

BOOK RECOMMENDER SYSTEM

UCS503 Software Engineering Project Report

End-Semester Evaluation

Submitted by:

**Vasu Verma(102003142)
Aryaman Kalia(102003099)
Rajneesh Verma(102183032)**

Submitted to Ms. Anamika Sharma



THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Computer Science and Engineering Department

TIET, Patiala

November 2022

TABLE OF CONTENTS

S.No.	TOPIC	Page No.
1.	Project Selection Phase	
1.1	i. Software Bid	
2.	Planning Phase	
2.1	i. Project Write Up,	
2.2	ii. Feasibility Report	
2.3	iii. Gantt Chart	
3.	Analysis Phase	
3.1	i. Use-Case diagram	
3.2	ii. Use Case Scenario	
3.3	iii. Swimlane diagrams	
3.4	iv. Data Flow Diagrams –Level 0, Level 1	
3.5	User Story Cards	
3.6	v. Software Requirement Specification (SRS) in IEEE Format	
4.	Design Phase	
4.1	Class Diagram & Object Diagram	
4.2	Sequence Diagram	
4.3	Collaboration Diagram	

UCS503- Software Engineering Lab

4.4	Database Design : ER Diagram	
4.5	State Chart Diagrams	
5	Implementation	
5.1	Component Diagrams	
5.2	Deployment Diagrams	
5.3	Screenshots of Working Project	
6	Testing	
6.1	Cyclomatic Complexity	
6.2	Test Cases	
6.3	Test Reports	

UCS503- Software Engineering Lab

Software Bid/ Project Teams

UCS 503- Software Engineering Lab

Group : _
3coe6

Dated:-25August2022

Team Name:

Team ID (will be assigned by Instructor):

Please enter the names of your Preferred Team Members. :

- You are required to form **a three to four person** teams'
- Choose your team members wisely. You will not be allowed to change teams.

Name	Roll No	Project Experience	Programming Language used	Signature
Vasu Verma	102003142	Wine Quality Predictor	Python	
Rajneesh Singh Verma	102183032	Chess Game	Python	
Aryaman Kalia	102003099	Plagiarism Analyzer	Python	

Programming Language / Environment Experience

List the languages you are most comfortable developing in, **as a team**, in your order of preference. Many of the projects involve Java or C/C++ programming.

1. Python
2. C++
3. C

Choices of Projects:

Please select **4 projects** your team would like to work on, by order of preference:

First Choice	Book Recommendation System : A ML model to recommend book according to given dataset using machine learning algorithms.
Second Choice	Email Spam Classifier : A ML model to classify between spam & non spam emails to given dataset using machine learning
Third Choice	Image compression : A ML model to to compress the image size using variational autoencoders .

PROJECT WRITE-UP

OVERVIEW

This is book recommendation system. This project aims to analyse and read the preferences of the user and provide the user with more similar book title that user will like. This has been achieved using KNN algorithm in Machine Learning. A k-nearest-neighbor algorithm, often abbreviated k-nn, is an approach to data classification that estimates how likely a data point is to be a member of one group or the other depending on what group the data points nearest to it are in. This software will make finding books easier for user.

FUNCTIONAL REQUIREMENTS

- Applicants dashboard page
- A page with list of books.
- A feature to filter and organize books based on rating, genre and other descriptions.
- A feature to enable applicants to explore.
- The different books based on their preferences.

Database: The database based on past preferences of the user will be created.

Algorithm: It will capture preferences and suggest books based on it.

Historical Data: It will be collected by the system.

Searching Reporting Requirement: Will tell users how to search data.

NON FUNCTIONAL REQUIREMENTS

- **EASE OF USE** :- A new user should be able to use the recommender engine without putting too much efforts on learning how to use it, and, in case of doubt, there must be some help to solve their doubts.
- **SPEED**:- The recommender engine should generate recommendations within a time frame of 500 milliseconds.
- **DATA SECURITY & INTEGRITY**:- The data of user will remain secure and safe.
- **MAINTAINABILITY**:- The software will be easy to maintain & repair if documentation is properly used.

FEASIBILITY

Operational Feasibility

The project will be implemented in a way that it will allow the functioning of recommendations smoothly. It will provide a user-friendly user interface in a modular fashion

Economic Feasibility

Clients will need to handle no extra software or hardware apart from a stable high speed Internet connection and a computer device . so it is okay to conclude that the product is cheap and provides immense benefits

Technical feasibility

The main technologies and tools are :-

Frontend-HTML, CSS,NODEJS

Backend-Python

Each of technologies are freely available and the technical skills required are manageable. Time limitations of the product development and the ease of implementing using these technologies are synchronized.

Initially the web site will be hosted in a free web hosting space, but for later implementations it will be hosted in a paid web hosting space with a sufficient bandwidth. From these it's clear that the project Book recommender system is technically feasible.



















Legal feasibility

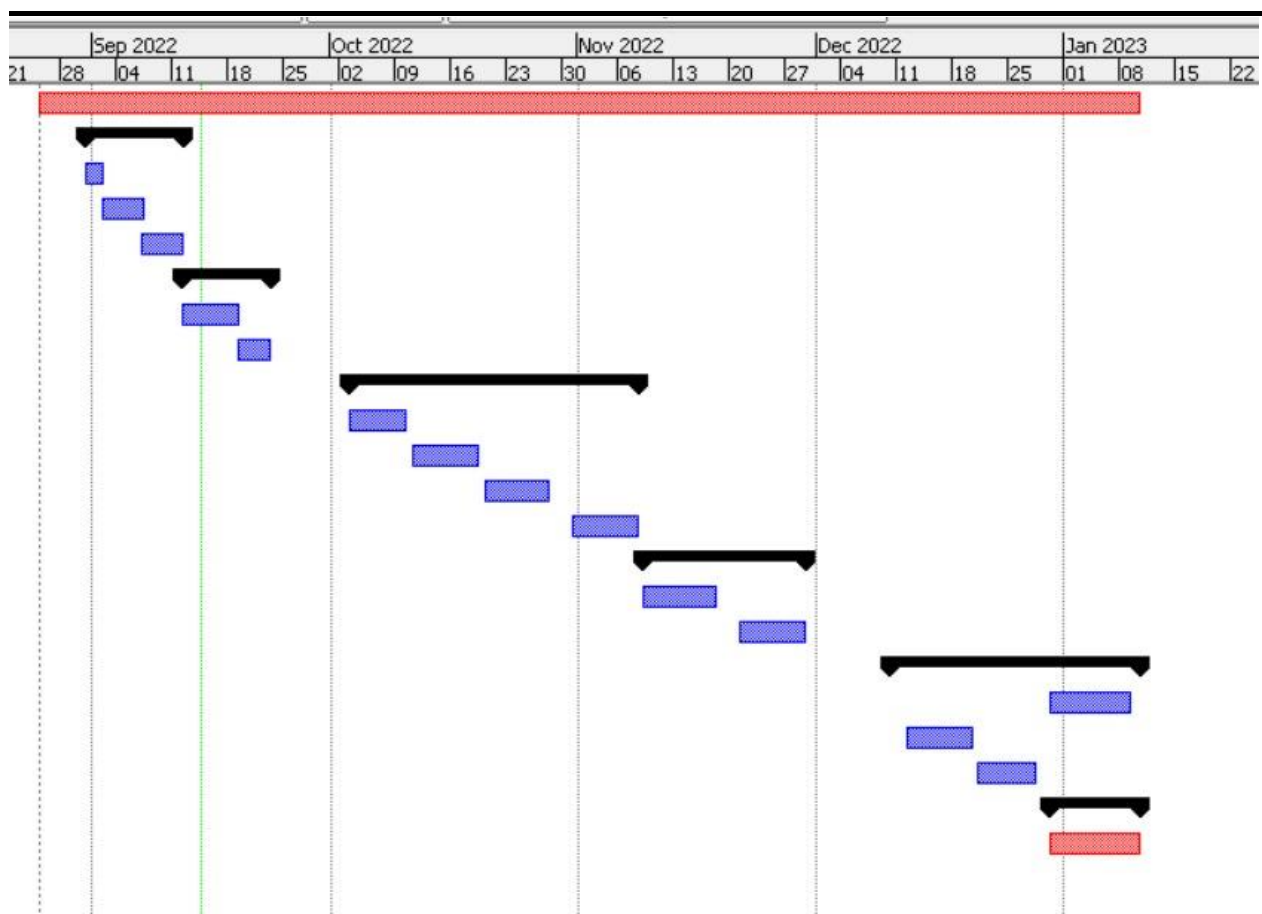
The website is examined In legal feasibility. Protection of user data, project certification and examining legal impediments to project execution. Overall our proposed enterprise meets legal and ethical standards.

