Software Requirements Specification

for

Fast Service

Prepared by

Khos Mahmuda Akter BKH2025036F

Hasanur Rahman Shishir MUH2025022M

Mahbub Hasan Talukdar MUH2025011M

Md. Nasim Molla MUH2025024M

Institute of Information Technology
Noakhali Science and Technology University

List of Figure

List of Tables

1. Introduction

This is our System Requirements Specification (SRS) for our Fast Service. Fast Service is an online serviceman hiring platform. This document outlines the functional and non-functional requirements of our platform, as well as the design constraints, scenario, activity diagram, use case diagram and user needs that will guide its development. The SRS includes a description of the system's capabilities, user interfaces, paper-prototype and performance requirements, as well as any relevant technical details and specifications.

1.1 Problem Statement

The aim of our Fast Service is solve the difficulty that faced by homeowners to finding reliable and trustworthy serviceman for their home repair and maintenance work like electric work, painting, plumbing and other services. Traditional methods of finding serviceman, such as asking for recommendations from friends or neighbor, can be time-consuming and may not efficient. Sometimes they are not providing expected services or they didn't have enough experience. Our Fast service aims to provide a convenient and efficient solution for homeowners by connecting them with qualified and reliable serviceman in their area. Through our platform, user can easily search for serviceman based on their expertise, location, and availability, and can read reviews and ratings from previous customers to make informed decisions about who to hire. By streamlining the process of finding and hiring serviceman, we hope to make it easier for user to get the home repairs and maintenance they need. It also connects servicemen to a large community. So, they can get more client using this platform.

1.2 Purpose

The purpose of Fast Service is to provide a one-stop-shop for user to find and hire skilled and reliable serviceman, saving them time and effort in the process. It also helps serviceman to connect with potential customers and grow their businesses.

The purpose of the SRS is to provide a comprehensive and clear outline of the requirements and specifications for a system or software application. It helps any designer and developer to assist in software delivery lifecycle processes. It also serves as a reference document for users, customers, and maintenance personnel. It provides a detailed description of the system's capabilities and functionality, which can be used to troubleshoot any issues or make updates in the future.

1.3 Definitions, acronyms, and abbreviations

This section provides definitions for all document names, acronyms, and abbreviations. The application domain's terms and concepts are defined.

GUI - Graphical User Interface
API – Application Programming Interface
SRS – Software Requirement Specification
UI – User Interface
SDLC – Software Development Life Cycle
MFS – Mobile Finical Services

XML – Extensible Markup Language
RESTful – Representational State Transfer
HTML – Hyper Text Markup Language

1.5 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

1.6 Overview

Now-a-days maximum of our population are connected through internet. We use internet most of the time in a day. So that our project Fast Service will be reach to people widely and easily.

Day by day people and houses are increasing. Besides if our daily-used things like refrigerator, air condition, gas stove and other things get wasted or any other problem occurs like those then it becomes quite irritating and hard to carry out those somewhere to repair or to find correct mechanic to repair. To remove this burden our online platform is here to provide quality service providers for almost everything. It will not only provide service providers to people, but also provide job to the service providers.

Some key features of the serviceman hiring system include:

- A search function that allows homeowners to search for serviceman by location and specialty
- Profiles for serviceman that include information about their skills, qualifications, and experience
- Reviews and ratings from previous customers that help homeowners evaluate the quality of serviceman's work
- A messaging system that allows homeowners to communicate with serviceman and schedule appointments

The Fast Service is designed to be user-friendly with a clean and modern interface that makes it easy for user to find and hire the right handyman for their needs. The system is also designed to be scalable, allowing it to handle a large number of users and transactions as it grows.

2. Stakeholders and Characteristics

2.1 Customer

The Customer are most of the time homeowners, caretakers. For getting a customer haven't require any cost. They can hire any number of serviceman.
There are 3 types of service Mode for an user:

i. **Normal Mode:**
Here, user are get service according to system scrolling algorithm.

ii. Urgent Mode:
Here, user get higher priority for getting response on his problem. For this service user have to pay extra 50 taka to system for each work.

iii. Advance Booking:
In this mode , user have to book a serviceman in advance. Customer have to set a start time for starting job. Customer also have to pay extra for get this service.

The benefits to the consumer include continuous improvement of the software over time. Google offers App Engine which is a platform as a service. The platform can run your service with features like scaling to support any load of traffic. In this model they are selling access to their infrastructure as a service.

2.2 Service Provider

Service Providers are serviceman's they are electrician, plumber, sanitary workers, painting workers etc. They can get more customer by using this Fast Services. They have to buy subscription for use this service.

2.3 Other payment organization

For buy a subscription service provider can use many types of payment methods like mobile financial services (MFS), banks, cards and other method. A payment service are also stakeholder of our service.

3. Design and Implementation Constrains

We have employed design and implementation constraints to ensure the success of this project. It also refers to a tool that allows developers and testers to inspect and interact with the application's user interface (UI) elements.

3.1 Interfaces

There are many types of interfaces are used to this Fast Service application. Such as, User Interface, Software Interface and Hardware Interface.
The protocol used shall be HTTP.
The Port number used will be 80.
The logical address is IPv4 format.

3.1.1 User Interface

The user can use this application using Android devices or iOS devices. Also, It have a web version. So, anyone use more easily using browser. We, use Material design frameworks for attractive user interface.

3.1.2 Hardware Interface

Since the application must run over the Android, iOS and web over the internet, all the hardware shall require to connect internet will be hardware interface for the system.

.

3.1.3 Software Interface

1. The system shall communicate with the Configurator to identify all the available components to configure the service

2. The system shall communicate with the content manager to get the product specifications, offerings and promotions

3. The system shall communicate with MFS services to identify available payment methods, validate the payments and process payment.

4. The system shall communicate with Google Map API for location finding and tracking.

3.1.4 Communication Interface

The Fast Service shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

3.2 Licensing Requirements

Not Applicable

3.3 Legal, Copyright, and Other Notices

Fast Service should display the disclaimers and copyright.

3.4 Applicable Standards

It shall be as per the industry standard.

3.5 Programming Language

We use Flutter framework for developing the application. Flutter is a mobile application development framework created by Google. It uses the Dart programming language and allows developers to build cross-platform apps for android, iOS, and the web.

One of the main benefits of Flutter is that it allows developers to write code that runs natively on both Android and iOS, which can save a lot of time and effort compared to developing separate apps for each platform. Flutter also has a hot reload feature that allows developers to see the changes they make in real-time, which can speed up the development process. We also take a support of java to write native android code.

3.6 XML

Extensible Markup Language (XML) is used to describe data. The XML standard is a flexible way to create information formats and electronically share structured data via the public Internet, as well as via corporate networks. XML data is known as self-describing or self-defining, meaning that the structure of the data is embedded with the data, thus when the data arrives there is no need to pre-build the structure to store the data; it is dynamically understood within the XML.

3.7 Implemented Tools and Platform

Tactics, tools, and strategies are at the heart of every project or business plan. These three essential elements are required if we are to create and implement a social media marketing strategic plan.

3.8 Web Server

A web server is a software program that stores and communicates with web browsers and serves them web pages. It listens for requests from web clients and sends back responses with the requested web pages or other content.
Web servers are an essential part of the World Wide Web, as they are responsible for hosting web sites and making them available to users. We will use the Apache HTTP server to implement this project. We will use RESTful API to retrieve data from our server to mobile application. RESTful stands for Representational State Transfer. And API stands for Application Programming Interface.

3.8 Data Server

MySQL is a free-to-use, open-source database that facilitates effective management of databases by connecting them to the software. That's why we will use MySQL database server to store all of the information of this system. It is a stable, reliable and powerful solution with advanced features like the following:

- Data Security
- On-Demand Scalability
- High Performance
- Round-the-clock Uptime
- Comprehensive Transactional Support
- Complete Workflow Control
- The Fiilexibility of Open Source
- Fault tolerance

4. Requirement Specification

All the requirements based on the elicitation process are described in this section.

4.1 Functional Requirement

Functional requirements specify what a system or component should do and how it should behave. They describe the functions that a system or component must be able to perform and the constraints that must be satisfied. These requirements are typically defined in the early stages of a development project and serve as the basis for design and implementation. They may be expressed in various forms, such as user stories, use cases, or scenarios, and may include details such as input data, output data, processing steps, and error conditions.

4.1.1 Create account

FR-1	Both house owner and service provider have to create account		
Description	User should register his/her account for the first time and be able to login to the account which was registered once. Already registered users will not face this stage. For this process user have to provide their basic details and verify them.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.2 Sets Location

FR-2	Both user have to set their location		
Description	In the setting of Fast Service, customer and Service provider may need to set location. It would help to find near location of service provider, besides customer can find his location. Our system will provide him/her a near location available service provider.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.3 Customer Search a Service

FR-3	Customer Search a Service or select category		
Description	Customer can search any type of serviceman what he/she looking for. And also can hire a serviceman by category.		
Stakeholders	Customer	**Priority**	High

4.1.4 Show Service list

FR-4	Show Service list		
Description	Here customer will be able to select a serviceman using category and find the required serviceman directly using profile and reviews		
Stakeholders	Customer	**Priority**	High

4.1.5 Post a problem

FR-5	Post a problem		
Description	Customer can also post their problem to the system and waiting for response of service man and hire them.		
Stakeholders	Customer	**Priority**	High

4.1.6 Emergency post

FR-6	Emergency post		
Description	Customer can add a service emergency tag which means this service require serviceman immediately. For this service customer have to pay some extra money to Fast-Service.		
Stakeholders	Customer, Service Provider	**Priority**	Medium

4.1.7 Schedule post

FR-7	Schedule post		
Description	Customer can set a starting time to the job. The customer can book a serviceman advance for future use.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.8 Sent work request

FR-8	Sent work request		
Description	A customer can sent a work request directly to the service provider		
Stakeholders	Customer, Service Provider	**Priority**	Medium

4.1.9 Cancel work request

FR-9	Cancel work request		
Description	Customer and service provider can cancel their confirmed work request with valid reason. Who cancel the work he maybe have to pay some money.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.10 Provide rating

FR-10	Provide rating and review		
Description	A customer have to give a rating and review to the service man based on his work. And reviews will be add on service provider's profile. New customer select a service man based on their rating.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.11 Ability to update profile

FR-11	Able to update profile if needed		
Description	Both customer and Service provider can see profile of each other and can able to own profile information if required.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.12 User should be filter search

FR-12	User should be filter search based on ratings and reviews		
Description	Customer can filter his search based on salesman rating, review and other criteria		
Stakeholders	Customer	**Priority**	High

4.1.13 User should be able to communicate serviceman using System

FR-13	User should be able to communicate service using system		
Description	Customer and serviceman both can communicate each other Phone calls , chatting, voice call etc.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.14 User should be able to view history and upcoming service list

FR-14	User should be able to view their service history and upcoming list		
Description	Customer and serviceman both can see their own service history. Only serviceman can see their upcoming service list		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.15 User should be able to request multiple service at same time

FR-15	User should be able to request multiple service at a time		
Description	Here customer able to request multiple service at a time. Other side, serviceman can't directly get multiple request. In proceeding one service he only can accept scheduling tasks.		
Stakeholders	Customer, Service Provider	**Priority**	Medium

4.1.16 User should be notify about any update of their work

FR-16	User should be notify about any update of their work		
Description	Both customer and serviceman able to notify about any updates like serviceman arrival time of their System. The app also notify about upcoming work before start the service.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.17 User should be able to see real time location

FR-17	User should be able to see real time location		
Description	Customer can see live location of service man when serviceman accept request. Service man also can show the location of customer house.		
Stakeholders	Customer, Service Provider	**Priority**	Medium

4.1.18 User should be able to see estimated arrival time of serviceman

FR-18	User should be able to see estimated arrival time of serviceman		
Description	The customer can able to see the estimated service man time based on their location they also can live track serviceman's location during deal.		
Stakeholders	Customer	**Priority**	Medium

