Project Report On

**BOOKNERDS**

BACHELOR OF TECHNOLOGY

**Semester-V**

**COMPUTER ENGINEERING**

**Submitted By Guided By**

Patel Trusha

Enroll-No:-18012011086 Prof. Om Parkash Pal

Patel Yukta

Enroll-No:-18012011090

**Submitted to,**

Computer Engineering Department

U.V. Patel College of Engineering, Ganpat University



Nov-Dec 2020

INDEX

[LIST OF FIGURES 3](#_Toc58831860)

[LIST OF TABELS 3](#_Toc58831861)

[CHAPTER-1 INTRODUCTION 4](#_Toc58831862)

[1.1 Purpose: 4](#_Toc58831863)

[1.2 Scope: 4](#_Toc58831864)

[1.3 Literature Survey: 4](#_Toc58831865)

[1.4 Overview: 5](#_Toc58831866)

[CHAPTER-2 SYSTEM REQUIREMENT STUDY 7](#_Toc58831867)

[2.1 Function Requirement: 7](#_Toc58831868)

[2.2 Non-Functional Requirement: 13](#_Toc58831869)

[2.3 User characteristics: 14](#_Toc58831870)

[2.4 Operating Environment: 14](#_Toc58831871)

[CHAPTER-3 INTER REQURIMENT 15](#_Toc58831872)

[3.1 System Interface 15](#_Toc58831873)

[3.2 User Interface 15](#_Toc58831874)

[3.3 Hardware Interface 15](#_Toc58831875)

[3.4 Software Interface 15](#_Toc58831876)

[3.5 Communication Interface 16](#_Toc58831877)

[CHAPTER-4 SYSTEM DESIGN 17](#_Toc58831878)

[4.1 Use-case Diagram 17](#_Toc58831879)

[4.2 Class Diagram 18](#_Toc58831880)

[4.3 Sequence Diagrams 19](#_Toc58831881)

[4.4 Activity Diagram 22](#_Toc58831882)

[4.5 Data Flow Diagram 23](#_Toc58831883)

[CHAPTER-5 DATABASE DICTIONARY 25](#_Toc58831884)

[5.1 Database Dictionary 25](#_Toc58831885)

[CHAPTER-6 IMPLEMENTATION PLANNING AND DETAILS 28](#_Toc58831886)

[CHAPTER-7 CONCLUSION 30](#_Toc58831887)

[CHAPTER 8 REFERENCES 31](#_Toc58831888)

# LIST OF FIGURES

[Figure 1 Use case diagram 18](#_Toc58437788)

[Figure 2 Class diagram 19](#_Toc58437789)

[Figure 3 Sequence Diagram User Login 20](#_Toc58437790)

[Figure 4 Sequence diagram Admin Login 20](#_Toc58437791)

[Figure 5 Sequence diagram Book Search 21](#_Toc58437792)

[Figure 6 Sequence diagram Shopping cart 21](#_Toc58437793)

[Figure 7 Activity Diagram 22](#_Toc58437794)

[Figure 8 Level-0 23](#_Toc58437795)

[Figure 9 Level-1 23](#_Toc58437796)

[Figure 10 Level-2 24](#_Toc58437797)

[Figure 11 Project planning and implementation 28](#_Toc58437798)

# LIST OF TABELS

[Table 1 Software Interface 15](#_Toc58438560)

[Table 2 User 25](#_Toc58438561)

[Table 3 Customer Details 25](#_Toc58438562)

[Table 4 Admin 25](#_Toc58438563)

[Table 5 Book 26](#_Toc58438564)

[Table 6 Book Order 26](#_Toc58438565)

[Table 7 Shopping cart 26](#_Toc58438566)

[Table 8 Exchange Book 27](#_Toc58438567)

[Table 9 project planning and implementation details 28](#_Toc58438568)

# CHAPTER-1 INTRODUCTION

##### 1.1 Purpose:

**Online book buy, sell and exchange website.** This system will help the users for reusability of their books , buying and selling their books and also helps the users to know the suggestion of their intrests being introduced into the market like fiction, science, technology, education, etc.

##### 1.2 Scope:

* There are three basic users –Anonymous, Registered Members(customers), Administrator.
* The customer, through the process of account creation, will have the option to become a member of this site.
* The system will allow customers to browse, search, select, buy/sell, exchange and add books to a shopping cart.
* Admin has the authority to add/delete users, grant permissions to members and user to purchase/exchange their books.