Cultural/ Behaviour Feasibility

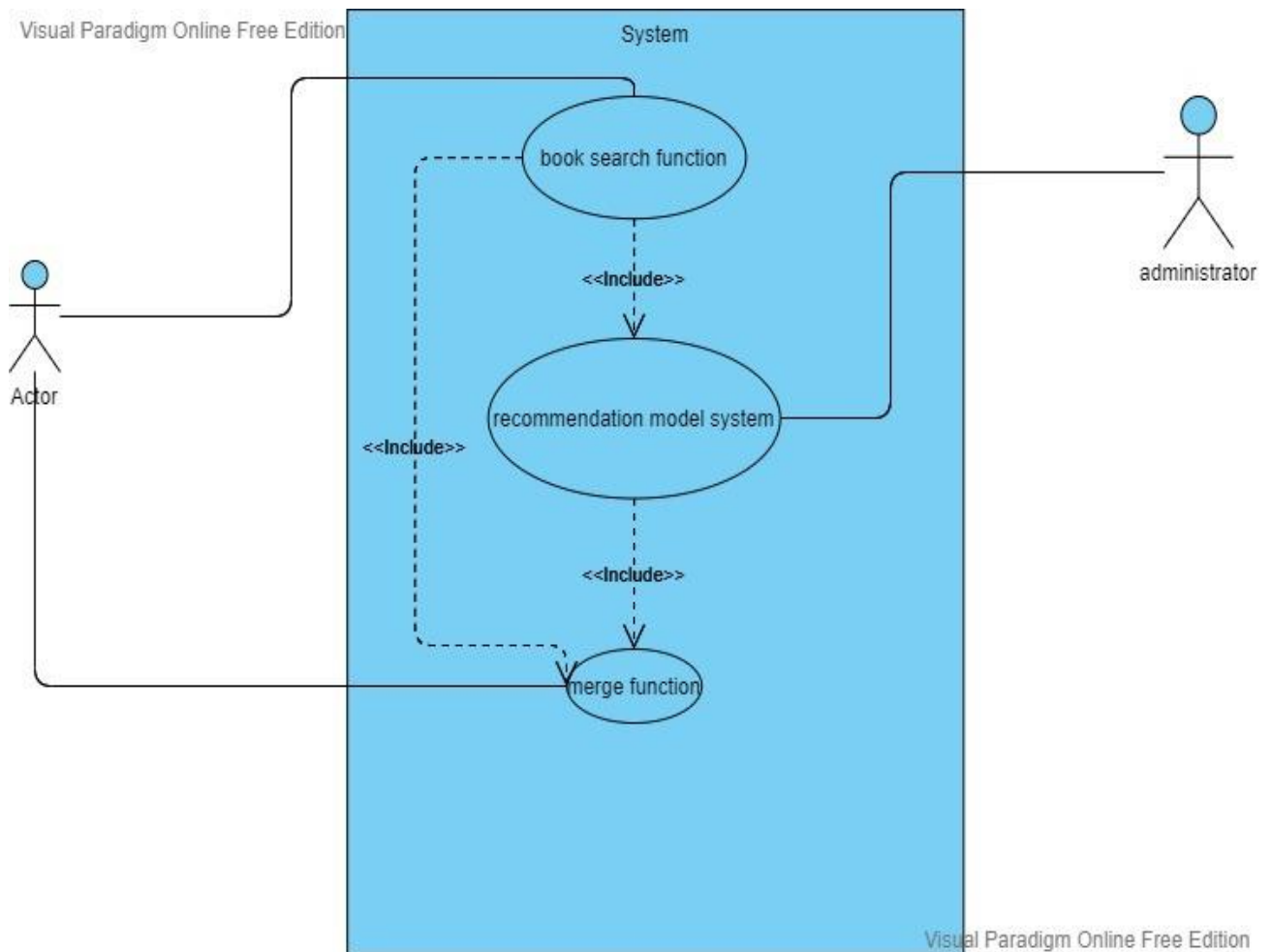
Our project does not hurt the sentiments of any religion, their ethical and behavioral practices and mean to be used by anyone from anywhere.

GANTT CHART

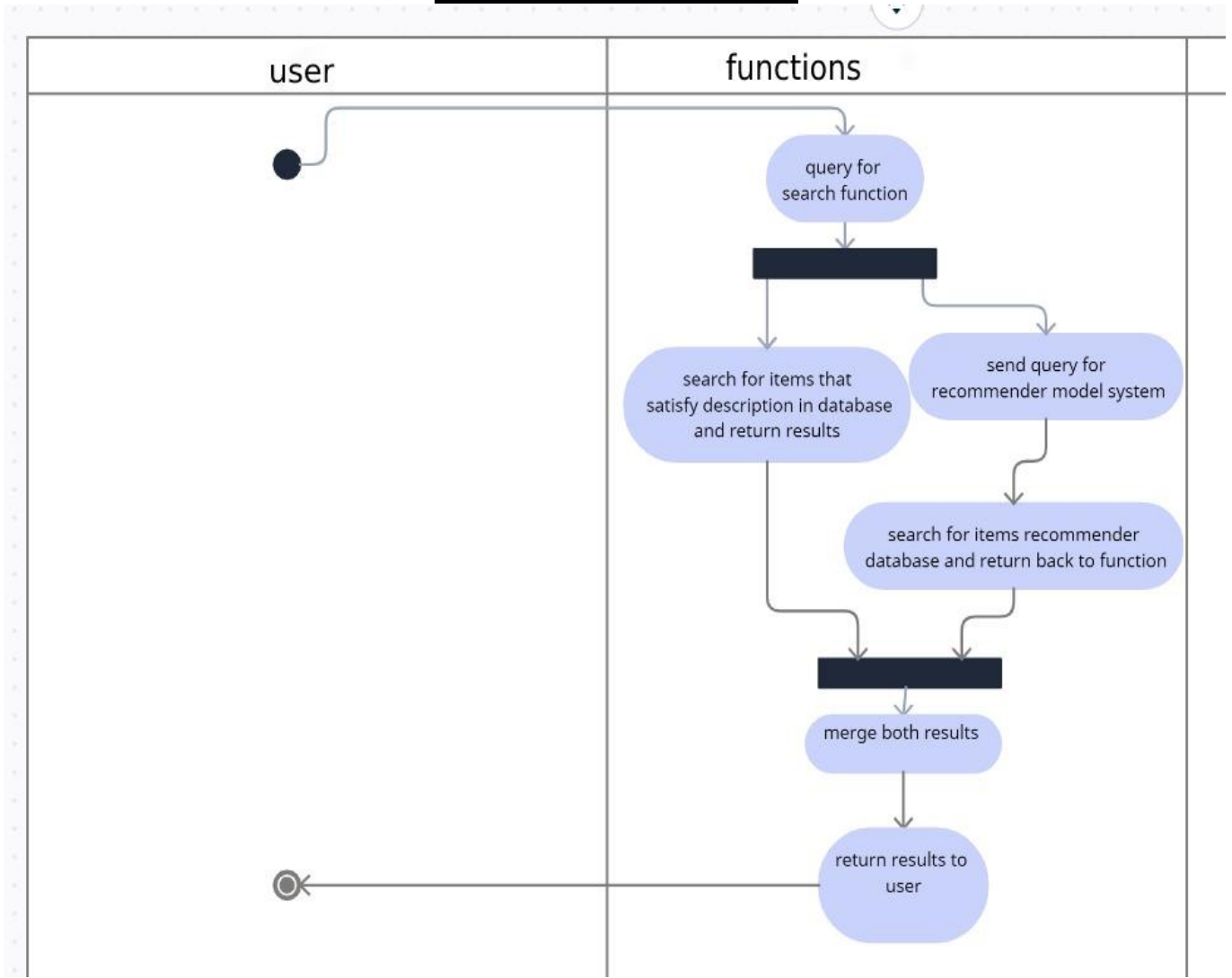
		Name	Duration	Start	Finish
1		Book Recommender System	99 days?	25/8/22 8:00 AM	10/1/23 5:00 PM
2		Project Initiation	9 days?	31/8/22 8:00 AM	12/9/22 5:00 PM
3		Project charter	3 days?	31/8/22 8:00 AM	2/9/22 5:00 PM
4		Research	4 days?	2/9/22 8:00 AM	7/9/22 5:00 PM
5		Guidelines	4 days?	7/9/22 8:00 AM	12/9/22 5:00 PM
6		Project Planning	10 days?	12/9/22 8:00 AM	23/9/22 5:00 PM
7		Requirement Gathering	6 days?	12/9/22 8:00 AM	19/9/22 5:00 PM
8		Create SRS	5 days?	19/9/22 8:00 AM	23/9/22 5:00 PM
9		Project Execution	27 days?	3/10/22 8:00 AM	8/11/22 5:00 PM
10		Task Managemet	6 days?	3/10/22 8:00 AM	10/10/22 5:00 PM
11		KPI'S Management	7 days?	11/10/22 8:00 AM	19/10/22 5:00 PM
12		Task review	7 days?	20/10/22 8:00 AM	28/10/22 5:00 PM
13		Chart/Diagram Updates	7 days?	29/10/22 8:00 AM	8/11/22 5:00 PM
14		Project Implementation	15 days?	9/11/22 8:00 AM	29/11/22 5:00 PM
15		Installation of Project	8 days?	9/11/22 8:00 AM	18/11/22 5:00 PM
16		Configuration of Project	7 days?	19/11/22 8:00 AM	29/11/22 5:00 PM
17		Testing of Project	22 days?	10/12/22 8:00 AM	10/1/23 5:00 PM
18		Unit Testing	7 days?	30/12/22 8:00 AM	9/1/23 5:00 PM
19		Run Testing	7 days?	10/12/22 8:00 AM	20/12/22 5:00 PM
20		Review Testing	6 days?	21/12/22 8:00 AM	28/12/22 5:00 PM
21		Project Closure	8 days?	30/12/22 8:00 AM	10/1/23 5:00 PM
22		Final Documenatation	8 days?	30/12/22 8:00 AM	10/1/23 5:00 PM
Book Recommender System - page1					



