Sri Lanka Institute of Information Technology

Software Requirement Specification

Submitted by:

1.IT16109018 –A.M.I.L. Ariyarathna
2. IT16103832 –Y.N. Senanayake
3. IT16104068 –L.L.M. Wickramasinghe

Submitted to:

Mr. Pushpitha Sandaruwan

03/30/2018

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The Software Requirement Specification (SRS) is mainly designed to give a detailed description of designed and functionalities of 4C Knowledge Book Club Web Application. It will illustrate the purpose and complete declaration for the development of the Web Application. The SRS helps End Users, Project Managers and Software Engineer to have a clear understanding of 4C Knowledge Book Club Web Application and its functionalities that allow correct software to be developed. The SRS provide the foundation for this project. From this SRS, this web application will be developed and tested.

This is a two-way document that assures that both the client and the organization understand the other's requirements from that perspective at a given point of time. This contain detailed description of the both functional and non-functional requirement. This document will be useful for the client to ensure all the specification and requirements to design the system.

1.2 Document Convention

This document is prepared using Microsoft Word 2016.The document is written using the standard font, ‘Times New Roman' and the fixed font size that has been used for is 12pt with 1.5 line spacing. Bold face and indentation is used on general topics. Use case scenario is written according to Alistair Cockburn's template. UML diagrams have been created according to UML standards. The appearance and the flow of the document is organized using standard IEEE format.

1.3 Intended Audience and Reading Suggestions

The rest of this SRS contains, Overall Description External Interface Requirements web application Features Non-Functional Requirements and Other Requirements.

The SRS document can be used in any case regarding the requirements of the project and the solutions that have been taken. The document would final provide a clear idea about the web application that is building.

1.4 Product Scope

The objective of this project is to develop an e- book club where books can be bought from the comfort of home through the Internet. An online book club is a virtual store on the Internet where customers can browse the catalog and select books of interest.

Aim is to develop a web application using java web framework.

Develop a web application in 3 tier architecture involving user interface, controller and database. The user interface will be a web page hosted on a server. The web page consists of both static and dynamic content. All the data required for the application is stored in database tables. Controller accesses the data from the database and provides it to the user through user interface (web page).

1.5 References

[1] IEEE Format –

https://www.ieee.org/conferences_events/conferences/publishing/templates.html

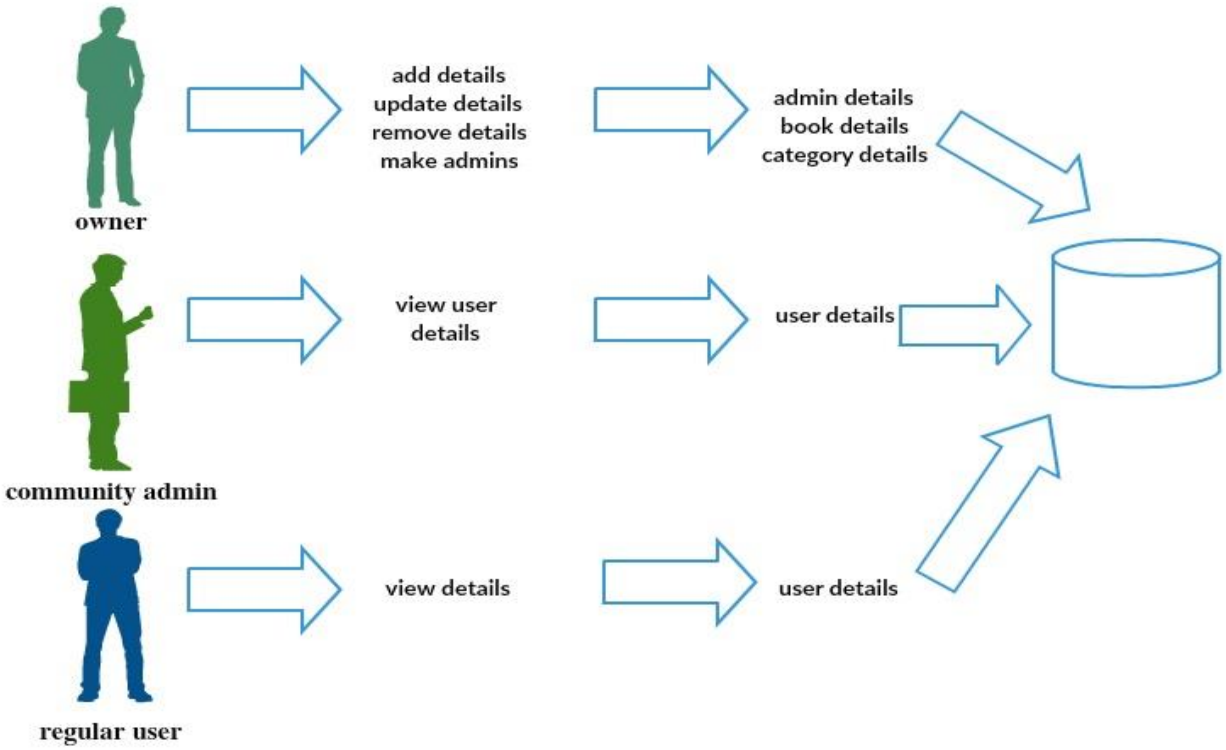
[2]
https://en.wikipedia.org/wiki/Software_requirements_specification

