Software Requirements Specification

For

A Web Application for a Book Store

Smirithika. C   IT16514614

Gowshalini. R   IT16113800

A. A. Ahamed Arshad   IT16032316

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the web application of a book store. It will illustrate the purpose and complete declaration for the of system including the features and functionalities. It will also explain system constraints, interface and interactions with other external applications.

1.2 Product Scope

This web-based application is intended to fulfill users to carry on registration and management of their profile for an online book store. This software will help people to register as users to the book store system and view their personal details and manage it. It also provides access to the owner of the system to view all the user details and the access to the admin of the system to view specific set of details of the registered users.
All the system information is maintained in a database.

1.3 Glossary

Term	Definition
User	Someone who interacts with the web application
Registered User	Someone who is already registered their details to the book store app, who can access the web application.
Owner	The person who is given specific permission for viewing managing the system
Admin	The person who is given specific permission for viewing managing the system, but less permission than the owner.

1.4 References

http://www.studentstutorial.com/java-project/send-mail-in-java.php [Accessed 2018 Mar]

https://www.youtube.com/watch?v=bDHGnDirC14&t=15s [Accessed 2018 Mar]

https://www.youtube.com/watch?v=F51dZmaMCQchttps://www.youtube.com/watch?v=F51dZma MCQc [Accessed 2018 Mar]

https://www.youtube.com/watch?v=WXmTxgDg52o [Accessed 2018 Mar]

https://stackoverflow.com/questions [Accessed 2018 Mar]

https://tutorialspoint.com [Accessed 2018 Mar]

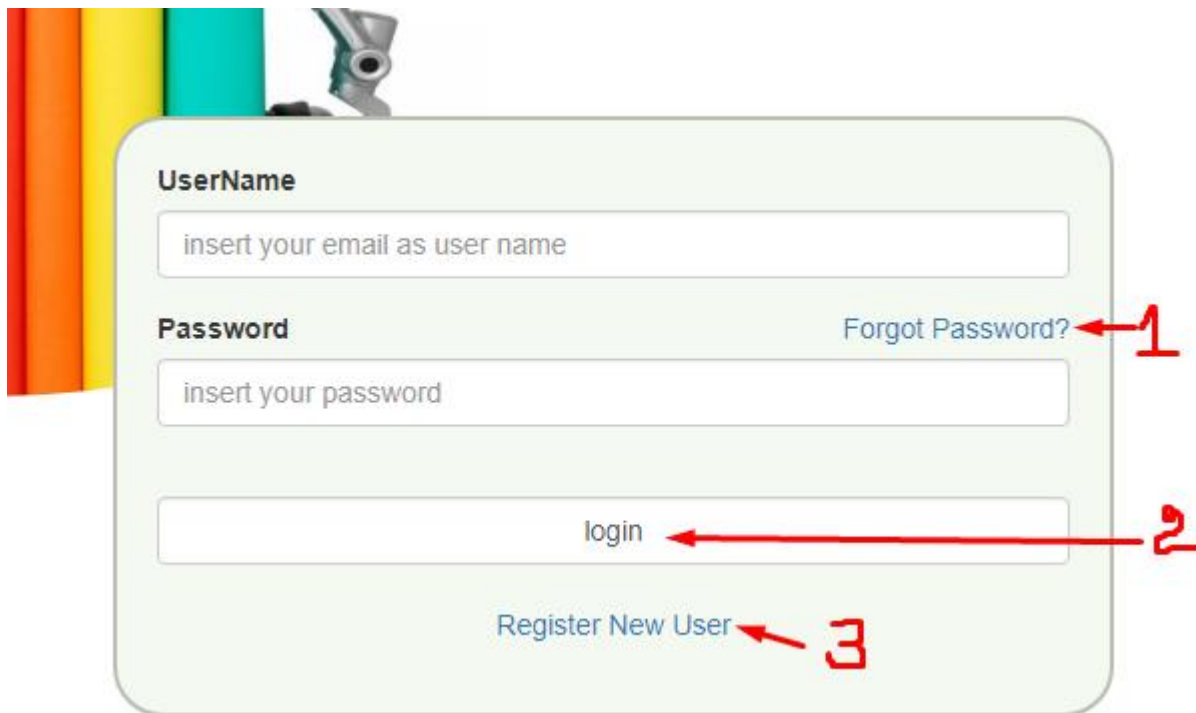https://www.w3schools.com/ [Accessed 2018 Mar]

