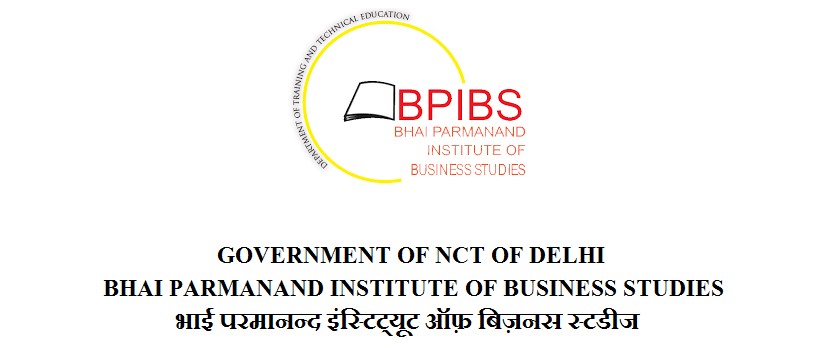
**THREAT MODELLING**

**Submitted in Partial Fulfilment of the Requirements for the Award of the Degree of**

**MASTER OF COMPUTER APPLICATIONS (M.C.A)**



# *Submitted By*

**POOJA MAURYA**

Enrollment No: 03111404419

# Guided By

Mr. Pravir Chitre Mr. Ramar Bose

Assistant Professor Edunet Foundation



**DE**

**Department of computer science and application**

**Bhai Parmanand Institute of Business Studies**

**Guru Gobind Singh Indraprastha University, Delhi**

**Session: (2019-2022)**

**GOVERNMENT OF NCT OF DELHI**

**BHAI PARMANAND INSTITUTE OF BUSINESS STUDIES**

**Opp. Madhuban Colony: Shakarpur Extn. Delhi-110092 Email: bpibs.delhi@nic.in Phone: 22543891, 22017393 Fax: 22016134**

# 

# Certificate by the Head of Department

This is to certify that this project report entitled **“BookStore- An online book store with recommendations**” is submitted as a diligent work of **Pooja (Enrollment No. – 03111404419)**, who is undergoing 6th Semester Industrial training at **Edunet Foundation**. She has been given regular updates about the training work throughout the semester and has prepared the training report under the guidance of the assigned guide.

## Date: Dr. Girish Sharma

**(Professor & Head of MCA**

**Department)**

**GOVERNMENT OF NCT OF DELHI**

**BHAI PARMANAND INSTITUTE OF BUSINESS STUDIES**

**Opp. Madhuban Colony: Shakarpur Extn. Delhi-110092 Email: bpibs.delhi@nic.in Phone: 22543891, 22017393 Fax: 22016134**

# 

# Certificate by the Internal Guide

This is to certify that this project report entitled **“BookStore- An online book store with recommendations**” is submitted as a diligent work of **Pooja (Enrollment No. – 03111404419)**, who is undergoing 6th Semester Industrial training at **Edunet Foundation**. She has been given regular updates about the training work throughout the semester and has prepared the training report under my guidance.

## Date: Project Guide

**Mr. Pravir Chitre**

**(Assistant Professor)**

**GOVERNMENT OF NCT OF DELHI**

**BHAI PARMANAND INSTITUTE OF BUSINESS STUDIES**

**Opp. Madhuban Colony: Shakarpur Extn. Delhi-110092 Email: bpibs.delhi@nic.in Phone: 22543891, 22017393 Fax: 22016134**

## 

## CANDIDATE’S DECLARATION

I hereby declare that the work which is being presented in this project work entitled “**BookStore- An online book store with recommendations**” in partial fulfillment of the requirements for the award of the degree of Master in Computer Applications at Bhai Parmanand Institute of Business Studies, is an authentic record of my own work carried out during the period November to May 2022 at Edunet Foundation under the supervision and guidance of Mr. Ramar Bose (AI/ML Instructor).

I assure you that I have not submitted the matter embodied in this project work anywhere for the fulfillment of any degree or diploma.

Date: Pooja (03111404419)

New Delhi Signature

The Principal

BPIBS,

Shakarpur,

Delhi-92

**UNDERTAKING- MCA DISSERTATION**

I **Sudhanshu Talwar** Roll No. **004711404417** may kindly be sponsored to **NCR Corporation Gurgaon**, for Practical Training- Software/ Project Preparation.

I here by undertake that:

* 1. I will abide by rules of the organization and observe total discipline as per their conditions.
  2. I will follow all the rules, regulations and ordinances of the GGS Indraprastha University, Delhi.
  3. I will prepare and submit the duly completed Software/Project in the institute along with requisite reports well in time.
  4. I will arrange to remain present in the institute for presentation discussion and viva as per schedule of internal/ external examination.

Yours faithfully

Date: May 15 2020

Signature (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

Name: SUDHANSHU TALWAR

Complete Address:

D-117 A South Anarkali, Street No.-2

Delhi-110051

Tele. No.: +91 9953046167

### PROFORMA OF CERTIFICATE FOR PROJECT

This is to certify that is a bonafide record of the project work done satisfactorily at **Edunet Foundation** by Ms. **Pooja Registration No. 03111404419** in practical fulfillment of MCA 6th Semester Examination.

This report or a similar report on the topic has not been submitted for any other examination and does not form part of any other course undergone by the candidate.

SIGNATURE( )

(Candidate)

PLACE: New Delhi NAME: Pooja

DATE: May 2022 ROLL NO: 03111404419

SIGNATURE( )

(Project Guide)

PLACE: New Delhi NAME: Mr. Pravir Chitre

DATE: DESIGNATION: Ass. Professor

ADDRESS:Opp. Madhuban Colony, Near Vikas

Marg, Shakarpur,

New Delhi, Delhi 110092

Name & Seal of the Institution

## ACKNOWLEDGEMENT

Itis my proud privilege to express my profound gratitude to the entire management of Bhai Parmanand Institute of Business Studies & faculty members of the institute for providing me with the opportunity to avail the excellent facilities and infrastructure. The knowledge & values inculcated in me during the academic session 2019-2022 have proved to be of immense help at the starting of my career. **I am very thankful to our Principal Mr. Girish Sharma for providing us an excellent infrastructure & experienced faculty members with immense knowledge at BPIBS.**

I am grateful to my project guide Assistant Prof. **Mr. Pravir Chitre** for their astute guidance, constant encouragement & sincere support for this project work. Without them it was impossible to understand the use cases of **BookStore- An online book store with recommendations**.