2. Overall Description

2.1 Product Perspective

This project deals with developing a simple web application for a Online Book Club. For this book club there are three types of user's levels namely Owner, Community Admin, Regular User. According to their user level there is a restriction to the view of the web application. There are three different views to specific user. Owner can view any number of details inside the Data store and retrieve any one's details. Community Admin can only view full name, permanent address, mobile and interests. Regular User can only view his details. Every member first need to register by providing their Full Name, Date of Birth, Permanent Address, Current Address, Mobile Number, Home Telephone Number and Interests.

Figure 2.1.1



2.2 Product Functions

Figure 2.2.1

Function	Login.
Input	Enter login credentials (username, password).
Output	View the main interface.
Process	Selecting the user levels and view the relevant interface.

Figure 2.2.2

Function	Register members.
Input	Enter full name, dob, permanent address, current address, mobile number, telephone number, interest and email etc.
Output	View the login page.
Process	Add a new member for member table.

Figure 2.2.3

Function	Add book.
Input	Enter book details (book name, author, price, category).
Output	Load the book list.
Process	Add a new book for book table.

Figure 2.2.4

Function	Add a new category.
Input	Enter category details (category name, category id).
Output	Load the category list.
Process	Add a new category for category table.

2.3 User Classes and Characteristics

There are three user levels in 4C Knowledge Book Club.

Figure 2.3.1

User Classes	Function Used	Security Privilege Level	Technical Knowledge
Owner	Add Users, Community Admins. View all the details.	Full Access	Can use the system
Community Admin	View details. View regular users some details.	Limited access rather than owner	Can use the system
Regular User	View regular users details only .	Limited access	Can use the system

2.4 Operating Environment

To run this system successfully, recommended hardware and software requirements are as follows.

2.4.1 Software Requirements

Web Technology : **Eclipse IDE**

Web Server : **Tomcat Server** (Server is used to host the application. The developed web application is deployed onto server.)

Database **: MySQL Workbench** (Database is used to store all the required data for the application. Database tables are designed according to requirement)

Code Behind : **Java, Java EE (JSP,Servlets) , HTML ,JavaScript**

Operating System: **Linux OS**

Version Controller: **Git Hub**

2.4.2 Hardware Requirements

Ram :1 GB

Hard disk Capacity : Hard disk with up to 1GB free space

Display type : Display Resolution 1920*1080

2.5 Design and Implementation Constraints

- **Web page:** The webpage serves as user interface to the application. Web page is developed using Jsp, HTML, Java Script
- **Database:** Database is used to store all the required data for the application. Database tables are designed according to requirement and developed using MYSQL Workbench
- **Server:** Tomcat Server, is an open-source Java Servlet Container developed by the Apache Software Foundation

  Tomcat implements several Java EE specifications including Java Servlet, Java Server Pages (JSP), Java EL, and WebSocket.

  Apache Tomcat includes tools for configuration and management but can also be configured by editing XML configuration files.
- **Servlets:** A servlet is a Java class that runs in a Java-enabled server.
- **JSP (Java Server Pages)** Java Server Pages (JSPs) are a Sun Microsystems specification for combining Java with HTML to provide dynamic content for Web pages.
- Used MVC Architecture model. MVC Pattern stands for Model-View-Controller Pattern. This pattern is used to separate application's concerns.

  Model - Model represents an object or JAVA POJO carrying data. It can also have logic to update controller if its data changes.

  View - View represents the visualization of the data that model contains. This is simply user interface.

  Controller - Controller acts on both model and view. It controls the data flow into model object and updates the view whenever data changes. It keeps the connection between model and view.

2.6 Assumptions and Dependencies

2.6.1 Assumptions

- Assuming certain web application owner should have enough knowledge to operate the application.
- Assuming client uses a computer with internet connection

2.6.2 Dependencies

- Access to the information of the web application will be depending on the staff hierarchy.
- Each user must keep their password as confidential. More over the user must have individual ID for creating a login in the application.
- Only owner can control admin, book and categories addition and deletion in the system.