2. External Interface Requirements

2.1   User Interfaces

Front end software: HTML, CSS, Bootstrap, Java, JavaScript

Database: MySQL (Xampp)

- Here you can see the interface forms:

Figure 2.1.1

Buttons and functions

1) Redirects to ForgotPw.jsp page. (Can reset password. When reset finishes a conformation mail will be sent to the user)
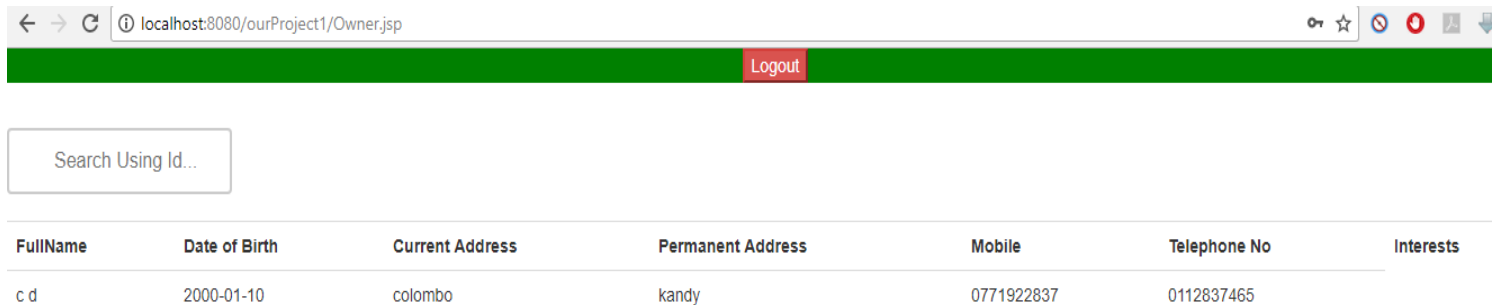
Figure 2.1.2 ForgotPw.jsp page

2) Redirects to UserView page. (According to the user permission the table will be displayed.)
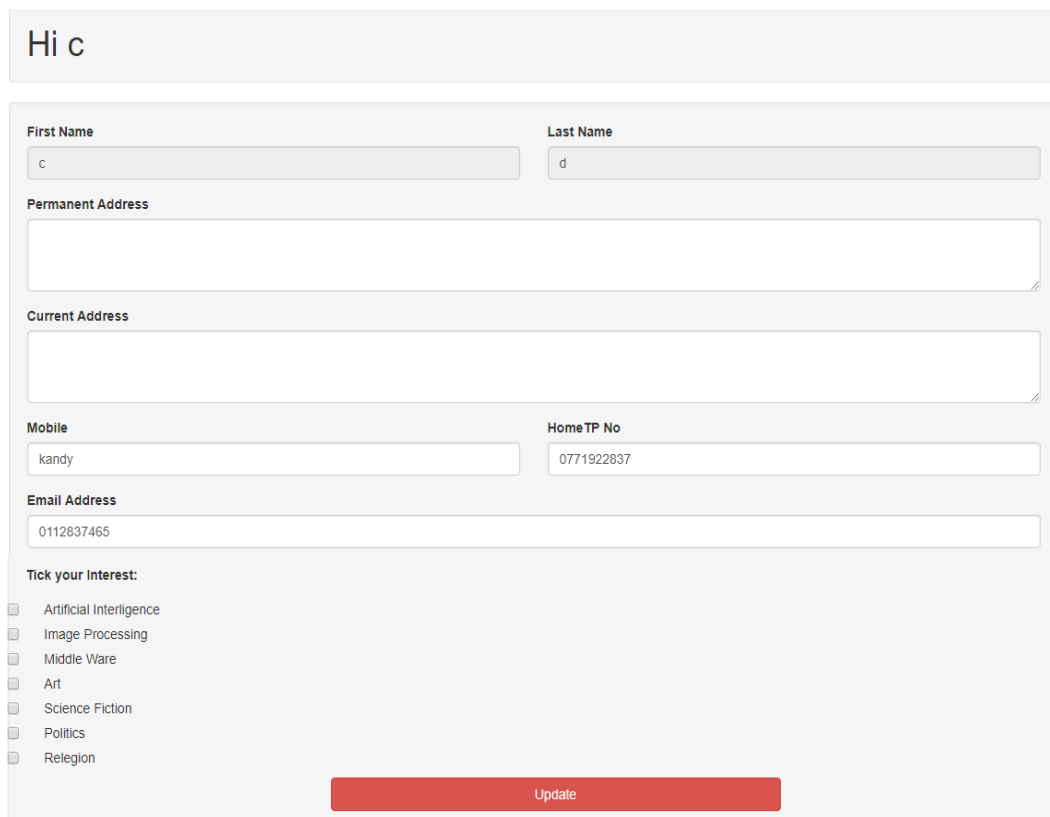
If Owner: The owner of the system can view all the user details.

