**The Lebanese University**

**Faculty of Economics and Business Administration**

**First Branch**

**Senior Project**

**Online library Website**

For obtaining a Bachelor's degree in Business Computer

**Prepared By**

Mira Farhat 20220608

Aya bitar 20220507

**Committee members:**

Supervisor: Dr. Ali Kalakesh

Co-Supervisor: Dr. Mustapha Sarji

**Academic Year 2023-2024**

**Dedication**

This project is dedicated to the brave martyrs and fighters who have sacrificed their lives for our safety and freedom. Your courage and selflessness inspire us to strive for a brighter future. We are eternally grateful for your sacrifices and remain committed to honoring your legacy through our work and aspirations.

We also dedicate this project to the countless hours of collaboration, learning, and perseverance that have brought us here. To our families and friends who supported us unwaveringly, offering encouragement and understanding during late nights and tough challenges, thank you. Special thanks to our mentors and doctors whose guidance shaped our journey and broadened our horizons. This project stands as a testament to our dedication, passion, and the power of teamwork.

**Acknowledgement**

First and foremost, we would like to extend our deepest gratitude to our university for providing us with the opportunity and resources to bring this project to fruition. The support and facilities made available to us were instrumental in completing this work.

We are especially thankful to our supervisor, Dr. Ali Kalakesh, for his unwavering support and guidance. His invaluable insights and constructive feedback have been pivotal in shaping the direction of our project. The knowledge and wisdom he imparted, not only during this project but throughout our academic journey, have been truly inspiring.

We would also like to express our sincere thanks to our co-supervisor, Dr. Mustapha Sarji, for his continuous encouragement and assistance.

**Abstract**

This project aims to develop a comprehensive online library website that enables users to browse, purchase, and rent books, while also providing access to a variety of additional services. The platform is designed to enhance the user experience with features such as graphic design requests, where users can collaborate with designers to create custom content. The website also introduces a rental and subscription model, allowing users to rent books for specified periods and low prices. Developed with modern web technologies, this website ensures a seamless interface, secure transactions. Overall, it offers a vast collection of books alongside creative services and responsive support, making it a versatile tool for readers and designers alike.

**Table of Contents**

[Introduction 6](#_Toc186103549)

[Benefits of online library websites: 6](#_Toc186103550)

[Problem Statement: 6](#_Toc186103551)

[Objectives: 7](#_Toc186103552)

[Methodology: 7](#_Toc186103553)

[Outline 8](#_Toc186103554)

[Chapter 1: Related Works 9](#_Toc186103555)

[Introduction: 9](#_Toc186103556)

[1. Existing Online Library Websites 9](#_Toc186103557)

[2. Comparative Study 11](#_Toc186103558)

[3. Proposed Solution 12](#_Toc186103559)

[Chapter 2: System Analysis and design 13](#_Toc186103560)

[Description of the System 13](#_Toc186103561)

[Requirements and Business Rules 13](#_Toc186103562)

[System Requirements: 14](#_Toc186103563)

[Scope of the System 14](#_Toc186103564)

[Data Model description 15](#_Toc186103565)

[Data Dictionary 15](#_Toc186103566)

[Chapter 3: Implementation 21](#_Toc186103567)

[3.1 Programming Languages & Frameworks 21](#_Toc186103568)

[3.2 Development Tools 24](#_Toc186103569)

[3.3 Interfaces 25](#_Toc186103570)

[CHAPTER 4: Future Plan and Conclusion 36](#_Toc186103571)

[System Pages and Features 36](#_Toc186103572)

[Role-Specific Dashboards 36](#_Toc186103573)

[Future Enhancements 37](#_Toc186103574)

[Conclusion 37](#_Toc186103575)

[References 38](#_Toc186103576)

**Table of content**

[Figure 1.Kindle 10](#_Toc187016807)

[Figure 2.Barnes & Noble 11](#_Toc187016808)

[Figure 3.Blurb 11](#_Toc187016809)

[Figure 4.CDM 19](#_Toc187016810)

[Figure 5.PDM 20](#_Toc187016811)

[Figure 6.Home Page 26](#_Toc187016812)

[Figure 7.About us 27](#_Toc187016813)

[Figure 8.Contact us 27](#_Toc187016814)

[Figure 9.Sign in page 28](#_Toc187016815)

[Figure 10.sign up page 29](#_Toc187016816)

[Figure 11.Checkout page 30](#_Toc187016817)

[Figure 12.Cart 31](#_Toc187016818)

[Figure 13.Library 32](#_Toc187016819)

[Figure 14.Services 33](#_Toc187016820)

[Figure 15.user profile 34](#_Toc187016821)

[Figure 16.admin dashboard 35](#_Toc187016822)

# Introduction

This online library website is designed to offer users a unified platform for purchasing and renting books, as well as accessing additional services such as graphic design requests. By integrating these functionalities, the website provides a seamless experience where users can browse, buy, and utilize creative services efficiently. The platform features intuitive navigation, secure payment processing, and user-friendly dashboards for both customers and administrators. Additionally, the inclusion of a book rental service allows users to borrow books for a limited time, enhancing accessibility and flexibility.

## Benefits of online library websites:

 **Easy Book Access**: Effortlessly browse, search, and purchase books across various categories.

** Flexible Book Rentals:** Rent books for a limited period, with the option to extend or return.

** Custom Design Services:** Request tailored graphic design work, including uploading design requirements and managing feedback.

** Secure and Efficient Transactions:** Ensure safe, seamless payment processing for purchases and subscriptions.

** Comprehensive Dashboards:** Simplify user and admin operations with personalized dashboards, offering access to order management, book listings, rentals, and design requests.

## **Problem Statement**:

Most platforms cater to either book sales or freelance services but rarely combine these functionalities into one system. Users often face inefficiencies when switching between platforms for book purchases, rentals, and design requests, leading to fragmented order management and communication. This project aims to unify these services into a cohesive, user-friendly platform, addressing gaps in existing systems and simplifying the process for both users and administrators.

## Objectives:

The online library website aims to provide a comprehensive solution that enhances user experience through the following objectives:

* **Simplify Book Access**: Enable effortless browsing, searching, and purchasing of books across categories.
* **Offer Book Rentals and Subscriptions**: Provide flexible options for renting or subscribing to books on a monthly basis.
* **Integrate Design Services**: Allow users to request and manage custom graphic designs, including uploading specifications and reviewing drafts.
* **Ensure Secure Transactions**: Implement robust payment gateways to facilitate safe purchases and rentals.
* **Streamline Communication**: Enable direct interactions between users, designers, and support teams through integrated chat and support features.

## Methodology:

The website requires a sign-up page for user registration and a sign-in page to access personalized dashboards. Within the dashboard, users can browse books, receive recommendations, purchase or rent books, request design services, and interact with the bot for assistance. The platform also supports secure payment processing and communication between customers and designers.

The website is designed using **HTML5** and **CSS3** to create a visually appealing and responsive interface. **JavaScript** is used to add dynamic and interactive elements. **PHP** serves as the server-side scripting language, processing requests and generating HTML that is delivered back to the user. The integration of these technologies ensures a robust and user-friendly website that meets the diverse needs of book buyers and creative service seekers.

## Outline

 **Chapter 1: Related Works**

* Discusses existing systems similar to our online library website, including their major functionalities such as book sales, design services, and customer support. It also identifies gaps in these systems and how our project addresses these shortcomings.

 **Chapter 2: System Requirements and Design**

* Covers the system requirements, tasks, and permissions for both users and admins. Includes analysis and design for the system, along with conceptual and physical data models and their corresponding data dictionary.

 **Chapter 3: Implementation**

* Focuses on the implementation of the online library website, including the programming languages, development tools, user, and admin interfaces. Each section includes images and explanations of the various components.

 **Chapter 4: Conclusion and Future Plans**

* Discusses the conclusion, limitations, and future plans for the system, highlighting potential enhancements and features that could be added.

# Chapter 1: Related Works

## Introduction:

In this chapter, we provide an overview of existing related works by reviewing several prominent online library websites and analyzing their system designs and features. The goal is to identify key functionalities and incorporate them into our online library website. By outlining, comparing, and analyzing the advantages, limitations, and missing features of these platforms, we aim to build a more robust and user-centric solution.

## 1. Existing Online Library Websites

Online library websites are not new, but they continuously evolve to meet the needs of book enthusiasts and creative service seekers. For this study, we analyzed three platforms:

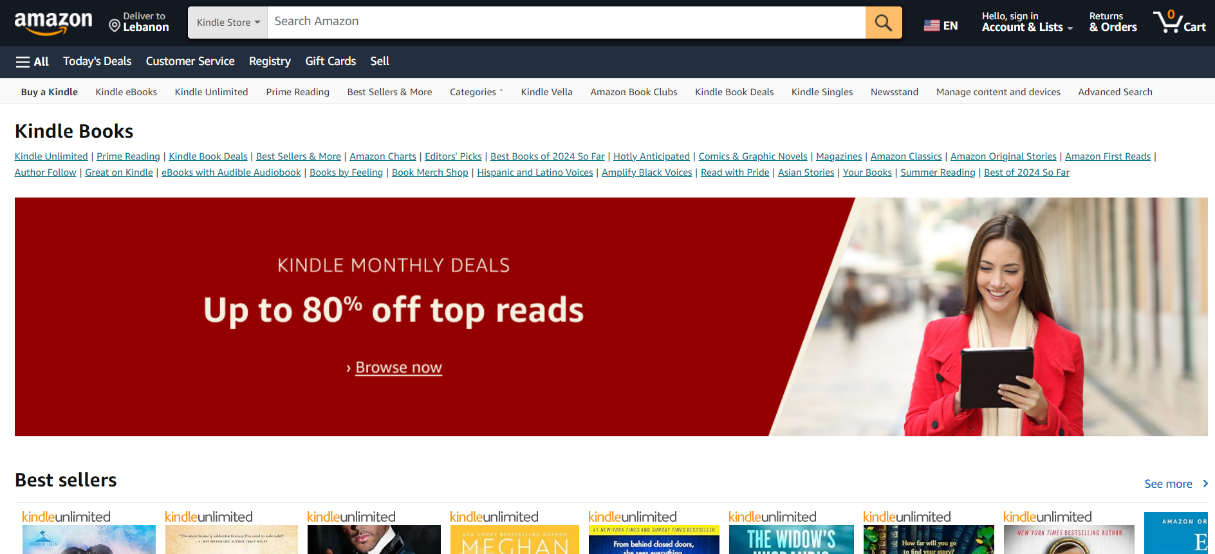
* 1. **Amazon Kindle Store**

Figure 1.Kindle

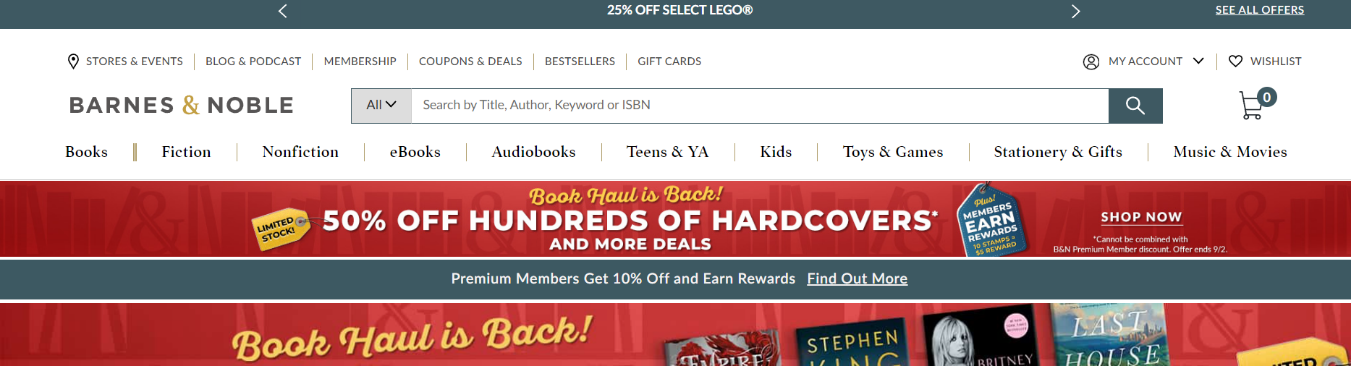
* **Platform**: Amazon
* **Description**: The Amazon Kindle Store allows users to purchase and download eBooks to Kindle devices or apps. It offers a vast selection of titles, device synchronization, and personalized recommendations. Seamless integration with the Amazon ecosystem—including secure payment processing and Amazon Prime benefits—makes it a market leader.
  1. **Barnes & Noble Online**

Figure .Barnes & Noble

* **Platform**: Barnes & Noble
* **Description**: Barnes & Noble's online platform provides a diverse collection of books, eBooks, and related products. It emphasizes user-friendly navigation, with genres, bestsellers, and personalized recommendations easily accessible. A membership program offers discounts and perks for frequent buyers, enhancing customer loyalty.
  1. **Blurb**

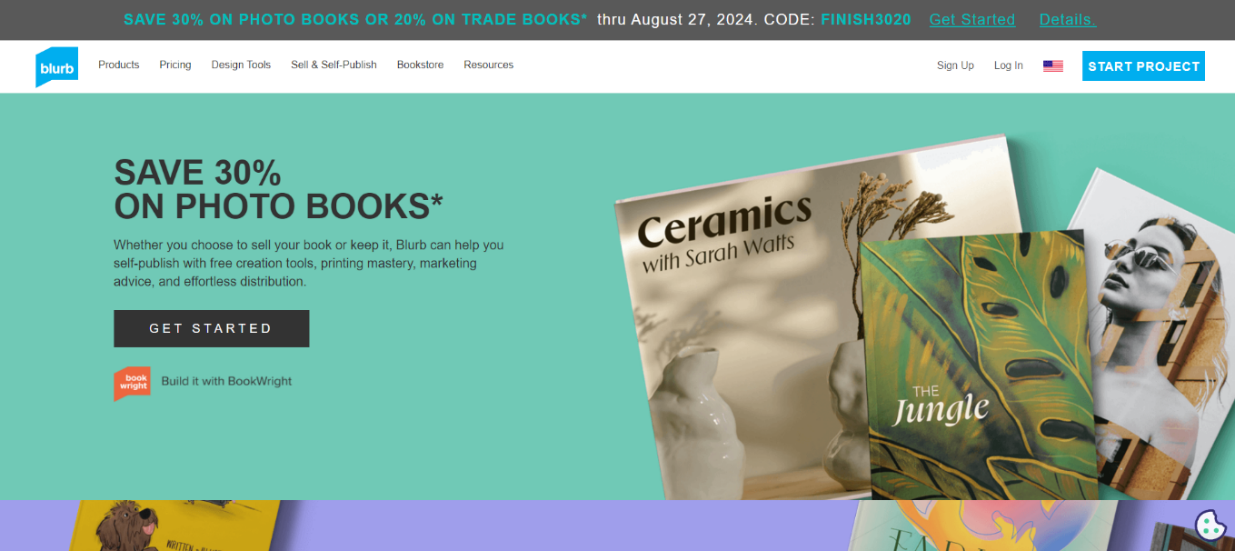


Figure .Blurb

* **Platform**: Blurb, Inc.
* **Description**: Blurb enables users to create, publish, and sell their own books. Its tools for designing both print and digital books set it apart from traditional book-selling platforms. Integration with creative services, like custom book design, appeals to authors and creatives seeking a tailored approach.

## 2. Comparative Study

Each platform offers unique strengths:

* **Amazon Kindle Store**: Known for its vast library, synchronization across devices, and integration with the Amazon ecosystem.
* **Barnes & Noble Online**: Offers a user-friendly interface and a membership program with customer benefits.
* **Blurb**: Excels in allowing users to create and sell their own books with professional design tools.

**Gaps Identified**:

While these platforms cater to specific needs, none fully integrate book-related services with graphic design options or a robust rental system.

## 3. Proposed Solution

Our online library website combines the best features of existing platforms while addressing their gaps:

**Features Adopted**:

* Comprehensive book browsing, purchasing, and renting options.
* Integration of creative services, such as custom graphic design requests.

**Advantages Over Existing Platforms**:

* Tailored for a broader target audience with flexible rental and subscription options.
* Unified user experience, combining books and design services in one platform.
* Scalable architecture for future enhancements based on user feedback.

# Chapter 2: System Analysis and design

**System Description**

The Online Library Website is a multi-functional platform offering book sales, rentals, and design services. It ensures a seamless user experience through efficient system management and secure transactions.

**Core Features**:

* **Book Management**: Admins manage inventory, including adding, editing, and removing books.
* **Rental and Subscription Services**: Users can rent books temporarily or subscribe to monthly plans.
* **Design Service Requests**: Users can submit custom graphic design requests via an intuitive interface.
* **Customer Support**: Dedicated staff address user inquiries and fulfill service requests.

## Requirements and Business Rules:

**Business Rules**

* **Book Inventory Management**: Admins handle all inventory tasks, ensuring book availability.
* **Customer Assistance**: Staff manage inquiries, design requests, and support tickets.
* **Rental and Subscription Management**: Admins define and enforce rental terms and subscription policies.
* **User Role Management**: Admins assign and oversee roles for users and staff.
* **Secure Transactions**: Payment gateways ensure secure processing of purchases, rentals, and design services.

**System Requirements**

* **Server Requirements**:
  + Apache Server with PHP extensions enabled.
  + MySQL database for managing data.
* **Client Requirements**:
  + A modern web browser with internet connectivity.

**Scope of the System**

* **Admin**:
  + Oversee user roles, book inventory, rental services, and design requests.
  + Ensure smooth platform operations.
* **User**:
  + Browse, purchase, and rent books.
  + Submit design service requests and interact with staff for support.

**Roles**

**Admin**

* Full control of the platform.
* Manages user roles, transactions, and book inventory.

**User**

* Registered users with access to book browsing, purchases, and rentals.
* Can request design services and seek customer support.

## Data Model description

## 

### Data Dictionary

A Data Dictionary is a list of data entity names, descriptions, and attributes used or collected in a database, management system, or section of a research study.