I would like to thank our Training & Placement Officer **Mr. Kaushal Mehta** for arranging placement drives . I would like to thank Edunet Foundation, for providing me with an opportunity to pursue my Artificial Intelligence training. I feel pride and privileged in expressing my deep sense of gratitude to all those who have helped me in presenting this assignment.

I express my sincere gratitude to **Mr. Ramar Bose** for his inspiration, constructive suggestion, mastermind analysis and affectionate guidance in my work, without which this project work completion would have been impossible for me.

At the end, I am really thankful to all the faculty members of BPIBS & my colleagues . Without their support & believe, I couldn’t reach the place where I am today. I am feeling overwhelming today.

## ABSTRACT

A book recommendation system is a type of recommendation system where we have to recommend similar books to the reader based on his interest. The books recommendation system is used by online websites which provide e-books like google play books, open library, goodReads, etc.

In this Project we will be using this book recommendation system in an online book store website where users can purchase/rent books and based on their purchase history  books will be recommended to him/her.

Recommendation engines are a subclass of machine learning which generally deal with ranking or rating products / users. Loosely defined, a recommender system is a system which predicts ratings a user might give to a specific item. These predictions will then be ranked and returned back to the user.

Online Book store is an online web application where the customer can purchase books online. Through a web browser the customers can search for a book by its title or author, later can add to the shopping cart and finally purchase using credit card transaction. The user can login using his account details or new customers can set up an account very quickly. They should give the details of their name, contact number and shipping address. The user can also give feedback to a book by giving ratings on a score of five. The books are divided into many categories based on subject like Software, Database, English, Architecture etc.

## Table of Contents

|  |  |  |
| --- | --- | --- |
| **Serial No.** | **Description** | **Page No.** |
| **1** | **Chapter-1 Introduction** |  |
|  | **1.1 Problem Statement**  **1.2 Proposed Solution**  **1.3 Project Milestones & Deliverable** |  |
| **2** | **Chapter-2 Project Description** |  |
|  | **2.1 System Interface**  **2.2 System Specification**  **2.3 Methodology used**  **2.4 Platforms/Technology Used**  **2.5 Building blocks of Threat Modelling**  **2.6 Entities involved in Threat Modelling**  **2.7 Use Case description**  **2.8 Risks, Assumptions & Dependencies** |  |
| **3** | **Chapter-3 System Diagrams** |  |
|  | **3.1 Use Case diagram**  **3.2 Process Flow Diagram**  **3.3 Data structure/tables used**  **3.4 Class Diagram**  **3.5 State Transition Diagram**  **3.6 PFD of a threat modelled application** |  |
| **4** | **Chapter-4 UI Design & Screen shots** |  |
|  | **4.1 UI Design**  **4.2 Email**  **4.3 Website screenshot** |  |

|  |  |  |
| --- | --- | --- |
| **5** | **Chapter-5 Testing & Deployment** |  |
|  | **5.1 Testing**  **5.2 Deployment** |  |
| **6** | **Chapter-6 Conclusion** |  |
|  | **6.1 Conclusion**  **6.2 Limitations**  **6.3 Future Scope of Project**  **6.4 Useful links**  **6.5 References** |  |

### PREFACE

This dissertation report provides the detailed description of industrial training taken in Edunet Foundation. It includes detailed description of technologies on which the Artificial Intelligence trainee has worked upon during the training period November, 2021 to May 2, 2022.

The following is a brief description of the points that are covered & discussed in each chapter:

**Chapter 1** is about the introduction problem statement, proposed solution & project milestones & deliverables.

**Chapter 2** includes the project description where system interface, hardware & software requirements, software development methodology used, platform/technology used in Threat Modelling, building blocks of Recommendation System, entities involved in Recommendation system ,Use Case Description ,Risks, Assumptions & dependencies are discussed in detailed manner.

**Chapter 3** includes system diagrams Use Case diagram, Tables/Data Structure used, process flow diagram, Bookstore applied sample application process flow diagram.

**Chapter 4** includes UI design & screenshots of backend table where the data is being stored, email trigger screenshots.

**Chapter 5** includes Testing & deployment, different levels of testing performed in the project along with respective entities, how deployment activity is carried out.

**Chapter 6** concludes the project, limitations of project, future scope of the project, useful resource links & references.

