\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Software Requirements Specification

for

<Home Service Provider >

Version 1.0 approved

Prepared by

Meet Kamani

Rushit Kashipara

Harsh Dhankecha

Date:17-Sep-2022

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 3

1.1 Purpose 3

1.2 Document Conventions 3

1.3 Intended Audience and Reading Suggestions 3

1.4 Project Scope 4

1.5 References 5

2. Overall Description 5

2.1 Product Perspective 6

2.2 Product Features 6

2.3 User Classes and Characteristics 7

2.4 Operating Environment 7

2.5 Design and Implementation Constraints 8

2.6 User Documentation 8

2.7 Assumptions and Dependencies 9

3. System Features 9

3.1 System Feature 1 9

3.2 System Feature 2 (and so on) 10

4. External Interface Requirements 13

4.1 User Interfaces 13

4.2 Hardware Interfaces 13

4.3 Software Interfaces 13

4.4 Communications Interfaces 14

5. Other Nonfunctional Requirements 14

5.1 Performance Requirements 14

5.2 Safety Requirements 14

5.3 Security Requirements 14

5.4 Software Quality Attributes 15

6. Other Requirements 15

Appendix A: Glossary 15

Appendix B: Analysis Models 15

Appendix C: Issues List 15

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Home Service Provider | 13-sep-2022 | initial changes | Version 1 |

# Introduction

## Purpose

Our purpose of developing this project is mainly online there website and android application.

We have observed how limitation in exiting system:

* No guarantied service
* No security
* Difficult to manage records.

So, our purpose is to overcome this limitation with following features:

* Household service Easy Available.
* Provide Service any time.
* Easy online payment.
* Save time

## Document Conventions

All terms are in italics style.

Main features or important terms are in bold style.

## Intended Audience and Reading Suggestions

Anyone with some basic knowledge of programming can understand this document. The document is intended for Developers, Software architects, Testers, Project managers and Documentation Writers. But anyone with programming background and some experience with UML can understand this document.

This Software Requirement Specification also includes

Overall description of the product

External interface requirements System Features

Other non-functional requirements

## 

## Project Scope

The scope of our project is to designing a complete environment to provide a safe and user-friendly environment for online service booking. The main aim of the project is to provider an easy to use application for services provided for customer.

We often get frustrated while taking the appointment of service provider because there the many problems are occur, like the service provider is busy art somewhere else or his not receiving our call or his cost is very high according to problem. So in this project we will remove this headache.

## References

<https://ukdiss.com/examples/android-app-for-household-services.php>

e-Home service

# Overall Description

## Product Perspective

This document contains the problem statement that the current system is facing which is hampering the growth opportunities of the company. It further contains a list of the stakeholders and users of the proposed solution. It also illustrates the needs and wants of the stakeholders that were identified in the brainstorming exercise as part of the requirements workshop. It further lists and briefly describes the major features and a brief description of each of the proposed system.

**2.1 Product Features**

This app is build with several features. Some of them are as follows:

1. This app provides details of all the service providers which are required in daily life like painter, plumbers, cleaning, electricians, etc.
2. This app is beneficial for both the persons requiring service as well as the service providers.
3. For persons requiring service in a way that they can get details of the service providers around their locality as well as around any region / area. Secondly, it saves their time as they get services sitting at home itself.
4. For the service providers in a sense that they get jobs/ orders sitting at homes.
5. This app filters out the results of these service providers according to the region / area.
6. It gives all the details of the persons providing services like name, age, address, mobile no., job experience, previous records, no. of complaints, etc.
7. This app provides option for filling the feedback form after they get the required services to tell others whether the person is good or not in his/ her job.

## User Classes and Characteristics

The users of the system include:

1.  Admin

2.  Service Provider

3. Customer

1. Admin

Admin can verify service provider or customer.

Admin can manage categories of services.

Admin can add new advertising.

1. Service Provider:

In this android application service provider first do registration and then login after this process the service provider can view the service which are ordered by user and send acknowledgement to the user in positive reply then service provider comes at the place of customer then do his work.

