

# *Online Book Renting Application*

---

*Rohith Yogi N*

*Srujan Swaroop G*

*Siddarth Reddy D*

*Puneeth K*

---

# **Software Requirements Specification**

*International Institute of Information Technology, Bangalore*

*11th November 2018*

## ***Table of Contents***

***Table of Contents      ii***

***Revision History      ii***

***1.      Introduction    1***

***1.1     Purpose        1***

***1.2     Document Conventions        1***

***1.3     Intended Audience and Reading Suggestions    1***

***1.4     Product Scope    1***

***1.5     References      1***

***2.      Overall Description    2***

***2.1     Product Perspective    2***

***2.2     Product Functions    2***

***2.3     User Classes and Characteristics      2***

2.4	<i>Operating Environment</i>	2
2.5	<i>Design and Implementation Constraints</i>	2
2.6	<i>User Documentation</i>	2
2.7	<i>Assumptions and Dependencies</i>	3
<b>3.</b>	<b><i>External Interface Requirements</i></b>	<b>3</b>
3.1	<i>User Interfaces</i>	3
3.2	<i>Hardware Interfaces</i>	3
3.3	<i>Software Interfaces</i>	3
3.4	<i>Communications Interfaces</i>	3
<b>4.</b>	<b><i>System Features</i></b>	<b>4</b>
4.1	<i>System Feature 1</i>	4
4.2	<i>System Feature 2 (and so on)</i>	4
<b>5.</b>	<b><i>Other Nonfunctional Requirements</i></b>	<b>4</b>
5.1	<i>Performance Requirements</i>	4
5.2	<i>Safety Requirements</i>	5
5.3	<i>Security Requirements</i>	5
5.4	<i>Software Quality Attributes</i>	5
5.5	<i>Business Rules</i>	5
<b>6.</b>	<b><i>Other Requirements</i></b>	<b>5</b>

### *Revision History*

<i>Name</i>	<i>Date</i>	<i>Reason For Changes</i>	<i>Version</i>

# 1. Introduction

## 1.1 Purpose

- This is Software Requirement Specification document for a web application called “Online Book Rental Service”.
- The primary purpose of this project is to facilitate the user to rent books online for a period of time on a predefined payment basis.
- It also enables users to donate their books if they are no longer useful to them.

## 1.2 Document Conventions

IEEE standards and typographical conventions were followed when writing this SRS.

The priorities for higher-level requirements are assumed to be inherited by detailed requirements.

Some of the requirement statements have its own priority.

## 1.3 Intended Audience and Reading Suggestions

The primarily intended readers of the document are:

- Developers: It serves them as a reference for design and implementation.
- Users: Enables them to check whether the specified requirements are met or not.
- Testers: Serves the purpose of a base, so that they will test the implementation of the project according to the document.

Recommended flow of reading would be just to follow the sequence in which the document is drafted.

## 1.4 Product Scope

The project is web-based platform which is accessible 24/7.

It is convenient and easy-to-use web application.

*Users can avail following services:*

- *Can register themselves.*
- *Can explore and search wide range of books for renting.*
- *Can request a book if unavailable.*
- *Can track their order.*
- *Can cancel their order.*
- *Can give rating and provide their valuable feedback.*
- *Can donate the books.*

*These services make it more flexible and convenient environment for the purpose.*

*Hence the application can be considered as dependable and trustable choice because of its flexibility in availability of books and their corresponding prices.*

## **1.5 References**

*The document corresponds to IEEE standards,*

*source of IEEE template:- [https://web.cs.dal.ca/~hawkey/3130/srs\\_template-ieee.doc](https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc)*

## **2.Overall Description**

### **2.1 Product Perspective**

*Our project is follow on of “online book rentals type projects”. It is a stand alone project and does not depend on any other software. It runs on web-portal only and all its features are accustomed only to web version.*

*Since this is a data centric project,it will need somewhere to store the data (such as information about books,users etc), for this purpose database is used. As our application mainly deals with data,to execute all the features it has to communicate with the database.*

- *To execute the service of process of renting books, web application requests the user to provide the name of book and duration for which he wants to rent the book, which in turn communicates with database to get info of books ,so that user selects and confirms the order.*
- *To executive the service of process of donating ,web application requests the user to provide the details of book (such as title, author, edition, publication etc),after which the received details get stored in the database so that the admin accepts the request and sends an executive to collect the the book.*
- *The above mentioned details(was previously "These") are the main aspects of our web application.*

## 2.2 Product Functions

- **Login / sign-up:** Allow user to create an account if he/she doesn't have one or sign in into their account to continue their transaction.
- **Search:** User can search for a book by its author's name or book name.
- **Donate:** User can donate his/her books
- **Request:** User can request for a book if it is unavailable.
- **Track\_order:** User can know about the status of order. i.e (date of dispatch,date of arrival)
- **Feedback:** User can provide his/her rating based on their delivery and condition of book.
- **Cancel\_order:** User can cancel his/her order if they want.

## 2.3 User Classes and Characteristics

<b>Class</b>	<b>Characteristic</b>	<b>Function</b>
<i>Admin</i>	<i>The one who maintains the website</i>	<p><i>Accepts the request for canceling</i></p> <p><i>Provides details of order tracking</i></p> <p><i>(Here , what about adding/deleting users)</i></p>
<i>User</i>	<i>The class of members who use the web application</i>	<p><i>Search</i></p> <p><i>Donate</i></p> <p><i>Request</i></p> <p><i>Track_order</i></p> <p><i>Cancel_order</i></p> <p><i>Provide feedback</i></p>

## 2.4 Operating Environment

*Operating environment for our project is as listed below.*

- *distributed database*
- *client/server system*
- *Operating system: Windows,Linux*
- *database: mysql + database*
- *platform: Java*
- *Browser: supports all browsers.*

## 2.5 Design and Implementation Constraints

*Language:Entire presentation of website is/will be in English(only).*

*Limited Servers and data:The number of servers available to us is very less,so there is restriction for(previously was “to”) data storage.*

## 2.6 User Documentation

- *Provides links ,which has decent information about the usage of application.*
- *Provides ‘Help’ column in our website which has brief description of the application*

## 2.7 Assumptions and Dependencies

- *Default Payment mode is only COD(Cash On Delivery).Other Payment modes will be tested after the completion of project if there is time left.*

## 3. External Interface Requirements

### 3.1 User Interfaces

- Front-end software:- html, css, javascript,php.
- Back-end software:- Mysql , Ruby with rails .

The front-end of our project is completely based on html,css ,javascript and php. Very little of back-end deals with php and is mostly based on mysql,ruby on rails.

### 3.2 Hardware Interfaces

- Windows and linux devices
- A browser which supports html, CSS , javascript.

*Supported devices include desktops (containing windows or any linux operating systems). We do not plan to release an android version of the product , instead android devices can open them normally through the standard browsers present with them.*

### 3.3 Software Interfaces

The softwares required and the versions used for this project are:-

1. Windows 10
2. Ubuntu 16.0.4
3. Sql (Mysql v 5.7)

Windows has been used for its best support and user friendliness. Ubuntu has been used to implement the backend(sql) where Mysql has already been installed. For sql we used Mysql for its ease of usage and understanding.

### 3.4 Communications Interfaces

*This project supports all types of web browsers. Simple electronic forms for the online renting ,cancellation and donation are being used.*

## 4. System Features

*This section contains all the features implemented in the system.*

### 4.1 Request Book

*The user can request for a book if he/she does not find the book they were looking for the admin includes the book requested within a specific amount of time mentioned beforehand.*

#### 4.1.1 Description and Priority

*The books requested by the user gets included in the website so it allows them to access a wide range of books.*

*It's a major priority for any book rental system because, if the books that the user was looking for is not found then there is no scope for improvement in the project.*

*There is always an incremental growth in the application because of this feature.*

#### 4.1.2 Stimulus/Response Sequences

*There is a separate portal through which the users can request for the books that they are looking for. Only admin can get the access of these requests and act accordingly.*

*The Admin specifies the time period required for him/her to make the book available for the user.*

#### 4.1.3 Functional Requirements

*There are various functional requirements associated with this feature.*

*A google form is required for the user to request a book from and that must be accessible by the admin.*

*If the user is requesting for some book that was already present in the website it should give an error saying already present.*

*This request book portal should contain all the books requested by that specific user and the it should also contain the time at which they will be made available.*

## **4.2 Track order**

*The User can track the status order by logging into their accounts where the current order is present along with all his previous orders ,so that they can check their previous orders also*

### **4.2.1 Description and Priority**

*User can get real time updates of his order , so he can be readily available to collect his delivery.*

*Tracking order is one of the few features that is not present in many .Including it in our project .Including it in our project differentiates us from many other websites.*

*There is always an incremental growth in the application because of this feature.*

### **4.2.2 Stimulus/Response Sequences**

*User can find all the orders that have been placed and for the current orders , there is an option to track the order where order can be tracked.*

*User can make himself readily available at the time of delivery mentioned in the page.*

### **4.2.3 Functional Requirements**

*There are various functional requirements associated with this feature.*

*If there is any delay in delivery of the book, there must be a notification about that.*

*Admin can change the book's status of delivery(like reached or ready for delivery etc.).*

*This portal must specify the present location of delivery like if it is in a facility or is it out for delivery etc.*

## **4.3 Extend renting time**

User can extend the return status of the book that has been ordered by using extension option which is available in their account.