3. External Interface Requirements

3.1 User Interfaces

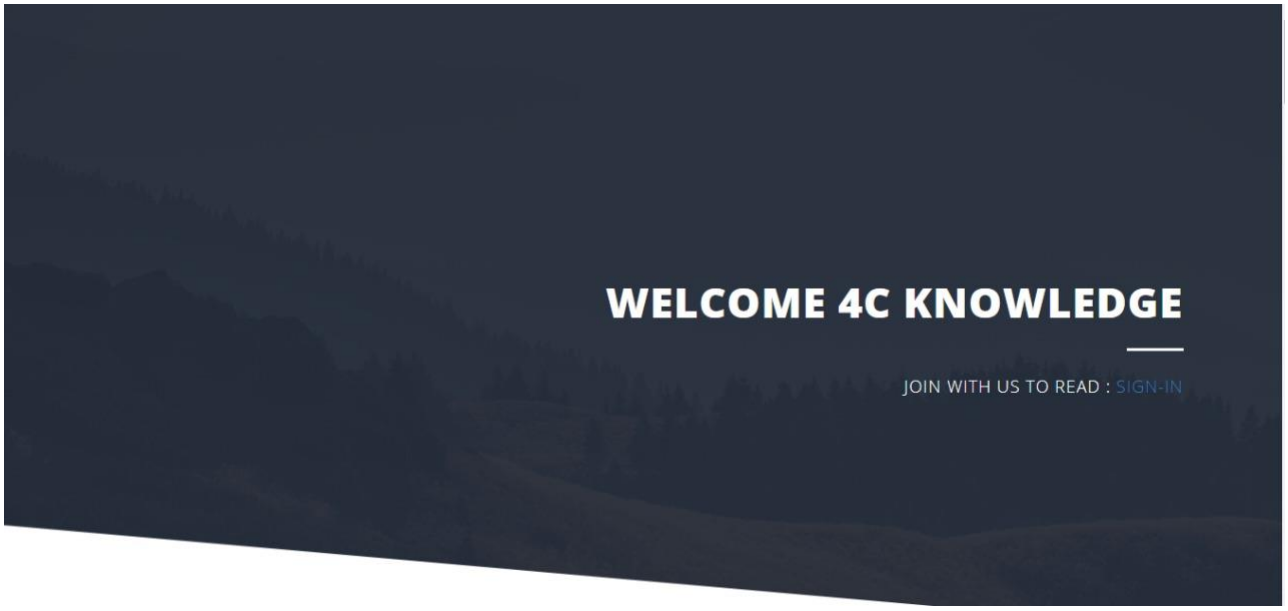


Figure 3.1.1: Main menu Interface

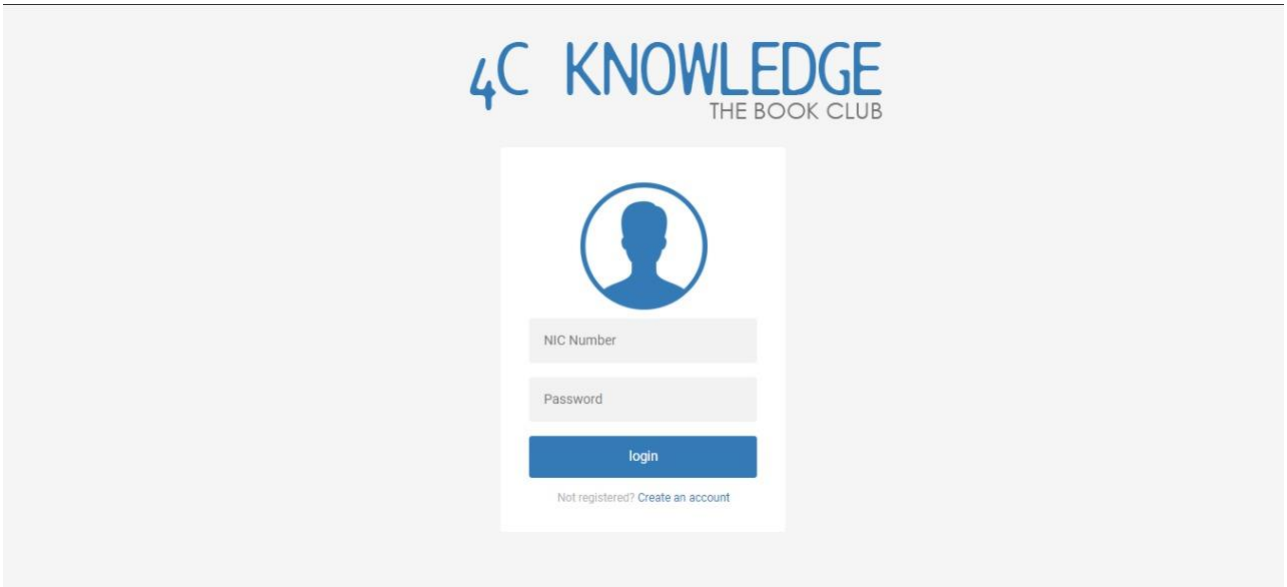