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Variable Name** | **Type** | **Constraints** |
| **Books** | book\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| bookTitle | VARCHAR(200) | UNIQUE, NOT NULL |
| categoryId | INT | FOREIGN KEY REFERENCES Category(category\_id), NOT NULL |
| authorId | INT | FOREIGN KEY REFERENCES Author(author\_id), NOT NULL |
| publisherId | INT | FOREIGN KEY REFERENCES Publishers(publisher\_id), NOT NULL |
| bookPrice | FLOAT | NOT NULL |
| stockQuantity | INT | DEFAULT 0, NOT NULL |
| genre | VARCHAR(50) | NOT NULL |
| description | VARCHAR(1000) | NULL (Optional) |
| language | VARCHAR(50) | NOT NULL |
| image | VARCHAR(255) | NOT NULL |
| rentalPrice | FLOAT | NOT NULL |
| **Publishers** | publisher\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| publisherName | VARCHAR(50) | UNIQUE, NOT NULL |
| email | VARCHAR(100) | UNIQUE, NOT NULL |
| address | VARCHAR(255) | NOT NULL |
| **Orders** | order\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| totalPrice | FLOAT | NOT NULL |
| userId | INT | FOREIGN KEY REFERENCES Users(user\_id), NOT NULL |
| phoneNumber | VARCHAR(15) | NULL (Optional) |
| street | VARCHAR(255) | NULL (Optional) |
| city | VARCHAR(100) | NULL (Optional) |
| address | VARCHAR(255) | NULL (Optional) |
| **Author** | author\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| authorFname | VARCHAR(100) | NOT NULL |
| authorLname | VARCHAR(100) | NOT NULL |
| Nationality | VARCHAR(200) | NOT NULL |
| **User** | user\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| password | VARCHAR(255) | NOT NULL |
| name | VARCHAR(100) | DEFAULT 'N/A', NOT NULL |
| email | VARCHAR(100) | UNIQUE, NOT NULL |
| roleId | INT | FOREIGN KEY REFERENCES Roles(role\_id), NOT NULL |
| **Roles** | role\_id | INT | PRIMARY KEY |
| role | VARCHAR(50) | NOT NULL |
| **Category** | category\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| categoryName | VARCHAR(100) | NOT NULL |
| **Cart** | cart\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| userId | INT | FOREIGN KEY REFERENCES Users(user\_id), NOT NULL |
| bookId | INT | FOREIGN KEY REFERENCES Books(book\_id), NOT NULL |
| quantity | INT | NOT NULL |
| created\_at | TIME | NOT NULL |
| isRental | TINYINT(1) | DEFAULT 0, NOT NULL |
| price | DECIMAL(10, 2) | NULL (Optional) |
| **BookReview** | review\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| userId | INT | FOREIGN KEY REFERENCES Users(user\_id), NOT NULL |
| bookId | INT | FOREIGN KEY REFERENCES Books(book\_id), NOT NULL |
| rating | INT | CHECK (rating BETWEEN 1 AND 10), NOT NULL |
| comment | VARCHAR(100) | NULL (Optional) |
| reviewDate | DATE | NOT NULL |
| **Rented\_books** | rental\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| userId | INT | FOREIGN KEY REFERENCES Users(user\_id), NOT NULL |
| bookId | INT | FOREIGN KEY REFERENCES Books(book\_id), NOT NULL |
| rent\_until | DATE | NOT NULL |
| rented\_at | DATE | NOT NULL |
| orderId | INT | FOREIGN KEY REFERENCES Orders(order\_id), NOT NULL |
| **Contact\_form\_Submission** | id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| first\_name | VARCHAR(100) | NOT NULL |
| last\_name | VARCHAR(100) | NOT NULL |
| email | VARCHAR(255) | NOT NULL |
| phone | VARCHAR(15) | NULL (Optional) |
| message | TEXT | NOT NULL |
| submitted\_at | DATETIME | NOT NULL |
| userId | INT | FOREIGN KEY REFERENCES Users(user\_id), NOT NULL |
| **Design\_Requests** | request\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| user\_id | INT | FOREIGN KEY REFERENCES Users(user\_id), NOT NULL |
| request\_type | VARCHAR(255) | NOT NULL |
| created\_at | DATETIME | DEFAULT CURRENT\_TIMESTAMP |
| status | VARCHAR(50) | DEFAULT 'Pending' |
| description | VARCHAR(1000) | NOT NULL |
| uploaded\_file | VARCHAR(255) | NOT NULL |
| email | VARCHAR(255) | NOT NULL |
| phone | VARCHAR(15) | NOT NULL |
| **Order\_Books** | order\_book\_id | INT | AUTO\_INCREMENT, PRIMARY KEY |
| order\_id | INT | FOREIGN KEY REFERENCES Orders(order\_id), NOT NULL |
| book\_id | INT | FOREIGN KEY REFERENCES Books(book\_id), NOT NULL |
| quantity | INT | NOT NULL |

#### Conceptual Data Model

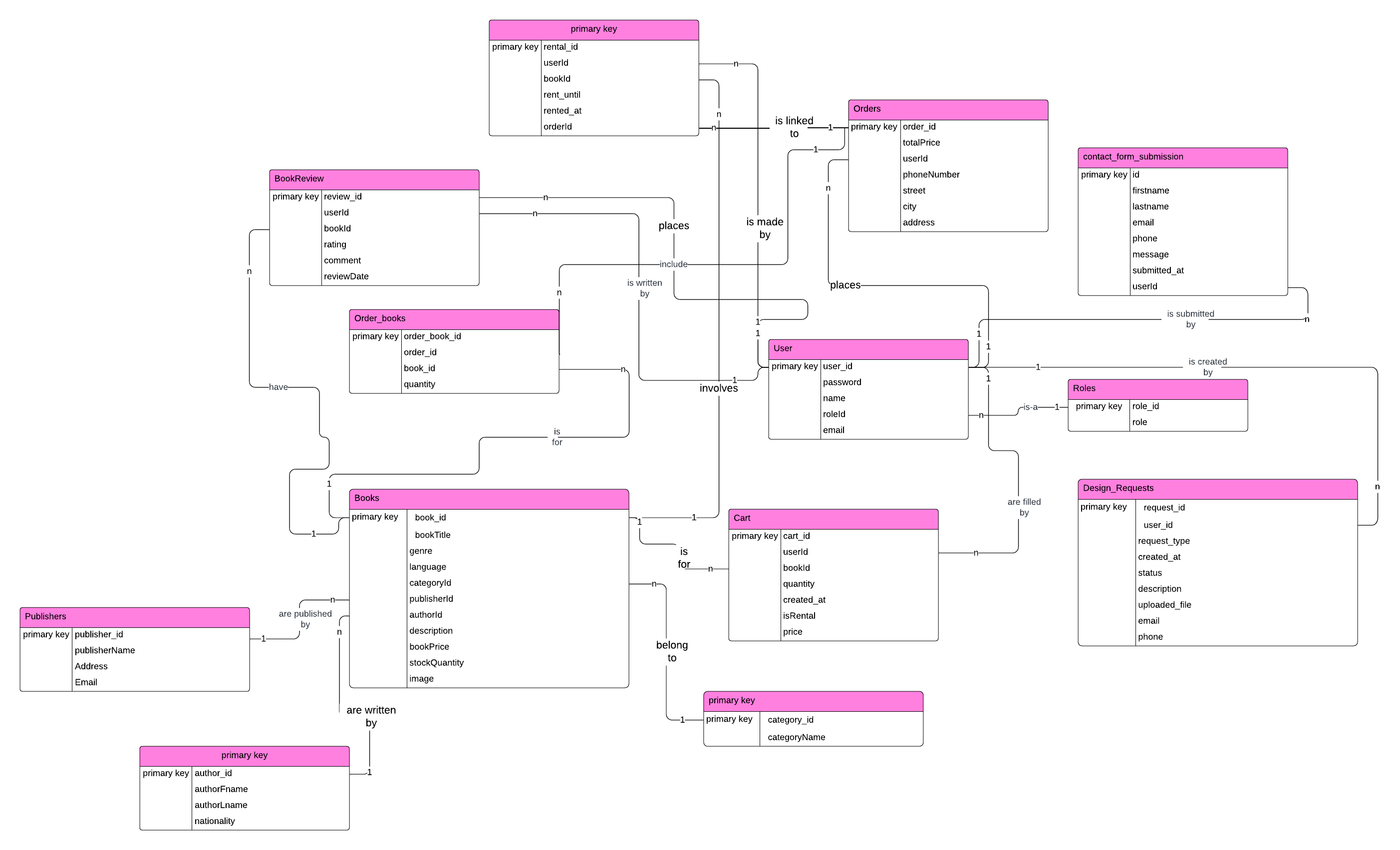
What the system comprises is specified by the logical data model. This Data Model involves the entities that define the information and the connections among these entities. The goal of building a conceptual data model is to configure entities, their attributes, and their relationship.

Figure .CDM

#### Physical data model

the physical data model is taking the logical data model and making it a specific database. it will have all the entity names converted to table names and all the attributes converted to columns and columns will have their data types also defined we can also specify all the indexes and partitions of the tables in the physical data model.

Physical data models characteristics include:

* All tables and columns are listed.
* For the definition of relationships between columns, foreign keys are used.
* For various RDBMS, the physical data model will differ. So it might look a little different in your oracle server versus your MySQL server each DB company creates their own slight detailed differences.

The physical data model is built directly from the previous conceptual data model except it has all the details in exactly how you're going to spell everything and what the data type will be.

The physical data model is based on the modification of entities to tables, the conversion of relationships into foreign keys, the conversion of attributes into columns, and the adjustment of the PDM is depending on limits and requirements.



Figure .PDM

# Chapter 3: Implementation

### 

### 3.1 Programming Languages & Frameworks

#### 3.1.1 PHP

PHP originally standard for a personal home page, nowadays it stands for hypertext pre-processor.

Basically, PHP language helps you to dynamically write webpages. It is a server-side scripting language, which ensures you can write it essentially in any text editor and run all the code on the server itself, not on the client computer. All the code then runs on the server and only the code output is then submitted to the web browser.

In particular, PHP is used to build Static Websites or Dynamic Websites, or Mobile Apps, and in order to run its code, you need to install a local server like XAMPP.

The reasons that you should be practicing PHP:

* It is used a tremendous amount in web applications that wide used such as WordPress.
* It can create useful apps and websites like Zen Cart (e-commerce Applications), WordPress (Web Content Management Systems).
* It is really easy to write code in PHP.
* In addition, PHP has a great range of libraries and plugins available that expand its abilities even further like the laravel framework.

#### 3.1.2 JavaScript

JavaScript is a scripting language that is run mainly from the browser, unlike PHP which typically a server-side language.it is responsible for the interaction with the webpage so that you can create an active user interface. JavaScript code can be nested into HTML pages and explain through the Web browser. JavaScript is an interpreted language. Thus, it doesn't need to be compiled. This allowing to enhancing the performance of the web page by exhibit special effects, accept variable text, validate data, create cookies, catch a user’s browser, et.

#### 3.1.3 HTML5

HTML5 it's the next increment in the version of HTML (hypertext mark-up language) the builder of the websites.

There are plenty of new things that Html5 supports.it used the <video> and <audio> tags to add video or audio and is supported in all major browsers, draws graphics with the new elements like the canvas selectors, inline SVG, and the use of 2D and 3D transforms. Besides that, HTML5 released new semantic elements such as <header>, <footer>, <nav>, <article>.

Moreover, HTML5 added some of the features to make the whole development (web and mobile development) process easier like local data storage which allows web pages’ store information locally within the client browser, applications cache to be accessible without on internet, web workers that allow JavaScript to run the background without affecting the performance of the whole page.

#### 3.1.4 CSS3

CSS is a shortcut to Cascading Style Sheets is used to design the webpage and make it more colourful. And CSS3 is the new standard for CSS.

CSS3 includes the old CSS specification. in addition to new modules like Selectors,2D/3D Transformations, animation, Box Model.

#### 3.1.5 JQuery

JQuery is the simplest and most popular JavaScript library.it enables people to write less and do more means that you can do things in JavaScript that need a lot of code line On the other hand you can do the same stuff in a single line or a few lines in JQuery. IT is also great for making it compatible with multiple browsers, multiple devices, tablets, phones, and any other kind of desktop resolution. To set things up you will download the JQuery library into the system and includes it in the project folder.

#### 

#### 3.1.6 MYSQL

MySQL is a relational database management system. It allows us to store data in tables by using SQL (structured query language), and has adopted its own dialect for example their extra features in MySQL that are not specified inside of this standard and those extra features are not the same as SQL server or oracle. Also it's used for smaller projects such as websites.