##### 1.3 Literature Survey:

* We are going to take the website named (swapthebook.com) as our idealistic guideline website. It is developed by slidescope. The purpose of these website/application is to allow person to make virtual library of their own choice on their platform. The motive is that one person A can create the virtual library if he/she wants to sell/borrow/exchange their books with others, the next person B will be also creating the virtual library as per his/her need for sell/borrow/exchange the books. these web/app will be helping them to swap/exchange their books as no medium will be affected. The exchanging will be done by the slidescope delivery person itself.
* One another website is BookChor.com developed by bookchor team. They provides second hand books at lower price, so users can buy very expensive books. User can sell their old books on this website. Like a person1 can want sell their book he/she fill one page for that then person2 want to buy that book he/she can contact person1 directly if both are agreed on that person2 get that book.
* We take another idea from BookChums.com it is developed by Milestone Machine. This website gives suggestions to user which books they want to read next. And also gives short description of book for users. They listed book by categories.
* This all are facilities on different sites, so we want to make one unique website can cover all these things:

-buy new books

-buy old books at lower price

-sell books

-suggestions for next book

-sell their old books

-make categories for users

##### 1.4 Overview:

1. **Existing system:**

* Manually registration

1. **Drawbacks:**

* No webcam support
* Not all users have access to website
* People who doesn’t have a knowledge of computer and internet cannot use the system

1. **Proposed system:**

* Registration for users(customers)
* Buy/sell/exchange of books with email intimation
* Notification through mobile message when new books arrive

1. **Our plan:**

* Registration for members
* Online maintenance of books and sales
* Acknowledgement for registration, sales and arrival of new books through email
* Feedback from users

# CHAPTER-2 SYSTEM REQUIREMENT STUDY

##### 2.1 Function Requirement:

* **Functional Requirement 1:**
* Description:

Ask user to enter details in order to become a registered user.

* Input:

Name in characters, email and password in alphanumeric values

* Processing:

Storing the values entered in database and check if a person with same email is available or not. If not then store in database else an error is generated.

* Output:

Confirmation message registered successfully.

* **Functional Requirement 2:**
* Description:

Initialize username and password for login.

* Input:

Username and password in alphanumeric values.

* Processing:

Storing the values and checking whether true or not.

* Output:

Confirmation message login successful else login failed.

* **Functional Requirement 3:**
* Description:

Form validation.

* Input:

Name in characters, email and password in alphanumeric values.

* Processing:

Storing values entered and checking whether they are valid or invalid.

* Output:

Confirmation message valid.

* **Functional Requirement 4:**
* Description:

Profile update to update the current details to new one.

* Input:

Name in characters, email and password in alphanumeric value, mobile number, address, date of birth.

* Processing:

Storing the values entered by the user in the database.

* Output:

Confirmation message profile updated successfully.

* **Functional Requirement 5:**
* Description:

Search for a particular book.

* Input:

Name of the book in characters.

* Processing:

Checking whether the entered book is available or not.

* Output:

Displays book details if available else an error is generated.

* **Functional Requirement 6:**
* Description:

Sell a book.

* Input:

Name of book, author name in characters and price in number format.

* Processing:

Store the information of book in database.

* Output:

Book added for sale.

* **Functional Requirement 7:**
* Description:

Buy a book.

* Input:

Name of the book, author name.

* Processing:

Fetch the data of book if available.

* Output:

If a book is available, display message “added to cart” else display book is unavailable.

* **Functional Requirement 8:**
* Description:

Exchange a book.

* Input:

Name of book, author name, condition of book, time taken by customer.

* Processing:

Exchanging the book by book for validation of time entered by users.

* Output:

If the exchanged book is available then display message “book is available” else unavailable.

* **Functional Requirement 9:**
* Description:

Proceed to checkout

* Input:

Enter the delivery address and click on checkout.

* Processing:

Stores the delivery address and buyers details to database.

* Output:

Redirects to payment gateway.

* **Functional Requirement 10:**
* Description:

Payment gateway

* Input:

Card details including card number, CVV, expiry date

* Processing:

Check the information provided to bank server. If successful, money gets debited from user bank account and credited to book seller account, else display error.

* Output:

If no error, display message order successful and order id generated else display error message.

* **Functional Requirement 11:**
* Description:

Track order

* Input:

Order id.

* Processing:

Fetch the tracking details from database.

* Output:

Displays the current location of the book and when it is going to be

Delivered.

* **Functional Requirement 12:**
* Description:

View all orders.

* Input:

Click on my orders

* Processing:

Fetch the all past order placed by user from database

* Output:

Display the list of all books delivered with delivery date.