Figure 3.1.2: Login Interface

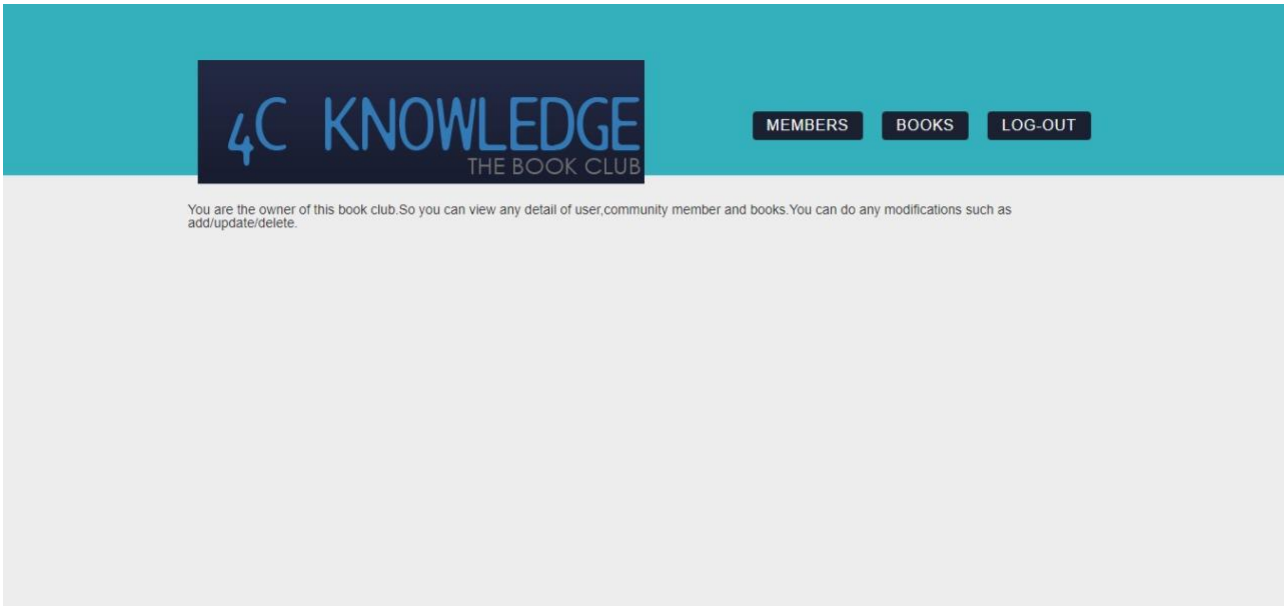


Figure 3.1.3: Owner's Interface

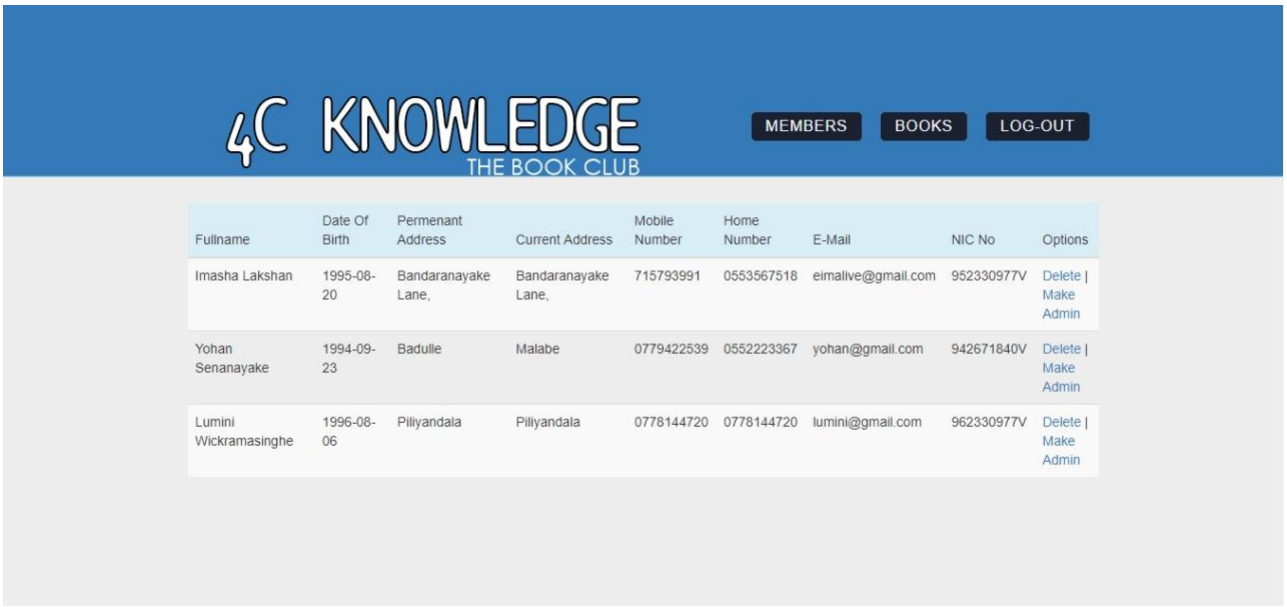


Figure 3.1.4: Owner's Interface (members)