Figure 2.1.3 Owner.jsp pag

If Member: The members of the web application can view their account

Figure 2.1.4 UserView.jsp page

If admin: The admin of the system can view full name, permanent address, mobile number and interests of the member.
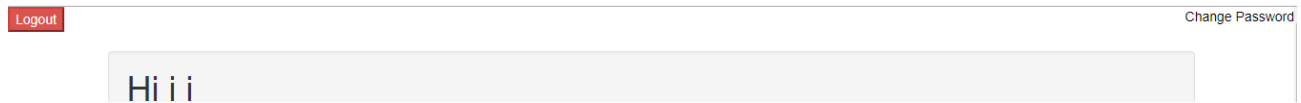
Figure 2.1.5 admin.jsp page

3) Redirects to Register.jsp page.

Figure 2.1.6 Register.jsp page
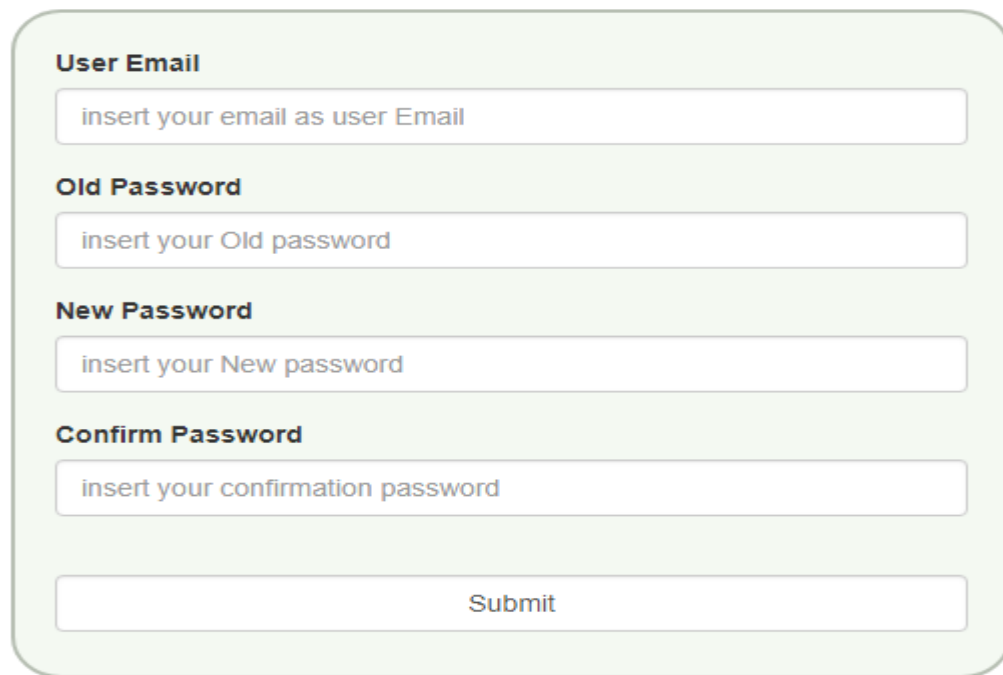
- Every user has change password and logout option after logging in.