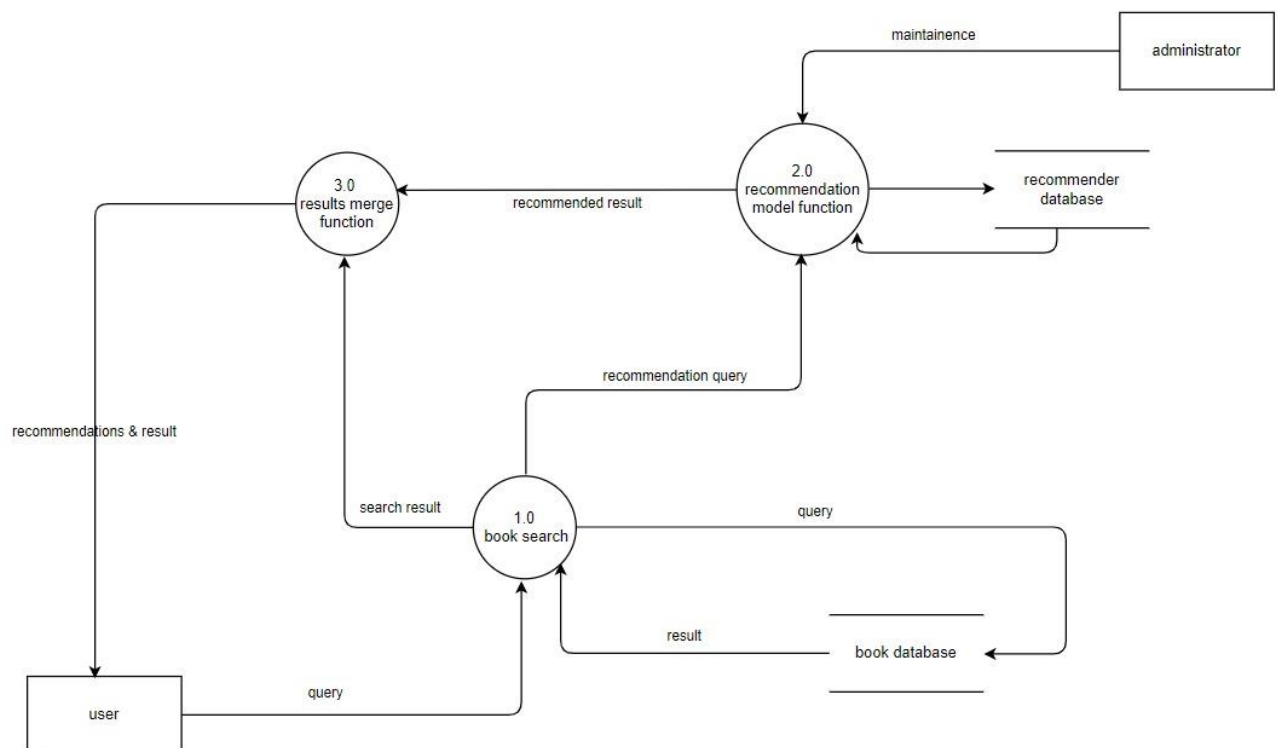
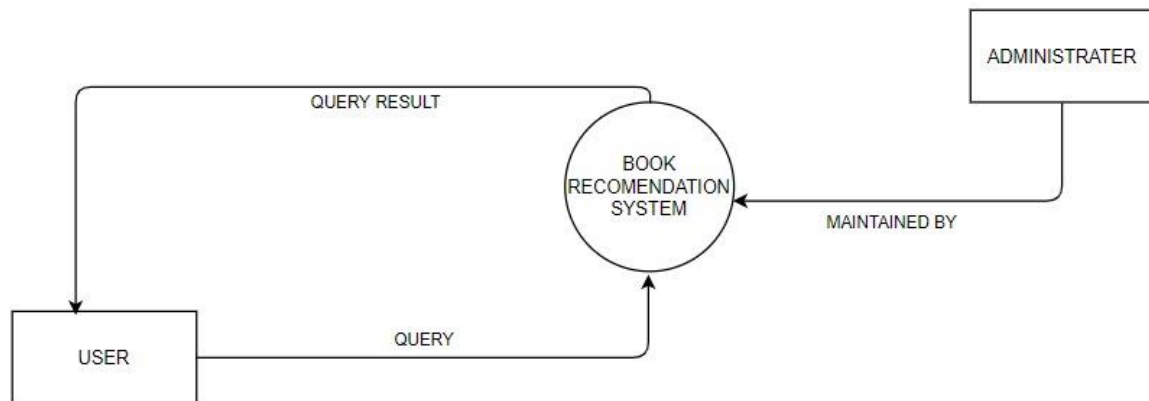
Use-Case diagram



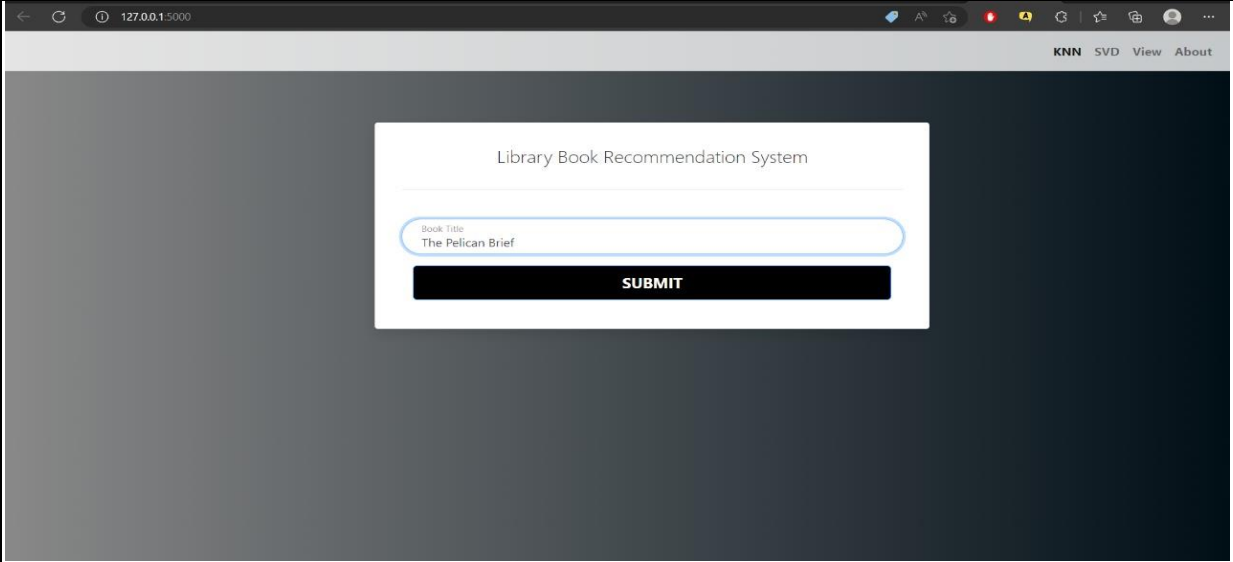
Swimlane diagrams



Data Flow Diagrams –Level 0, Level 1



User Story Cards

#0001	Book Recommendation form using KNN	KNN#1
As a user, I want to enter the name of the book so that I can get the name of the recommended books using KNN		
		

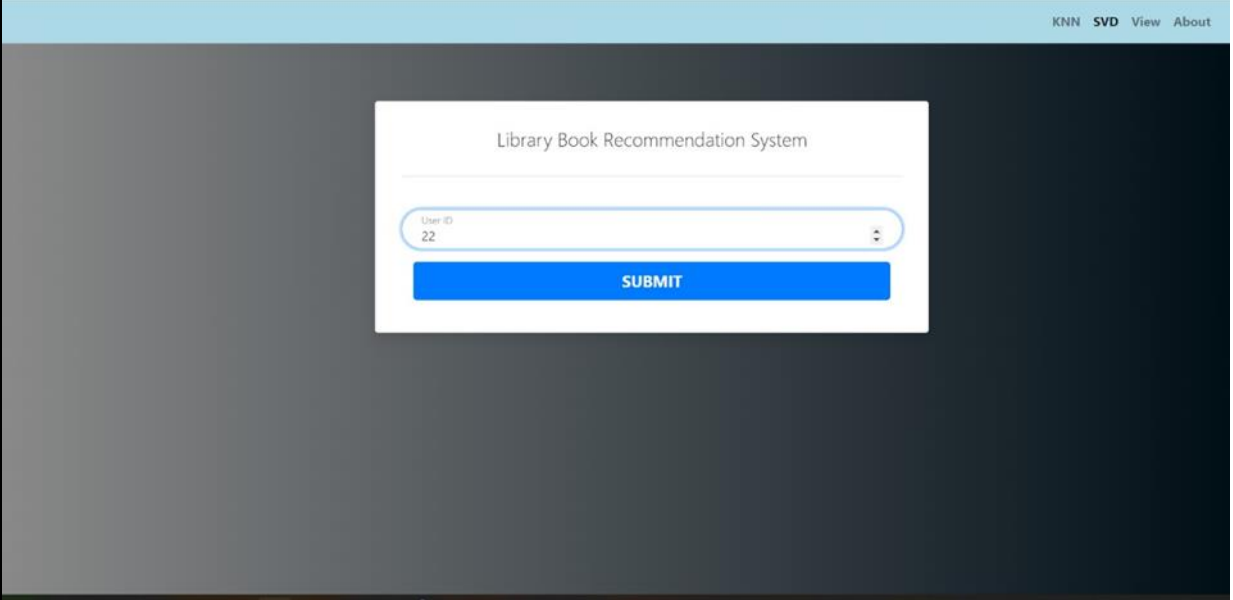
Book search using KNN

Success

- If appropriate name entered in the dialog box then the you are redirected to results Page .