* **Functional Requirement 13:**
* Description:

Rate a seller

* Input:

Comments and star rating for seller

* Processing:

Stores the information entered by the users,

* Output:

Display message “thanks for rating”.

* **Functional Requirement 14:**
* Description:

Rate a book.

* Input:

Comments and star rating for book.

* Processing:

Stores the information entered by the user to book details.

* Output:

Display message “thanks for rating”.

* **Functional Requirement 15:**
* Description:

Top selling books.

* Input:

Click on top selling books.

* Processing:

Fetches the data of books from the database in descending order of their rating.

* Output:

Displays the list of books.

* **Functional Requirement 16:**
* Description:

Notify for out of stock books.

* Input:

Enter email address.

* Processing:

Stores the email address in database for further processing.

* Output:

Displays message “thanks! you will be informed as soon as we get the new stock of the book.

* **Functional Requirement 16:**
* Description:

Contact/report

* Input:

Enter the user details and query/problem faced.

* Processing:

Stores the details in database for further processing.

* Output:

Displays message “thanks! We will get back to you as soon as possible”.

##### 2.2 Non-Functional Requirement:

* **Safety Requirements:**
* Highly recommended Quick Heal 2017 internet security to been installed in to users Pc to prevent the harm that may occurs by unwanted malicious software’s, phishing URLs and all the types of viruses attacks during using these website.
* **Security requirements:**
* All users should be properly authenticated before allowed entry into the system authentication will be based on an E-mail address and a password. All activities must be logged.
* **Non-Functional Requirements:**
* Secure access of confidential data(user’s details).SSL can be used.
* 24x7 availability.
* Better component design to get better performance at the peak time.
* Flexible service based architecture will be highly desirable for future extensions.

##### 2.3 User characteristics:

* **Guest/anonymous:** non members can visit the pages of website only.
* **Registered members(customers):** Registered members can access all the functionalities provides on website. Registered members have their own profiles.
* **Administrator/developer:** Admin has authority to grant/delete registered members as well as view profiles. The Administrator will be provided with special functionalities like:
* Add or delete a book category
* Add or delete a member
* Manage member orders.

##### 2.4 Operating Environment:

* The online books buy, sell and exchange website will be developed in any c#/java preferable platform.
* C#/Java will be used as programming language.

# CHAPTER-3 INTER REQURIMENT

##### 3.1 System Interface

* Online Payment system: The system will access the online payment system via its web services API.
* Credit card system: The system will access the credit card processing system via its web services API.
* Bookstore Inventory database The system will interact with the inventory database via

##### 3.2 User Interface

* Front-end software: C#/Java
* Back-end software: SQL server/Postgre SQL
* Client side software: .Net framework/JSP

##### 3.3 Hardware Interface

* There is no particular hardware device to access this website.
* Any hardware is compatible.

##### 3.4 Software Interface

Table 1 Software Interface

|  |  |
| --- | --- |
| **Software used** | **Description** |
| Operating system | We have chosen windows operating system for its best support and user friendliness. |
| Database | To save the book details, user details, sells details, profit details, return orders, book return details we have chosen SQL server. |
| Python | To implement the project we have chosen C#/Java language for its more interactive support. |

##### 3.5 Communication Interface

* This website support all type of web browsers.
* The website order system shall send an e-mail confirmation to the customer that books they order.
* The website order system shall send an e-mail confirmation to the customer that books they return.
* This website also confirm payment process by e-mail.
* This website provides e-form to user for give feed