4.1.19 User should be able to access system using multiple device

FR-19	User should be able to access system using multiple device		
Description	Both Customer and service man have to login their system in multiple device. The security is username and password		
Stakeholders	Customer, Service Provider	**Priority**	Low

4.1.20 Serviceman should be able to view reviews and reply them

FR-20	Serviceman should be able to see the review and reply them		
Description	Serviceman should be able to see the reviews and ratings by customer. Also he will be able to reply them		
Stakeholders	Service Provider	**Priority**	High

4.1.21 Customer should be able to modify his review

FR-21	Customer should be able to modify his review		
Description	Customer can edit or modify review and rating which will be provided by him. Also a modify tag will be appeared on his review and it will be sent a notification to service provider		
Stakeholders	Customer	**Priority**	Low

4.1.22 Serviceman should be able to change his availability status

FR-22	Serviceman should be able to change his availability status		
Description	Serviceman can change his active status to available and unavailable. This thing will be reduce misunderstanding.		
Stakeholders	Service Provider	**Priority**	High

4.1.23 Serviceman should be able to set his service area

FR-23	Serviceman should be able to set his service area		
Description	All serviceman are assigned with a specific service area according to his location. He can do little bit modify on his service area.		
Stakeholders	Service Provider	**Priority**	Low

4.1.24 User can submit report against illegal activities

FR-24	User can submit report against illegal activities		
Description	Both Customer and service provider can submit report against illegal activities or fake profiles.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.1.25 User should be able to direct communicate with system authority

FR-25	User should be able to direct communicate with system authority		
Description	Customer and serviceman both can able to direct contract with system authorities using chat or email. System authority will interact with user throw support person and AI based chat bots.		
Stakeholders	Customer, Service Provider, System author	**Priority**	High

4.2 Data Requirement

In our system we need store all personal data like name, NID, location. So we have to need ensure data confidentiality. To provide confidentiality we use end to end encryption technique. But can be arise a problem with that, it require more time to encrypt and decrypt process. So it will make system slow. To resolve this problem we will field level encryption. For example, we will not encrypt whole database. We encrypt only confidential field of data like NID, Phone number, location etc.

Several requirements for Data are:

- The system shall input, process and output data types of integers, strings, characters and real due to the variation of information that will be stored
- The system shall have accurate and up-to date information
- The system shall handle information from various sources

Firstly, we need build a data model. We need to focus on entity of data, quantity of data, capacity of data resource, data availability etc. For collecting data, we are following these steps:

Data Object Selection: A data object is a representation of information which has different properties or attributes that must be understood by software. Firstly we find all Attributes of our System.

Data Objects and Attributes: we find necessary attribute for each Entity.

Relation between Data object: Now we define all relation between two entity and it's attributes.

E-R Diagram: Entity relationship diagram displays the relationships of entity set stored in a database.

Schema Diagram: Using E-R diagram, we draw Schema of our database

Create table: Now create table and set data types it's attribute.

Normalization: Normalization is the process of reorganizing data in a database so that it meets two basic requirements:

1. There is no redundancy of data (all data is stored in only one place), and
2. Data dependencies are logical

At last, we do Normalization.

4.2.1 Data Encryption

DR-1	Field level end to end data encryption technique		
Description	Both service providers data and customers data will be encrypted for ensure security. But reducing drawback we encrypt only confidential field of data like NID, phone number, location.		
Stakeholders	Customer, Service Provider, System Author	**Priority**	High

4.3 Performance Requirement

It is important to maintain the performance of the software system. To ensure performance we maintain these steps:

4.3.1 Respond System in weak network

PR-1	System will response in weak network		
Description	Customer can interact service with weak network. And provide low latency. It also provide low loading and refresh time.		
Stakeholders	Customer, Service Provider, System Author	**Priority**	High

4.3.2 System will be run in low configuration devices

PR-2	System will be run in low configuration devices		
Description	User can able to use this app in a lower configuration device. It consume low CPU and memory. So, user can be use this system in an entry level smartphone.		
Stakeholders	Customer, Service Provider, System Author	**Priority**	High

4.3.3 Speed & Latency Requirements

PR-3	Search must be faster and loading time will be minimize		
Description	When customer search for service, the search result must show within one second. Also load system very fast. And system will response in weak network		
Stakeholders	Customer, Service Provider	**Priority**	High

4.3.4 Precision & Accuracy Requirements

PR-4	Search Result must be Accurate		
Description	When Customer search for service, the result must be According to input value given by customer.		
Stakeholders	Customer	**Priority**	High

4.4 Capacity Requirement

This system can load up to hounded thousands of customer's, service provider's information and advertisements information. Also it will be able to manage them efficient manner.

4.5 Safety Critical Requirement

A "Safety Critical Requirement" refers to a requirement that is essential for ensuring the safety of users or the public. In Fast Service no any "Safety Critical Requirements". However, user safety and privacy should be little bit considered.

4.6 Dependability Requirements

The systems have the ability to deliver the service when it's requested, the ability to deliver the service as specified, the ability to operate without catastrophic failure, the ability to protect itself against accidental intrusion, the ability of to resist or recover from damaging event. So the term Dependability measures as five dimensions such as

- Availability
- Reliability
- Safety
- Security
- Resilience

4.6.1 Reliability & Availability Requirements:

DR-1	The system must be Available on 24X7		
Description	The system must be available 24 hours, every day in a week.		
Stakeholders	Customer, Service Provider	**Priority**	High

4.7 Robustness or Fault-Tolerance Requirements

Fast service is a critical component of the home maintenance and repair industry, providing a convenient platform for customer to connect with skilled serviceman. To ensure that the service is reliable, secure, and meets the needs of its users, it is essential that the system is robust and fault-tolerant. This section outlines the requirements that the system must meet to ensure its robustness and fault-tolerance, including system availability, data backup and recovery, error handling, security, scalability, and performance.

4.7.1 Data Backup and Recovery

RR-1	System will able to Backup Data and Recover them		
Description	Provisions for backing up and recovering user data in the event of a system failure.		
Stakeholders	Customer, Service Provider, System Author	**Priority**	High

4.7.2 Error Handling

RR-2	System will able to Handling Errors		
Description	Strategies for handling errors and exceptions, and for logging and reporting problems to the development team.		
Stakeholders	Customer, Service Provider, System Author	**Priority**	High

4.8 Maintainability and Supportability

Both maintainability and supportability are crucial requirements for the success and long-term viability of the Fast service. By ensuring that the platform is both maintainable and supportable, the company can minimize downtime and ensure that the system remains relevant and effective over time.

Maintainability refers to the ability of the system to be easily maintained, modified and updated to meet changing requirements over time.

4.8.1 Maintenance Requirements

MR-1	Make the code maintainable.		
Description	Code must be developed so that it can be modified or readable by other developer.		
Stakeholders	System Author	**Priority**	High

4.9 Supportability Requirements

Supportability refers to the ability of the system to provide adequate support to users and administrators during its life cycle. In the context of a Fast service, this would refer to the availability of support to customers in the form of a help desk, chat bots or online forums, and to administrators in the form of documentation and troubleshooting guides.

This system meets Testability, Maintainability, Compatibility, Configurability, Serviceability, and install ability which are related to supportability requirements.

4.10 Security Requirements

Securing information is much more important for a system to get users dependability. Here are some of them:

4.10.1 User authentication

The system should require users (customers and service providers) to create an account and log in to use the service. Passwords should be encrypted and stored securely.

4.10.2 Data privacy

The system should protect sensitive user information, such as personal details and NID, phone number, from unauthorized access and theft.

4.10.3 Secure communication

All data transmitted between the user's device and the server should be encrypted to prevent eavesdropping and tampering.

4.10.4 Incident Response

The system should have a plan in place to respond to security incidents, such as data breaches, and be able to quickly restore normal operations.

4.10.5 Regular Security Audits

Regular security audits should be conducted to identify and fix vulnerabilities in the system.

4.10.6 Compliance

The system should comply with relevant security and privacy regulations, such as the General Data Protection Regulation (GDPR).

4.11 Usability and Human Integrity Requirements

Usability refers to the ease with which users can interact with and utilize the system. In the context of Fast service, this would refer to the ease with which customers can search for and book serviceman, and the simplicity of the platform's interface.

Human Integrity refers to the protection of user data and privacy. In the context of Fast service, this would refer to the measures in place to ensure that customer information is protected and not misused or abused. This could include measures such as secure data storage, encryption of sensitive information, and strict access controls.

4.11.1 Ease of Use Requirements

Our system will be easier to use by any type of people and they don’t need any training to use the system.

4.11.2 Accessibility Requirements

To get access to the application, the application provides authorization/authentication. This application uses various modules.

SR-1	The system provides security strategies.
Description	The system is designed in a way that allows all modules to access a mechanism that provides security services.

4.12 Look and Feel Requirements

The Look and Feel requirement refer to the visual design and aesthetic of the platform. This includes the layout, color scheme, typography, imagery, and other visual elements that contribute to the overall appearance and user experience of the platform.

In the context of Fast service, the look and feel of the platform can have a significant impact on customer engagement and perception of the service. A well-designed platform that is visually appealing and easy to navigate can increase customer trust and confidence in the service, while a poorly designed platform can have the opposite effect.

4.12.1 Appearance Requirements

AR-1	Text color and font
Description	Our system has to be different and attractive from other existing audio players using a better look and feel.

4.12.2 Style Requirements

After keeping all contents, it is very essential to load stylesheet to the application. For mobile application like android system, extensive mark-up language or xml is used. It is to be said that we are going to develop our system at android platform.

SR-1	The appearance must be controllable using stylesheet file
Description	For android application style-sheet files are xml. So, all stylesheet must be controllable by the xml file.

4.13 Legal Requirements

Legal requirements normally refer to the terms and conditions or privacy policy of any organization. The terms and condition of our application is that, no third-party software or person is allowed to use our data for their business purpose also we ensure data encryption.

5. Requirement Engineering Process

Requirements Engineering (RE) determines software requirements according to customer requirements or needs. Requirements engineering process includes –

- Requirements elicitation
- Needs modeling
- Requirements analysis
- Requirements assurance & validation
- Requirements management.

5.1 Requirement Elicitation Techniques

Requirements elicitation is the practice of researching and finding system requirements for users, customers, and other stakeholders, also referred to as "Requirement Gathering". Requirement elicitation can be done by contacting participants directly or by doing some research, analysis and testing.

5.1.1 Hold Elicitation Interviews

Stakeholder interviews are a vital part of the requirement engineering process for Fast Service. One-on-one interviews with customers and service providers are an efficient way to gather requirements and understand their specific needs. This information can then be used to resolve conflicts more effectively during group workshops. By conducting individual interviews beforehand, the time spent in group workshops is reduced and allows for a focus on resolving any remaining conflicts. We mainly perform our interview based on some specific criteria.

- Short description about Fast Service
- Inform about use of personal information
- Dealing with customer
- Review System
- Payment System

5.1.2 Perform Document Analysis

Existing documentation can help to show how systems are currently operating or what they are what I should do. Documents include written information about current programs, business processes, needs specifications, and competitor research. Review once textual analysis can help determine which performance should remain and functionality that isn't in use. By reviewing and analyzing these documents, it is possible to determine the necessary functionality that should be maintained, as well as any unused functionality.

5.1.3 System Interface Analysis

The initial step in System Interface Analysis is to determine the systems with which the new system will need to communicate. This can involve various types of systems, such as servers on the internet, software on the same host, hardware devices, or other systems with different functions. To ensure effective communication between the new system and other systems, it is crucial to accurately identify the systems that will be involved and to understand their functions and requirements.

5.1.4 Distribute Questionnaires

The questionnaire is an effective tool for gathering information on styles, changes in attitudes and preferences, and user satisfaction. To minimize fatigue or frustration for the respondent, our questions were kept concise and grouped together based on topics. This allowed the respondent to focus on specific areas and provided a clear rationale for each question. The main advantage of using this survey approach was the ability to collect responses in a standard manner, allowing for the consolidation of information from a large number of people.
We use two separate set of questionnaires for this process-

- For Customer
- For Serviceman

5.2 Requirement Validation

Requirement validation plays a critical role in ensuring that the requirements are accurate, complete, and aligned with the desired quality standards for the program. Although the requirements may have initially appeared to be well-defined, upon further review and implementation, ambiguities and gaps were discovered. To ensure the requirements meet the necessary quality standards, it is important to validate them throughout the development process to identify and address any issues before they can impact the success of the program.