1. Customer:

In this application the customer first do registration and then do login, after the user search for the particular service and receiver the list of service available on our android application. The user then selects the service and request for the service after this process

## Operating Environment

In [computer software](https://en.wikipedia.org/wiki/Computer_software), an operating environment or integrated applications environment is the [environment](https://en.wikipedia.org/wiki/Deployment_environment) in which users run [application software](https://en.wikipedia.org/wiki/Application_software). The environment consists of a [user interface](https://en.wikipedia.org/wiki/User_interface) provided by an applications manager and usually an [application programming interface](https://en.wikipedia.org/wiki/Application_programming_interface) (API) to the applications manager.

An operating environment is *not* a full [operating system](https://en.wikipedia.org/wiki/Operating_system), but is a form of [middleware](https://en.wikipedia.org/wiki/Middleware) that rests between the OS and the application. For example, the first version of [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [Windows 1.0](https://en.wikipedia.org/wiki/Windows_1.0), was not a full operating system, but a [GUI](https://en.wikipedia.org/wiki/GUI) laid over DOS albeit with an API of its own. Similarly, the [IBM U2](https://en.wikipedia.org/wiki/IBM_U2) system operates on both [Unix](https://en.wikipedia.org/wiki/Unix)/[Linux](https://en.wikipedia.org/wiki/Linux) and [Windows NT](https://en.wikipedia.org/wiki/Windows_NT). Usually the user interface is [text-based](https://en.wikipedia.org/wiki/Text-based_user_interface) or [graphical](https://en.wikipedia.org/wiki/Graphical_user_interface), rather than a [command-line interface](https://en.wikipedia.org/wiki/Command-line_interface) (e.g., [DOS](https://en.wikipedia.org/wiki/DOS) or the [Unix shell](https://en.wikipedia.org/wiki/Unix_shell)), which is often the interface of the underlying operating system.

In the mid-1980s, [text-based](https://en.wikipedia.org/wiki/Text-based_user_interface) and [graphical](https://en.wikipedia.org/wiki/Graphical_user_interface) user interface operating environments surrounded [DOS](https://en.wikipedia.org/wiki/DOS) operating systems with a [shell](https://en.wikipedia.org/wiki/Shell_(computing)) that turned the user's [display](https://en.wikipedia.org/wiki/Computer_monitor) into a [menu](https://en.wikipedia.org/wiki/Menu_(computing))-oriented "[desktop](https://en.wikipedia.org/wiki/Desktop_metaphor)" for selecting and running [PC](https://en.wikipedia.org/wiki/IBM_PC_compatible) applications. These operating environment systems allow users much of the convenience of [integrated software](https://en.wikipedia.org/wiki/Integrated_software) without locking them into a single package.

### Design and Implementation Constraints

The information off all users and services must be stored I database.

## User Documentation

Accuweather will be designed in such a way that it should be very much easy to use and will be very simple

navigate with. However the user of the product might face some difficulties while using it. In order to

overcome these problems the product the product will provide its users with a functionality that is FAQ.

All the frequently asked questions are answered there. And if user’s question is not there they can ask us in

feedback option.

Accuweather will be designed in such a way that it should be very much easy to use and will be very simple

navigate with. However the user of the product might face some difficulties while using it. In order to

overcome these problems the product the product will provide its users with a functionality that is FAQ.

All the frequently asked questions are answered there. And if user’s question is not there they can ask us in

feedback option.

Accuweather will be designed in such a way that it should be very much easy to use and will be very simple

navigate with. However the user of the product might face some difficulties while using it. In order to

overcome these problems the product the product will provide its users with a functionality that is FAQ.

All the frequently asked questions are answered there. And if user’s question is not there they can ask us in

feedback option.

Accuweather will be designed in such a way that it should be very much easy to use and will be very simple

navigate with. However the user of the product might face some difficulties while using it. In order to

overcome these problems the product the product will provide its users with a functionality that is FAQ.

All the frequently asked questions are answered there. And if user’s question is not there they can ask us in

feedback option.

