**

Bike Rescue

Software Requirement Specification

TABLE OF CONTENTS

[1 Introduction](#_Toc374529923)

[1.1 Purpose](#_Toc374529924)

[1.2 Scope](#_Toc374529925)

[1.3 Definitions, Acronyms, and Abbreviations](#_Toc374529926)

[1.4 References](#_Toc374529927)

[1.5 Overview](#_Toc374529928)

[2 Overall Description](#_Toc374529929)

[3 FUNCTIONAL Requirements](#_Toc374529930)

[3.1 <The use cases>](#_Toc374529932)

[3.2 <Use Cases Diagram>](#_Toc374529933)

[4 DIAGRAMS](#_Toc374529923)

[4.1 Concept Diagram](#_Toc374529924)

[4.2 Database Diagram](#_Toc374529925)

[4.3 Class Diagram](#_Toc374529926)

[4.4 Activity Diagram](#_Toc374529927)

[4.5 State Machine Diagram](#_Toc374529928)

1 Introduction

- Project name: Bike Rescue

- Product Type: mobile app

In this project, we would like to introduce an emergency rescue app when your bike has problem. It will be troublesome when you bike breaks down but there is no repair shop or people nearby to ask for assistance. And to solve those situations, we introduce you to an app on our phones called Bike Rescue.

* 1. Purpose

The purpose of this document is to give a detailed description of the requirements for the “Bike Rescue” software. It provides explanation of purpose, declaration of application, also system constraints, interfaces, interactions with other applications.

1.2 Scope

Bike Rescue app is built as a system which connect bikes repair shops and customer whose bikes have broken down within a certain range together. This mobile application is free to download from all mobile phone stores.

For the shop owners, they provide their shop fixing services to the app. These information of services will be used for search results displayed to customer. Besides customers and shop owners, there will be administrator team who will administer the system and keep information appropriate and accurate. He administrators can verify shop service or manage user information.

Bike Rescue app needs Internet and GPS connection. The information of the app will be stored in a database located on a web-server. This app also interacts with the GPS-Navigator which is on user phone to track the location of the customers need help the time their bikes broke down and also search for available shops nearby.

1.3 Definition, acronyms, and abbreviations

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Customer | Someone who interacts with the mobile phone application |
| Administrator | Who has permission for managing and controlling the application |
| Shop Owner | Someone who has bike shop to be part of the app |
| Web-Portal | A web application which present special facilities for shop owner and admin |
| GPS | Global Positioning System |
| GPS-Navigator | An installed software on mobile phone which could provide GPS connection and data, show locations on map and find paths from current position to defined destination |
| Application Store | An installed application on mobile phone which helps user to find new compatible applications with mobile phone platform and download them from Internet |

1.4 References

1.5. Overview

The remainder of this document includes three chapters and Supporting Information section. The second one provides an overview of the system functionality and system interaction with other systems. The chapter also mentions the system constraints and assumptions about the product.

The third chapter provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify the requirements more precisely for different audiences.

The fourth chapter deals with the prioritization of the requirements. It includes a motivation for the chosen prioritization methods and discusses why other alternatives were not chosen.

The Supporting Information section in the end of the document include Appendixes, all results of the requirement prioritization and a release plan based on them.

2 Overall description

This chapter will give an overview of the while system, show how the system interacts with other systems and introduce the basic functionality of it, present the constraints and assumptions for the system.