**Chapter 1**

### Introduction

In this chapter

1.1 Problem Statement

1.2 Proposed Solution

1.3 Project Deliverables & Milestones.

#### 

#### 1.1 Problem Statement

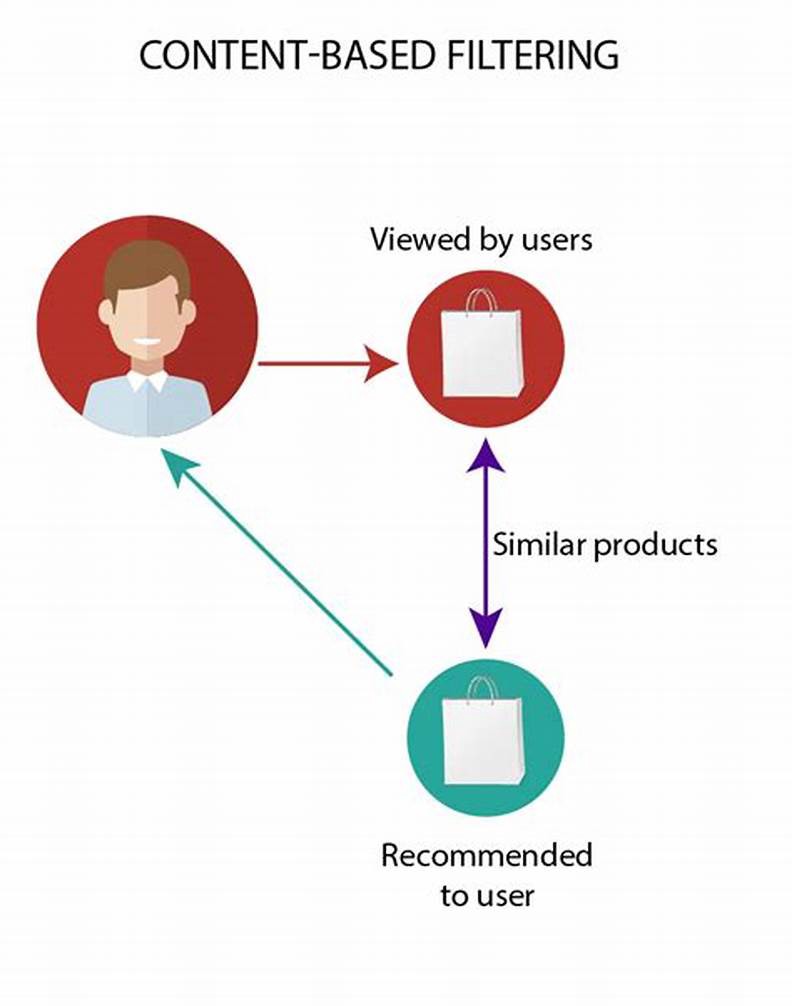
#### 1.2 Proposed Solution

**1.3.1 What is Recommendation System?**

A recommendation system is a subclass of Information filtering Systems that seeks to predict the rating or the preference a user might give to an item. In simple words, it is an algorithm that suggests relevant items to users. Eg: In the case of Netflix which movie to watch, In the case of e-commerce which product to buy, or In the case of kindle which book to read, etc.

**1.3.2 Types:**

* **Content based Filtering:**

****

In this type of recommendation system, relevant items are shown using the content of the previously searched items by the users. Here content refers to the attribute/tag of the product that the user like. In this type of system, products are tagged using certain keywords, then the system tries to understand what the user wants and it looks in its database and finally tries to recommend different products that the user wants.

Let us take an example of the movie recommendation system where every movie is associated with its genres which in the above case is referred to as tag/attributes. Now let assume user A comes and initially system don’t have any data about user A. so initially, the system tries to recommend the popular movies to the users or the system tries to get some information of the user by getting a form filled by the user. After some time, users might have given a rating to some of the movies like it gives a good rating to movies based on the action genre and a bad rating to the movies based on the anime genre. So here system recommends action movies to the users. But here you can’t say that the user dislikes animation movies because maybe the user dislikes that movie due to some other reason like acting or story but actually likes animation movies and needs more data in this case.

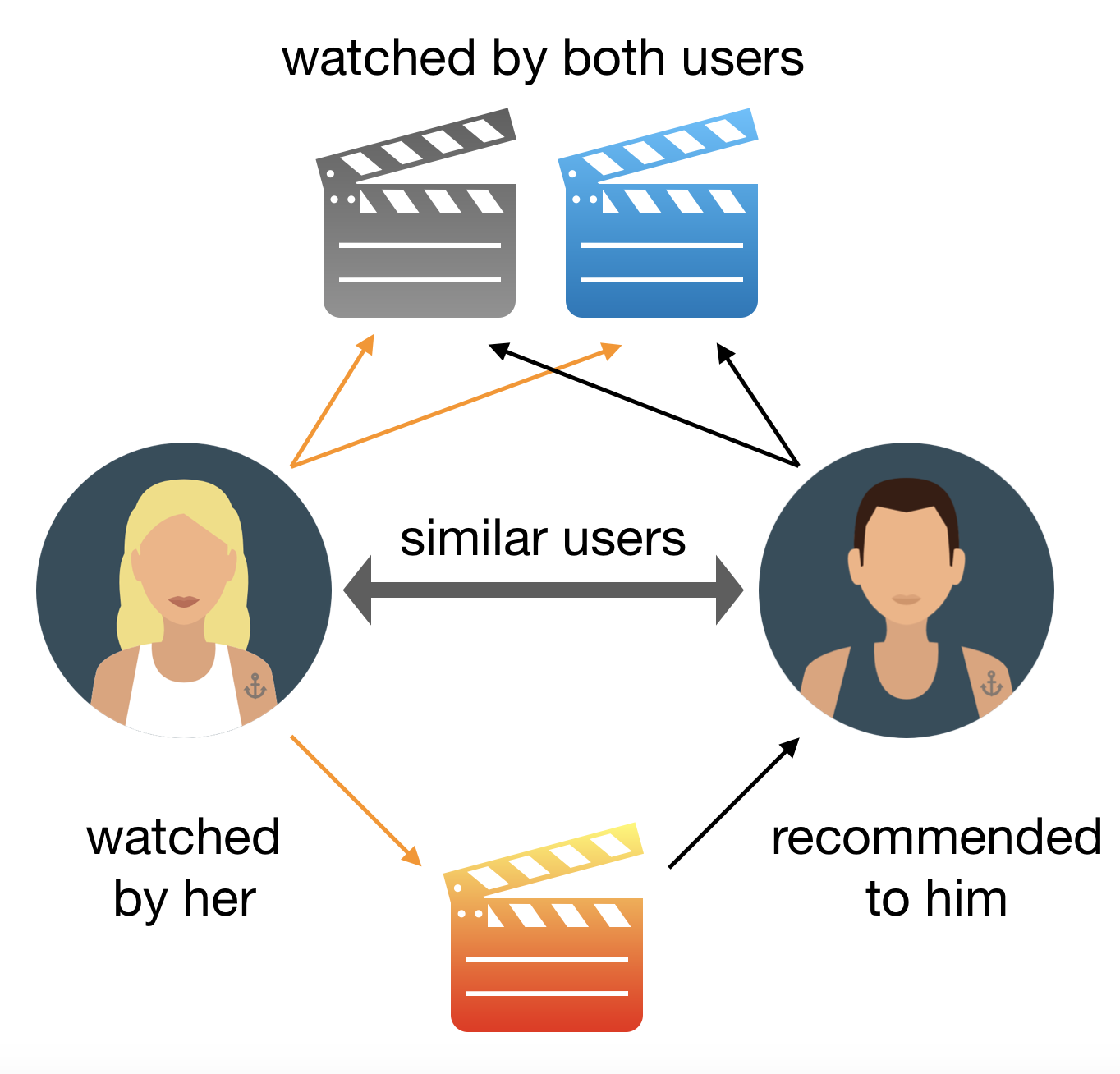
#### Advantage

1. Model doesn’t need data of other users since recommendations are specific to a single user.
2. It makes it easier to scale to a large number of users.
3. The model can Capture the specific Interests of the user and can recommend items that very few other users are interested in.

