

**Project Report
On
Book Exchange and Sell Website**



Date of Submission: 12th July 2020

BOOK EXCHANGE AND SELL

By

Md. Abir Hasan Dipu

ID: 161-15-660

AND

Israt Jahan Nourin

161-15-711

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Naznin Sultana

Assistant Professor

Department of CSE

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY DHAKA, BANGLADESH

APRIL 2020

APPROVAL

This Project titled “Book Exchange and Sell”, submitted by Md. Abir Hasan Dipu and Israt Jahan Nourin to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 12/07/2020.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman

Dr. Md. Ismail Jabiullah
Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Md. Reduanul Haque
Senior Lecturer

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. Mohammad Shorif Uddin
Professor and Chairman

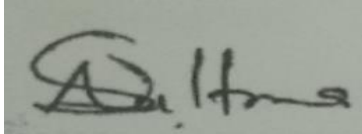
Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Naznin Sultana, Assistant Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



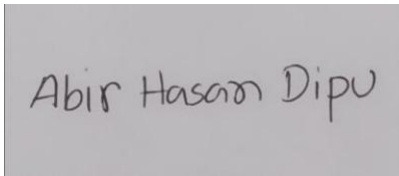
Naznin Sultana

Assistant Professor

Department of CSE

Daffodil International University

Submitted by:

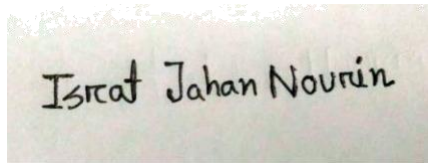


Md. Abir Hasan Dipu

Id: 161-15-660

Department of CSE

Daffodil International University



Israt Jahan Nourin

Id: 161-15-711

Department of CSE

Daffodil International University

ACKNOWLEDGEMENT

At first, we must give our gratitude to the Almighty because without his grace we never could finish the final year project successfully.

We are greatly indebted to our most respected supervisor **Naznin Sultana**, Assistant professor of Department of CSE Daffodil International University, Dhaka. Without her great understanding and knowledge of “Web Design and Development” which helped us in completion of the project. Her tremendous perseverance and her pinpoint inspection helped us in finishing this project. She pointed our mistakes and helped us overcome those. And directed us to the right decision. Without her help it would have been a lot harder to finish this project.

We would like to show our profound gratefulness to our Head, Department of CSE, without his gracious aid we could not have been able to finish our project.

And we would like to thank all the other members of faculty and the staffs of the CSE department of DIU.

We would also like to thank our companions who came this far with us in DIU, who helped us by discussing about the project and the course activity.

Finally, we would like to thank our parents. Because of their love and sacrifice we have come this far.

ABSTRACT

Most of our daily activities are now tied to the technology nowadays. It's a blessing that we live in an era of science and technology. Almost everyone is reliant on web-based systems now. because internet has taken over almost every single point in our life. As the world gets more modern the people also get modern, they want an easier life. Our project is based on this specification because we make the life of the book readers easier. If someone wants a book first, he has to travel to the book market if he wants a greater collection which they won't find in a book store. But buying a specific book from a book market is like finding a needle in a haystack. And even if we are just browsing the books and looking for something our heart desires, it is very hard to do. Because the place is always too crowded. But by using "BookMart" people can easily get their book of desire without the hassle of going to a book market, and finding the book and they can browse a vast sea of books. Users can register their credentials and can give positive or negative comments, so other people can get an idea what to expect. This website will bring us one step closer to the grand future.

TABLE OF CONTENTS

CONTENTS	PAGE NO
Title	i
Approval	ii
Declaration	iii-iv
Acknowledgements	v
Abstract	vi
 CHAPTER 1: INTRODUCTION	 1-3
1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	2
1.4 Expected Outcome	2
1.5 Report Layout	2
1.6 Summary	2
 CHAPTER 2: BACKGROUND	 4-8
2.1 Introduction	4
2.2 Related Work	4
2.3 Figures of Related Work	5
2.4 Comparative Studies	7
2.5 Challenges	7
2.6 Summary	8
 CHAPTER 3: REQUIREMENT SPECIFICATION	 9-15
3.1 UML Diagram	9-15
3.1.1 Use-case diagram	9
3.1.2 Description of Use case diagram	9
3.1.3 System Flow diagram	11
3.1.4 Entity Relationship diagram	13
3.1.5 UML class diagram	14
3.2 Design Requirements	14
3.3 Summary	15

CONTENTS	PAGE NO
CHAPTER 4: SYSTEM DETAILS	16-27
4.1 Introduction	16
4.2 Application Feature	16-17
4.2.1 Registration	16
4.2.2 Login	17
4.2.3 Updating Profile Picture	17
4.2.3 Search Accounts	17
4.2.5 Add and Delete Friends	17
4.2.6 Chatting System	17
4.2.7 Post Image and Text	17
4.2.8 Post in Friend's Profile	17
4.2.9 Delete Posts	17
4.2.10 Profile page	17
4.2.11 Infinite Home page	17
4.2.12 Accept Friend Request	17
4.2.13 Admin Panel	17
4.2.14 Post and View Ads	17
4.3 Front-End Design	18-25
4.4 Back-End Design	26
4.5 Implementation Requirements	26
4.6 Hardware Requirements	26
4.7 Software Requirements	26
CHAPTER 5: IMPLEMENTATION AND TESTING	28-39
5.1 Introduction	28
5.2 Implementation of Database	28-35
5.3 Implementation of front-end design	35-37
5.4 Testing Implementation	37-39
5.4.1 Unit Testing	37
5.4.2 Integration Testing	37
5.4.3 Functional Testing	38
5.4.4 System Testing	38
5.4.5 Usability Testing	38

CONTENTS	PAGE NO
5.5 Test Result	38-39
5.5.1 Results of Unit Testing	38
5.5.2 Results of Integration Testing	39
5.5.3 Results of Functionality Testing	39
5.5.4 Results of System Testing	39
5.5.5 Results of Usability Testing	39
CHAPTER 6: CONCLUSION & FUTURE SCOPE	40-41
6.1 Discussion & Conclusion	40
6.2 Limitations of the Project	40
6.3 Scope and Further Development	40
REFERENCES	41
PLAGIARISM CHECKING REPORT	42

LIST OF FIGURES

FIGURES	PAGE NO
Figure 2.3.1: Related Site1	5
Figure 2.3.2: Related site2	6
Figure 3.1.1: Use case of BookMart	9
Figure 3.1.3: System Flow Diagram	12
Figure 3.1.4: E-R Diagram	13
Figure 3.1.5: UML Class Diagram	14
Figure 4.3.1: Registration Page	18
Figure 4.3.2: Login Page	18
Figure 4.3.3: Infinite Home Page	19
Figure 4.3.4: Profile Page	19
Figure 4.3.5: Messaging Page	20
Figure 4.3.6: New Message	20
Figure 4.3.7: Friend Request Page	21
Figure 4.3.8: Post Advertisement Page	21
Figure 4.3.9: Advertisements Page	21

Figure 4.3.10: Book Details Page	21
Figure 4.3.11: Profile Picture Upload Page	22
Figure 4.3.12: Admin Login	23
Figure 4.3.13: Admin User Management	24
Figure 4.3.14: Admin Post Management	25
Figure 4.3.15: Admin Ad Management	

FIGURES	PAGE NO
Figure 4.4.1: Back-end database	28
Figure 5.2.1: Database Connection	29
Figure 5.2.2: Registration Handler (1-3)	30-31
Figure 5.2.3: Register	32
Figure 5.2.4: Login Handler	32
Figure 5.2.5: Login Form	33
Figure 5.2.6: Query for Insert Post	34
Figure 5.2.7: Query for Display Post	35
Figure 5.2.8: Query for Likes and Unlikes	36
Figure 5.3.1: Initialization of CSS and Javascript Library and Frameworks	36
Figure 5.3.2 Code for including header file	37
Figure 5.3.3 Code for Register and Login Toggle	38
Figure: Plagiarism checking report	42

LIST OF TABLES	
TABLES	PAGE NO
Table 3.1.2(a): Use case of login	10
Table 3.1.2(b): Use case of searching and friend requests	11
Table 3.1.2(c): Use case of sending message	11

CHAPTER 1

INTRODUCTION

1.1 Introduction

This is an age of science. And internet is playing a major role in our life. Almost all of our activities are actively or passively related to the internet.

In this era, we do not beat around the bushes when we need something. We just open up a browser and search for the content of our desire on a search engine and get it. It actually saves up a lot of time which can be invested elsewhere.

Everybody in this world knows that knowledge is power. In order to gain that power, we have to study a lot. There isn't a better way to gain knowledge than reading books. It can be any book, from the formal text books to the very informal story or novels. Every single book takes a part in making our mind rich. But the main problem is getting the book of our desire.

Anyone who went to a book market to find a book which is remotely old will say it is not an easy task. We have to go through a lot just to get a book. In this process a lot of time gets wasted.

This is why we came up with the website "Book Mart" for the people with a hunger of knowledge.

We have put tremendous effort to make this website which will successfully get rid of all of the problems related to buying books. In this website one can buy and sell brand new as well as used books. And members can exchange their books with each other.

1.2 Motivation

To get make a platform which is very convenient way of getting book from anywhere in the country. We thought the inconvenience we face during a trip to the book market. It is easy to give up on a rare book because of the searching hundreds of stores unsuccessfully and after finding the book it will be priced so much that no one is willing to buy an old book with such a high price.

By using the simple childhood practice of sharing books as an inspiration we have developed a website where book lovers can buy, sell for fair price and even exchange books with each other.

We discussed about this project with our honorable supervisor Naznin Sultana (Assistant Professor, DIU-UC). She approved our project with enthusiasm and directed us to start the project.

1.3 Objectives

For developing our website, we have some objectives which must be fulfilled. The main objectives are following

- Developing a website which is easy to use.
- Gathering data of about the design of the website.
- Getting all the resources.
- Giving controls to the users for maintaining their profile.
- Giving the users a way to communicate with each other.
- Developing the website with the resources and codes.
- Finding and fixing the bugs.

1.4 Expected outcome

The expected outcomes of the website are following:

- It will be user friendly and aesthetic.
- Users will be able to maintain their profile to their liking.
- People can visit the website without registration.
- In order to interact registration is mandatory.

1.5 Report Layout

The layout of this report on the “Book Mart” website is given below:

- All of the topics are about related information.
- Images are used to make the topics easier to understand.
- Font and font size are used as per requirements.
- Margins and spaces are used as per requirement.
- Order and menus are applied per requirement.

1.6 Summary

Books should be a very important part of our life. Because books are the best source of knowledge. By making the process of getting books easier we are opening a door for more knowledge. This

platform will bring people closer by giving them the opportunity to share their used books. People can also sell their books to a person who wants to enrich their collection so both parties will get benefitted. This website will open a new horizon for book lovers.

CHAPTER 2

BACKGROUND

2.1 Introduction

This idea behind this website is to make life easier for the book lover, it gets rid of the problems related to going to a book market and searching for a book for hours. It makes the manual work of finding book fully automatic. If someone goes to the book market looking for a book, he has to go through a lot of stores to find the book which is very inefficient and time consuming. Our main goal is to save the time and energy. So, we focused on making the website user friendly and easy to use.

2.2 Related work

There are systems which are similar to our system. We wanted to make something exceptional in our system. There are systems which are more imposing but many emergent information are missing there. If we compare our system with those systems, our system is not that much entrancing but it provides all the necessary information and the price rate will be comparatively cheap. We tried to make it well disposed.

Here are some example of shopping system and those websites features and drawbacks.

Website link:

- <https://bikroy.com/bn> [1]
- <https://www.ebay.com/> [2]

2.3 Figures of related work

Figure 2.3.1 and 2.3.2 show an online shopping system named as <https://bikroy.com/bn> [1].

