

**BOOKIE – EBOOK SYSTEM**

**Software Requirement Specification**

– Hanoi, September 2022 –

**Contents**

[I. Overview 4](#_Toc119093868)

[1. Introduction 4](#_Toc119093869)

[2. System Functions 5](#_Toc119093870)

[a. Screen Flow 5](#_Toc119093871)

[b. Screen Details 5](#_Toc119093872)

[c. User Authorization 6](#_Toc119093873)

[d. Non-Screen Functions 8](#_Toc119093874)

[3. Common Requirements 8](#_Toc119093875)

[II. Functional Requirements 9](#_Toc119093876)

[1. Homepage 9](#_Toc119093877)

[2. User authorization 10](#_Toc119093878)

[2.1. Login 10](#_Toc119093879)

[2.2. User register 11](#_Toc119093880)

[2.3. Forget Password 12](#_Toc119093881)

[2.4. Logout 13](#_Toc119093882)

[2.5. Change password 14](#_Toc119093883)

[2.6. Change email 15](#_Toc119093884)

[3. User Information 16](#_Toc119093885)

[3.1. View profile (User) 16](#_Toc119093886)

[3.2. Update profile (User) 17](#_Toc119093887)

[3.3. View profile (Admin) 18](#_Toc119093888)

[3.4. Update profile (Admin) 19](#_Toc119093889)

[3.5. View security status 20](#_Toc119093890)

[4. User Management 21](#_Toc119093891)

[5. Books 21](#_Toc119093892)

[5.1. Manage Books 21](#_Toc119093893)

[5.2. View list of books 23](#_Toc119093894)

[5.3. View book’s description 24](#_Toc119093895)

[5.4. Reading interface 26](#_Toc119093896)

[5.5. Manage Table of Contents 28](#_Toc119093897)

[5.6. Report Book 29](#_Toc119093898)

[5.7. Rate book 30](#_Toc119093899)

[5.8. Update Library’s View (Search Engine + Paging) 31](#_Toc119093900)

[5.9. List book of this author 33](#_Toc119093901)

[6. Novels 34](#_Toc119093902)

[6.1. View/Add/Update Novels 34](#_Toc119093903)

[6.2. Add/Edit/Delete Novels 36](#_Toc119093904)

[6.3. Add/Edit/Delete chapter for user’s novel 38](#_Toc119093905)

[7. View/Add Author 41](#_Toc119093906)

[8. View/Add/Edit/Remove Category 43](#_Toc119093907)

[a. View 43](#_Toc119093908)

[b. Remove 43](#_Toc119093909)

[c. Update 44](#_Toc119093910)

[d. Add 45](#_Toc119093911)

[9. Add to/Delete from Favourite 45](#_Toc119093912)

[10. Comment 47](#_Toc119093913)

[10.1 Add/Reply/Edit/Delete Comment 47](#_Toc119093914)

[10.2. Report Comment 49](#_Toc119093915)

[10.3. Ban Comment 50](#_Toc119093916)

[11. View/Close/Reject Reports 52](#_Toc119093917)

[12. Transaction 54](#_Toc119093918)

[12.1. Purchase a book 54](#_Toc119093919)

[12.2. Purchase novel’s chapter 55](#_Toc119093920)

[12.3. Payment 56](#_Toc119093921)

[12.4. Report Transactions 59](#_Toc119093922)

[13. View author’s information 61](#_Toc119093923)

[14. Bookshelf 62](#_Toc119093924)

[15. Ticket 63](#_Toc119093925)

[16. Admin Panel 64](#_Toc119093926)

[17. Notification 64](#_Toc119093927)

# I. Overview

## 1. Introduction

The Bookie – eBook System is a new web system that replaces the current library and all kind of it. Nowadays, people are getting more busy and lazier. They don’t have enough time or don’t want to go to the library to find a book or a shop to buy one. The system is expected to prevent the dead of knowledge come from books. Bookie want to create a brand new online-library that people can have access anytime, anywhere.

Main actors:

* Administrator
* Customers: Normal person and authors

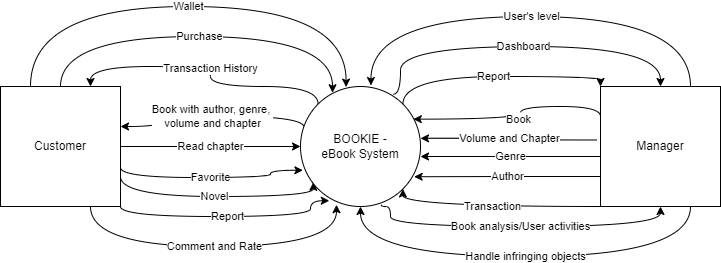


Figure : Context diagram

## 2. System Functions

### a. Screen Flow

Diagram

Description automatically generated

Figure : Screen Flow

### b. Screen Details

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Common | Home Page | Default screen where all users can visit. There is information about the system, contact of developer. The site can lead to login page and register page |
| 2 | Common | User Register | Screen where users can create a new account with personal information. Click Submit to submit new account |
| 3 | Common | Login | Screen where the user logs into the system with the registered account. If the account or password is wrong, the system will notify the user |
| 4 | Common | Change Password | Allow users who want to change their password account |
| 5 | Common | User Profile | Display information about the user who is logged in to the system |
| 6 | Admin | Dashboard | Display the dashboard that helps administrators manage the system. |
| 7 | Common | Forgot Password | Allow user to reset password using registered email in case user forgot password |
| 8 | Customers | Top Favourite | Display books having the most users adding to favour |
| 9 | Customers | Weekly sales | Display books chosen with special discount weekly |
| 10 | Customers | Book List | Display all books/novels |
| 11 | Customers | View Book’s Information | Display information about name, authors, category, description, price, index… of the book |
| 12 | Customers | Library | Display all books/novels customers purchase |
| 13 | Customers | Wallet | Display balance in customer’s wallet |
| 14 | Customers | Purchase | Allow customers to purchase a book/chapter with their wallet |
| 15 | Customers | Read content | Allow customers to read the book/novel they purchased |
| 16 | Customers | Comment | Allow customers to write their comment in book’s section |
| 17 | Customers | Rating | Allow customers to rate a book/novel on a 5-star scale |
| 18 | Customers | Searching | Allow customers to search for specific book by enter key words |
| 19 | Customers | Report a purchase | Allow customers to report defective work or problematic transactions to request a refund/exchange |
| 20 | Customers | Report a book | Allow customers to report a book that violates the community standards |
| 21 | Customers | Create a novel | Allow authors to write and post their own work (must satisfy special rules for self-work) |
| 22 | Customers | Paid mode | Allow authors to open the charge for their work |
| 23 | Admin | Manage Category | Allow admins who want to add, edit and delete category of books |
| 24 | Admin | Manage Book | Allow admins who want to view, add, edit and delete book’s information |
| 25 | Admin | Manage Novel | Allow admins who want to view/ban user’s novel |
| 26 | Admin | Manage Comment | Allow admins who want to view and delete if there are any inappropriate/offensive comments |
| 27 | Admin | Manage User | Allow admins who want to up user to admin or ban any excessive user |
| 28 | Admin | View transactions | Display all purchase in the system |
| 29 | Admin | View reports | Allow admins who want to receive and handle any report by users |
| 30 | Common | Change email | Allow users who want to change their email |

Table : Screen Details

### c. User Authorization

|  |  |  |
| --- | --- | --- |
| **Screen** | **Admin** | **Customers** |
| Home Page | X | X |
| User Register | X | X |
| Customers Login | X | X |
| Admin Login | X |  |
| User Profile | X | X |
| Admin Screen | X |  |
| Forgot Password | X | X |
| Top Favourite | X | X |
| Weekly sales | X | X |
| Book List | X | X |
| View Book’s Information | X | X |
| Library | X | X |
| Wallet |  | X |
| Purchase | X | X |
| Read content | X | X |
| Comment | X | X |
| Rating | X | X |
| Searching | X | X |
| Report a purchase |  | X |
| Report a book |  | X |
| Create an OW |  | X |
| Paid mode |  | X |
| Manage Category | X |  |
| Manage Book | X |  |
| Manage OW | X |  |
| Manage Comment | X |  |
| Manage User | X |  |
| Change email | X | X |
| Change Access Level | X |  |

Table : User Authorization

In which:

* Admin: The manager of the organisation, has access to all system functions, decides every access to the system.
* Customers: People who want to read book, novels and do the simple function of an eBook system. They can also be an author to write their own books, novels to make money

### d. Non-Screen Functions

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **System Function** | **Description** |
| 1 | Common | User Authorization | Implement the authorization mechanism into the system. Using both frontend and backend code to prevent users from accessing links without proper role. |

Table : Non-screen functions

## 3. Common Requirements

* Websites must be accessed according to user authorization.
* Day format must be DD/MM/YYYY.
* Font used: San-serif.
* Font normal size: 14.
* Font header size: 20.

# 

# II. Functional Requirements

## 1. Homepage

* Function trigger: when user visits the page or when they click on the logo.
* Function description:
* Actor: Users and Guests
* Purpose: Landing page for all users
* Interface:
* A logo can access Homepage
* A fixed Menu collapsed
* An introduction slide show
* A quote about book
* Photos of featured product
* Highlight/latest blogs
* Contact interface
* Screen Layout:



Figure : Homepage

* Function Detail: When access website, user can see overview of page. The fixed Menu when open can see all the options that mention in Logout function. Slides Show part contains 3 pictures. Clicking on “See more” or “Read more”, user can see all of books or blogs. The contact carry information to connect to Administrators.