Logout   Change Password

Hi i i

✓ When user clicks change password option it will redirect to changePw.jsp page. If user wishes can change the password by conforming user existence in the web application.

User Email

insert your email as user Email

Old Password

insert your Old password

New Password

insert your New password

Confirm Password

insert your confirmation password

Submit

Figure 2.1.7 changePw.jsp page

✓ When user clicks the logout button the login page will be displayed.In that case we user can't access the previous page he/she accessed.

2.2   Hardware Interfaces

Since this project is web based, it is compatible with all the browsers and can be run on any operating system and processor.

2.3   Software Interfaces

Kali Linux Operating System

A browser which supports HTML, JavaScript

MySQL database

Java EE IDE-Eclipse

2.4   Communications Interfaces

The system will send automatic confirmation email to the user when resetting the password.

Http protocol is used in maintaining the session.

3. Functional Requirements

3.1   Use Case Diagram
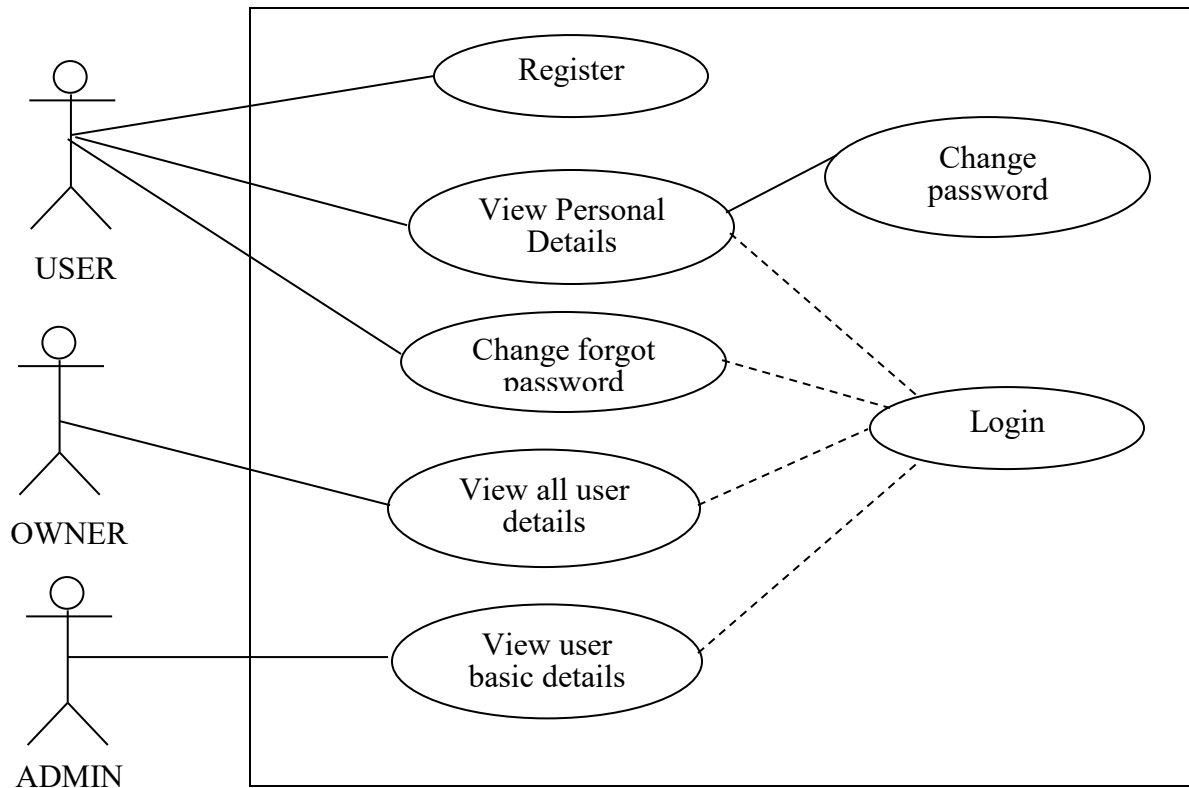
Figure 3.1.1

3.2 Use Case Scenarios

Use case name	Login to the system	
Preconditions	Load the login page successfully. Database details loaded to the system.	
Success End conditions	Successfully logged into the system.	
Actors	Admin / Owner / Registered User	
Main Flow	**Step**	**Actions**
	1.	Enter username and password
	2.	Login

Table 3.2.1

Use case name	View all details of registered users.
Preconditions	Successfully Logged into the system as an owner. Database details loaded to the system.
Success End conditions	View all details of all the registered users.
Actors	Owner

Table 3.2.2

Use case name	View basic details of registered users.
Preconditions	Successfully Logged into the system as an admin. Database details loaded to the system.
Success End conditions	View basic details of all the registered users.
Actors	Admin

Table 3.2.3

Use case name	View personal details
Preconditions	Successfully Logged into the system as a registered user. Database details loaded to the system.
Success End conditions	View all personal details.
Actors	Registered User

Table 3.2.4

Use case name	Update personal details
Preconditions	Successfully Logged into the system as a registered user and view all personal details. Database details loaded to the system.
Success End conditions	Successfully update personal details.
Actors	Registered User

Table 3.2.5

Use case name	Register as a new user	
Preconditions	Load the login page successfully. Database details loaded to the system.	
Success End conditions	Successfully register as a user.	
Actors	User	
Main Flow	**Step**	**Actions**
