Abstract

An Online Library System in my project (Mero Library) has an innovative idea that saves the time of book seekers and book authors or owners. It helps readers to seek for a book through online platform whereas book authors can also provide the information through online platform. Because of this, book readers will get a platform where they can get a huge platform to choose different genres of books and articles.

You can save valuable time and focus on other works too instead of wasting a whole day. Nowadays Internet facilities are over most of the part of country, so to get into the system through an online platform would be much easier for anybody. The idea is simple and innovative to provide the facility of online library in the palm of your hand. A good Interface among book authors and readers.

ACKNOWLEDGEMENT

Foremost, the project team would like to express my sincere thanks to the adviser and mentor **Er. Yuyutshu Banjara** Sir for all his continuous support in this project. I got a chance to have a knowledge and experience through this project. Beside my adviser, I would like to thank our Program Director **Mr. Kiran Ghimire** sir who allowed this project topic and gave me opportunity to work on this innovative and wonderful project on the topic named as "Mero Library".

I am so thankful to my mentors to motivate, encourage and guide me throughout the project. I appreciate all the technical support by BCA program Mahendra Morang Aadarsh Multiple Campus and all the help provided in order to keep this project aligned with its actual objectives.

I have tried to maintain and give credit to everyone who helped me in this project, along with the sources from where I collected required data and information which supported this project. Yet, there may be some unintended errors and some sources or individuals may have missed to mention. I shall feel obligated if they are brought to my notice.

Table of Contents

List of Abbreviations

Abbreviations		Full Forms
BCA	:	Bachelors in Computer Application
CSS	:	Cascade Style Sheet
DFD	:	Data Flow Diagram
HTML	:	Hyper Text Markup Language
JS	:	Javascript
SQLite:		Structured Query Lite
NGO	:	Non Governmental Organization
UI	:	User Interface
IDE	:	Integrated Development Environment

Table of Figures

List of Tables

1. Chapter 1: Introduction

1.1. Introduction

Mero Library is an online library system. It is a modern and useful system for growing authors or publishers. Mero Libraryis a web-based system where once an author registers then he/she can upload his/her articles or the literature details and the readers can see the uploaded files and information and can be able to get access easily if interested. There are 2 clients; Client-A: literature/article author and Client-B: literture/article reader where Client-A and B registers and uploads file details whereas Client-B explores the available details. Both clients can exchange roles according to the time and needs.

Mero Library is a system that:

- Provides a platform where Authors and readers can interact easily and safely.
- Saves Time, money and energy by few minute's exploration on any smart devices.
- Will reduce paper publishing and hoarding boards for reading articles and books.

1.2. Problem Statement

Still in this modern technological era people have to loiter whole day or even weeks or more in search of books or articles to read. People keeps on searching for the report samples, articles and books to read or to have a concept of some document here and there physically. In rural areas specially, searching for important materials is a big headache. Similarly article owners/authors are also not able to get proper publishers to publish their mterials. These all results in waste of time, waste of energy and even waste of money even someone can't find the readers or proper audience.

- Even though the author have done good they wont get audience.
- If anyone finds publishers it might not be at suitable and affordable price.
- Same for readers, they might not get articles or books for long time and should pay high amount to get those.

It takes quite a time for the students, teachers as well as other writers and readers. But my system/site (Mero Library) gathers the details of books or articles which are available for readers and the information of author who is in seek of audience. Book seekers and book authors can access to the materials and audience in one platform.

1.3. Objectives

The main objective of developing the Mero Library is to have a quality and feasible reading system with the aim of following objectives:

- To act as an intermediary between two clients; the author/writer and the reader/audience.
- To study and analyze the requirement specifications of the online open library system.
- To create a system management that would store the necessary information for the owners and audience benefits.

1.4. Scope and Limitation

The scope of an online library system is extensive and continually evolving, encompassing various aspects that cater to the modern needs of users and information managementSince many persons migrate to different places for the studies or works, this online library system will definitely help such persons to find the appropriate and important materials. So, I feel this system has a good scope.

Despite of good scope of this system, there are some limitations too like Users in areas with poor internet connectivity may face difficulties accessing online library resources. Not all books, journals, or resources are available in digital format. Some materials may be restricted due to copyright issues, or the library may not have digital versions of certain older or niche publications. Online libraries lack the physical atmosphere of a traditional library, which some users may miss. [1]

1.5. Development Methodology

After analyzing all the available data models, I found that agile methodology is suitable for my project because agile project management is in an iterative approach to software development projects and ensures feedback can be acted on quickly and responsive changes can be made at each stage of a product cycle. Agile project management incorporates continuous testing and responsiveness to change.

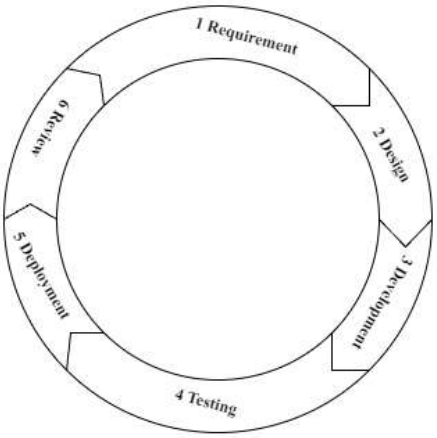

Figure 1: Agile Methodology

1.6. Report Organization

On completion of this project development, we have documented the milestones and the final document report has been organized under the following chapters. It shows the total overview and the organization flow of the report.

Chapter 1: Introduction: This chapter consists of introductory portion along with project's objectives, scope and limitations.

Chapter 2: Background Study and Literature Review: This chapter includes the background study and literature review of the project.

Chapter 3: System Analysis and Design: This chapter contains the requirement analysis and feasibility study, system design with system modeling and activity design with all necessary diagram.

Chapter 4: Implementation and Testing: This chapter consists of implementation and testing with development methodology, tools used and testing process implemented throughout the development along with details about the system testing. It describes the different techniques used for the entire development process of front-end and back-end. How modules are developed and testing for these modules and system.

Chapter 5: Conclusion and Future Recommendation: This presents the conclusions of the project with future enhancements possible for the project.

Chapter 6: Appendices: It has screenshots of all the implementation with their descriptions.

Chapter 2: Background Study and Literature Review

2.1. Background Study

A background study of an online library system involves understanding its origins, development, components, and the context in which it operates.The existing systems in the context of this project include traditional method and few websites.

Few existing web portals platform are offering open source online library system and few are paid platforms. Some of them cover only limited area while some are of different countries that are more developed in comparison.

- **Traditional Systems**

The traditional method of library is quite similar as buying/selling or lending/renting other general products or goods in the market. Traditional libraries primarily housed physical materials such as books, journals, newspapers, magazines, maps, and other printed resources. Users has to physically visit the library to borrow books and return them within a specified timeframe.