## 2. User authorization

### 2.1. Login

* Function trigger: Click “Login” on pop up home page
* Function description:
* Actor: Users (include administrators and customers, staff)
* To access all function in website
* Users that have not yet registered cannot log in. They must sign up to create a new account.
* Function detail:
* Users shall login to the system by using the username and password which is registered (Characters require).
* Users shall be able to use system functionalities and be directed to the formal screen after successful login.
* The Users will receive “errors” to the login screen if the username or password is invalid.
* Screen layout:



Figure : Login

* 2 input text fields
* 1 button “Sign In” to head into the home page of the system
* 1 button “forget password” to head the users to the take back account screen
* 1 button “sign up” to help users create their accounts
* Function details:
* All Categories can be chosen to search book by category
* If click to a Book, users can see the Book details

### 2.2. User register

* Function trigger: User clicks the “Sign Up” on Log In display.
* Function description:
* Actor: Guests
* Purpose: Register for the page
* Interface:
* 7 fields that users have to fill in all: Email, Username, Password. Phone, Full Name in the text style, Gender in option style and Birthday in date style.
* 1 link head back to the Login screen
* Screen Layout

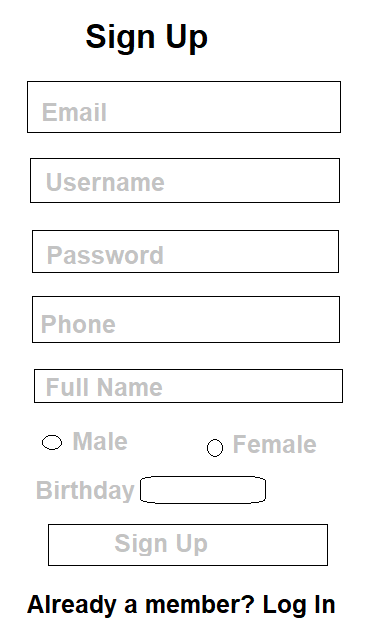


Figure : User register

* Function Detail:
* All fields must be filled up with data
* Email and phone must be unique
* Phone: start with 0, have 10 letters of number
* Email: must have @
* After click “Sign Up”, if all fields are validated and account created successfully, Users will head back to the Login Screen
* Now the user can use the newly created account to login.

### 2.3. Forget Password

* Function trigger: When users click the “Forget Password” in the Login Screen
* Function description:
* Actor: Users
* Purpose: Help users find back their account by receiving an instruction mail sent into the email they provided when they created an account.
* Interface:
* An input field for email linked to account that user want to retake
* A button to process
* Screen Layout:

Text, letter

Description automatically generated

Figure : Forget password

* Function details:
* Users have to fill the form with an existing email that he/she had registered while they created the account
* When he/she clicks the “Process” button, system will check the email they entered, check if it is existing in the database or not
* If it is a valid email, an instruction with recover link will be sent to that email

### 2.4. Logout

* Function trigger: After logged In, user clicks on Menu and then selects Logout. The manager can also logout by the logout button in the Management screen.
* Function description:
* Actor: All user
* Purpose: Exit the account access status
* Interface: A button at the bottom of the menu (appeared after log in)
* Screen Layout:

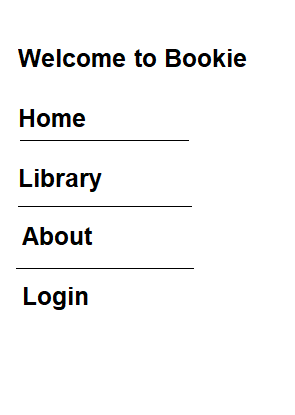


Figure : Before Log out Figure : After log out

* Function Detail: When a user clicks the “Logout” option, its session will be destroyed. The user will be directed to the Homepage.

### 2.5. Change password

* Function trigger: This use case starts when the user clicks on “Change” beside password level in the security screen.
* Function description:
* Role: User
* Purpose: Change password
* Interface:
  + - 3 text boxes to enter password, new password and confirm password
    - 2 button “Cancel” or “Confirm”
* Function layout:

Graphical user interface, text, application

Description automatically generated

Figure : Change password

* Function detail:
* When the user chooses to change password, this pop-up will open. User enters the current password to authorise this action, then enters the new password 2 times.
* Then choose confirm to change password.
* If the user does not want to change password, click “Cancel”.
* This action will fail if the current password is wrong, or confirm the password does not match with the new password, or if the new password is invalid. New password is invalid when:
* It is empty.
* Its length is less than 6 characters or more than 32 characters.
* It starts with a space.

### 2.6. Change email

* Function trigger: This use case starts when the user clicks on the “Change” button beside linked email in the security screen.
* Function description:
* Role: User
* Purpose: Change email
* Interface:
* Current email
* 2 textboxes to enter new email and password
* 2 buttons “Cancel” or “Confirm”
* Function layout:



Figure : Change email

* Function detail:
* When the user chooses to change email, this pop-up will open. The user enters a new email, then enter current password to authorise this action. Then choose to confirm to change email.
* If the user does not want to change email, click “Cancel”.
* The user cannot confirm if the new email is invalid or if the password is wrong. Email is invalid when:
* It is empty.
* It has a space character.
* It is not in the form of something@domain.

## 3. User Information

### 3.1. View profile (User)

* Function trigger: This use case starts when a user clicks on the navigation bar button on top right of the website and clicks “My profile”.
* Function description:
* Role: User
* Purpose: User view their profile
* Interface:
* A fixed menu contains: My profile, Security, Settings, Support
* A logo access homepage
* A navigation
* An interface head “My profile” contains fields: Name, Email, Birth, Gender, Phone Number, Address
* Screen layout:

Graphical user interface, application, table

Description automatically generated

Figure : View Profile (User)

* Function detail:
* Displays the user's profile including his user id, username, name, date of birth, gender, email, phone, address.
* From this screen, the user can choose to edit his profile.

### 3.2. Update profile (User)

* Function trigger: This use case starts when the user clicks on the “Edit” button in the view profile screen.
* Function description:
* Role: Customers
* Purpose: Update user’s profile
* Interface:
* 2 input text boxes for to enter name, and address
* 1 input date to enter date of birth
* 1 input number to enter phone number
* 2 radio buttons to choose gender
* 2 button “Cancel” and “Save”
* Function layout:

Graphical user interface, table

Description automatically generated

Figure : Update Profile (User)

* Function detail:
* The user can see his current information and change them (except user id, username and email). All fields cannot be empty except the address.
* Phone number must be a string of 10 digits.
* The name only has letters and space characters. Name cannot be longer than 40 characters.
* The user can choose “Save” to save new information or choose “Cancel” to go back to the view profile screen without any change.

### 3.3. View profile (Admin)

* Function trigger: Dashboard page click “Profile”
* Function description:
* Actor: Admin/Staff
* Purpose: Admin/Staff can view account’s profile
* Interface:
* 5 fields are Email, Username, Password, Phone, Full Name
* “Sex” field in choice field
* Date of Birth in date field
* Screen layout:

Graphical user interface, text

Description automatically generated

Figure : View Profile (Admin)

* Function detail: N/A

### 3.4. Update profile (Admin)

* Function trigger: User click button Update profile in Admin’s profile screen
* Function description:
* Actor: Admin/Staff
* Purpose: Admin can update profile if he/she need to
* Interface:
* 5 fields are Email, Username, Password, Phone, Full Name
* “Sex” field in choice field
* Date of Birth in date field

● Screen layout:

Graphical user interface, text

Description automatically generated

Figure : Update Profile (Admin)

* Function detail:
* There is a list of data from the admin account.
* Admin can check and rewrite what they want   
  Username and password: character required  
  Phone: 9 - 11 numbers required  
  Full Name: character required, space included
* Click the “Update” button to save the new details
* Notice updating successfully

### 3.5. View security status

* Function trigger: This use case starts when the user clicks on the “Security” tab in the left sidebar.
* Function description:
* Role: User
* Purpose: View account’s security
* Interface: 2 change buttons to change password or change email
* Function layout:

Graphical user interface, text, application

Description automatically generated

Figure : View security status

* Function detail:
* The user can see his email and password level of the account.
* Password level show how strong current password is:
* If the password just has numbers or alphabetic characters, the password level is weak.
* If the password has both alphabetic characters and numbers, the password level is medium.
* If the password has alphabetic characters, numbers and special characters, the password level is strong.