MySQL is open-source, which means you can go and view the code that actually makes the database. Also, it has one of the most flexible licensing systems.

Even more for MySQL, the learning curve is small compared to other database management systems.

#### 3.1.7 Bootstrap

Bootstrap is a free front-end web development framework that is simple and easy to use.

Bootstrap provides interface models for typography, shapes, icons, tables, navigations, etc. based on HTML and CSS. It also has some modules for JavaScript. In addition, it provides numerous benefits, such as providing many responsive features and compatibility with all the new browsers.

#### 

### 3.2 Development Tools

#### 3.2.1 XAMPP

XAMPP is a single installer that allows you to install the packages that most web developers need in order to deploy websites. It can install on Linux,

Windows, and Mac platforms. And it provides in one package apache,

MySQL DB, PHP, and Perl.

#### 3.2.2 Visual Studio Code

Visual studio code is a code editor, it is positioned kind of in-between text editors like atom and real IDE like visual studio this means it has all these text editor functions that you would expect when writing code, but it also comes with some selected IDE features for debugging. It's available for all operating systems and can be used with many programming languages like java, C++.

#### 3.2.3 GIT Bash

Git Bash is a simple windows program installer that is very straightforward, it's a bash shell environment there's no fee to use git or git bush in your projects without commercial licenses involved. used for version control and connecting the code base with the cloud repository.

#### 3.2.4 Lucidchart

Lucidchart is a cloud-based diagramming tool used to create conceptual and physical data models (CDMs and PDMs). It provides an intuitive interface for designing database schemas, process workflows, and system architecture. Lucidchart’s collaboration features make it ideal for team projects, ensuring efficient planning and documentation of database structures.

### 3.3 Interfaces

The project is a system constituted mainly of nine pages: Home, library, Cart, Checkout, Services, About Us, Contact Us, Sign in, and Sign up. In addition to the dashboard (team interface)

#### 3.3.1 Home page

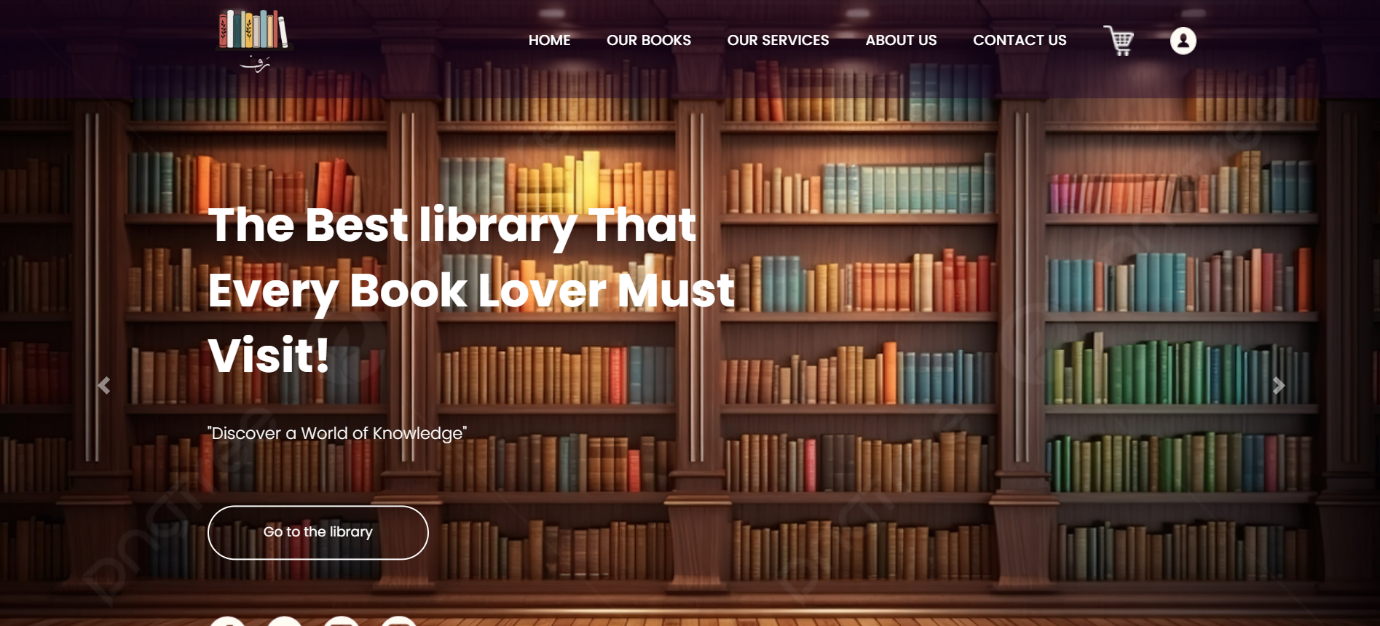


Figure .Home Page

This is the home page of our site. Once viewed, user can understand the topic and what the system is all about. It attracts the user’s attention by its colourful slide images, legible font, animation, and structuring of the page. It’s also user friendly. It is easy to navigate through out the page and the whole system.

#### 3.3.2 About Us page:

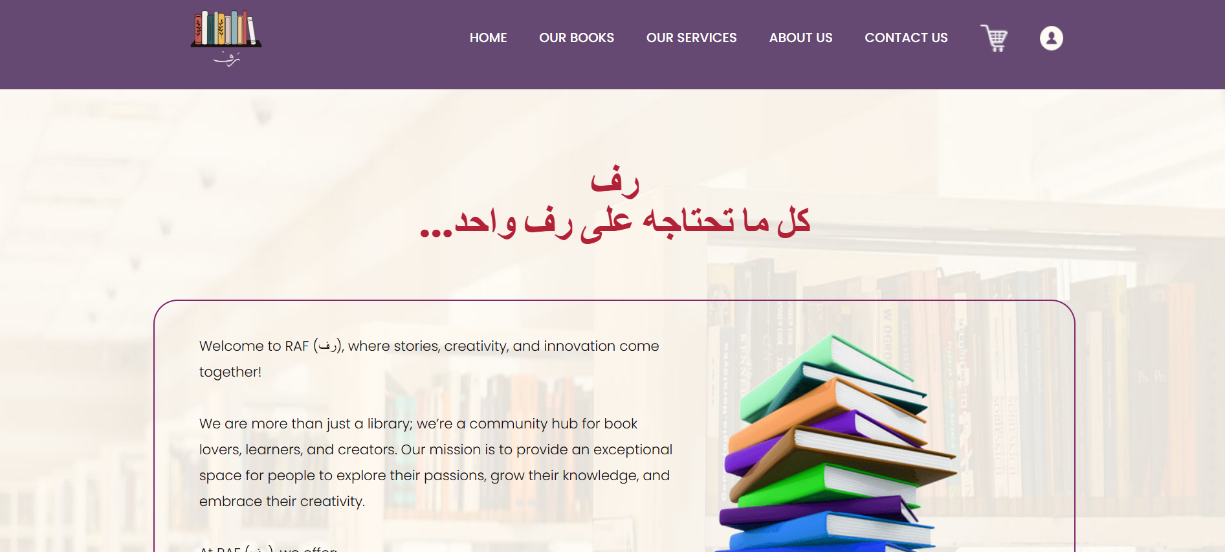


Figure .About us

About Us page is a brief description about the library and what we offer. It also includes our slogan.

#### 3.3.3 Contact Us page:

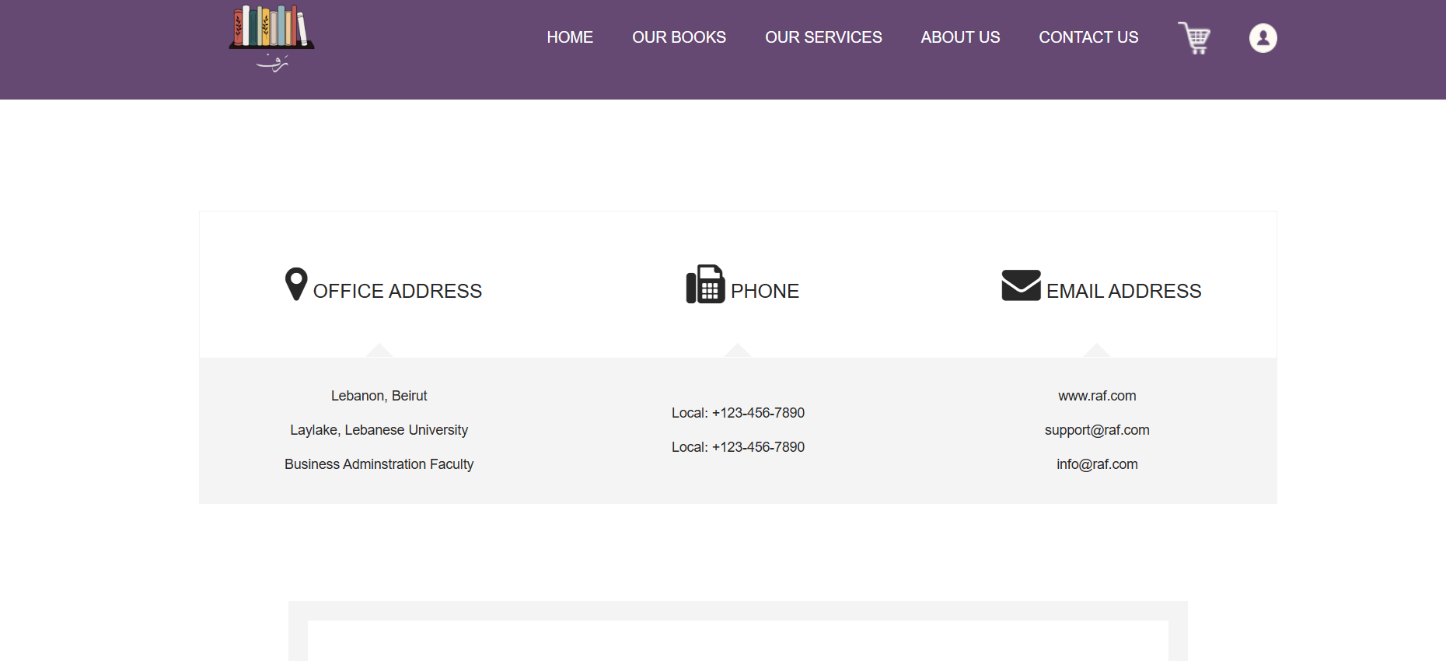


Figure .Contact us

The **Contact Us** page includes a form where users can submit their details such as first name, last name, email, phone number, and message. Upon clicking the "Send Message" button, the form triggers a **JavaScript validation method** to ensure all required fields are properly filled out.If validation is successful, the form is submitted to a PHP script that processes the data. The PHP script, utilizing the contact() function, stores the user's submitted information (e.g., name, email, and message) in a database for record-keeping and further action.If validation fails, the form displays specific error messages to inform the user of the missing or incorrect inputs (e.g., "Please enter a value for the message"). This interactive feedback helps users correct their mistakes before resubmitting the form.Additionally, the form provides visual cues for successful submission or errors during the submission process, ensuring a seamless user experience.