5.2.1 Review the Requirements

The process of negative peer review, particularly the thorough evaluation method, is a hallmark of the highest-quality software development processes. Our team of reviewers, representing diverse perspectives, thoroughly evaluated written requirements, analysis models, and information related to disability to ensure that the software meets the highest standards. This type of rigorous review helps to identify any weaknesses or areas for improvement in the software development process.

5.2.2 Test the Requirements

We test constitute an alternative view of the requirements. We also conduct writing tests about how to tell if the expected functionality was correctly implemented. Derive tests from the user requirements to document the expected behavior of the product under specified conditions.

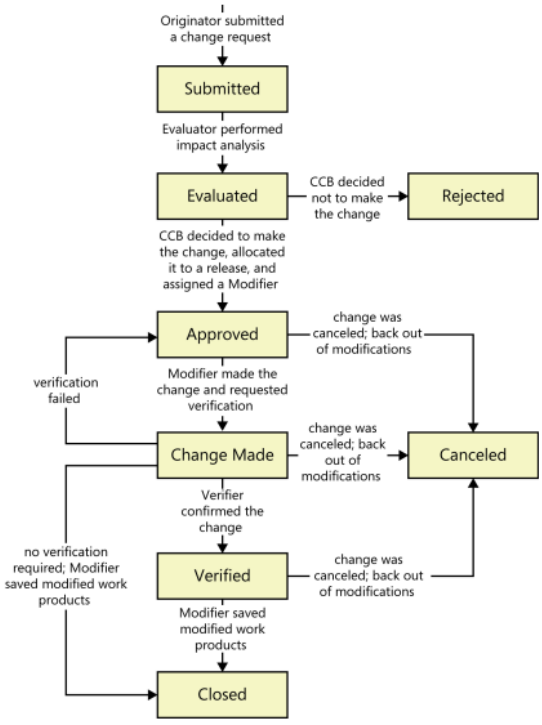
5.2.3 Simulate the requirements

To stimulate requirements, trading tools are available that we have used to simulate a proposed system in place or to add details of written requirements. The simulation takes prototyping to the next level. It providing a more interactive and comprehensive understanding of the system being developed.

5.3 Requirement change management

To effectively manage changes, it is important to have a system in place for handling change requests and keeping track of open issues. In a contract-development scenario, change management practices are crucial in controlling scope creep and ensuring that changes are made in a controlled and organized manner.
Our change management process includes the following steps:

- Evaluate and prioritize defect corrections and enhancement requests
- Dynamically adjust the scope of future releases or iterations
- Evaluate the impact of proposed changes on users and business processes
- Participate in making change decisions



6. Use Case Diagram

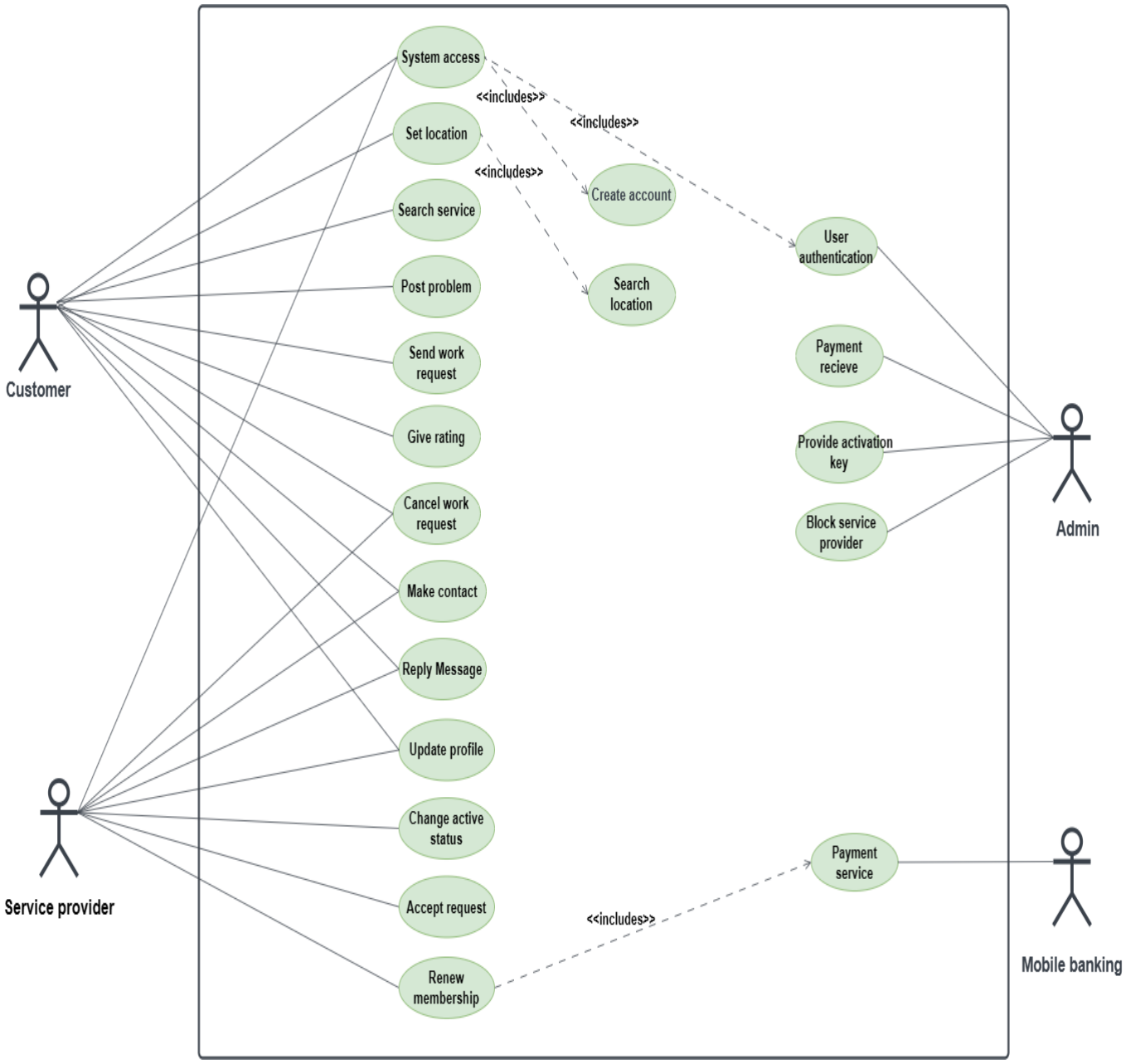


Figure 1: Use case Diagram

7. Use Case Description

Table 1: System access

Use Case	System access

Goal	User wants to get access into the “Fast Service” application.	
Preconditions	N/A	
Success End Condition	A user gets access.	
Failed End Condition	User can’t get access.	
Primary Actors: **Secondary Actors:**	Customer, Service provider Admin	
Trigger	User opens the application.	
Main Success Flows	**Step**	**Action**
	1	User opens the “Fast Service” application.
	2	User clicks “login" button
	3	User provides Username and password
	4	User clicks “Login” button
	5	Server checks information and provides confirmation.
	6	Login successful
Alternative Flows	4a	User clicks create “Sign up” button.
	4a1	User provides username, email, password, Profile Picture, payment information, Location.
	4a2	User clicks “Sign up” button
	4a3	Server saves information
	4a4	Signup complete
Quality Requirements	**Step**	**Requirement**
	1	The system should be able to create a new user account and log the user in within 5 seconds.

Table 2: Set location

Use Case	Set location	
Goal	The customer will set his location so that a service provider can find his location.	
Preconditions	Goto update profile	
Success End Condition	User sets his location on application.	
Failed End Condition	Application shows error during set location.	
Primary Actors: **Secondary Actors:**	Customer, Service provider N/A	
Trigger	The “Set location” needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	Click “Edit” button beside the location.
	2	Set location
	3	Click “Save” button to save location
Alternative Flows	**Step**	**Branching Action**
	N/A	N/A
Quality Requirements	**Step**	**Requirement**
	3	The system should respond to user login attempts within 3 seconds.

Table 3: Search service

Use Case	Search service	
Goal	The customer wants to search for a service.	
Preconditions	User must log into the service	
Success End Condition	The customer selected a service.	
Failed End Condition	The customer doesn't find his/her desired service.	
Primary Actors: **Secondary Actors:**	Customer N/A	
Trigger	"Search box" needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	The customer clicks the search box.
	2	The customer provides a search keyword.
	3	User's desired category is shown.
Alternative Flows	**Step**	**Branching Action**
	3a	The customer finds different result
	3a1	Search by proper keyword
Quality Requirements	**Step**	**Requirement**
	1	Show results within 5 seconds.

Table 4: Post problem

Use Case	Post problem	
Goal	The customer wants to post his/her problem on application.	
Preconditions	User must have an account	
Success End Condition	The customer can post his problem.	
Failed End Condition	The customer can't post a problem	
Primary Actors: **Secondary Actors:**	Customer N/A	
Trigger	Click "Post" button.	
Main Success Flows	**Step**	**Action**
	1	The customer clicks on "Post" button.
	2	Customer writes his/her problem statement into text box.
	3.	Customer adds photos or videos with the post
	4.	Click "Post" button to make a post.
Alternative Flows	**Step**	**Branching Action**
	N/A	N/A
Quality Requirements	**Step**	**Requirement**
	1	Server should be capable to publish post within 10 seconds.

Table 5: Send work request

Use Case	Send work request	
Goal	The customer wants to send a work request to a service provider.	
Preconditions	User must have an account Search a service man.	
Success End Condition	Customer request sent successfully.	
Failed End Condition	Application system fails to send work request to the service man.	
Primary Actors: **Secondary Actors:**	Customer Admin	
Trigger	User clicks “Send request”.	
Main Success Flows	**Step**	**Action**
	1	Select service man
	2	The customer writes a proper statement about his/her problem.
	3	The customer clicks “Send request” button and sends a request to the service provider.
Alternative Flows	**Step**	**Branching Action**
	N/A	N/A
Quality Requirements	**Step**	**Requirement**
	3	Service man will receive work request within 5 seconds

Table 6.1: Cancel work request (for customer)

Use Case	Cancel work request	
Goal	The customer wants to cancel work request after giving a request to a service provider	
Preconditions	User must have an account. The customer sent work request to a service man.	
Success End Condition	The customer successfully cancelled his/her work request.	
Failed End Condition	The service provider has already accepted the request.	
Primary Actors: **Secondary Actors:**	Customer N/A	
Trigger	User clicks on “Request unsent” button.	
Main Success Flows	**Step**	**Action**
	1	The customer clicks “Request unsent” to cancel work request.
	2	Unsent request successfully
Alternative Flows	**Step**	**Branching Action**
	1a	Contact with service provider to cancel work request
Quality Requirements	**Step**	**Requirement**
	N/A	N/A

Table 6.2: Cancel work request (for service provider)

Use Case	Cancel work request	
Goal	The service provider wants to cancel work requests.	
Preconditions	The service provider is requested for a service by a customer.	
Success End Condition	The service man successfully cancelled his/her work request.	
Failed End Condition	The customer has already unsent the request.	
Primary Actors: **Secondary Actors:**	The serviceman N/A	
Trigger	User clicks on “Request unsent” button.	
Main Success Flows	**Step**	**Action**
	1	The service provider makes phone call to cancel work request.
	2	Cancelled request successfully
Alternative Flows	**Step**	**Branching Action**
	N/A	N/A
Quality Requirements	**Step**	**Requirement**
	N/A	N/A