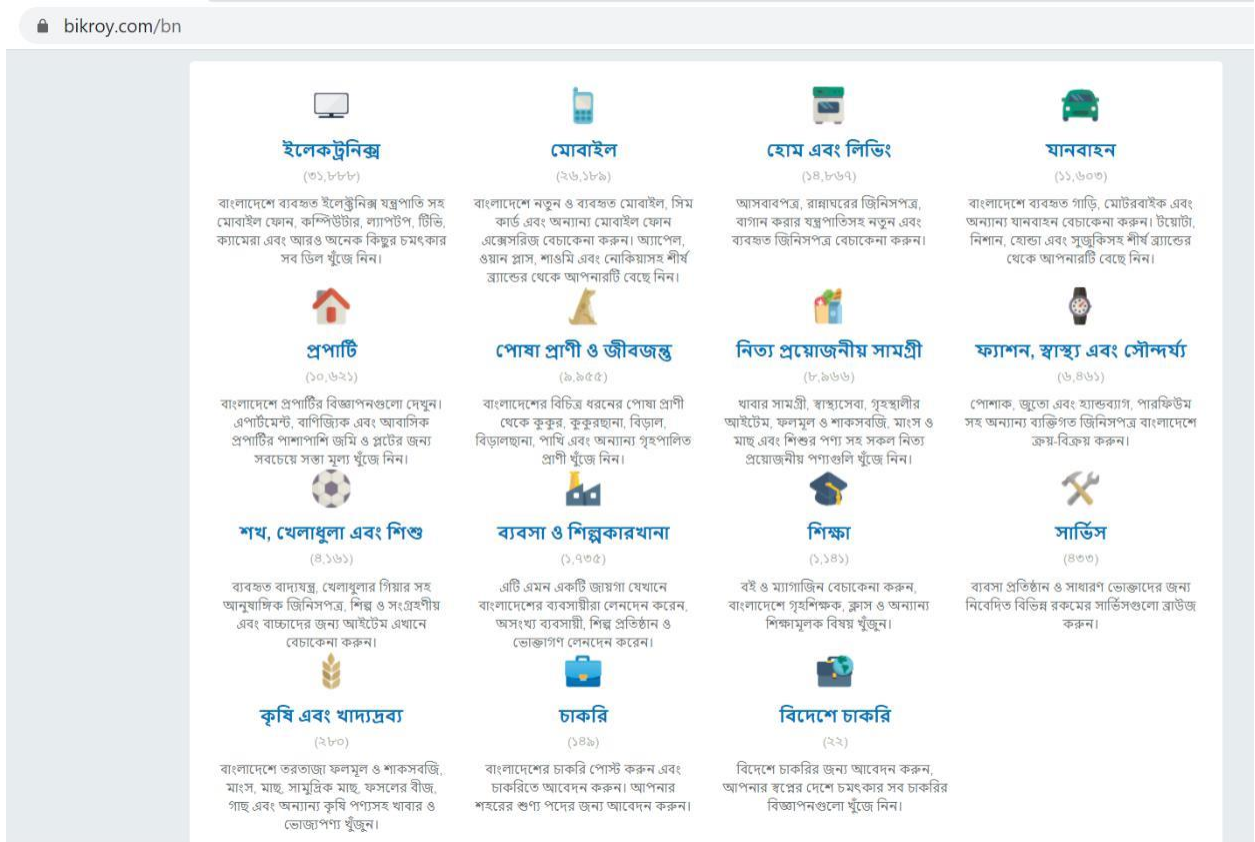


Figure 2.3.1: Related site 1 (page 1)

সকল শ্রেণী

🏠 শিক্ষা

টিউশন (827)

পাঠ্য বই (227)

বিদেশে শিক্ষা (49)

কোর্স (27)

অন্যান্য শিক্ষা (11)

অবস্থান

বাংলাদেশ-এর সবগুলো

ঢাকা (998)

চট্টগ্রাম (39)

ঢাকা বিভাগ (17)

খুলনা (16)

সিলেট (15)

আরও দেখান ▼



masters in English book set sale

ঢাকা, পাঠ্য বই

ট ১,০০০

৫ ঘণ্টা



Accounting Past Paper for O levels

চট্টগ্রাম, পাঠ্য বই

ট ৪৮০

৫ ঘণ্টা



এস.এস.সি পরীক্ষার ছুটিতে Excel, Word এ...

সদস্য ঢাকা, কোর্স

ট ৩,০০০

৯ ঘণ্টা



EX ISD TUTOR AVAILABLE AT ONLINE

সদস্য ঢাকা, টিউশন

ট ৬,০০০

৯ ঘণ্টা



Dedicate English medium tutor giving for...

সদস্য ঢাকা, টিউশন

ট ৬,০০০

৯ ঘণ্টা



Textile Engineering book for B.Sc (NEW)

খুলনা বিভাগ, পাঠ্য বই

ট ১,৫০০

১৫ ঘণ্টা

Figure 2.3.2: Related site 1 (page 2)

Figure 2.3.3 shows another related site of online shopping system named as <https://www.ebay.com/> [2].

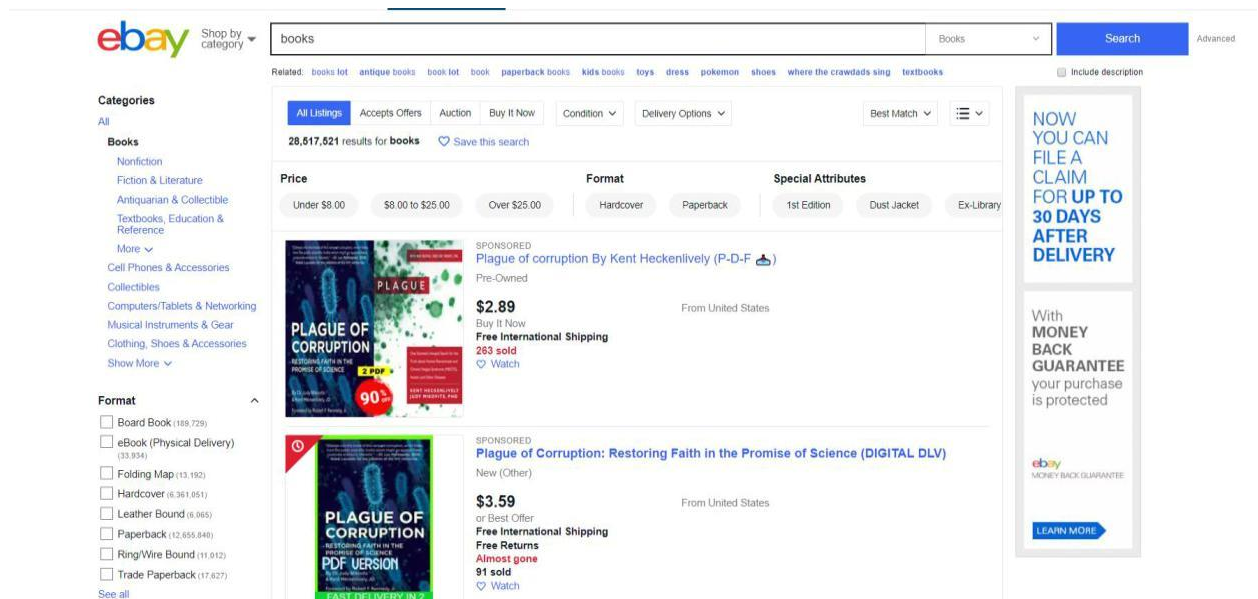


Figure2.3.3: Related site 2

2.4 Comparative Studies

There are some websites which are pretty similar to our website but still there are some dissimilation between our website and those websites. Those websites are pretty imposing but still they have some flaws. As we can see the price rate is not that cheap, but in our website people can get books in a low price and also they will be able exchange books by chatting with each other through our website. Though our website is not that much entrancing but we tried our best to make it more user friendly.

2.5 Challenges

We have faced many complications and challenges while developing this website. We overcame some of the challenges with relative ease. But some of them was totally new for us and required unconventional thinking and problem-solving knowledge to overcome the solutions like

- **Developing a user-friendly interface:** This is the core for any website. If the interface is not easily understood by users, the site will never thrive. This is why we tried to make a very user-friendly interface for our website.

- Creating appropriate database relations: Creating a data base and the relations between tables is a challenging and complicated task. We faced a hard situation while working with database.
- Backend Processing: Websites must have a backend dashboard for maintenance, and it's a hard process to do. We faced the challenge of doing the backend and front-end process simultaneously.
- Developing the chat system: developing the chat system was the hardest challenge for us to overcome. We tried our best to develop it flawless.
- Bugs: Finding out the bugs and fixing them is a very tedious challenge. But we tried our best to keep our website bug free.

2.6 Summary

To develop any system, the most important thing is to explore the related work of that system. We must have proper knowledge about the related systems. Without analyzing no work can be done properly. So, after a lot of analysis we settled upon to amplify it.

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 UML Diagram

In this section we are going to describe out UML diagram. There are many types of UML diagrams, we are going to review four of them.

3.1.1 Use Case Diagram

Use case diagram exemplifies all the actors who are pertinent in the system and then show how they behave in the system.

Figure-3.1.1 shows the use case diagram of BookMart.

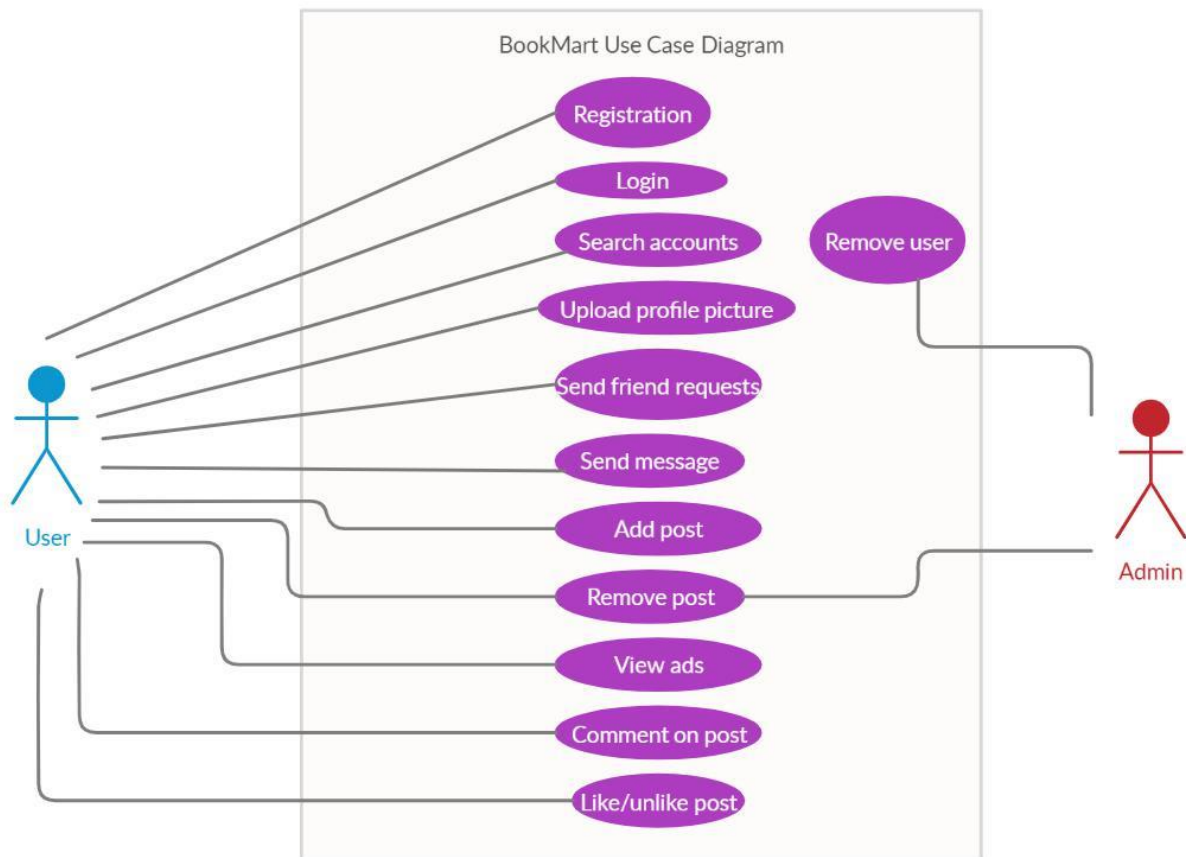


Figure 3.1.1: Use Case of BookMart

3.1.2 Description of Use Case Model

In the use case diagram, we can observe that there are two actors in our system and they are (1) User and (2) Admin. In this system admin can delete anything posted by the users if it does not comply with the rules of the website. Admin can also delete any user for breaking rules. User can visit the site with login and also can contact people and can add people to their friend list. User

can message, write any post to their own page or their friend's page. User can update profile picture and will be able to see all posts posted by their friends. This table 3.1.2(a) shows the login process of the system.

Table 3.1.2(a): Use Case of Login

Cases	Details
Actors	Admin, Users.
Pre-conditions	-Registration is obligatory.
Post-conditions	Ensuing successful login actors will go to their desired dashboard.
Basic Flows	Actors will login with their login credentials. Afterwards they will reach home page.

This table 3.1.2(b) shows the process of searching for a profile and sending friend request.

Table 3.1.2(b): Use case of searching and sending friend request

Cases	Details
Actors	User.
Pre-condition	-Login into the system. -Home page. -Search name of person in the search bar.
Post-condition	After searching the name, user can click and enter the desired profile.
Basic Flow	User will find their friends or send friend request.