### **4.3.1 Description and Priority**

*User may want to extend the return date of their order due to many issues, so there will be an option in their account to do the same.*

*Extend renting time is one of the few features that is not present in many .Including it in our project differentiates us from many other companies.*

*There is always an incremental growth in the application because of this feature.*

### **4.3.2 Stimulus/Response Sequences**

*User can extend the delivery*

*User can make himself readily available at the time of delivery mentioned in the page.*

### **4.3.3 Functional Requirements**

*There are various functional requirements associated with this feature.*

*User must be provided with an option to extend their delivery date.*

*The new delivery date must be provided after the user has opted for extended rent date.*

## **4.4 Extend delivery date**

Admin is able to extend the delivery date or cancel the order in case of unexpected issues.

### **4.4.1 Description and Priority**

*Admin may want to delay the delivery of the order due to many reasons. This change in delivery date can be done here.*

*This is a much required feature because circumstances may change very frequently. This is not a compulsory but a viable option in times of need.*

*This may sometimes bring in downfall of the application but this is a very much required option .*

## **4.5 View Orders**

Admin can view all the orders that have been placed .

## **4.6 Add To Cart**

*Users can add items or books that they want into the cart.*

## **4.7 View/Manage Cart**

*User can manage the cart by removing unwanted books or by going into the delivery page.*

## **4.8 Request Books**

*If user is not able to find their required book,they can request the book with the admin.*

## **4.9 Delete Users/Books**

*Admin is able to delete worn or torn books from the database.He can also remove users if required.*

## 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

*Request book : For this feature we need a timer to start when the book is requested to when the book will be made available. For this we need a clock generic so that even when the application is closed it runs in the background.*

### 5.2 Safety Requirements

*All the books must be accounted for. So the admin can add a criteria where the caution money could be collected before hand incase of any damage or lost book replacement. The details of the user is collected in the database. So if there is a damaged or a lost book case the users should pay a fine to the delivery agent.*

### 5.3 Security Requirements

*As they say do not judge a book by its cover so the summary is displayed in the web application where user can read the summary, look at the rating. These will help the user to buy a book. The web application can determine the similar books to be displayed to the user. All these passwords are securely stored in the database through encryption so that every users data remains private.*

### 5.4 Software Quality Attributes

*Admin can add books and remove books at will and he/she is able to view all the orders placed.*

*Admin checks for the book quality and ensures it is ready to use before it goes online.*

*Admin can postpone or prepone a delivery on his/her will.*

*Admin can remove the users on his will because of his/her account authentication failure*

*Admin can also view all the users using the website.*

## 5.5 Business Rules

*Every business has its own rules ours does as well. These are some of the rule associated with our project:*

1. *All users must sign up in order to rent a book.*
2. *Users should provide the credentials correctly. So they should always remember them.*
3. *Every user must provide the details correctly in order for them to receive the book.*
4. *Book renting price is subjected to change.*
5. *The time period of the return/request book is subjected to change.*
6. *The delivery time is also subjected to change.*
7. *The user has an option of paying by card/cash once the book is delivered to them.*

## 6. Other Requirements

*The admin could assign prices to the books donated to him and place them in the website. Also the check will be done to see if the books are in good condition or not. If not in good condition the books will be repaired and placed on the website. These are all internal legal and reuse objectives that will be taken before the books are made available to everyone.*

# **Online Book Renting Application:**

## **User stories:**

### **1.User :**

- As a user, I want to search for a book.(based on isbn number, author name, book name, keyword)
- As a user, I want to set the duration for which I want to rent the book.
- As a user, I want to request for a book if not found.
- As a user, I want to donate my books.
- As a user, I want to log-in/sign-up to my account.
- As a user, I want to place the order.
- As a user, I want to extend my duration of renting a book.
- As a user, I want to track my order details.
- As a user, I want to look at my past orders.
- As a user, I want to rate the book I ordered.
- As a user, I want to cancel my order.

## **2.Admin :**

- As an admin, I want to maintain the user details.
- As an admin, I want to add books to the website.
- As an admin, I want to accept donated books.
- As an admin, I want to remove books from the website.
- As an admin, I want to keep track of transaction details.
- As an admin, I want to update the tracking details.
- As an admin, I want to impose penalty on damaged books.
- As an admin, I want to manage the prices of the books.
- As an admin, I want to cancel or extend the delivery.

## **3.GUEST :**

- As a guest , I want to search for a book (based on ISBN number,author name,publication,book name)
- As a guest, I want to create an account in the website.
- As a guest, I want to explore all the books that are available in your website.



# Software Engineering Project

## Online Book Renting Application

Rohith Yogi

IMT2016072

Puneeth K

IMT2016018

Siddarth Reddy

IMT2016037

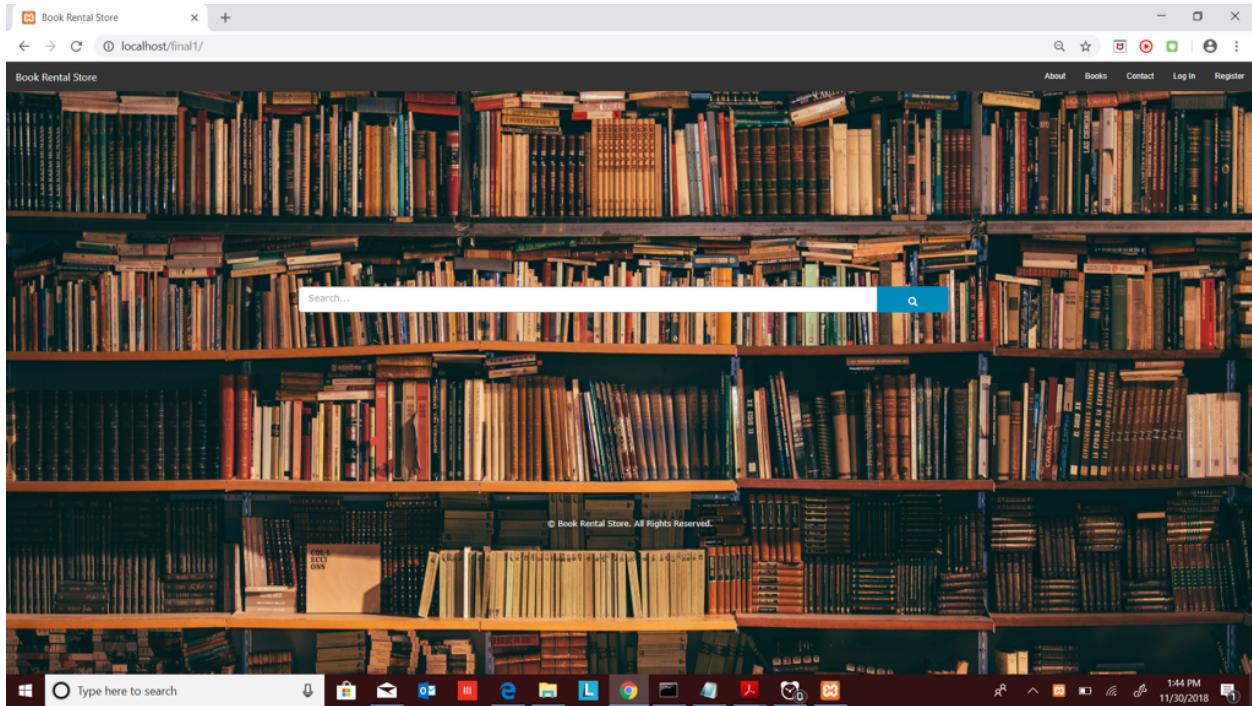
Srujan Swaroop

IMT2016033

## Design Document

**Note:**

Not every screen has been placed in the documentation but only the main screens has been placed.

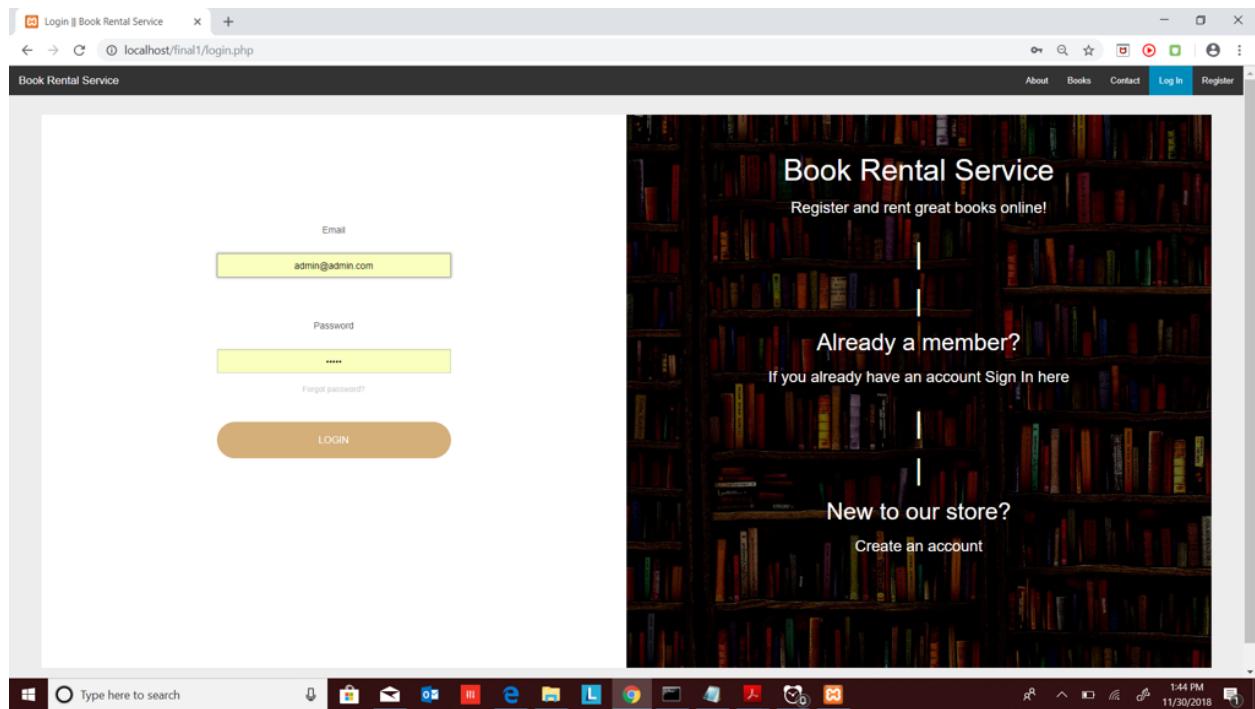
**INDEX PAGE:****User Stories Involved:**