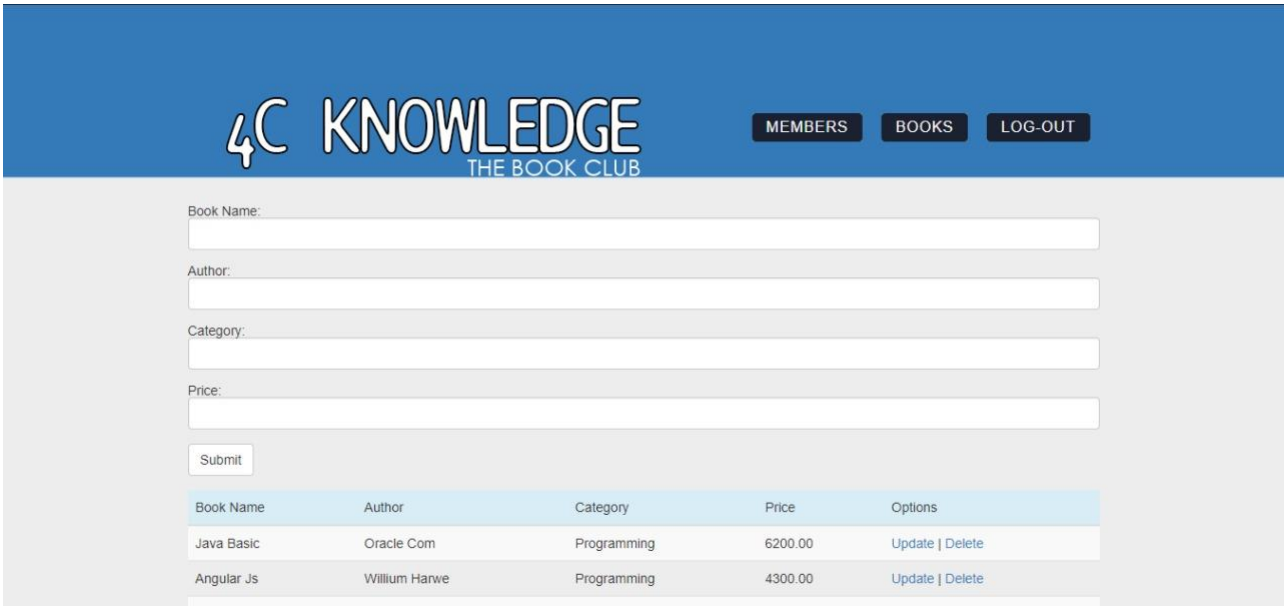


Figure 3.1.5: Owner's Interface (books)

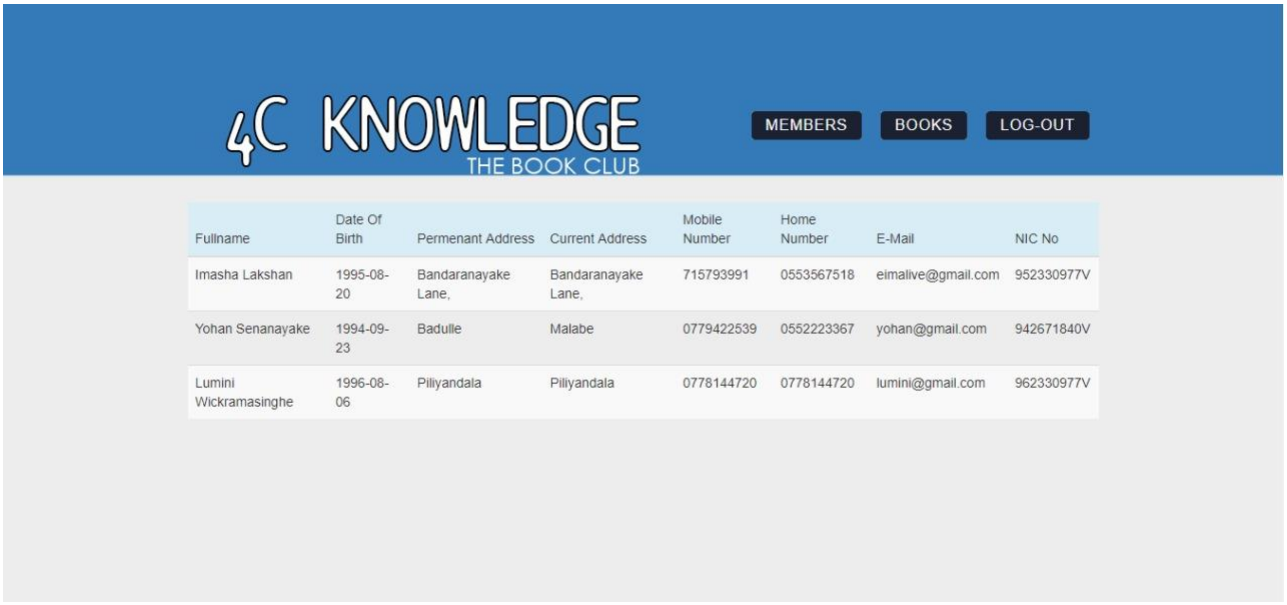


Figure 3.1.6: Owner's Interface (members)