Failure

- Redirected to error Page.
- Showing where the code failed .
- If there is spelling mistake in the Book title

#0002	Book Recommendation form using SVD	SVD#1
As a user, I want to enter my user Id so that I can get the name of the recommended books for me using SVD		
		

Book Recommendation using KNN

Success

- If eligible user id is entered in the dialog box then the you are redirected to results Page .

Failure

- Redirected to error Page , if user id does not exist .
- Showing where the code failed .

Software Requirements Specification

For

Book Recommender System

Version 1.0 approved

Prepared by Vasu Verma (102003142)

Aryaman Kalia (102003099)

Rajneesh Verma (102183032)

Thapar Institute of Engineering & Technology

15 September 2022

UCS503- Software Engineering Lab

Copyright © 1999 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction	3
1.1 Purpose	3
1.2 Document Conventions	3
1.3 Intended Audience and Reading Suggestions	3
1.4 Product Scope	3
1.5 References	3
2. Overall Description	4
2.1 Product Perspective	4
2.2 Product Functions	4
2.3 User Classes and Characteristics	4
2.4 Operating Environment	4
2.5 Design and Implementation Constraints	4
2.6 User Documentation	4
2.7 Assumptions and Dependencies	4
3. External Interface Requirements	5
3.1 User Interfaces	5
3.2 Hardware Interfaces	5
3.3 Software Interfaces	5
3.4 Communications Interfaces	5
4. System Features	6
4.1 System Feature 1	6
4.2 System Feature 2 (and so on)	6
5. Other Nonfunctional Requirements	8
5.1 Performance Requirements	8
5.2 Safety Requirements	8
5.3 Security Requirements	8
5.4 Software Quality Attributes	8

Revision History

Name	Date	Reason For Changes	Version
Book Recommendation System	24-08-2022		V0.1

*Software Requirements Specification for Book
Recommender system*

- **Introduction**

- **Purpose**

The goal of this document is to provide support information on the project (current version v1.0). It will attempt to explain the functionality of the program and the features it provides. It will illustrate the purpose and complete declaration for the development of system. It will also explain System constraints, interface and interactions with other external applications. This document is primarily Intended to be proposed to the customer company for their approval and a reference for development of the System.

- **Document Conventions**
- **Entire content should be justified**
- **The table gives the idea that how the document/srs layout is going to be**

<u>FONT</u>	<u>STYLE</u>	<u>SIZE</u>
Headings Times New Roman	HEADING Bold	Heading 18
Sub-Heading Times New Roman	Sub-Heading Bold	Sub-Heading 14
Other's Arial	Other's Regular	Other's 12

- **Intended Audience and Reading Suggestions**

This Software Requirements document is intended for:

1. Developers who can review project's capabilities and more easily understand where their efforts should be Targeted to improve or add more features to it (design and code the application - it sets the guidelines for Future development).
2. Project testers can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testing becomes more methodically organized.
3. End users of this application who wish to read about what this project can do.

- **Product Scope**

Book Recommender is a book recommendation system, which provides users books which they may like, based on the books that they previously saw. Every logged in user should have access to the Recommender system. The system will go through the books that user previously saw and rated, then According to those information it should provide books to the user. The project's main aim is to provide Accurate book recommendations to the user. This project is beneficial for the users and the companies. For users, they

may find books that they may like without consuming time and even they can encounter new books which they like from the recommendations. For the company, they make the website more attractive, so they draw more users to the website and the system makes the users of the website spend more time online.

- **References**

<https://www.crio.do>

<https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>.

- **Overall Description**

- **Product Perspective**

In this hustling world, entertainment is a necessity for each one of us to refresh our mood and Energy. Entertainment regains our confidence for work and we can work more enthusiastically. For Revitalizing ourselves, we can listen to our preferred audiobooks or can read books of our choice. For Watching favorable books online we can utilize movie recommendation systems, which are more Reliable, since searching of preferred books will require more and more time which one cannot Afford to waste. This product is a follow - on member of already existing Book recommendation System. This system will have some UI where people can go and check out the recommended Movies for them, after following some step.

- **Product Functions**

There will be two major functions which includes

1. Recommend the books, to the users, which are trending in their region in a particular week.
2. It will recommend few books based in the genre provided by the user

- **User Classes and Characteristics**

The main aim of this product is to eliminate the tedious task of searching through a plethora of books in a collection and hence save the user time by curating a list of books the user might like based on the user's preferences , hence the product is not class oriented . People of every age groups can use this product.

- **Operating Environment**

Operating environments for the book recommendation system is a listed as below.

- **Distributed database**
- **Client/server system**
- **Operating system: windows**

- **Database:** sql+database
- **Platform:** Chrome/Edge/Firefox

- **Design and Implementation Constraints**

- SQLcommands for above queries/applications
- Python language for the implementation of book recommendation system.
- HTML, CSS, JavaScript, ReactJS for the development of **UI**.

- **User Documentation**

How to use?

Set your minimum five genres in the section which asks for the same, and click on the button which says Recommend and the algorithm will provide you with a list of books.

Before you start

Login to our website adhering all the terms and conditions and you are good to go.

- **Assumptions and Dependencies**

One assumptions is that the applications is used on a computer or a mobile phone with enough performance ability, and the use of an up-to-date internet browser.

- **External Interface Requirements**

- **User Interfaces**

The user interface for the software shall be compatible to any browser such as internet explorer , Mozilla or Google Chrome by which user can access to the system .The user interface shall be implemented using any tool or software package like Java Applet.

- **Hardware Interfaces**

Since the applications must run over the internet, this brings out the requirement of a network interface on the device. User should have a device with valid internet connections, Wi-Fi or 4G.

- **Software Interfaces**

- The data of the project is organized in a relational database as it makes it easier to curate data with a large number of attributes
- Python 3.9.6 will be used as the predominant programming language for this project .Since the project is based on Machine Learning Python is the clear as it has built in modules for recommendation model e.g. SKIKIRLearn

.

- **Communications Interfaces**

The book recommender system shall use the HTTP protocol for the communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

- **System Features**

This project includes sorting of different books according to the mood of the user which is based on the past experiences of the user, it is based on different past preferences of the user and what kind of films user would like to read. We only implement a simple user interface for showing system recommendations. In main system user logs in first and starts giving options of books which are created on our recommendation service data. The output is shown in main application interface.

- **SORTING OF BOOKS**

- ..1 **Description and Priority**

Recommending books on past preferences of the user. High priority is given to preferences of the user.

- ..2 **Stimulus/Response Sequences**

GENERATE DATA: User can get book recommendation on the device. Book information will be collected according time of action, mood, author and rating. The database will be filled with info.

RECOMMENDATION: It can suggest book as recommendation based on data set ny user's approach. The main function will show books on recommendation algorithm. When a user will choose movie recommendation the system will give it recommend books through their past experiences.

- ..3 **Functional Requirements**

1. Applicants dashboard page
2. A page with list of books.
3. A feature to filter and organize books based on rating, genre and other descriptions.
4. A feature to enable applicants to explore.
5. The different books based on their preferences.

Database: The database based on past preferences of the user will be created.

Algorithm: It will capture preferences and suggest books based on it.

Historical Data: It will be collected by the system.

Searching Reporting Requirement: Will tell users how to search data.

- **Other Nonfunctional Requirements**

- **Performance Requirements**

NORMALIZATION:

The basic objective of normalization is to reduce redundancy which means that information is to be stored only once. Storing information several times leads to wastage of storage space and increase in the total size of the data stored. If a database is not properly designed it can give rise to modification anomalies. Modification Anomalies arise when data is added to, changed or deleted from a database table. Similarly, in Traditional databases as well as improperly designed relational databases, data redundancy can be a problem. These can be eliminated by normalizing a database. Normalization is the process of breaking down a table into smaller tables. So that each table deals with a single theme. There are three different kinds of modifications of anomalies and Formulated the first, second and third normal forms (3NF) is considered sufficient for most practical purposes. It should be considered only after a thorough analysis and complete Understanding of its implications.

Safety Requirements:

No safety requirements has been required.