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.4 Constraints

2.6 Apportioning of requirements

3 Functional Requirements

3.1 The Use cases

3.1 Use Case: Login action

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – BR\_01 | | | |
| Use Case No. | BR\_01 | Use Case Version | 1.0 |
| Use Case Name | | Login | |
| Author | | Lâm Bảo Khang | |
| Date | 01/07/2020 | Priority | Normal |
| Actor:  \_ Unauthorized User  Summary:  \_ This use case check user login information  Goals:  \_ User will successfully login  Trigger:  \_ Click to login button  Pre-condition:  \_ User opened app, display main page  Post-condition:  \_ Display view bases on user role  Main Success Scenario   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | User click login button | Display login page | | 2 | User input username and password | Waiting | | 3 | User click submit to login | If login information username and password are correct, system display view page bases on user role  [Exception 1,2] |   Alternative  N/A  Exception   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Connection lost, internet error | Show message “Connection error. Please check your connection and login again later.” | | 2 | Username or password is null or incorrect | Show message “Your username or password is incorrect”, remain at login page |   Relationship  Business rules  \_When login action is permitted, system display  + If user is a customer: display customer view page  + If user is a shop owner: display shop owner view page  + If user is an administrator: display admin view page | | | |

3.2 Use Case: Register action

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – BR\_02 | | | |
| Use Case No. | BR\_02 | Use Case Version | 1.0 |
| Use Case Name | | Register | |
| Author | | Lâm Bảo Khang | |
| Date | 02/07/2020 | Priority | Normal |
| Actor:  \_Unauthorized user  Summary:  \_This use case check information to register new user  Goals:  \_New user will successfully register  Trigger:  \_User click register button  Pre-condition:  \_ User opened app, display main page  Post-condition:  \_System show message “Register successfully”, display login page  Main Success Scenario   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Click register action | System display register page | | 2 | Fill in register form | Waiting | | 3 | Click submit to register | System check if all information filled in the register form is valid  [Exception 1,2,3]  Show message “Register successfully”  Display login page |   Alternative  N/A  Exception   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Connection lost, internet error | Show message “Connection error. Please check your connection and login again later.” | | 2 | Null fields or invalid information | Show message “Your register information is invalid. Please check them again”, remain on register page | | 3 | Confirmed password not match password | Show message “Your confirmed password must match password”, remain on register page |   Relationship  Business rules:  \_ User must be connected to the Internet  ­\_User must have specific role (customer/shop owner/administrator) | | | |

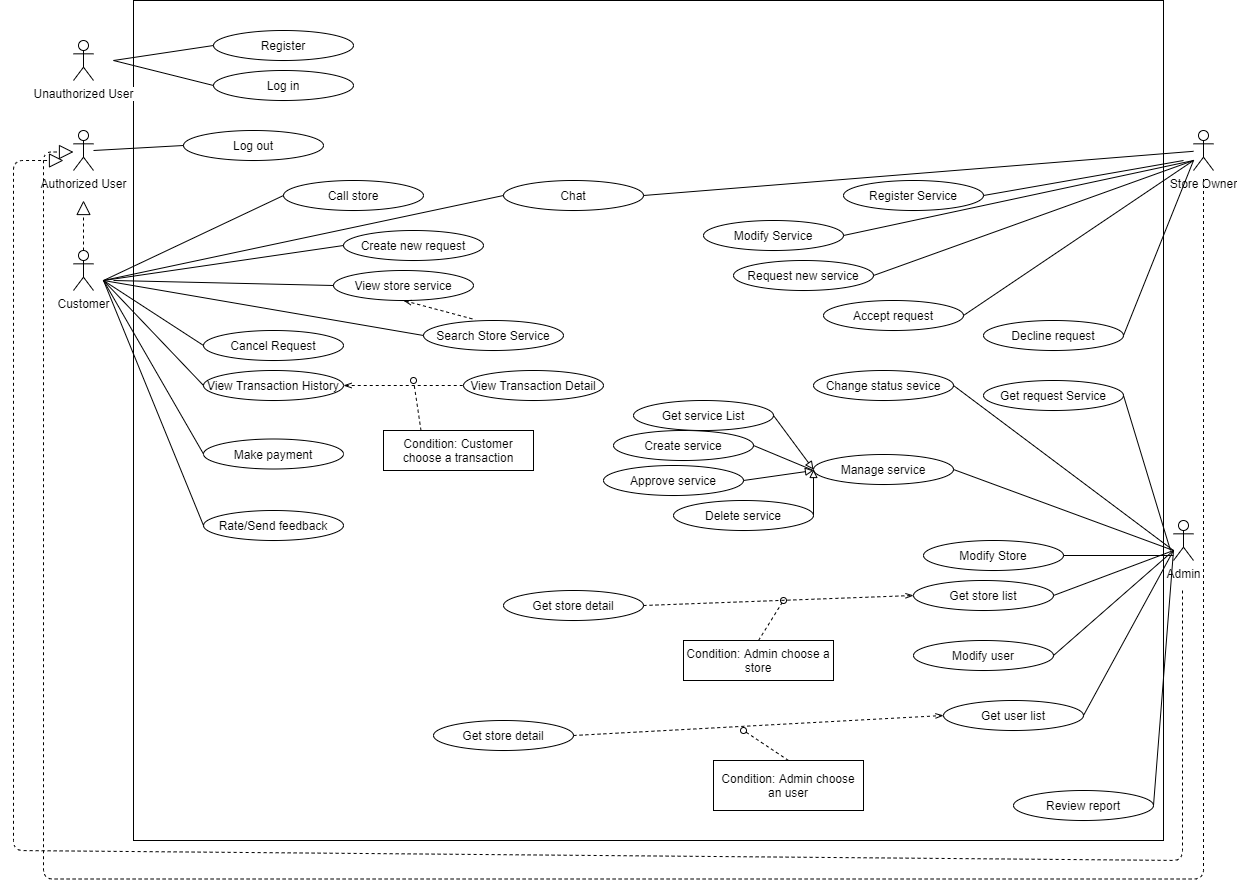
3.3 Use Case: Customer view request history