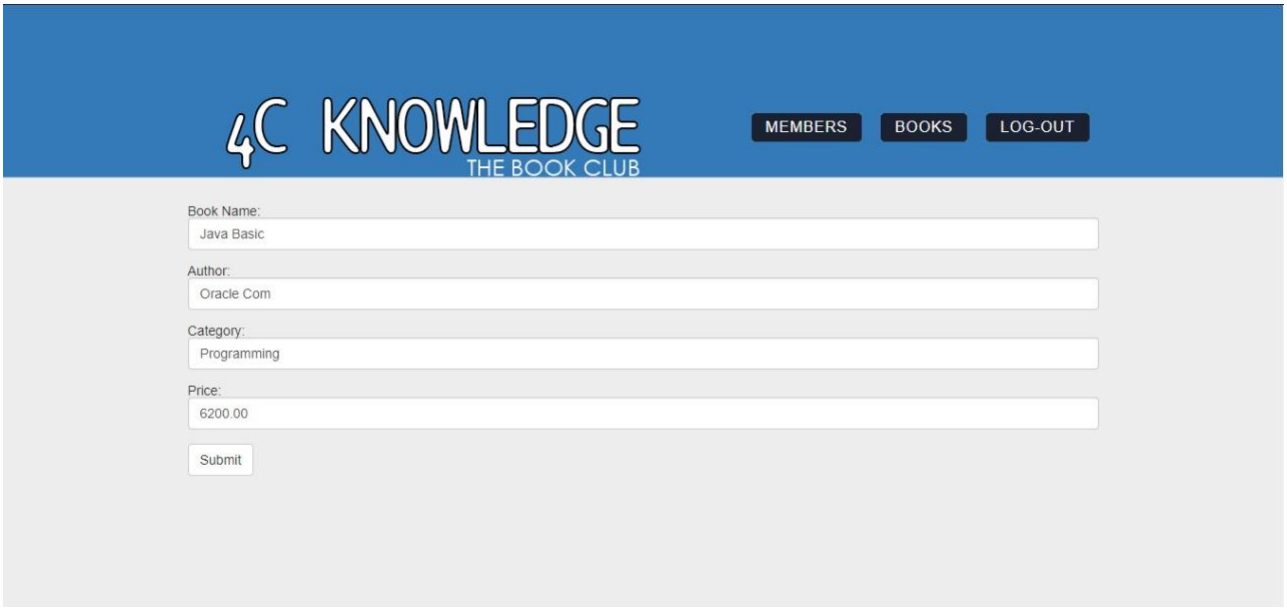


Figure 3.1.7: Owner's Interface (books)

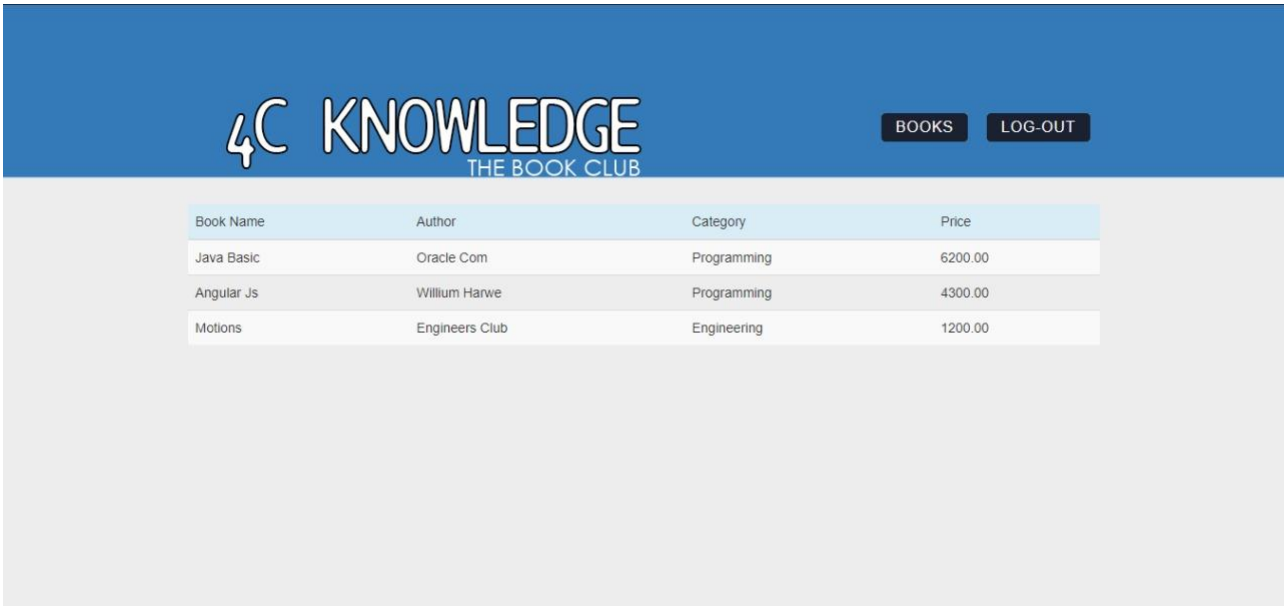


Figure 3.1.8: member's Interface (books)