- As a user, I want to search for a book.(based on isbn number, author name, book name, keyword)
- As a guest , I want to search for a book (based on ISBN number,author name,publication,book name)

**Validation:** Name has to be exact to implement the search operation.

**Navigation:** (Search-->Books Page)

## Login screen:



### User Stories Involved:

- As a user, I want to log-in to my account..

**Validation:** Passwords must be Alphanumeric.

**Navigation:** (login button, guest button → Home screen)

## Register screen:

The screenshot shows a web browser window titled "Register || Book Rental Service". The URL in the address bar is "localhost/final1/register.php". The page has a header "Book Rental Service" and a navigation menu with links for "About", "Books", "Contact", "Log In", and "Register". Below the header, there are seven input fields labeled "First Name", "Last Name", "Address", "City", "Pin Code", "E-Mail", and "Password". At the bottom of the form are two buttons: "Register" and "Reset". A small copyright notice "© Book Rental Service. All Rights Reserved." is at the very bottom.



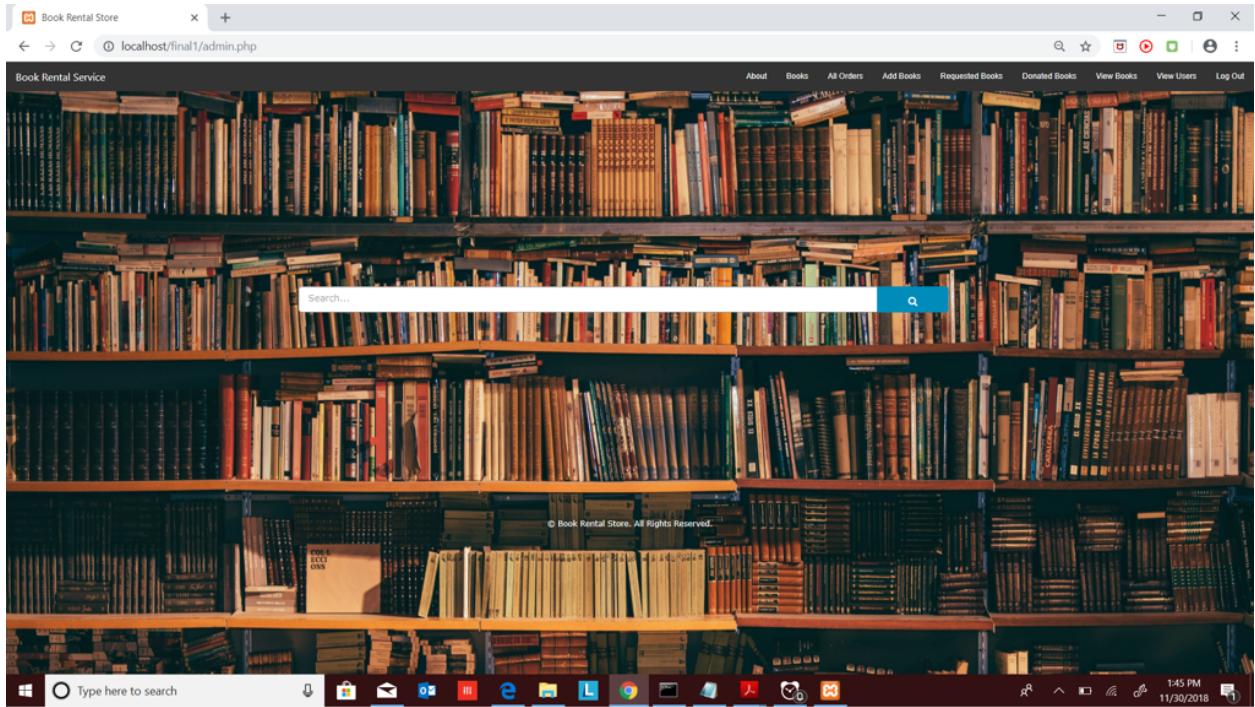
### User Stories Involved:

- As a user, I want to sign-up to my account
- As a guest, I want to create an account in the website.

**Validation:** Passwords must be Alphanumeric.

**Navigation:** ( Register Button → Home screen)

## Admin screen:



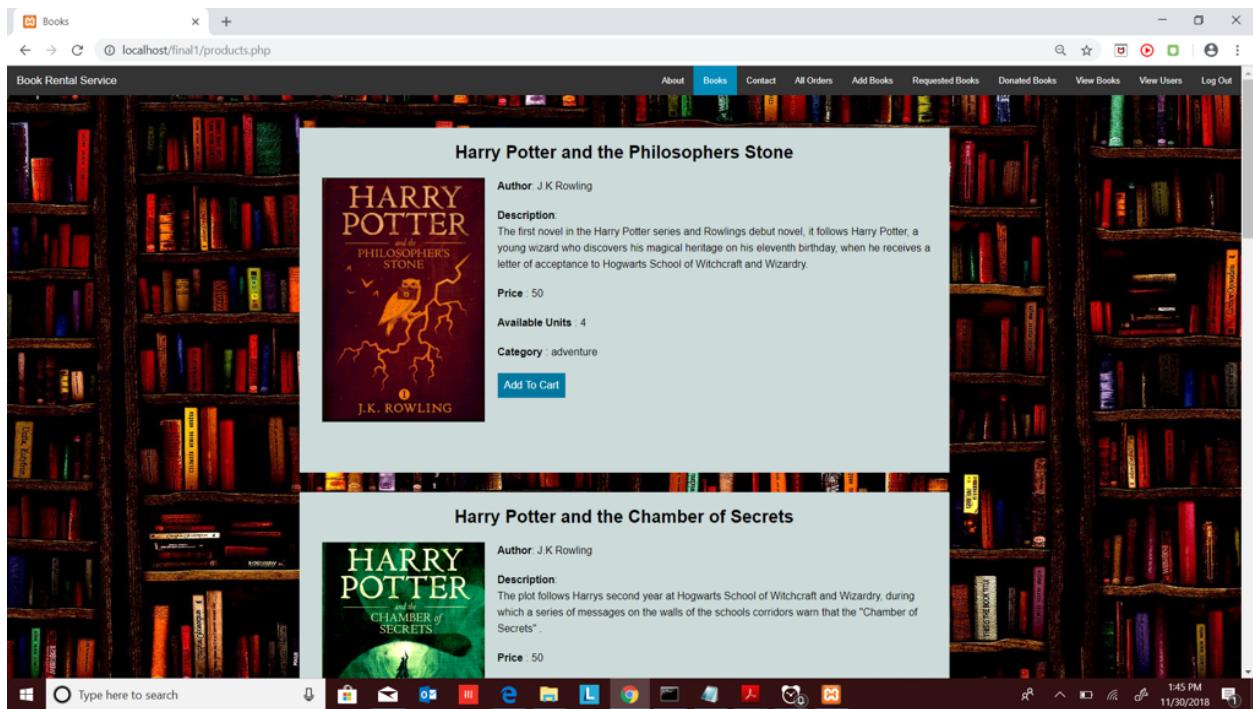
### User Stories Involved:

- As an admin, I want to maintain the user details

**Validation:** Admin must login using login option only.

**Navigation:** ( Login Button → Admin access Home page)

## Products Screen:



### User Stories Involved:

- As a guest, I want to explore all the books that are available in your website.

**Validation:** For every product you add the website will redirect you to cart page, if you want to add extra books you can come back to this page and add again other books to cart.