#### Disadvantage

1. Feature representation of items is hand-engineered to some extent, this tech requires a lot of domain knowledge.
2. The model can only make recommendations based on the existing interest of a user. In other words, the model has limited ability to expand on the user’s existing interests.

* **Collaborative Filtering:**

****

Recommending the new items to users based on the interest and preference of other similar users is basically collaborative-based filtering. For eg:- When we shop on Amazon it recommends new products saying *“Customer who brought this also brought”*

This overcomes the disadvantage of content-based filtering as it will use the user Interaction instead of content from the items used by the users. For this, it only needs the historical performance of the users. Based on the historical data, with the assumption that user who has agreed in past tends to also agree in future.

There are 2 types of collaborative filtering:-

### A. User-Based Collaborative Filtering

Rating of the item is done using the rating of neighbouring users. In simple words, It is based on the notion of users’ similarity.

Let see an example. On the left side, you can see a picture where 3 children named A, B, C, and 4 fruits i.e, grapes, strawberry, watermelon, and orange respectively.

Based on the image let assume A purchased all 4 fruits, B purchased only strawberry and C purchased strawberry as well as watermelon. Here A & C are similar kinds of users because of this C will be recommended Grapes and Orange as shown in dotted line.

#### B. Item-Based Collaborative Filtering

The rating of the item is predicted using the user’s own rating on neighbouring items. In simple words, it is based on the notion of item similarity.

Let us see with an example as told above about users and items. Here the only difference is that we see similar items, not similar users like if you see grapes and watermelon you will realize that watermelon is purchased by all of them but grapes are purchased by Children A & B. Hence Children C is being recommended grapes.

Now after understanding both of them you may be wondering which to use when. Here is the solution if No. of items is greater than No. of users go with user-based collaborative filtering as it will reduce the computation power and If No. of users is greater than No. of items go with item-based collaborative filtering. For Example, Amazon has lakhs of items to sell but has billions of customers. Hence Amazon uses item-based collaborative filtering because of less no. of products as compared to its customers.

#### Advantage

1. It works well even if the data is small.
2. This model helps the users to discover a new interest in a given item but the model might still recommend it because similar users are interested in that item.
3. No need for Domain Knowledge

#### Disadvantage

1. It cannot handle new items because the model doesn’t get trained on the newly added items in the database. This problem is known as Cold Start Problem.
2. Side Feature Doesn’t have much importance. Here Side features can be actor name or releasing year in the context of movie recommendation.

* **Hybrid**

**1.3.3 Benefits:**

There are many use-cases of it. Some are

A. **Personalized Content:**Helps to Improve the on-site experience by creating dynamic recommendations for different kinds of audiences like Netflix does.

B. **Better Product search experience**: Helps to categories the product based on their features. Eg: Material, Season, etc.

#### 1.3 Project Deliverables & Milestones

|  |  |
| --- | --- |
| **Milestones** | **Deliverable** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Chapter 2**

## Project Description

In this chapter

2.1 System Interface

2.2 System Specification

2.3 Methodology Used

2.4 Building blocks of Threat Modelling

2.5 Use Case description

2.6 Risks, assumptions & dependencies

**2.1 System Interface:**

**System analysis** or study is an important phase of any application development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system.

### 2.2 System Specifications

#### 2.2.1 Hardware Requirements

From Development perspective

* Operating System: Windows 10
* Processor: Core i3, 8th Generation
* RAM: 4 GB

**2.2.2 Software Requirements** From Development perspective,

* Frontend  :  HTML, CSS, BootStrap, Javascript
* Backend  : Python , Django Framework
* Database : Postgresql
* IDE Used : VSCode

### 2.3 Methodology Used

Our organization follows Agile Methodology, thus it is very important to understand Agile Methodology.

Stemming from the values and principles of the Agile Manifesto, it was created as a response to the inadequacies of traditional development methods such as Waterfall method.

**2.3.1 Agile Project Management:**

Agile Project Management is a methodology that is commonly used to deliver complex projects due to its adaptiveness. It emphasizes collaboration, flexibility, continuous improvement, and high quality results. It aims to be clear and measurable by using six main “deliverables” to track progress and create the product.

Basically there are two most popular Agile Development Life Cycle SCRUM & KANBAN. Our organization follows SCRUM Development Cycle.