Table 7: Give review

Use Case	Give review	
Goal <	Customer can give review to the service.	
Preconditions	A job needs to be assigned and work should be done.	
Success End Condition	Customer can successfully give review	
Failed End Condition	Customer can't give review.	
Primary Actors: **Secondary Actors:**	Customer N/A	
Trigger	Clicks "Review" button	
Main Success Flows	**Step**	**Action**
	1	Customer go to history
	2	See Running work
	3	Select running work
	4	Click "Done" button make as complete
	5	Then Click "Review" button
	6	Open review interface and provide review.
Alternative Flows	**Step**	**Branching Action**
	N/A	N/A
Quality Requirements	**Step**	**Requirement**
	1	Review should be added to service provider's profile within 5 seconds

Table 8: Update a review

Use Case	Give review	
Goal	Customer can update his existing review	
Preconditions	Already given review	
Success End Condition	Review will be updated, and notification sent to service provider.	
Failed End Condition	Review isn't updated	
Primary Actors: **Secondary Actors:**	Customer N/A	
Trigger	Update review button	
Main Success Flows	**Step**	**Action**
	1	Customer go to review
	2	Click update review button
	3	Update review
	4	Click "Update" Button
Alternative Flows	**Step**	**Branching Action**
	1	N/A
Quality Requirements	**Step**	**Requirement**
	1	N/A

Table 1.1: Make contact (for customer)

Use Case	Make contact	
Goal	Customer wants to contact with the serviceman.	
Preconditions	Go to serviceman profile.	
Success End Condition	Customer contacts with the service man.	
Failed End Condition	Application shows error and customer can't send massage to service man.	
Primary Actors: **Secondary Actors:**	Customer, Service man N/A	
Trigger	Click "Contact" option to start conversation.	
Main Success Flows	**Step**	**Action**
	1	Click "Contact" button.
	2	Write a message.
	3	Click "Send" button.
Alternative Flows	**Step**	**Branching Action**
	2a.	Make phone call using "Call" icon
Quality Requirements	**Step**	**Requirement**
	1	System should provide user friendly interface for messaging feature.

Table 2.2: Make contact (for Serviceman)

Use Case	Make contact	
Goal	The serviceman wants to contact with the customer.	
Preconditions	Receive notification of work request	
Success End Condition	The serviceman contacts successfully with the service man.	
Failed End Condition	Application shows error and serviceman can't send massage to customer.	
Primary Actors: **Secondary Actors:**	Customer, Service man N/A	
Trigger	Click on customer's post.	
Main Success Flows	**Step**	**Action**
	1	Click on customer's post.
	2	Write a message.
	3	Click "Send" button.
Alternative Flows	**Step**	**Branching Action**
	2a.	Make phone call
Quality Requirements	**Step**	**Requirement**
	1	System should provide user friendly interface for messaging feature.

Table 3: Reply message

Use Case	Reply message	
Goal	Customer wants to reply message to service provider	
Preconditions	Post problem	
Success End Condition	Customer can be able to communicate with serviceman	
Failed End Condition	Can't be able to contact with serviceman	
Primary Actors: **Secondary Actors:**	Customer N/A	
Trigger	Click on notification	
Main Success Flows	**Step**	**Action**
	1	Get notification
	2	Click on notification
	3	See message from serviceman
	4	Reply the message
	5	Click on send button
Quality Requirements	**Step**	**Requirement**
	N/A	N/A

Table 4: update profile

Use Case	update profile	
Goal	User wants to update profile	
Preconditions	User must have a account	
Success End Condition	Update profile is completed successfully	
Failed End Condition	Profile not updated	
Primary Actors: **Secondary Actors:**	Customer N/A	
Trigger	User clicks “update profile” button	
Main Success Flows	**Step**	**Action**
	1	Customer goes to profile
	2	Click on “Edit” button
	3	Customer update profile
	4	Update is successful.
Alternative Flows	**Step**	**Branching Action**
	N/A	N/A
Quality Requirements	**Step**	**Requirement**
	N/A	N/A

Table 5: Active status

Use Case	Active status	
Goal	Change active status	
Preconditions	N/A	
Success End Condition	Profile status activated	
Failed End Condition	Profile status not activated	
Primary Actors: **Secondary Actors:**	Customer, Service provider N/A	
Trigger	Click on “Active status” to change active status.	
Main Success Flows	**Step**	**Action**
	1	Users go to his/her profile.
	2	Click the “Active status” button.
	3	Changed active status
Alternative Flows	**Step**	**Branching Action**
	N/A	N/A
Quality Requirements	**Step**	**Requirement**
	N/A	N/A

Table 6: Accept request

Use Case	Accept request	
Goal	The serviceman wants to accept customer's request	
Preconditions	Customer must request to serviceman.	
Success End Condition	The serviceman successfully accepted customer's request.	
Failed End Condition	The request is not accepted.	
Primary Actors: **Secondary Actors:**	Serviceman N\A	
Trigger	Click on notification.	
Main Success Flows	**Step**	**Action**
	1	Click on notification
	2	Click on customer post
	3	Conversation with customer to make a deal.
	4	Request accepted.
Alternative Flows	**Step**	**Branching Action**
	3.a	Couldn't make a deal
	4a.	Request rejected
Quality Requirements	**Step**	**Requirement**
	N/A	N/A

Table 7: Renew membership

Use Case	Renew membership	
Goal	Serviceman wants to renew the membership	
Preconditions	Serviceman login the profile	
Success End Condition	Serviceman renews his/her membership successfully	
Failed End Condition	Renew membership is not completed	
Primary Actors: **Secondary Actors:**	serviceman Admin	
Trigger	"Renew membership" button needs to be clicked.	
Main Success Flows	**Step**	**Action**
	1	Serviceman clicks in "renew membership" button
	2	Select payment method
	3	Make payment
	4	Payment successful
Alternative Flows	**Step**	**Branching Action**
	3a.	Payment failed for Insufficient balance
Quality Requirements	**Step**	**Requirement**
	3a.	System notifies payment confirmation within 10 seconds.

8. Activity Diagram

1. System access (Create Account, Login)

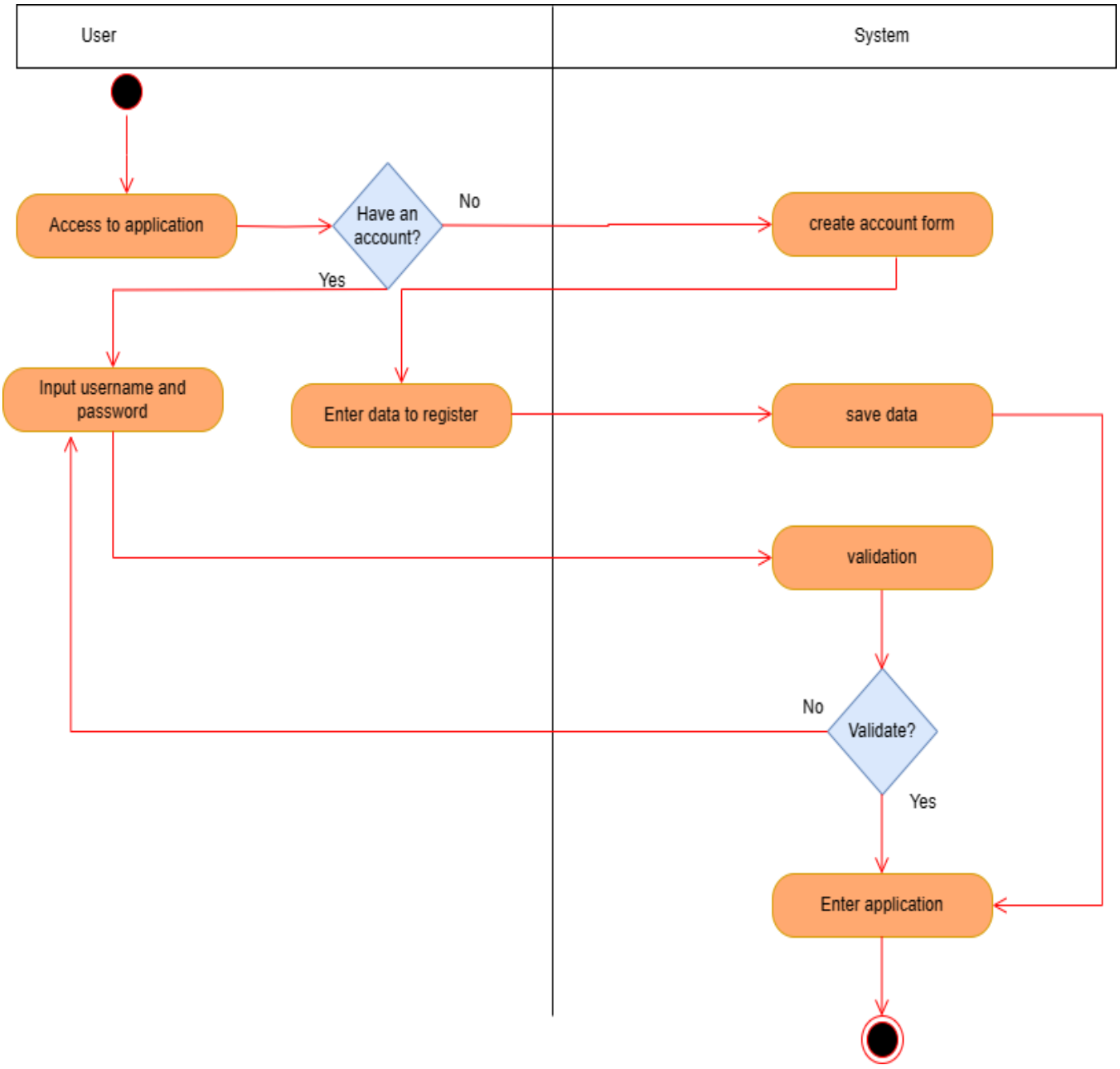


Figure 1: System Access (Create account, Login)

2.Activity Diagram (Set location):

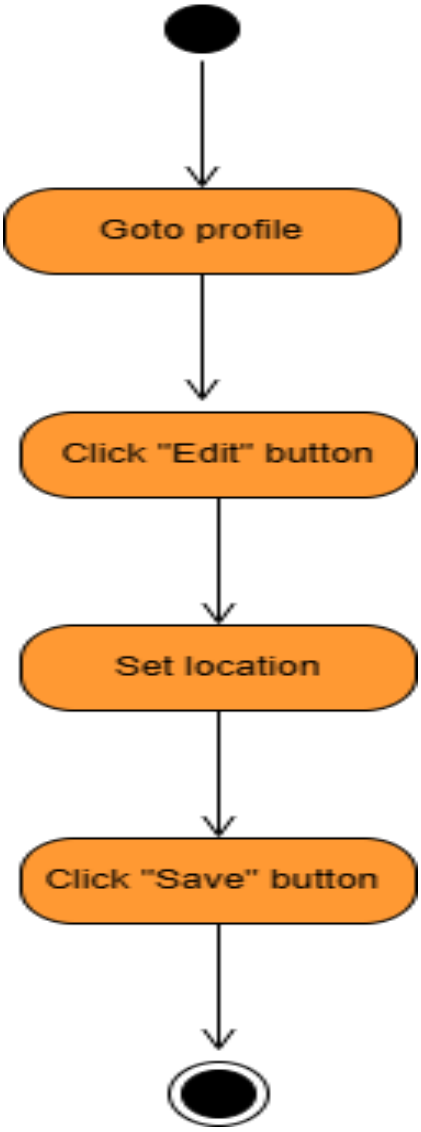


Figure 3: Set location

Activity Diagram (Search service):

