer to Table 2.0

Table 2.2.2



Req. #	Rationale Categorization	Business Requirement	Req. Type *	Priority **	Originator ***	BR Traced to Business Requirement / Use case ID	Impacted Stakeholders
Req_ 3.4	User Registration	Clicking 'Submit' should validate the datatype constraints for each field	F	Critical	NA	Req_3	
Req_ 3.5	User Registration	Users failing to provide information on the mandatory fields be provided with an alert message – 'Please update the highlighted mandatory field(s).' Also, highlight the missed out field in red	E	Medium	NA	Req_3	
Req_ 3.6	User Registration	Post-successful field level validation, save the information in the database	F	Critical	NA	Req_3	
Req_ 3.7	User Registration	Upon saving the information in the database, display the message 'Your details are submitted successfully'.	E	Medium	NA	Req_3	
Req_ 4.1	User Credential Authentication	A registered user – is able click 'Login' link, after keying in 'Email' & 'Password' field and get his credentials authenticated with the existing database entry.	F	Critical	NA	Req_4	
Req_ 5.1	Worker Details	After successful login, admin should be able to view a list of workers, and their status, i.e. they're allotted to request or not	UI	Critical	NA	Req_5	
Req_ 5.2	Worker Details	On clicking name of any work, admin should be able to alot a worker to a service request	UI	Critical	NA	Req_5	
Req_ 6.1	Service Details	A logged in admin should be able to view list of Services, and update them, or add a new service.	UI	Critical	NA	Req_6	
Req_ 6.2	Service Details	On clicking a new service button, Admin should be redirected to a form to fill some of the basic attributes/fields as mentioned below: Service Name, Service Details, Service Price.	UI	Critical	NA	Req_5	
Req_ 6.3	Service Details	Admin should be able to select workers from checkboxes for a new service.	UI	Critical	NA	Req_5	
Req_ 6.4	Service Details	Clicking 'Submit' should validate the datatype constraints for each field	E	Medium	NA	Req_5	
Req_ 6.5	Service Details	Post-successful field level validation, save the information in the database	E	Critical	NA	Req_5	

Table 2.2.3



Req. #	Rationale Categorization	Business Requirement	Req. Type *	Priority **	Originator ***	BR Traced to Business Requirement / Use case ID	Impacted Stakeholders
Req_ 6.6	Service Details	Admins failing to provide information on the mandatory fields be provided with an alert message – 'Please update the highlighted mandatory field(s).' Also, highlight the missed out field in red.	UI	Medium	NA	Req_5	
Req_ 6.7	Service Details	Upon saving the information in the database, display the message 'Your details are submitted successfully'.	E	Critical	NA	Req_6	
Req_ 7.1	Request Service	A logged in user, after viewing list of services, on clicking a Service Name should be able to view all service details mentioned below – 'Service Name', 'Service Details', 'Price', 'Available Times'	UI	Critical	NA	Req_7	Refer to Table 3.0
Req_ 7.2	Request Service	On clicking request service button, a worker should be allotted to the user with the service time.	UI	Critical	NA		
Req_ 7.3	Request Service	After submission of a Service Request, save the details in the database.	F	Critical	NA	Req_6	
Req_ 7.4	Request Service	Upon saving the information in the database, display the message 'Your service request is submitted successfully'	E	Medium	NA	Req_6	

Table 2.2.4

2.3. Non-Functional Requirement

- 24*7 availability
- Reliability
- Maintenance
- Portability
- Security

CHAPTER 3

DFD

3.1. DFD

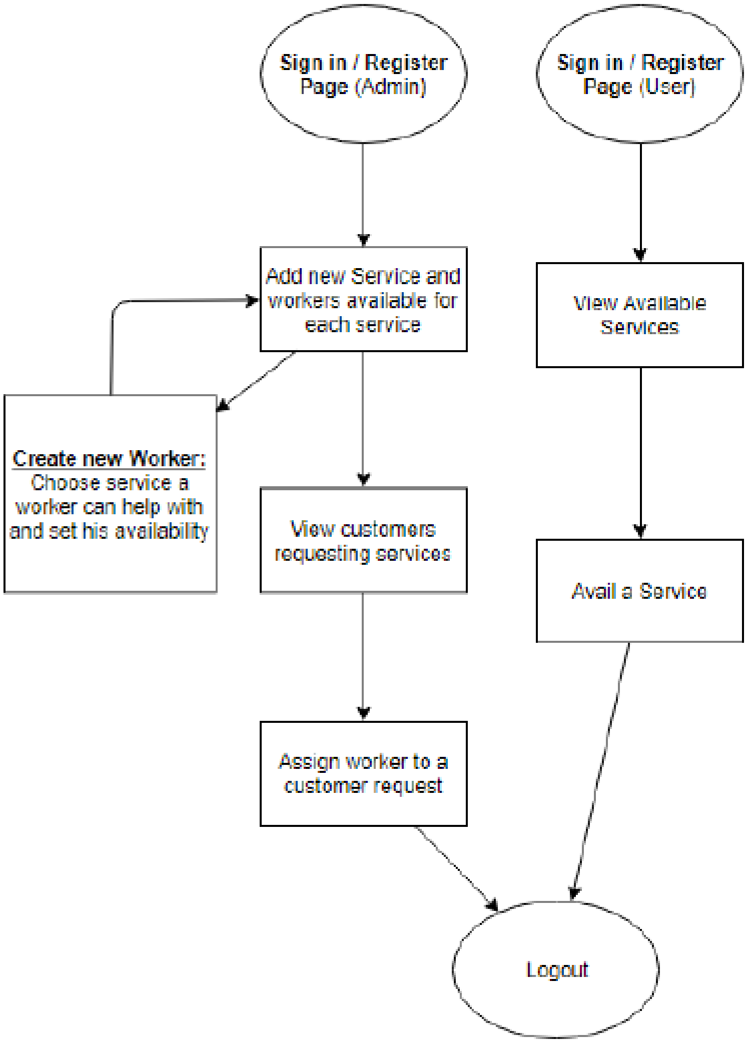

Sign in / Register Page (Admin)
Sign in / Register Page (User)
Add new Service and workers available for each service
View Available Services
Create new Worker: Choose service a worker can help with and set his availability
View customers requesting services
Avail a Service
Assign worker to a customer request
Logout

Figure 3.1(Flow Chart)

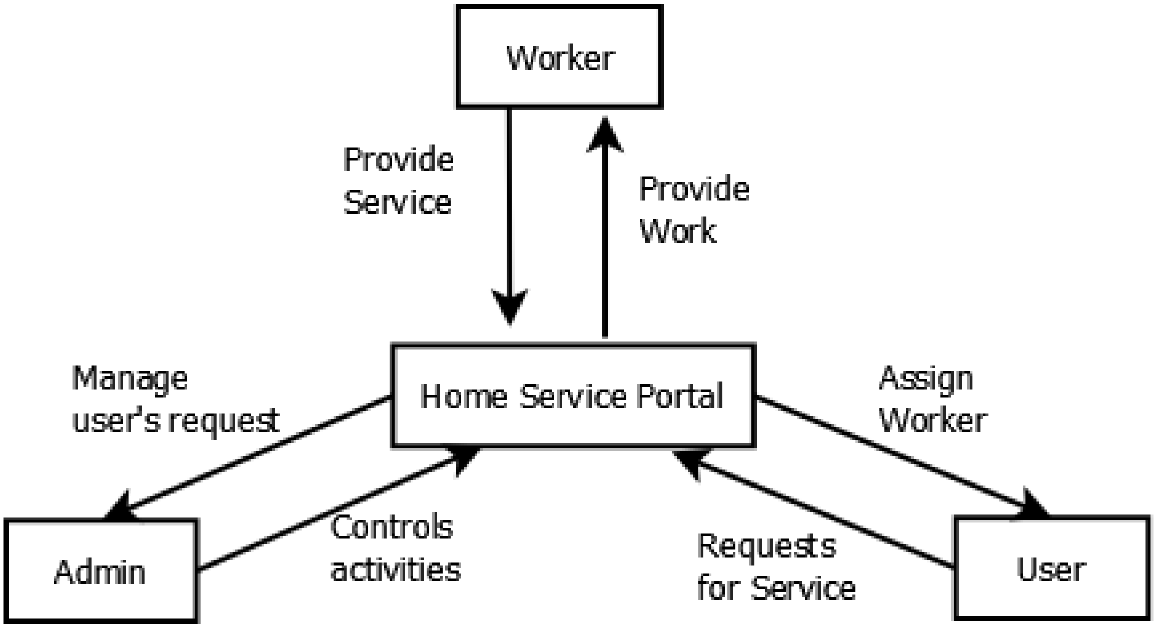


Figure 3.2(Level 0 DFD)