Accuweather will be designed in such a way that it should be very much easy to use and will be very simplenavigate with. However the user of the product might face some difficulties while using it. In order toovercome these problems the product the product will provide its users with a functionality that is FAQ.All the frequently asked questions are answered there. And if user’s question is not there they can ask us infeedback option.

## Assumptions and Dependencies

Common features including Login, Logout, forgot password, Change Password, User management features etc. which will be used across all software applications as part of Mission Mode Project will be developed commonly and uniformly. The availability of software applications with which ISS would be interoperating will be crucial for smooth functioning of ISS It is assumed that the third-party tools and applications software wherever required to fulfill the functionality of ISS will be available on the machines where such features will be executed. These may include office tools for viewing the PDF and charts generated by the package. Further, a use case wise description of assumptions has been described, wherever applicable.

Note: At a minimum Internet Explorer 6.0 + would be required on the client machines to access the software

# System Features.

## Feasibility Study

Feasibility study is carried out when there is a complex problem or opportunity. It is considered as the primary investigation which emphasizes on “Look before You Loop” approach to any project a Feasibility study is undertaken to determine the possibility of either improving the existing system or developing a completely new system.

We are going to developed the new system which is feasible as our application is very user friendly and easy to understand.

3.2 Technical Feasibility

In this type of study the current technology in used in an organization is checked such as the existing software, hardware, and personnel staff to determine whether it will work for the proposed system or completely new ones is to be used. The technology that was important in developing a new system such as Development tools, back-end database system were available from within the organization. The proposed system is capable of adding, changing, enhancing functionality, features etc. The proposed system is capable of handling large storage of data. The back-end and front-end technology has greater important for providing an accurate, error-free, frequencies of data to be used.

Our project is technically feasible in terms of current technology. Our project will provide latest platform like android technology.

#### 3.3 Economical Feasibility

For proving that system developed is economical, the economical feasibility study takes place to check the cost of developing a system against the benefits that it provides. If the cost are less and benefits are more than we can define our system to be economically developed. User saves time in searching for a particular product to be purchased by simply few clicks. The registration process is speedier than the registered manually. The saving of papers as all data are stored computerized. The record is of free of human errors as there is less chance of mistakes. The above benefits are in terms of saving time, minimize errors and provide efficiency in work done.

In terms of economical feasibility our application is very reasonable in cost. So application is economically feasible.

3.4 Operational Feasibility

The operational feasibility is concerned with the operability of the system after it has been installed. That is, some programmer may not like changes in their routine method of work or has fear that they will lose their peer group .The following areas will have the operational feasibility in the proposed project

* The organization has approved this system as their working system.
* The User of the system has accepted the proposed system as their new working system and realized the benefits of it.
* The system will work in a proper way after it has been installed and the installation process is easy to use.p0p;/-[‘

.

# External Interface Requirements

## User Interfaces

Login Interface:

In case the user is not registered yet, he can enter the details and register. Which asks the user to type his username and password. If the user entered either his username or password incorrectly then an error message occurs.

Search:

The customer can enter the home page and he is looking for service then he can search for the need service by entering name.

Categories view:

Categories view shows the Services categories.

## Hardware Interfaces

.

Only the recommended configuration (android mobile devices) no other specific hardware is required.

## Software Interfaces

1. The e-store system shall communicate with the Configurator to identify all the available components to configure the product.
2. The e-store shall communicate with the content manager to get the product specifications, offerings and promotions.
3. The e-store system shall communicate with bill Pay system to identify available payment methods, validate the payments and process payment.
4. The e-store system shall communicate to credit management system for handling financing options.
5. The e-store system shall communicate with CRM system to provide support.
6. The e-store system shall communicate with Sales system for order management.
7. The e-store system shall communicate with shipping system for tracking orders and updating of shipping methods.
8. The e-store system shall communicate with external Tax system to calculate tax.
9. The e-store system shall communicate with export regulation system to validate export regulations.

# Other Nonfunctional Requirements

## Performance Requirements

The following performance requirements are to be satisfied by the proposed solution

Throughput: The solution should be capable of handling a load of 30,000 queries per minute.