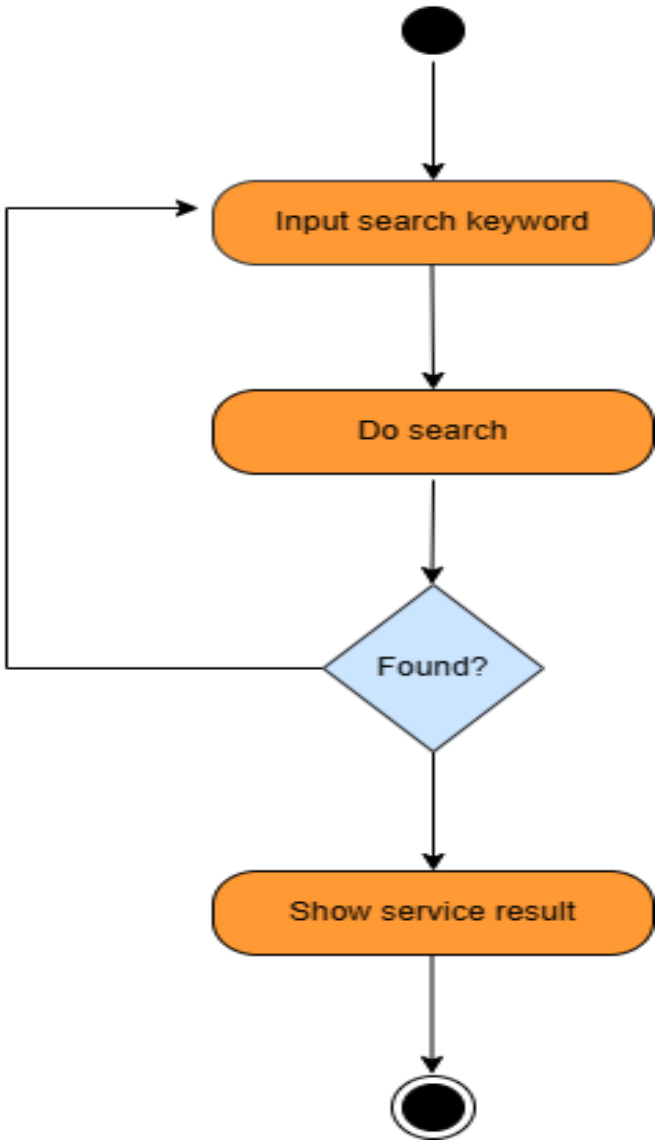


Figure 4: Search service

Activity Diagram (Post problem):

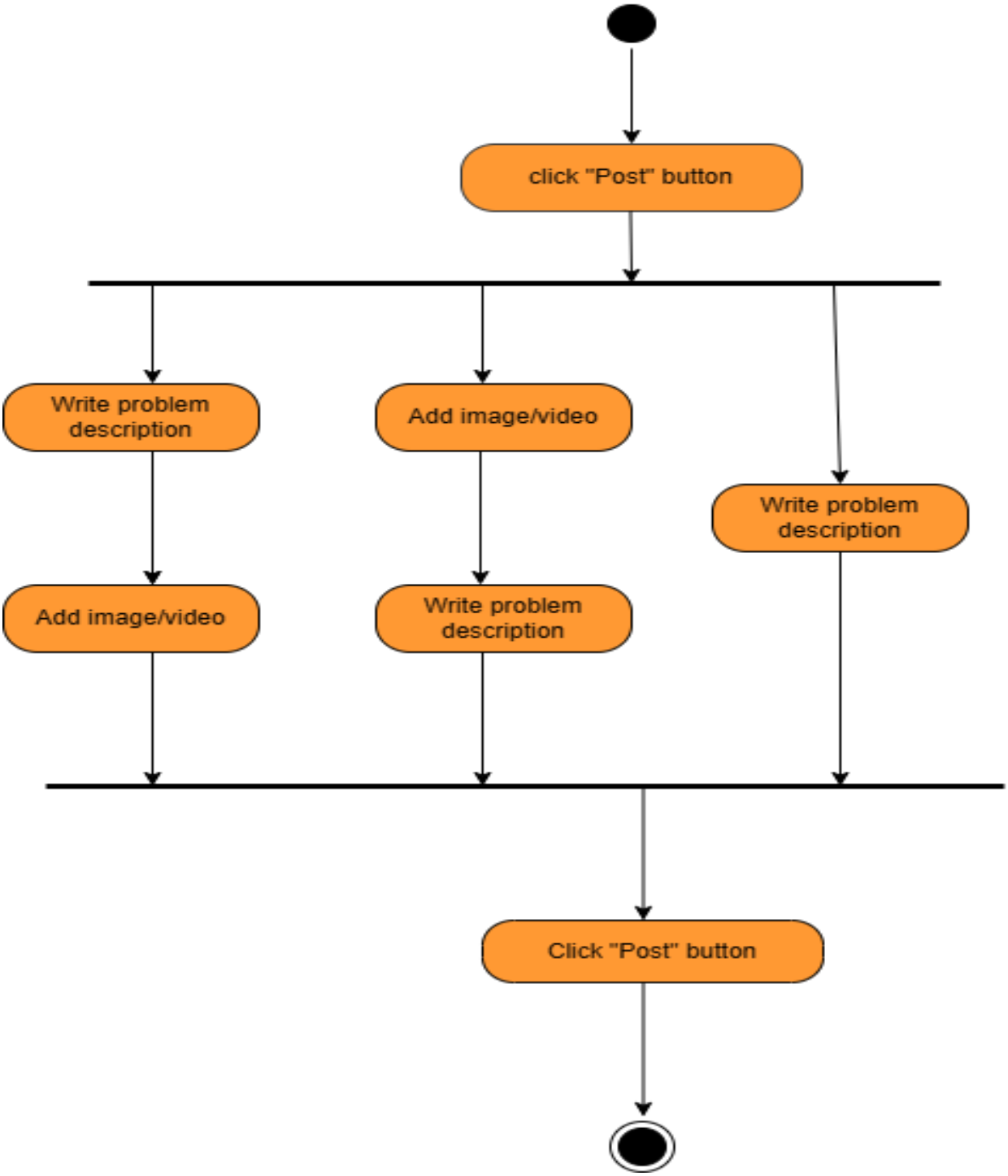


Figure 5: Post problem

Activity Diagram (Send work request)

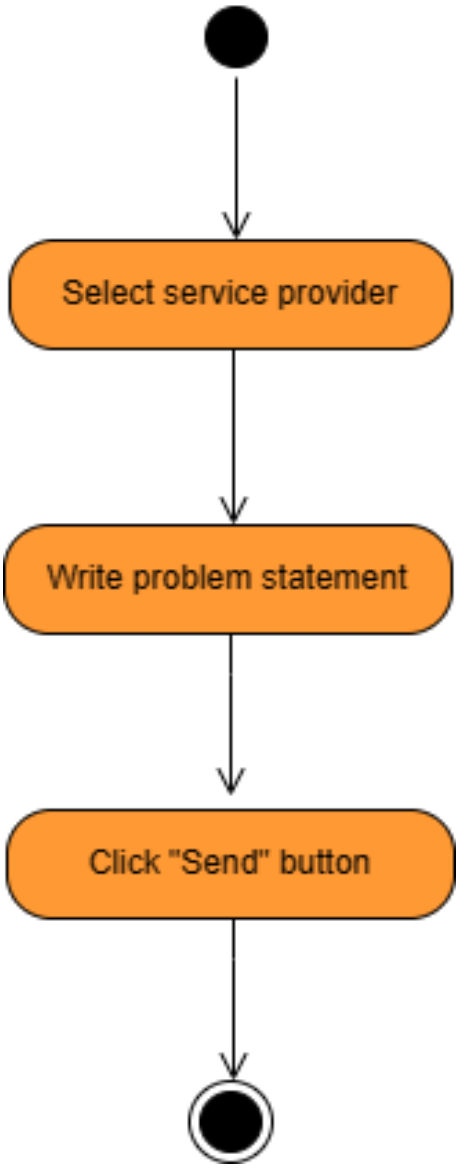


Figure 6: Send work request

Activity Diagram (Cancel work request):

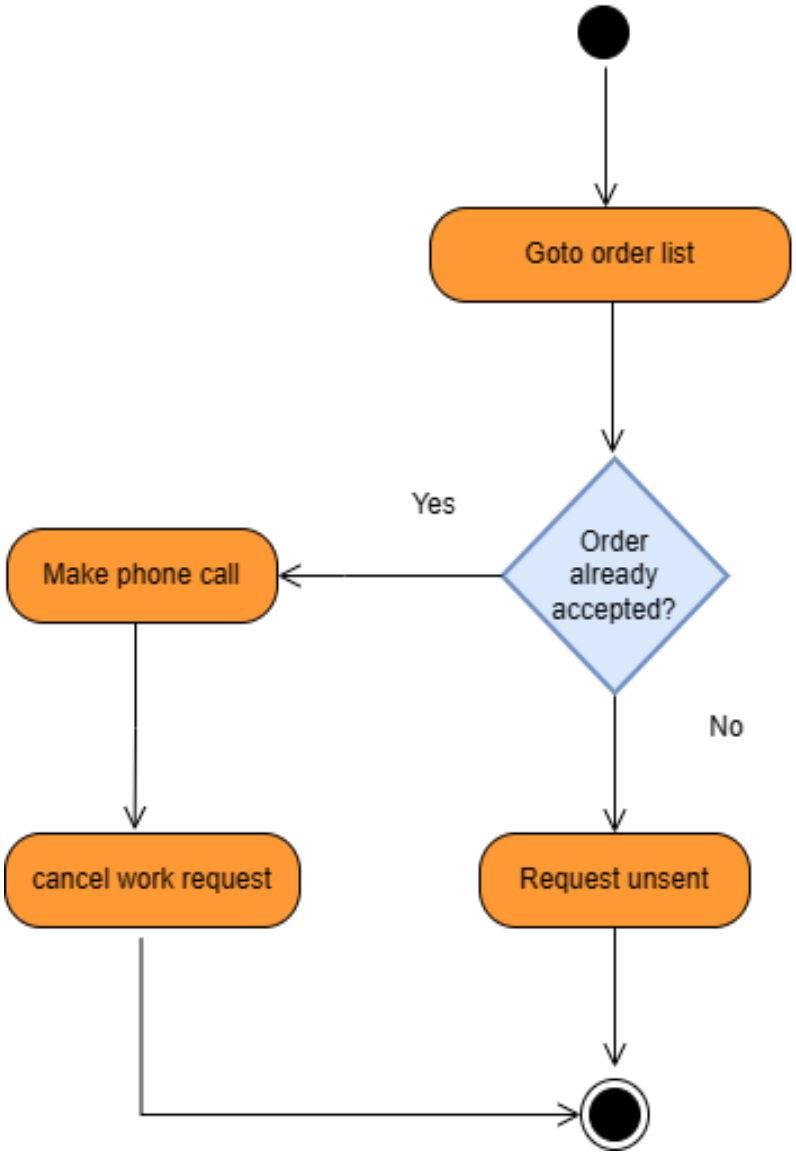


Figure 7: Cancel work request (for customer)

Activity Diagram (Give review):

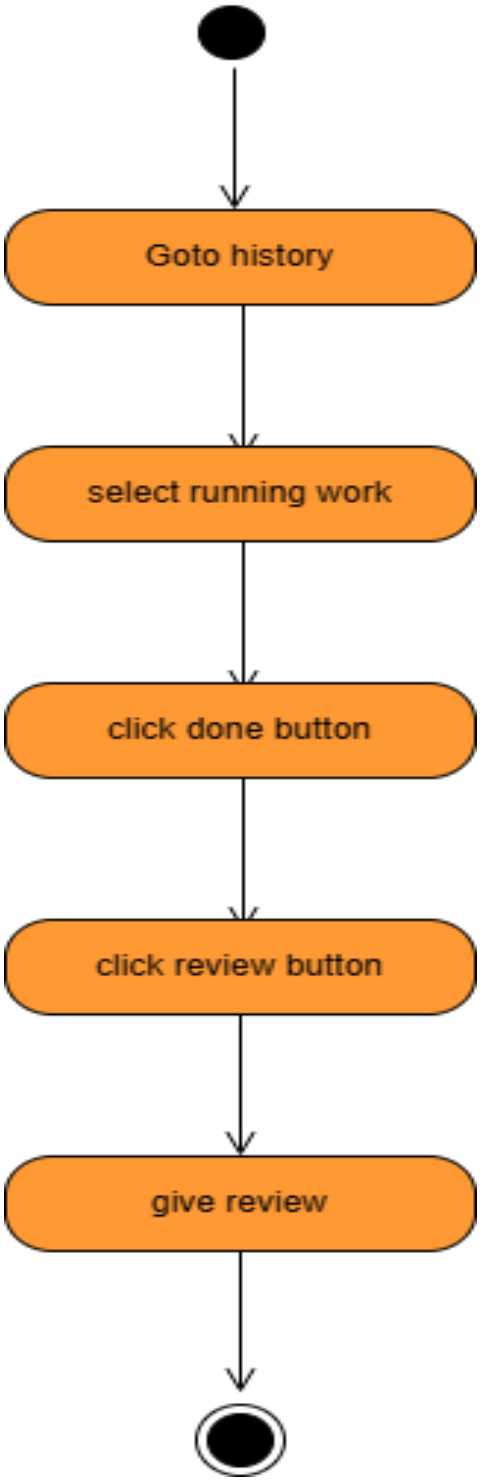


Figure 8: Give review

Activity Diagram (Update review):