**Navigation:** ( Add to cart → Cart page)

## Cart screen:

The screenshot shows a web browser window titled "Shopping Cart || Book Rental Ser". The address bar displays "localhost/final1/cart.php". The page header "Book Rental Service" has a dropdown menu with links: About, Books, Contact, All Orders, Add Books, Requested Books, Donated Books, View Books, View Users, and Log Out. Below the header is a table titled "Your Shopping Cart". The table has columns: id, title, Price per week, Time duration (in weeks), and Total cost. One item is listed: "Harry Potter and the Philosophers Stone" at \$50 for 1 week, totaling \$50. At the bottom of the table, it says "Total Amount" and shows \$50. Below the table are three buttons: "Empty Cart" (red), "Continue Shopping" (green), and "COD" (blue).

id	title	Price per week	Time duration (in weeks)	Total cost
1	Harry Potter and the Philosophers Stone	50	1 <span style="color: green;">+ 1</span> <span style="color: red;">- 1</span>	50

Total Amount: 50

[Empty Cart](#) [Continue Shopping](#) [COD](#)

© Book Rental Service. All Rights Reserved.



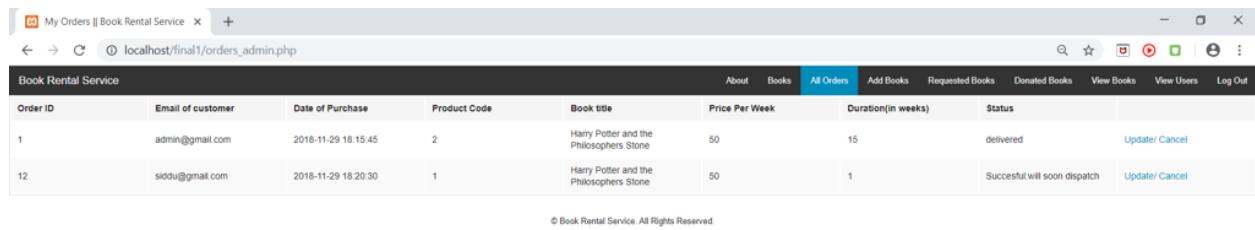
### User Stories Involved:

- As a user, I want to place the order

**Validation:** You can increase the quantity only upto a certain limit.

**Navigation:** (COD → Order will be placed)

## Admin Orders:



Order ID	Email of customer	Date of Purchase	Product Code	Book title	Price Per Week	Duration(in weeks)	Status	Action
1	admin@gmail.com	2018-11-29 18:15:45	2	Harry Potter and the Philosophers Stone	50	15	delivered	<a href="#">Update/ Cancel</a>
12	siddu@gmail.com	2018-11-29 18:20:30	1	Harry Potter and the Philosophers Stone	50	1	Successful will soon dispatch	<a href="#">Update/ Cancel</a>

© Book Rental Service All Rights Reserved.



### User Stories Involved:

- As an admin, I want to keep track of transaction details.

**Validation:** The details will be in a listed sequence with the latest ordered first.

**Navigation:** There is no navigation here.

## Add Books screen:

The screenshot shows a web browser window with the title 'Register || Book Rental Service'. The address bar displays 'localhost/final1/add.php'. The page header 'Book Rentals Service' includes links for About, Books, All Orders, Add Books (which is highlighted in blue), Requested Books, Donated Books, View Books, View Users, and Log Out. The main content area contains fields for Book Title, Author, Edition, Category, Price, and Quantity, each with a corresponding input field. Below these is a 'Product Photo' section with a file input field labeled 'Choose File' and a note 'No file chosen'. A large text area labeled 'Description' is present. At the bottom are two buttons: 'Add Books' (highlighted in blue) and 'Reset'.



### User Stories Involved:

- As an admin, I want to add books to the website.

**Validation:** User must add a pic of the book ,It is essential but not compulsory.

**Navigation:** (Add Books → Home screen)

## View books screen:

Book Rentals Service							<a href="#">About</a>	<a href="#">Books</a>	<a href="#">All Orders</a>	<a href="#">Add Books</a>	<a href="#">Requested Books</a>	<a href="#">Donated Books</a>	<a href="#">View Books</a>	<a href="#">View Users</a>	<a href="#">Log Out</a>
Book Title	Author	Edition	Category	Price	Available Units	Description									
Harry Potter and the Philosophers Stone	J.K Rowling	1	adventure	50	4	The first novel in the Harry Potter series and Rowlings debut novel, it follows Harry Potter, a young wizard, as he begins his magical heritage on his eleventh birthday, when he receives a letter of acceptance to Hogwarts School of Witchcraft and Wizardry.	<a href="#">Edit</a>	<a href="#">Remove</a>							
Harry Potter and the Chamber of Secrets	J.K Rowling	1	adventure	50	7	The plot follows Harrys second year at Hogwarts School of Witchcraft and Wizardry, during which a series of messages on the walls of the schools corridors warn that the "Chamber of Secrets".	<a href="#">Edit</a>	<a href="#">Remove</a>							
Harry Potter and the Prisoner of Azkaban	J.K Rowling	1	adventure	50	6	The book follows Harry Potter, a young wizard, in his third year at Hogwarts School of Witchcraft and Wizardy. Along with friends Ronald Weasley and Hermione Grange, Harry investigates Sirius Black, an escaped prisoner from Azkaban who they believe is one of Lord Voldemort's old allies.	<a href="#">Edit</a>	<a href="#">Remove</a>							
Harry Potter and the Goblet of Fire	J.K Rowling	1	adventure	50	9	It follows Harry Potter, a wizard in his fourth year at Hogwarts School of Witchcraft and Wizardry and the mystery surrounding the entry.	<a href="#">Edit</a>	<a href="#">Remove</a>							
Harry Potter and the Order of the Phoenix	J.K Rowling	1	adventure	3	50	The fifth novel in the Harry Potter series. It follows Harry Potters struggles through his fifth year at Hogwarts School of Witchcraft and Wizardy, including the return of the antagonist Lord Voldemort, O.W.L exams, and an obstructive Ministry of Magic. The novel was published on 21 June 2003 by Bloomsbury in the United Kingdom.	<a href="#">Edit</a>	<a href="#">Remove</a>							
						The sixth and penultimate novel in the Harry Potter series. Set during the summer before Harry's sixth year at Hogwarts, it follows Harry as he prepares for the Triwizard Tournament.									

### User Stories Involved:

- As an admin, I want to manage books.

**Validation:** All the books with same details except edition will be displayed as different rows.

**Navigation:** (Add Books → Home screen)

## Edit Books:

The screenshot shows a web browser window titled "Register || Book Rental Service" with the URL "localhost/final1/edit.php?edit=1". The page has a dark header bar with links for "About", "Books", "All Orders", "Add Books", "Requested Books", "Donated Books", "View Books" (which is highlighted in blue), "View Users", and "Log Out". Below the header is a form for editing book details. The fields are as follows:

ISBN	1
Book Title	Harry Potter and the Philosopher's Stone
Author	J.K.
Edition	1
Category	adventure
Price	50
Available Units	4

Below the form is a "Product Photo" section with a "Choose File" button and a message "No file chosen". Underneath is a "Description" section containing the text: "The first novel in the Harry Potter series and Rowling's debut novel. It follows Harry Potter, a young wizard who discovers his magical heritage on his eleventh birthday, when he receives a letter of acceptance to Hogwarts School of Witchcraft and Wizardry."

A blue "Update" button is located at the bottom left of the form area. At the very bottom of the page, a small note reads "© Book Rental Service. All Rights Reserved."



### User Stories Involved:

- As an admin, I want to manage the prices of the books.
- As an admin, I want to remove books from the website.

**Validation:** All the books with same details except edition will be displayed as different rows.

**Navigation:** (Update → Books)

### Contact details screen:

The screenshot shows a web browser window titled "Contact || Book Rental Service". The URL in the address bar is "localhost/final1/contact.php". The page content includes the website's header "Book Rental Service" and a navigation menu with links for About, Books, Contact (which is highlighted in blue), Cart, My Orders, Donate Book, Request Book, My Account, and Log Out. Below the menu, the text "Software Engineering Project" and "International Institute of Information Technology, Bangalore" is displayed. A note says "Contact the authors through GitHub". Below this, four names are listed: Rohith Yogi, Siddarth Reddy, Srujan Swaroop, and Puneeth. At the bottom of the page is a copyright notice: "© Book Rental Service. All Rights Reserved."



### User Stories Involved:

- As an admin, I want to manage the prices of the books.
- As an admin, I want to remove books from the website.

**Validation:** Clicking on the contact person's details will redirect to their GitHub accounts.

**Navigation:** No Navigation.

## Donate Books screen:

The screenshot shows a web browser window with the title 'Register || Book Rental Service'. The address bar displays 'localhost/final1/donate.php'. The page header 'Book Rentals Service' includes links for About, Books, Contact, Cart, My Orders, Donate Book (which is highlighted in blue), Request Book, My Account, and Log Out. Below the header are five input fields for Book Title, Author, Edition, Category, and Quantity. A large text area labeled 'Description' follows. At the bottom are 'Donate' and 'Reset' buttons, and a small copyright notice: '© Book Rental Service. All Rights Reserved.'



### User Stories Involved:

- As a user, I want to donate my books.

**Validation:** Books can be donated by users who are logged in only and must be accepted by admin to be found in website.

**Navigation:** (Donate → Books accepted message) (Reset → set all values empty to rewrite)

## Request screen:

The screenshot shows a web browser window titled "Register || Book Rental Service" with the URL "localhost/final1/request.php". The page has a dark header bar with links for About, Books, Contact, Cart, My Orders, Donate Book, Request Book (which is highlighted in blue), My Account, and Log Out. Below the header is a form for requesting a book. The form includes fields for Book Title (Operating systems), Author (Thomas), Edition (6), Category (9), and a large Description text area. At the bottom are two buttons: "Request" (in blue) and "Reset". A small copyright notice at the bottom right reads "© Book Rental Service. All Rights Reserved."