Security Requirements

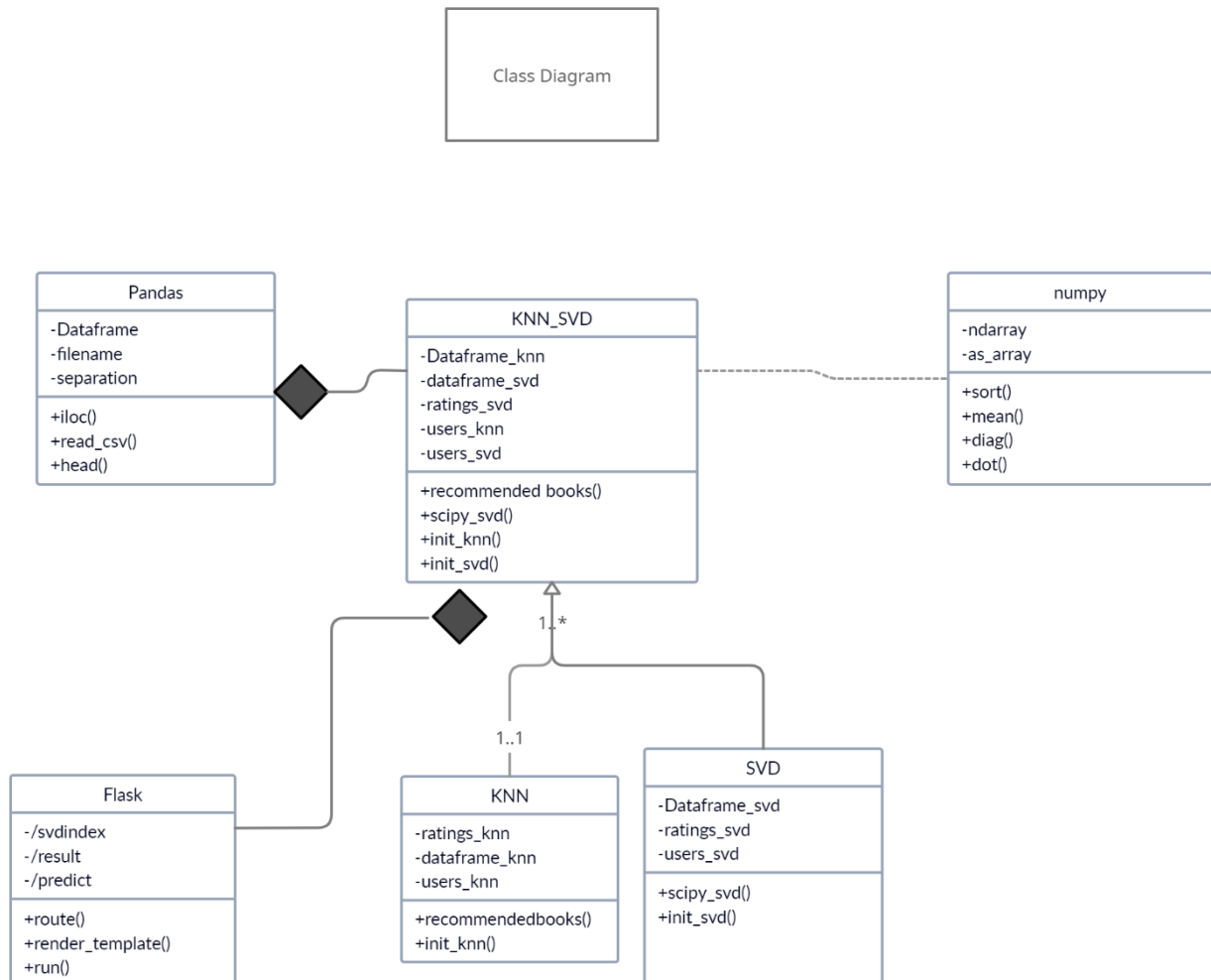
Database should be reached securely and the data should not be lost. It also should not change

Except inter - agent updates. Moreover, since our data - set contain some information of user such as tracks he / she listened, security design is important in the web service, so we must choose their database partner carefully.

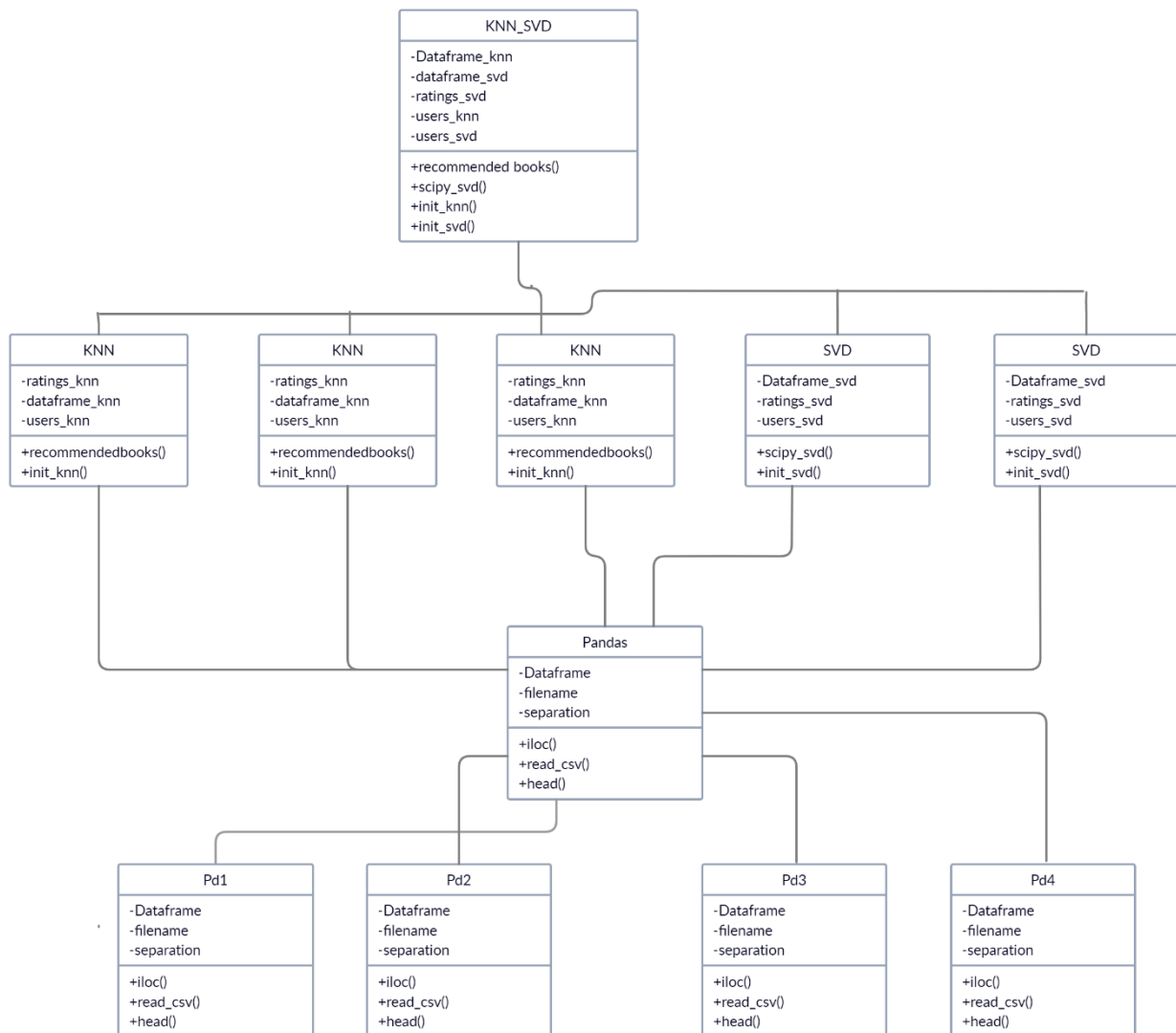
● **Software Quality Attributes**

1. **AVAILABILITY**: The site should be available on the specified date and specified time as many users are doing their search.
2. **CORRECTNESS**: The site will give proper recommendation based on ones's view.
3. **MAINTAINABILITY**: The administrators and back-end developers should maintain correct output of recommendations.
4. **USABILITY**: The site should satisfy a maximum number of users needs.