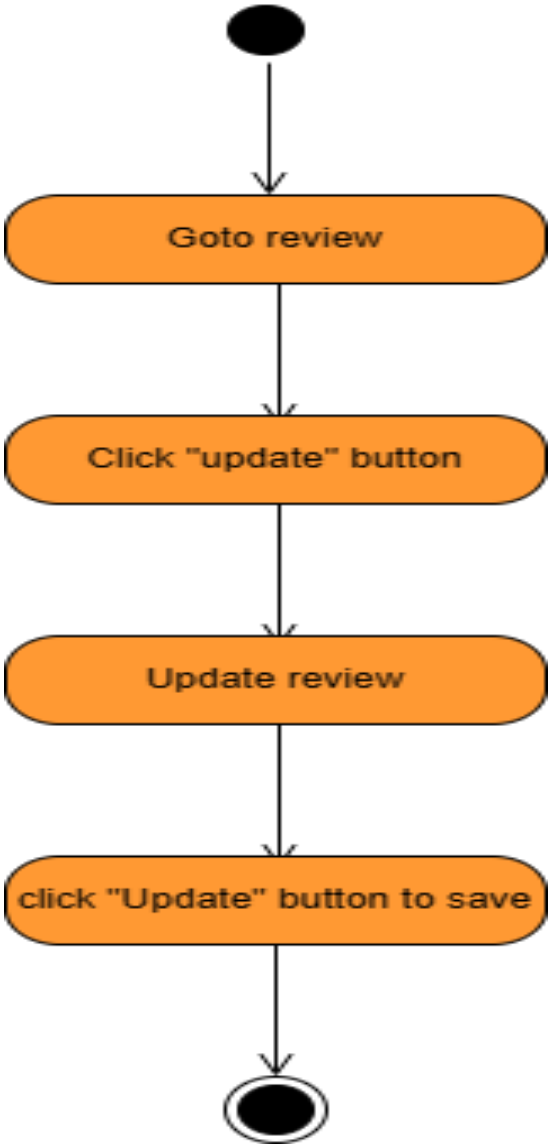


Figure 9: Update review

Activity Diagram (Make contact):

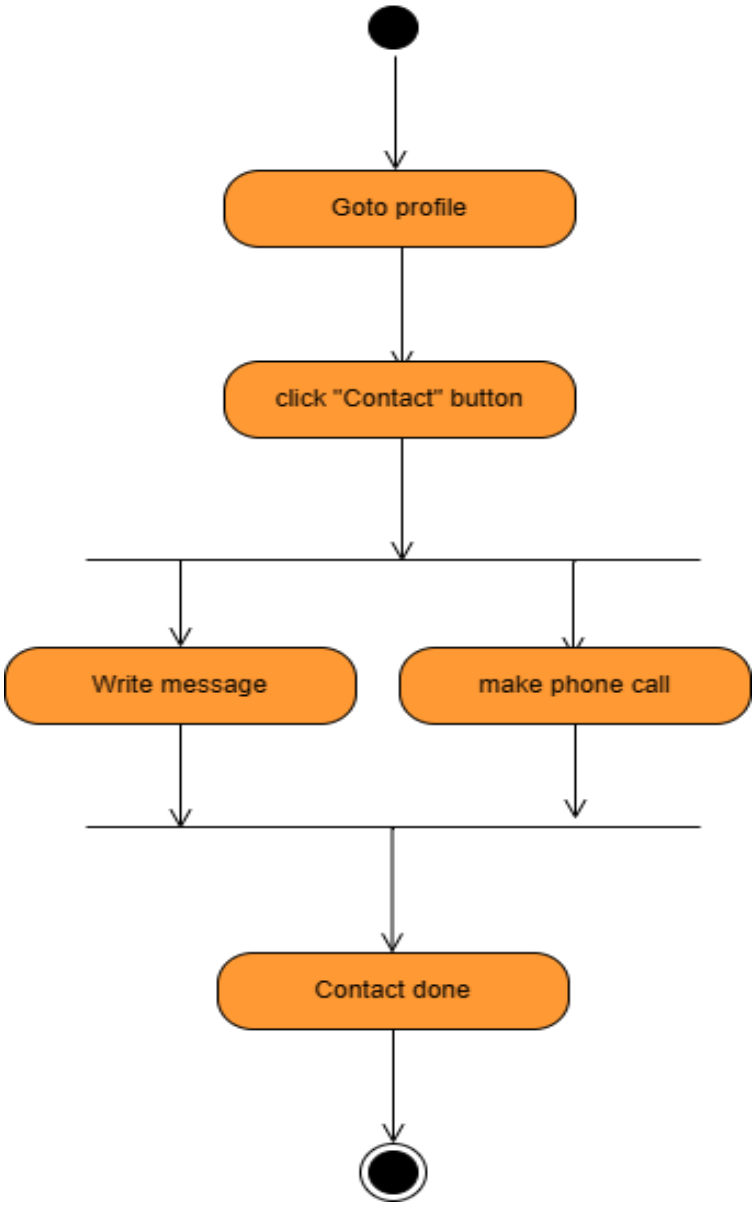


Figure 10: Make contact (for customer)

Activity Diagram (Reply message):

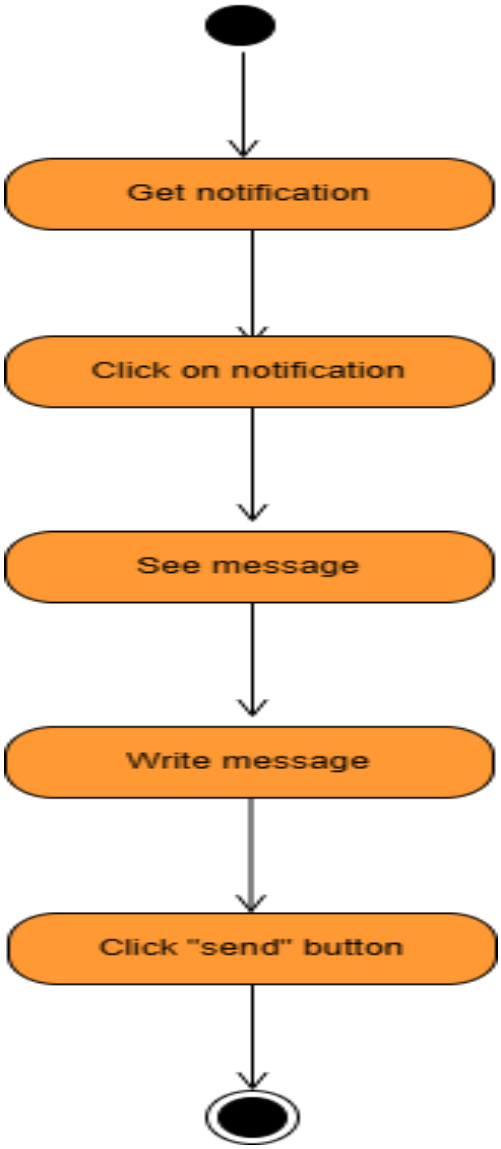


Figure 11: Get notification

Activity Diagram (Update profile):

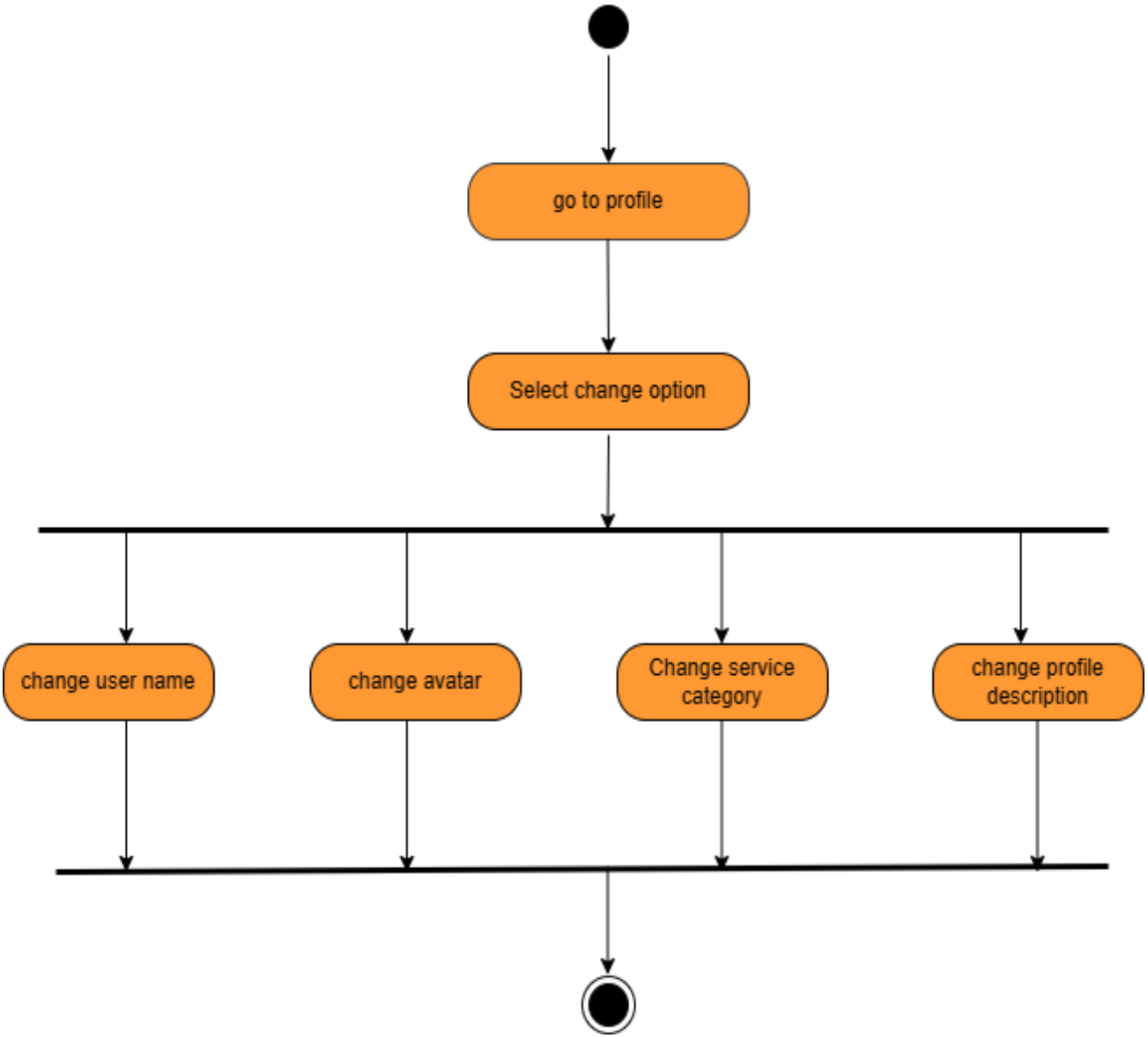


Figure 12: Update profile

Activity Diagram (Active status)