The software package can be operated as independent system. The amount of transactions generated by the software can be very well tackled by the computer system as proposed. User may be familiar with operating of computer applications. Software package can be operated using mouse or keyboard. Package will provide consistent look and feel as well proper navigation for easy usability Safety Requirements

## Safety Requirements

Scalability refers to the how the proposed system will be scaled up with need and time. In the current scenario, there will be average 100,000 application Users of the system at central, state and village level. At any point of time, no more than 25,000 simultaneous application Users would be expected in the system. The system architecture should be capable of scaling up the Users need and handle increase of Users with no major functionality is changed and within permissible downtime.

## Security Requirements

### Data Transfer

The system shall use secure sockets in all transactions that include any confidential customer information.

The system shall automatically log out all customers after a period of inactivity.

The system shall confirm all transactions with the customer’s web browser.

The system shall not leave any cookies on the customer’s computer containing the user’s password.

The system shall not leave any cookies on the customer’s computer containing any of the user’s confidential information.

### Data Storage

The customer’s web browser shall never display a customer’s password. It shall always be echoed with special characters representing typed characters.

The customer’s web browser shall never display a customer’s credit card number after retrieving from the database. It shall always be shown with just the last 4 digits of the credit card number.

The system’s back-end servers shall never display a customer’s password. The customer’s password may be reset but never shown.

The system’s back-end servers shall only be accessible to authenticated administrators.

The system’s back-end databases shall be encrypted.

## Software Quality Attributes

### Reliability

The system’s reliability is dependent on various factors like ensuring proper validations for each field and form in the system, ensuring a seamless transition in between different forms, displaying proper alerts for errors, ensuring no http or system-based errors are displayed and ensuring that proper markings are done for the mandatory and non-mandatory fields in the system so as to ensure consistent response while saving or updating the data.

### Availability

This attribute is indicative as to whether an application will execute the tasks it is assigned to perform. Availability also includes certain concepts that relate to sIn addition, top-notch availability indicates that a software-driven system will repair any operating faults so that service outage periods would not exceed a specific time value.

# Other Requirements

• Team of skill resources needs to be deputed specifically to run the system at centre and state level

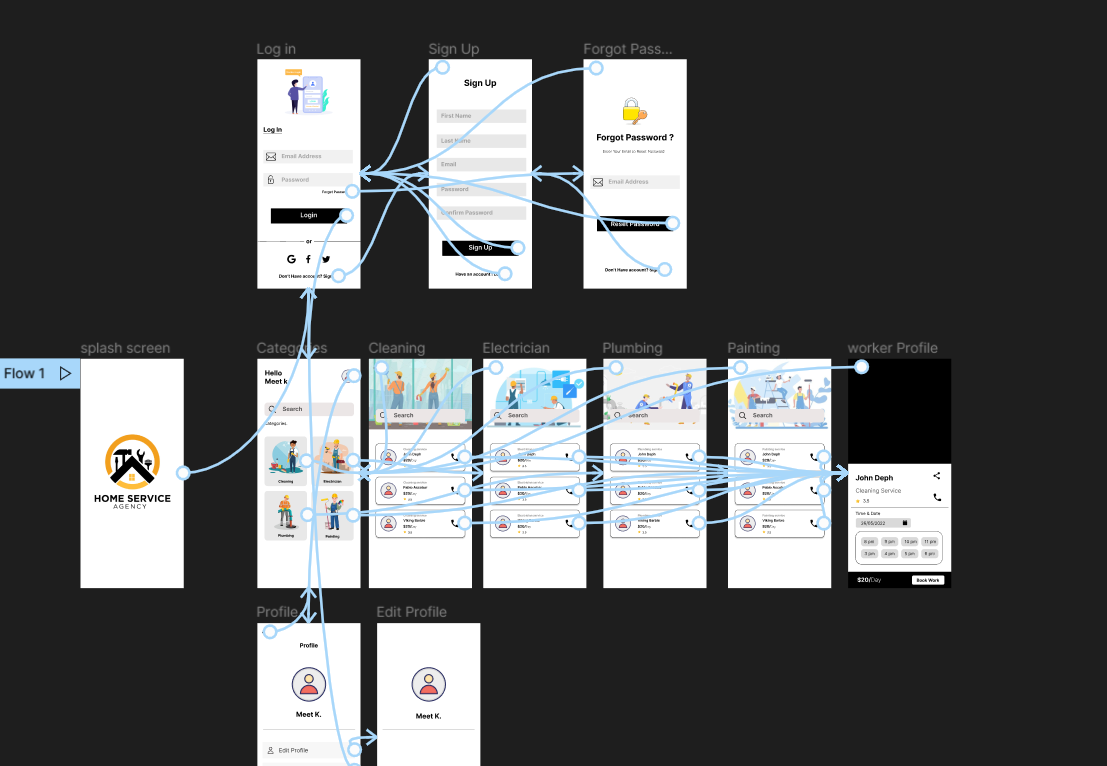
• Regular updations from the side of Centre and State based on the requirements