### User Stories Involved:

- As a user, I want to request for a book if not found.

**Validation:** Books can be requested by users who are logged in only and must be accepted by admin to be found in website.

**Navigation:** (request → Books request message sent response) (Reset → set all values empty to rewrite)

## Update Account:

The screenshot shows a web browser window titled "My Account || Book Rental Service". The URL is "localhost/final1/account.php". The page header includes links for About, Books, Contact, Cart, My Orders, Donate Book, Request Book, My Account (which is highlighted in blue), and Log Out. The main content area starts with a greeting "Hi siddu". Below it is a section titled "Account Details" with a note: "Below are your details in the database. If you wish to change anything, then just enter new data in text box and click on update." There are six input fields for First Name (siddu), Last Name (vivek), Address (earth), City (blr), Pin Code (3423), and Email (siddu@gmail.com). A password field is also present. At the bottom are "Update" and "Reset" buttons, and a footer note: "© Book Rental Service. All Rights Reserved."



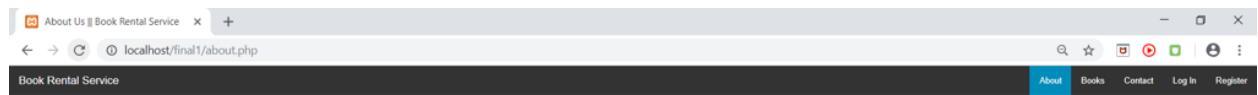
### User Stories Involved:

- As a user, I want to update my details if incorrect.
- As an admin, I want to maintain the user details.

**Validation:** email can't be changed.

**Navigation:** (Update → Home page) (Reset → set all values empty to rewrite)

### About screen:



Online Book Rental Service is developed by our  
group in the software engineering course under  
**Prof B.Thangaraju sir**.

© Book Rental Service. All Rights Reserved.



### User Stories Involved:

- As a user, I want to know about the website.

### Accept request screen:

The screenshot shows a web browser window titled "Register || Book Rental Service" with the URL "localhost/final1/accept\_req.php?req=8". The page is titled "Accept request screen:" and displays a form for updating a book record. The form fields are as follows:

ISBN	8
Book Title	Operating
Author	Thomas
Edition	6
Category	9
Price	
Available Units	1

Below the form, there is a "Product Photo" section with a "Choose File" button and a note "No file chosen". There is also a large "Description" text area. At the bottom of the form is a blue "Update" button.

At the top of the browser window, there is a navigation bar with links: About, Books, All Orders, Add Books, Requested Books, Donated Books, View Books, View Users, and Log Out. The "Books" link is highlighted.



### User Stories Involved:

- As an admin, I want to accept requested books

**Validation:** Requested books will be added only if they are good.

**Navigation:** (Update → Books page)

## Update order:

The screenshot shows a web application titled "Book Rentals Service". The URL in the address bar is "localhost/final1/update\_order.php?edit=1". The page has a navigation menu with links: About, Books, All Orders (which is highlighted in blue), Add Books, Requested Books, Donated Books, View Books, View Users, and Log Out. Below the menu is a table with the following data:

Order ID	Date of Purchase	Product Code	Book title	Price Per Week	Duration(in weeks)	Status
1	2018-11-29 18:15:45	2	Harry Potter and the Philosophers Stone	50	15	delivered

At the bottom left is a blue "Update" button. At the bottom right, there is a copyright notice: "© Book Rental Service. All Rights Reserved." The system tray at the bottom of the screen shows various icons and the date/time as 11/30/2018 1:54 PM.

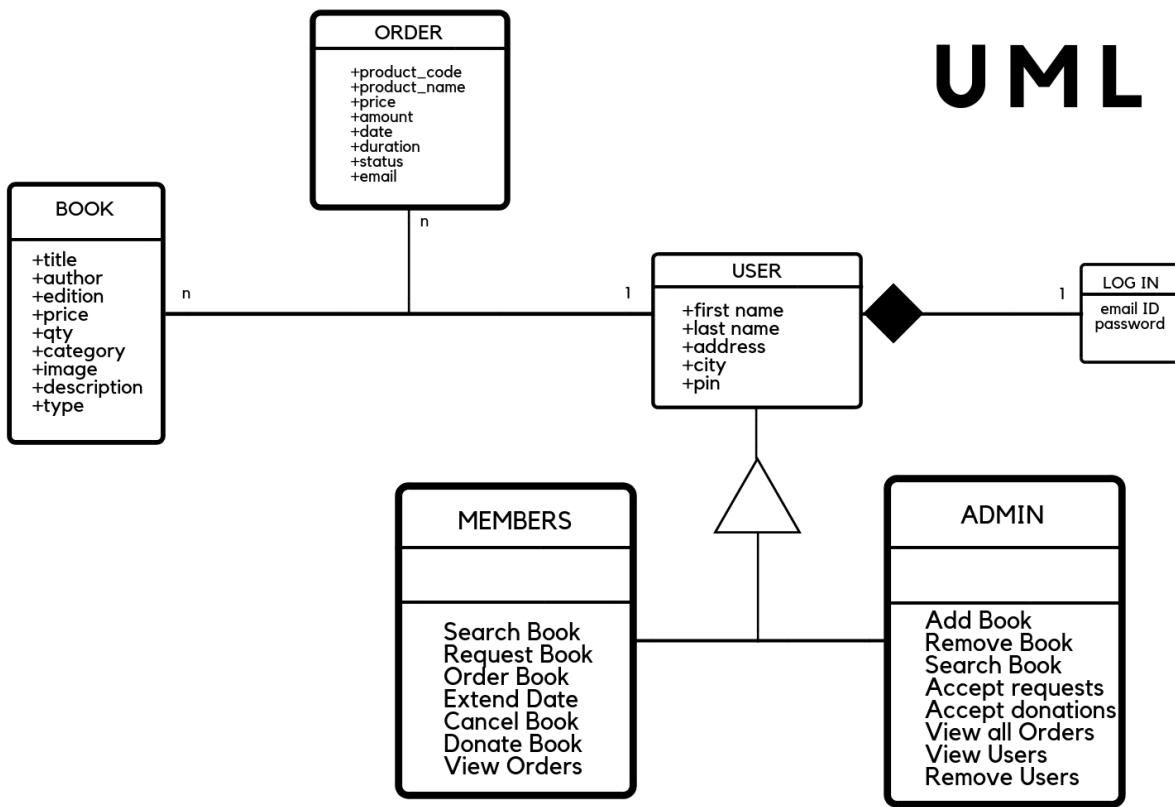
### User Stories Involved:

- As an admin, I want to update the tracking details.
- As an admin, I want to cancel or extend the delivery

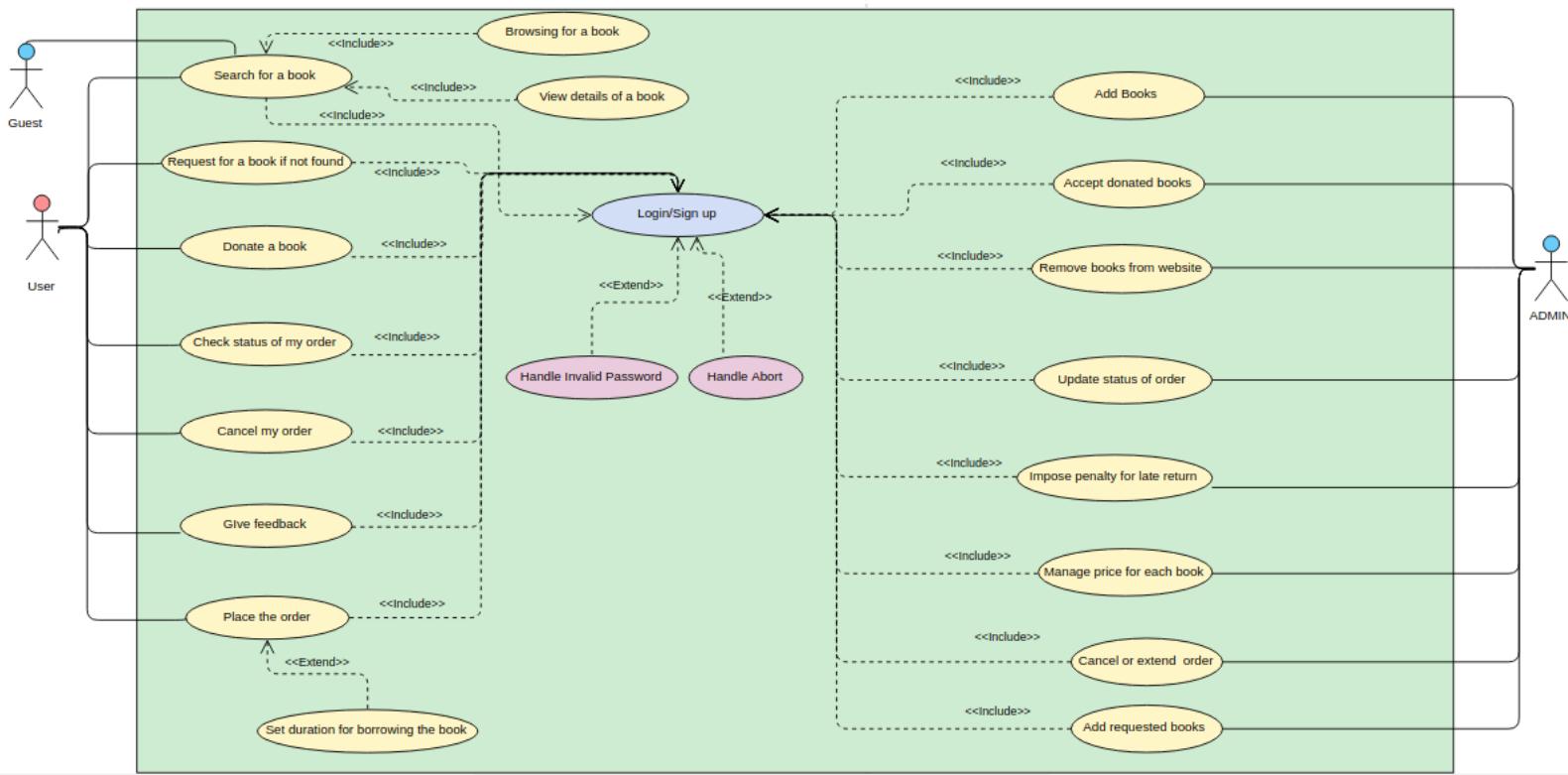
**Validation:** Updates will be notified to user via their account.