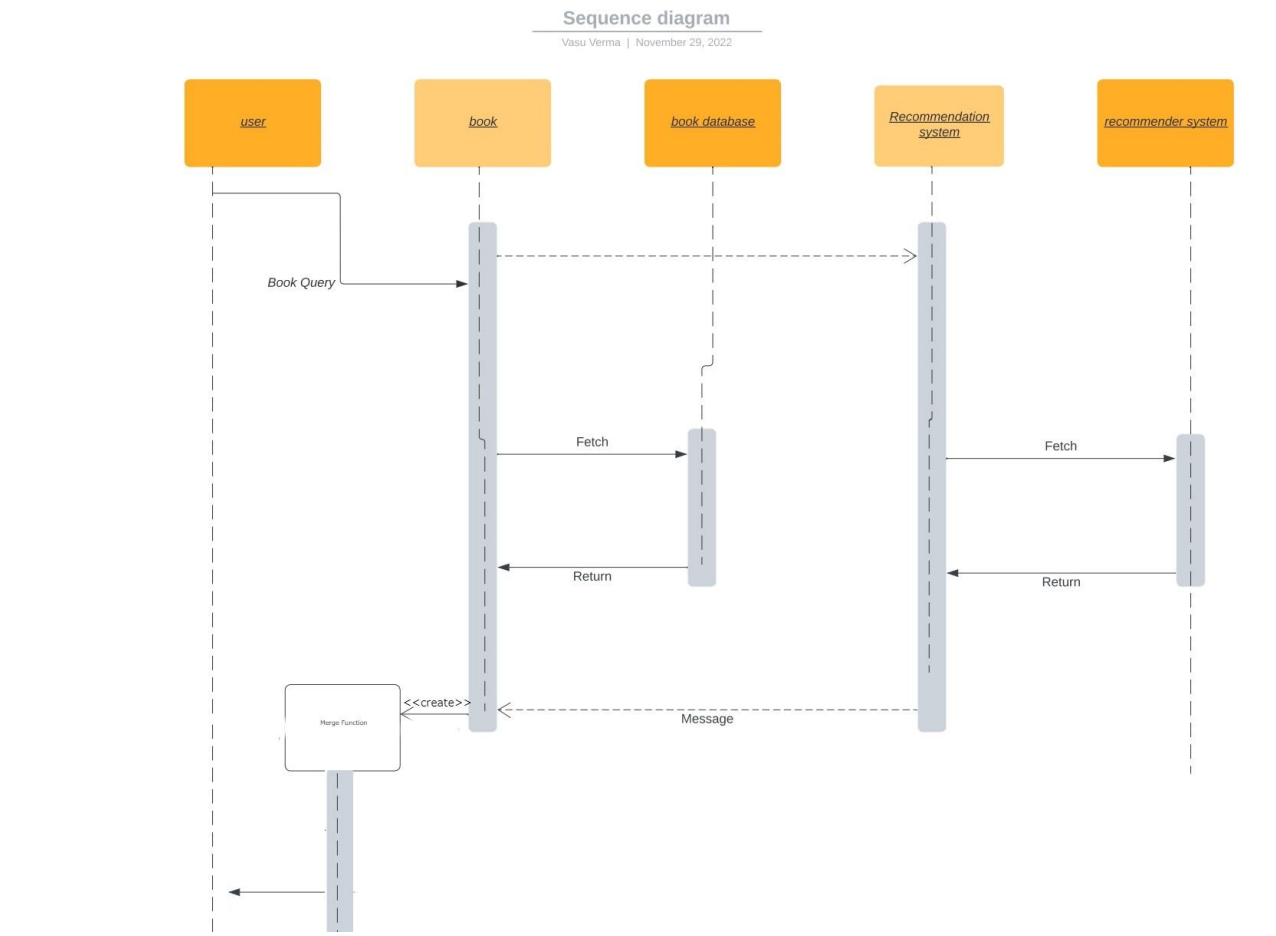
Class Diagram



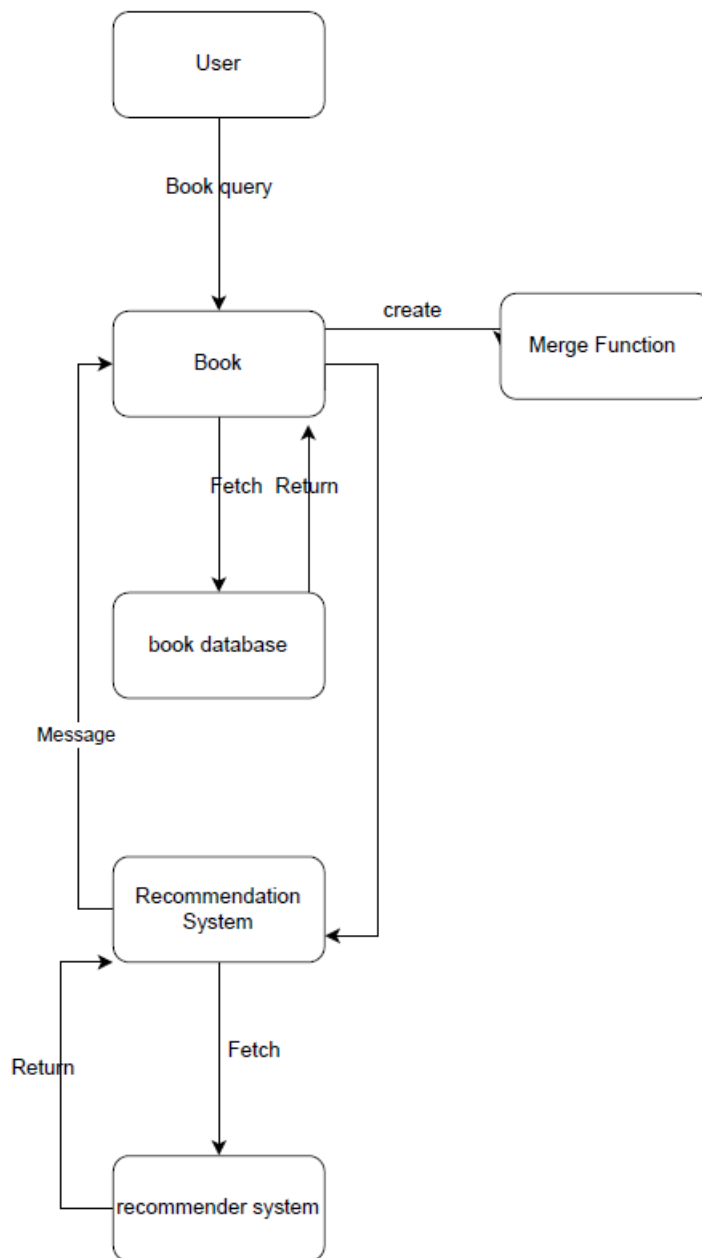
Object Diagram



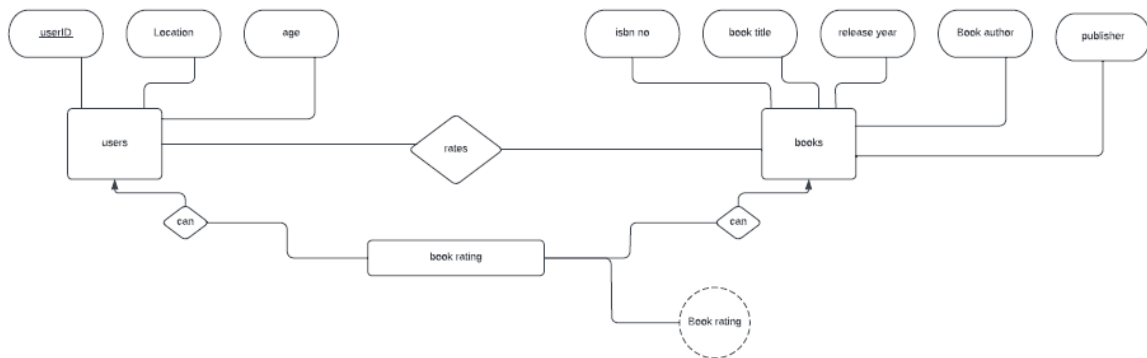
Sequence Diagram



Collaboration Diagram



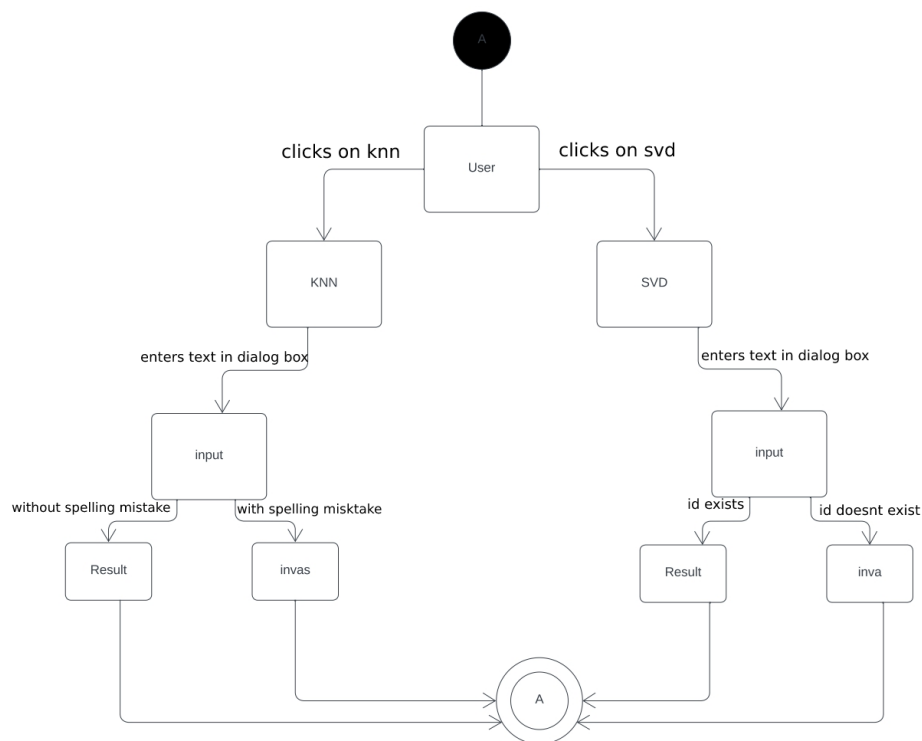
Database Diagram: ER Diagram