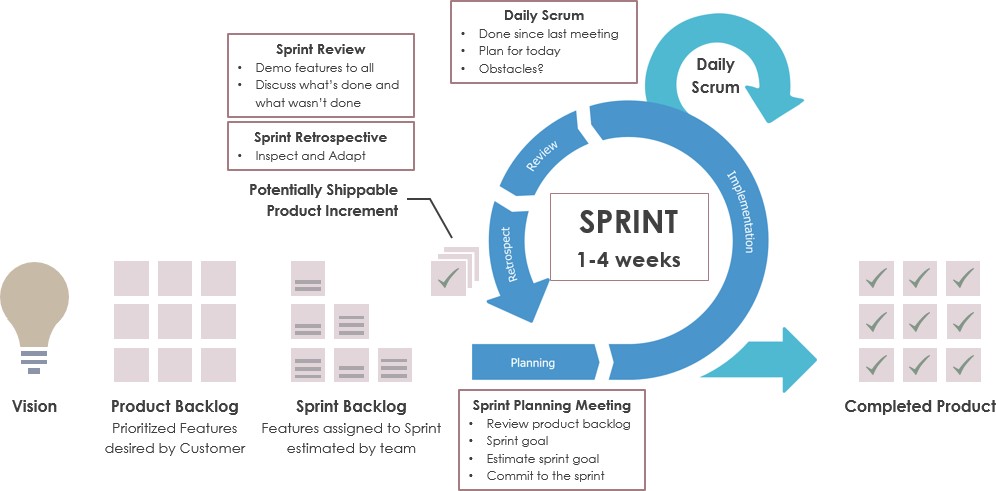
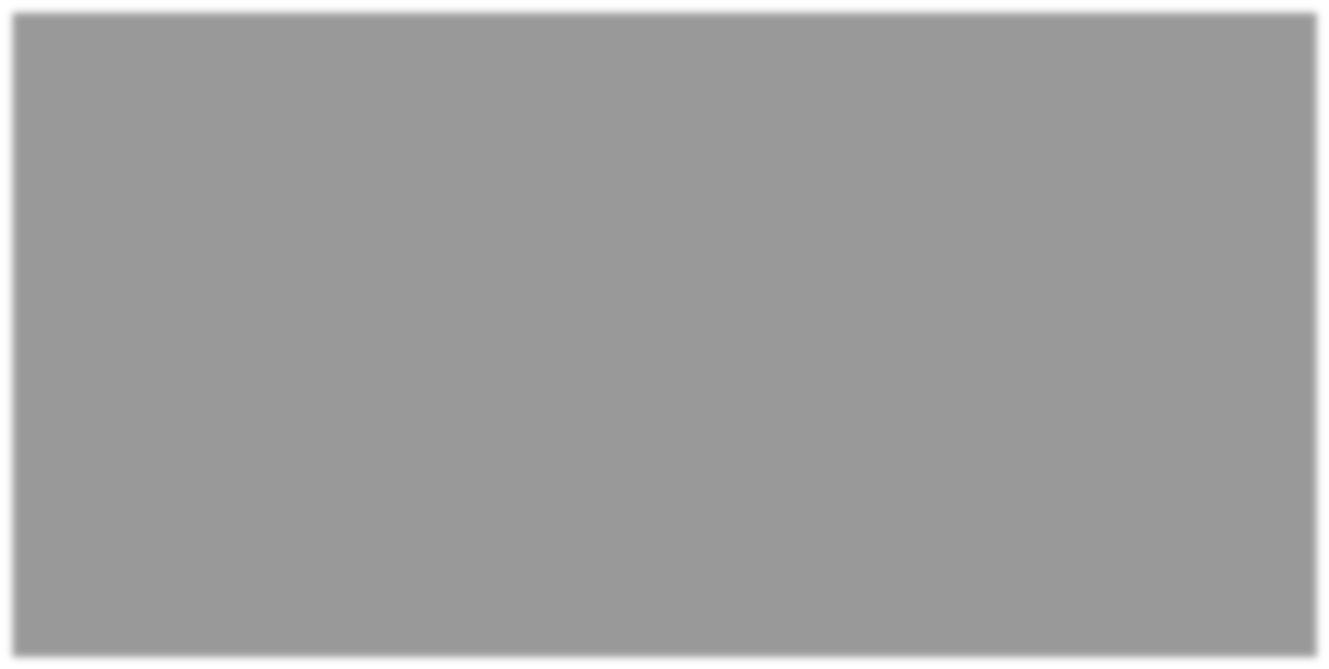
It is important to understand team roles, events & artifacts of the SCRUM Agile Life Development lifecycle.

#### 2.3.2 Scrum team roles

* **Product owner**: Product expert who represents the stakeholders, and is the voice of the customer.
* **Development team**: Group of professionals who deliver the product (developers, programmers, designers).
* **Scrum master**: Organized servant-leader who ensures the understanding and execution of Scrum is followed.

**2.3.3**

**Scrum events**



* **Sprint**: Iterative time boxes where a goal is accomplished. Time frame does not exceed one calendar month and are consistent throughout the development process.
* **Sprint planning**: Where the entire Scrum team get together — at the beginning of every

Sprint — to plan the upcoming sprint.

* **Daily Scrum**: 15 minute time boxed meeting held at the same time, every day of the Sprint, where the previous day’s achievements are discussed, as well as the expectations for the following one.
* **Sprint review**: An informal meeting held at the end of every Sprint where the Scrum team present their Increment to the stakeholders, and discuss feedback.
* **Sprint retrospective**: A meeting where the Scrum team reflect on the proceedings of the previous Sprint and establish improvements for the next Sprint.

#### 2.3.4 Scrum Artifacts

* **Product backlog**: Managed by the Product Owner, it’s where all the requirements needed for a viable product are listed in order of priority. Includes features, functions, requirements, enhancements, and fixes that authorize any changes to be made to the product in future releases.
* **Sprint backlog**: A list of the tasks and requirements that need to be accomplished during the next Sprint. Sometimes accompanied by a Scrum task board, which is used to visualize the progress of the tasks in the current Sprint, and any changes that are made in a ‘To Do, Doing, and Done’ format.

### 2.4 Technology/Platform Used

**2.4.1 Frontend Technologies:**

**HTML:** The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript

**BootStrap:** Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

**JavaScript:** JavaScript, often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries.

**2.4.2 Introduction to Django Framework:**

Django is an MVT web framework that is used to build web applications. The huge Django web-framework comes with so many “batteries included” that developers often get amazed as to how everything manages to work together. The principle behind adding so many batteries is to have common web functionalities in the framework itself instead of adding latter as a separate library.

One of the main reasons behind the popularity of Django framework is the huge Django community. The community is so huge that a separate website was devoted to it where developers from all corners developed third-party packages including authentication, authorization, full-fledged Django powered CMS systems, e-commerce add-ons and so on. There is a high probability that what you are trying to develop is already developed by somebody and you just need to pull that into your project.

**2.4.3. Why should you use Django?**

Django is designed in such a way that encourages developers to develop websites fast, clean and with practical design. Django’s practical approach to getting things done is where it stands out from the crowd.

If you’re planning to build a highly customizable app, such as social media website, Django is one of the best frameworks to consider. Django strength lies in its interaction between users or its ability to share different types of media. One of the great advantage of django is its ability to utilize large community-based support which gives you highly customizable third-party ready to use plugins in your applications.