• Systems used for running the portal at different stakeholder’s place needed to be capable to run the application smoothly.

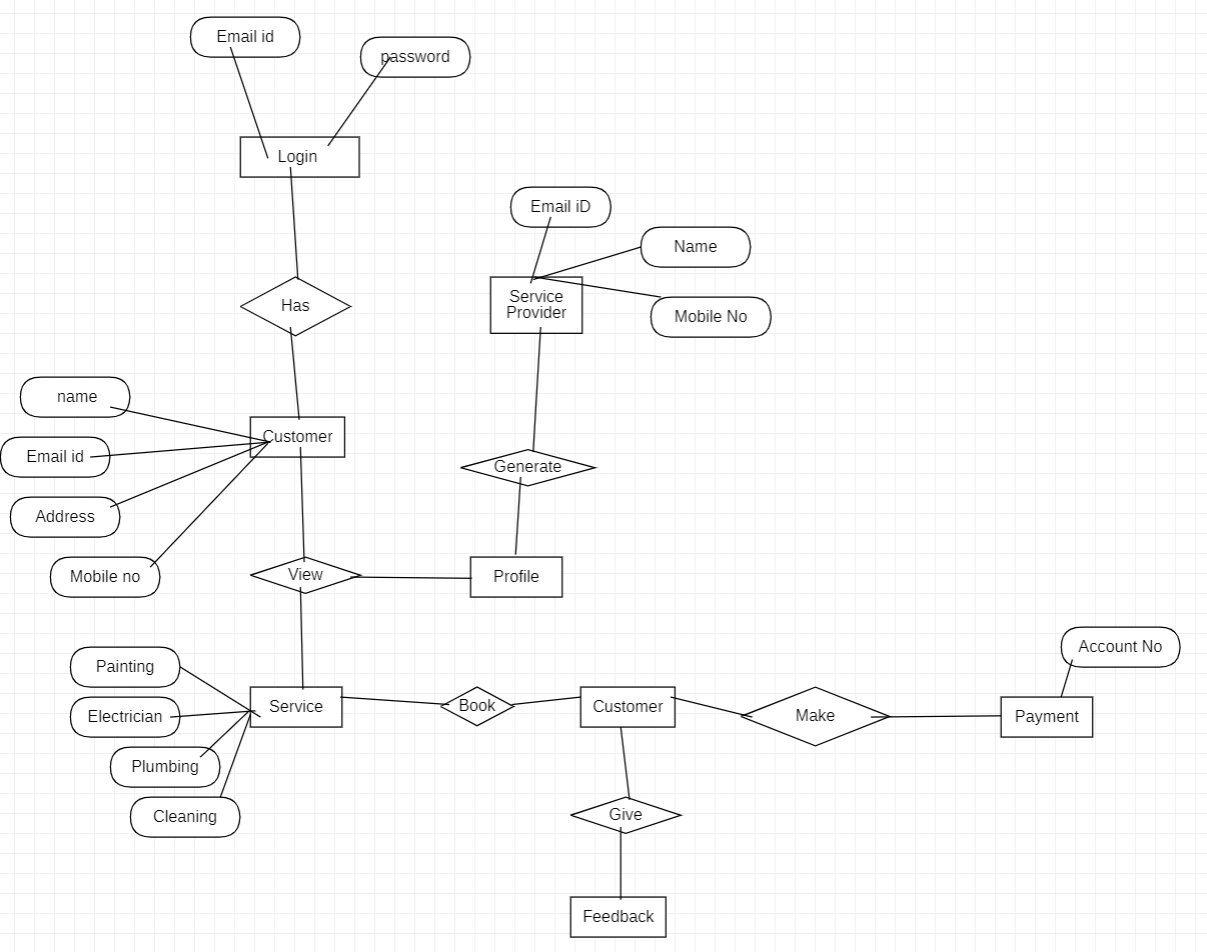
• 24X7 internet connectivity is must to run the system smoothly.