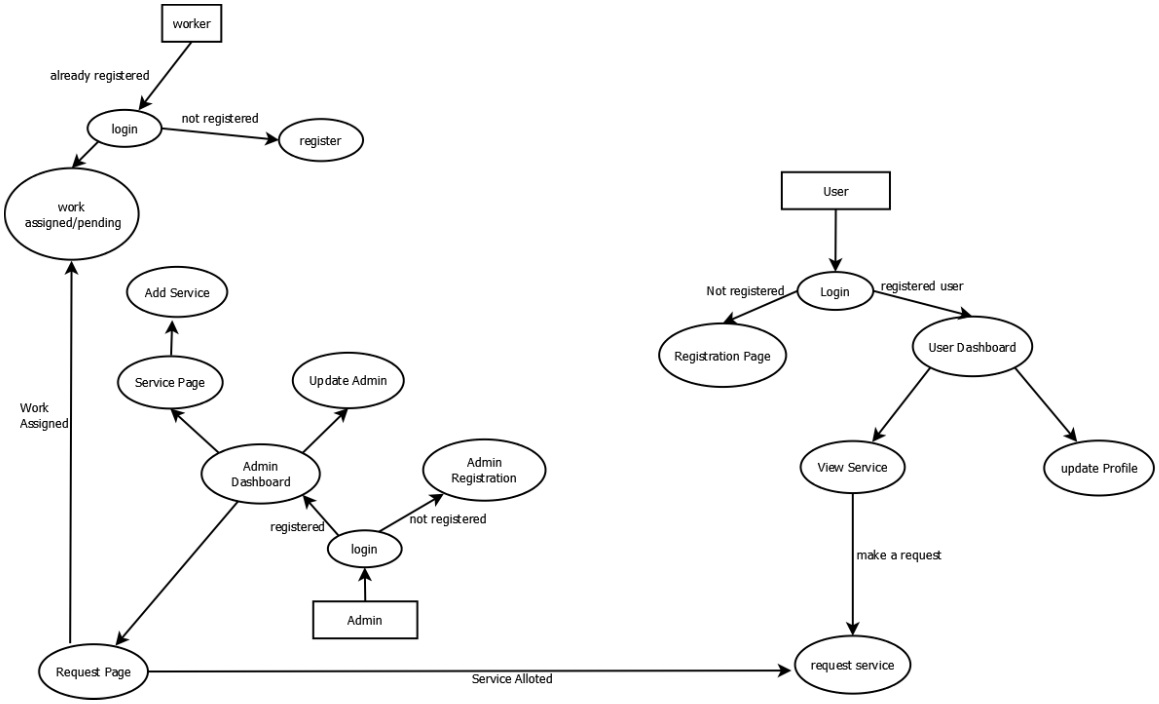
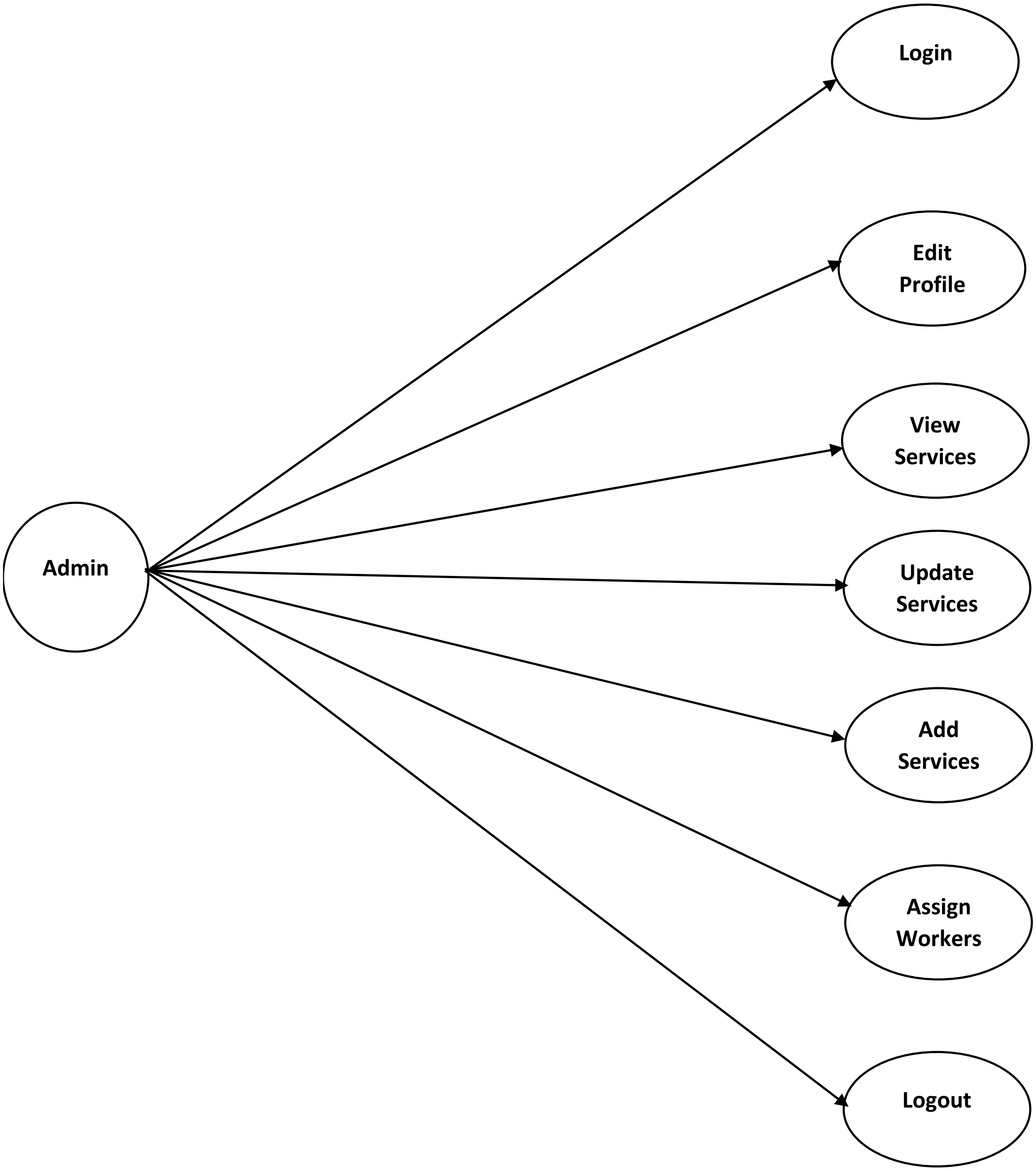


Figure 3.3(Level 1 DFD)

3.2. Use Case Diagram

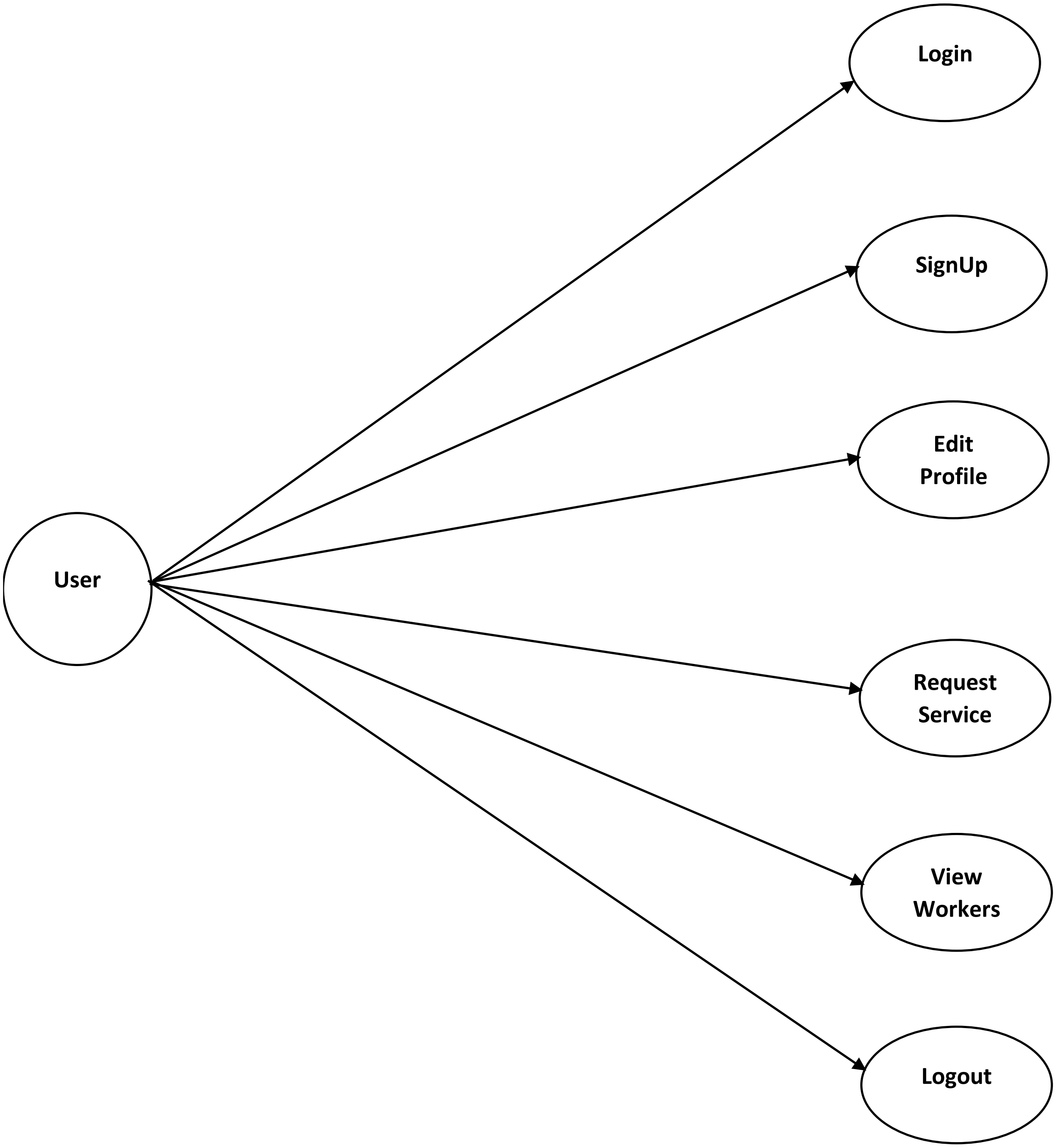
USE CASE

(admin)



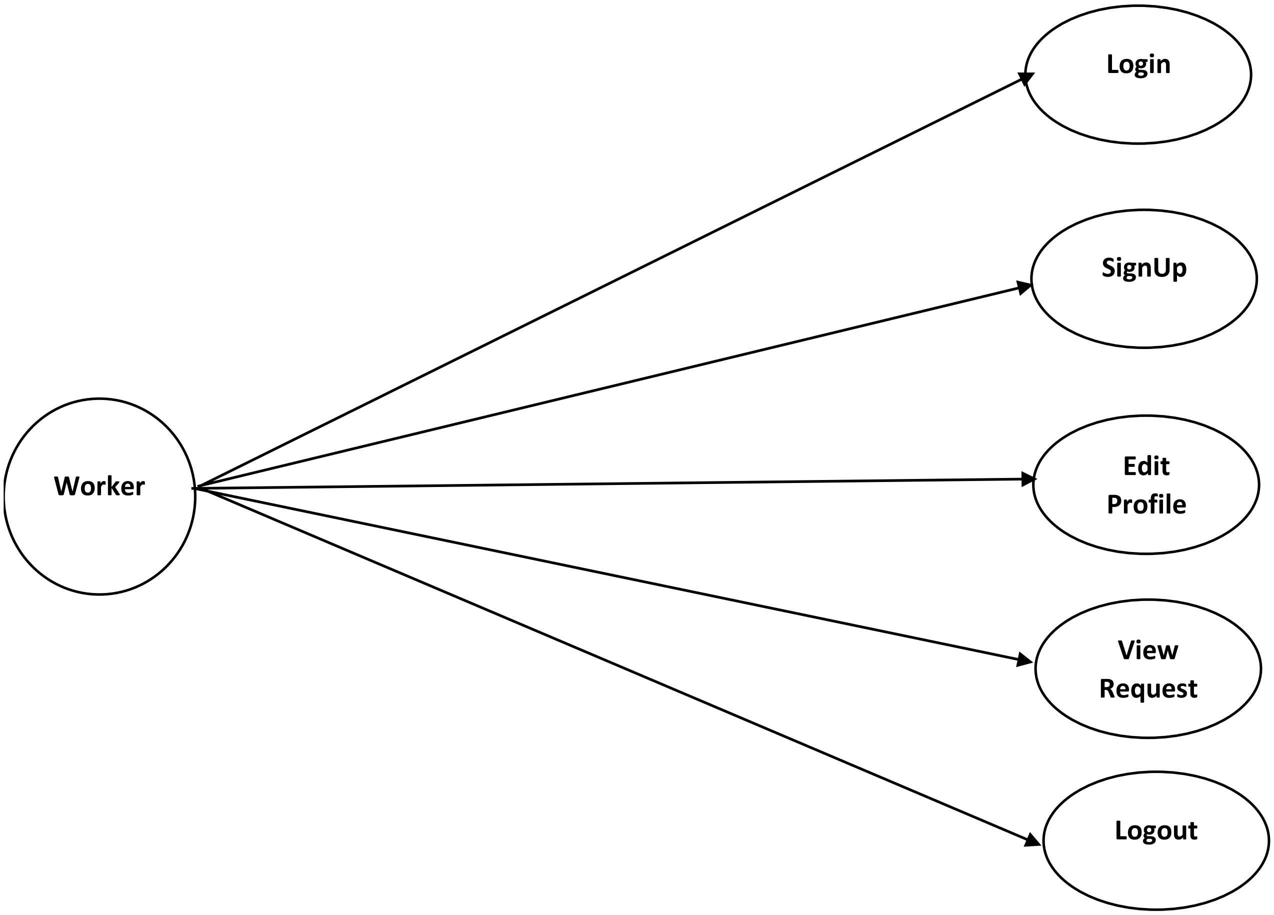
USE CASE

(User)