This table 3.1.2(c) shows the process of sending message.

Table 3.1.2(c): Use case of sending message

Cases	Details
Actors	Users.
Pre-condition	-Login required
Post-condition	Actor can get messages from their friends
Basic Flow	When an user login to the website he clicks navigation bar at the top and search to send messages to others.

This table 3.1.2(d) shows the process creating a post

Table 3.1.2(d): Use case of creating a post

Cases	Details
Actors	Users.
Pre-condition	Actor can register and create a post containing a photo of their book.
Post-condition	After successfully uploading their post it will be visible to the user and his friends.
Basic Flow	-Firstly user needs to go to his profile -Click Upload profile picture button at the bottom of the profile picture. -Select a picture from the computer -Crop the picture and click save.

3.1.3 System Flow Diagram

Momentarily we are impelling to confer about our functioning flow diagram in figure-3.1.3. Aforementioned system has two types of main actors they are the admin and the users. For user there is no need of being registered for visiting the site, searching products, viewing product details, giving review about product, adding product into cart, and updating cart. But if the user wants to place any order then he/she must need to be logged in by going through registration process. Registered users can manage their profile and able to see purchasing history. Now for getting access of admin panel must need to be registered. After successfully log in the admin will redirect to admin dashboard from where admin can perform the task of changing password, order management, user management, inserting product and managing product. Admins can also visit site as a user after logout.

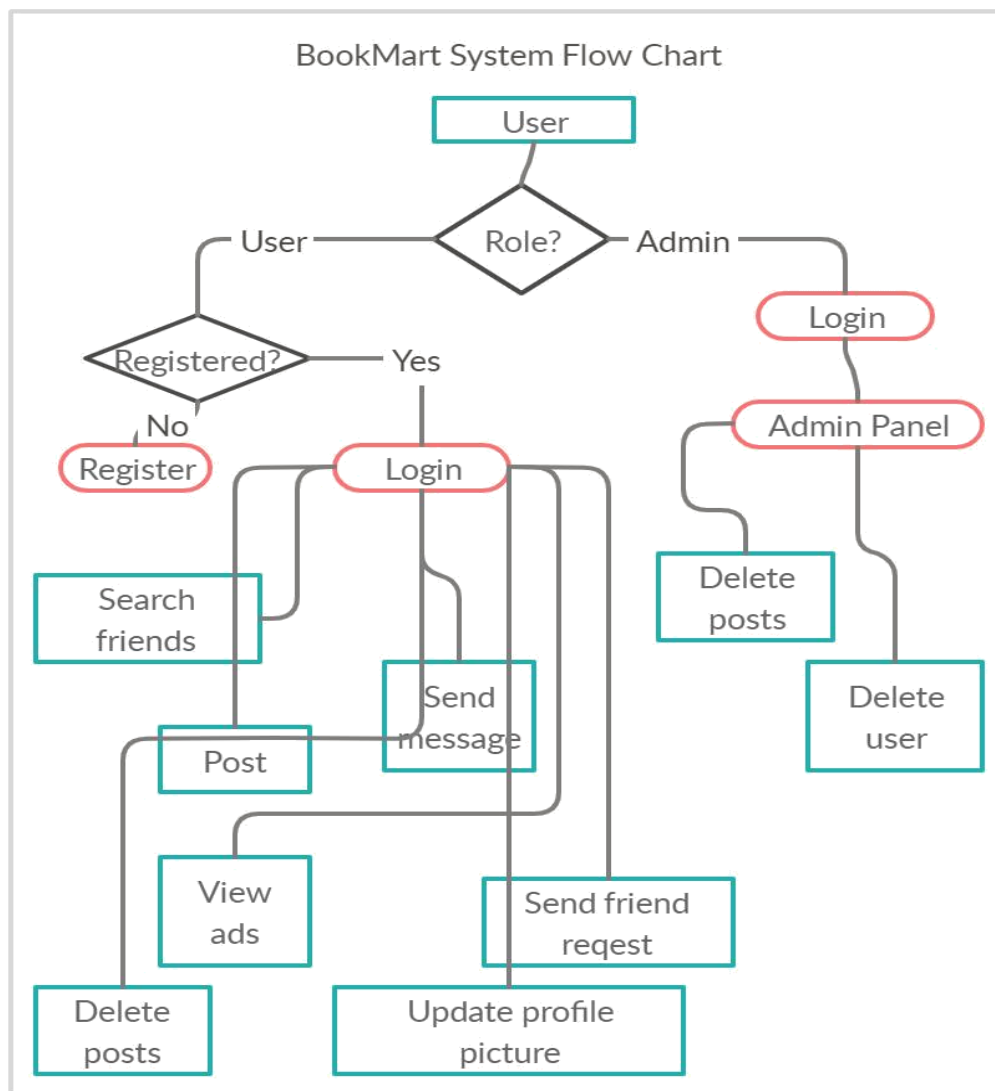


Figure 3.1.3: System flow diagram

3.1.4 Entity Relationship Diagram

The figure 3.1.4 shows the ER diagram of BookMart.

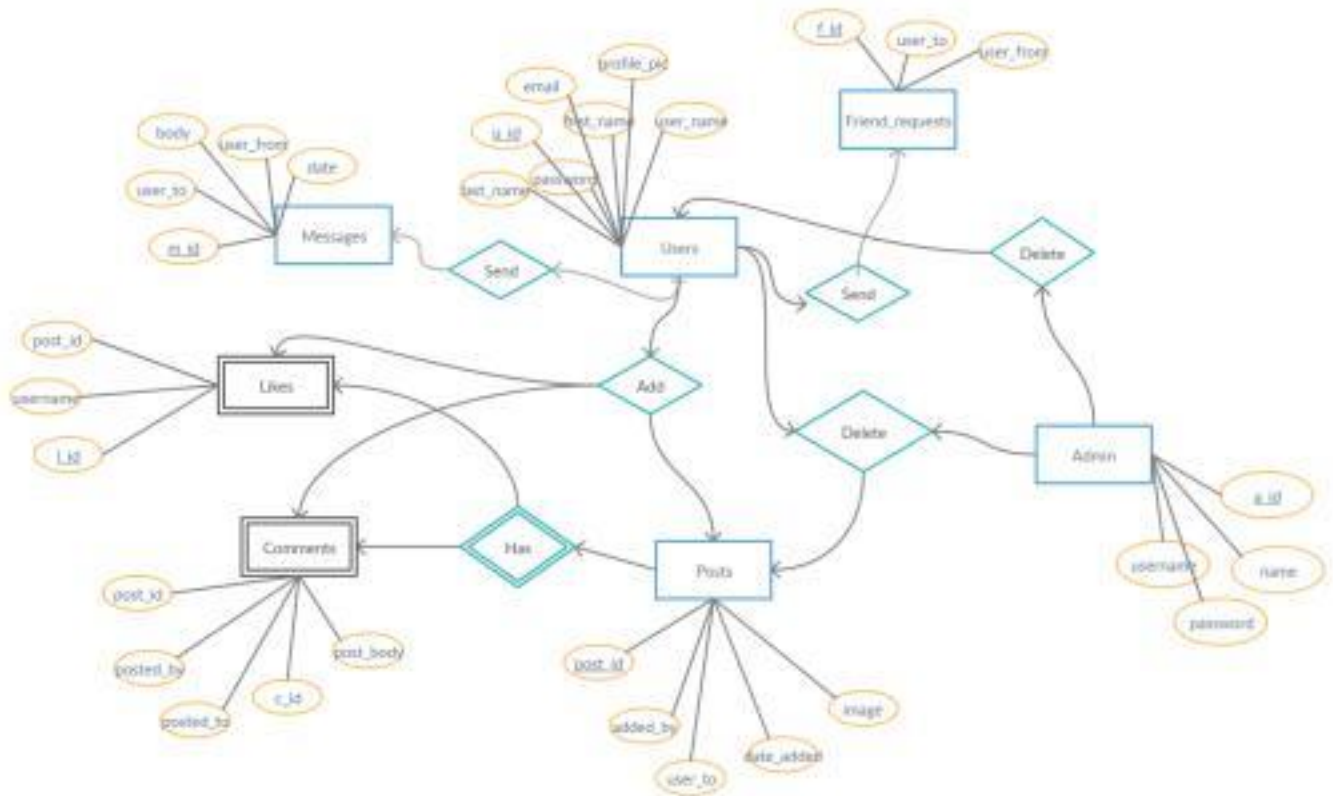


Figure 3.1.4: E-R Diagram

3.1.5 UML Class Diagram

The figure 3.1.5 shows the UML class diagram of BookMart.

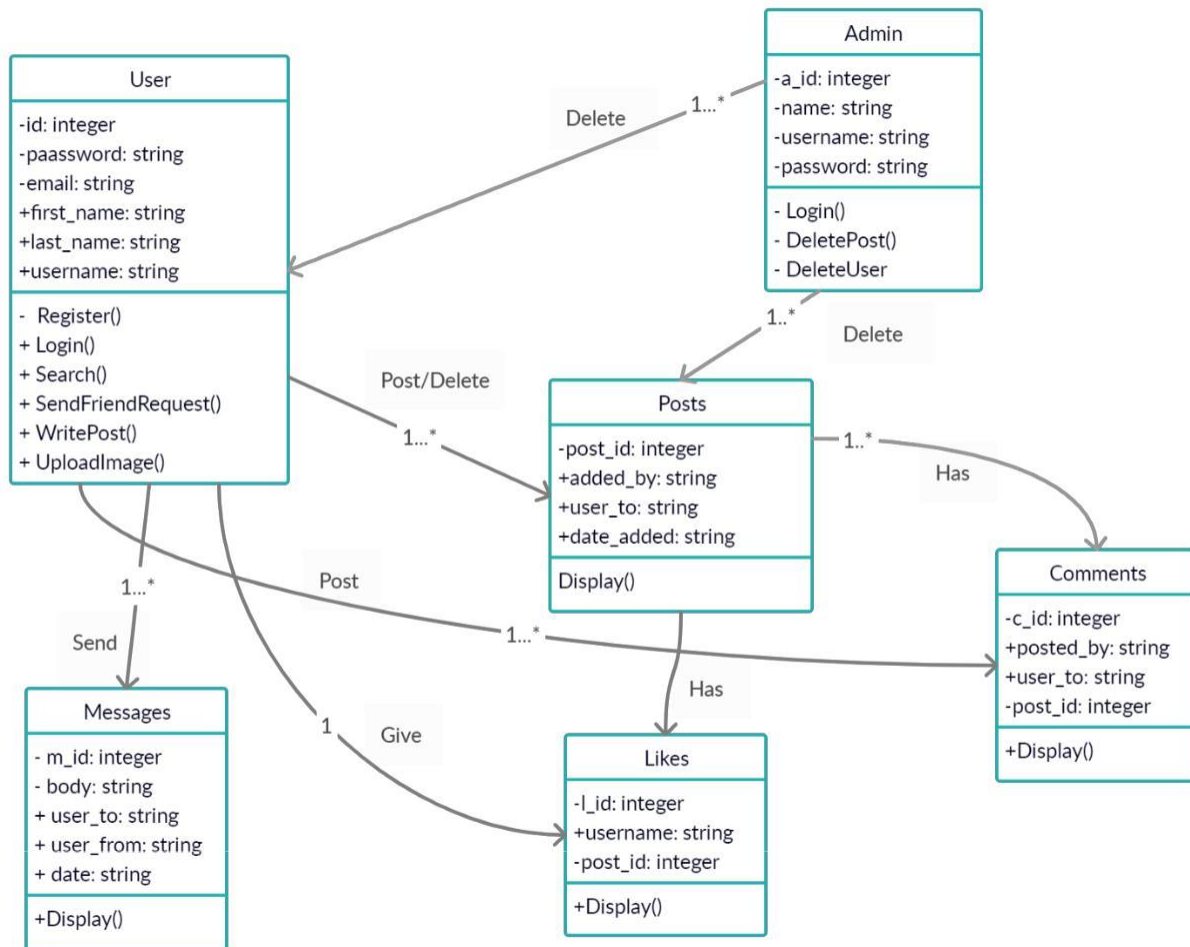


Figure 3.1.5: UML Class Diagram

3.2 Design Requirements

We have to accomplish some designs to form this system more accessible and predictable to purchaser. So in aforementioned system the designs are used they are the frontend and the backend design.

For accomplishing the frontend design we have the obligation to portrayal the layouts into paper first so that we can detect how the website will look agnate. For the front end design we adopt HTML[8], CSS[7], PHP, and urgency to incorporate bootstrap framework, bootstrap, js[5] and

some jQuery[6] for completing the layout. Then we need to design the project as the documentation desideratum and accomplishment coding.