• Supply of the stationary related to the system should be there based on the requirements.

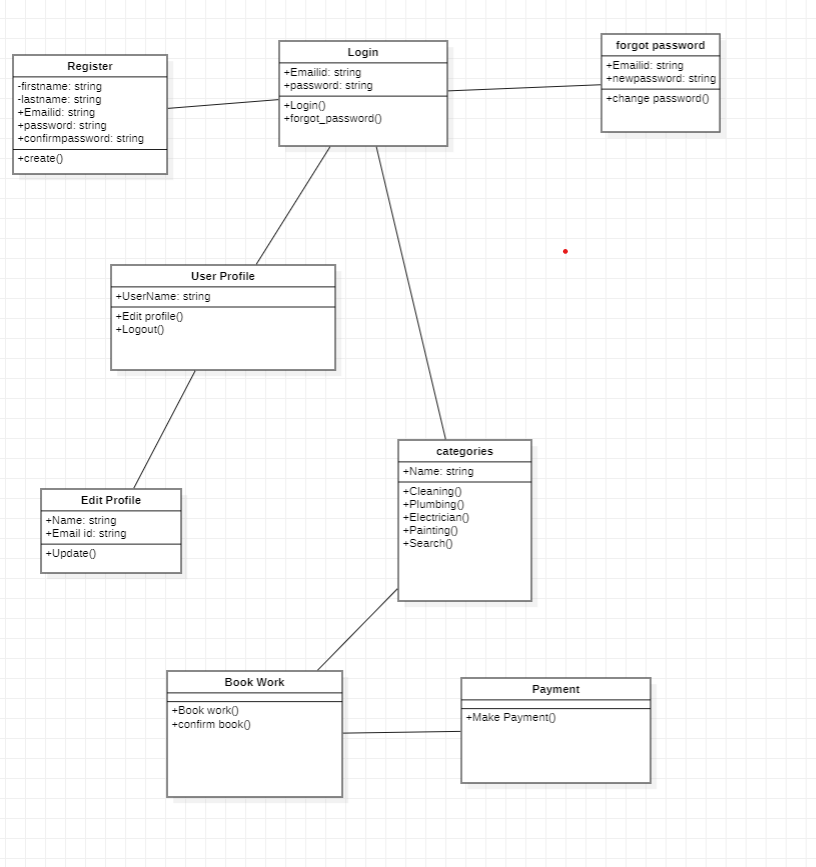