Below are the top reasons to choose Django for web development −

1. **Python**

Python is arguably one of the easiest programming languages to learn because of its simple language constructs, flow structure and easy syntax. It is versatile and runs websites, desktop applications and mobile applications embedded in many devices and is used in other applications as a popular scripting language.

1. **Batteries Included**

Django comes with common libraries which are essential to build common functionalities like URL routing, authentication, an object-relational mapper (ORM), a templating system and db-schema migrations.

1. **Built-in admin**

Django has an in-built administration interface which lets you handle your models, user/ group permissions and to manage users. With model interface in place, there is no need for a separate database administration program for all but advanced database functions.

1. **Doesn’t get in your way**

Creating a Django application adds no boilerplate and no unnecessary functions. There’s no mandatory imports, third-party libraries and no XML configuration files.

1. **Scalable**

Django is based on MVC design pattern. It means that all the entities like db (database), back-end and front-end code are individual entity. Django allows us to separate code from the static media including pictures, files, CSS and JavaScript that make up your site.

1. Django supports a full list of **third-party libraries for web servers**, caching, performance management, clustering and balancing. One of the advantages Django provides is the support for major email and messaging applications and services like ReST and OAuth.
2. **Battle tested**

Django was first open-sourced in 2005. After 12 years of growth, Django now not only runs news publishing websites but also runs all or part of major global enterprise like Pinterest, Instagram, Disqus, Bitbucket, EventBrite and Zapier. This makes it a robust and reliable web framework to work with.

1. **Huge package support**

Because of its large community support and huge developers network, there is a high possibility that whatever you intend to do might have been done before. Large international community of developers contribute to the community by releasing their projects as open-source packages.

One such repository of these projects is Django Package site. Currently, Django packages list over 3400 plus reusable Django apps, sites and tools to use in our Django projects.

1. **Actively developed**

One of the biggest risks associated with open source project is its sustainability. We cannot be sure if it lasts long.

There is no such risk with Django as it is 12 years old. Its consistent releases, newer/better versions and active community is growing every-day with a large core team of voluntary contributors who maintains and improve the code base every-day.

1. **Stable releases**

Open-source software projects like Django are, in many cases, actively developed and more secure than competing proprietary software as many developers are developing and testing it every day. However, the drawback of an open-source software project is the absence of a stable codebase to commercially viable development.

**2.5Django MVT Architecture**

Django is based on **MVT (Model-View-Template)** architecture. MVT is a software design pattern for developing a web application.



**MVT Structure has the following three parts –**

**Model:**The model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySql, Postgres).

**View:** The View is the user interface — what you see in your browser when you render a website. It is represented by HTML/CSS/Javascript and Jinja files.

**Template:**A template consists of static parts of the desired HTML output as well as some special syntax describing how dynamic content will be inserted.

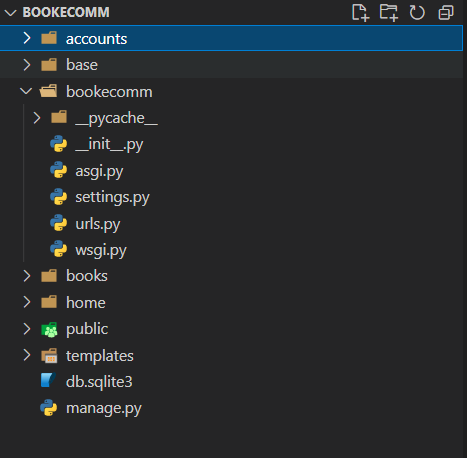
**2.6Project Structure**

A Django Project when initialized contains basic files by default such as manage.py, view.py, etc. A simple project structure is enough to create a single-page application. Here are the major files and their explanations. Inside the geeks\_site folder ( project folder ) there will be the following files-

To start a project:

>>django-admin startproject bookecomm

Then these following files will be generated by Django-admin.



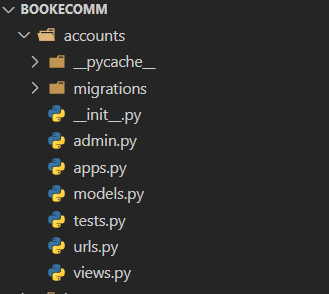
**Manage.py :** This file is used to interact with your project via the command line(start the server, sync the database… etc).

>>python manage.py startapp accounts

**Folder(bookecomm):** This folder contains all the packages of your project. Initially, it contains four files –

* **\_init\_.py –**It is a python package. It is invoked when the package or a module in the package is imported. We usually use this to execute package initialization code, for example for the initialization of package-level data.
* **settings.py –**As the name indicates it contains all the website settings. In this file, we register any applications we create, the location of our static files, database configuration details, etc.
* **urls.py –**In this file, we store all links of the project and functions to call.
* **wsgi.py –**This file is used in deploying the project in WSGI. It is used to help your Django application communicate with the webserver.

**Accounts:**



### 2.7 Use Case Description

**2.7.1**

|  |  |
| --- | --- |
| **Use Case 01** | **Search** |
| Objective | A user can search for a book of his choice by selecting category and title. Then a select query is used to retrieve data from the database and display the selected information |
| Actor | User |
| Pre-Condition | - |
| Post Condition | User needs to register and login himself |
| Input | The user will select a category and enter title in a text box provided |
| Output | The system will display the books which matches the selected search criteria. A dataset is created as a result of select query. Later the dataset is binded to the data repeater to display the selected data. |

#### 2.7.2

|  |  |
| --- | --- |
| **Use Case 02** | **Register** |
| Objective | If the user doesn’t have an account then he will be asked to register. |
| Actor | User |
| Pre-Condition | User should have valid email Id. |
| Post Condition | User needs to login himself |
| Input | The user will enter details in the registration form according to the required fields. |
| Output | After registration the user will be directed to the login page |