## 4. User Management

* Function trigger: When admin click Users Management title
* Function description
* Roles: Admin
* Purpose: Let admin can see the list of users account detail below which can have some recent actions
* Interface:
* 1 button for ban user account
* 2 up down button to adjust the account level
* 1 field for information searching
* Screen Layout:

Graphical user interface, application

Description automatically generated

Figure : User Management

* Function Detail:
* The screen will display the users account detail
* Depend on admin’s level, it will display all the accounts which have the lower level
* Admin can adjust the level of that account from: normal user to author or admin
* Admin can ban the account
* Admin can search, sort by each title

## 5. **Books**

### 5.1. Book Management

* Function trigger: When users click “Books” panel in the Management Panel
* Function description:
* Actor: Admin
* Purpose: Admin can add new books or disable old ones. It is also possible to edit or update the information of the student.
* Interface:
* View all book:
  + Button: Book Management, Add Book, Update, TOC (Tables of Contents), Disable
  + Table includes Book ID, Title, Author, Image, Category, Price, Sale Status
* Detail book:
  + Button: Add/Update
  + Inputs field for Book’s information includes: Title, Author, Genre, Price, Sale, Summary, Image
* Screen Layout:

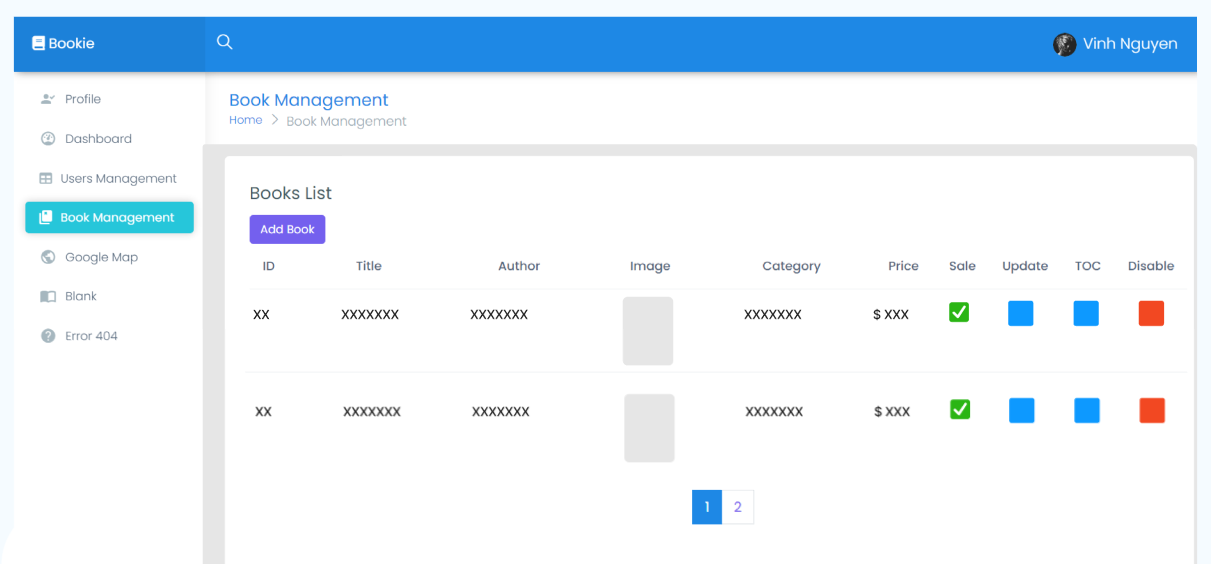
****

Figure : Book Management

* Function detail:
* Click on Add Book button to add new books, there will be information text boxes which are needed to add a book.
* If you want to disable the book, click button at Disable column at the end of that row
* If you want to edit the information of the book, click button at Update column
* Users shall enter the required fields:
* Title (required 1 - 200 characters)
* Author (required 1-100 characters or an existed author is chosen)
* Genre (check at least 1 checkbox)
* Price (enter a positive float number)
* Sale (checked or not)
* Summary (required 1-2000 characters)
* Image (required 1-2000 characters)
* If any of the text fields are left blank the user should see error messages.

### 5.2. View list of books

* Function trigger: When admin click the “Book Manager” in the Admin Panel
* Function description:
* Actor: Admin
* Purpose: See overview of all books
* Interface:
* 1 list of Categories for users to choose
* List books by category or all kind of books
* Screen Layout:

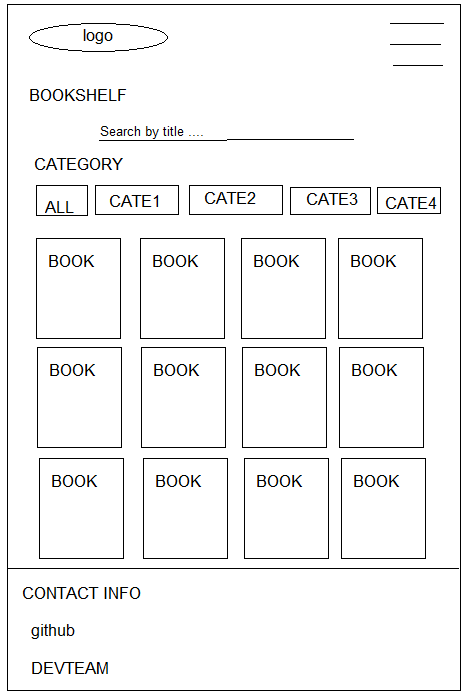


Figure : View list of books

* Function details: In Book Manager page, the interface will display all books. Admin can search for book they want to find.

### 5.3. View book’s description

* Function trigger: When users click any position of an image of a book
* Function description:
* Actor: Admin/User
* Purpose: View information of a book
* Interface:
* 1 button GET: send book to user’s carts where user can pay money to buy books
* 1 button Add to Favourite: send this book to user’s favourite list
* 1 button Report: help users report the book that violate the rules
* 1 textbox: give space for user to type their comment
* 1 button to help user post their comments
* 1 list of check boxes that represent as star: help user vote/rate this book
* Screen Layout:

Diagram

Description automatically generated

Figure : Book Detail

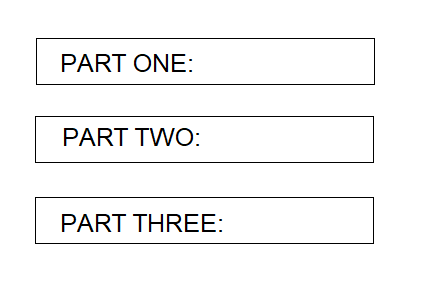
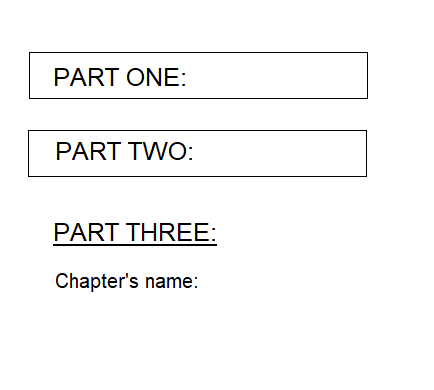


Figure : Table of Contents (extended) Figure : Table of Contents (collapsed)

* Function detail:
  + When user click on a book, this page will display. It has all information of book. User can purchase, add to favourite or report this book. They can give comment for everyone about this book.
  + The Table of Contents only be displayed after user purchase for the book. When they click on Chapter’s name, it will redirect to the book-reading interface. The book-reading page displays the content of the chapters that the user has chosen.

### 5.4. Reading book

* Function trigger: when user (has paid for the book) click on chapters
* Function description:
* Actor: Users
* Purpose: Setting interface for reading book
* Interface:
* A logo can access Homepage
* A fixed Menu collapsed
* A header of book’s name
* A header of Chapter
* Content of chapter
* For reading: A fixed navigation has option: back/next, home, setting interface
* Screen Layout:

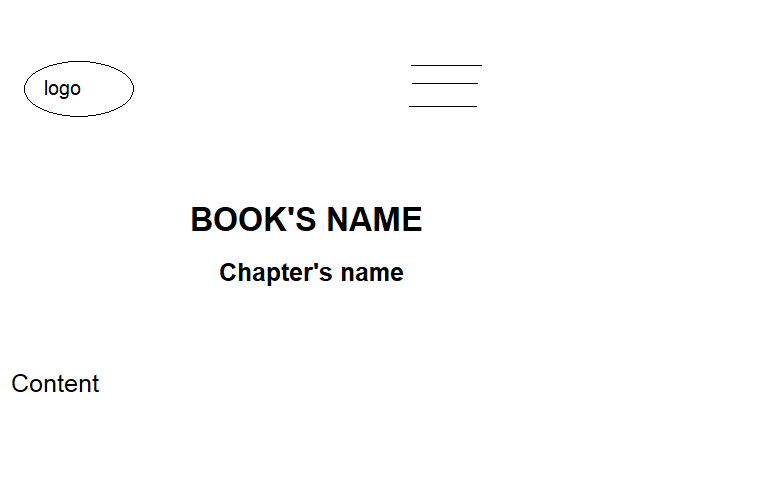


Figure : Book prereading

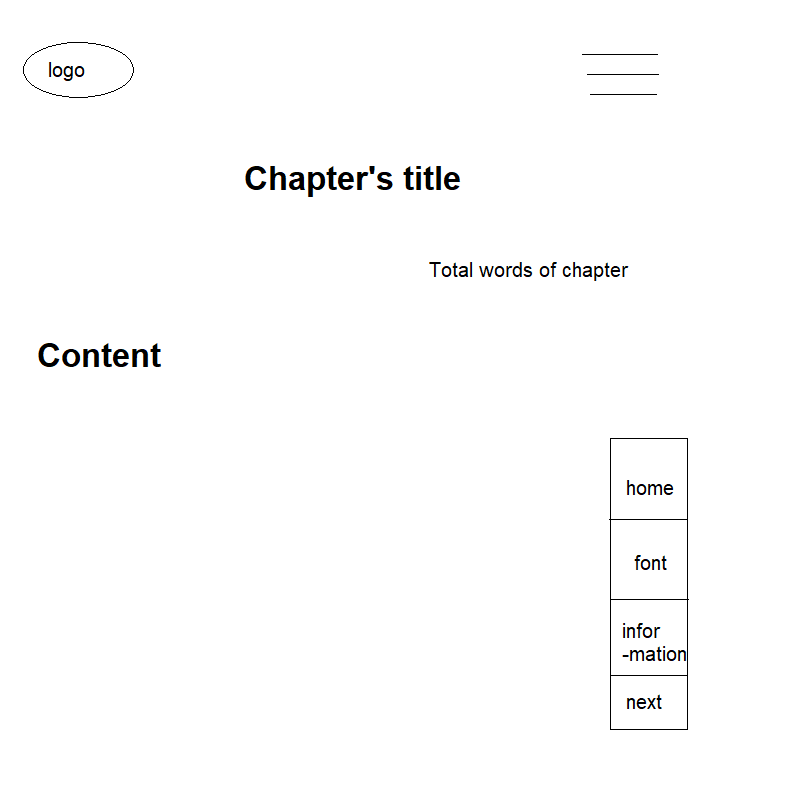


Figure : Book reading interface

* Function details:
  + Before purchasing a book, the user can read the first chapter before deciding to purchase or not. If user hasn’t paid for book yet, table of contents only direct to pre-read chapter.
  + After purchasing for books, the user can read all chapters from Table of Contents. The book-reading page displays content of the chapters that the user chooses.