USE CASE

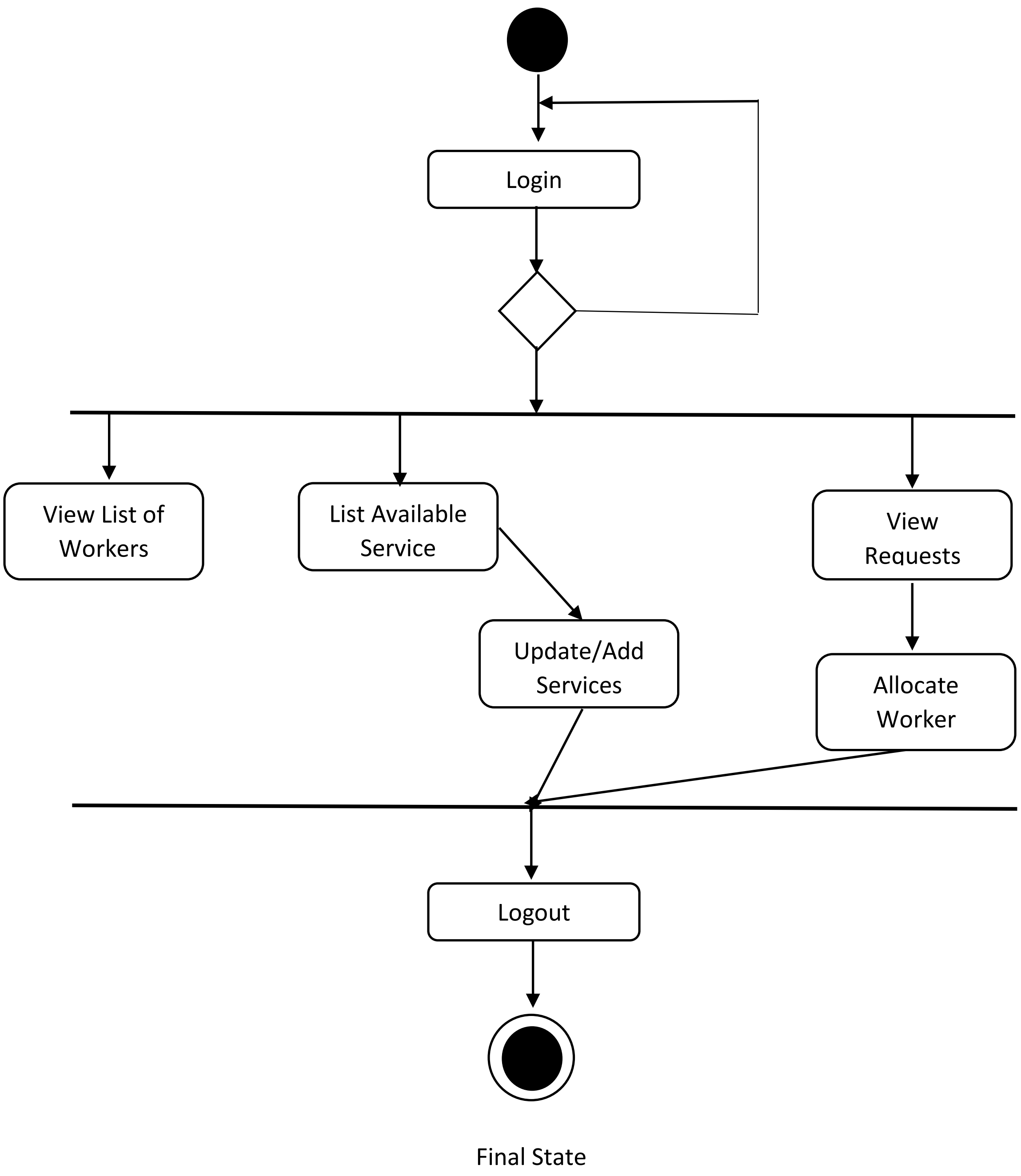
(Worker)



3.3. Activity Diagram

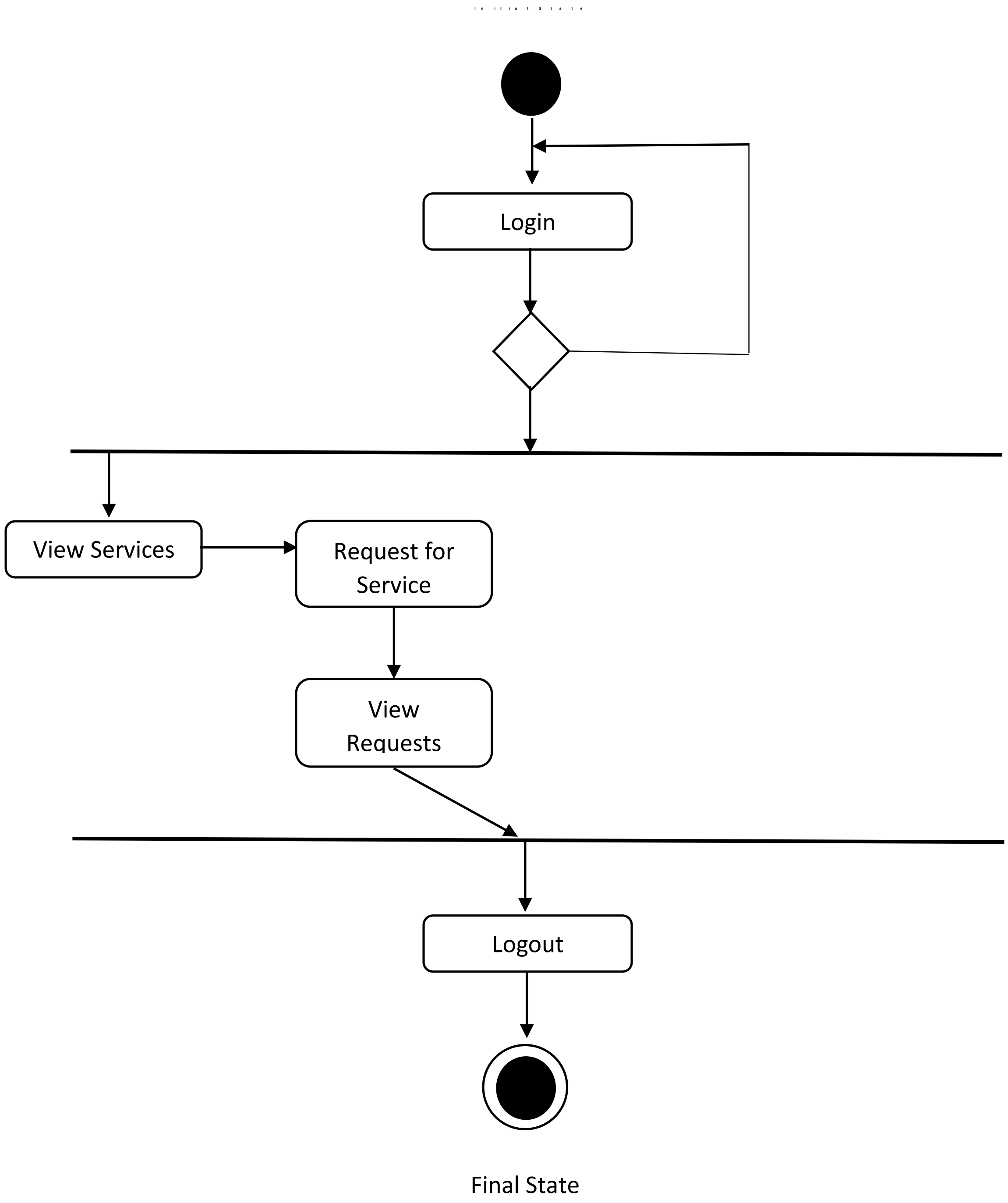
Activity Diagram

(Admin)



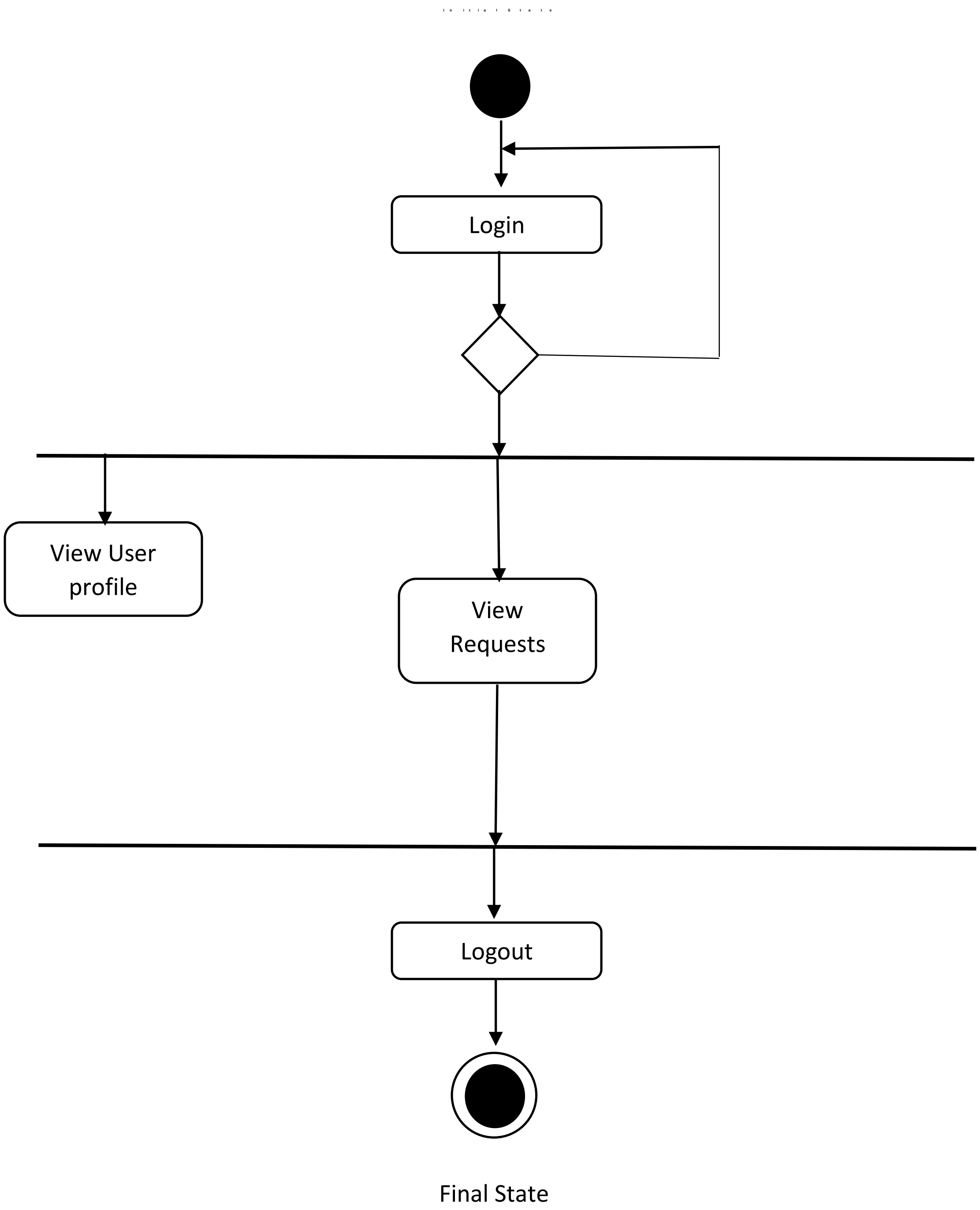
Activity Diagram

(User)



Activity Diagram

(worker)



CHAPTER 4

PROBLEM ANALYSIS

4.1. Product Definition

Our project “Home Service” is merely based upon different technologies like Spring, Hibernate, MySQL, HTML5, JavaScript, Bootstrap, Google Map API and so on. Web Development is very useful technology for any product right now, and often seem to be used interchangeable. Perception can lead to confusion sometime.

The “Home Service” is an independent stand–alone product. Its idea originated to provide a common platform for workers as well as users. With the use of this platform user can request a work from the admin and admin can allot a worker to that user on the basis of their requirement. It is a new and self-contained product. It is a Web based Application. This application provides simple mechanism for the Users to provide best possible worker near to their location.