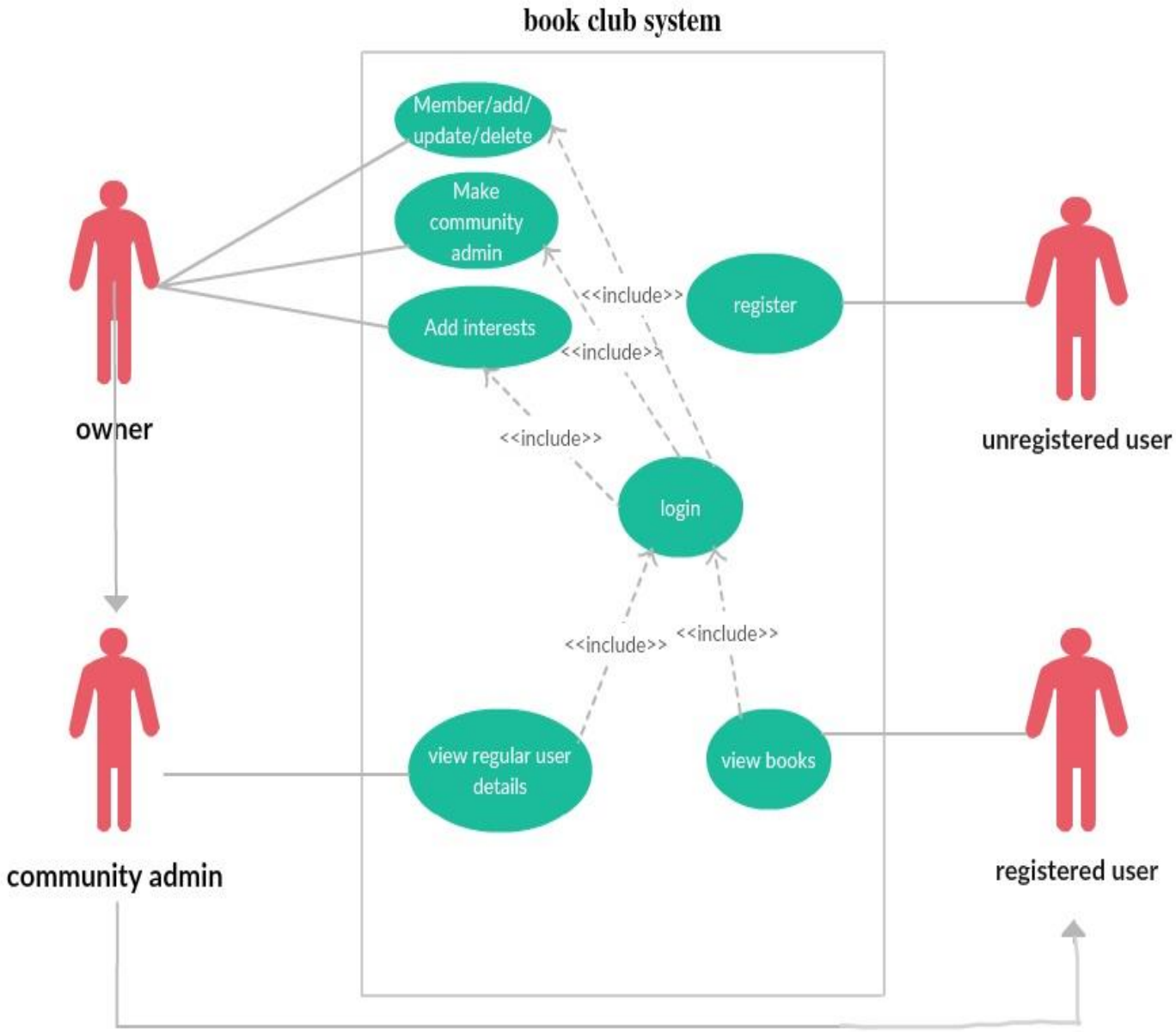
3.2 Software Interfaces

- This system is developed to work on any web browser.
- Eclipse Java Oxygen. Ink will be used to design the interfaces of the system.
- MySQL Workbench used to retrieve and store data.

4. System Features

4.1 use case diagram

Figure 4.1.1



4.2 Use case scenario

Use Case Name	login
Description	This use case is about login to the system.
Precondition	None
Primary Actor	User
Main Success Scenario	1.Use case begins when the user enter his login details. 2.System will load the main view . 3.Use case ends showing user logged in successfully.

Figure 4.2.1

Use Case Name	Register user
Description	This use case is registering an user to the system.
Precondition	None
Primary Actor	User
Main Success Scenario	1.Use case begins with system will load the main view 2.user enter his registration details. 3.System will load the interface. 4.Use case ends showing user added successfully.

Figure 4.3.1

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- Individual function Is well organized and well tested for performance and for reliability (test for validation with every possible value)
- Shortcuts are available in interfaces to every user to gain performance. (E.g.: - Home button).
- Headers and footers are statically implemented. So that will increase the speed of the implementation.

5.2 Safety Requirements

- Users can only access to the system if he/she had an account in the book club. If not, they can't view any details about the web application. Community admin also need to register as a user then owner need to higher their position as a community admin.
- Passwords are encrypted using post methods.