### 5.5. Manage Table of Contents

* Function trigger: When users click button in TOC column in Book Management page
* Function description:
* Actor: Admin/Staff
* Purpose: Admin/Staff can add new volumes, new chapters or delete old ones. It is also possible to edit or update the information of the volumes or the chapters
* Interface:
* Volume:
  + Button: Add Volume, Edit Volume, Delete Volume, Save
  + Input fields for Volume’s information: Title, Summary
* Chapter:
  + Button: Add Chapter, Edit Chapter, Delete Chapter
  + Input fields for Chapter’s information: Title, Status, Content
* Screen Layout:

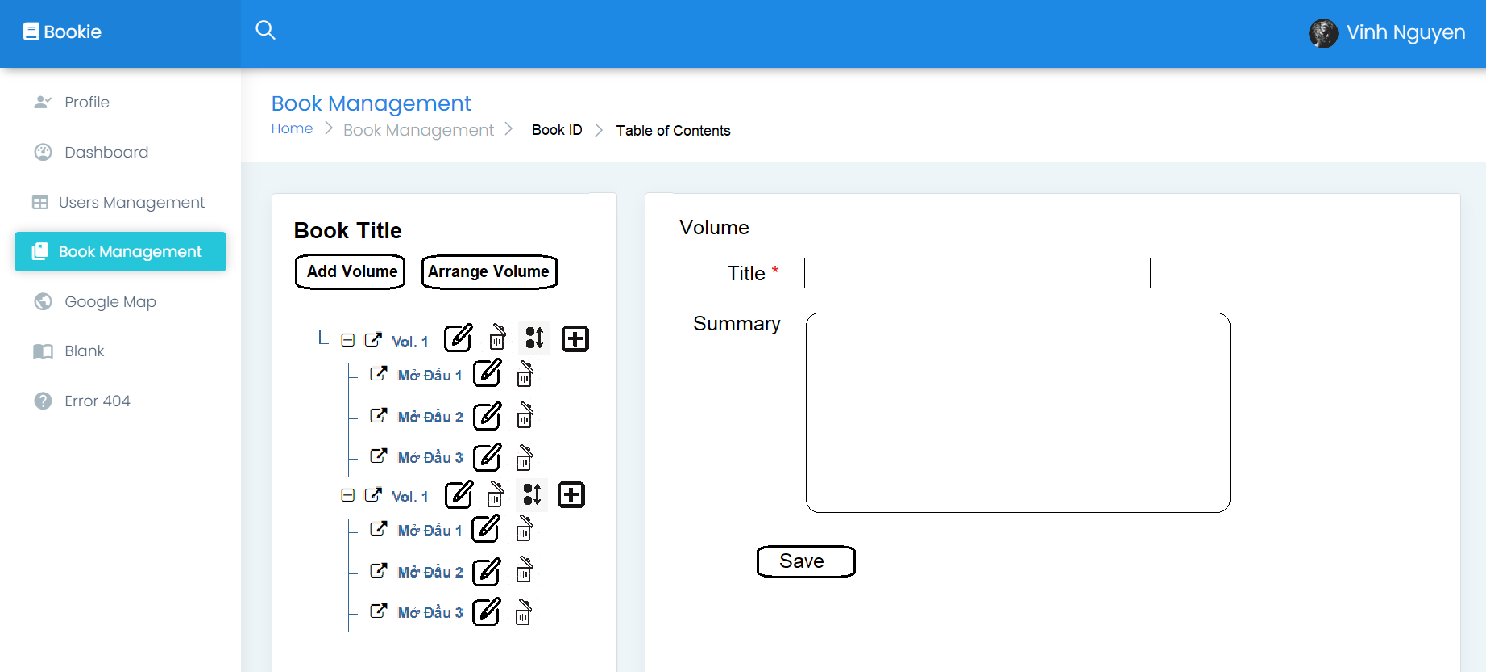


Figure : Volume Management

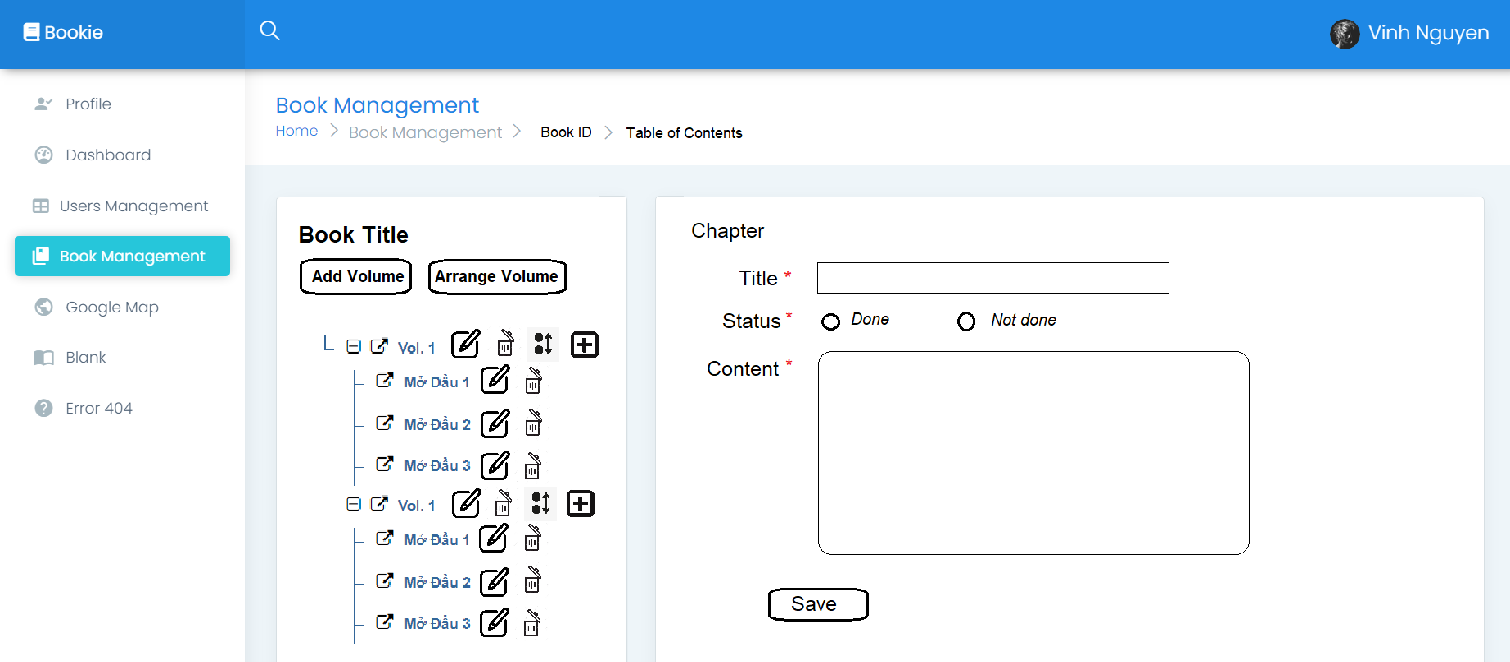


Figure : Chapter Management

* Function detail:
* Click on Add Volume to add new volumes, there will be information text boxes which are needed to add a volume
* To arrange the order of the book’s volumes, click on Arrange Volume
* To delete the volume, click at the end of each volume row, select Delete
* To edit the volume, do the same as above and select Edit
* Click on Add Chapter to add new chapters, there will be information text boxes which are needed to add a chapter, including chapter’s contents
* To arrange the order of the volume’s chapters, click at the end of their volume row, and select Arrange
* To delete the chapter, click at the end of each chapter row, select Delete
* To edit the chapter, do the same as above and select Edit
* For Add/Edit Chapter, user shall enter the required field:
* Title (required 1-200 characters)
* Status (check 1 radio button)
* Content (required at least 1 characters)
* For Add/Edit Volume, user shall enter the required field:
* Title (required 1-200 characters)
* Summary (not required/max to 2000 characters)

### 5.6. Report Book

* Function trigger: When user click the Report button in the Book detail page
* Function description
* Roles: Users
* Purpose: let user send report to admin that the book violated the community’s rules
* Interface: Information of book, contains: Name, Author and Image
  + List of violence type in check box type
  + A button “Report”
* Function Layout:

**Letter

Description automatically generated**

Figure : Report book

* Function Detail:
* The screen displays the book name, author, image
* The checkboxes that store options for user can check it and click the Report button to send it to the admin

### 5.7. Rate book

* Function trigger: When user click on the Star
* Function description:
* Role: Users
* Purpose: Let user rate this book
* Interface:
* 5 stars that represent the rating from 1 to 5
* 1 button to submit rate
* Function Layout:

Diagram

Description automatically generated

Figure : Rate book

* Function Detail: This function will not be shown unless users log in into the system

### 5.8. Update Library’s View (Search Engine + Paging)

* Function trigger: When user click on the Search button in Library and select the book’s category in Genre list (Search Engine), whenever the library shows list of books, the list will be split into pages (Paging)
* Function description:
* Actor: Users
* Purpose:
* Let user search the book of their wants (Search Engine)
* Split the collection of books into pages to make the library more succinct
* Interface:
* Page number in the nearby footage of the page: Click will send users to others pages of library
* List of Category in buttons on the left
* Search Engine: Search by title and Advanced Search
* Screen layout:

Diagram, text

Description automatically generated

Figure : Library

Table

Description automatically generated

Figure : Search Engine

* Function detail:
* Genre list in both outside and inside advanced search will show all the categories that have in the database
* Genre Lists are checkbox, let user choose more than 1 category
* When user click button “Search”, the system will return all the book/novel that meet the requirements in pages (paging)
* In both advanced search and library interface will have textbox for user to enter title of the book (in the library interface) and enter author’s name (in advanced search)
* Each page will have 12 books
* While users are in 1 page, that page’s number will have a different colour

### 5.9. List book of this author

* Function trigger: User click on the book, it will automatically display list books of that author
* Function description:
* Actor: Users
* Purpose: See all book of a authors
* Interface:
* A logo to access homepage
* A fixed Menu collapsed
* A header named “From this author”
* All books from the same author, 3 books per row
* Contact interface
* Screen Layout

Diagram

Description automatically generated

Figure : List book of this author

* Function details: When the user clicks on the author's name, the system redirects to a new page to display all books that have the same author that they have chosen.

## 6. Novels

### 6.1. View/Add/Update Novels

* Function trigger: When users click the “My Novels” in the User Profile screen
* Function description:
* Actor: Customer
* Purpose: Customer can create his/her own novel or delete/disable old ones. It is also possible to edit or update the information of the novel.
* Interface:
* View Novels:
  + Button: Create, Update, TOC (Tables of Contents), Disable
  + Table includes Title, Image (default if no input), Genres, Paid Status
* Add Novel:
  + Button: Create
  + 6 fields: Title, Genre, Summary, Price, Paid, Image
* Update Novel:
  + Button: Update, Delete
  + 6 fields: Title, Genre, Summary, Price, Paid, Image
* Screen Layout:

**Graphical user interface

Description automatically generated with low confidence**

Figure : View Novel

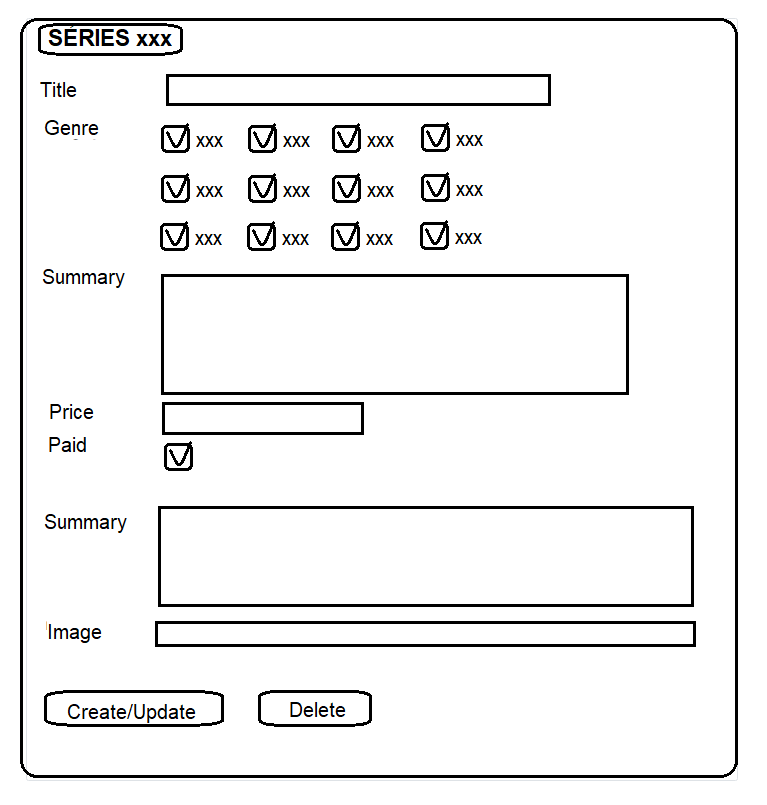


Figure : Manage Novel

* Function detail:
* View Novels:
  + Click on “Create!” button to create a new novel, there will be information text boxes which are needed to add a novel.
  + If you want to edit the information of the novel, click the button at the Update column. In the update screen, if the novel hasn’t been bought by any customer, “Delete” button can be displayed and clicked to be permanently deleted.
* Add/Update Novel:
* Users shall enter the required fields:
  + Title (required 1 - 200 characters)
  + Genre (check at least 1 checkbox)
  + Price (enter a positive float number) (only available if customer reaches level 3)
  + Sale (checked or not) (only available if customer reaches level 3)
  + Summary (required 1-2000 characters)
  + Image (not required/up to 2000 characters)
* If customers reach level 3, the field for Price and Paid status can be displayed and altered by themselves.

### 6.2. Add/Edit/Delete chapter for user’s novel

* Function trigger: User clicks on the “+” button, system will display the interface that user can add content for chapter.
* Function description:
* Actor: Users
* Purpose: Create a new chapter
* Interface:
  + Add chapter:
* A logo to access homepage
* A fixed Menu collapsed
* A fixed User menu
* A catalogue box to add new chapters. “Title” and “Content” are in text-form, Status is in option-form. All fields are required.
* Edit chapter:
* A logo to access homepage
* A fixed Menu collapsed
* A fixed User menu
* A header “Series of” this book
* A div tag that display information of chapter, contains chapter’s name, status, and content, in disable mode
* Two buttons “Edit” and “Delete”
* Delete chapter:
* A logo to access homepage
* A fixed Menu collapsed
* A fixed User menu
* A catalogue box contains the book's name, option add chapter and back to book-detail page.
* A div tag that display information of chapter, contains chapter’s name, status, and content, in disable mode
* Two buttons “Edit” and “Delete”
  + Alert pop-up
    - Message confirm decision
    - Two buttons “Close” and “Delete”
* Screen Layout