4.2.Feasibility Analysis

A feasibility study assesses the possibilities of development of the product. The feasibility study is intended to be the preliminary review of the facts to see if it is worthy of proceeding to the analysis phase. From the perspective of analyst, the feasibility analysis is a primary tool for evaluating whether to proceed to the next level or to discontinue the product. The key consideration in feasibility analysis are:

- Technical feasibility
- Behavioral feasibility
- Operational feasibility
- Economic feasibility

4.2.1. Technical Feasibility

The technical needs of the system might include:

- Facility to provide data in given time.
- Response under certain time.
- Ability of user to generate request of the work.
- We have used reliable databases MYSQL.
- Ability of Admin to Assign a worker to user request of the work.
- Proper use of internet connection required to access the location of the workers using Google Map API.
- We studied that under which circumstances our project will give output and under which circumstances it does not.

- We studied the language and tools used to develop the project are reliable or not.
- We studied whether our product will be useful for the intended audience in any way or not.
- We ensure that our system security is genuine and powerful or not.

4.2.2. Behavioral Feasibility

As it is easy to access different features in different modules, user will act positively to this project. It will be easily acceptable by the user as all the modules in our system provides a GUI environment that make it user friendly.

4.2.3. Operational Feasibility

Proposed projects are beneficial only if they can be turned out into information system i.e. it should benefit the concerned mass of people who tends to use it. That will meet the organization’s operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

- Is there enough support for the management from the users?
- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible application benefits?
- This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So, there is no question of resistance from the users that can undermine the possible application benefits. The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

4.3. Project Plan

A project plan is produced as management activities commence. The plan defines the process and tasks to be conducted, the people who will do the work, and the mechanisms for assessing risks, controlling change, and evaluating quality. To develop this project, we follow Evolution Model. In which our requirements can change during development only.

4.4. Development Schedule

We set our schedule according to hours. As we spend daily few hours on project only.
We got less than 40 days to complete it. Our schedule is as follow:

- Problem Identification --- 3Days

- Feasibility Study -- 3Days
- Requirement Analysis -- 2Days
- Design and Implementation ---30Days
- Testing --- 5Days

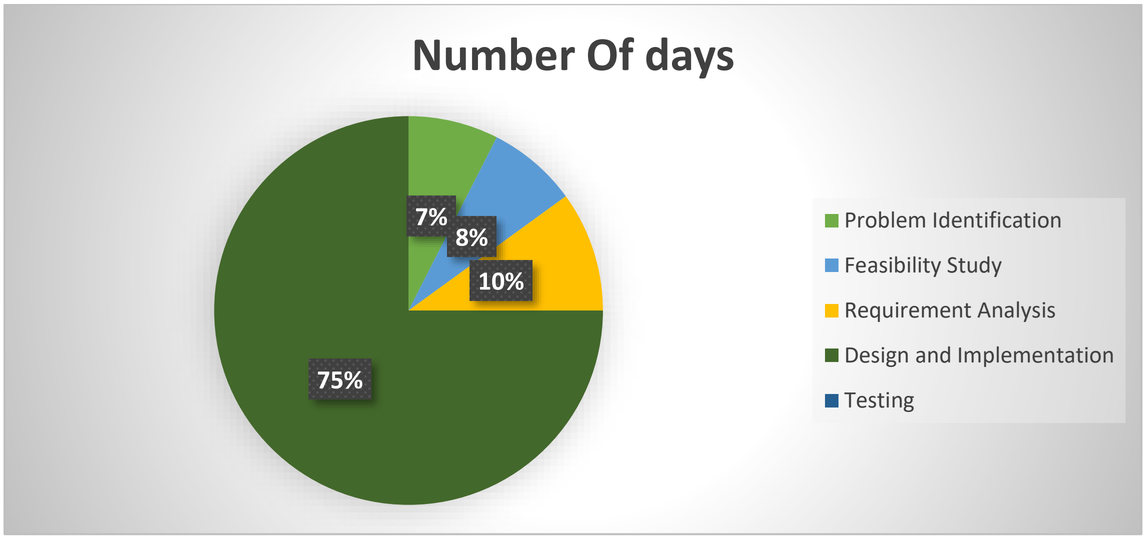


Figure 4.4.1

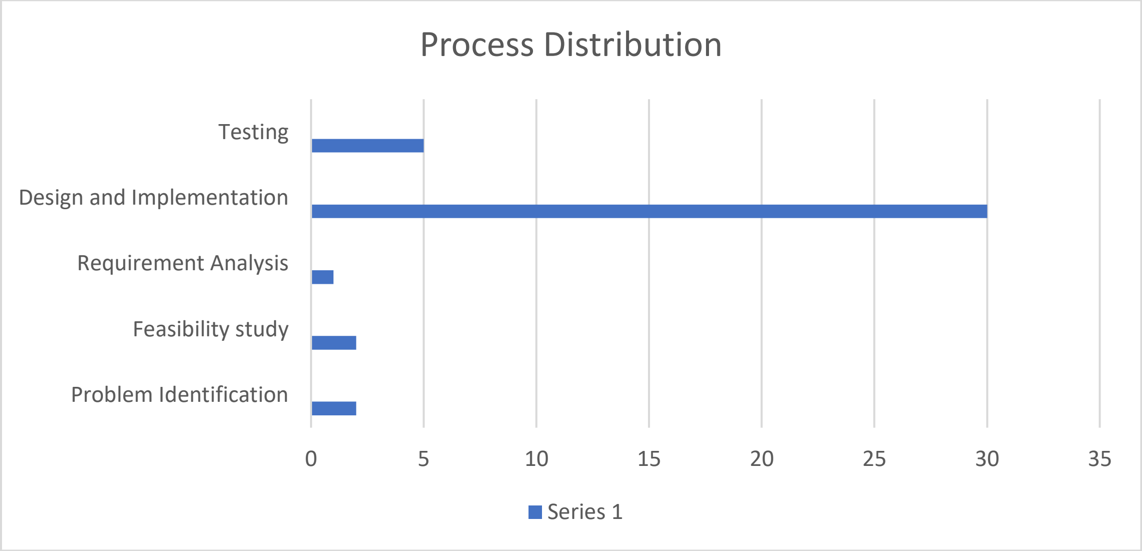
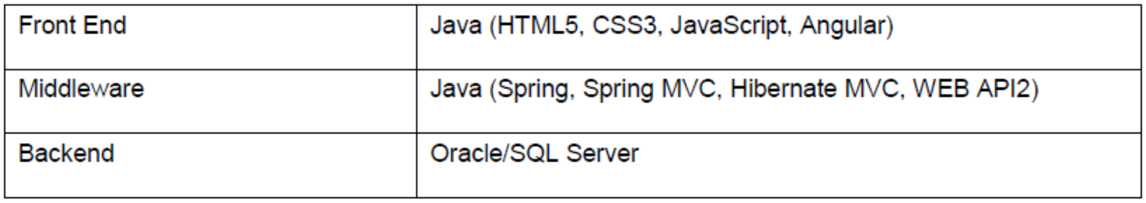


Figure 4.4.2

CHAPTER 5

SOFTWARE REQUIREMENT

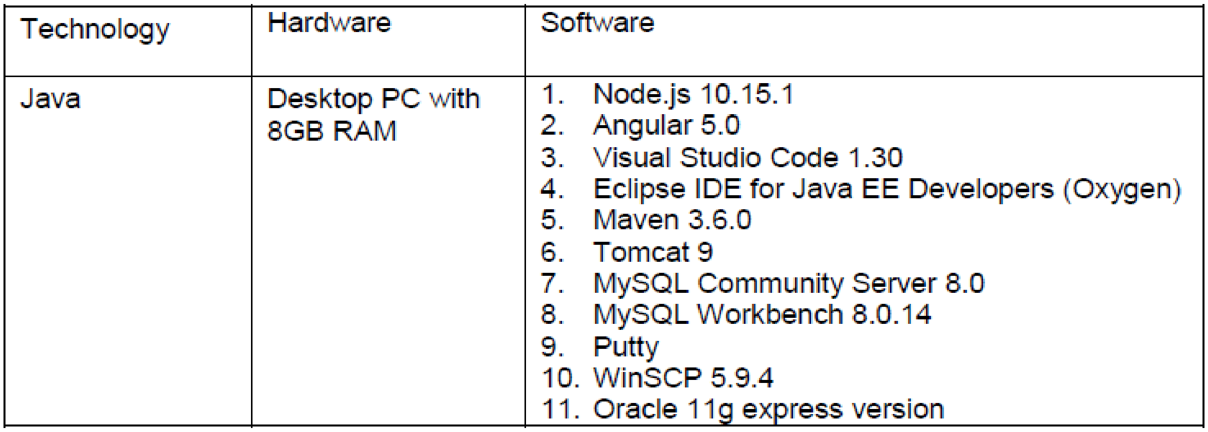
5.1. Technologies Recommended



Front End	Java (HTML5, CSS3, JavaScript, Angular)
Middleware	Java (Spring, Spring MVC, Hibernate MVC, WEB API2)
Backend	Oracle/SQL Server

Table 5.1.1

5.2. Hardware and software Requirement



Technology	Hardware	Software
Java	Desktop PC with 8GB RAM	1. Node.js 10.15.1 2. Angular 5.0 3. Visual Studio Code 1.30 4. Eclipse IDE for Java EE Developers (Oxygen) 5. Maven 3.6.0 6. Tomcat 9 7. MySQL Community Server 8.0 8. MySQL Workbench 8.0.14 9. Putty 10. WinSCP 5.9.4 11. Oracle 11g express version

Table 5.2.1

CHAPTER 6

DESIGN

Design is defined as a multistage process that primarily focuses on four distinct features of a program:

- Data structure
- Software architecture
- Interface representation
- Procedural detail.

The design process translates the requirements into a representation of the software that can be accessed for quality product before code generation begins. It is the most creative and challenging stage of the SDLC (Software Development Life Cycle). It refers to the technical specifications that will be applied to the system.

6.1. System Design

Design goals basically describe the qualities of the system that developers should keep in mind. Such goals are normally derived from the non-functional requirements of the system. Design goals are grouped into five categories. These are

- Performance
- Dependability
- Maintenance
- End user criteria
- Usability

- **Performance**

  The system should have a fast response time (real time) with maximum throughput. The admin has chosen fast response time over throughput and hence the system should try to be more interactive. The player should be able to move smoothly on the terrain and shoot. The game should work smoothly.

- **Dependability**

  The users need the system to be highly dependable as it is expected to be used by end users who have better knowledge about games. The system should run the game smoothly and there should not be any dead end.

- **Maintenance**

  The system should be easy to manage and easily modifiable to make changes to the features and functionalities of the system.

- **End User Criteria**

  End User criteria means to define a set of methods, techniques and tools that help the nonprofessional software developers to create, modify or extend a software artefact.

- **Usability**

  Usability is the extent to which a product can be used by specified users to achieve goals with effectiveness, reliability, efficiency and satisfaction in a specified context of use. From the end user's perspective, the system should be designed in such a way that it is easy to learn and use, efficient and having few errors and bugs if any.

CHAPTER 7

TESTING

Software testing is a critical section of software quality assurance and represents the ultimate review of specification, design and coding. Testing presents an interesting view of a system using various test data or cases. Preparation of the test data plays an important role in the system testing. After preparing the test data, the system under study is tested those test data. Errors and bugs are found and corrected by using the following testing steps and corrections are recorded for future references. Thus, series of testing is performed on the system before it is ready for implementation

7.1. Functional Testing

Functional testing is a type of black box testing that bases its test cases on the specifications of the software component under test. Functions are tested by feeding them input and examining the output, and internal program structure is rarely considered. Functional testing differs from system testing in a way that functional testing "verifies a program by checking it against design document or specification", while system testing "validates a program by checking it against the published user or system requirements". Functional testing typically involves five steps:

- The identification of functions that the software is expected to perform.
- The creation of input data based on the function's specifications.
- The determination of output based on the function's specifications.
- The execution of the test case.
- The comparison of actual and expected outputs.