Although we have access to the resources required, there are some advantages of traditional system over this project. Few advantages of traditional system are as follows:

- Does not require internet or web enabled device.
- Does not require knowledge to operate web-enabled devices.

Similarly, there are disadvantages more than advantages of traditional method over this system, some of them are:

- Time Consuming
- Lack of reach of materials
- It is costly
- Searching articles, books and other materials might be difficult in new areas

- **Web Portals**

Web portals are quite similar to the system that has been developed. There are few web apps/system in Nepal which provides online library system services which are not sufficient. Those systems are limited in certain areas. Some of them are beyond user's choice.

There are some advantages of those developed portals and systems, few of them are:

- Low cost
- Easy searching and finding
- Non-physical platform

There are some additional disadvantages apart from the advantages. They work as a commercial book shops. Some of the disadvantages of existing systems are:

- Treats like general commodities
- Only specific for certain regions or locations

2.2. Literature Review

The physical library system is one of the elemental part of the society. Considering these problems and requirements different developers developed different system for solving this problems so that the publishers/authors and owners can easily get and read articles, books and other materials through different platforms.

Open Library: One web page for every book ever published. It's a lofty but achievable goal. To build Open Library, we need hundreds of millions of book records, a wiki interface, and lots of people who are willing to contribute their time and effort to building the site. To date, Founders have gathered over 20 million records from a variety of large catalogs as well as single contributions, with more on the way. Open Library is an open project: the software is open, the data are open, the documentation is open, and they welcome our contribution. The ultimate goal of the Open Library is to make all the published works of humankind available to everyone in the world. While large in scope and ambition, this goal is within our grasp. Achieving it will require the participation of librarians, authors, officials and technologists. Open Library is an open project: the software is open, the data are open, the documentation is open, and they welcome other's contribution. Whether you fix a typo, add a book, or write a widget--it's all welcome. It is affilated to The Internet Archive. [2]

Project Gutenberg: Project Gutenberg is an online library of free eBooks. Project Gutenberg was the first provider of free electronic books, or eBooks. Michael Hart, founder of Project Gutenberg, invented eBooks in 1971 and his memory continues to inspire the creation of eBooks and related content today. The mission of Project Gutenberg is to encourage the creation and distribution of eBooks. Project Gutenberg has a long history,

which predates the modern Internet and continues until today. The Inernet Archive and The Librivox.org are the affilated partners of Project Gutenberg. Project Gutenberg ensures that the texts they offer are in the public domain. They do not distribute copyrighted material without permission and comply with copyright laws in different countries. [3]

E-Pustakalaya: E-Pustakalaya is an extensive digital repository that hosts a wide range of educational materials, including textbooks, reference materials, academic publications, literature, audiovisual resources, and more**.** E-Pustakalaya is an education-focused free and open digital library. They provide free and easy access to age-appropriate books that help improve children's reading skills and develop a reading culture in schools. Teachers can benefit widely from our selection of books, various teaching resources and educational materials in core subjects as well as various other subjects such as agriculture, health, environment, and technology. We also have plenty of reference materials professional development. The communities around the schools can also access the content on the digital library on their laptops, tablets or mobile phones. The platform may collaborate with educational institutions, publishers, NGOs, and volunteers to curate, digitize, and contribute educational content to the repository. [4]

Chapter 3: System Analysis and Design

3.1. System Analysis

3.1.1. Requirement Analysis

Requirements convey user's expectations from the software product. The requirements can be obvious or hidden, known or unknown, expected, or unexpected from a client's perspective. Therefore, it is important to make it clear to the development team and stakeholders.

i. Functional Requirements

The proposed system is able to:

- Allow administrator to add materias, users and defaulters details
- Allow the administrator to delete publishers/authors, readers and defaulters details
- Allow the administrator to search data in the database.

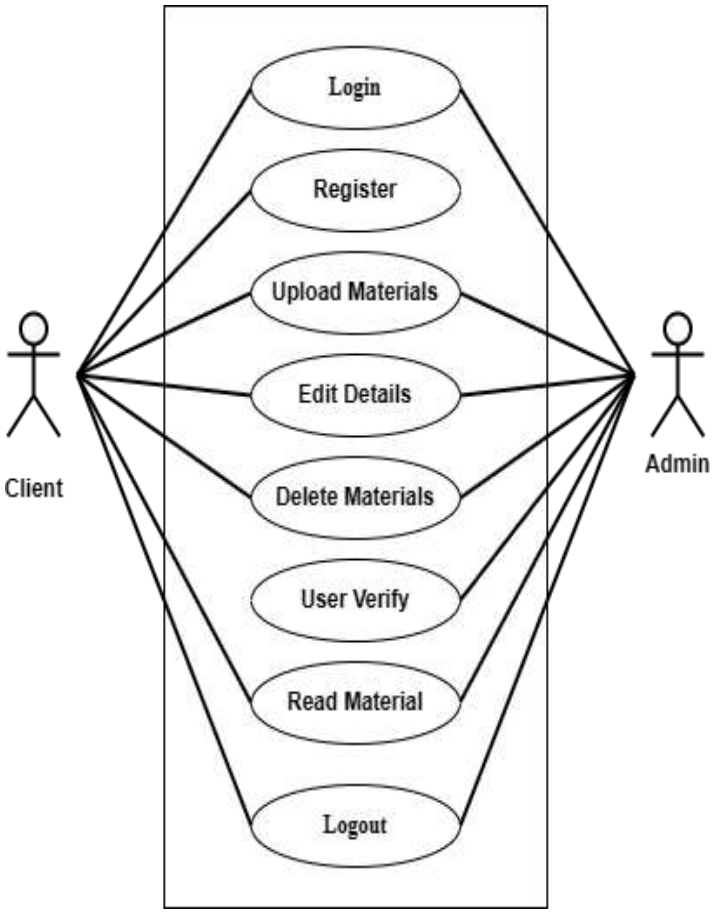

Figure 2: Use-Case diagram of Online Library System

ii. Non-Functional Requirements

This system:

- Typically grants access to accounts when users enter the correct username and password.
- Can be portable since it is web-based.
- Must have faster login and responding.
- Must have storage capacity to store as many data as possible.

3.1.2. Feasibility Analysis

As feasibility study is an assessment of the practicality of a proposed plan or method, my proposed solution might satisfy all the requirements and is flexible enough. Changes can be easily done for the further future requirements.

According to its workability, impacts on the organization, ability to meet user needs and effective use of the resources, its main task done during feasibility study are:

i. Technical Feasibility

We can say that this system is technically feasible since Microsoft Edge browser is used for scrutinizing and it has been developed on MS-Windows 10 platform and a high configuration of 12GB RAM on Intel Core i5-7th Generation processor.