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – BR\_03 | | | |
| Use Case No. | BR\_03 | Use Case Version | 1.0 |
| Use Case Name | | View request history action | |
| Author | | Lâm Bảo Khang | |
| Date | 05/07/2020 | Priority | Normal |
| Actor:  \_Customer  Summary:  \_This use case allows customer to view request history  Goals:  \_Request History is showed  Trigger:  \_Customer clicks “view history” button  Pre-condition:  \_User login as Customer  \_Customer profile page  Post-condition:  \_Success: Show request history  \_Fall: Show error message  Main Success Scenario   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Customer click view history button | System searches all finished request in database  System show list of result  [Exception 1,2] |   Alternative  N/A  Exception   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Connection lost, internet error | Show message “Connection error. Please check your connection and login again later.” | | 2 | There is no finished request in this customer history | Show message “The request history is empty. No finished request found” |   Relationship  Business rules  \_ Search results should be showed as a list. Each result display as name and date  \_ Each page of a list displays at most 10 results  \_ The result list should be sorted by newest date  \_ The list view page should have a button for customer to filtering results by options (date, price, etc.) | | | |

3.4 Use Case: Shop owner add new service

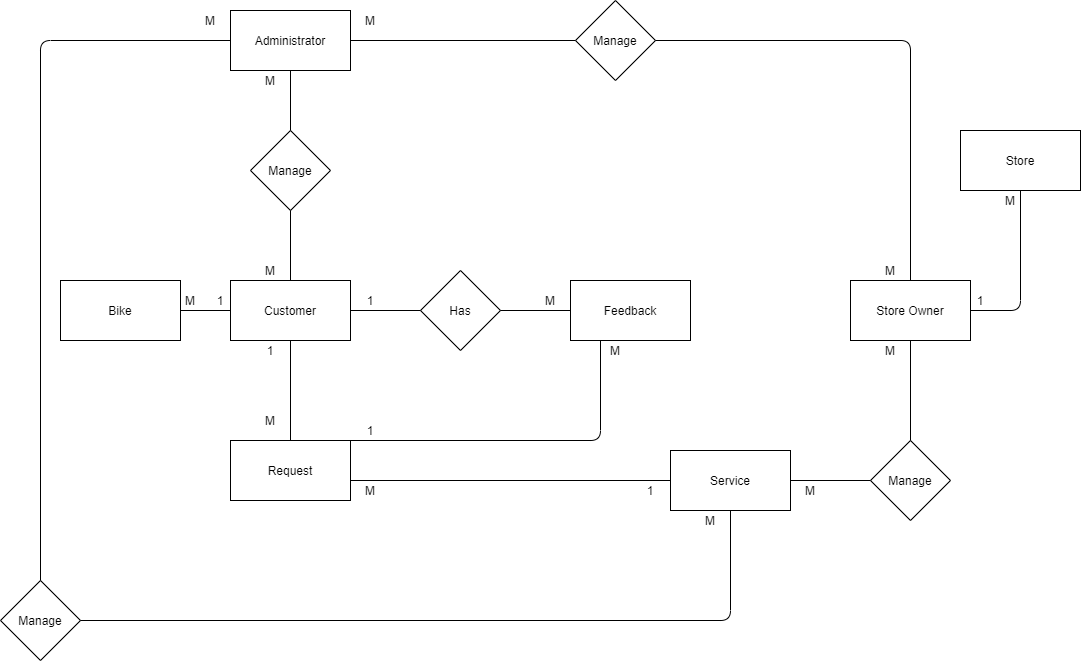
|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE – BR\_04 | | | |
| Use Case No. | BR\_04 | Use Case Version | 1.0 |
| Use Case Name | | Add new service action | |
| Author | | Lâm Bảo Khang | |
| Date | 10/07/2020 | Priority | Normal |
| Actor:  \_Shop owner  Summary:  \_This use case allows shop owner to add new service to the shop  Goals:  \_New service is added successfully  Trigger:  \_Shop owner click add new service button  Pre-condition:  \_User login as shop owner and has shop  \_At shop profile page  Post-condition:  \_Success: New service is added  \_Fall: Show error message  Main Success Scenario   |  |  |  | | --- | --- | --- | | Step | Actor Action | System Response | | 1 | Shop owner clicks add new service button | System displays add new service page | | 2 | Shop owner inputs information and submits | System checks if all information fill in the register service form is valid  [Exception 2]  Then display validate page for shop owner to input password to validate account | | 3 | Shop owner inputs password and submit | System checks if password is correct  [Exception 3]  then show message “New service is added to shop”  Systems display new service information page  [Exception 1] |   Alternative  N/A  Exception   |  |  |  | | --- | --- | --- | | No | Actor Action | System Response | | 1 | Connection lost, internet error | Show message “Connection error. Please check your connection and login again later.” | | 2 | Null fields or invalid information | Show message “Your service information is invalid. Please check them again”, remain on add new service page | | 3 | Wrong confirmed password | Show message “Your password is incorrect”, remain on validate page |   Relationship  Business rules:  \_ Shop owner must be connected to the Internet. | | | |