Forthwith for the backend design we have the obligation to perceive about the database structure and the expertise on relational database. There is also obligation to dividend the relationship betwixt the tables in the database and also obligation to keep in mind redundancy things. So that we have urgency of accomplish a few normalizations to avert this considerate of stuff. And for backend coding we tried to follow the object-oriented coding impression. So that's all the design requirements that desired in this project.

3.3 Summary

In aforementioned chapter we conferred about the full project adopting use case diagram, system flow diagram, ER diagram and UML class diagram. We additionally illustrate them sophisticatedly.

CHAPTER 4

SYSTEM DETAILS

4.1 Introduction

In this chapter the system will be described briefly. After reading this chapter one will gain around 75% of knowledge about the application mentioned previously. This chapter contains the description of the system, front-end, back-end, database and the overall implementation.

4.2 Application Feature

We have used several features in our system. Essentially we are impelling to spectacle our features list and posterior we will depict them. So, the features of this system are-

- Registration- Any user must create an account to use this website.
- Login- Logging in will give access to website.
- Updating Profile Picture- Users can update their profile pictures.
- Search Accounts- Users are able to search other profiles.
- Add and Delete Friends- Users are able to add and delete friends.
- Chatting System- Users can talk to each other via chatting system.
- Post Images and Texts- Users can post images and texts in their own profile.
- Post Texts to Friend's Profile- Users can post images and texts in their own profile.
- Delete posts- The posts can be deleted by the user later.
- Profile page- Profile page will allow other users to contact with another user and send him/her friend requests
- Infinite home page
- Admin Panel- Admin panel functionalities contain deleting of ads, posts and users if required

4.2.1 Registration

Any user needs to register to visit this website. To use this website any user must create their account before moving forward.

4.2.2 Login

After registration the uses must login to view or use any other content in the website.

4.2.3 Updating Profile Picture

Any user can choose and upload a picture to set as a profile picture.

4.2.4 Search Accounts

User can search their friends and other people available in the website.

4.2.5 Add and Delete Friends

Users can send friend requests and delete existing friends.

4.2.6 Chatting System

Users can chat with their friends using the messaging functionality.

4.2.7 Post Image and Text

Users are able to post images and texts in their own profile which will be visible in home page.

4.2.8 Post in Friend's Profile

The users can post texts in their friend's profile.

4.2.9 Delete Posts

Users can later delete their own posts.

4.2.10 Profile page

Each use has a profile page

4.2.11 Infinite home page

The home page has infinite post loading functionality

4.2.12 Accept Friend Request

The friend request page shows all the friend requests, from which the user can choose to accept or deny requests

4.2.13 Admin Panel

The Admin have the power to take down any User, Post, or Ad that go against the website guidelines.

4.2.14 Post and View Ads

User can Post Advertisements, from where people can contact the contact the Seller.

4.3 Front End Design

This figure 4.3.1 shows the Registration Page of BookMart website.

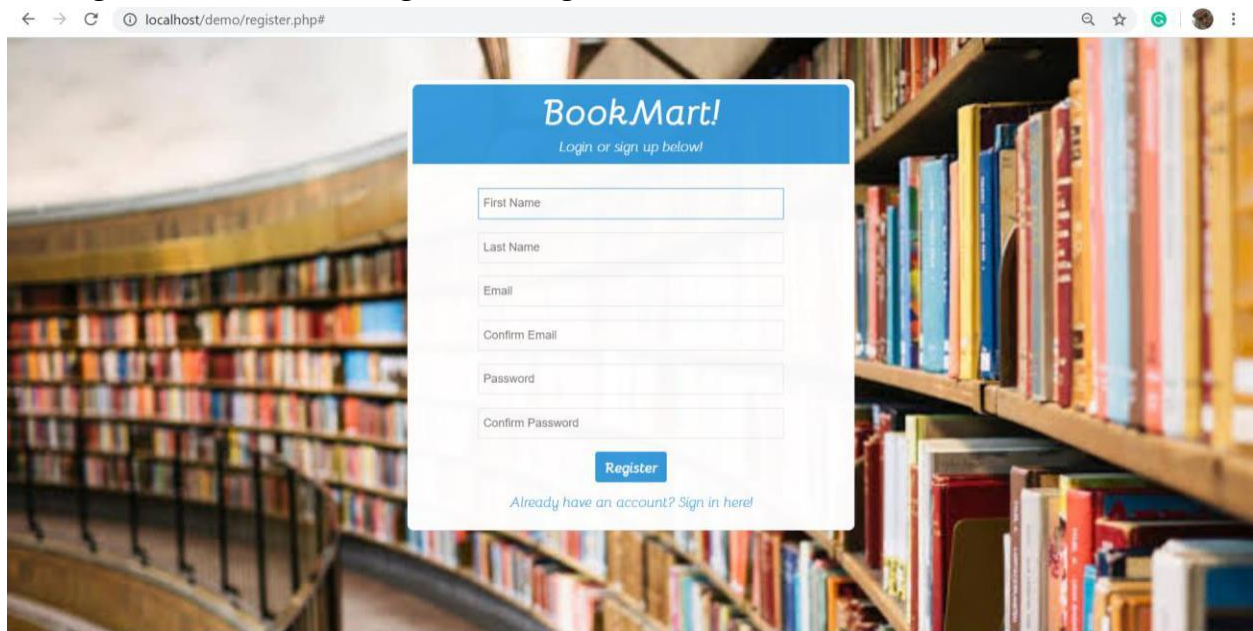


Figure 4.3.1: Registration Page

This figure 4.3.2 shows the Login Page of the website

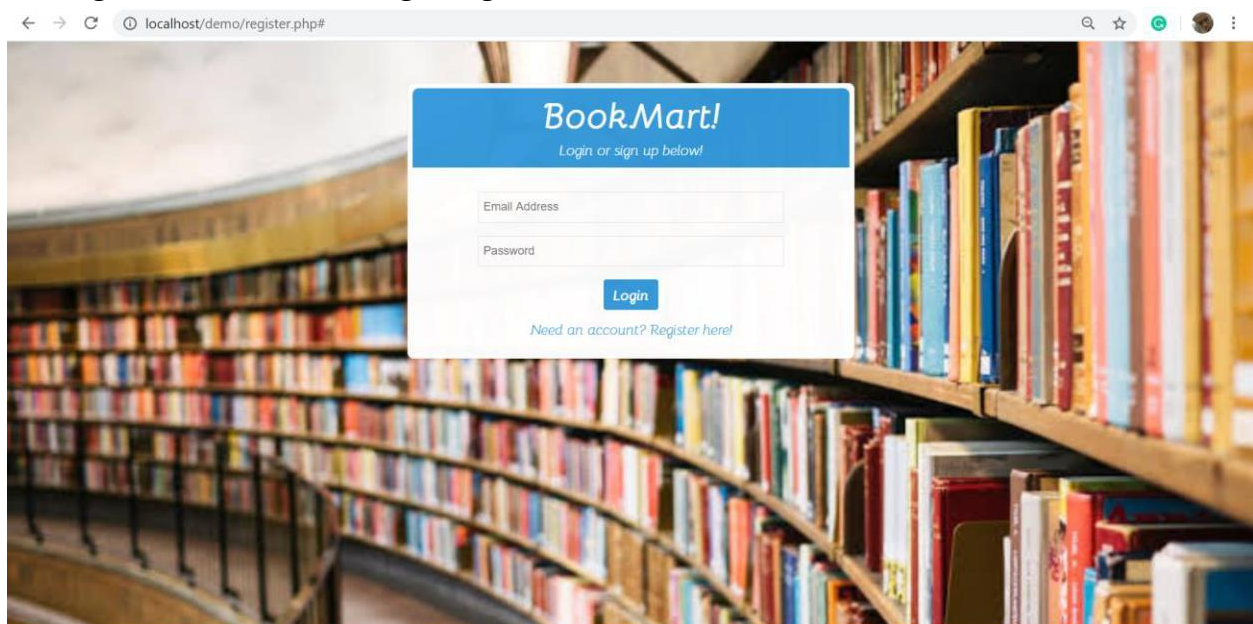


Figure 4.3.2: Login Page

This figure 4.3.3 shows the Home Page of the website.

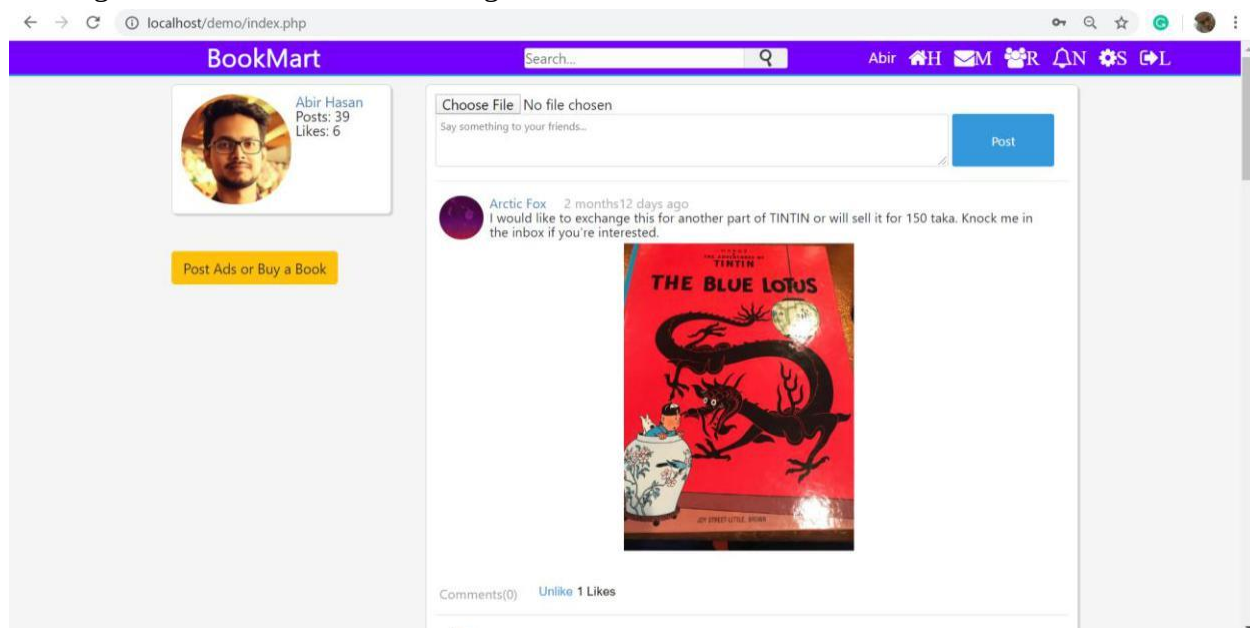


Figure 4.3.3: Infinite Home Page

This figure 4.3.4 shows the Profile Page of Users

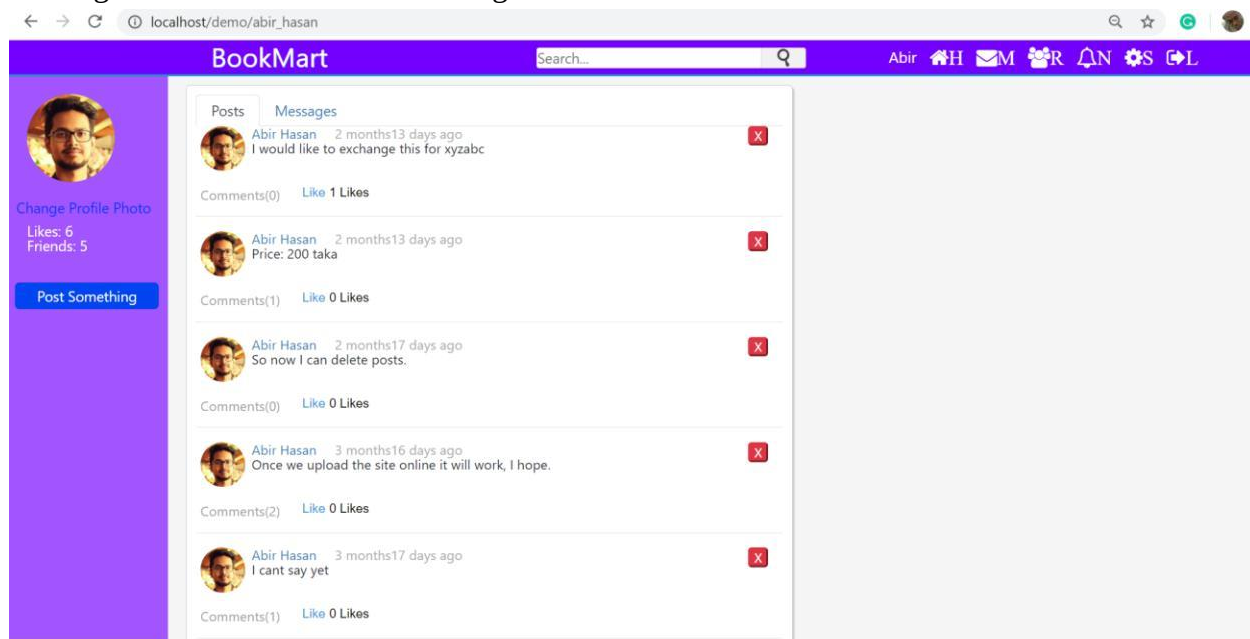


Figure 4.3.4: Profile Page

This figure 4.3.5 shows the Messaging system.