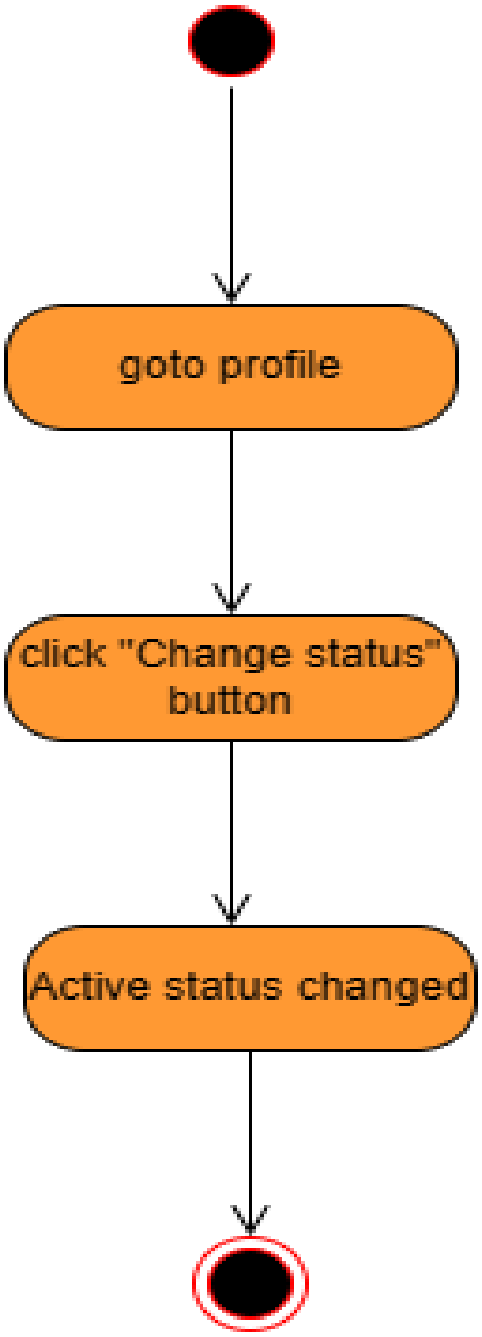


Figure 13: Active status

Activity Diagram (Accept request):

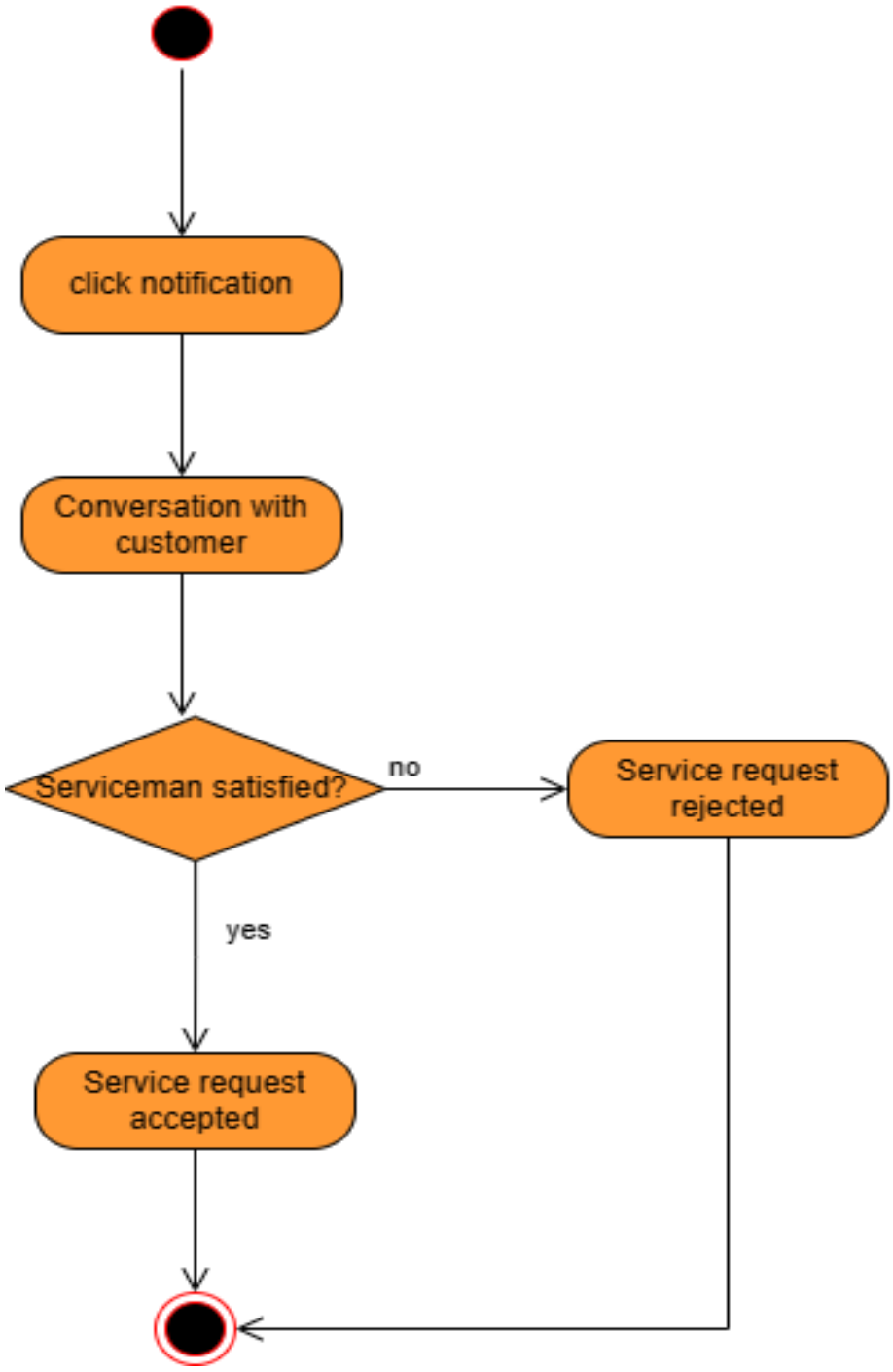


Figure 14: Accept request

Activity Diagram (Renew membership):

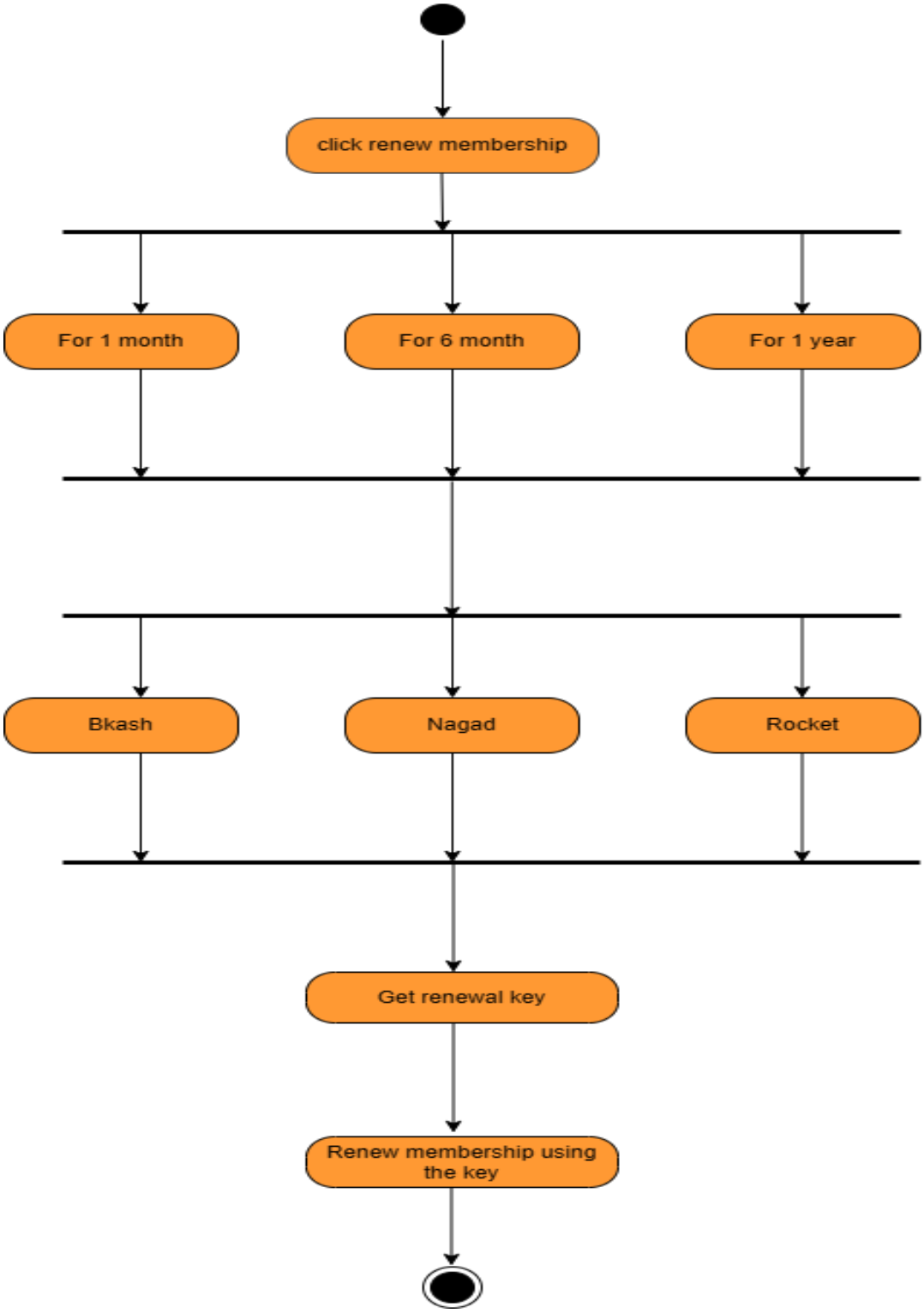


Figure 15: Renew membership

Activity Diagram (Log out)

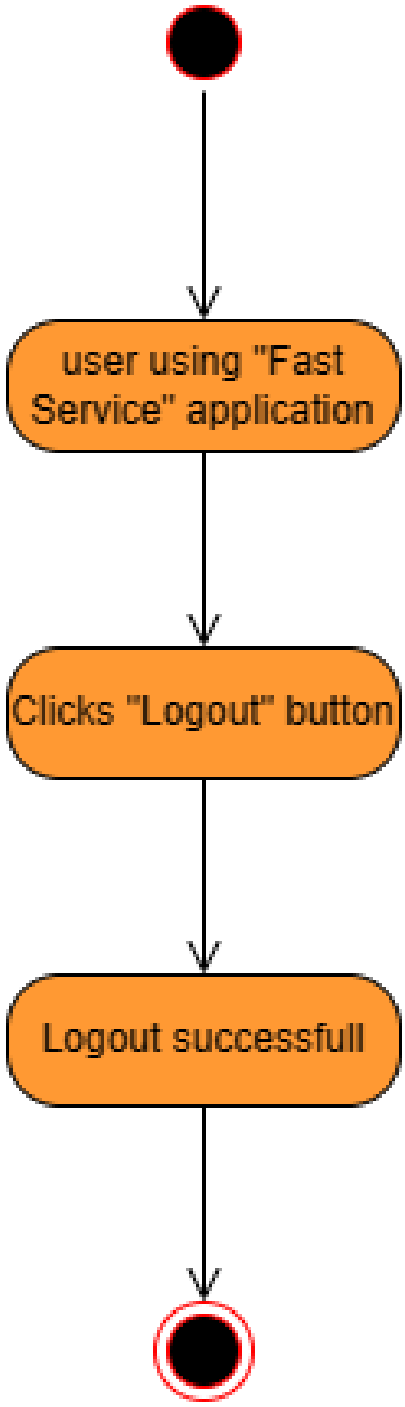


Figure 16: Log out

9. Requirement Traceability Matrix

Traceability Matrix

A Traceability Matrix is a document that co-relates any two-baseline documents that require a many-to-many relationship to check the completeness of the relationship. It is used to track the requirements and to check the current project requirements are met.