3.2 Use case diagram

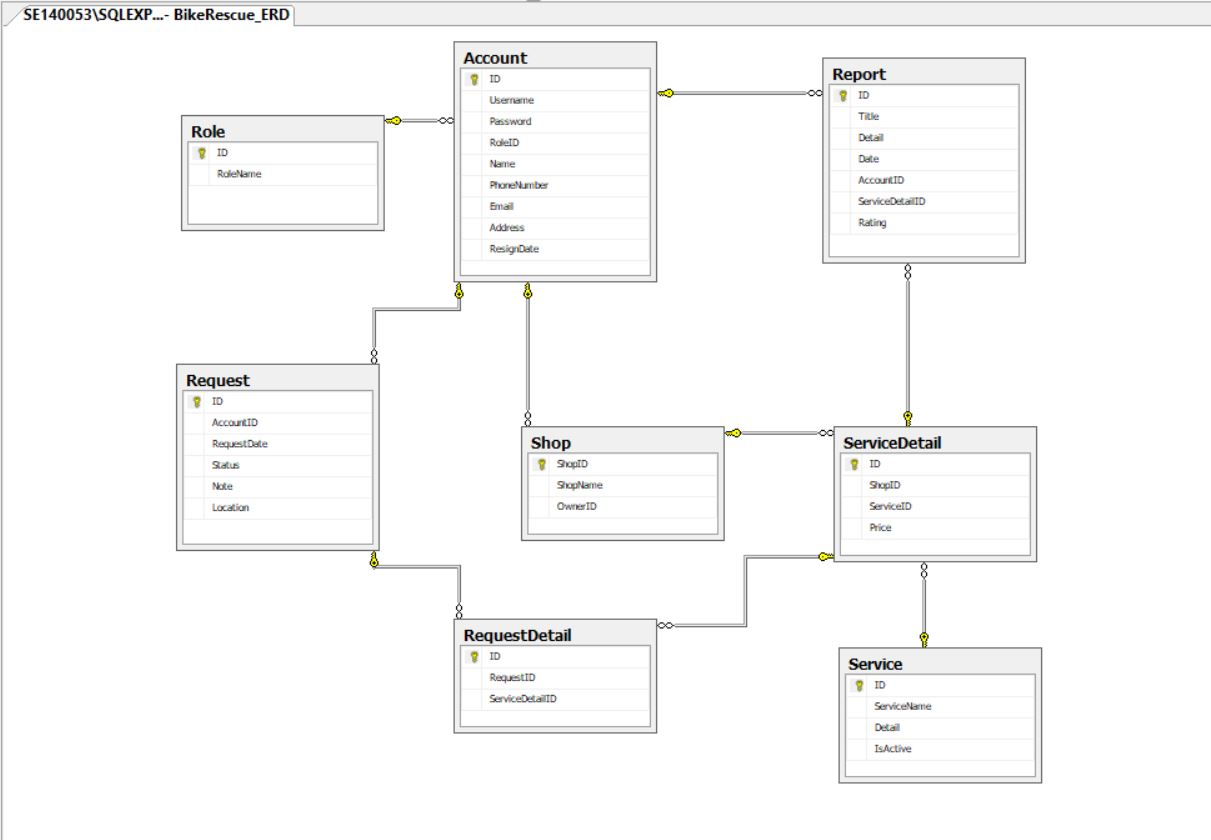


4 Diagrams

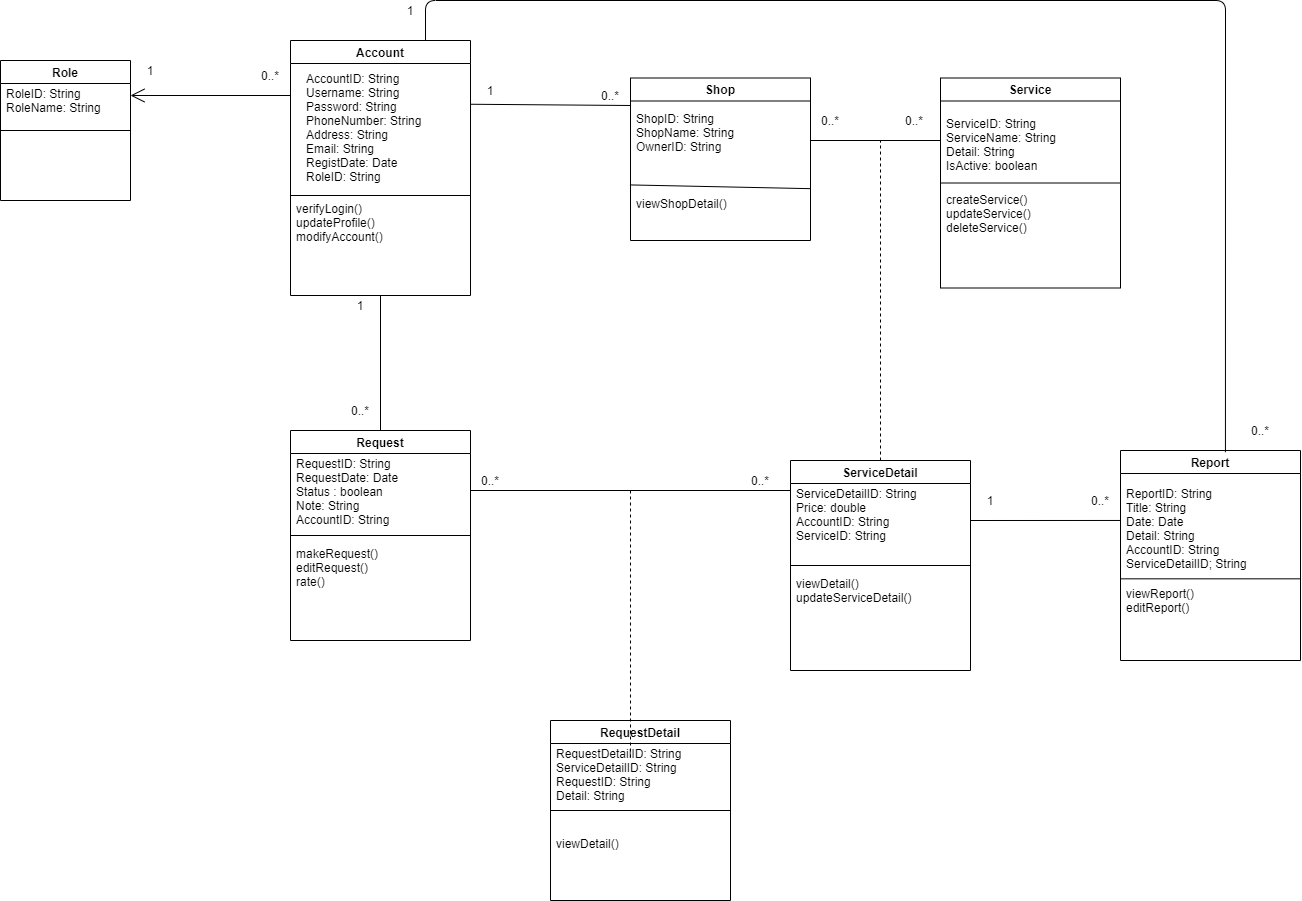
4.1 Concept Diagram



4.2 Database Diagram