#### 3.3.4 Sign in page:

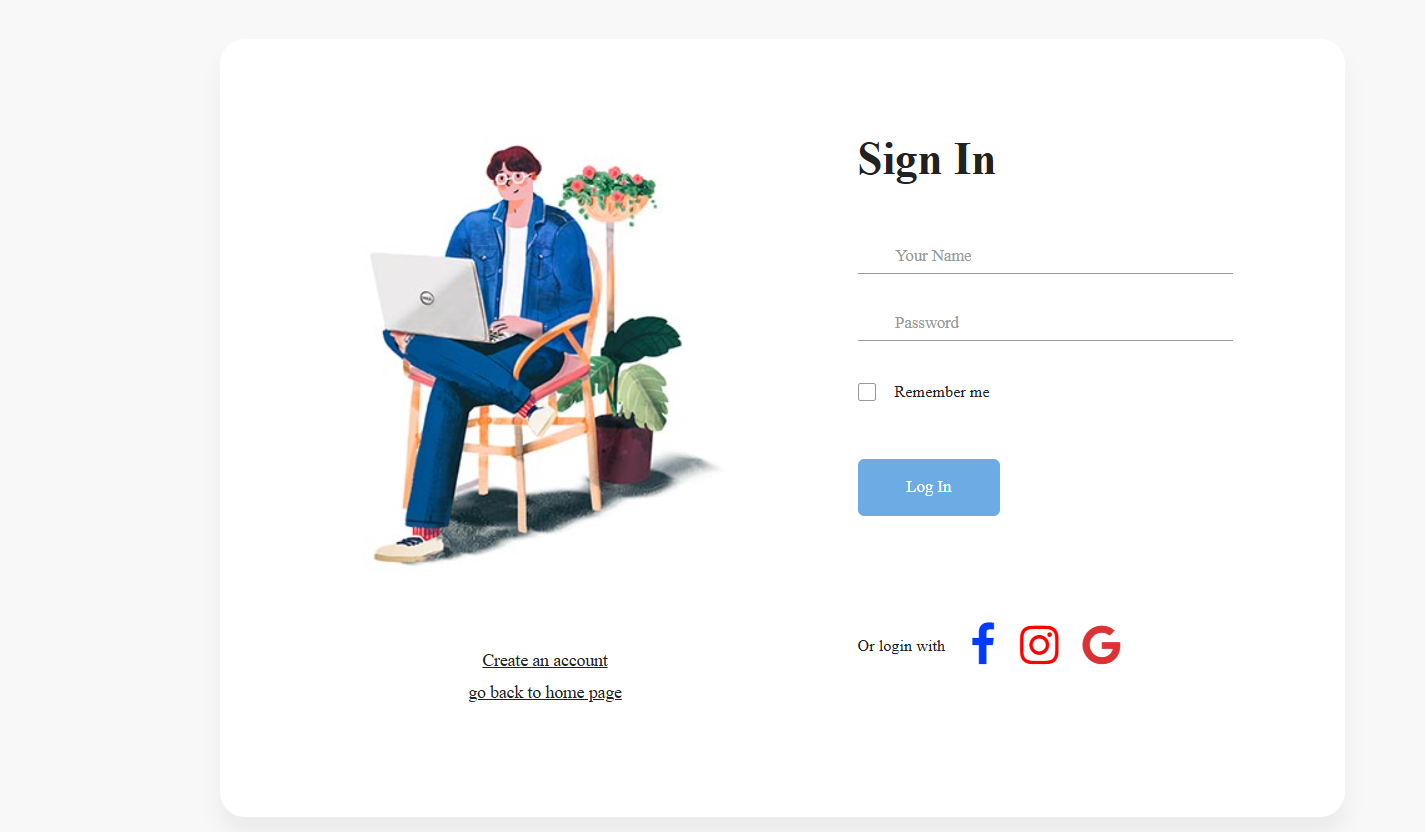


Figure .Sign in page

Sign in page is a simple page. User enters the name and password created in the registration page to login. When click on login button will invoke validation method (JavaScript) to validate with the non-empty check name format and password. Then request to PHP backend, the login () function checks if there any match for the entered login details. If the match found, it clears the authentication and allows the user to access the dashboard. Otherwise the code will display” user not registered”.

#### 3.3.5 Registration page:

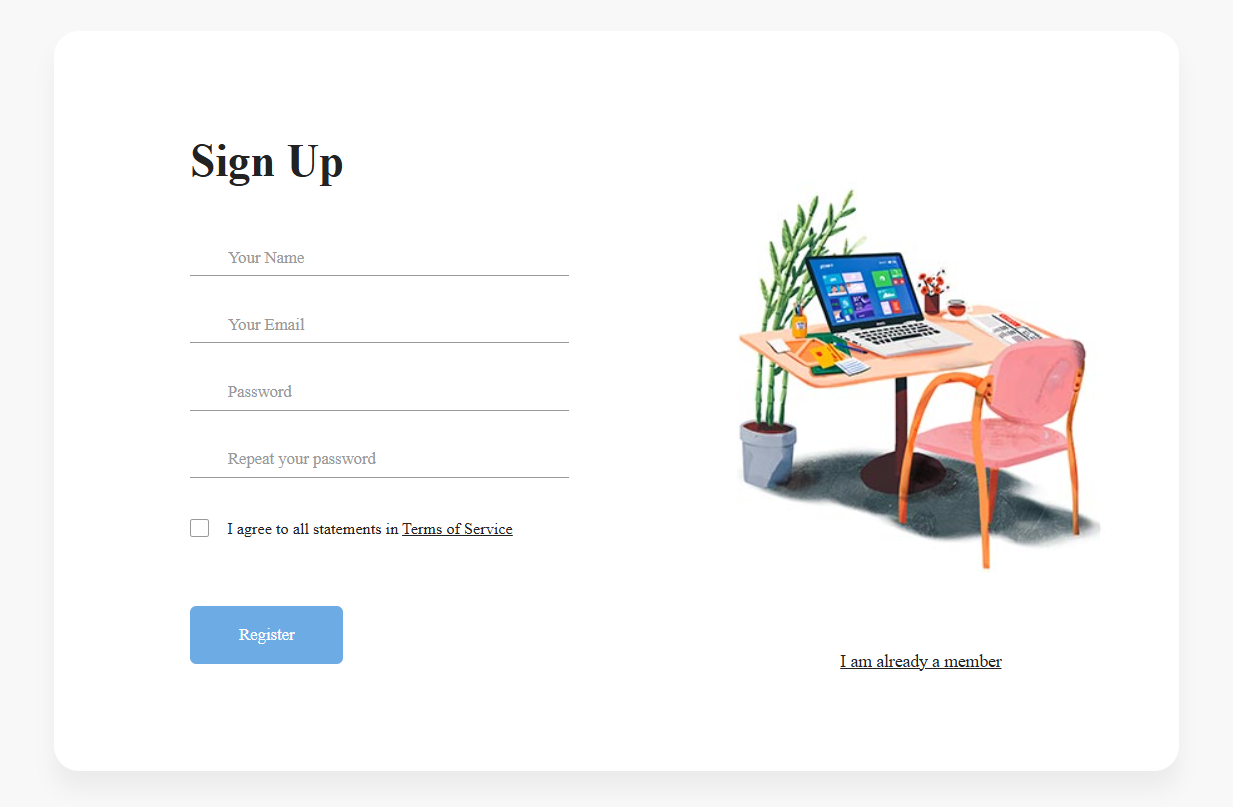


Figure .sign up page

In the Registration page, user can create his/her account by filling necessary information in the form. When submitting the registration form “create account” it will invoke the validation method (JavaScript) to insure that there is no data missing, and that data entered is correct and in the right format. After the validation, request PHP backend Using XHRform() request to send an Ajax request .The register () function, it checks if the posted email already exists. If so, it truncates the registration flow and returns the error. Otherwise, it creates the insert query to add the user record into MySQL database.