The Requirement Traceability Matrix (RTM)

It is a critical document in software development that links and tracks user requirements with corresponding test cases. It serves as a one-stop-shop for capturing all client requirements and their traceability throughout the software development life cycle. The main purpose of RTM is to validate that all requirements have been thoroughly tested and no functionality goes unchecked during the testing phase.

Why RTM is Important?

RTM is important for several reasons. Firstly, it helps the testing team to understand the client's requirements and ensure that the output product is defect-free. By thoroughly understanding the requirements, testers can create positive and negative test cases, which are further split into scenarios and test cases. RTM provides a way to track the testing of each requirement and scenario to ensure that no requirement is left untested.

Furthermore, RTM helps to eliminate any confusion regarding the requirements, their corresponding test cases, and their current status. It also serves as a record of the testing activities done for a specific product, allowing the testing team to monitor their progress and identify any gaps. RTM is a critical tool that ensures the completeness of the relationship between the requirements and test cases. It helps the testing team to deliver a high-quality product that meets the client's requirements and provides a clear record of the testing activities throughout the software development life cycle.

RTM also helps to increase the traceability of changes to the requirements during the development process. It enables the development and testing teams to quickly identify the impact of any changes and take necessary actions to ensure that the requirements continue to be met. This helps to prevent scope creep and ensures that the product is delivered on time and within budget.

Parameters for Requirement Traceability Matrix

- Requirement ID
- Requirement Type and Description
- Test Cases with Status

A requirement traceability matrix can be

- Show the requirement coverage in the number of test cases
- Design status as well as execution status for the specific test case
- If there is any User Acceptance test to be done by the users, then UAT status can also be captured in the same matrix.
- The related defects and the current state can also be mentioned in the same matrix.

Use Case

Use Case No	Use Case Name
UC1	Create Account
UC2	Search Location
UC3	Set Location
UC4	Search Service
UC5	Show Service List
UC6	Post Problem
UC7	Send Work Request
UC8	Cancel Work Request
UC9	Give Rating
UC10	Make Contact
UC11	See notification
UC12	Update Profile
UC13	Login
UC14	Add Basic Details
UC15	Verify

UC16	Update Avatar
UC17	Emergency Post
UC18	Bound Service Time
UC19	Change Active Status
UC20	Accept Work Request
UC21	User Access Control
UC22	Payment
UC24	User Access Control
UC25	Block Service Provider

Test Case

Test Case No	Test Case Name
TC1	Verify if user able to create an account
TC2	Hospital Road
TC3	Sonapur
TC4	Verify if user able to set his location
TC5	Electric
TC6	Plumbing
TC7	Verify if customer able to show service list
TC8	Verify if customer able to post a problem
TC9	Verify if customer able to send work request

TC10	Verify if customer and service provider able to cancel work request
TC11	Verify if customer able to Give Rating
TC12	Verify if service man able to contract with customer
TC13	Verify if customer and service man able to see notification
TC14	Verify if user able to Contract with system
TC15	Verify if user can Update Profile
TC16	Verify if user can enter the system
TC17	Verify if user can provide his information
TC18	Verify if system verify user information
TC19	Verify if user can Update Avatar
TC20	Verify if customer can make an emergency post
TC21	Verify if customer able to post a problem with time limit
TC22	ON
TC23	OFF
TC24	Accepted
TC25	Decline
TC26	Verify if user can get access of the system
TC27	Verify if customer able to block a service provider

Requirements Traceability Matrix

Project Name	**Fast Service**	**Business Area**			**Noakhali**		
Project Manager	**Md. Nasim Molla**	**Business Analyst Lead**			**Hasanur Rahman Shishir,** **Md. Mahbub Hasan Talukdar**		
QA Lead	**Md. Zahid Hasan,** **Khos. Mahmuda Akter Mumu**	**Target Implementation Date**					
Category/ Functional Activity	**Requirement Description**	**Use Case Reference**	**Design Document Reference**	**Code Module/ Reference**	**Test Case Reference**	**User Acceptance Validation**	**Comments**
FR1	Both customer and service provider have to create account	UC1,UC14,UC15,UC16			TC1	Verified	
FR2	Both user have to set their location	UC2,UC3			TC2,TC3	Pass	
FR3	Customer Search a Service or select category	UC4			TC5,TC6	Pass	
FR4	Show Service list	UC5			TC7	Verified	
FR5	Post a problem	UC6			TC8	Verified	
FR6	Emergency post	UC6,UC17			TC20	Verified	

FR7	Schedule post	UC6,UC18			TC21	Verified	
FR8	Sent work request	UC7			TC9	Verified	
FR9	Cancel work request	UC8			TC10	Verified	
FR10	Provide rating and review	UC9			TC11	Verified	
FR11	Able to update profile if needed	UC12			TC15	Verified	
FR13	User should be able to communicate service using system	UC10			TC14	Verified	
FR16	User should be notify about any update of their work	UC11			TC13	Verified	
FR22	Serviceman should be able to change his availability status	UC19			TC22,TC23	Pass	
FR24	User can submit report against illegal activities	UC25			TC27	Verified	

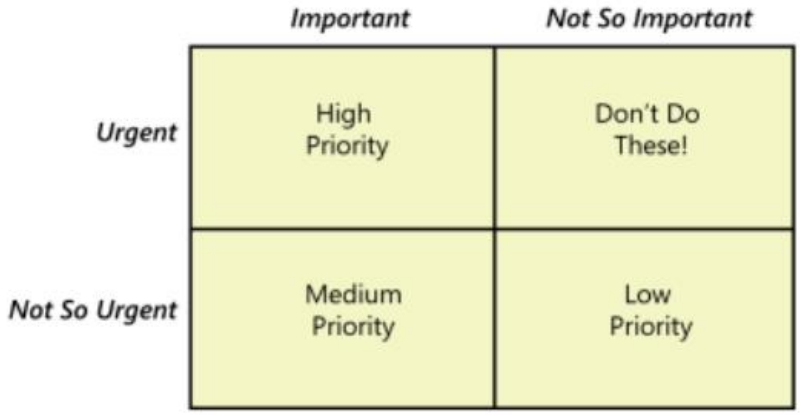
11. Appendix

10.1 Prioritization of requirements

We've prioritized the functional requirements by following Three-level Scale technique.

10.1.1 Three-level Scale

When a Business Analyst categorizes the requirements in any of the ordering or ranking scale, it is subject to the analyst's understanding of the business. Many analysts suggest that this method has some drawbacks and advocate methods that have more than one scale.



The priority of the different sections in the diagram is indicated by the numbering system. The highest priority is given to important items, while the lowest priority is given to less urgent items.

High Priority: These requirements are critical and cannot be ignored. They pertain to compliance or contract obligations and must be addressed in the current release. Failure to implement these requirements can have negative consequences for the business.

Medium Priority: These requirements are important but not as pressing as high priority items. They should be addressed after completing the high priority items. Within this category, items on the right side of the dividing line are of higher priority.

Don't Do These: These items are less important but still urgent. They should be addressed after completing the higher priority medium priority items. The items on the right side of the dividing line within this category have a higher priority.

Low Priority: These items are neither important nor urgent and can be addressed at a later time, after completing the items in the first three categories.

The priority order should be followed by starting with the bottom-right corner of the high-priority section and working up and to the left.

10.1.2 Prioritization of the requirements of Fast Service

FR1 – High priority: Both house owner and service provider have to create account. This is a essential requirement of our system.

FR2 – High priority: Both user have to set their location. To use our system user have to set the location.

FR3 – Medium priority: Customer Search a Service or select category. Customer can search any type of serviceman what he/she looking for. And also can hire a serviceman by category. To get a service customer can use alternation way like make a post or direct contact approach. So it is medium priority.

FR4 – High priority: Show Service list. Here customer will be able to select a serviceman using category and find the required serviceman directly using profile and reviews. After enter a system customer need see the offered service list.

FR5 – Medium priority: Post a problem. This requirement is one of the main feature of our system.

FR6 – Medium priority: Customer can add a service emergency tag which means this service require serviceman immediately. For this service customer have to pay some extra money to Fast-Service. It will be done after implemented FR5.

FR7 – Medium priority: Customer can set a starting time to the job. The customer can book a serviceman advance for future use.

FR8 – High priority: Sent work request. A customer can sent a work request directly to the service provider. This is also one of the main feature of our system.

FR9 – High priority: Customer and service provider can cancel their confirmed work request with valid reason. Who cancel the work he maybe have to pay some money.

FR10 – High priority: A customer have to give a rating and review to the service man based on his work. And reviews will be add on service provider's profile. New customer select a service man based on their rating. This is also an core requirement of our system.

FR11 – Medium priority: Both customer and Service provider can see profile of each other and can able to own profile information if required. This requirement important for the system not urgent.

FR12 – Low priority: User should be filter search based on ratings and reviews. Customer can filter his search based on salesman rating, review and other criteria. This feature is always don't required. Without this function the system will operate his task easily.

FR13 – High priority: User should be able to communicate service using system. Customer and serviceman both can communicate each other Phone calls, chatting, voice call etc. This is a important and urgent requirement.

FR14 – High priority: User should be able to view their service history and upcoming list. This is also an important requirement for our system.

FR15 – Medium priority: User should be able to request multiple service at a time. This is a important feature but not so urgent. So will can provide this on next update.

FR16 – High priority: User should be notify about any update of their work. This is also an important feature of our system.

FR17 – Medium priority: User should be able to see real time location. Customer can see live location of service man when serviceman accept request. Service man also can show the location of customer house. This feature can be include next update.

FR18 – Medium priority: User should be able to see estimated arrival time of serviceman. The customer can able to see the estimated service man time based on their location they also can live track serviceman's location during deal. This also can be include in next update.

FR19 – Low priority: User should be able to access system using multiple device. This feature in not mandatory for operate our service.

FR20 – High priority: Serviceman should be able to see the reviews and ratings by customer. Also he will be able to reply them.

FR21 – Medium priority: Customer can edit or modify review and rating which will be provided by him. Also a modify tag will be appeared on his review and it will be sent a notification to service provider. This feature can be add on next updated version.

FR22 – High priority: Serviceman can change his active status to available and unavailable. This thing will be reduce misunderstanding. This is a core feature of our system.

FR23 – Low priority: Serviceman should be able to set his service area. All serviceman are assigned with a specific service area according to his location. He can do little bit modify on his service area.

FR24 – Low priority: User can submit report against illegal activities. Both Customer and service provider can submit report against illegal activities or fake profiles.

FR25 – Medium priority: Customer and serviceman both can able to direct contract with system authorities using chat or email. System authority will interact with user throw support person and AI based chat bots. This can be provide in next update of system.

DR1 – High priority: Both service providers data and customers data will be encrypted for ensure security. But reducing drawback we encrypt only confidential field of data like NID, phone number, location.

PR1 – High priority: System will response in weak network. Customer can interact service with weak network. And provide low latency. It also provide low loading and refresh time.

PR2 – High priority: System will be run in low configuration devices. User can able to use this app in a lower configuration device. It consume low CPU and memory. So, user can be use this system in an entry level smartphone.

PR3 – High priority: Search must be faster and loading time will be minimize. When customer search for service, the search result must show within one second. Also load system very fast. And system will response in weak network.

PR4 – High priority: Search Result must be Accurate. When Customer search for service, the result must be According to input value given by customer.

RAR1 – Medium priority: The system must be Available on 24X7.

RR1 – High priority: System will able to Backup Data and Recover them.

RR2 – High priority: System will able to Handling Errors. Strategies for handling errors and exceptions, and for logging and reporting problems to the development team.

MR1 – High priority: Code must be developed so that it can be modified or readable by other developer.

AR1 – Low priority: The system is designed in a way that allows all modules to access a mechanism that provides security services.