It meets the need of the proposed system. This includes the study of function, performance and constraints that may affect the ability to achieve an acceptable system.

ii. Operational Feasibility

This system can be operationally feasible since it can reduce the cost of developing the system without undermining its quality or product. Well trained manpower is not necessary to operate. Users can easily access the system and each process are smoothly done which also makes easy to operate. It provides timely, suitable, accurate and useful formatted information.

It perfectly intends to solve the stated problem and leverage the opportunities identified during the scope definition phase. Additionally, it will also satisfy every requirement identified in its requirement analysis phase.

iii. Economic Feasibility

It is designed with freeware software. Also no external personal is needed for its development and the whole system cycle will be operated by two people. This factor will decrease the overall expenses of the project development life cycle. There is no extra cost required for this project. This system will be available free of cost for all users. Thus, it can be called Economically Feasible.

iv. Schedule Feasibility

A Gantt chart is used for planning of this project, and it became a useful way of showing what work is scheduled to be done on a specific day. It also helped me to view the start and end dates of a project in one simple chart. A Gantt chart was incredibly useful because it allowed me to simplify complex projects into an easy-to-follow plan and track the status of tasks as work progresses. The following Gantt chart shows the timeline required for completion of the project. However, due to some inconveniences, timeline may be affected and updated as per the requirement.

Table 1: Gantt-Chart

		WEEKS											
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
ACTIVITIES	**Planning**												
	Research												
	Design/coding												
	Implementation												
	Testing												
	Documentation												

3.1.3. Object Modeling (Class Diagram)

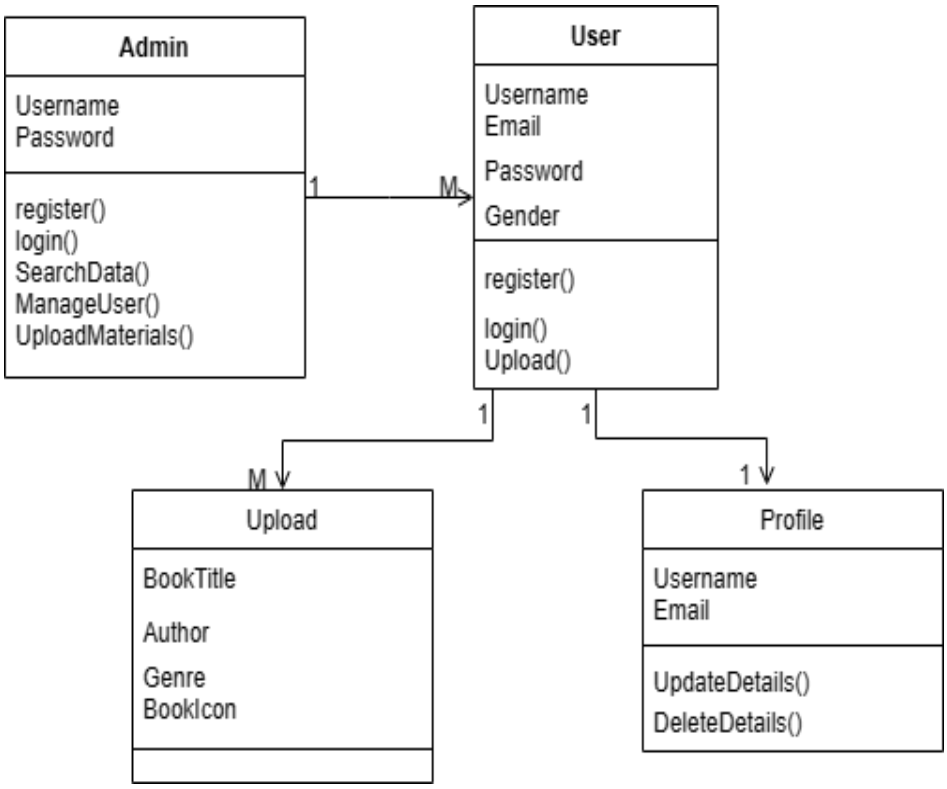

Figure 3: Class Diagram of Online Library System

3.1.4. Dynamic Modeling: state and sequence diagram

This system's sequence diagram aids in understanding both the needs for new features and applications as well as those that are already there.