#### 3.3.6 Checkout:

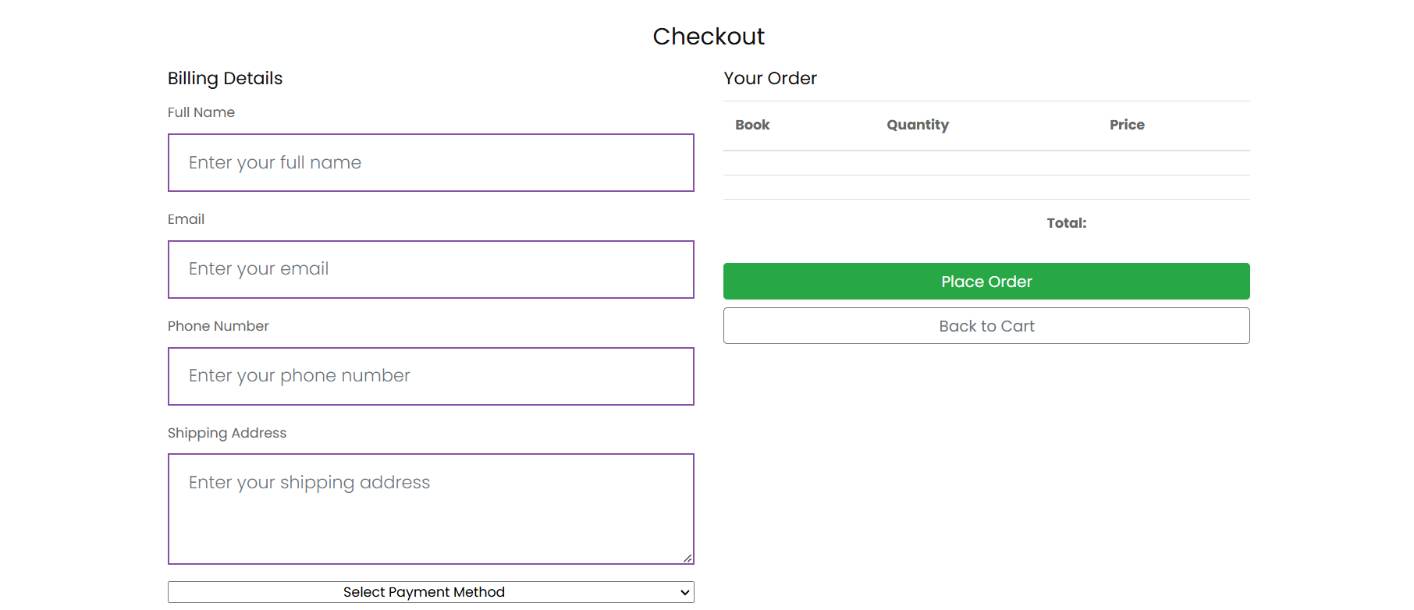


Figure .Checkout page

The Checkout page allows users to finalize their purchase or rental order. It includes a form to collect billing and shipping details. JavaScript validates the input fields to ensure all required data is filled out correctly before submission. Upon submitting the form, an Ajax request sends the data to the PHP backend, where the processOrder() function creates a new order record in the MySQL database. A confirmation message is displayed to the user if the order is successful, with error handling for any failures.

#### 3.**3.7 Cart:**

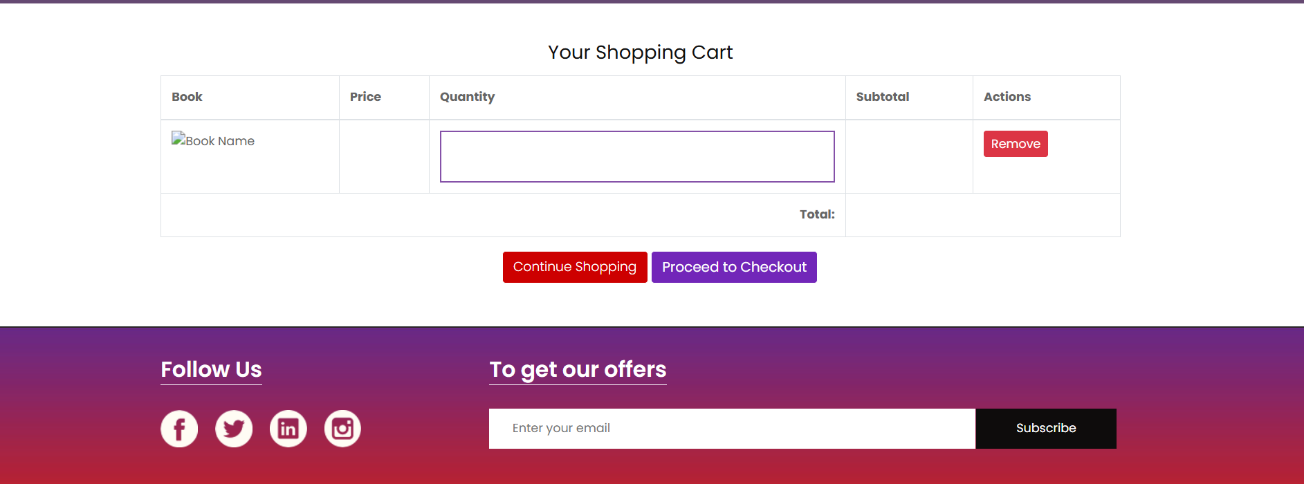


Figure .Cart

The Cart page displays the list of items that the user has added for purchase or rental. Each item includes its details, such as title, quantity, price, and total cost. Users can adjust item quantities or remove items from the cart. When changes are made, JavaScript dynamically updates the total cost without reloading the page. A "Proceed to Checkout" button redirects users to the Checkout page. Server-side PHP ensures the cart's contents are stored and updated securely in the backend database.

#### 3.3.8 Library

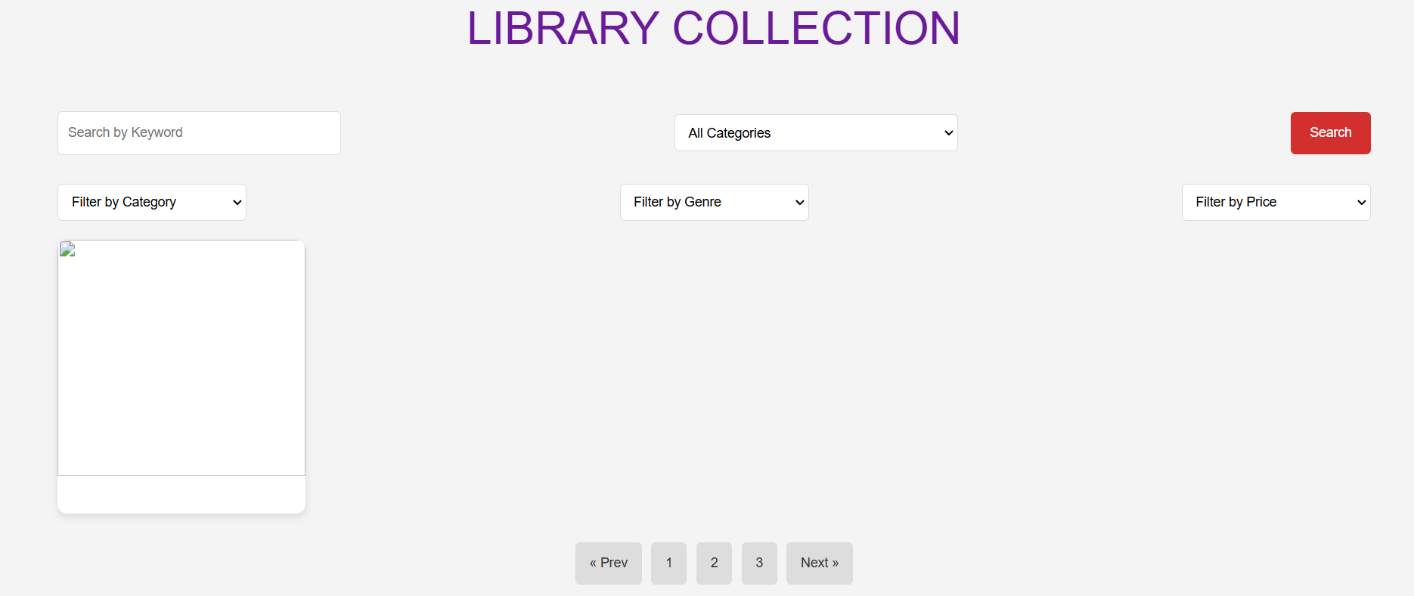


Figure .Library

The Library page serves as the main catalog of books available for browsing, purchase, or rental. Books are displayed in a grid or list view, with each entry showcasing its cover image, title, genre, and stock availability. Users can use a search bar or category filters to narrow down their choices. Clicking on a book directs the user to its detailed view. The page integrates with the backend via PHP to fetch and display book data dynamically from the database.

#### 3.3.9 Services

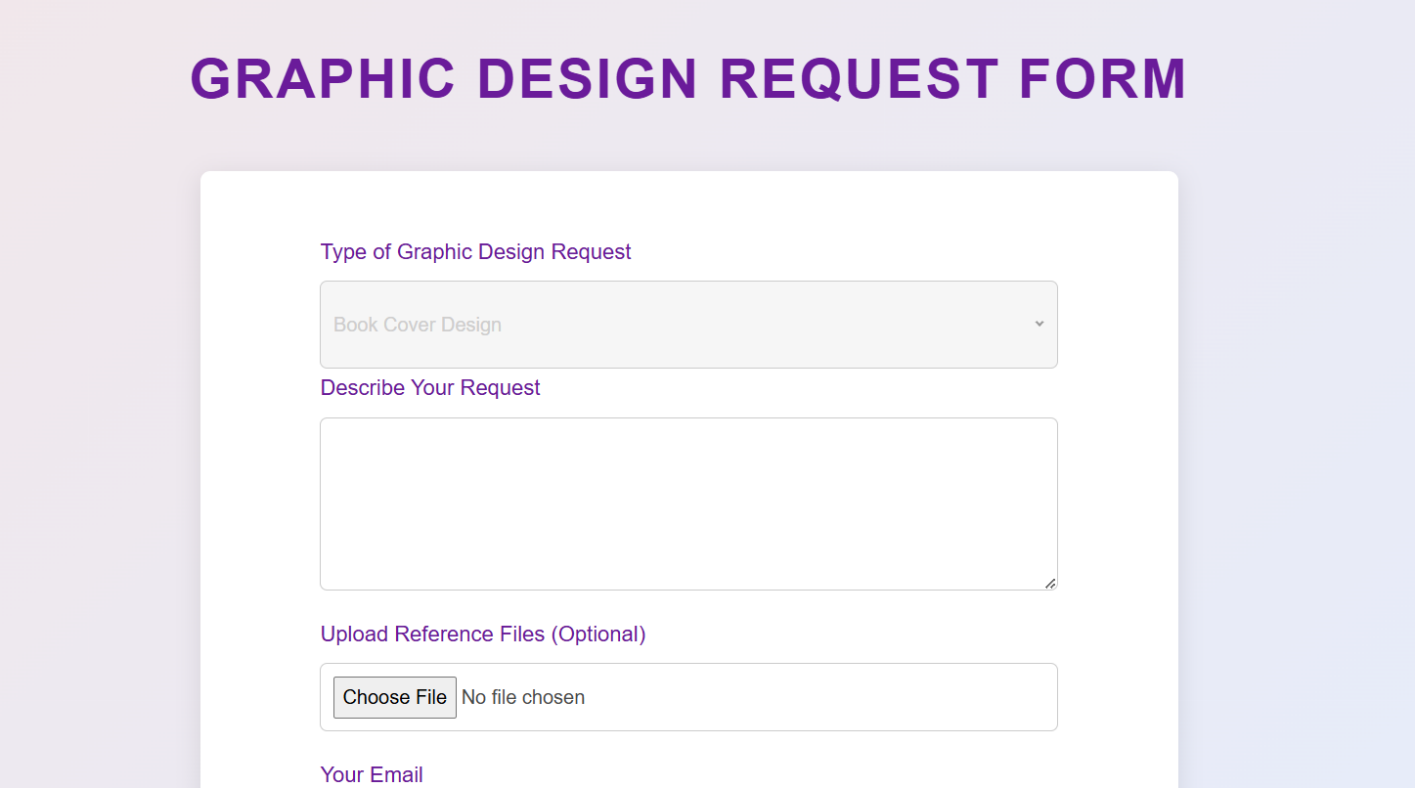


Figure .Services

The Services page outlines additional offerings such as graphic design requests or other creative services. Users can fill out a detailed request form specifying their requirements. JavaScript validates the form inputs to ensure accuracy and completeness. Upon submission, an Ajax request forwards the data to a PHP backend function, which stores the service request in the database for processing. A confirmation message is displayed, and users can track their requests through their dashboard.