	1.	Enter all personal details.
	2.	Successfully register as a user.

Table 3.2.6

Use case name	Change password	
Preconditions	Load the user view page successfully. Database details loaded to the system.	
Success End conditions	Successfully password changed.	
Actors	Registered User	
Main Flow	**Step**	**Actions**
	1.	Enter email address.
	2.	Enter old password.
	3.	Enter new password.
	4.	Re-enter new password.
	5.	Confirm and change password successfully.

Table 3.2.7

Use case name	Forget password	
Preconditions	Load the login page successfully. Database details loaded to the system.	
Success End conditions	Successfully password changed.	
Actors	Registered User	
Main Flow	**Step**	**Actions**
	1.	Enter email address.
	2.	Enter new password.
	3.	Re-enter new password.
	4.	Confirm and change password successfully.

Table 3.2.8

4. Other Nonfunctional Requirements

4.1 Performance Requirements

Performance will not be an issue because all the queries have small data. There are no immediate delays in every action and response of the system. 95% of the transactions will be proceed in less than 2 second. In the case of searching it is done in the client side without from database. Therefore, it reduces the process time of searching. The system provides access to multiple users simultaneously.

4.2 Safety Requirements

The data in data base are recoverable if there is any break down in back end since we use MySQL server. This web app doesn't affect any internal components and data of the device where the web app is accessed.

4.3 Security Requirements

This program uses object-oriented concept to protect data. The user authentication provides secure access to the data according to the access permission. Session is maintained to prevent the access from others and maintain the user details. The passwords are stored in encrypted format. It provides security from hackers.

4.4 Software Quality Attributes

4.4.1 Reliability

The probability of system failure occurring is very less since it is a portion of a web app which contains only the views.

4.4.2 Availability

The application is available at any time except for integration period since it is a part of the software.

4.4.3 Maintainability

The document could be easier to the users who uses and maintains the application and the developers who wish to edit or develop further.

4.4.4 Portability

The application supports the new version of browsers, different platforms and operating system.

4.4.5 Usability

The interfaces are user friendly and easy to use by users from any technical background. The application is easy to handle with no delays. The interfaces are supportable for any devices are scalable.