7.2.Black Box Testing

Black box testing is done by the professional testing team. This does not require knowledge of internal knowledge of the application. Testing the application against the functionality of the application without the knowledge of internal coding of the software. In Black box testing the structure of the program is not at all considered. Test cases are decided solely based on the requirements or specification of the program or module. Black box testing mainly applicable to higher levels of testing: Acceptance 24 Testing and System Testing. Implementation knowledge is not required for black box testing.

7.3.Levels of Testing

Software testing is a critical section of software quality assurance and represents the ultimate review of specification, design and coding. Testing presents an interesting view of a system using various test data or cases. Preparation of the test data plays an important role in the system testing. After preparing the test data, the system under study is tested those test data. Errors and bugs are found and corrected by using the following testing steps and corrections are recorded for future references. Thus, series of testing is performed on the system before it is ready for implementation.

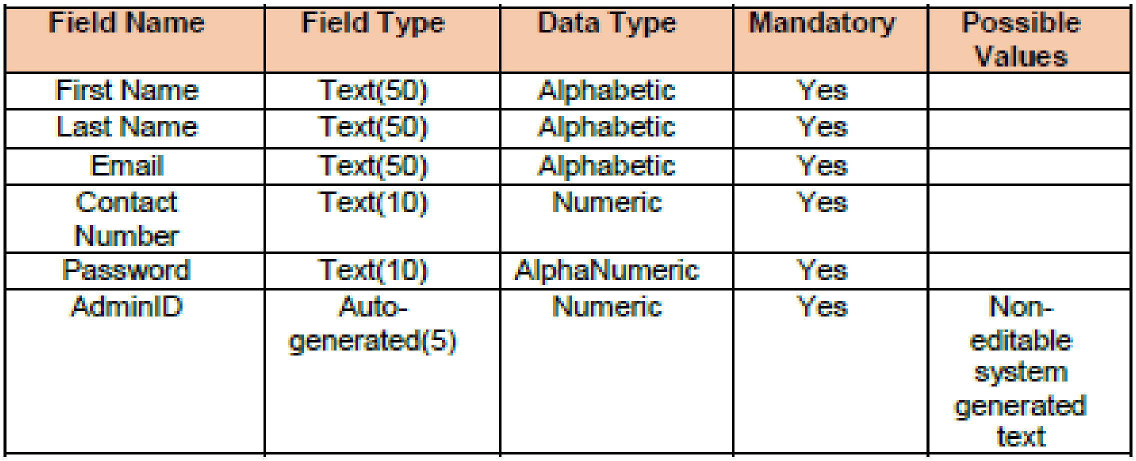
The various types of testing on the system are:

- Unit testing
- Integration testing
- System testing
- User acceptance testing

CHAPTER 8

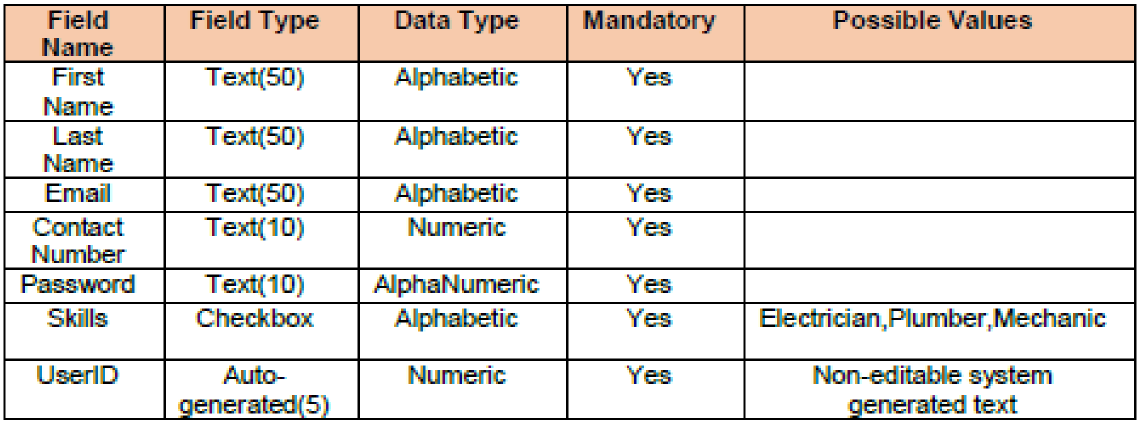
REFRENCES

8.1. Tables



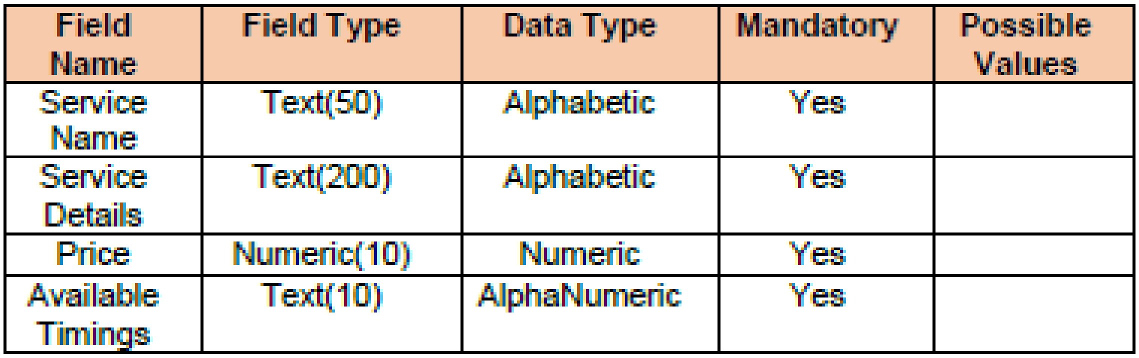
Field Name	Field Type	Data Type	Mandatory	Possible Values
First Name	Text(50)	Alphabetic	Yes	
Last Name	Text(50)	Alphabetic	Yes	
Email	Text(50)	Alphabetic	Yes	
Contact Number	Text(10)	Numeric	Yes	
Password	Text(10)	AlphaNumeric	Yes	
AdminID	Auto-generated(5)	Numeric	Yes	Non-editable system generated text

Table 8.1



Field Name	Field Type	Data Type	Mandatory	Possible Values
First Name	Text(50)	Alphabetic	Yes	
Last Name	Text(50)	Alphabetic	Yes	
Email	Text(50)	Alphabetic	Yes	
Contact Number	Text(10)	Numeric	Yes	
Password	Text(10)	AlphaNumeric	Yes	
Skills	Checkbox	Alphabetic	Yes	Electrician,Plumber,Mechanic
UserID	Auto-generated(5)	Numeric	Yes	Non-editable system generated text

Table 8.2



Field Name	Field Type	Data Type	Mandatory	Possible Values
Service Name	Text(50)	Alphabetic	Yes	
Service Details	Text(200)	Alphabetic	Yes	
Price	Numeric(10)	Numeric	Yes	
Available Timings	Text(10)	AlphaNumeric	Yes	

Table 8.3

CHAPTER 9

TECHNOLOGY DESCRIPTION

Technologies and tools that are used to build this project are: -

9.1. Java: -

Java is a high-level programming language originally developed by Sun Microsystems and released in 1995. Java runs on a variety of platforms, such as Windows, Mac OS, and the various versions of UNIX. Java is guaranteed to be **Write Once, Run Anywhere.**

Java is −

- **Object Oriented** − In Java, everything is an Object. Java can be easily extended since it is based on the Object model.
- **Platform Independent** − Unlike many other programming languages including C and C++, when Java is compiled, it is not compiled into platform specific machine, rather into platform independent byte code. This byte code is distributed over the web and interpreted by the Virtual Machine (JVM) on whichever platform it is being run on.
- **Simple** − Java is designed to be easy to learn. If you understand the basic concept of OOP Java, it would be easy to master.
- **Secure** − With Java's secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.
- **Architecture-neutral** − Java compiler generates an architecture-neutral object file format, which makes the compiled code executable on many processors, with the presence of Java runtime system.
- **Portable** − Being architecture-neutral and having no implementation dependent aspects of the specification makes Java portable. Compiler in Java is written in ANSI C with a clean portability boundary, which is a POSIX subset.
- **Robust** − Java makes an effort to eliminate error prone situations by emphasizing mainly on compile time error checking and runtime checking.

- **Multithreaded** − With Java's multithreaded feature it is possible to write programs that can perform many tasks simultaneously. This design feature allows the developers to construct interactive applications that can run smoothly.
- **Interpreted** − Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light-weight process.
- **High Performance** − With the use of Just-In-Time compilers, Java enables high performance.
- **Distributed** − Java is designed for the distributed environment of the internet.
- **Dynamic** − Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry extensive amount of run-time information that can be used to verify and resolve accesses to objects on run-time.

9.2. Spring: -

Spring is a lightweight framework. It can be thought of as a framework of frameworks because it provides support to various frameworks such as Struts, Hibernate, Tapestry, EJB, JSF etc. The framework, in broader sense, can be defined as a structure where we find solution of the various technical problems.

The Spring framework comprises several modules such as IOC, AOP, DAO, Context, ORM, WEB MVC etc. We will learn these modules in next page. Let's understand the IOC and Dependency Injection first.

The core features of the Spring Framework can be used in developing any Java application, but there are extensions for building web applications on top of the Java EE platform. Spring framework targets to make J2EE development easier to use and promotes good programming practices by enabling a POJO-based programming model.

Following is the list of few of the great benefits of using Spring Framework −

- Spring enables developers to develop enterprise-class applications using POJOs. The benefit of using only POJOs is that you do not need an EJB container product such as an application server but you have the option of using only a robust servlet container such as Tomcat or some commercial product.

- Spring is organized in a modular fashion. Even though the number of packages and classes are substantial, you have to worry only about the ones you need and ignore the rest.
- Spring does not reinvent the wheel, instead it truly makes use of some of the existing technologies like several ORM frameworks, logging frameworks, JEE, Quartz and JDK timers, and other view technologies.
- Testing an application written with Spring is simple because environment-dependent code is moved into this framework. Furthermore, by using JavaBeanstyle POJOs, it becomes easier to use dependency injection for injecting test data.
- Spring's web framework is a well-designed web MVC framework, which provides a great alternative to web frameworks such as Struts or other over-engineered or less popular web frameworks.
- Spring provides a convenient API to translate technology-specific exceptions (thrown by JDBC, Hibernate, or JDO, for example) into consistent, unchecked exceptions.
- Lightweight IoC containers tend to be lightweight, especially when compared to EJB containers, for example. This is beneficial for developing and deploying applications on computers with limited memory and CPU resources.
- Spring provides a consistent transaction management interface that can scale down to a local transaction (using a single database, for example) and scale up to global transactions (using JTA, for example).

9.3. Hibernate: -

Hibernate is an **O**bject-**R**elational **M**apping (ORM) solution for JAVA. It is an open source persistent framework created by Gavin King in 2001. It is a powerful, high performance Object-Relational Persistence and Query service for any Java Application.

Hibernate maps Java classes to database tables and from Java data types to SQL data types and relieves the developer from 95% of common data persistence related programming tasks.

Hibernate sits between traditional Java objects and database server to handle all the works in persisting those objects based on the appropriate O/R mechanisms and patterns.

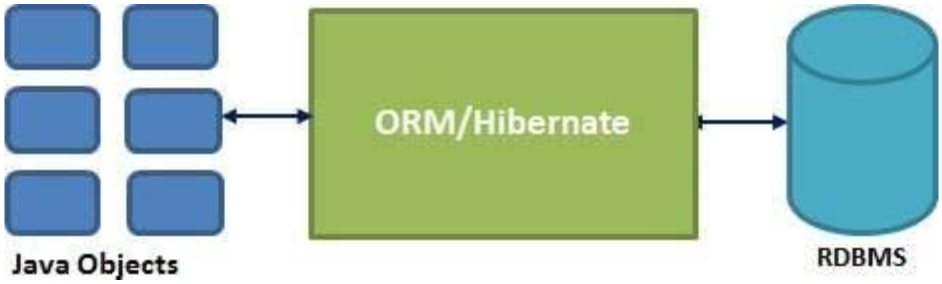


Figure 11.3.