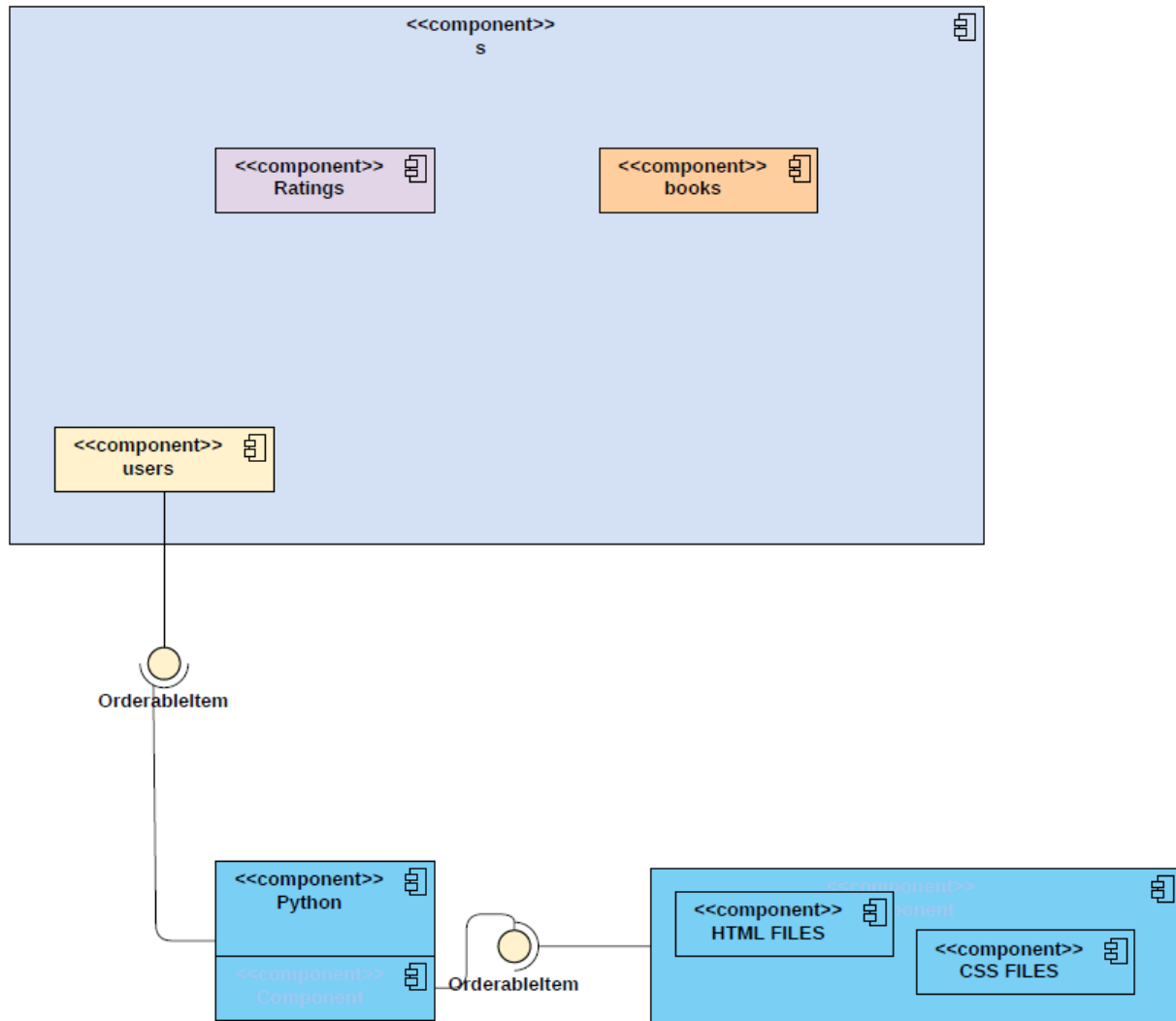
State Chart Diagram

State Chart

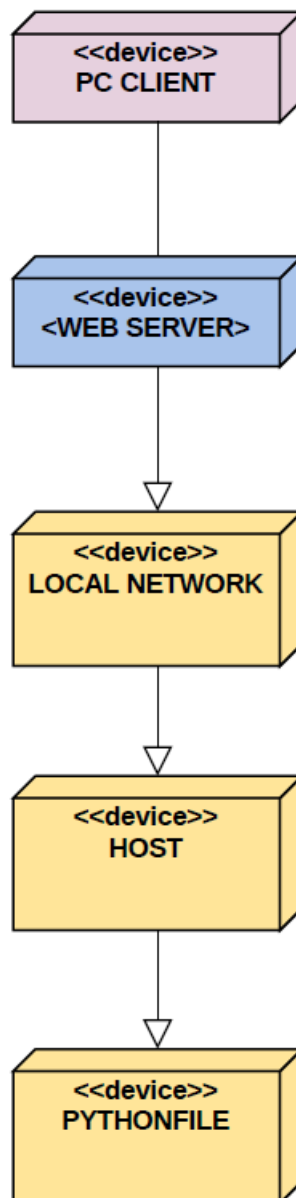
Vasu Verma | November 29, 2022



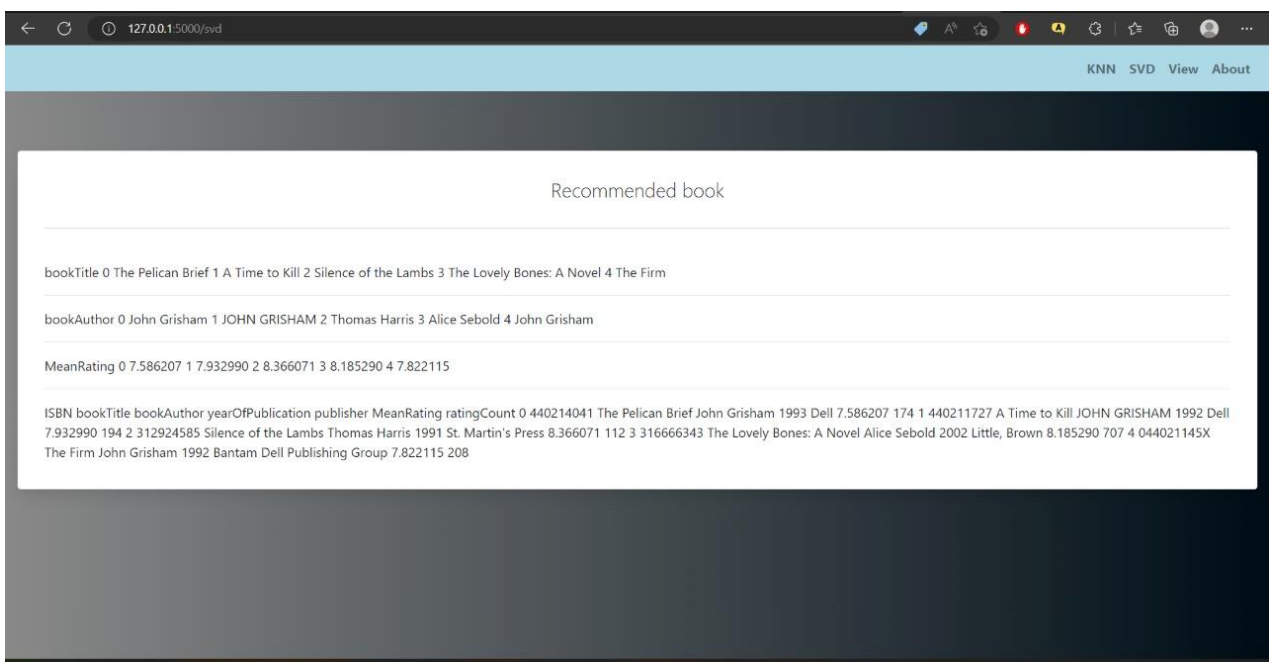
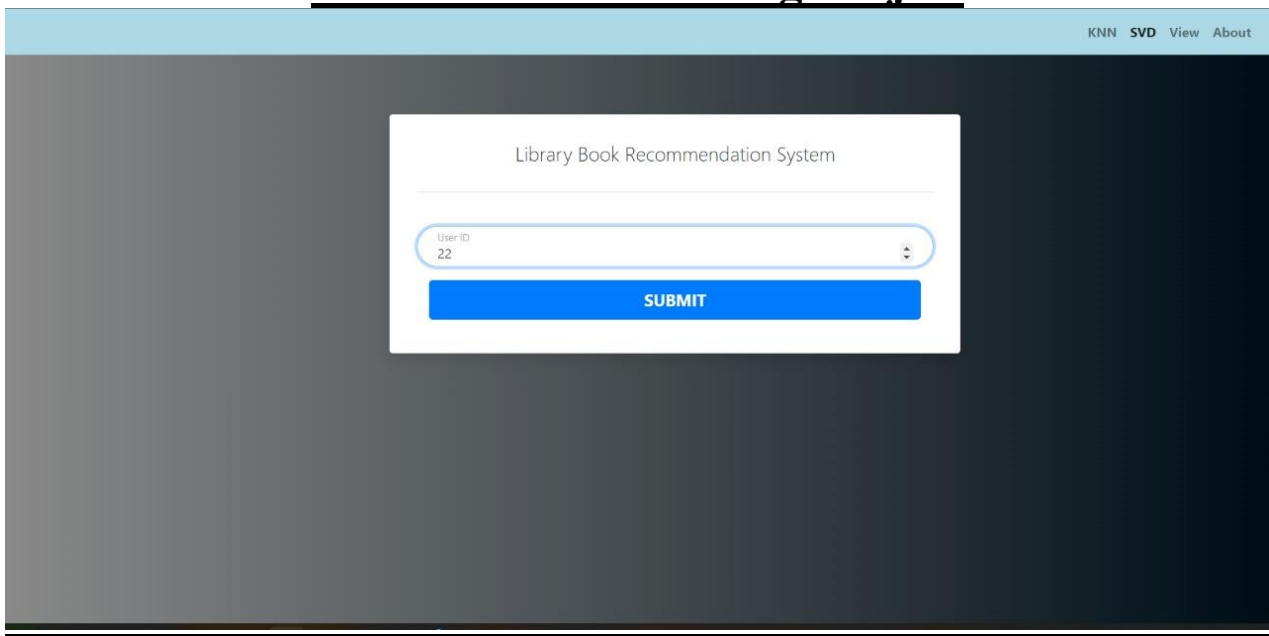
Component Diagram