#### 3.3.10 Profile (dashboard)

#### 3.3.10.1 profile for users:

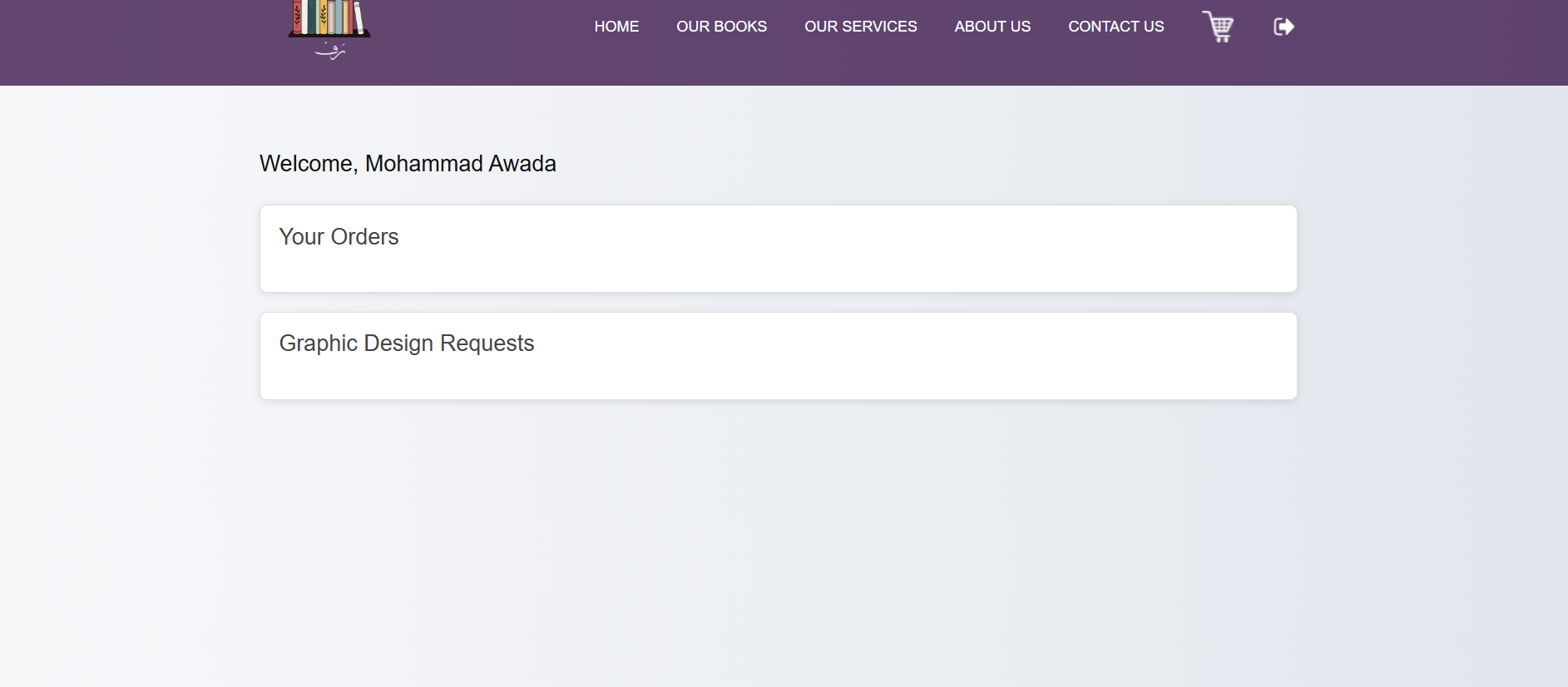


Figure .user profile

The **User Profile Page** provides users with an overview of their personal information, orders, rented books, and graphic design requests. It greets users by name, fetched from the database, and displays their past orders, including details such as total price, shipping address, and books ordered, along with rented books tied to each order. The page also lists rented books, showing rental prices and periods. Additionally, users can view their graphic design requests with information like request type, creation date, and status. This page is designed to give users easy access to their transaction history and requests.

#### 3.3.*10.2* profile for Admin

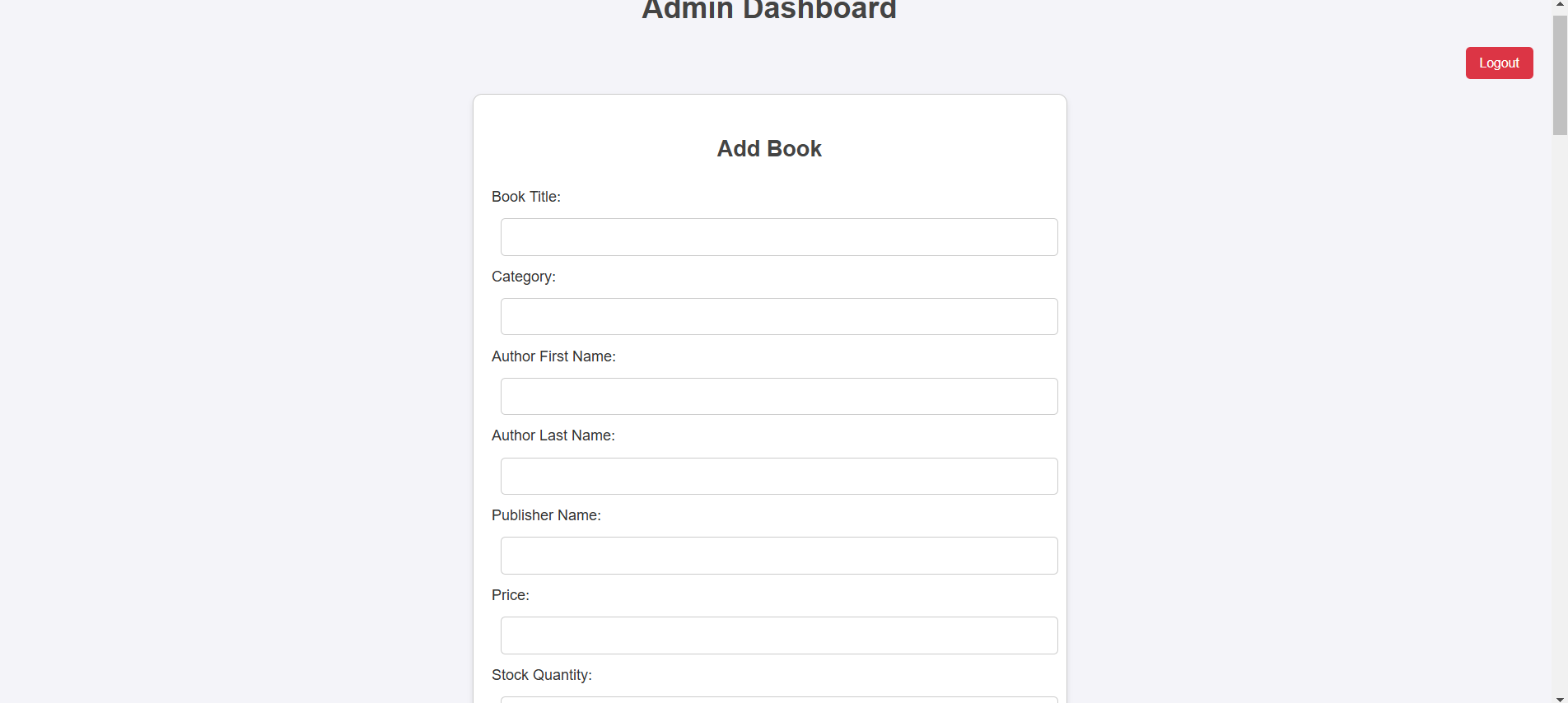
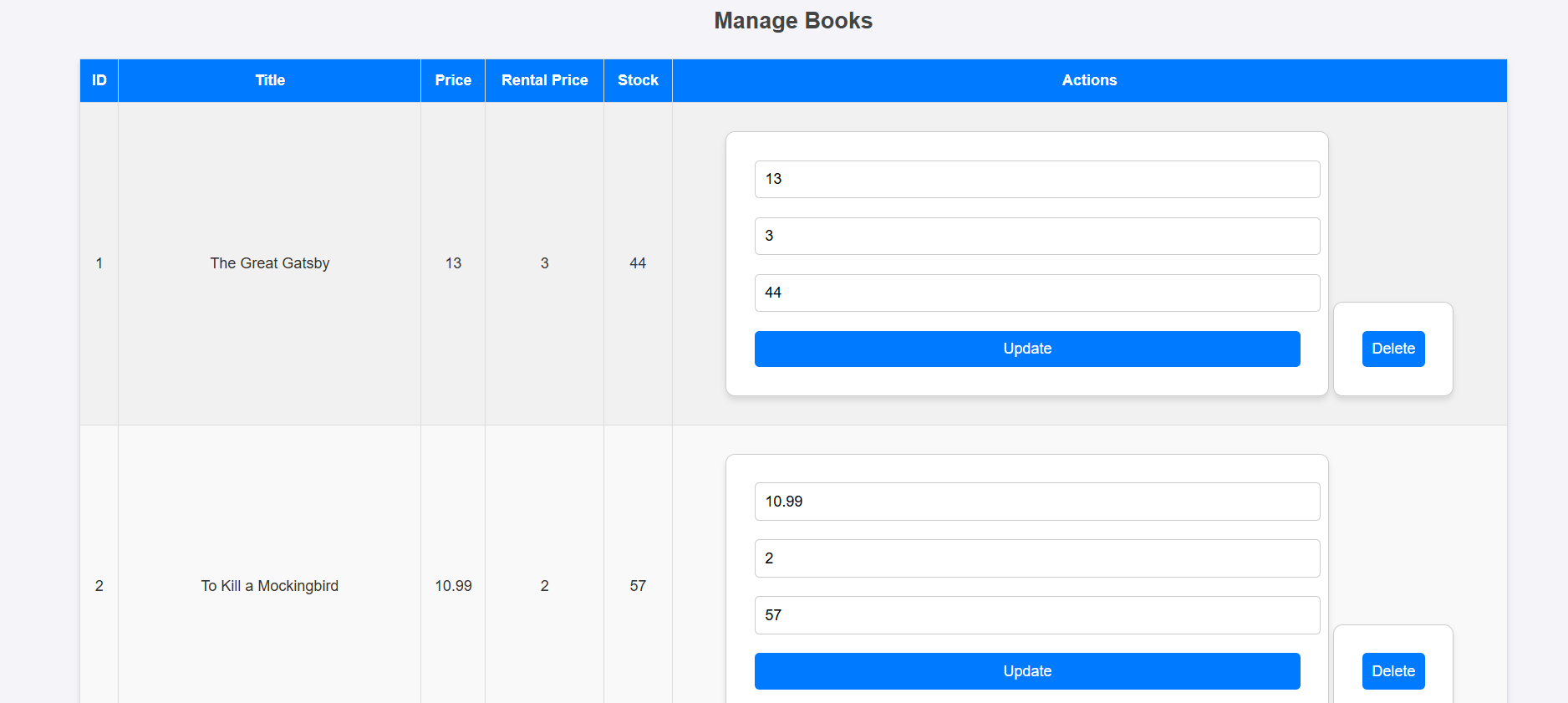
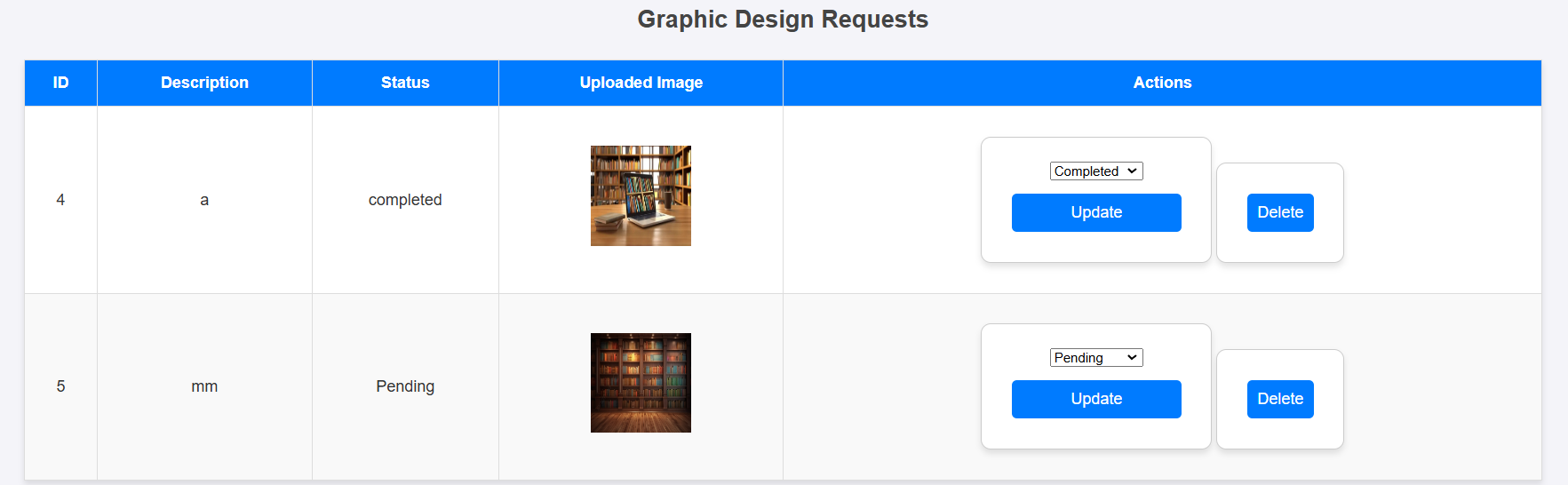