**Wireframe :**

****

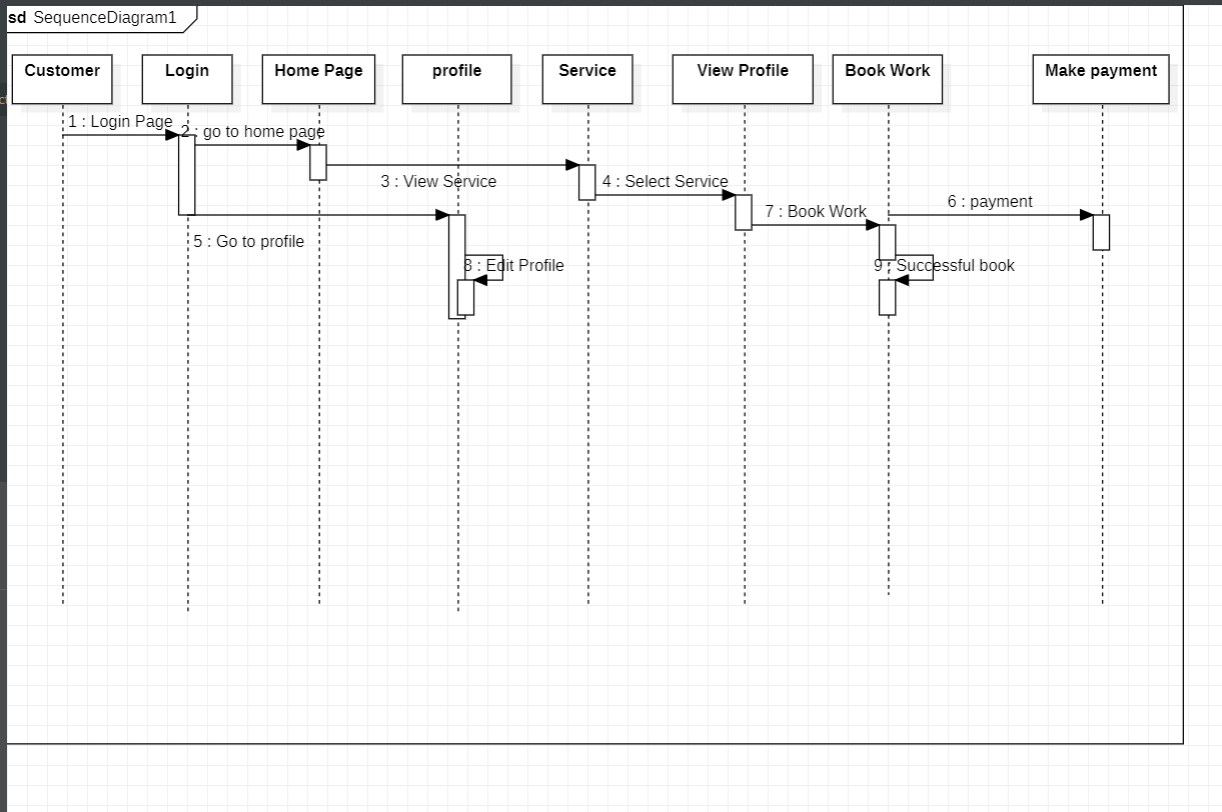
**Flow Chart :**

****

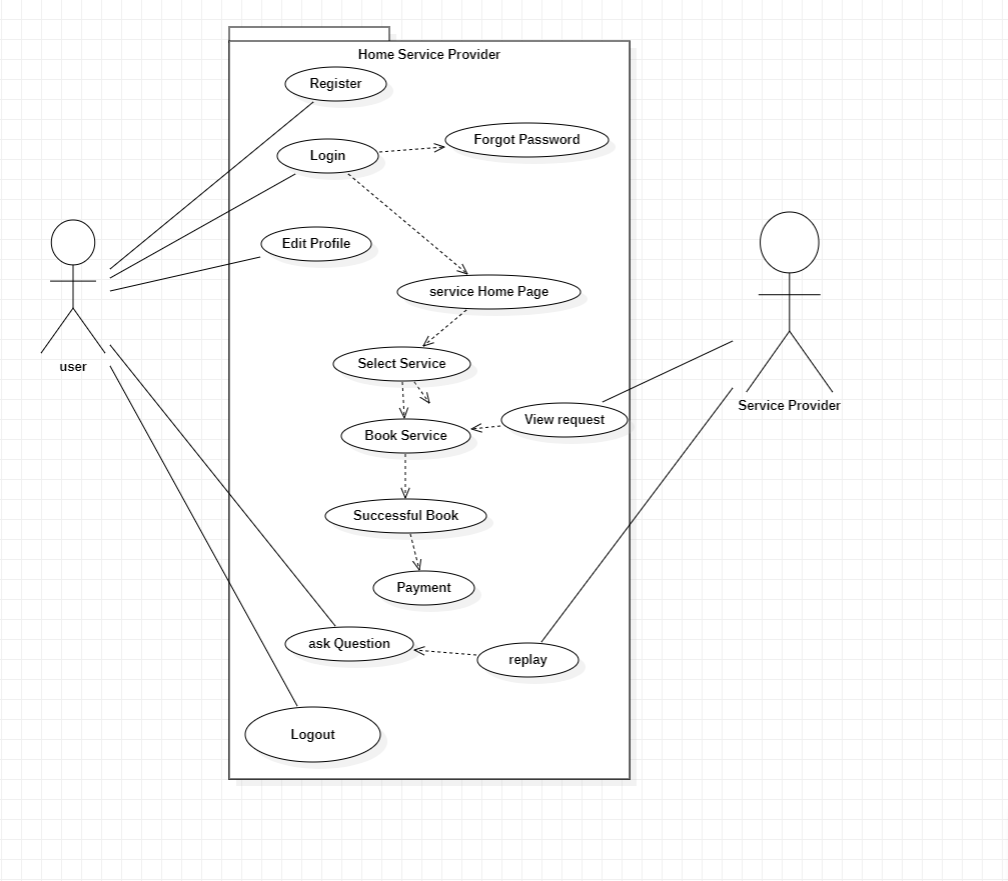
**Class Diagram:**

****