Sequence Diagram for Admin

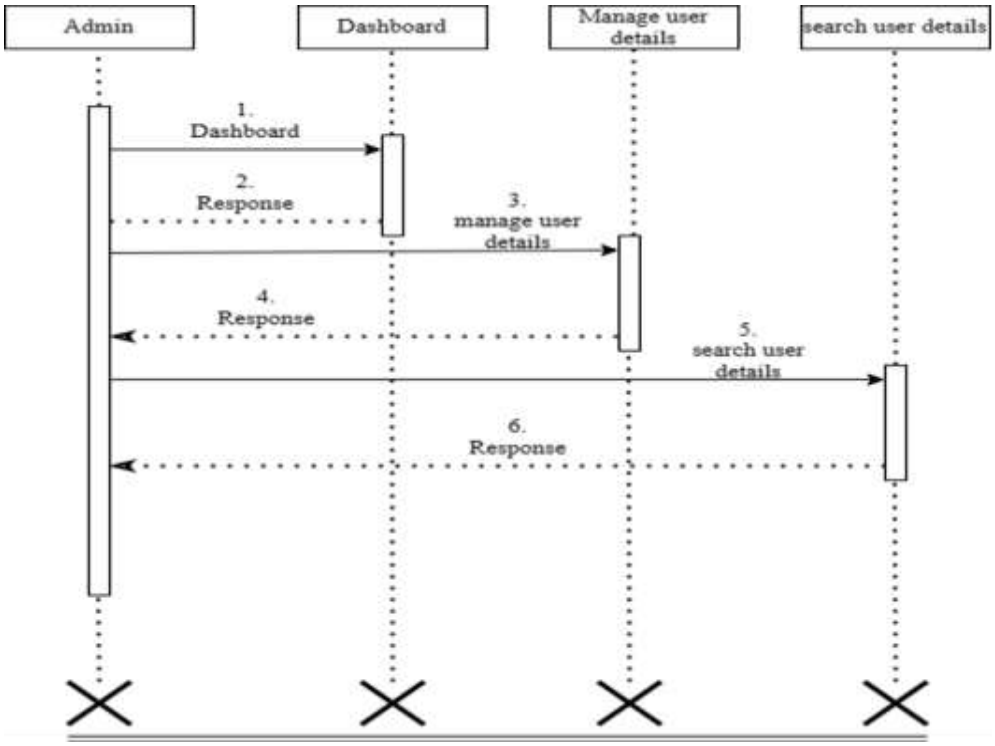

Figure 4: Admin Sequence Diagram of Online Library System

Sequence Diagram for User

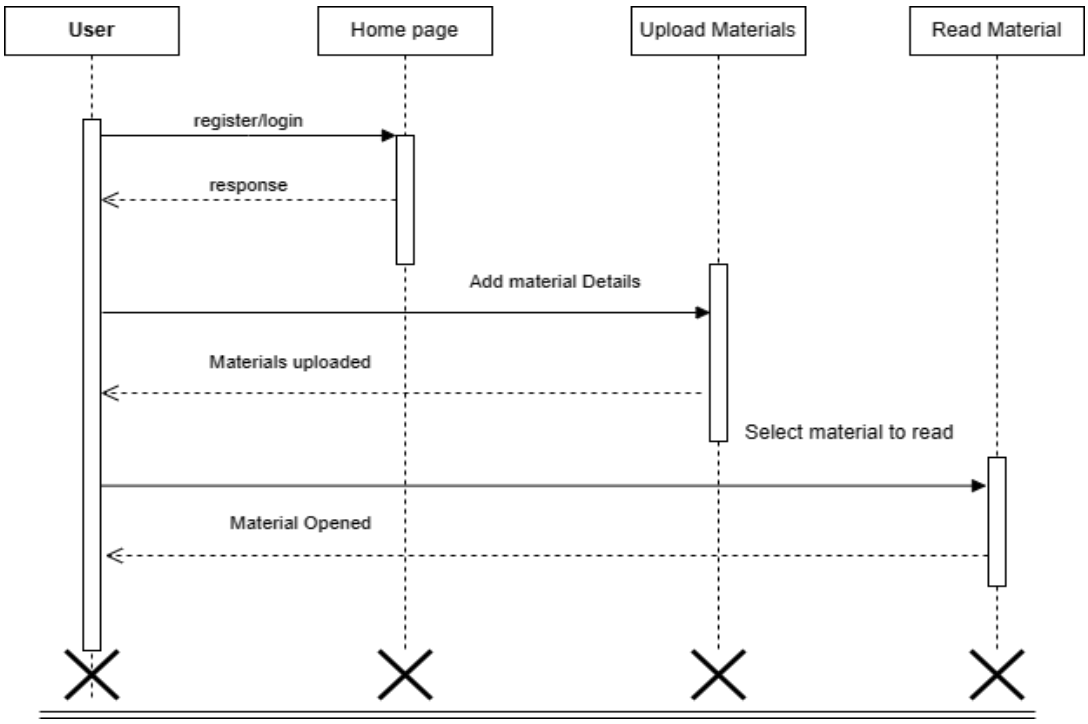

Figure 5: User Sequence Diagram of Online Library System

3.1.5. Process Modeling: Activity Diagram

The system starts its operation by generating a user and determining whether it is a new user or not. After confirming that the administrator may review the information, the user is given access to it.

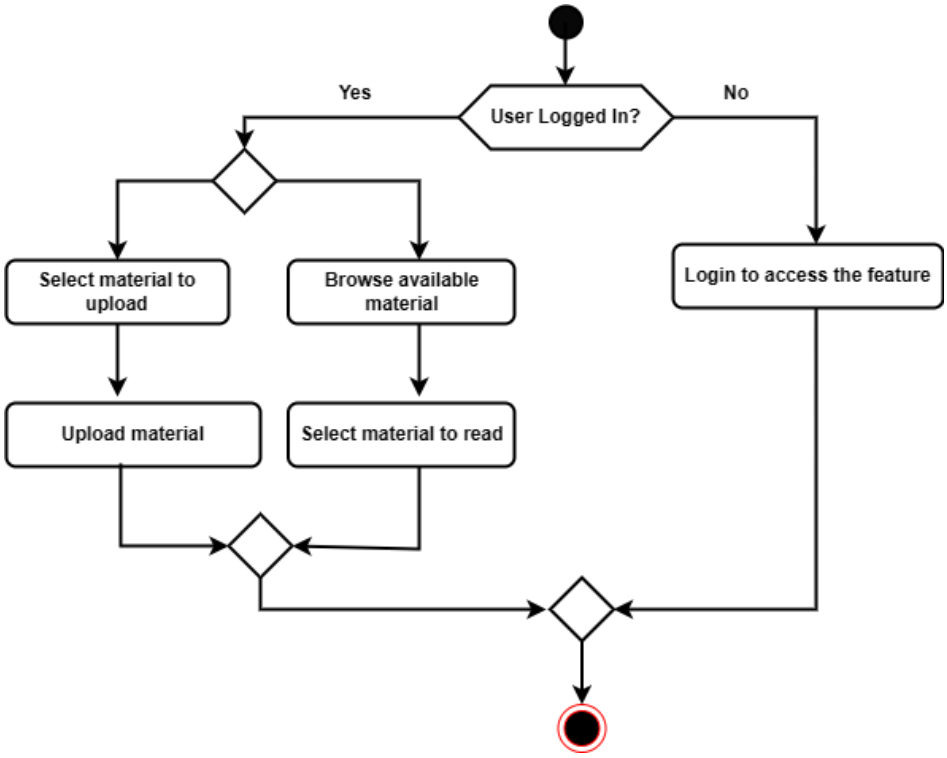

Figure 6: Activity Diagram of Online Library System

3.2. System Design

The block diagrams below show a simple visual representation for describing actions of the complex system. The blocks are often referred as black boxes, which represent mathematical or logical operations that occur in sequence.