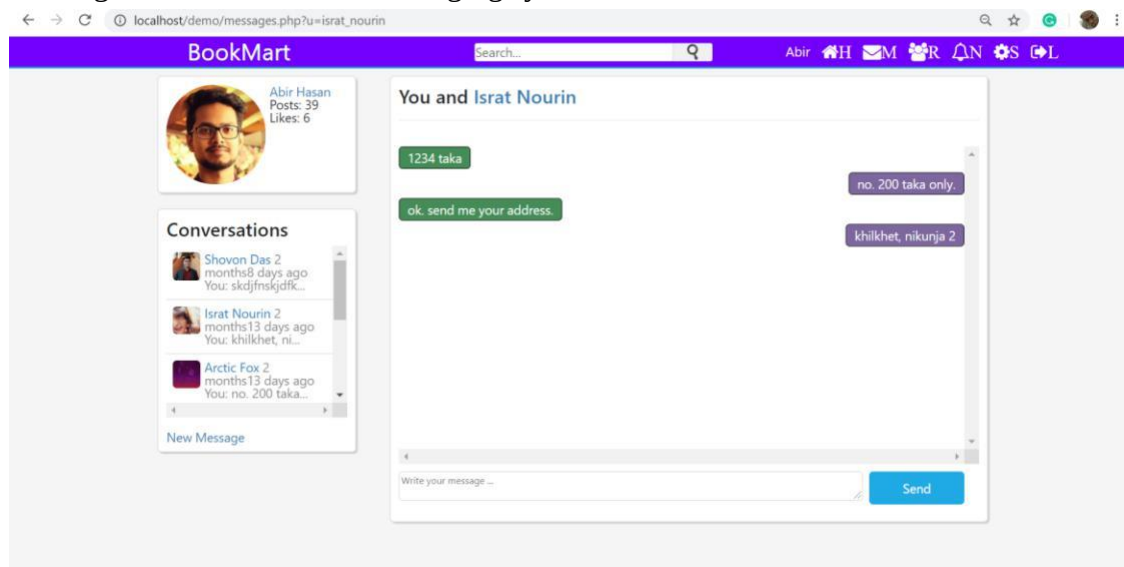


Figure 4.3.5: Messaging Page

This figure 4.3.6 search users to start new conversation

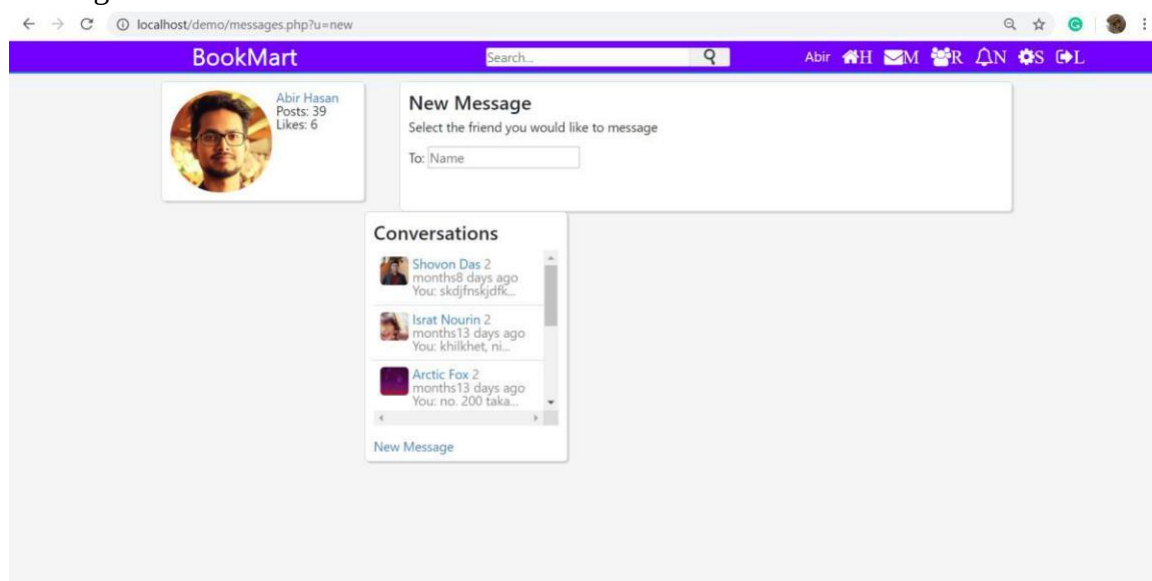


Figure 4.3.6: New Message

This figure 4.3.7 shows the Friend Request page

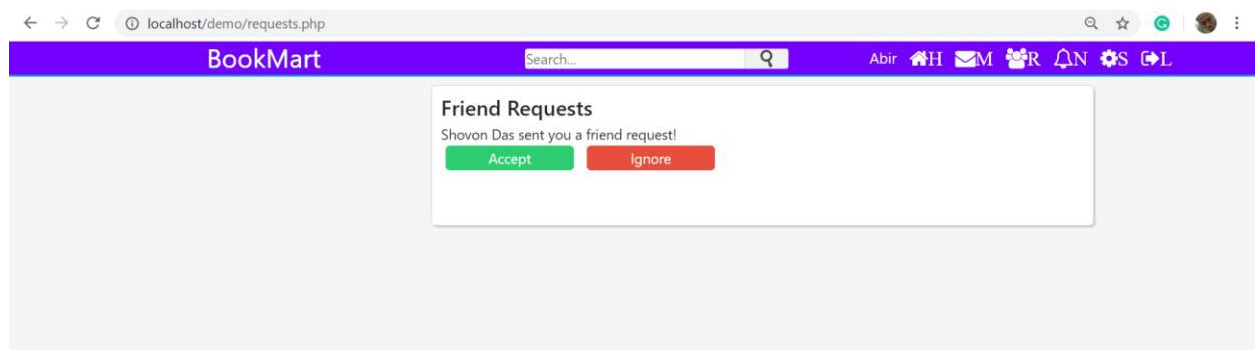


Figure 4.3.7: Friend Request page

This figure 4.3.8 shows the Post Advertisement page.

Post advertisements

No file chosen

Figure 4.3.8: Post Advertisement page

This figure 4.3.9 shows the Advertisement page.

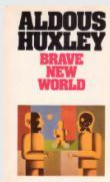
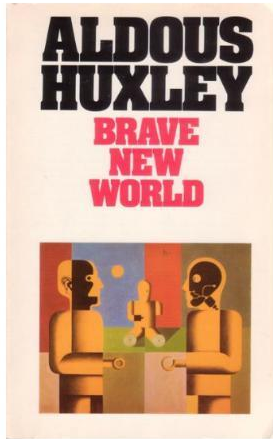
Book Name	Writer Name	Price	Description	Genre	Book Cover	Contact Seller	
Harry Potter	JK Rowling	500 taka	5 years old books	Fiction		Message Seller	X
Brave New World	Aldous Huxley	80 taka	Good Condition	Sci-fi		Message Seller	
Bhoot Shomora	Humayun Ahmad	300 taka	Old, second hand, good condition	Fiction		Message	

Figure 4.3.9: Advertisements page

This figure 4.3.10 shows the Ads Book Details

Brave New World



Genre: Sci-fi

Price: 80

Description and Condition: Good Condition

[Message Seller](#)

Figure 4.3.10: Book Details

This figure 4.3.11 shows how to Upload and Crop Profile picture

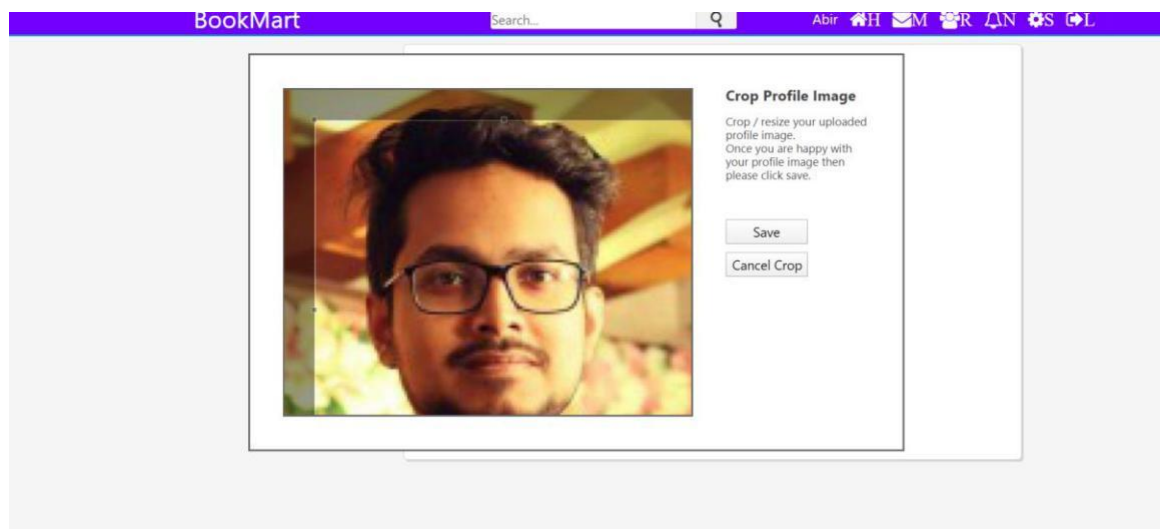


Figure 4.3.11: Profile Picture Upload page

This figure 4.3.12 shows the Admin login

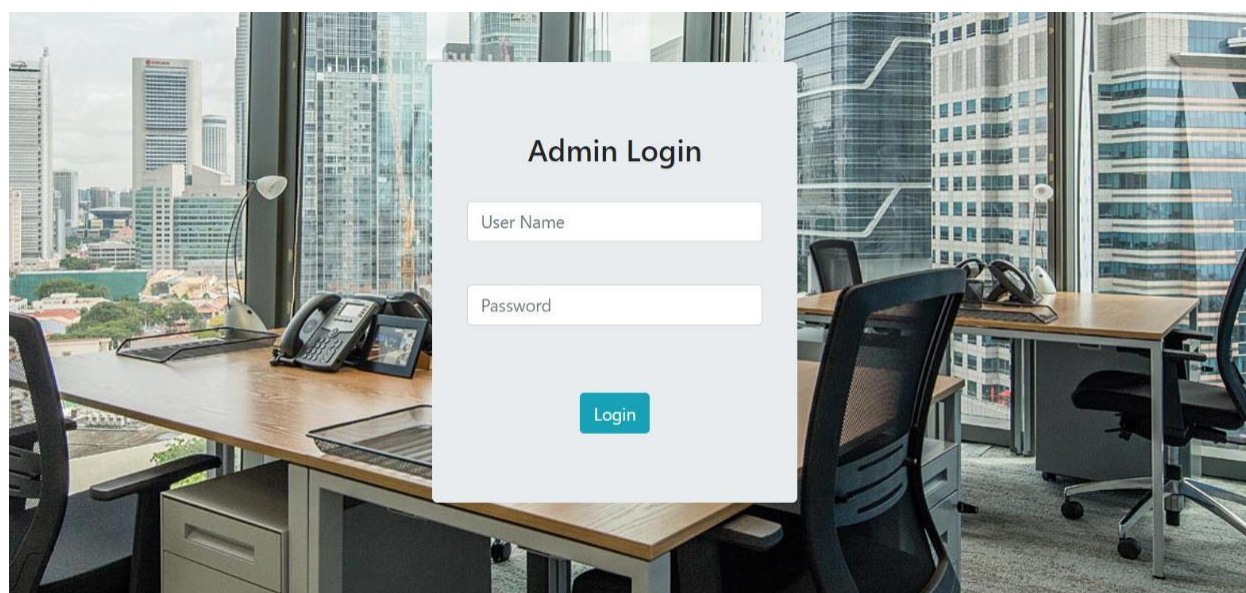


Figure 4.3.12: Admin Login page

This figure 4.3.13 shows the Admin User Management

User Management

Total Registered users: 5

Post management		Ads management		Logout	
S.No	First Name	Last Name	Email	Profile Image	Delete User
1	Abir	Hasan	Abir@gmail.com		Delete User
2	Hamayun	Kabir	Hamayun@gmail.com		Delete User
3	Israt	Nourin	Israt@gmail.com		Delete User

Figure 4.3.13: Admin User Management page

This figure 4.3.14 shows the Admin Post Management

Post Management

Total Number of Posts: 4

User Management		Ads management		Logout	
S.No	User Name	Post Body		Image	Delete Post
1	arafat_hossain	Hi			Delete Post
2	abir_hasan	I want to read this.			Delete Post
3	abir_hasan	Has anyone read this book?			Delete Post

Figure 4.3.14: Admin Post Management Page

This figure 4.3.15 shows the Admin Ads Management

Ads management

Total ads available: 3

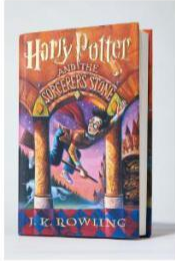
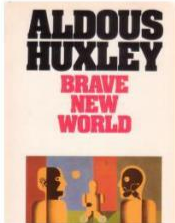
Post management		User management		Logout		
Book Name	Writer Name	Price	Description	Genre	Cover image	Delete Ad
Harry Potter	JK Rowling	500	5 years old books	Fiction		Delete Ad
Brave New World	Aldous Huxley	80	Good Condtion	Sci-fi		Delete Ad

Figure 4.3.15: Admin Ads Management Page

4.4 Backend Design

This figure 4.4.1 spectacles the backend database of the aforementioned system.