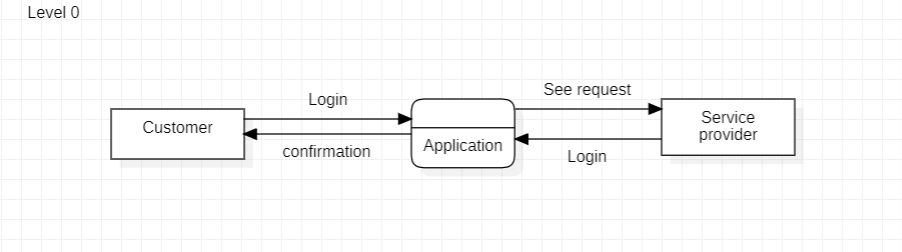
**Sequence Diagram:**

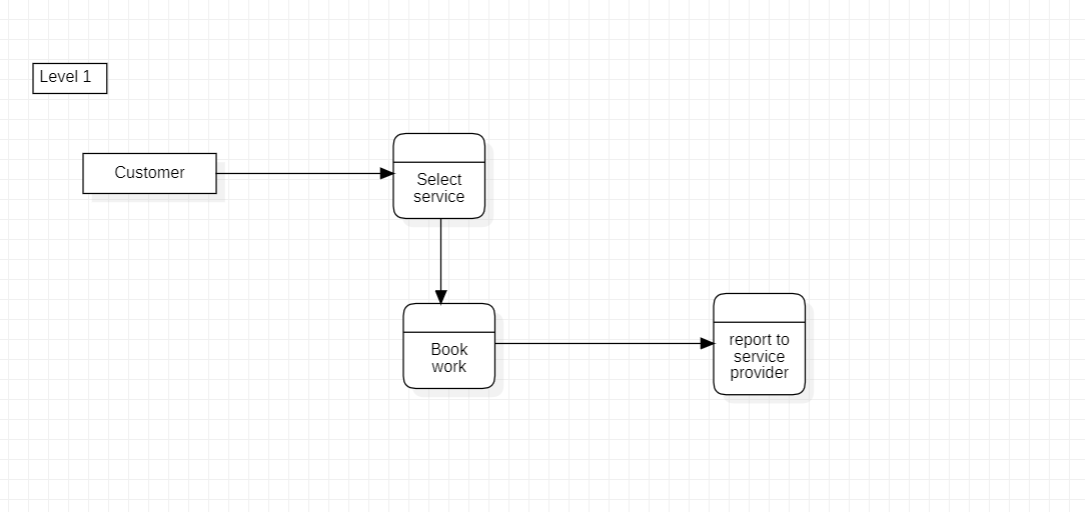
****

**Use Case Diagram:**

****

**DFD Diagram:**

****

****