3.2.1. Refinement of classes and objects

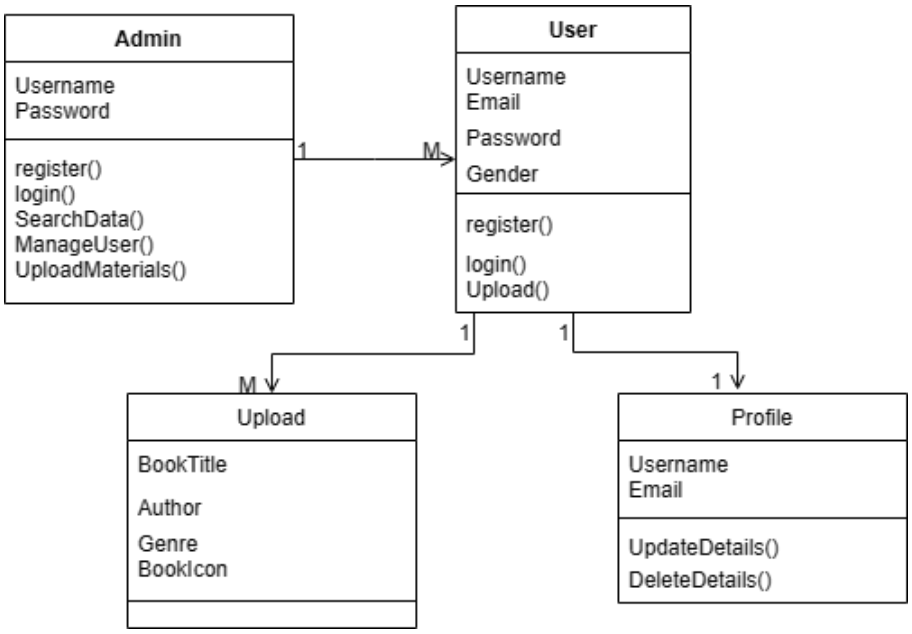

Figure 7: Refinement of Classes and Objects for Online Linrary System

3.2.2. Component Diagram

For Admin

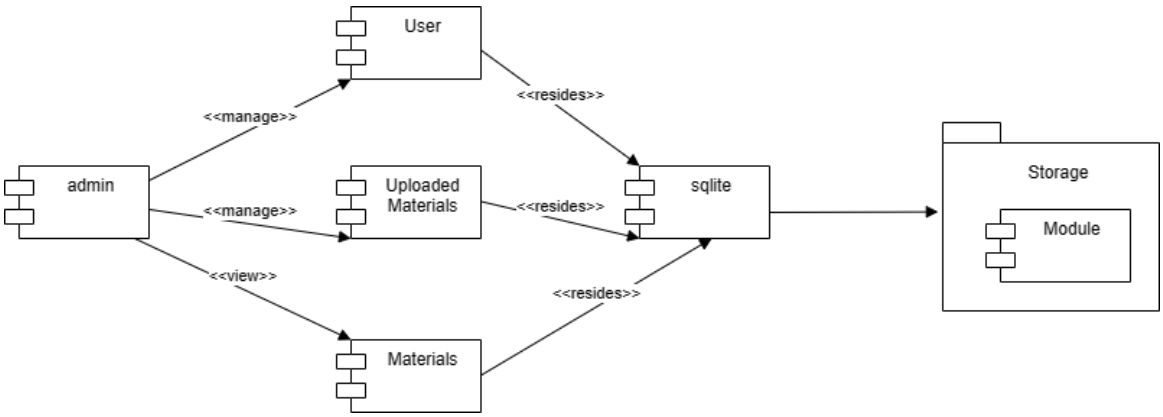

Figure 8: Component Diagram of Online Library System

For User

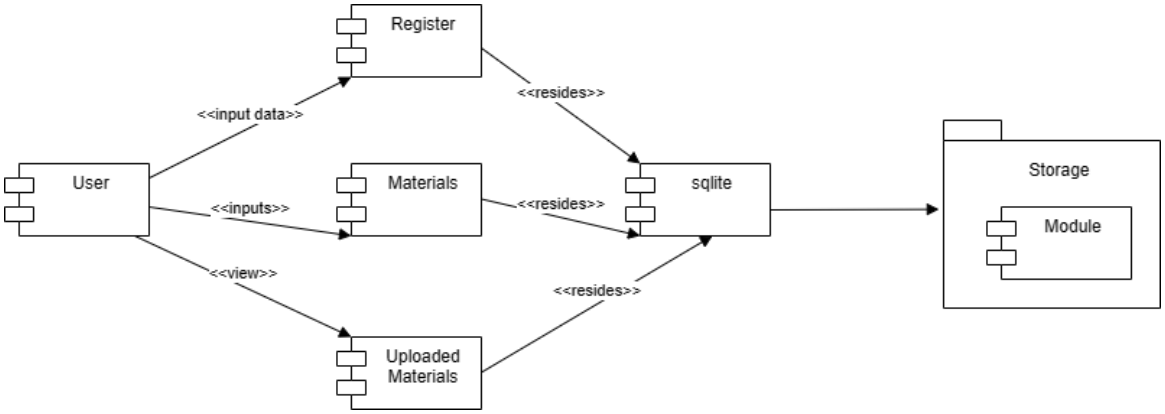

Figure 9:User Component Diagran of Online Library System

3.2.3. Deployment Diagram

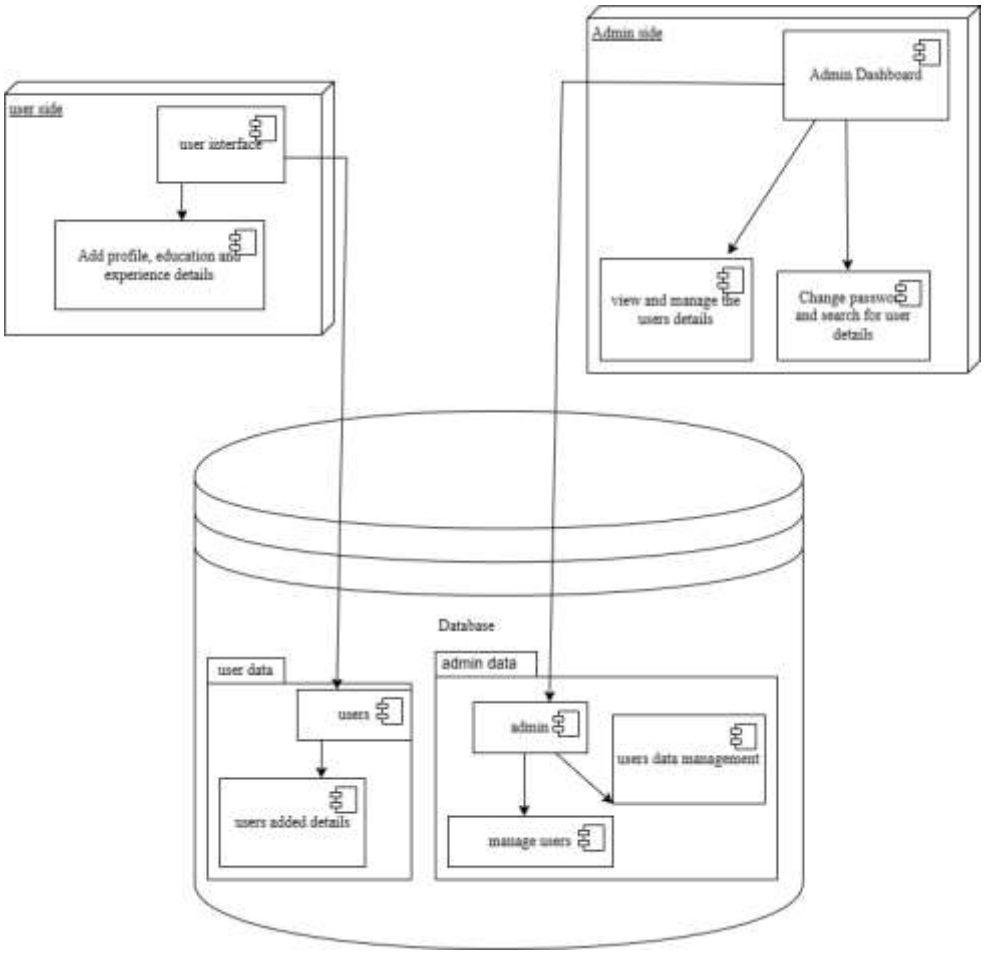

Figure 10: Deployment Diagram of Online Library System

3.3. Algorithm

stepwise algorithm for a content-based filtering algorithm:

Step 1: Start

Step 2: Input Content metadata (e.g., features, categories, keywords)

Step 3: Clean and preprocess the data, removing noise and irrelevant information.