Figure .admin dashboard

****



The admin has several management capabilities through the Admin Dashboard. They can add new books by entering details such as the book title, category, author, publisher, price, stock quantity, genre, description, and language, along with uploading an image. The admin can also update existing book information, including price, rental price, and stock quantity, ensuring everything is accurate. Additionally, they have the ability to delete books from the system. The admin can manage graphic design requests submitted by users, viewing each request’s description, status, and any uploaded images. They can update the status of these requests to pending, in-progress, or completed, and also delete requests when necessary. This centralized platform enables the admin to efficiently manage both the books and the design requests, maintaining smooth operations for the system.

# CHAPTER 4: Future Plan and Conclusion

The online library project is a user-focused platform that seamlessly integrates book browsing, purchasing, and creative services with an easy-to-navigate interface. The system is comprised of several key pages and dashboards, each designed to meet specific user roles and functionalities. These pages include **Home**, **Library**, **Cart**, **Checkout**, **Services**, **About Us**, **Contact Us**, **Sign In**, and **Sign Up**, along with dashboards tailored to users, admins, and managers.

## System Pages and Features

1. **Home Page**:
   * Serves as the central hub, showcasing the platform’s purpose, trending books, and featured services.
2. **Library Page**:
   * Allows users to browse and search books by genre, category, author, or language, with filters and sorting options for a refined experience.
   * Features personalized book recommendations based on user preferences.
3. **Cart and Checkout Pages**:
   * Enables users to add books to their cart and complete purchasing or rental transactions seamlessly, including options for digital and physical goods.
4. **Services Page**:
   * Provides a platform for users to submit custom graphic design requests, such as book cover design, promotional materials, and illustrations.
5. **About Us Page**:
   * Highlights the mission, vision, and development team behind the project, building trust and credibility.
6. **Contact Us Page**:
   * Includes a feedback form and contact details for users to submit inquiries, suggestions, or feedback.
7. **Sign In and Sign Up Pages**:
   * Allow users to securely log in to their accounts or register for new accounts with easy navigation.

**Role-Specific Dashboards**

1. **User Dashboard**:
   * Explore, rent, or purchase books.
   * Submit custom design requests.
   * Access a personalized book collection, reading history, and book recommendations.
   * View and manage reviews and ratings.
2. **Admin Dashboard**:
   * Manage books, categories, and authors.
   * Oversee user accounts and assign roles (e.g., user, admin).
   * Review and approve custom design requests.
   * Manage orders and track inventory.
3. **Manager Dashboard**:
   * Monitor book rentals, purchases, and other user activities such as reviews and feedback.
   * Generate and analyze system usage reports.
   * Handle subscription-based rentals, respond to user inquiries, and resolve complaints.

**Future Enhancements**

To ensure continuous evolution and address users' needs, the following features are planned for future development:

* **AI Chatbot**: Integrate a smart assistant capable of providing 24/7 support, offering book recommendations, answering FAQs, and assisting with user issues.
* **User Profile Pictures**: Allow users to personalize their dashboards by uploading profile pictures for a more tailored experience.
* **Private Messaging**: Add private chat capabilities for users to directly communicate with admins, support staff, or other users.
* **File-Sharing Capabilities**: Facilitate collaboration by enabling users to securely share files for design requests or feedback.
* **Mobile-Friendly Optimization**: Adapt the platform to be fully responsive on mobile devices, ensuring smooth navigation and access on the go.
* **Publishing the Website Online**: Make the platform globally accessible, supporting a broader audience, with future integration for international markets and diverse languages.

**Conclusion**

This online library system combines modern technologies, user-friendly design, and innovative features to create a comprehensive digital platform for book enthusiasts and creative service seekers. With already implemented functionalities and planned future enhancements, the platform is well-positioned for growth and scalability. By continuously addressing evolving user needs and integrating advanced features, this platform aims to provide a superior user experience in the ever-growing online library ecosystem.

# References

 Atlassian. (n.d.). **Trello**. Retrieved from <https://trello.com/en>

 Atlassian. (n.d.). **Trello**. Retrieved from <https://trello.com/>

 Bootstrap. (n.d.). **Bootstrap Framework**. Retrieved from <https://getbootstrap.com/>

 CSS. (2016). **CSS3 Introduction**. Retrieved from <http://www.w3schools.com/css/css3_intro.asp>

 Datatables. (n.d.). **Manual**. Retrieved from <https://datatables.net/manual/data/>

 Dave. (n.d.). **Asana**. Retrieved from <https://asana.com>

 Documentation, S. O. (2019). **MySQL**. Retrieved from <https://books.goalkicker.com/MySQLBook/>

 DUCKETT, J. (2014). **JAVASCRIPT & JQUERY**.

 Jacobs, S. (2006). **Beginning XML with DOM and Ajax**. Retrieved from <https://books.google.com.lb/books/about/Ajax.pdf>

 Javascript. (2015). **Programmer's Reference**. Retrieved from <http://indexof.es/JS/JavaScript/>

 Krieger, F. (n.d.). **Project Management Features**. Retrieved from <https://www.scoro.com/blog/features-of-project-management-software/>

 PHP. (2016). **PHP Tutorials**. Retrieved from <https://www.tutorialspoint.com/php/php_tutorial.pdf>

 Rouse, M. (2014). **HTML5 Definition**. Retrieved from <http://searchsoa.techtarget.com/definition/HTML5>

 Sybase PowerDesigner. (2008). **Data Modeling Guide**. Retrieved from <http://img2.insight.com/graphics/at/vendor/sybase/pd-15-overview.pdf>

 Visual Studio. (2017). **Professional Visual Studio 2017**. Retrieved from https://www.slideshare.net/zapuxij/professional-visual-studio-2017-book