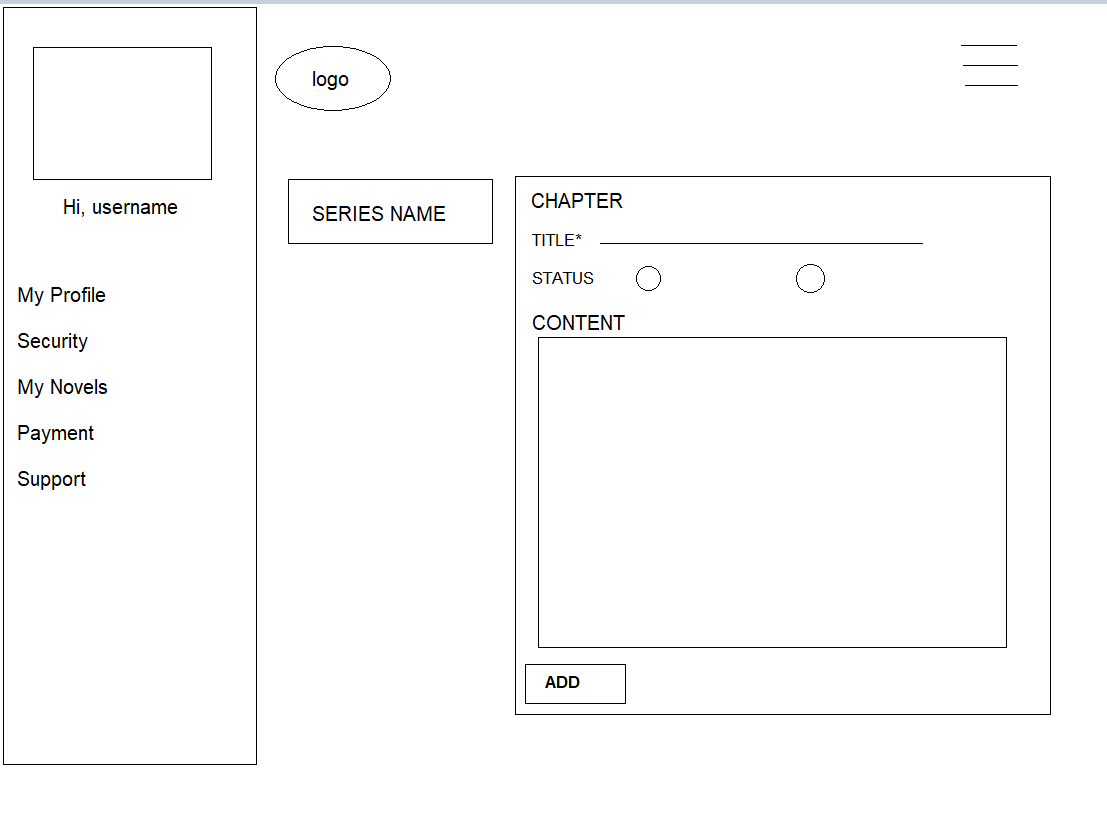


Figure : Before add chapter

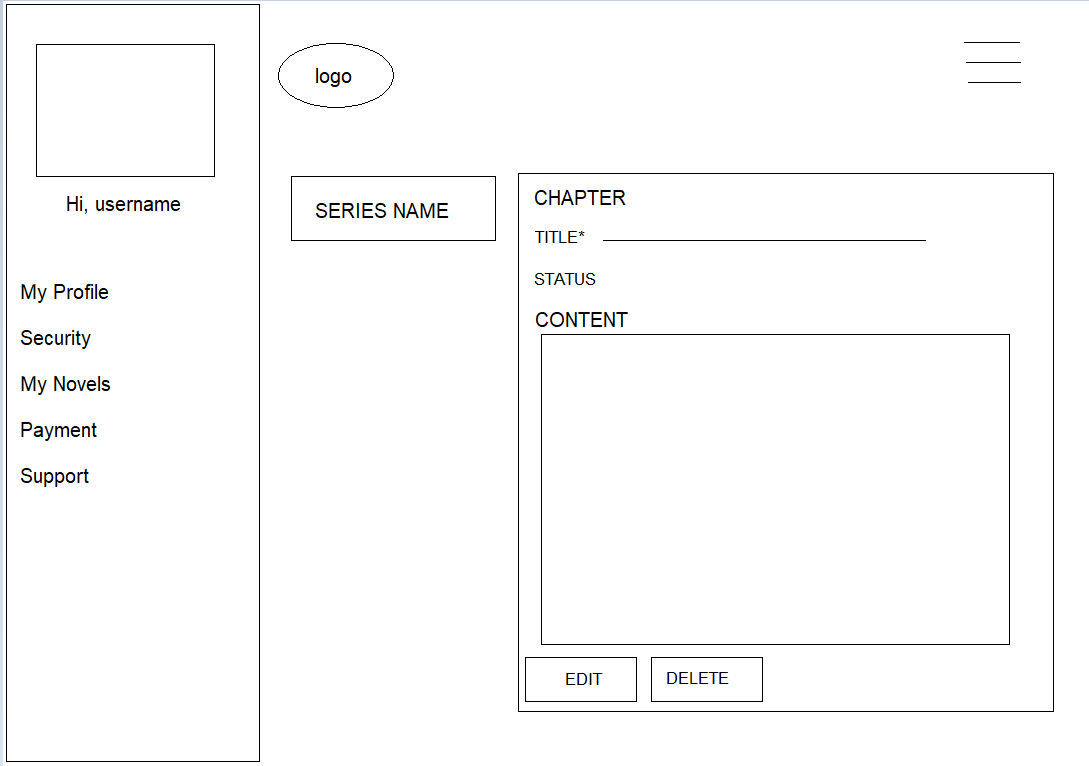


Figure : After chapter

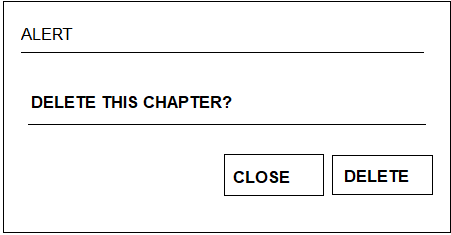


Figure : Confirm pop-up

* Function details:
* Add chapter: When the user clicks on the “Add” button, the create chapter interface appears. After filling in all fields of the pop-up, the pop-up is changed to disable mode and appear to be buttons “Edit” and “Delete”.
* Edit chapter: User clicks on the name of the chapter, the pop-up appears with two options “Edit’ and “Delete”. All fields are in disable mode. The user clicks on “Edit” to change information about the chapter. After choosing “Edit”, all fields are allowed to edit. To finish work, the user re-click on the “Edit” button and then all fields are back to disable mode.
* Delete chapter: User clicks on the name of the chapter, the pop-up appears with two options “Edit’ and “Delete”. All fields are in disable mode. The user clicks on “Delete” to remove the chapter. After choosing “Delete”, a message warning is displayed. User click on “Delete” to confirm, or “Close” to recall decision.

## 7. View/Add Author

* Function trigger: When admin click Author Management title or page button
* Function description
* Roles: Admin
* Purpose: let admin can see the list of authors below
* Interface:
* 1 button for update author
* buttons for paging
* 2 field for information searching
* 1 button “add” for adding new author
* Screen Layout:

Graphical user interface

Description automatically generated

Figure : View List Author

Graphical user interface

Description automatically generated with medium confidence

Figure : Edit Author

* Function Detail:
  + The screen will display the authors-generated and authors-users detail
  + Paging with 10 authors each page
  + Admin click on “ADD” button to add author
  + The screen will display the authors-generated and authors-users detail
  + Each table has different search and will return the list after enter  
    Input Requires: characters, space include
  + Input new name and click “button” on the right will update the new name for that author  
    Input Requires: characters, space include
  + Click on the id of author will return a list of books belong to that author
  + Click “Add” button send to Add-book page

## 8. View/Add/Edit/Remove Category

### a. View

* Function trigger: This use case starts when the user clicks on “Category management” on the left side bar in the profile screen.
* Function description:
* Role: Admin
* Purpose: View list of categories
* Interface:
* 1 button to add new category
* 1 button to edit name for each category
* 1 checkbox to activate/deactivate for each category
* Function layout:

Table

Description automatically generated

Figure : View Category

* Function detail:
* Admin can see all of the categories and how many books in each category.
* The user can edit category name, activate/deactivate category or add new category.

### b. Remove

* Function trigger: This use case starts when user click on garbage can icon in the line of category that they want to remove
* Function description:
* Role: Admin
* Purpose: Remove a category
* Interface: A pop-up has confirmed message with two buttons “Cancel” and “Ok”
* Function layout:

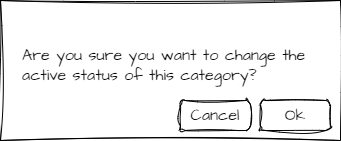


Figure : Remove confirm pop-up

* Function detail:
* When the admin clicks on the active checkbox, a pop-up appears to confirm his request.
* If the admin clicks “Ok”, change the active status of this category. Otherwise, close the pop-up without any change.

### c. Update

* Function trigger: This use case starts when user click on pen icon next to category’s name
* Function description:
* Role: Admin
* Purpose: Rename category
* Interface:
* 1 textbox to enter a new category’s name.
* 2 button to cancel or confirm the change
* Function layout:

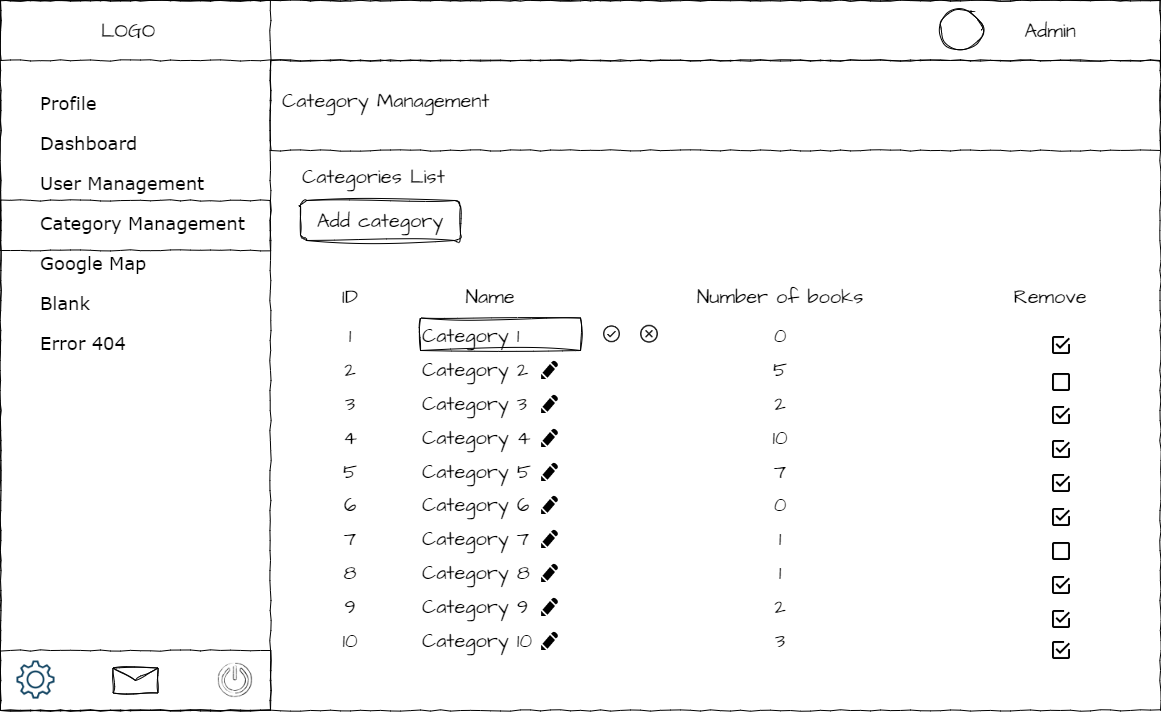


Figure : Update Category

* Function detail:
* When the admin clicks on the update button, this button is replaced by the “Save” and “Discard” buttons and the category name field becomes an input field.
* After entering a new category name, if the admin clicks “Save”, the category name will be updated. Otherwise, back to the normal category view without any change.
* Category’s name cannot be empty.