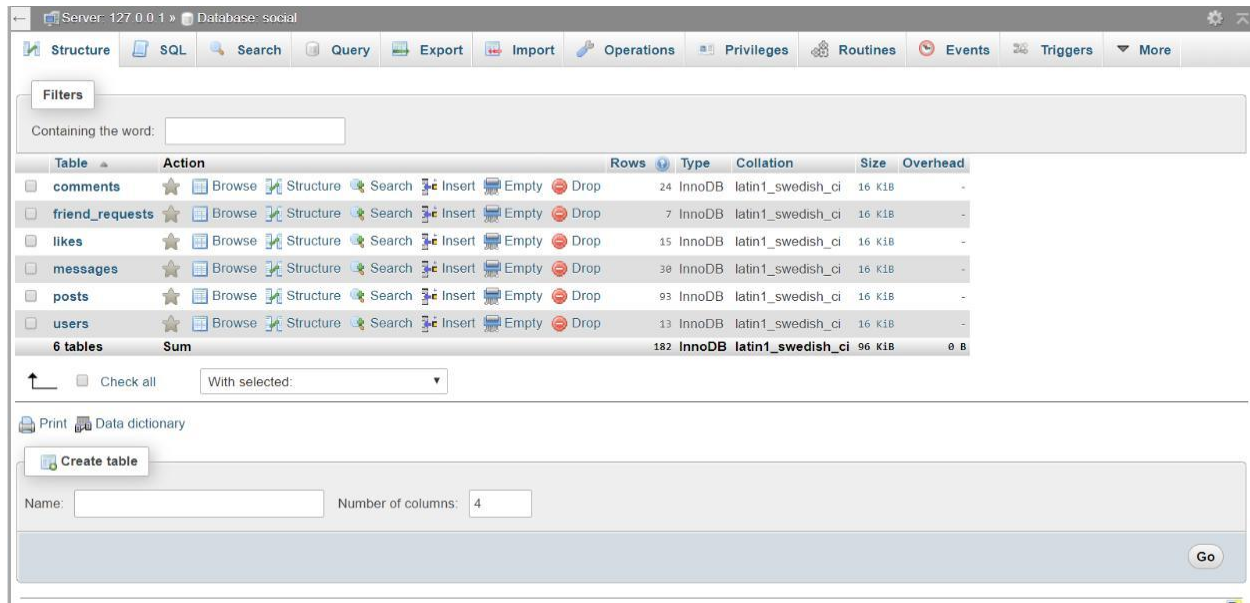


Figure 4.4.1: Back-end database

4.5 Implementation Requirements

The Implementation Requirements for the project:

- Text Editor: Sublime Text
- Xampp Localhost Server
- Bootstrap 4 [5]
- PHP7
- jQuery3.0 [6]
- Ajax
- CSS3 [7]
- HTML5 [8]
- JavaScript

4.6 Hardware Requirements

- Ram 1 GB or More
- HDD 100 GB or More

4.7 Software Requirements

- Internet Browser (Google Chrome, Internet Explorer, Mozilla Firefox)
- Windows Operating System

CHAPTER 5

IMPLEMENTATION AND TESTING

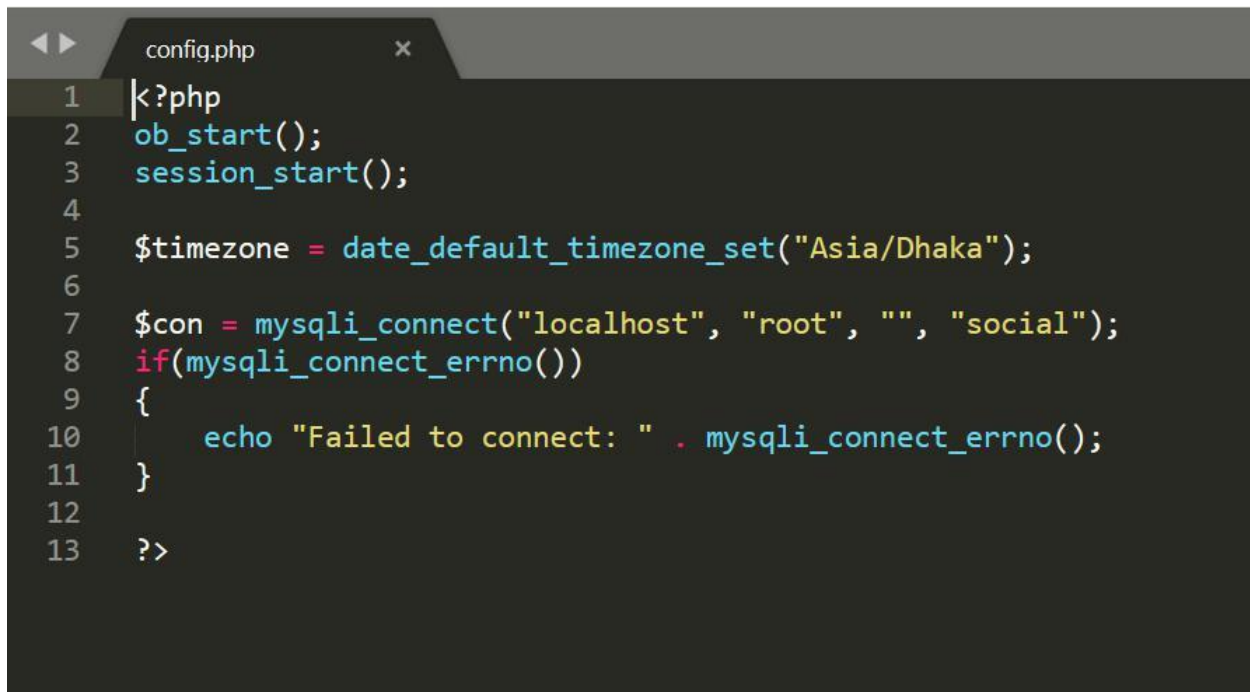
5.1 Introduction

In this chapter the implementation process of the project will be discussed. As this is a web based application, it has a Front-end and a Back-end component. Both of which will be explained with proper screenshots.

5.2 Implementation of Database

This part will show how the database was implemented. How the database was connected to the website will be covered.

This figure 5.2.1 shows the database connection process.

A screenshot of a code editor window titled 'config.php'. The code is written in PHP and shows the process of connecting to a MySQL database. It includes session management, setting the default timezone to 'Asia/Dhaka', and attempting to connect to a database named 'social' on 'localhost' with the username 'root' and an empty password. An error handling block is present to display a message if the connection fails.

```
1 <?php
2 ob_start();
3 session_start();
4
5 $timezone = date_default_timezone_set("Asia/Dhaka");
6
7 $con = mysqli_connect("localhost", "root", "", "social");
8 if(mysqli_connect_errno())
9 {
10     echo "Failed to connect: " . mysqli_connect_errno();
11 }
12
13 ?>
```

Figure 5.2.1: Database connection

We used a separate file named “config.php” to connect to the database. It is pretty straightforward as we have used pre-defined method for accessing the database name, username and password. We have created the database beforehand. The data would be inserted by the users. The code for this insertion is written in a separate file named “register_handler.php”.

Firstly, when any user registers in the website they insert their personal information through a PHP form. Figure 5.2.2 shows the code for User Registration in the system.

Register_handler.php-

```
1 <?php
2 //Declaring variables to prevent errors
3 $fname = ""; //First name
4 $lname = ""; //Last name
5 $em = ""; //email
6 $em2 = ""; //email 2
7 $password = ""; //password
8 $password2 = ""; //password 2
9 $date = ""; //Sign up date
10 $error_array = array(); //Holds error messages
11
12 if(isset($_POST['register_button'])){
13
14     //Registration form values
15
16     //First name
17     $fname = strip_tags($_POST['reg_fname']); //Remove html tags
18     $fname = str_replace(' ', '', $fname); //remove spaces
19     $fname = ucfirst(strtolower($fname)); //Uppercase first letter
20     $_SESSION['reg_fname'] = $fname; //Stores first name into session variable
21
22     //Last name
23     $lname = strip_tags($_POST['reg_lname']); //Remove html tags
24     $lname = str_replace(' ', '', $lname); //remove spaces
25     $lname = ucfirst(strtolower($lname)); //Uppercase first letter
26     $_SESSION['reg_lname'] = $lname; //Stores last name into session variable
27
28     //email
29     $em = strip_tags($_POST['reg_email']); //Remove html tags
30     $em = str_replace(' ', '', $em); //remove spaces
31     $em = ucfirst(strtolower($em)); //Uppercase first letter
32     $_SESSION['reg_email'] = $em; //Stores email into session variable
33
34     //email 2
35     $em2 = strip_tags($_POST['reg_email2']); //Remove html tags
36     $em2 = str_replace(' ', '', $em2); //remove spaces
37     $em2 = ucfirst(strtolower($em2)); //Uppercase first letter
38     $_SESSION['reg_email2'] = $em2; //Stores email2 into session variable
39
40     //Password
41     $password = strip_tags($_POST['reg_password']); //Remove html tags
42     $password2 = strip_tags($_POST['reg_password2']); //Remove html tags
43
44     $date = date("Y-m-d"); //Current date
45 }
```

Figure 5.2.2(a): Registration handler (page 1)

```
44 $date = date("Y-m-d"); //Current date
45
46 if($em == $em2) {
47     //Check if email is in valid format
48     if(filter_var($em, FILTER_VALIDATE_EMAIL)) {
49
50         $em = filter_var($em, FILTER_VALIDATE_EMAIL);
51
52         //Check if email already exists
53         $e_check = mysqli_query($con, "SELECT email FROM users WHERE email='$em'");
54
55         //Count the number of rows returned
56         $num_rows = mysqli_num_rows($e_check);
57
58         if($num_rows > 0) {
59             array_push($error_array, "Email already in use<br>");
60         }
61     }
62 }
63 else {
64     array_push($error_array, "Invalid email format<br>");
65 }
66
67
68 }
69 else {
70     array_push($error_array, "Emails don't match<br>");
71 }
72
73
74 if(strlen($fname) > 25 || strlen($fname) < 2) {
75     array_push($error_array, "Your first name must be between 2 and 25 characters<br>");
76 }
77
78 if(strlen($lname) > 25 || strlen($lname) < 2) {
79     array_push($error_array, "Your last name must be between 2 and 25 characters<br>");
80 }
81
82 if($password != $password2) {
83     array_push($error_array, "Your passwords do not match<br>");
84 }
85 else {
86     if(preg_match('/[A-Za-z0-9]/', $password)) {
87         array_push($error_array, "Your password can only contain english characters or numbers<br>");
88     }
89 }
```