9.3.1. Hibernate Advantages

- Hibernate takes care of mapping Java classes to database tables using XML files and without writing any line of code.
- Provides simple APIs for storing and retrieving Java objects directly to and from the database.
- If there is change in the database or in any table, then you need to change the XML file properties only.
- Abstracts away the unfamiliar SQL types and provides a way to work around familiar Java Objects.
- Hibernate does not require an application server to operate.
- Manipulates Complex associations of objects of your database.
- Minimizes database access with smart fetching strategies.
-

9.4. JSP: -

Java Server Pages (JSP) is a server-side programming technology that enables the creation of dynamic, platform-independent method for building Web-based applications. JSP have access to the entire family of Java APIs, including the JDBC API to access enterprise databases.

JavaServer Pages (JSP) is a technology for developing Webpages that supports dynamic content. This helps developers insert java code in HTML pages by making use of special JSP tags, most of which start with <% and end with %>.

A JavaServer Pages component is a type of Java servlet that is designed to fulfill the role of a user interface for a Java web application. Web developers write JSPs as text files that combine HTML or XHTML code, XML elements, and embedded JSP actions and commands.

Using JSP, you can collect input from users through Webpage forms, present records from a database or another source, and create Webpages dynamically.

JSP tags can be used for a variety of purposes, such as retrieving information from a database or registering user preferences, accessing JavaBeans components, passing control between pages, and sharing information between requests, pages etc.

9.4.1. Advantages of JSP

Following table lists out the other advantages of using JSP over other technologies −

vs. Active Server Pages (ASP)

The advantages of JSP are twofold. First, the dynamic part is written in Java, not Visual Basic or other MS specific language, so it is more powerful and easier to use. Second, it is portable to other operating systems and non-Microsoft Web servers.

vs. Pure Servlets

It is more convenient to write (and to modify!) regular HTML than to have plenty of println statements that generate the HTML.

vs. Server-Side Includes (SSI)

SSI is really only intended for simple inclusions, not for "real" programs that use form data, make database connections, and the like.

vs. JavaScript

JavaScript can generate HTML dynamically on the client but can hardly interact with the web server to perform complex tasks like database access and image processing etc.

vs. Static HTML

Regular HTML, of course, cannot contain dynamic information.

9.5. Eclipse: -

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications

9.6. RDBMS

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

Other kinds of data stores can be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those types of systems.

So nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as foreign keys.

Relational Database Management System (RDBMS) is a software that:

- Enables you to implement a database with tables, columns and indexes.
- Guarantees the Referential Integrity between rows of various tables.
- Updates the indexes automatically.
- Interprets an SQL query and combines information from various tables.

9.7. RDBMS TERMINOLOGY: -

Before I proceed to explain MySQL database system, let's revise few definitions related to database.

Database: A database is a collection of tables, with related data.

- **Table:** A table is a matrix with data. A table in a database looks like a simple spreadsheet.
- **Column:** One column (data element) contains data of one and the same kind, for example the column postcode.
- **Row:** A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.
- **Redundancy:** Storing data twice, redundantly to make the system faster.
- **Primary Key:** A primary key is unique. A key value cannot occur twice in one table. With a key, you can find at most one row.
- **Foreign Key:** A foreign key is the linking pin between two tables.

- **Compound Key:** A compound key (composite key) is a key that consists of multiple columns, because one column is not sufficiently unique.

- **Index:** An index in a database resembles an index at the back of a book.

- **Referential Integrity:** Referential Integrity makes sure that a foreign key value always points to an existing row.

9.8. MYSQL DATABASE: -

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed, and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons:

- MySQL is released under an open-source license. So you have nothing to pay to use it.

- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.

- MySQL uses a standard form of the well-known SQL data language.

- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.

- MySQL works very quickly and works well even with large data sets.

- MySQL is very friendly to PHP, the most appreciated language for web development.

- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

CHAPTER 10

SYSTEM SNAPSHOT

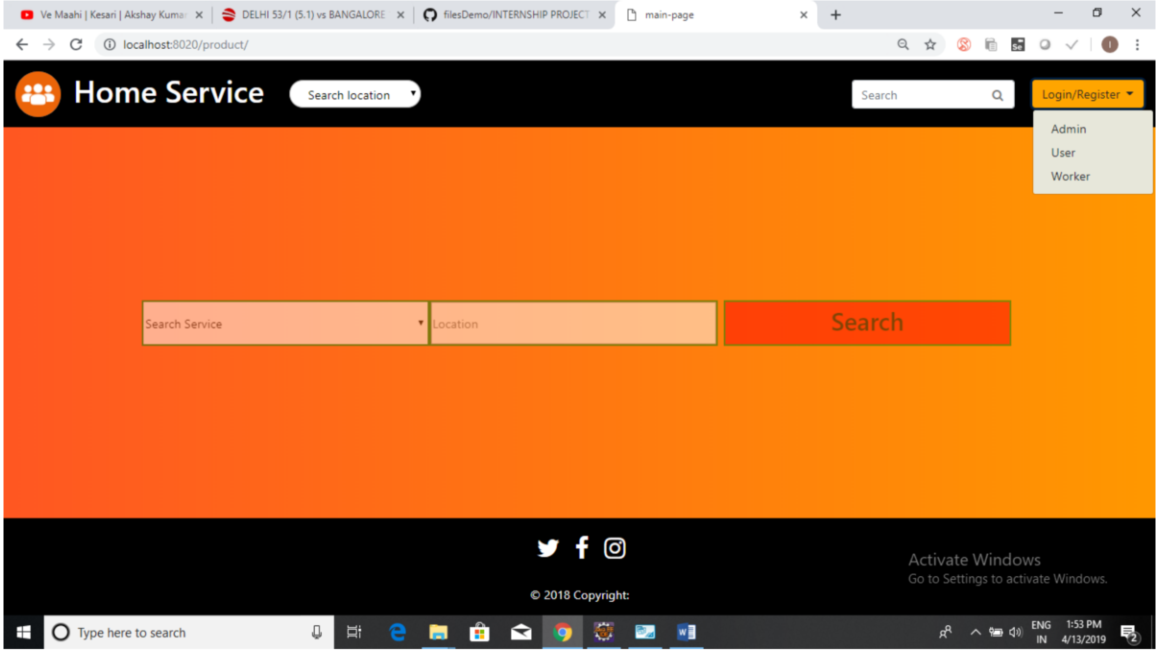
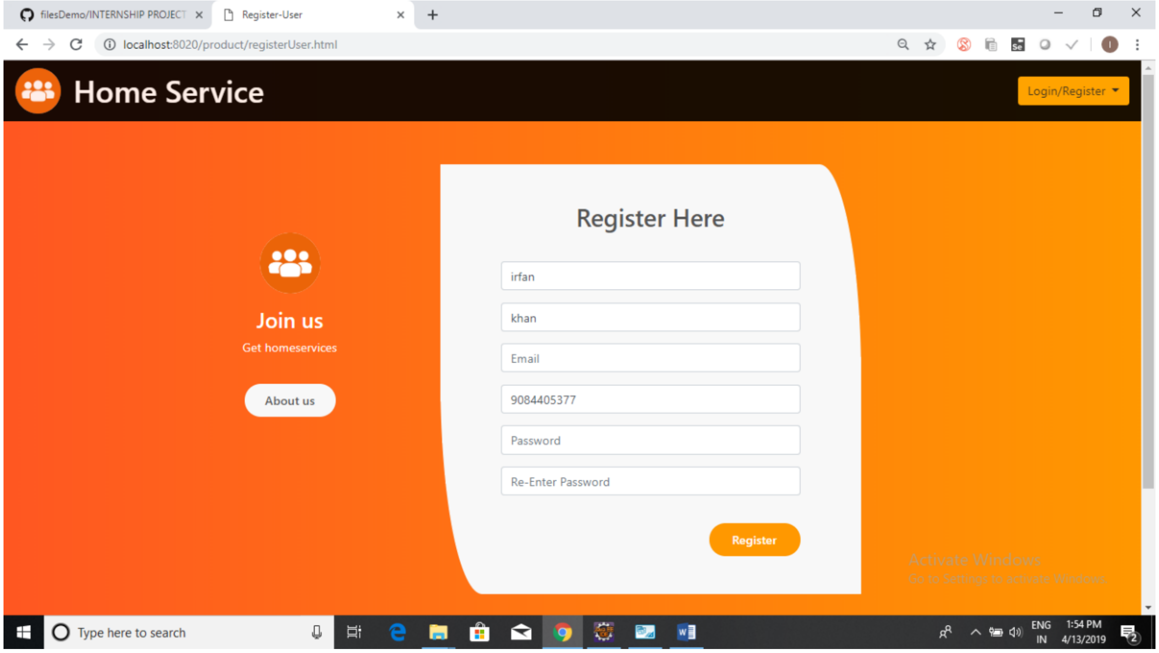


Figure 11.1



irfan

khan

9084405377

Figure 11.2

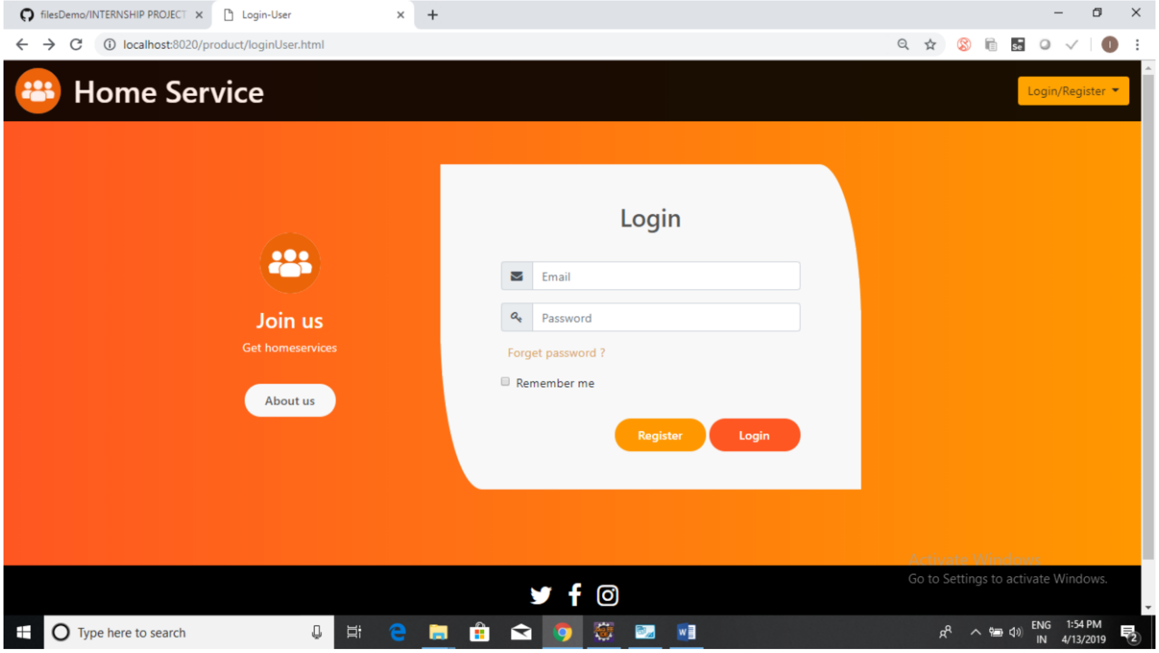
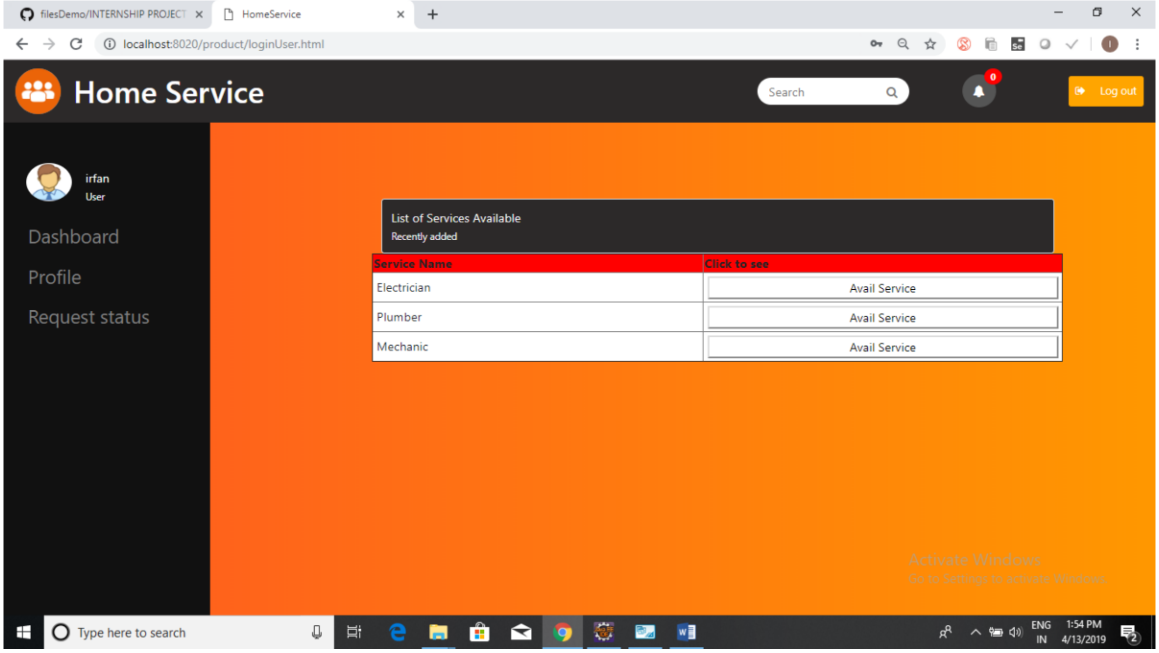