### d. Add

* Function trigger: This use case starts when the user clicks on the “Add category” button above the table of categories.
* Function description:
* Role: Admin
* Purpose: Add new category
* Interface: Popup
* 1 text field to enter new category name
* 2 buttons to choose add or cancel
* Function layout:

A picture containing diagram

Description automatically generated

Figure : Add Category pop-up

* Function detail:
* Admin can enter a new category name and click on the “confirm” button to add a new category.
* If the admin doesn't want to add a new category anymore, click on the “cancel” button.
* Show a message if this action is successful.

## 9. Add to/Delete from Favourite

* Function trigger: When user click the “Add to Favourite”/ “Delete Favourite” button in the Book detail page
* Function description:
* Role: Users
* Purpose: Send/Remove this book in user Favourite’s book list
* Interface: 4 buttons: Get (Purchase book), Delete Favourite (Remove this book from Favourite), Report (Report this book), Post (Send the comment to this book)
* 1 text input: where to let users enter the comment they want to
* 5 radio buttons represent for 5 stars rating
* Function Layout:

Diagram

Description automatically generatedDiagram

Description automatically generated

Figure : After Adding/Before Removing Figure : Before Adding/After Removing

* Function Detail:
  + Add to favourite: When user click “Add to Favourite”, the book automatically added in their Favourite, the button also changes to “Delete Favourite”. When they click again, the book will be removed from their Favourite, the button changed to “Add to Favourite”
* Delete from favourite: When user click “Delete Favourite”, the book automatically removed in their Favourite, the button also changes to “Add to Favourite”. When they click again, the book will be added their Favourite, the button changed to “Delete Favourite”

## 10. Comment

### 10.1 Add/Reply/Edit/Delete Comment

* Function trigger: When user write their comment in the text box and click Comment in the Book-detail page and choose option on interface
* Function description:
* Role: Users
* Purpose: let users write their feeling about the book. And then reply an existed comment, edit or their comments
* Interface:
  + Add Comment:
  + 1 text box for users to write their comments
  + 1 button to submit their comment
  + 1 textbox to display the previous comments
* Reply Comment:
* 1 button to reply comment
* Edit Comment:
  + 1 button to edit comment
* Delete Comment
  + 1 button to delete comment
* Function Layout:

Diagram

Description automatically generated

Figure : Add Comment

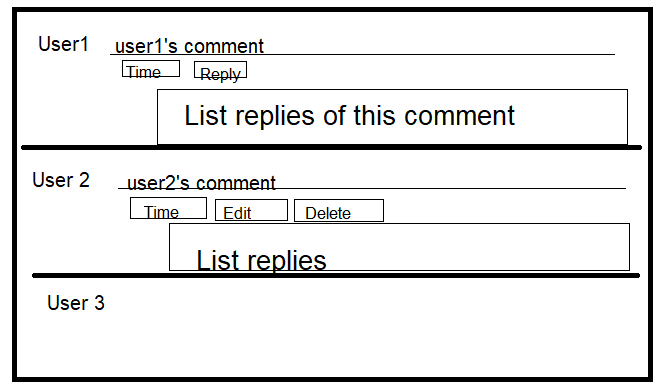
****

Figure : Edit/Reply/Delete Comment

* Function Detail:
* Text box “Write your comment here” can accept any kind of data (String)
* When the user clicks the button “POST”, their comment will be added to the textbox on the right, the page is refreshed at the same time.
* If customers are not login into the web, all the comments will have a Reply button. If they write and press Reply, they will be sent redirect to Login screen
* When click Reply: A text area will be shown to enter a comment. Next to it is Reply button that allows user post the reply
* Reply button only appears in other people's comments. Which mean: User 1 can’t see the Reply button below the comment that User 1 posted
* Edit and Delete button will be shown below the comments that users own
* If click Edit, a text area will be shown to let user edit their comment, next to it is button Edit
* If the user clicks Delete, a Warning box will be shown to ask the user whether to delete this comment or not.

### 10.2. Report Comment

* Function trigger: When clicks on report comment button
* Function description:
* Actor: User
* Purpose: Report comment that not suitable
* Interface:
* A logo can access Homepage
* A fixed Menu collapsed
* A book-detail brief
* A comment that has reported
* A list of violation type
* A contact interface
* Function layout:

Letter

Description automatically generated

Figure : Report Comment

* Function detail: When the user clicks the icon report on the book-details screen, it will direct to the report-comment screen. The user chooses violation type and input reason at the end of the list and clicks the “Report” button to finish.

### 10.3. Ban Comment

* Function trigger: When clicks on ban comment button
* Function description:
* Actor: Admin
* Purpose: Block comment that not suitable
* Interface:
* A logo can access Homepage
* A fixed Menu collapsed
* A book-detail display
* A comment area
* Contact interface
* Function layout:

Table

Description automatically generatedTable

Description automatically generated with medium confidence

Figure : After ban comment Figure : Before ban comment



Figure : Confirm pop-up

* Function detail: When admin click logo “x” on the screen, the confirmation pop-up appears. Click “Close” to skip and back, click “Yes, ban it” to confirm decision.

## 11. View/Close/Reject Reports

* Function trigger: When users click the “Reports” panel in the Management screen
* Function description:
* Actor: Admin
* Admin can view details and take action on all reports.
* Interface:
* View Reports:
  + Table includes ID, reporter, type of report, violations, sent date
  + Button: Open/Close/Reject, View Details, Select Type
* Reports Detail:
  + Information about object reported in the left (name, owner/creator, id, link to details page)
  + Report information: ID, Type, Reporter, Violations, Detail, sent date, status
  + Input: Action
  + Button: Close/Reject
* Screen Layout:

A screenshot of a computer

Description automatically generated

Figure : View Report

Graphical user interface, application, Teams

Description automatically generated

Figure : Report Detail

* Function detail:
* Click on Open/Close/Reject button to view reports of different state
* Click on another choice in select button to view reports of different type
* If you want to view details and take action with a report, click button at column at the end of that row
* If the reported object is a book, a link to its official page, a link to its analysis page, and a link to its author page are attached. If it is a novel, plus a link to the user's activities.
* If the reported object is a comment, a link to commenter’s activities and a link to its location are attached.
* A link to the reporter's activities is also attached.
* Click on the Close/Reject button on the report details page to mark a report as solved or rejected.

## 12. Transaction

### 12.1. Purchase a book

* Function trigger: This use case starts when user click on “Buy” in book detail screen
* Function description:
* Role: User
* Purpose: Remove a payment method
* Interface:
* Screen:
  + 1 button “Buy” to open the function
* Confirm pop-up
  + 1 input field to enter password
  + 2 buttons cancel and confirm
* Function layout:

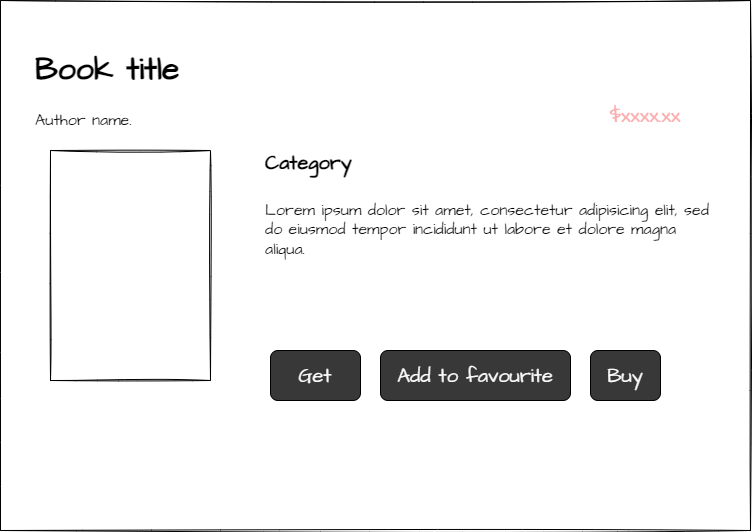


Figure : Get book

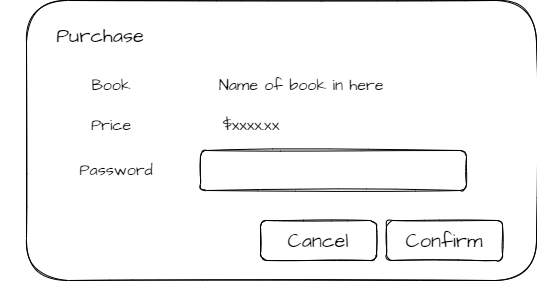


Figure : Purchase pop-up

* Function detail:
* After the user clicks on the “Buy” button, confirm the purchase pop-up will open.
* Entered password has to match with the current password of the user.
* If the user doesn't want to buy this book anymore, click cancel.
* If the user certainly wants to buy this book, click confirm.
* Before buying this book, the user should ensure that the book's price is not greater than the balance in his wallet.