Figure 5.2.2(b): Registration handler (page 2)

```

89     }
90
91     if(strlen($password > 30 || strlen($password) < 5)) {
92         array_push($error_array, "Your password must be between 5 and 30 characters<br>");
93     }
94
95
96     if(empty($error_array)) {
97         $password = md5($password); //Encrypt password before sending to database
98
99         //Generate username by concatenating first name and last name
100         $username = strtolower($fname . " " . $lname);
101         $check_username_query = mysqli_query($con, "SELECT username FROM users WHERE username='$username'");
102
103
104         $i = 0;
105         //if username exists add number to username
106         while(mysqli_num_rows($check_username_query) != 0) {
107             $i++; //Add 1 to i
108             $username = $username . " " . $i;
109             $check_username_query = mysqli_query($con, "SELECT username FROM users WHERE username='$username'");
110         }
111
112         //Profile picture assignment
113         $rand = rand(1, 2); //Random number between 1 and 2
114
115         if($rand == 1)
116             $profile_pic = "assets/images/profile_pics/defaults/head_deep_blue.png";
117         else if($rand == 2)
118             $profile_pic = "assets/images/profile_pics/defaults/head_emerald.png";
119
120
121         $query = mysqli_query($con, "INSERT INTO users VALUES ('', '$fname', '$lname', '$username', '$em', '$password', '$date', '$profile_pic',
122             '0', '0', 'no', ',', '0')");
123
124         array_push($error_array, "<span style='color: #14C800;'>You're all set! Go ahead and login!</span><br>");
125
126         //Clear session variables
127         $_SESSION['reg_fname'] = "";
128         $_SESSION['reg_lname'] = "";
129         $_SESSION['reg_email'] = "";
130         $_SESSION['reg_email2'] = "";
131     }
132 }

```

Figure 5.2.2(c): Registration handler (Page 3)

register.php-

```

54
55 <div id="second">
56
57     <form action="register.php" method="POST">
58         <input type="text" name="reg_fname" placeholder="First Name" value="<?php
59             if(isset($_SESSION['reg_fname'])) {
60                 echo $_SESSION['reg_fname'];
61             }
62             ?>" required>
63         <br>
64         <?php if(in_array("Your first name must be between 2 and 25 characters<br>", $error_array)) echo "Your first name must be
65             between 2 and 25 characters<br>"; ?>
66         <input type="text" name="reg_lname" placeholder="Last Name" value="<?php
67             if(isset($_SESSION['reg_lname'])) {
68                 echo $_SESSION['reg_lname'];
69             }
70             ?>" required>
71         <br>
72         <?php if(in_array("Your last name must be between 2 and 25 characters<br>", $error_array)) echo "Your last name must be
73             between 2 and 25 characters<br>"; ?>
74         <input type="email" name="reg_email" placeholder="Email" value="<?php
75             if(isset($_SESSION['reg_email'])) {
76                 echo $_SESSION['reg_email'];
77             }
78             ?>" required>
79         <br>
80         <input type="email" name="reg_email2" placeholder="Confirm Email" value="<?php
81             if(isset($_SESSION['reg_email2'])) {
82                 echo $_SESSION['reg_email2'];
83             }
84             ?>" required>
85         <br>
86         <?php if(in_array("Email already in use<br>", $error_array)) echo "Email already in use<br>";
87         <?php if(in_array("Invalid email format<br>", $error_array)) echo "Invalid email format<br>";
88         <?php if(in_array("Emails don't match<br>", $error_array)) echo "Emails don't match<br>"; ?>
89
90         <input type="password" name="reg_password" placeholder="Password" required>
91         <br>
92         <input type="password" name="reg_password2" placeholder="Confirm Password" required>
93         <br>
94         <?php if(in_array("Your passwords do not match<br>", $error_array)) echo "Your passwords do not match<br>";
95         <?php if(in_array("Your password can only contain english characters or numbers<br>", $error_array)) echo "Your password can
96             only contain english characters or numbers<br>";
97         <?php if(in_array("Your password must be between 5 and 30 characters<br>", $error_array)) echo "Your password must be between 5
98             and 30 characters<br>"; ?>
99     </form>
100
101     <div id="error">
102         <?php if(!empty($error_array)) {
103             foreach($error_array as $error) {
104                 echo "<div class='error'><div class='close'><span></span></div><div class='text'>$error</div></div>";
105             }
106         }
107     </div>
108
109     <div id="success">
110         <?php if(!empty($success_array)) {
111             foreach($success_array as $success) {
112                 echo "<div class='success'><div class='close'><span></span></div><div class='text'>$success</div></div>";
113             }
114         }
115     </div>
116
117     </div>

```

Figure 5.2.3: Registration System

This figure 5.2.3 shows the Login system of the website.

Login_handler.php-

```
1 <?php
2
3 if(isset($_POST['login_button'])) {
4
5     $email = filter_var($_POST['log_email'], FILTER_SANITIZE_EMAIL);
6
7     $_SESSION['log_email'] = $email;
8     $password = md5($_POST['log_password']);
9
10    $check_database_query = mysqli_query($con, "SELECT * FROM users WHERE email='$email' AND password='$password'");
11    $check_login_query = mysqli_num_rows($check_database_query);
12
13    if($check_login_query == 1) {
14        $row = mysqli_fetch_array($check_database_query);
15        $username = $row['username'];
16
17        $user_closed_query = mysqli_query($con, "SELECT * FROM users WHERE email='$email' AND user_closed='yes'");
18        if(mysqli_num_rows($user_closed_query) == 1) {
19            $reopen_account = mysqli_query($con, "UPDATE users SET user_closed='no' WHERE email='$email'");
20        }
21
22        $_SESSION['username'] = $username;
23        header("Location: index.php");
24        exit();
25    }
26    else {
27        array_push($error_array, "Email or password was incorrect<br>");
28    }
29
30 }
31
32
33 ?>
```

Figure 5.2.4: Login handler

Login form-

```
38 <div class="login_box">
39
40 <div class="login_header">
41 <h1>BookMart!</h1>
42 Login or sign up below!
43 </div>
44 <br>
45 <div id="first">
46
47 <form action="register.php" method="POST">
48 <input type="email" name="log_email" placeholder="Email Address" value="<?php
49 if(isset($_SESSION['log_email'])) {
50 | echo $_SESSION['log_email'];
51 | }>" required>
52 <br>
53 <input type="password" name="log_password" placeholder="Password">
54 <br>
55 <?php if(in_array("Email or password was incorrect<br>", $error_array)) echo "Email or password was incorrect<br>"; ?>
56 <input type="submit" name="login_button" value="Login">
57 <br>
58 <a href="#" id="signup" class="signup">Need an account? Register here!</a>
59
60 </form>
61
62 </div>
63
64
```

Figure 5.2.5: Login System

After that we have made the table for posts. This figure 5.2.4 shows the query to insert posts.

```
Post.php
1 <?php
2 class Post {
3     private $user_obj;
4     private $con;
5
6     public function __construct($con, $user){
7         $this->con = $con;
8         $this->user_obj = new User($con, $user);
9     }
10
11     public function submitPost($body, $user_to, $imageName) {
12         $date_added = date("Y-m-d H:i:s");
13         $query = mysqli_query($this->con, "INSERT INTO posts VALUES('', 'sda', 'asdfsdf', 'asdfsdf', '$date_added', 'no', 'no', '0', '$imageName')");
14         $body = strip_tags($body); //removes html tags
15         $body = mysqli_real_escape_string($this->con, $body);
16         $check_empty = preg_replace('/\s+/', '', $body); //Deletes all spaces
17
18         if($check_empty != "") {
19
20             //Current date and time
21             $date_added = date("Y-m-d H:i:s");
22             //Get username
23             $added_by = $this->user_obj->getUsername();
24
25             //If user is on own profile, user_to is 'none'
26             if($user_to == $added_by) {
27                 $user_to = "none";
28             }
29
30             //insert post
31             $query = mysqli_query($this->con, "INSERT INTO posts VALUES('', '$body', '$added_by', '$user_to', '$date_added', 'no', 'no', '0', '$imageName')");
32             $returned_id = mysqli_insert_id($this->con);
33
34 }
```

Figure 5.2.6: Query for insert posts

This figure 5.2.5 shows the query for Display post.

```
Post.php
43 public function loadPostsFriends($data, $limit) {
44
45     $page = $data['page'];
46     $userLoggedIn = $this->user_obj->getUsername();
47
48     if($page == 1)
49         $start = 0;
50     else
51         $start = ($page - 1) * $limit;
52
53
54     $str = ""; //String to return
55     $data_query = mysqli_query($this->con, "SELECT * FROM posts WHERE deleted='no' ORDER BY id DESC");
56
57     if(mysqli_num_rows($data_query) > 0) {
58
59
60         $num_iterations = 0; //Number of results checked (not necessarily posted)
61         $count = 1;
62
63         while($row = mysqli_fetch_array($data_query)) {
64             $id = $row['id'];
65             $body = $row['body'];
66             $added_by = $row['added_by'];
67             $date_time = $row['date_added'];
68             $imagePath = $row['image'];
69
70             //Prepare user to string so it can be included even if not posted to a user
71             if($row['user_to'] == "none") {
72                 $user_to = "";
73             }
74             else {
75                 $user_to_obj = new User($this->con, $row['user_to']);
76                 $user_to_name = $user_to_obj->getFirstAndLastName();
77                 $user_to = "posted on <a href='" . $row['user_to'] . "'>" . $user_to_name . "</a>'s profile";
78             }
79
80             //Check if user who posted, has their account closed
81             $added_by_obj = new User($this->con, $added_by);
82             if($added_by_obj->isClosed()) {
83                 continue;
84             }
85         }
86     }
87 }
```

Figure 5.2.7: Query for display post

This figure 5.2.6 shows the query for likes and unlikes in the posts.

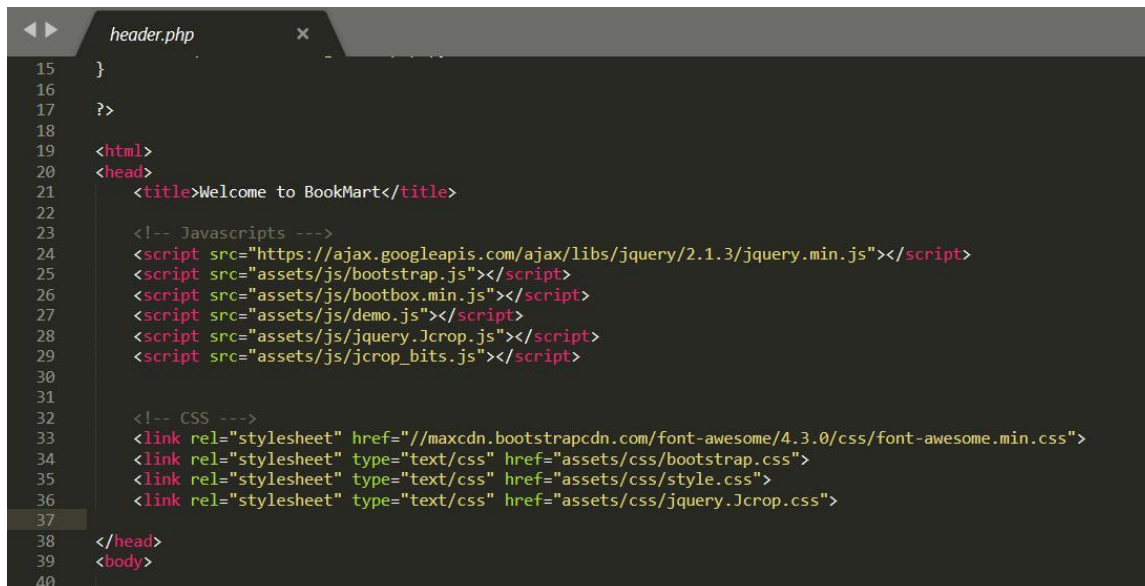
```
like.php
25 include('includes/classes/User.php');
26 include('includes/classes/Post.php');
27
28 if (isset($_SESSION['username'])) {
29     $userLoggedIn = $_SESSION['username'];
30     $user_details_query = mysqli_query($con, "SELECT * FROM users WHERE username='$userLoggedIn'");
31     $user = mysqli_fetch_array($user_details_query);
32 }
33 else {
34     header("Location: register.php");
35 }
36 if(isset($_GET['post_id'])) {
37     $post_id = $_GET['post_id'];
38 }
39 $get_likes = mysqli_query($con, "SELECT likes, added_by FROM posts WHERE id='$post_id'");
40 $row = mysqli_fetch_array($get_likes);
41 $total_likes = $row['likes'];
42 $user_liked = $row['added_by'];
43
44 $user_details_query = mysqli_query($con, "SELECT * FROM users WHERE username='$user_liked'");
45 $row = mysqli_fetch_array($user_details_query);
46 $total_user_likes = $row['num_likes'];
47 if(isset($_POST['like_button'])) {
48     $total_likes++;
49     $query = mysqli_query($con, "UPDATE posts SET likes='$total_likes' WHERE id='$post_id'");
50     $total_user_likes++;
51     $user_likes = mysqli_query($con, "UPDATE users SET num_likes='$total_user_likes' WHERE username='$user_liked'");
52     $insert_user = mysqli_query($con, "INSERT INTO likes VALUES('', '$userLoggedIn', '$post_id')");
53 }
54 if(isset($_POST['unlike_button'])) {
55     $total_likes--;
56     $query = mysqli_query($con, "UPDATE posts SET likes='$total_likes' WHERE id='$post_id'");
57     $total_user_likes--;
58     $user_likes = mysqli_query($con, "UPDATE users SET num_likes='$total_user_likes' WHERE username='$user_liked'");
59     $insert_user = mysqli_query($con, "DELETE FROM likes WHERE username='$userLoggedIn' AND post_id='$post_id'");
60 }
61 $check_query = mysqli_query($con, "SELECT * FROM likes WHERE username='$userLoggedIn' AND post_id='$post_id'");
62 $num_rows = mysqli_num_rows($check_query);
63
```

Figure 5.2.8: Query for Likes and Unlikes

5.3 Implementation of Front-end Design

Our utility is internet based totally. So that there are numerous workings with the front quit layout. Reason the customers are seeing the front quit and make their interplay with the utility. So for our internet based challenges we've got used HTML[8], CSS[7], bootstrap[5], jQuery[6] for the front quit implementation.