**Navigation:** (Update → Home page)

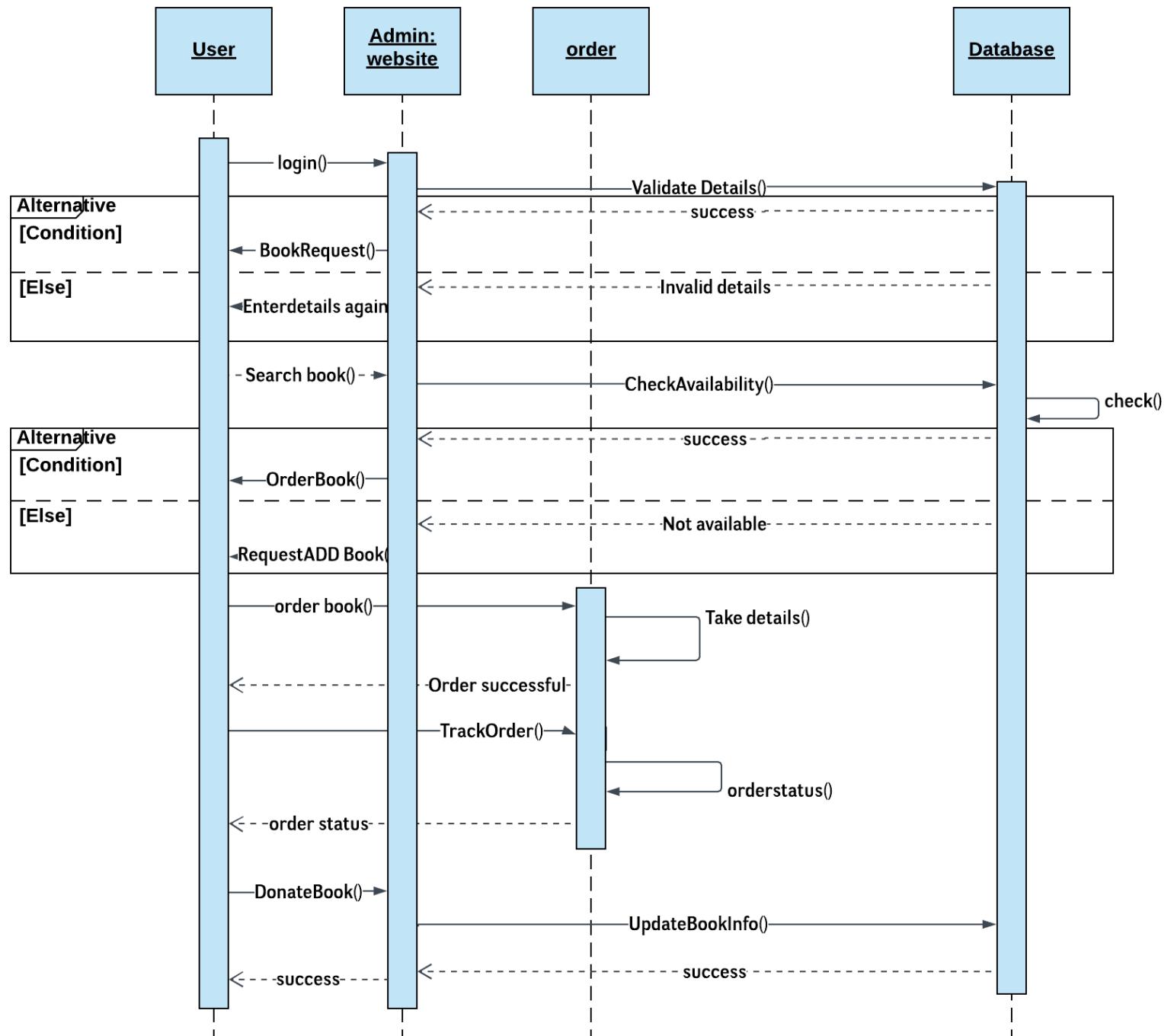
# UML

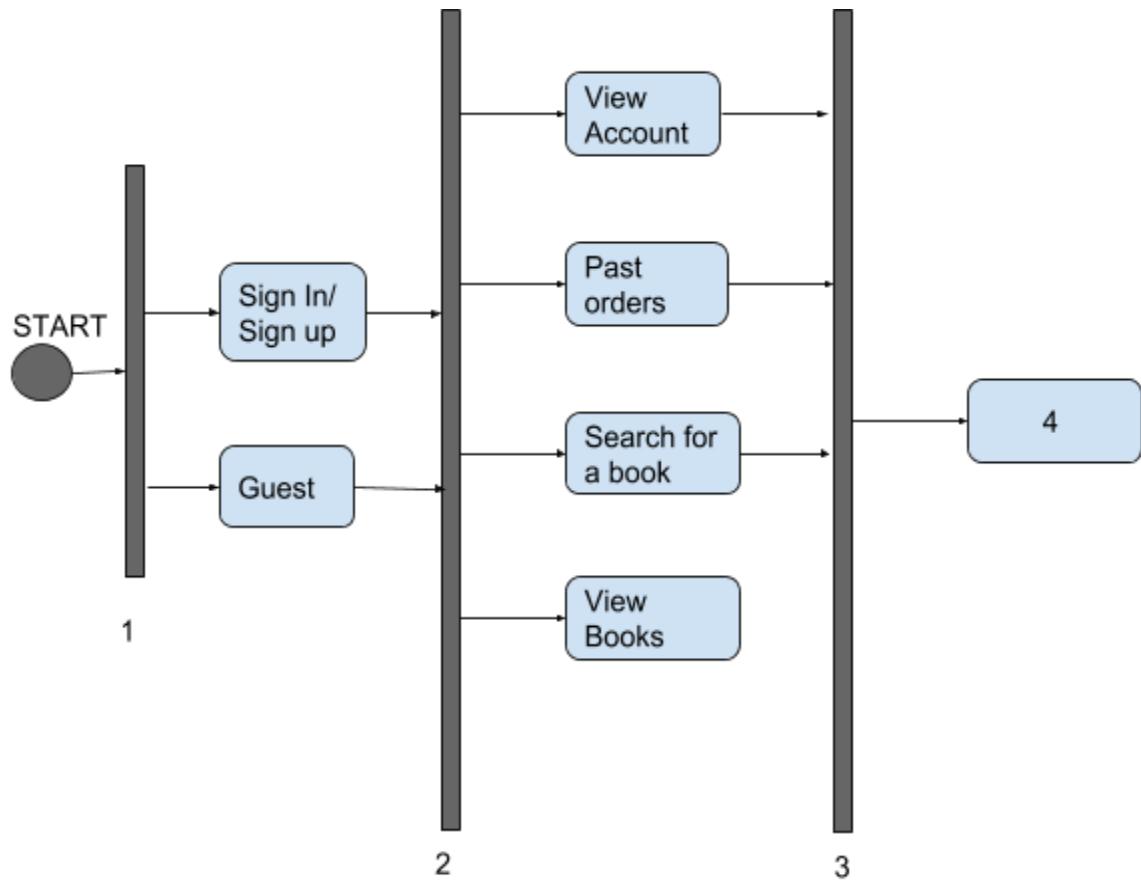


# Use Case Diagram



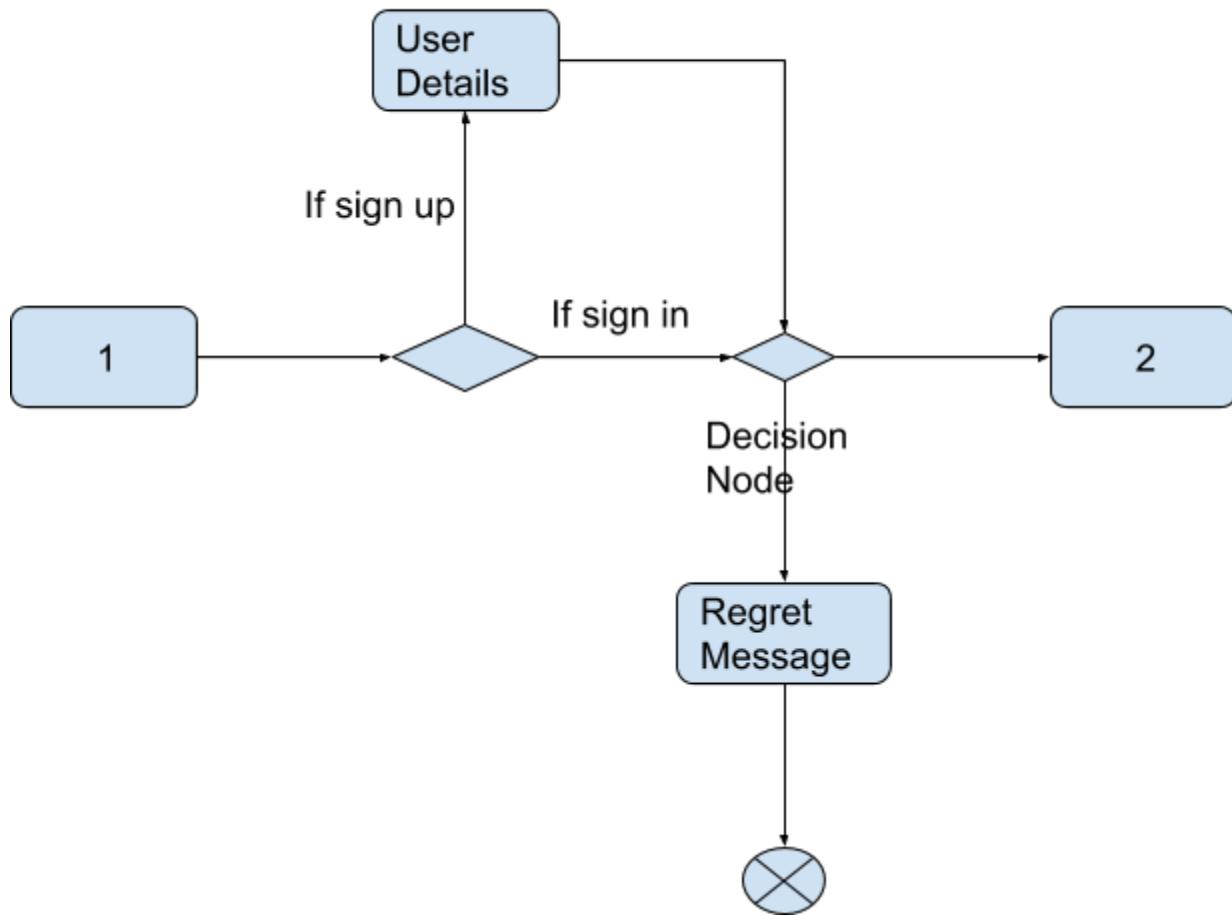
# Sequence Diagram



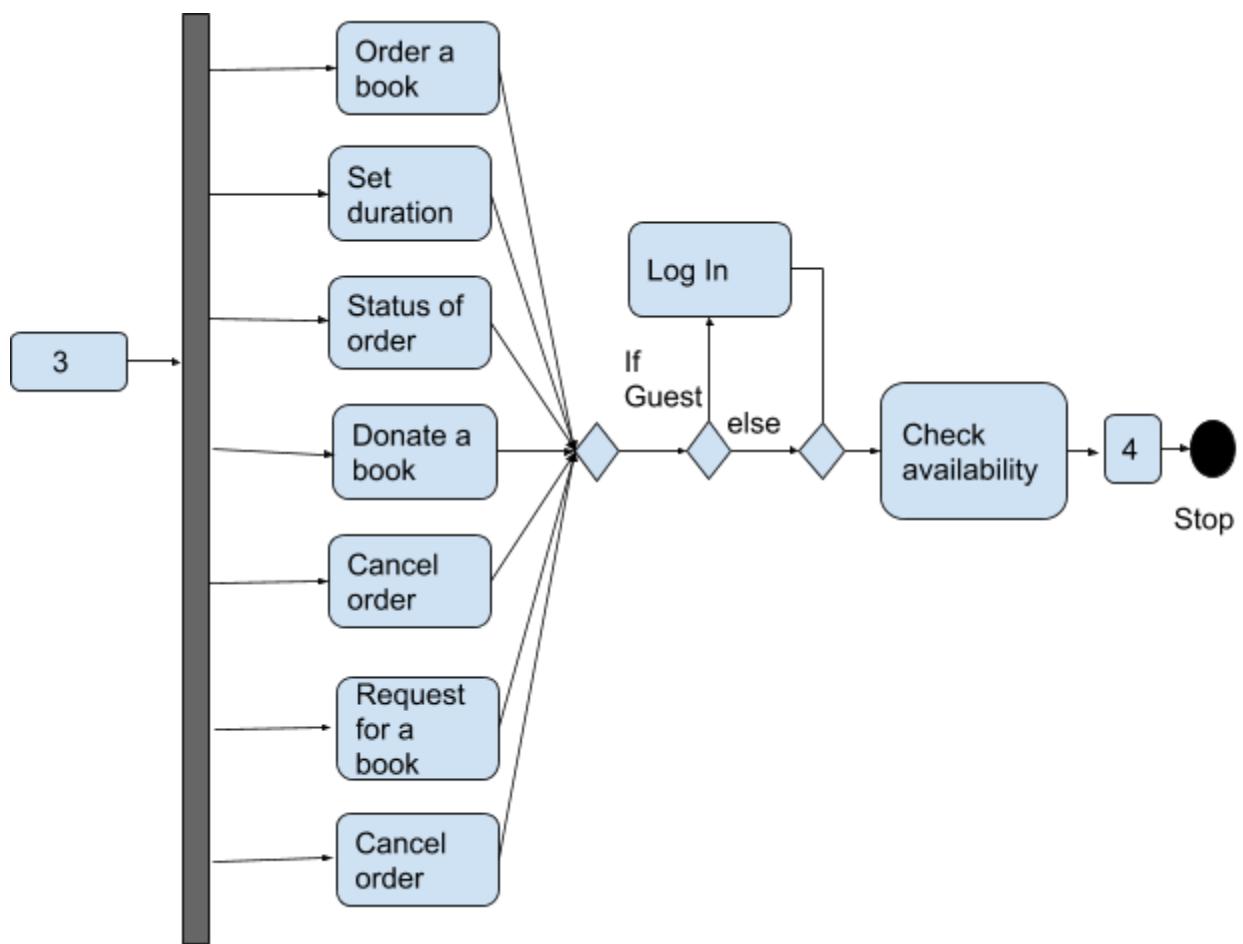


Activity Diagram 1

## ACTIVITY DIAGRAM 2 :



Activity Diagram 2



Activity Diagram 3

We create two new projects in jenkins which contains build and deploy our code.

The screenshot shows the Jenkins interface for the 'Book Rental Build' project. The left sidebar contains various Jenkins management icons. The main content area is titled 'Project Book Rental Build'. It features a 'Workspace' section with a folder icon and a 'Recent Changes' section with a notebook icon. Below these are sections for 'Downstream Projects' (listing 'BOOK\_RENTAL\_DEPLOY') and 'Permalinks' (listing recent build links). On the right, there are buttons for 'add description' and 'Disable Project'. The URL in the browser bar is 'localhost:8080/job/Book%20Rental%20Build/'.