### 12.2. Purchase novel’s chapter

* Function trigger: This use case starts when user click on locked chapter in book detail screen
* Function description:
* Role: User
* Purpose: Purchase a chapter of a novel
* Interface: Popup
* 1 input field to enter password
* 2 buttons cancel and confirm
* Function layout:

Text

Description automatically generated with low confidence

Figure : Purchase pop-up

* Function detail: Pop-up
* Entered password must match with the current password of the user.
* If the user doesn't want to buy this chapter more, click cancel.
* If the user certainly wants to buy this chapter, click confirm.
* Before buying this book, the user should ensure that the book's price is not greater than the balance in his wallet.

### 12.3. Payment

#### a. View

* Function trigger: This use case starts when the user clicks on “Payment” on the left side bar in the profile screen.
* Function description:
* Role: User
* Purpose: View balance in wallet and transaction history
* Interface:
* 2 buttons “Deposit” and “Withdraw”
* A table of all transactions
* Report button for each transaction
* Function layout:

Table

Description automatically generated

Figure : View payment

* Function detail:
* Wallet’s balance will be hidden at the start. The user can click on the eye icon to show or hide it.
* The user can see a list of all transactions he made.
* If a transaction has any problem, the user can report it by the report button at right of that transaction.

#### b. Deposit

* Function trigger: This use case starts when the user clicks on the “Deposit” button at right of balance.
* Function description:
* Role: User
* Purpose: Show deposit instructions and create a deposit request.
* Interface:
* 1 numberbox to enter amount
* 1 QR code of website’s bank account
* 1 reCAPTCHA
* 2 button “Cancel” and “Submit”
* Function layout:

Graphical user interface

Description automatically generated with medium confidence

Figure : Deposit Payment

* Function detail:
* The user is asked to enter the amount of money they want to deposit to his wallet on the website before making the transaction. This amount of money need to same with how much he sent to website’s receiving account
* The user must check the captcha before submitting the form.
* If the user doesn't want to deposit anymore, click the “Cancel” button to go back to the payment screen. Otherwise, click the “Submit” button.
* After submitting, the user can see the status of this transaction in the payment screen.

#### c. Withdraw

* Function trigger: This use case starts when the user clicks on the “Withdraw” button at right of balance.
* Function description:
* Role: User
* Purpose: Show withdrawal instructions and create a withdrawal request.
* Interface:
* 2 number boxes to enter amount and account number
* 1 select box to choose the bank
* 1 reCAPTCHA
* 2 buttons “Submit” and “Cancel
* 1 pop-up:
  + 1 textbox to enter user’s password
  + 2 buttons “Confirm” and “Cancel”
* Function layout:

Graphical user interface, application

Description automatically generated

Figure : Payment withdraw

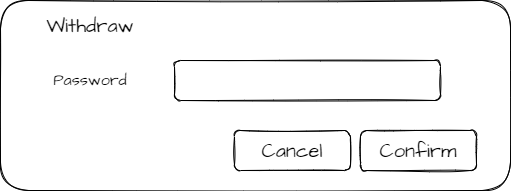
****

Figure : Payment withdraw confirm password dialog

* Function detail:
* The user is asked to enter the amount of money they want to withdraw and the payment account that they want to withdraw to it.
* The user must check the captcha before confirming.
* If the user doesn't want to deposit anymore, click the “Cancel” button to go back to the payment screen. Otherwise, click the “Submit” button.
* After clicking the “Submit” button, ask the user to enter his password to confirm the request.
* After submitting, the user can see the status of this transaction in the payment screen.

#### 46. Transaction manage

* Function trigger: This use case starts when the admin clicks on “Transaction management” at the left sidebar in the admin panel screen.
* Function description:
* Role: Admin
* Purpose: Manage transaction of user
* Interface:
* 2 list of pending transaction and all transactions
* 4 textboxes
  + 1 text box to enter user’s id for each list
  + 1 text box to enter product’s id for each list
* 1 select field to select the type of transaction for each list
* Function layout:

Graphical user interface, table

Description automatically generated

Figure : Transaction manage

* Function detail:
* Admin can see a list of pending transactions and a list of all transactions of all users.
* Admin can filter transactions by user id, type of transaction, status and product id.
* Admin can change the status of a pending transaction to failure or success. This action cannot be undone.
* If the admin rejects a transaction (change the status to fail), he has to enter the reason.

### 12.4. Report Transactions

* Function trigger: It will be revealed after the customer login into the system and click to Payment in the Profile
* Function description:
* Roles: User
* Purpose: Let users send reports to admin that tell what problems they are with
* Interface: (Like book’s detail interface)
* 1 button for send report next to each transaction
* A table display all information of transactions
* Function Layout:

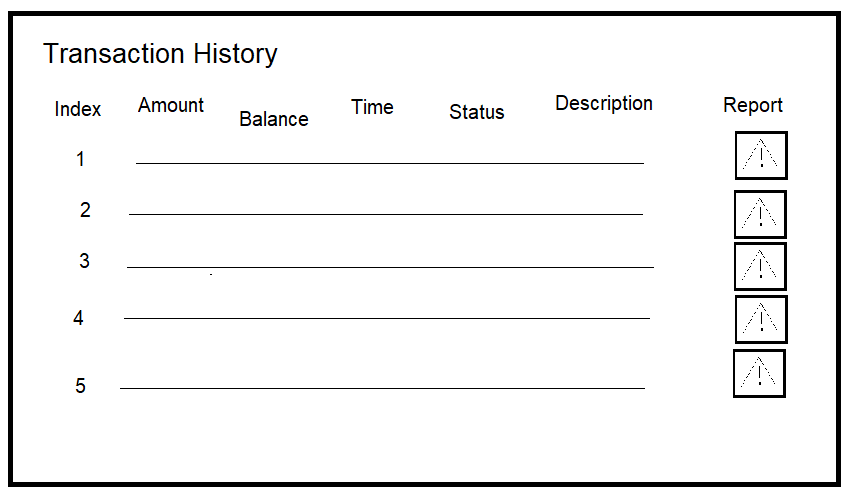
****

Figure : Report transaction

Table

Description automatically generated

Figure : Report detail

* Function Details:
* If click the “Report” button, a Pop up will be shown that let user choice what problems are and they can write the problem details
* Click “Report” will send report to database
* Click “Cancel will” close this pop up
* Users can choice more than 1 options

## 13. View author’s information

* Function trigger: When click on author’s name
* Function description:
* Actor: Users
* Purpose: See all book of a authors
* Interface:
* A logo to access homepage
* A fixed Menu collapsed
* A brief information of this author
* A header named “From this author”
* All books from the same author, 3 books per row
* Contact interface
* Screen Layout

**Diagram

Description automatically generated**

Figure : Author's information

* Function detail: When user click author’s name in book-details, it will direct to book-author screen to see author’s information with their novel

## 14. Bookshelf

* Function trigger: This use case starts when the user clicks on “Bookshelf” from the right menu.
* Function description:
* Role: User
* Purpose: Show all books of this user, include bought books, favourite books and creations
* Interface:
* Count number of bought books, favourite books and creations
* 3 tabs “My books”, “My favourites”, “My creations”
* All user’s bought books in the “My books” tab, paging by 12 books
* All favourite books in the “My favourites” tab, paging by 12 books
* All user’s novels in the “My creations” tab, paging by 12 books
* Function layout:

Diagram

Description automatically generated

Figure : Bookshelf

* Function detail:
* The user can see how many books he bought, added to favourite or he wrote.
* When clicking on the tab “My books”, the user sees all of the books that he bought.
* When clicking on “My favourites”, the user sees all of the books that he added to favourite.
* When click on “My creations”, user sees all of books that he wrote

## 15. Ticket

* Function trigger: When User click on Support on My Profile Title
* Function description:
* Roles: User
* Purpose: let user can view all the reports and send ticket to unlock account or monetization
* Interface
* 3 Title on top: read, unread, sending
* Ticket button for open the ticket form
* Screen Layout:

Text

Description automatically generated with medium confidence

Figure : See all ticket

Text

Description automatically generated with low confidenceShape

Description automatically generated with medium confidence

Figure : Ticket detail Figure : Report detail

* Function Detail:
* The screen will display some features: Read report, Unread report, Sending report. Each feature has its table
* Table will display the type of report, sent day, received day, status, note
* If user account is banned or total view of user’s books greater then 5000, user can write ticket   
  Input Require: 1 - 200 characters, space include
* Ticket is automatically match with user’s levels, ticket will be passed straightway if transaction for more than 3 times

## 16. Admin Panel

* Function trigger: When admin Dashboard
* Function description
* Roles: Admin
* Purpose: let admin can data research
* Interface:
* 3 charts: line chart, pie chart
* 1 table: special information
* Screen Layout:

Chart, box and whisker chart

Description automatically generated

Figure : Admin Panel

* Function Detail:
* The screen will display some features: total books, total comments, total
* Chart will display: the change of users approach per month, user approach per quarter in year, the amounts of authors and users-authors

## 17. Notification

* Function trigger: When Admin click on Mail icon on sidebar
* Function description
* Roles: Admin
* Purpose: let admin can view all the ticket and handle them
* Interface
* 1 dropdown menu for pass and reject button
* Pass and reject button for all ticket
* Screen Layout

Text

Description automatically generated

Figure : List tickets

Graphical user interface, application

Description automatically generated

Figure : Recent activities

* Function Detail:
* The screen will display List of with features: user id, sent day, note
* After click on user id, admin can see recent transaction and report
* If ticket is “Enable Monetization”, Admin can view all related book
* Dropdown in the end of row, admin pass and level up that user or reject the ticket
* Admin can pass or reject all