Figure 11.3

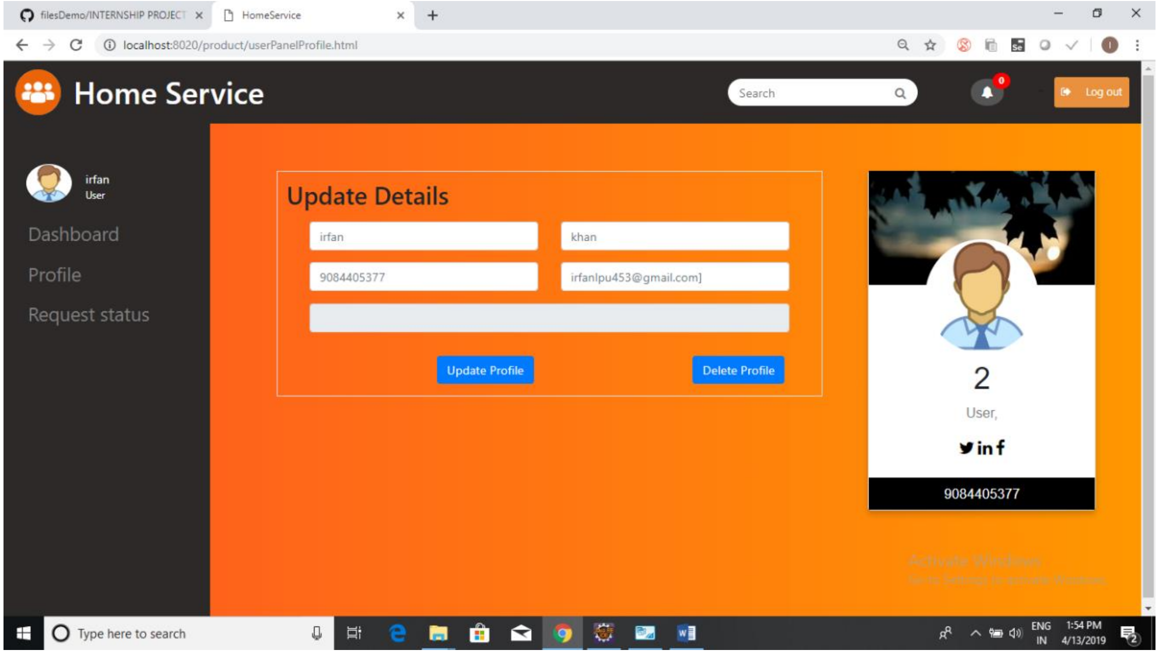


List of Services Available

Recently added

Service Name	Click to see
Electrician	Avail Service
Plumber	Avail Service
Mechanic	Avail Service

Figure 11.4

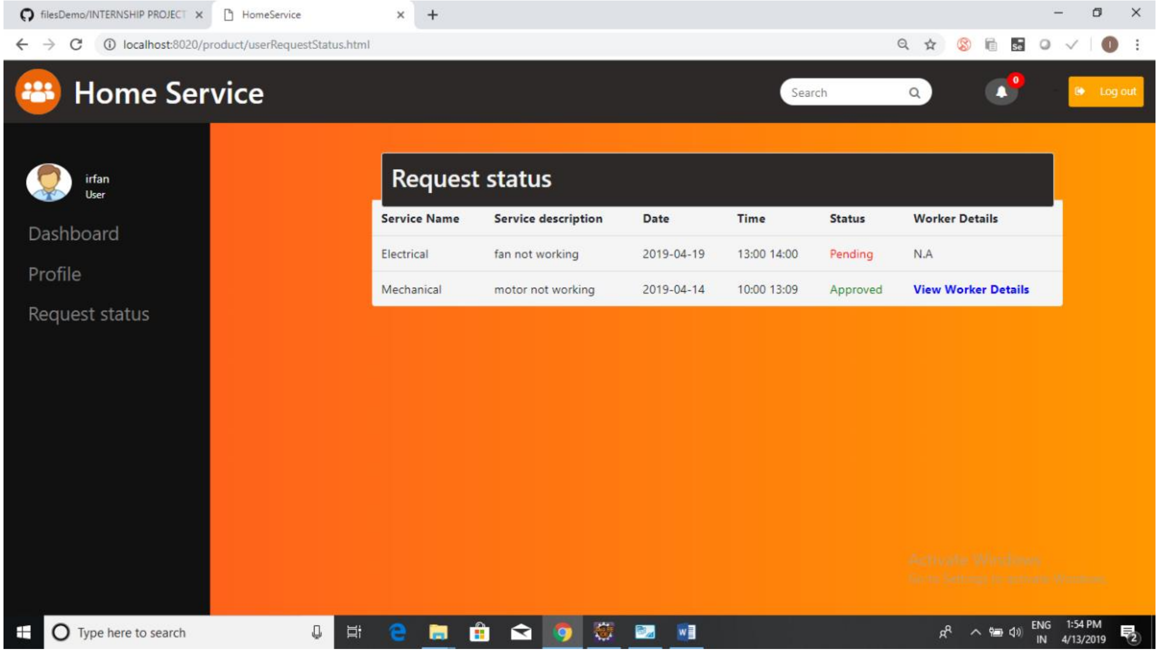


Update Details

irfan
khan
9084405377
irfanlpu453@gmail.com]

9084405377

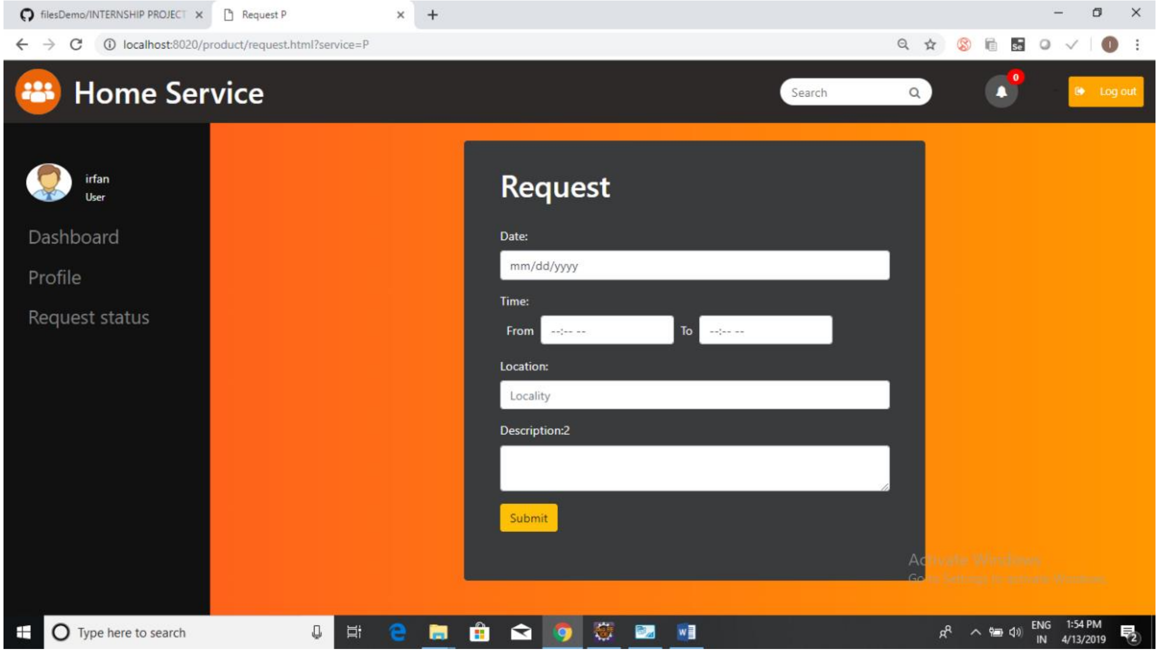
Figure 11.5



Request status

Service Name	Service description	Date	Time	Status	Worker Details
Electrical	fan not working	2019-04-19	13:00 14:00	Pending	N.A
Mechanical	motor not working	2019-04-14	10:00 13:09	Approved	View Worker Details

Figure 11.6



Request

Date:

mm/dd/yyyy

Time:

From --:-- -- To --:-- --

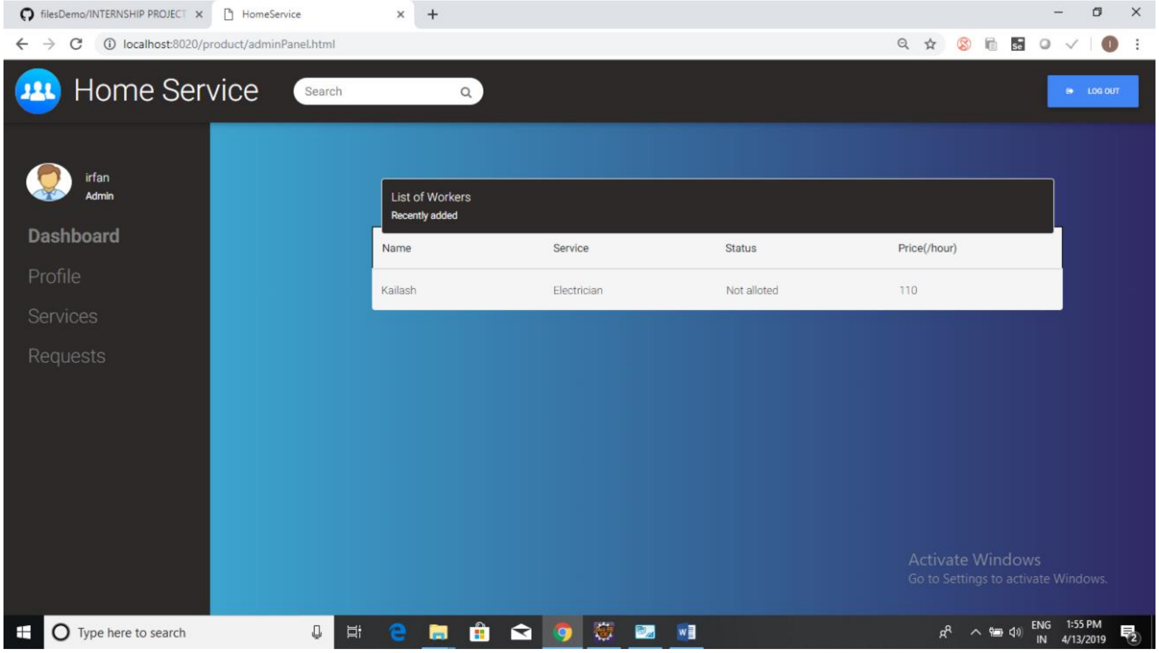
Location:

Locality

Description:2

Submit

Figure 11.7

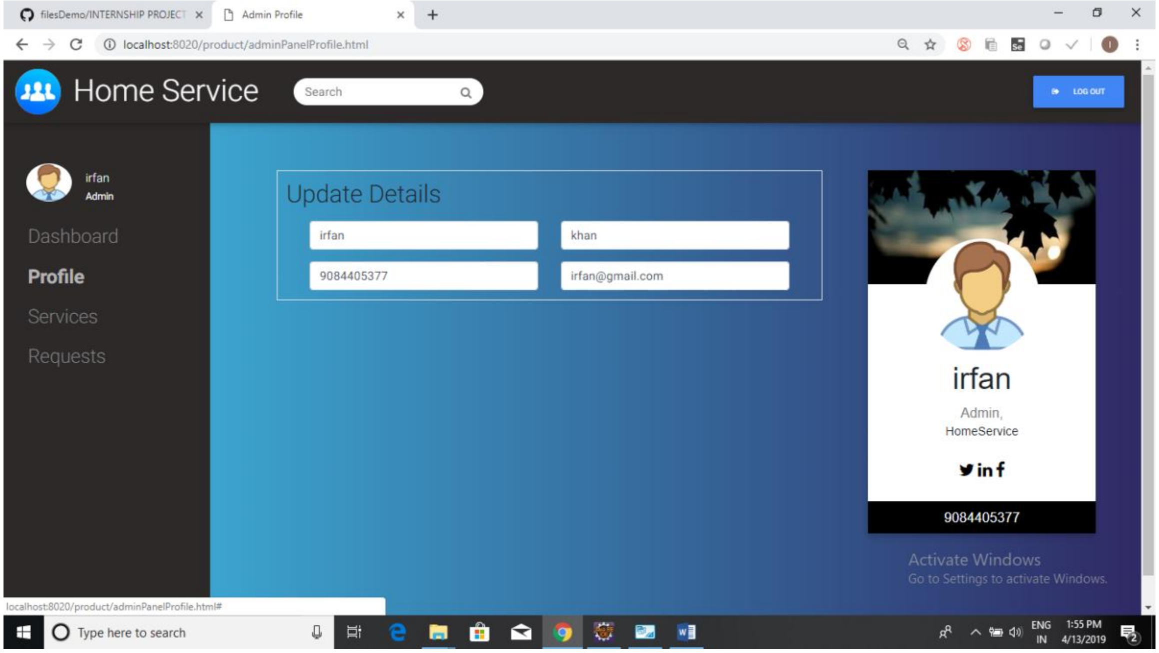


List of Workers

Recently added

Name	Service	Status	Price(/hour)
Kailash	Electrician	Not alloted	110

Figure 11.8



Update Details

irfan	khan
9084405377	irfan@gmail.com

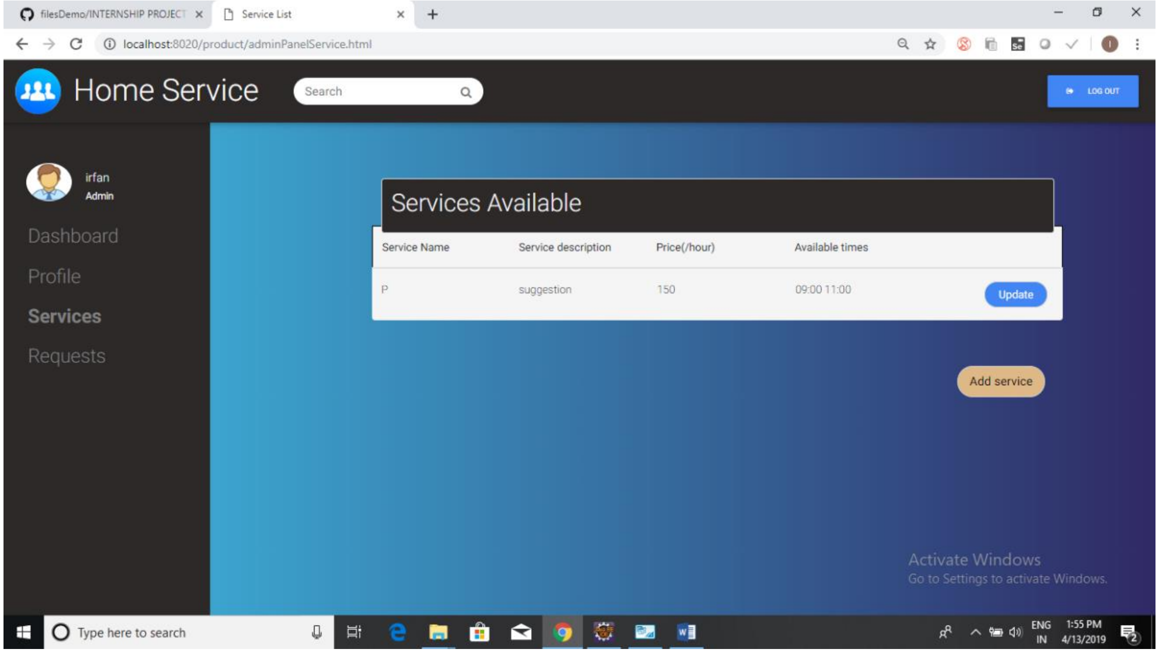
irfan

Admin,
HomeService

9084405377

localhost:8020/product/adminPanelProfile.html#

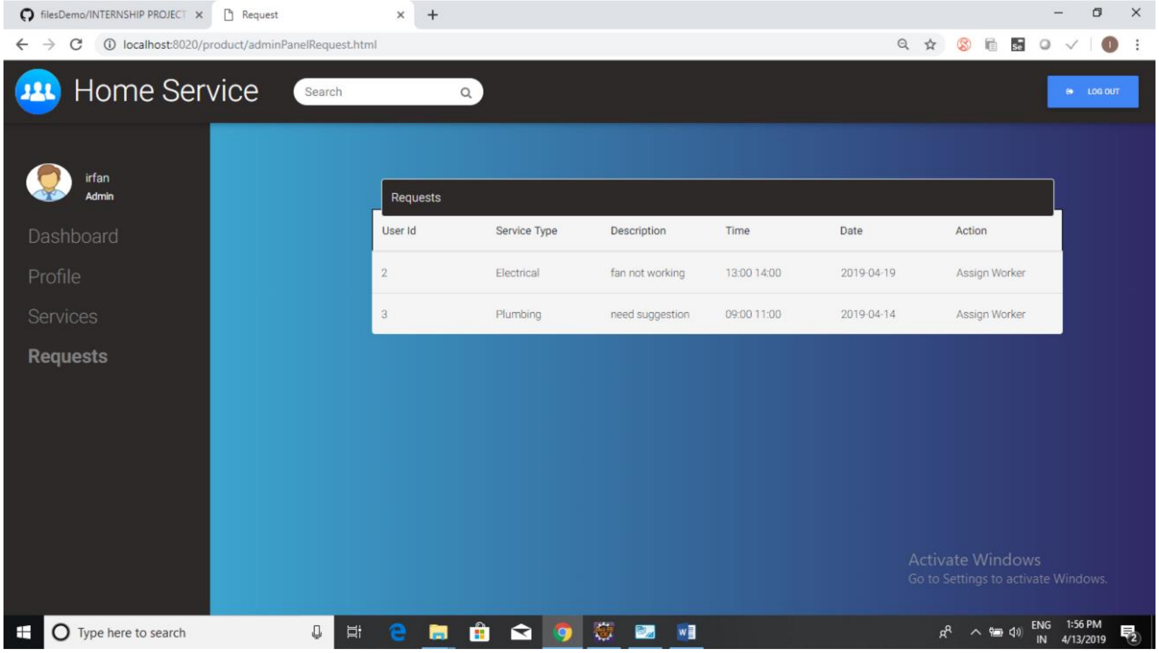
Figure 11.9



Services Available

Service Name	Service description	Price(/hour)	Available times	
P	suggestion	150	09:00 11:00	Update

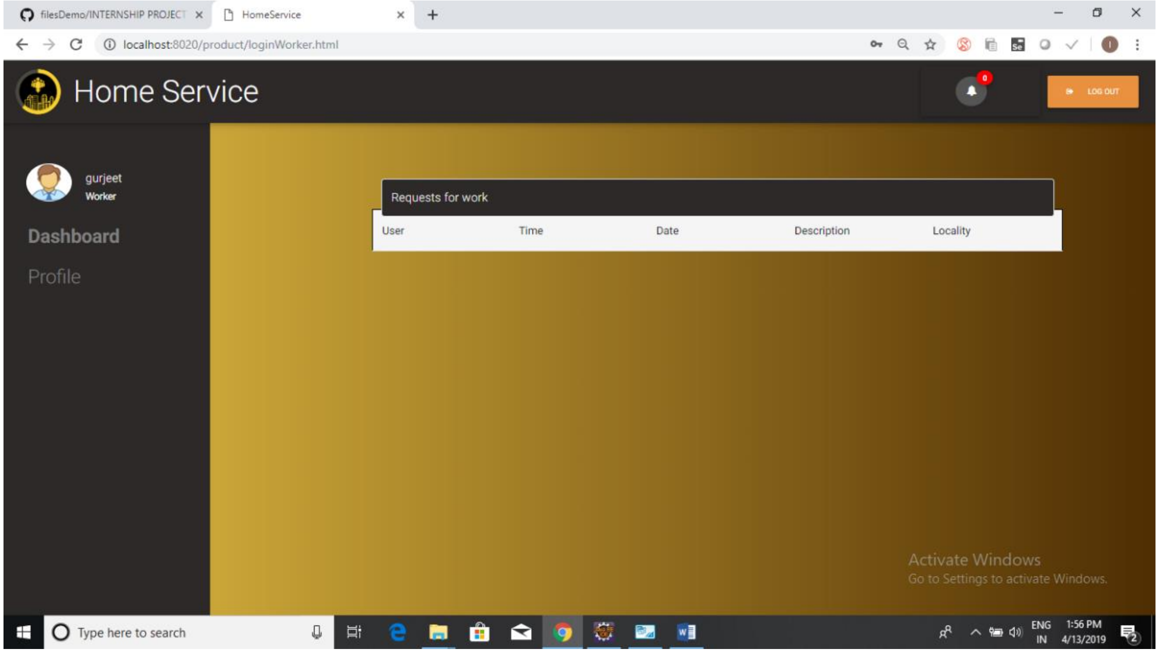
Figure 11.10



Requests

User Id	Service Type	Description	Time	Date	Action
2	Electrical	fan not working	13:00 14:00	2019-04-19	Assign Worker
3	Plumbing	need suggestion	09:00 11:00	2019-04-14	Assign Worker

Figure 11.11



Requests for work

User	Time	Date	Description	Locality

Figure 11.12

CHAPTER 11

BIBILIOGRAPHY