Step 4: Assign weights or scores to the features based on their importance or relevance to the user.

Step 5: Compute the similarity between items based on their content features.

Step 6: Select the top-n most similar items based on their content similarity scores.

Step 7: Compare the algorithm's performance against baseline methods.

Step8: Stop

The reasons behind using Content-based filtering recommendation system are:

- Recommendations are highly relevant to the user.
- Content-based filtering systems are generally easier to create. [5]

Chapter 4: Implementation and Testing

4.1. Implementation

4.1.1. Tools Used

• **Front End Tool**: Front end of this application is designed using HTML, CSS, and Bootstrap (CSS Framework). This Online Library System is built utilizing the most popular combination of web technology. Bootstrap is a free and open-source front-end development framework for the creation of websites and web apps. It is used as front-end framework to speed up development process. The Bootstrap framework is built on HTML, CSS and JavaScript (JS) to facilitate the development of responsive, mobile-first sites and apps.

• **Back End and Database Tool**: Back end of this system is built with Django. SQLite is used for managing database. The system is hosted in in-built python server.

• **Documentation Tool**: Microsoft Word or MS Word, a graphical word processing program where users can type is used for the documentation of our system Online Library System. Draw.io and PlantUML is used for drawing diagrams and figures.

• **Software Used**: For the development process, Notepad and VS Code is used for writing the code. Python-django in-built server is used for hosting the application locally. Edge browser is used while creating this web application.

4.1.2. Implementation Details of Modules

Client Module

It is the module for the client users. When the Clients login with his/her Username and password then this interface appears as the landing page. It consists of different material details, info about the system and the contact details.

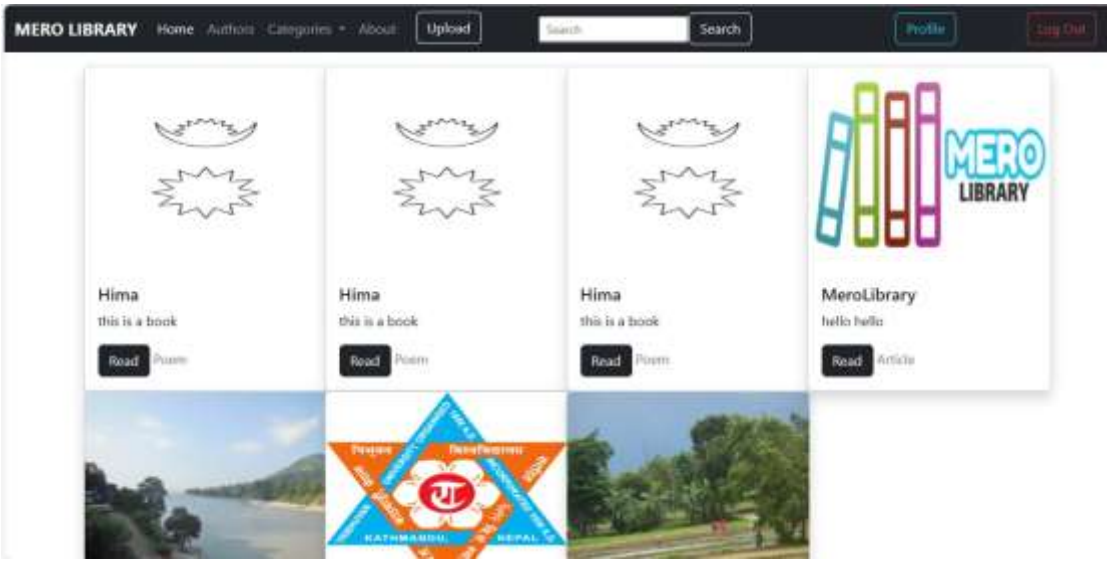

Figure 11: Client Module of Online Library System

Admin Module

It is the module which can be seen only to the admins of the system. Here Admins can verify clients, update and delete client details and also generate the result of the material uploads and interested client details.

Figure 12: Admin Module of Online Library System

4.2. Testing

Unit testing and system testing are done after the Implementation of the system. User module is tested and it is found that the system works as it is designed. Login is done with registered user data only. This system only consists user module so as system testing, user module is tested. Following test cases are done for Unit testing and System testing.

4.2.1. Test Cases for Unit Testing

- **Testing For Sign-up Page**

Table 2: Test cases for unit testing

S. N.	Objective	Test Description	Expected Outcome	Result	Remarks
1	Checking Registration Process	Filling the below details in register form: Username= “Aayush” Gender=(click on) “Male” Email=Wastiaayush789@gmail.com Password=”Aayush123” (Click on submit)	Account created successfully and user’s data stored in database successfully	Registration Successful	
2	To check user’s login process	Filling the details in form: Email=‘Wastiaayush789@gmail.com” Password= ”Aayush123” (click on login)	user must successfully get logged into otherwise wait for verification and try again	Unsuccessful	Not verified by the admins
3	Recheck user’s login after a while	Filling the details in form: Email=‘Wastiaayush789@gmail.com” Password= ”Aayush123” (click on login)	user must successfully get logged into	Successful	
3	To upload material details	Click on Upload and fill the pdf Material details: Book Icon=’Upload Icon’ Book_Name= ’Book1 description= “This is a book of horror.	Data must be inserted in the database and available in the user area in the table	Successfully inserted and displayed	

		Book File = “Upload pdf file” Genre = choose genre “story” (Click on Upload)			
4	To read the material	Checking the preferred material with satisfying conditions and clicking on read button	File should open in new tab	Successfully displayed	
5	To check To check the login process of admin	Filling the following details in the login form: Username= “admin@admin.com” Password= “admin123”	Admin must be logged into system successfully	Successful	

4.2.2. Test Cases for System Testing

This testing is done to ensure that the system meet the requirement. We have performed the system testing of our system and achieved the following result.