5.3 Security Requirements

- The external security (threads to the system externally) can be provided by providing login authentication.
- The data that are stored in the database must be private.

5.4 Software Quality Attributes

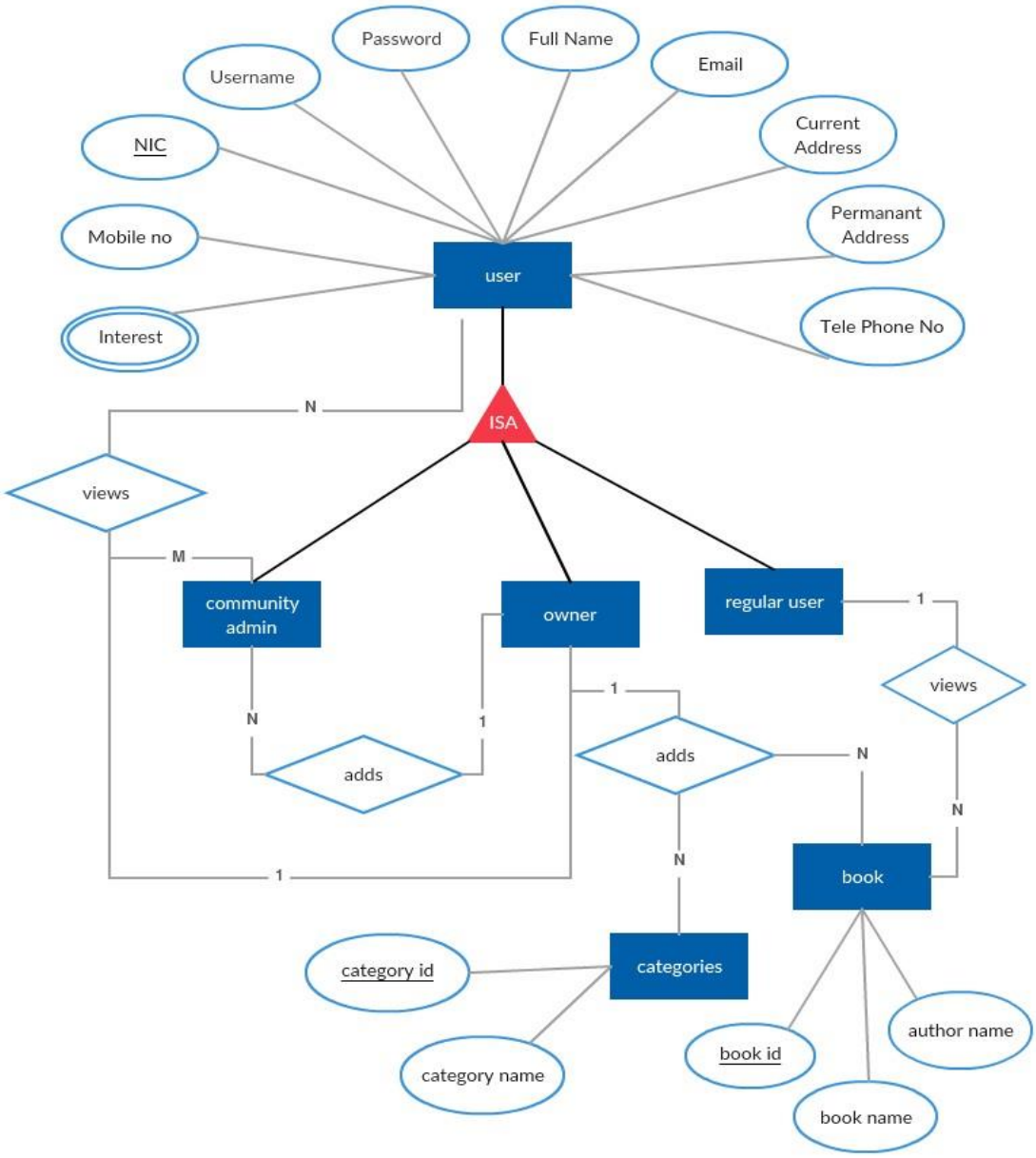
- Availability: - Whether the system and its functions can be used whenever the user wants Efficiency: - How much less number of resources and time are required to achieve a task through the system.
- Flexibility: - Ability to add new features to the system and handle them conveniently. Install Ability: - How easy to install an application or software to the system without any short coming.

- Integrity: - How the system would insecure the information in the system and how it avoids the data losses.
- Maintainability: - How easy is to keep the system as it is and correct defects with making changes.
- Reliability: - Mean time to failure, Probability of unavailability or rate of failure occurrence.
- Usability: - How easily a person can be taken the benefits of the system and the user friendliness

5.5 Business Rules

- The Owner has their own login, owner can higher their user's user level as community admin.
- Only the owner has the privilege to handle the all details such as modify add new details of books, and modify, view all the user details.
- In addition, community admin can view other normal user's some user details.

Appendix A: Analysis Model -ER Diagram



Class Diagram