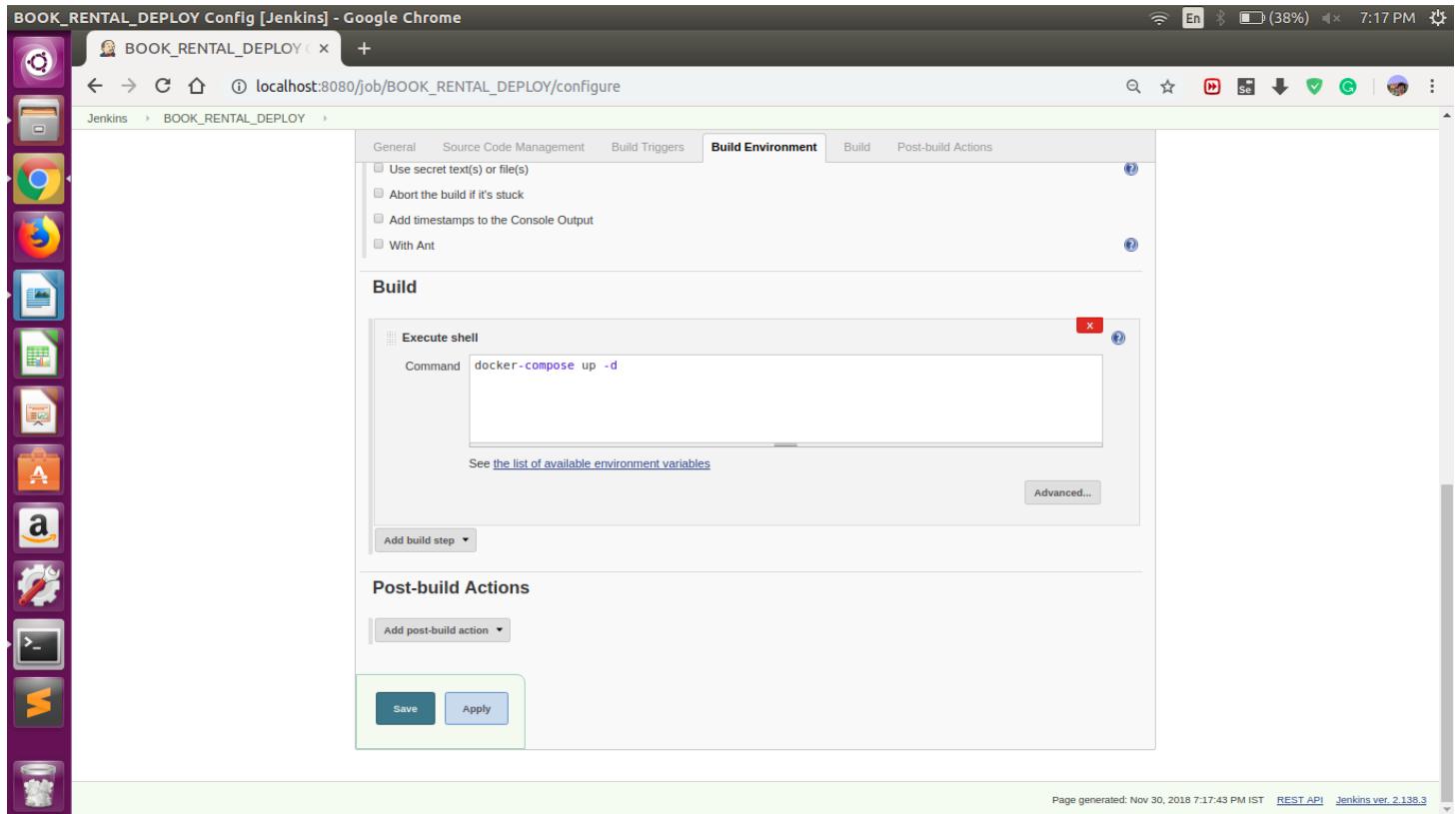
In “BOOK\_RENTAL\_BUILD” ,we integrate our project with github by giving our github credentials to jenkins and repository which contains our files.

The screenshot shows the Jenkins configuration interface for a job named "Book Rental Build". The "Source Code Management" tab is selected, showing the "Git" repository configuration. The "Repository URL" is set to [https://github.com/Srujan-Swaroop/Software\\_project.git](https://github.com/Srujan-Swaroop/Software_project.git) and the "Credentials" are set to "Srujan-Swaroop\*\*\*\*\*". The "Branches to build" section specifies "Branch Specifier (blank for 'any') /master". The "Build Triggers" section includes "Trigger builds remotely (e.g., from scripts)", "Build after other projects are built", and "Build periodically". The "Post-build Actions" section is currently empty.

In post build actions we select “build another projects” and add “BOOK\_RENTAL\_DEPLOY”,this means after collecting our code from github we ask jenkins to run this post build action.

The screenshot shows the Jenkins configuration interface for the same job. The "Build Environment" tab is selected, and the "Post-build Actions" section is highlighted. It shows a "Build other projects" action configured to build the project "BOOK\_RENTAL\_DEPLOY". The "Trigger only if build is stable" option is selected. The "Post-build Actions" section also includes "Add post-build action" and "Save" and "Apply" buttons.

In “BOOK\_RENTAL\_DEPLOY” ,we ask to execute docker files using the command we given in execute shell.



So these are the outputs of both the projects.

For BUILD :

Book Rental Build #13 Console [Jenkins] - Google Chrome

Book Rental Build #13 Co Book Rental Store +

localhost:8080/job/Book%20Rental%20Build/13/console

Jenkins | admin | log out

Back to Project Status Changes Console Output View as plain text Edit Build Information Delete Build Git Build Data No Tags Previous Build

## Console Output

```
Started by user admin
Building in workspace /home/srujan/Jenkins/workspace/Book Rental Build
> git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/Srujan-Swaroop/Software_project.git # timeout=10
Fetching upstream changes from https://github.com/Srujan-Swaroop/Software_project.git
> git --version # timeout=10
using GIT_ASKPASS to set credentials
> git fetch --tags --progress https://github.com/Srujan-Swaroop/Software_project.git
+refs/heads/*:refs/remotes/origin/*
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
> git rev-parse refs/remotes/origin/origin/master^{commit} # timeout=10
Checking out Revision 2f59993c736993224ed42a8f520a8def636a1cle (refs/remotes/origin/master)
git config core.sparsecheckout # timeout=10
> git checkout -f 2f59993c736993224ed42a8f520a8def636a1cle
Commit message: "Update Book_Rental.sql"
> git rev-list --no-walk 2f59993c736993224ed42a8f520a8def636a1cle # timeout=10
Triggering a new build of BOOK_RENTAL_DEPLOY
Finished: SUCCESS
```

Page generated: Nov 30, 2018 6:21:53 PM IST REST API Jenkins ver. 2.138.3

For DEPLOY :

BOOK\_RENTAL\_DEPLOY #14 Console [Jenkins] - Google Chrome

BOOK\_RENTAL\_DEPLOY Book Rental Store +

localhost:8080/job/BOOK\_RENTAL\_DEPLOY/14/console

Jenkins | admin | log out

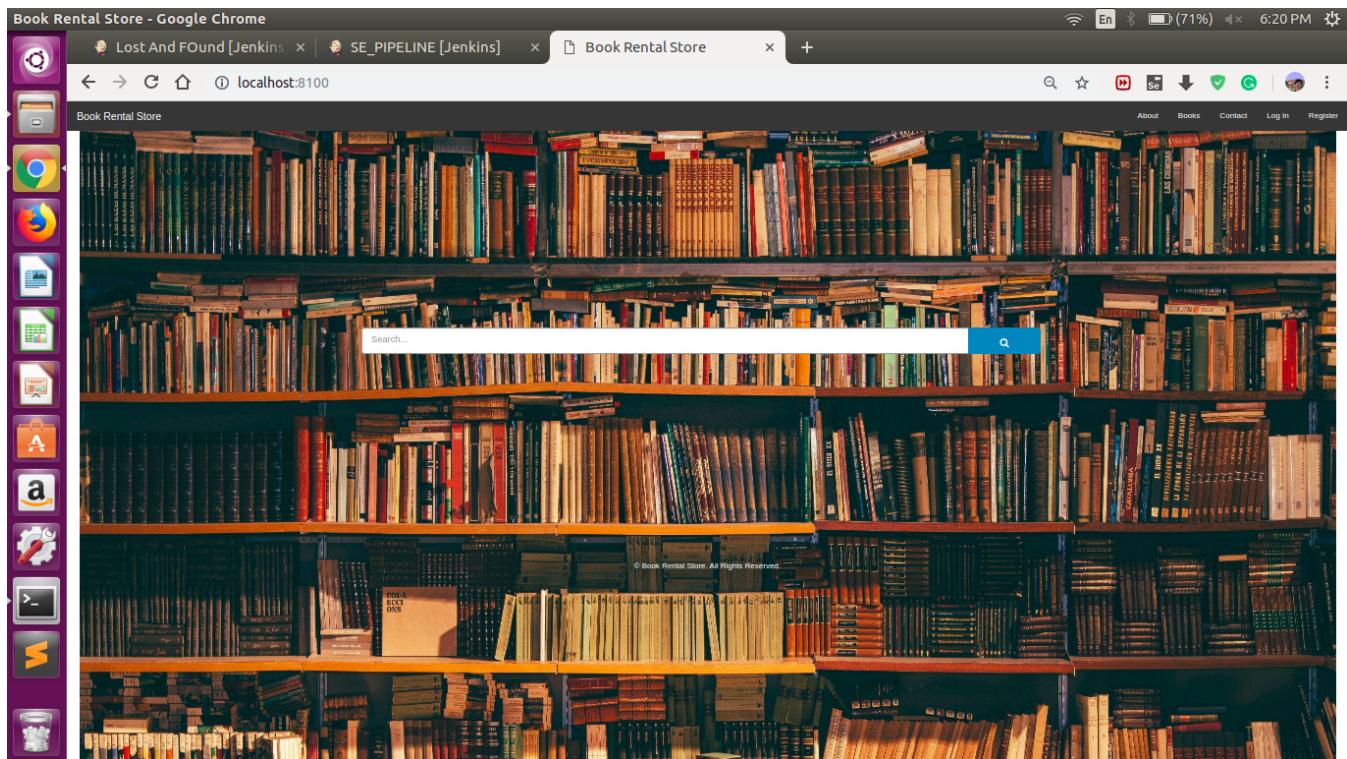
Back to Project Status Changes Console Output View as plain text Edit Build Information Delete Build Previous Build

## Console Output

```
Started by upstream project "Book_Rental_Build" build number 13
originally caused by:
Started by user admin
Building in workspace /home/srujan/Jenkins/workspace/Book Rental Build
[Book Rental Build] $ ./bin/sh -xe /tmp/jenkins1600818089693799271.sh
+ docker-compose up -d
bookrentalbuild_db_1 is up-to-date
php_web is up-to-date
Finished: SUCCESS
```

Page generated: Nov 30, 2018 6:22:14 PM IST REST API Jenkins ver. 2.138.3

So, after deploying we can access website using  
<http://localhost:8100/>



pipelining our project is :