Incipiently let's take a stare how we boot our bootstrap and our initial of the web page. So, here the figure 5.3.1 shows the initialization of front end and the linkage to bootstrap.



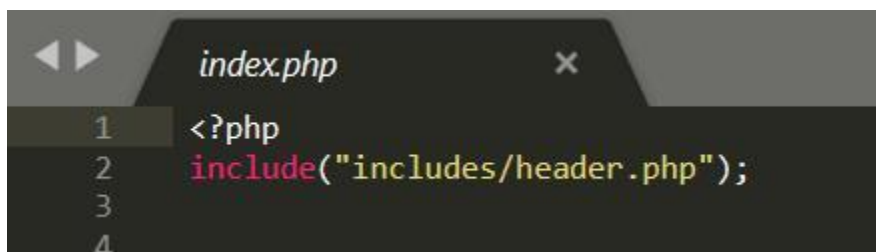
```

15 }
16
17 ?>
18
19 <html>
20 <head>
21 <title>Welcome to BookMart</title>
22
23 <!-- Javascripts --->
24 <script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>
25 <script src="assets/js/bootstrap.js"></script>
26 <script src="assets/js/bootbox.min.js"></script>
27 <script src="assets/js/demo.js"></script>
28 <script src="assets/js/jquery.Jcrop.js"></script>
29 <script src="assets/js/jcrop_bits.js"></script>
30
31
32 <!-- CSS --->
33 <link rel="stylesheet" href="//maxcdn.bootstrapcdn.com/font-awesome/4.3.0/css/font-awesome.min.css">
34 <link rel="stylesheet" type="text/css" href="assets/css/bootstrap.css">
35 <link rel="stylesheet" type="text/css" href="assets/css/style.css">
36 <link rel="stylesheet" type="text/css" href="assets/css/jquery.Jcrop.css">
37
38 </head>
39 <body>
40

```

Figure 5.3.1: Initialization of CSS, Bootstrap and Javascript libraies

Then we designed the header portion of the front page. We included the header files there. This figure 5.3.2 shows the code for include of header.



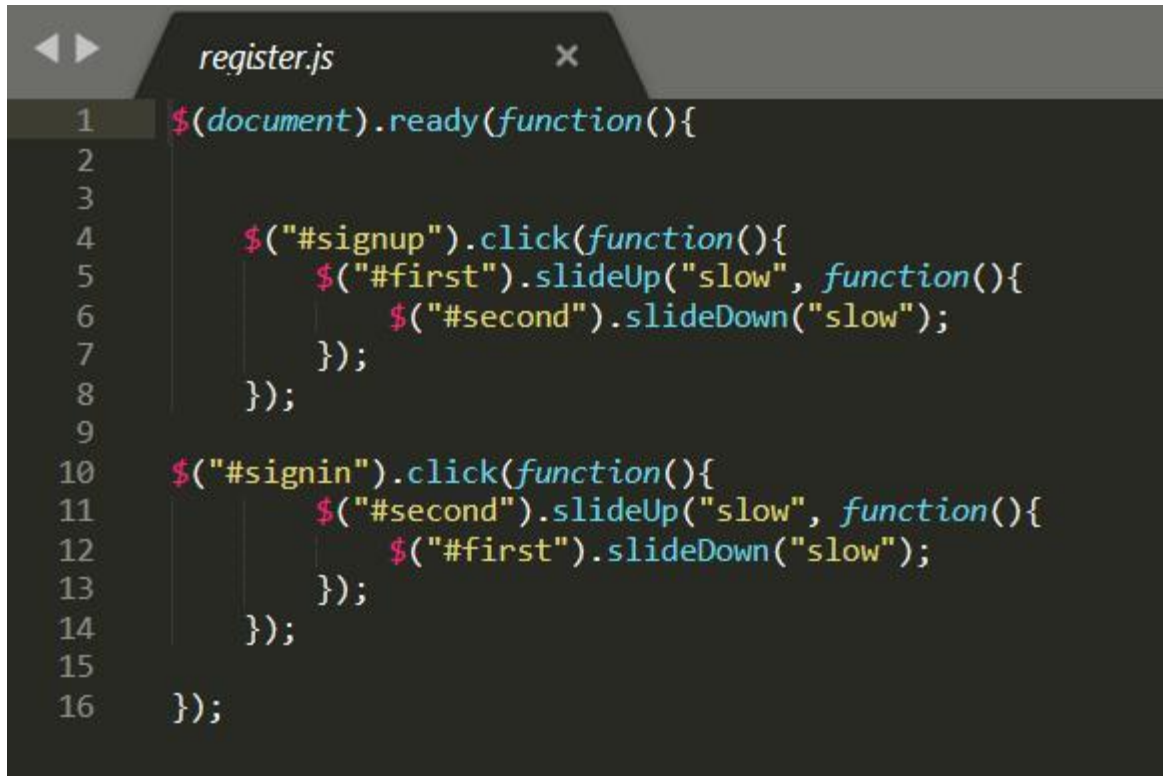
```

1 <?php
2 include("includes/header.php");
3
4

```

Figure 5.3.2: Include process of header file

This figure 5.3.3 shows the Register and Login toggle mechanism.



```
1 $(document).ready(function(){
2
3
4     $("#signup").click(function(){
5         $("#first").slideUp("slow", function(){
6             $("#second").slideDown("slow");
7         });
8     });
9
10    $("#signin").click(function(){
11        $("#second").slideUp("slow", function(){
12            $("#first").slideDown("slow");
13        });
14    });
15
16 });
```

Figure 5.3.3: Code for Register and Login toggle

Then we added the side menu. After the slider we displayed some of our featured products after that we included the footer file.

5.4 Testing Implementation

As this is a web based application the testing is very important before deploying the application. The testing will measure the overall performance and give the knowledge how to make necessary improvements to the project. The tests can also tell us out the time required to use the application and ways to reduce time wastage. There are many tests that can be conducted on this application. Here we have done some tests and the results are shown below:

5.4.1 Unit Testing

Unit testing is used when the coding is being implemented then the developers check for their own initial mistakes. In this case we have checked when we were coding and tried to find out the initial errors.

5.4.2 Integration Testing

Front-end and Back-end part was built simultaneously for this project. The integration of the two parts was a difficult task as the two parts showed errors from time to time, which were fixed we moved ahead. We ran another Integration test nevertheless.

5.4.3 Functional Testing

We have finished most of the functionalities that we had selected for this application. All of these functions have been tested separately.

5.4.4 System Testing

As the application is a website it has to work on different computers, different operating systems and browsers. We have tested the application on various browsers such as, Chrome, Mozilla, Internet Explorer, operating systems such as Windows and Ubuntu to see if the application works properly in other environments.

5.4.5 Usability Testing

The User Interface and the User Experience were designed keeping in mind the design trends of the modern social media applications that exists today such as, Facebook. As a result, the application is very user friendly. We have tested the website on a few people who have moderate knowledge of using internet.

5.5 Test Results

After doing the tests mentioned above we got some results that were satisfactory.

Results are given below:

5.5.1 Result of Unit Testing

The whole project was built with raw PHP coding. There are more than 30 files which includes thousands lines of code that was written to build this website. So it is not very uncommon to make mistakes when there is so much complexity going on. To find out the development mistakes we

did Unit Testing. The test results came out positive as we fixed most of the problems beforehand. We later fixed the few problems that we found later.

5.5.2 Result of Integration Testing

Integration test were being done alongside the development process of the application. Very few issues were found later on and they were fixed immediately.

5.5.3 Result of Functional Testing

Function testing was too much important. Because we did good front-end and back-end design but we did mistake in functional coding. So, what is the value of total coding. It is useless. Functional testing was the way to know that is every function working properly or not. One by one we checked every function. During this testing we got some bad behavior for logging in. And also got some bad behaviors in setting profile page. There was some logical error. And then we fixed them. Another front end site management there was a cache problem on the browser. This problem was overcome also by removing older files and inputting new one. Thus, we have completed our functional testing.

5.5.4 Result of System Testing

After testing the website in different environments we found the results satisfactory. We attempted to fix the responsiveness issues that occurred at first.

5.5.5 Result of Usability Testing

After doing some testing with different types of people on the website, the results were positive. They could use the website at ease. No usability problems were found.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Discussion and conclusion

We have tried our best in the making of this project. We also made this report to reflect our effort in this project. We have learned a lot and developed a lot of skills and most importantly we have learned to think unconventionally.

While working on this project we have confronted many kinds of problems which required us to learn new things think unconventionally. At the start we thought we will use only html5 and CSS after that we started using bootstrap framework, we were unfamiliar with this. We learned bootstrap after that we used PHP, which was another unknown thing to us, but we overcame our problems slowly but surely. This project was like a journey and during this journey we gained some precious knowledge about PHP, Ajax, jQuery, HTML, CSS and Bootstrap.

Through this entire project we have learnt a lot about programming which will be very helpful for our future. Designing the database was very important so while making the database we got to know about a lot of things.

6.2 Limitations of the project

This project has many incomplete parts because the system is very complex, as all social networking systems are. It was really difficult to us to do as much as we did in such a short time. The system is built mostly using raw PHP, so it lacks many modern features. The website is at a very basic level. In the next section (6.3), we discussed all the features we would like to develop in the future.

6.3 Scope for Further Development

In future we want to add lots of new features in our system. For example, phone verification, location tracking, AI to find taste of reading of the users and suggest books accordingly, payment system integration, rating system. Also, we can integrate a third-party delivery service like Pathao (pathao.com/), so that people can easily exchange or sell books. This project requires both web and mobile versions for user convenience.

All these developments require huge amount of time and resources which are not possible without investments and more skills. Although this can be considered a prototype, this project

has the potential to be a great real life project that can convert the book market of Bangladesh today.

REFERENCES

- [1] Bikroy.com available at: <https://bikroy.com/bn> [last accessed on 11-05-2020 at 12.00am]
- [2] Facebook available at: <https://www.facebook.com/> [last accessed on 11-05-2020 at 12.15am]
- [3] eBay available at: <https://www.ebay.com/> [last accessed on 11-05-2020 at 12.20am]
- [4] Xampp server available at: <https://www.apachefriends.org/index.html> [last accessed on 11-05-2020 at 12.30am]
- [5] Bootstrap available at: <https://getbootstrap.com/> [last accessed on 11-05-2020 at 12.40am]
- [6] jQuery available at: <https://jquery.com/> [last accessed on 11-05-2020 at 1.00am]
- [7] CSS available at: <https://www.w3schools.com/css/> [last accessed on 11-05-2020 at 1.00am]
- [8] HTML available at: <https://www.w3schools.com/html/> [last accessed on 11-05-2020 at 1.00am]

Plagiarism Report

Plagiarism Report by Daffodil International University:

ORIGINALITY REPORT

23%	17%	1%	23%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	10%
2	www.ukessays.com Internet Source	5%
3	Submitted to St. Petersburg High School Student Paper	2%
4	Submitted to University of Greenwich Student Paper	1%
5	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
6	Submitted to Segi University College Student Paper	<1%
7	Submitted to University of Huddersfield Student Paper	<1%
8	Submitted to London School of Commerce Student Paper	<1%
9	Submitted to HELP UNIVERSITY Student Paper	<1%
10	dspace.library.daffodilvarsity.edu.bd:8080 Internet Source	<1%
11	Submitted to Bolton Institute of Higher Education Student Paper	<1%