# CHAPTER-4 SYSTEM DESIGN

##### 4.1 Use-case Diagram

****

Figure 1 Use case diagram

##### 4.2 Class Diagram



Figure 2 Class diagram

##### 4.3 Sequence Diagrams

**User login**

****

Figure 3 Sequence Diagram User Login

**Admin Login**



Figure 4 Sequence diagram Admin Login

**Book Search**

****

Figure 5 Sequence diagram Book Search

**Add to shopping cart**

****

Figure 6 Sequence diagram Shopping cart

##### 4.4 Activity Diagram

****

Figure 7 Activity Diagram

##### 4.5 Data Flow Diagram

**Level-0 Diagram**

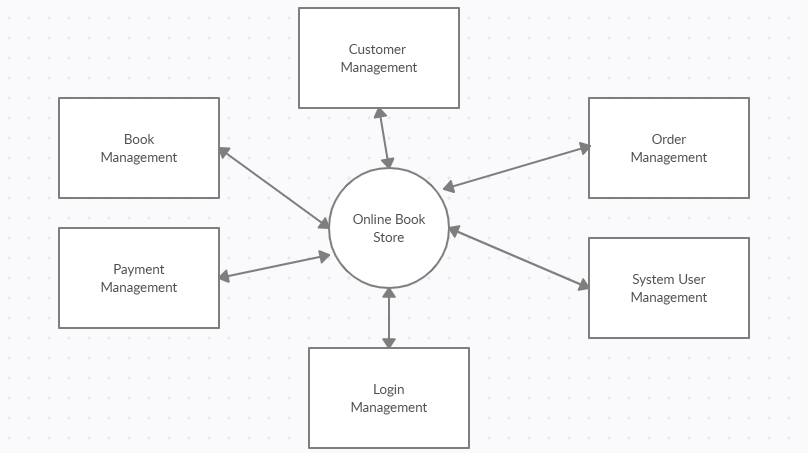
****

Figure 8 Level-0

**Level-1 Diagram**

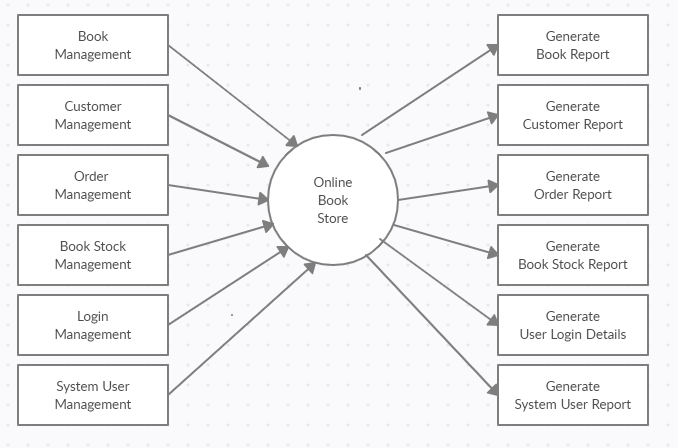


Figure 9 Level-1

**Level-2 Diagram**

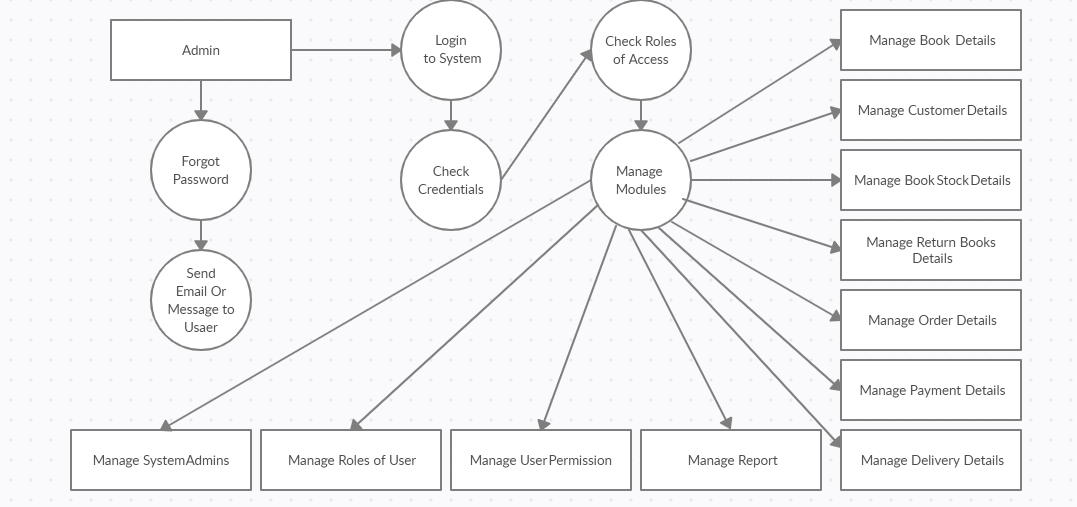
****

Figure 10 Level-2

# CHAPTER-5 DATABASE DICTIONARY

##### 5.1 Database Dictionary

Table 2 User

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Field name** | **Data Type** | **Description** |
| 1 | Username | Char | For unique identity of user |
| 2 | Email | Email | Save email of user |
| 3 | Validate | Boolean | Check if user is valid |
| 4 | UserId | Int | For unique number of user |