Table 3: Test case for System Testing

S.N.	Objectives	Test Cases	Actual Result	Result
1	To check the upload details of user	i. User logs in into the website using following details: Email= ‘Wastiaayush789@gmail.com’ Password=’Aayush123’ ii. he/she clicks on the upload menu in navbar.	Uploaded data must be inserted in the database	Successful

		iii. He/she uploads the material details as per listed details. iv. he/she clicks to upload button		
2	To check the verification request of user	i. admin opens the admin page and clicks on verify button ii. Admin checks the registered details iii. Admin clicks on the verify button and makes user verified	User must be able to log into his/her profile after verification	Successful

Chapter 5: Conclusion and Future Recommendations

5.1. Lesson Learnt

The development process was very informative. During the course of this system's development, we found that several features of web development could be used to develop a system. Different technologies that I was unknown about have been brought in use. Time management was the main factor. This system is developed with proper timely work division, resources management was the key. I learnt presentation ideas and skills. Some important learning's of this project are:

- Time management,
- Using different bootstrap modals,
- System modeling and diagrams,
- Database design,
- Plan a brief system development activity
- Schedule management, etc.

5.2. Conclusion

This system has been developed to help people find the platform where he/she can get reading materials as per their needs. It fulfills the needs of many book/articles owners and readers. It was fun and challenging working on this system. The system achieved the mentioned objectives in chapter 1 of this documentation using HTML, CSS, JavaScript (JS), Bootstrap, Django and SQLite. This system is designed and implemented to be capable of allowing users to post or find reading materials, add various descriptions of mterials, delete them etc. The option to be able to choose a reading material and list the materials increases functionality of the website. This system will receive regular updates.

5.3. Future Recommendations

Due to time bound, it was difficult to test and implement everything it was planned for. Future developments will be done according to the feedbacks and reviews obtained from the system. Regular and frequent upgrading and refinement in the database will be carried out.

Following are some of the future enhancements of this system:

- Online Payment and premium user system
- Separate Audio book and pdf book gallery
- Proper backend upgrade and maintenance

Appendices

Screenshots:

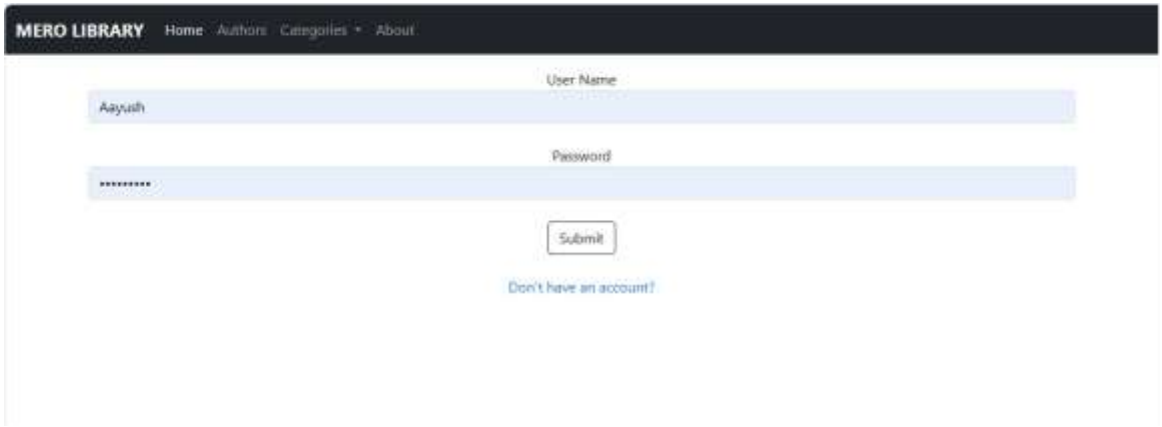

Figure 13: Login page of Online Library System

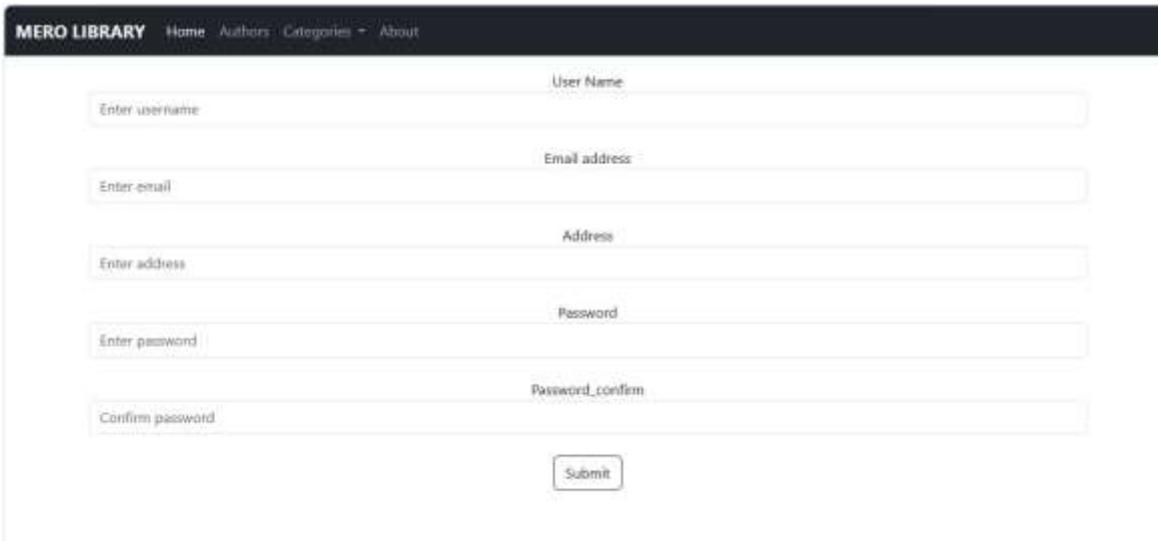

Figure 14: SignUp panel of Online Library System

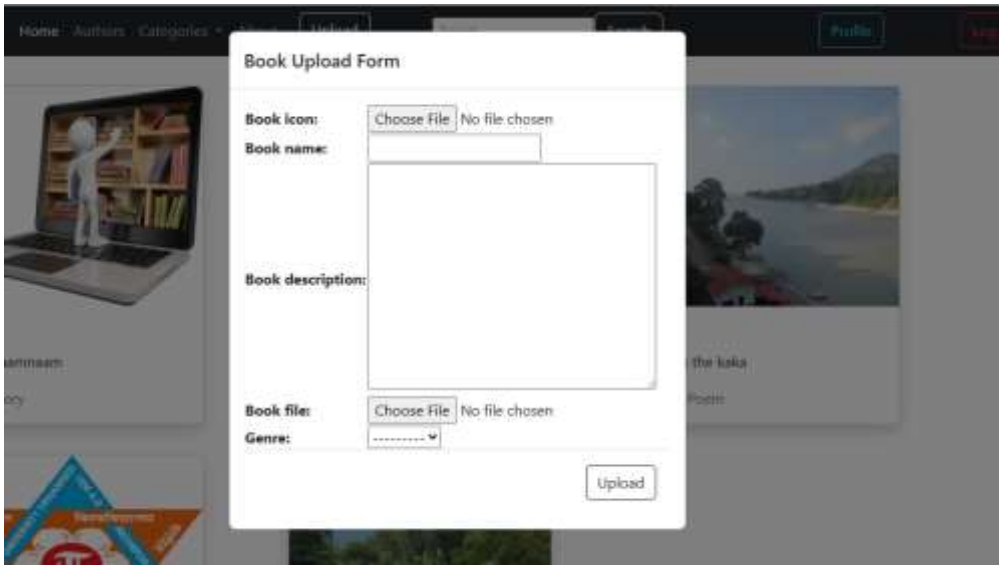

Figure 15: Material Upload panel of Online Library System

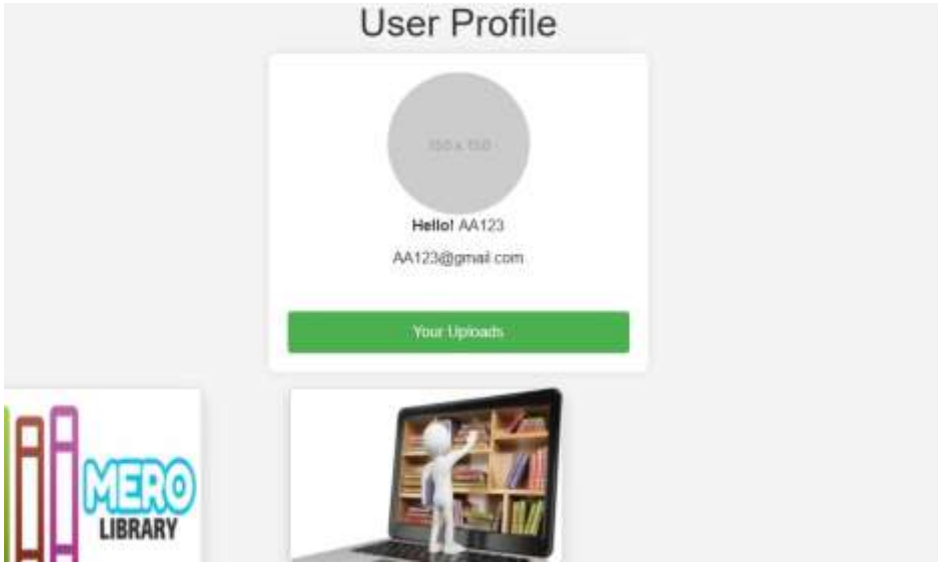

Figure 16: User Profile of Online Library System

Figure 17: Admin panel of Online Library System

Source Codes:

Home/Views.py

```
from django.shortcuts import redirect, render

from Home.forms import upload_book_form

from django.contrib import messages

from Home.models import books

from pathlib import Path

from django.contrib.auth.decorators import login_required

from django.http import Http404

from django.contrib.auth import get_user_model

# Create your views here.

def handel_book_upload(request):

    if request.method == 'POST':

        form_data = upload_book_form(request.POST, request.FILES)

        book_ext = request.FILES.get('book_file').name

        if Path(book_ext).suffix == '.pdf' or Path(book_ext).suffix == '.PDF':

            if form_data.is_valid():

                book_data = form_data.save(commit=False)

                book_data.author = request.user

                form_data.save()

                return messages.success(request, f"Uploaded file successfully...")

            else:

                return messages.error(request, "Failed to upload file...")

        else:
```

```
            messages.error(request, "Only pdf file format is accepted...")
@login_required(login_url='login')
def index(request):
    get_books = books.objects.all().order_by('-id')
    res = {
        'book_form': upload_book_form,
        'books': get_books,
    }
    if request.method == 'POST':
        handel_book_upload(request)
    return render(request, 'index.html', res)
@login_required(login_url='login')
def Authors(request):
    User = get_user_model()
    users = User.objects.all()
        res = {
        'book_form': upload_book_form,
        'users':users,
    }
    if request.method == 'POST':
        handel_book_upload(request)
    return render(request, 'Authors.html', res)


def About(request):
```

```
    res = {
       'book_form': upload_book_form,
    }
    if request.method == 'POST':
        handel_book_upload(request)
    return render(request, 'About.html', res)
@login_required(login_url='login')
def read_book(request):
    res = {}
    return render(request, 'read_book.html', res)
@login_required(login_url='login')
def categories(request, key):
    key = key
    keys = ('Poem', 'Article', 'Story' , 'Reports', 'Others')
    get_books = books.objects.all().filter(genre=key)
    if key not in keys:
        get_search = books.objects.filter(book_name__contains=key)
        get_books = get_search
    if request.method == 'POST':
        handel_book_upload(request)
    res = {
        'book_form': upload_book_form,
        'books': get_books,
    }
```

```
    return render(request, 'index.html', res)
@login_required(login_url='login')
def profile(request):
    u_books = books.objects.filter(author=request.user.username)
    res = {
        'books':u_books
    }
    return render(request, 'Profile.html', res)
@login_required(login_url='login')
def dele(request, id):
    try:
        book = books.objects.get(id=id)
    except book.DoesNotExist:
        raise(Http404)
    if book.delete():
        messages.error(request, "Book deleted..")
        return redirect('Home:profile')
    return render(request, 'Profile.html')
@login_required(login_url='login')
def reco(request, key):
    reco_book = books.objects.filter(author=key)
    res = {
        'r_books':reco_book,
    }
```

```
    return render(request, 'index.html', res)
```

Home/URLS.py

```
from django.contrib import admin
from django.urls import path
from Home import views

app_name = 'Home'

urlpatterns = [
    path('', views.index, name='Home'),
    path("Authors", views.Authors, name='Authors'),
    path("About", views.About, name='About' ),
    path('categories/<str:key>', views.categories, name='categories'),
    path('profile/', views.profile, name='profile'),
    path('delete/<int:id>', views.dele, name='delete'),
    path('reco/<str:key>', views.reco, name='reco'),
]
```

References

[1] G. Gaga, "The Digital Library Management System 2021," February 2021. [Online]. Available: https://www.researchgate.net/publication/369020218_The_Digital_Library_Management_System_2021_Design_and_Implementation_of_an_integrated_Library_Management_System. [Accessed 02 11 2023].

[2] Internet Archive, "Open Library," California State Library and the Kahle/Austin Foundation., California, 2008.

[3] P. Gutenberg, "About Project Gutenberg," ibiblio, 1971.

[4] "E-Pustakalaya," OLE Nepal, [Online]. Available: https://pustakalaya.org/en/. [Accessed 02 12 2023].

[5] The Upwork Team, "Upwork," Upwork® Global Inc., Jan 2021. [Online]. Available: https://www.upwork.com/resources/what-is-content-based-filtering. [Accessed 01 12 2023].