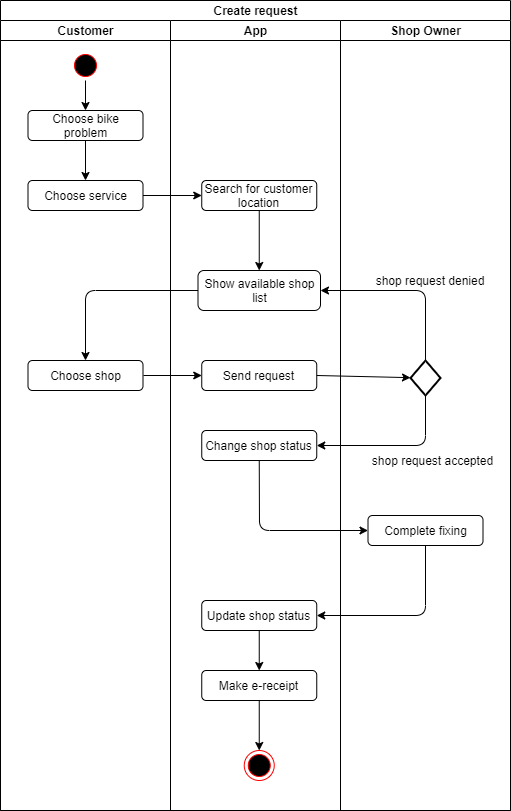


4.3 Class Diagram

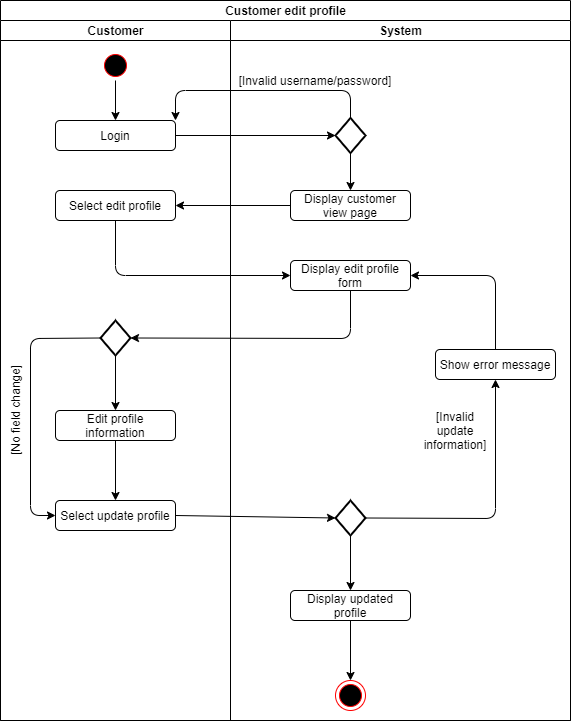


4.4 Activity Diagram

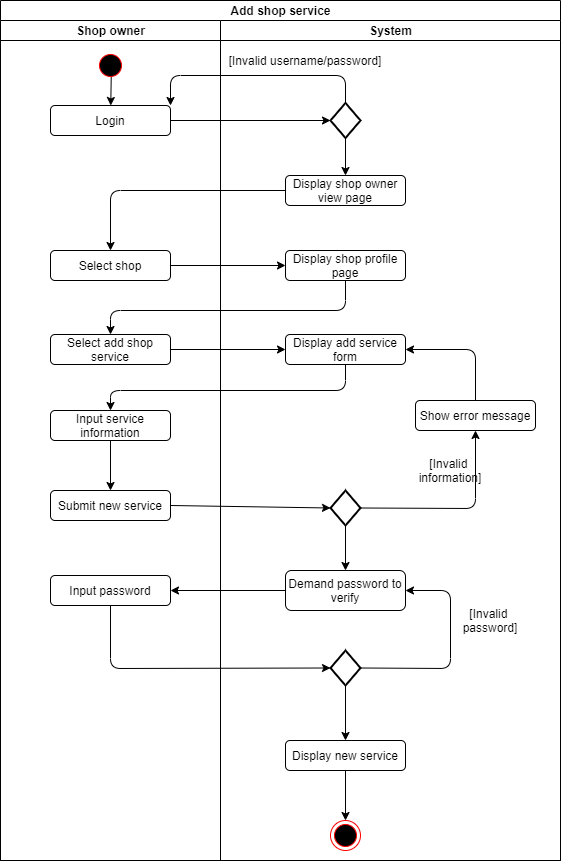