Table 3 Customer Details

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Field name** | **Data type** | **Description** |
| 1 | User | OneToOne | One user has one profile model |
| 2 | Profile image | Image | Contains profile picture of user |
| 3 | Birthdate | Datetime | Birth date of user |
| 4 | Gender | Char | Gender of user |
| 5 | Mobile no | Int | Mobile number of user |
| 6 | Address | String | Address of user |

Table 4 Admin

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Field Name** | **Data type** | **Description** |
| 1 | AdminName | Char | For unique identity of admin |
| 2 | AdminId | Int | Unique number of admin |
| 3 | Password | String | For confirm admin |
| 4 | E-mail | String | E-mail of admin |

Table 5 Book

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Field Name** | **Data type** | **Description** |
| 1 | BookName | String | Name of Book |
| 2 | Book Id | Int | Unique number of book |
| 3 | AuthorName | Char | Author’s name of book |
| 4 | Price | Int | Price of book |
| 5 | CategoryId | Int | Unique number of category |

Table 6 Book Order

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Field Name** | **Data type** | **Description** |
| 1 | OrderId | Int | Unique number of order |
| 2 | BookId | Int | Unique number of book |
| 3 | CustomerId | Int | Unique number of customer |
| 4 | Price | Int | Price of book |
| 5 | Quantity | Int | Quantity of book |

Table 7 Shopping cart

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Field Name** | **Data type** | **Description** |
| 1 | BookId | Int | Unique number of book |
| 2 | CustomerId | Int | Unique number of customer |
| 3 | Price | Int | Price of book |

Table 8 Exchange Book

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Field Name** | **Data type** | **Description** |
| 1 | BookId | Int | Unique number of book |
| 2 | BookName | String | Name of book |
| 3 | ReturnTime | Int |  |
| 4 | CustomerId | Int | Unique number of customer |

# CHAPTER-6 IMPLEMENTATION PLANNING AND DETAILS

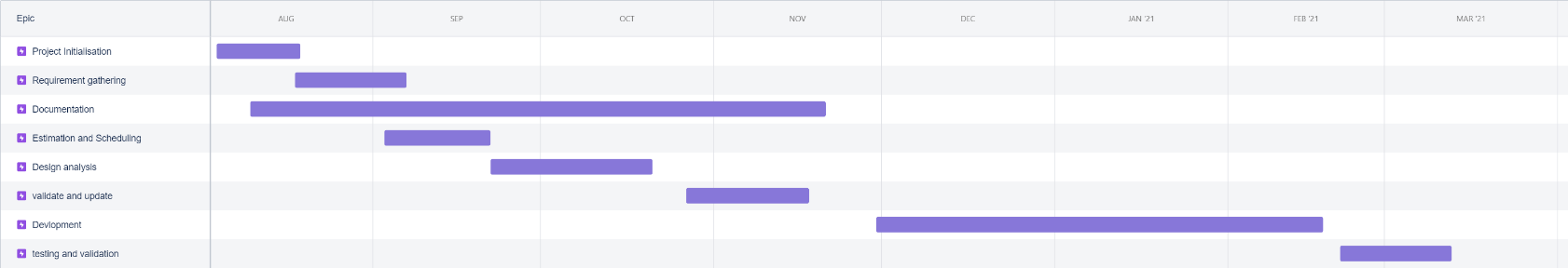


Figure 11 Project planning and implementation

Table 9 project planning and implementation details

|  |  |  |
| --- | --- | --- |
| **Stages** | **Timeline** | **Description** |
| Project initialization | 4thAug-18thAug | Problem statement , goals and objectives |
| Requirement gathering | 18th Aug -9th Sep | Gathering technical and non-technical requirements. |
| Documentation | 10th Aug -20th Nov | Documenting and adding details |
| Estimation and scheduling | 3rd Sep -21stSep | Estimating the time and work flow |
| Design analysis | 22ndsep-20thOct | Deciding models uml diagrams and data dictonary |
| validate and update | 27thOct -17thNov | Verify the work done till now and getting the feedback and update accordingly |
| Development | 30thNov-17thFeb | Developing the system according the document |
| Testing and validation | 21thFeb-12thMarch | Testing the developed system and validating and making changes if required |

# CHAPTER-7 CONCLUSION

* Book nerds is an online website where customers can do purchase, sell and exchange their books with other people in the world. Through a web browser the customer can search for their books as per their title/author ,they can also purchase the book by adding book in to shopping cart/wish list and can exchange as well with other customers.

# CHAPTER 8 REFERENCES

1. bookchums.com:

https://bookchums.com/

1. bookchor.com

https://www.bookchor.com/

1. swapthebook.com

http://swapthebook.com/