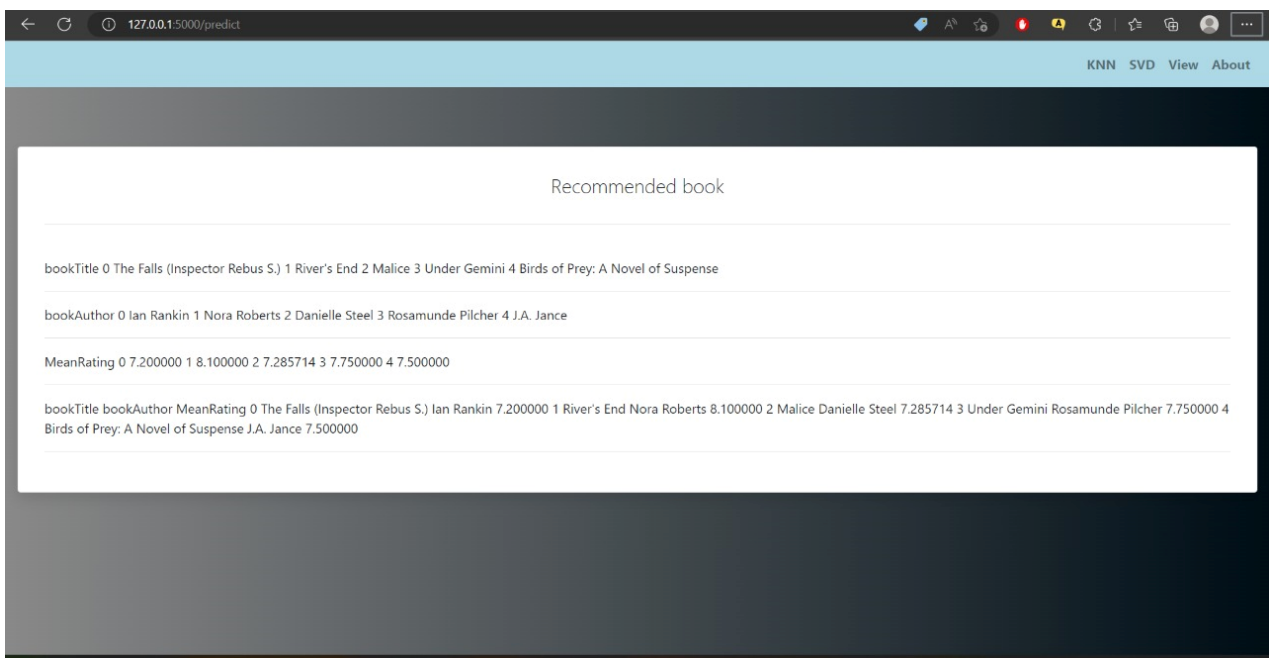
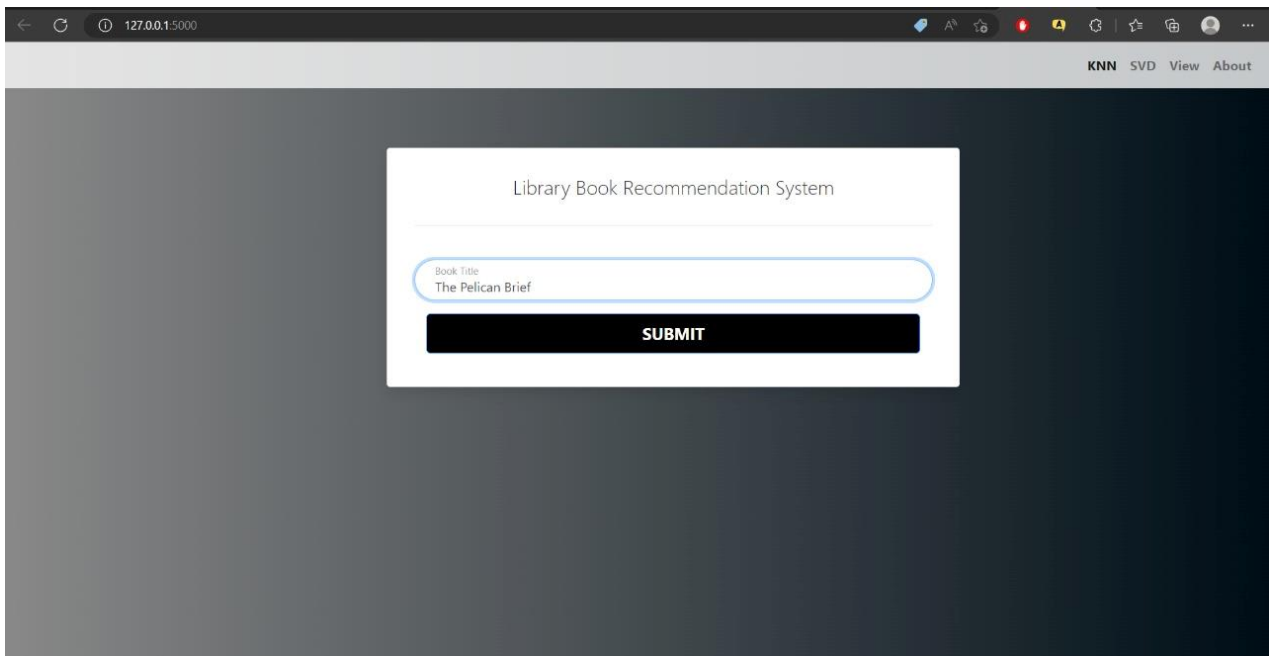
Deployment Diagram



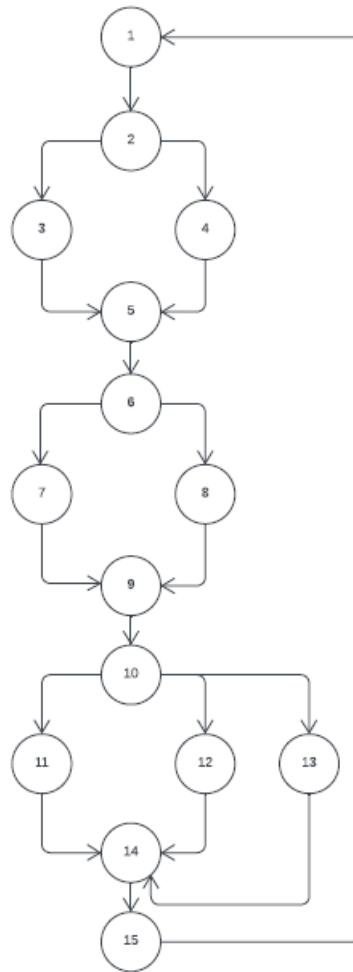
Screenshots of Working Project



Software Requirements Specification for Book Recommender system



Cyclomatic Complexity (All modules)



Since the number of regions are six the cyclometric complexity of our module is six

Test Cases

Test Case: 1

features

Test Case Name: drawing

System: Book recommendation system

Subsystem: KNN Search

Designed by: Aryaman Kalia

Design Date: 20-11-22

Executed By: Aryaman Kalia

Execution Date: 21-11-22

Short Description: to check recommendation on the base of Book entered

Pre-conditions

1. User has the software installed
2. Has a latest browser
3. No files should be missing

Step	Action	Expected System Response	Pass/ Fail	Comment
1.	Go to http://127.0.0.1:5000/	Get request is issued and in reciprocation Post request is sent	Pass	Fine
2.	Enter “The Hamlyn complete knitting course” in the Book title option	For this KNN request is issued and is shown by the “loading” on the top	Pass	None
3.	Waiting for the Result	The Webpage appears with the results of “Book Title , Author , Mean Distance”	Pass	None

Test Cases

Test Case: 2

Test Case Name: SVD

Recommendation

System: Book Recomender System

Subsystem: SVD

Designed by: Vasu Verma

Design Date: 20-11-22

Executed By: Vasu Verma

Execution Date: 21-11-22

Short Description: Recommendations on base of user id

Pre-conditions

1. User has the software installed
2. Has a latest browser
3. No files should be missing

Post Conditions:

NONE

Step	Action	Expected System Response	Pass/ Fail	Comment
1.	Go to http://127.0.0.1:5000	Get request is issued and in reciprocation Post request is sent	Pass	NONE
2.	Go to top right option where SVD is written	The page http://127.0.0.1:5000/svdIndex Is opened	Pass	NONE
3.	Enter User id in the dialog box	After waiting for the results the results page appear with Book recommendations.	Pass	NONE

Test Case: 3

Test Case Name: Spelling mistake

System: Book Recommendation system using KNN **Subsystem:** KNN

Designed by: Rajneesh

Design Date: 20-11-22

Executed By: Rajneesh

Execution Date: 21-11-22

Short Description: what happens when the book title is entered wrong

Pre-conditions

1. User has the software installed
2. Has a latest browser
3. No files should be missing

Post Conditions:

NONE

Step	Action	Expected System Response	Pass/ Fail	Comment
1.	Go to http://127.0.0.1:5000/	Get request is issued and in reciprocation Post request is sent	Pass	NONE
2.	Enter “The Hhamlyn complete knitting course” in the Book title option	For this KNN request is issued and is shown by the “loading” on the top .After waiting for the results the results page appear with Book recommendations.	Fail	Error message should be shown

Test Case Report

Executed Test Cases: 3

Failed Test Cases: 1

Passed Test Cases: 2

Functions	Description	%TCs Executed	%TCs Passed	TCs Pending	Priority
1. KNN function	to check recommendation on the base of Book entered.	3	3	0	HIGH
2.SVD function	Recommendations on base of user id	3	3	0	HIGH
3 spelling mistake in KNN	what happens when the book title is entered wrong	2	1	0	HIGH