4.4.1



4.4.2

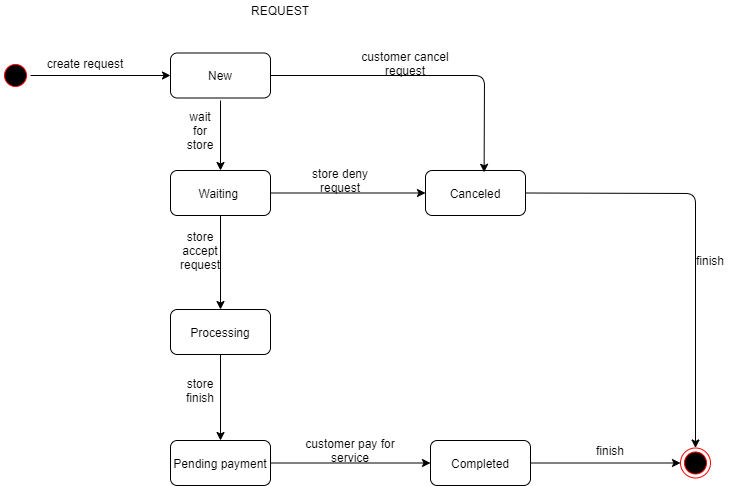


4.4.3



4.5 State Machine Diagram

4.5.1 Request created by customer



4.5.2 Service created by shop owner