**2.7.3**

|  |  |
| --- | --- |
| **Use Case 03** | **Login** |
| Objective | If the user wants to get access to all the functionalities of Online Book Store he should login using his username(email) and password. |
| Actor | User |
| Pre-Condition | User should be registered and verified his email. |
| Post Condition | - |
| Input | The user will enter his username and password. |
| Output | If it is a successful login the user will be directed to the main home page. Else if the user enters invalid information he will be asked to check the entered information. |

#### 2.7.4

|  |  |
| --- | --- |
| **Use Case 04** | **Update Profile** |
| Objective | If the user wants to change his personal account information then he can update his selected fields and the entire data will be updated in the data base through an update query. |
| Actor | User |
| Pre-Condition | User must have an account |
| Post Condition | User needs to save changes. |
| Input | The user will update his account information |
| Output | The system will update the entered information in the database using an update query. |

|  |  |
| --- | --- |
| **Use Case 05** | **Logout** |
| Objective | If the user wants to end his session and sign out of the website then he can use the logout option. |
| Actor | User |
| Pre-Condition | User should be logged in |
| Post Condition | - |
| Input | The user will select a logout option. |
| Output | The user’s account session comes to an end and he should login again if he wants to enter into the website. |

|  |  |
| --- | --- |
| **Use Case 06** | **Place an Order** |
| Objective | If the user wants to purchase a book then he can place an order by selecting the add to shopping cart button and entering the quantity required under the book description. |
| Actor | User |
| Pre-Condition | User must be logged in. |
| Post Condition | User needs to add something in cart. |
| Input | The user will enter the quantity required and click the add to shopping cart button. |
| Output | The order will be added to the user’s shopping cart |

|  |  |
| --- | --- |
| **Use Case 07** | **Update shopping Cart** |
| Objective | If the user wants to change the quantity of a book or change a book then he can update his shopping cart. |
| Actor | User |
| Pre-Condition | User must be logged in. |
| Post Condition | User needs to register and login himself |
| Input | The user will click the details button in the shopping cart summary to edit and update his order details.. |
| Output | The updated order details are reflected in the shopping cart summary |

|  |  |
| --- | --- |
| **Use Case 08** | **View Shopping Cart** |
| Objective | If the user wants to view the items he added to the shopping cart then he can click the shopping cart link at the top of the page. |
| Actor | User |
| Pre-Condition | User must be logged in. |
| Post Condition | - |
| Input | The user will click the shopping cart link at the top of every page. |
| Output | The user’s shopping cart summary will be displayed in the form of a tabular format with all the books and their quantity. A total cost of all the items is also displayed at the bottom. |

**Administrator –Use case**

|  |  |
| --- | --- |
| **Use Case 09** | **Admin Login** |
| Objective | If the Administrator wants to get access to all the functionalities of Online Book Store he should login using his username and password |
| Actor | Admin |
| Pre-Condition | Must be a registered Administrator. |
| Post Condition | - |
| Input | The Administrator will enter his username and password. |
| Output | If it is a successful login the Administrator will be directed to Django’s admin Interface. Else if the Administrator enters invalid information he will be asked to check the entered information. |

|  |  |
| --- | --- |
| **Use Case 10** | **Add or Delete Category** |
| Objective | If the Administrator wants to add or delete a book category then he can insert or delete a book category using his administration rights and the category table will be updated in the database. |
| Actor | Admin |
| Pre-Condition | Must be a registered Administrator. |
| Post Condition | - |
| Input | If the Administrator wants to add a book category the he should click the insert link button in the category page else he can delete a particular selected book category. |
| Output | The updated categories list will be displayed in the main home page |

|  |  |
| --- | --- |
| **Use Case 11** | **Add or delete Books** |
| Objective | If the Administrator wants to add or delete a book then he can insert or delete a book using his administration rights and the book table will be updated in the database. |
| Actor | Admin |
| Pre-Condition | Must be a registered Administrator. |
| Post Condition | - |
| Input | If the Administrator wants to add a book the he should click the insert link button in the book page and fill the following fields related to the book. 1. Title 2. Author 3. Price 4. Category 5. Description  If he wants to delete a book he can click the delete button to remove it from the database. |
| Output | The updated books list will be displayed in the main home page under their particular category. |

|  |  |
| --- | --- |
| **Use Case 12** | **Manage Orders** |
| Objective | If the Administrator wants to add or delete an order then he can insert or delete an order using his administration rights. |
| Actor | Admin |
| Pre-Condition | Must be a registered Administrator. |
| Post Condition | - |
| Input | If the Administrator wants to add an order the he should click the insert link button in the orders page else he can delete a particular selected order |
| Output | The updated orders list will be processed to the users. |

### 2.8 Constraints, Assumptions, Risks and Dependencies

|  |  |
| --- | --- |
| **Parameter** | **Description** |
|  |  |
|  |  |
|  |  |

**Chapter 3**

**System Diagrams**

In this chapter

3.1 Use Case Diagram

3.2 Process Flow Chart

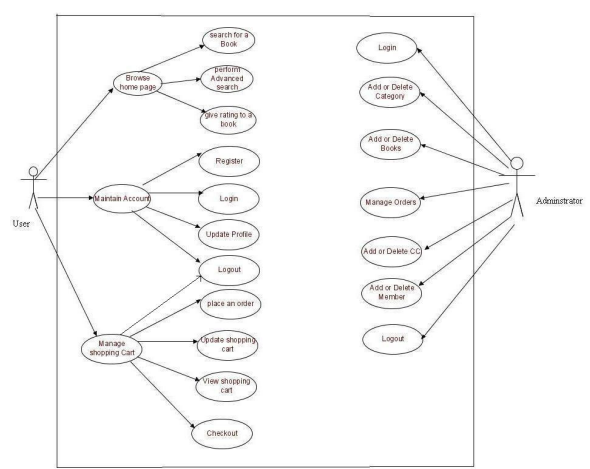
3.3 Data Structures/Tables used

3.4 Class Diagram

3.5 State Transition Diagram

3.6 PFD of a Threat Modelling applied application

### 3.1 Use Case Diagram



### 3.2 Process Flow

**3.2.1 Process Flow Description:**

### 3.3 Data Structures/Tables Used

### 3.4 Class Diagram

### 3.5 State Transition Diagram

### 3.6 Process Flow

**Chapter 4**

### UI Design & Screenshot

In this chapter

4.1 UI Design

4.2 Emails screenshot

4.3 Model screenshot

#### 4.1 UI Design

# CHAPTER 5 TESTING & DEPLOYMENT

**In this chapter:**

**5.1 Testing**

**5.2 Deployment**

## 5.1 Testing

Testing is a process rather than a single activity. Testing must be planned and it requires discipline to act upon it. The quality and effectiveness of software testing are primarily determined by the quality of the test processes used. The activities of testing can be divided into the following basic steps:

* Planning and Control
* Analysis and Design
* Implementation and Execution

**A list of available testing scripts**

USER:

• Registration

• Login

• Add To Cart

• Edit Cart

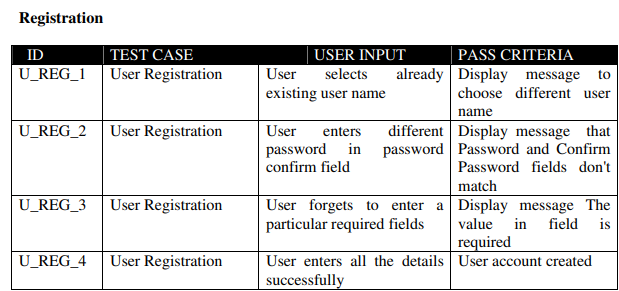
ADMIN:

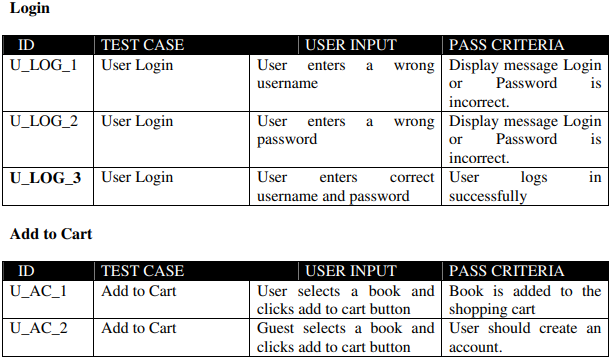
• Create and Delete book from Category

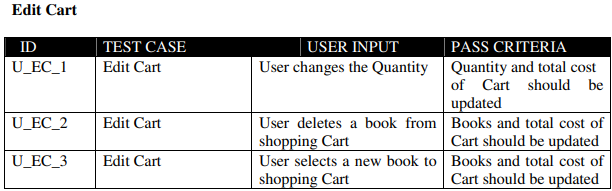
• Create and Delete a Category

• Manage Orders

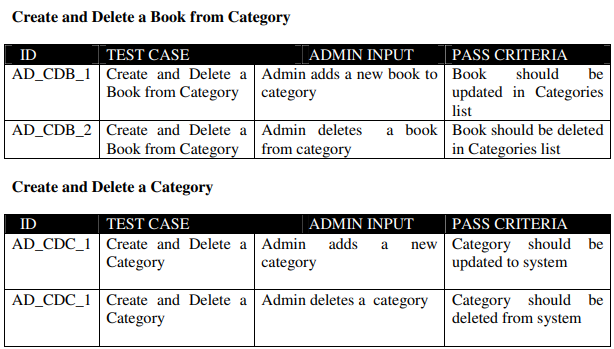
• Manage Members

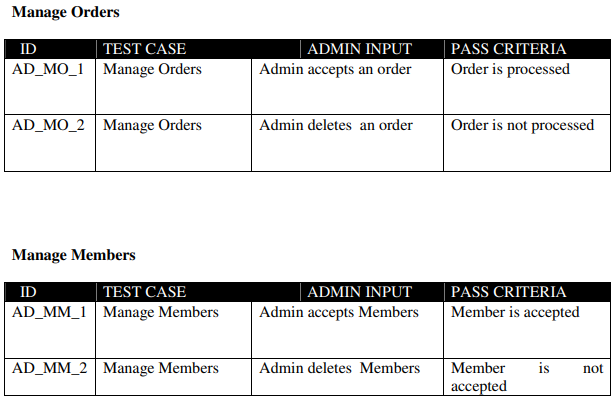






**Admin:**





**Unit testing**

Unit testing is a method of testing that verifies the individual units of source code are working properly. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct.

**Load testing**

Load testing is the process of creating demand on a system or device and measuring its response. It generally refers to the practice of modeling the expected usage of a software program by simulating multiple users accessing the program concurrently. As such, this testing is most relevant for multi-user systems; often one built using a client/server model, such as web servers

**System Testing**

Once the entire system has been built then it has to be tested against the Software Requirement Specification and System Specification to check if it delivers the features required. System testing can involve a number of specialist types of test to see if all the functional and non-functional requirements have been met.

**Performance Testing**

The system should meet the performance requirements as mentioned in the Vision document. The performance will be evaluated based on the response time of the GUI and the database commands.

**Manual Testing**

Manual Testing will be done to ensure the correctness of various parts of the code using test cases generated by the tester.

**Pass/fail criteria**

The system should satisfy all the functional requirements, in the Vision document. Each feature to be tested will be evaluated against its requirement as stated in the Vision Document. The pass or fail of a test depends on whether the system meets with all the particular post conditions. Test cases executed on the Online Book Store will pass if they meet the specific requirements as mentioned in the Vision Document.

**5.2 Deployment:**

# CHAPTER 6 Conclusion

**In this chapter:**

**6.1 Conclusion**

**6.2 Limitations**

**6.3 Future Scope**

**6.4 References**

## 6.1 Conclusion

This type of ecommerce websites with recommendation systems proves to be more popular and profitable due to maximum engagement of the user. Recommendations not only increase engagement of the user with the site but also increases curiosity and interest of the user with the products(in our case books). Due to which sales are increased.

## 6.2 Limitations

The Book Store is limited to Online websites only.

**6.3 Future Scope of BookStore:**

* Implementing Machine Learning in terms of Collaborative filtering.
* Extending Bookstore to Bokk rental and subscription model,
* Building an Mobile Application

### 6.4 References

1. <https://www.djangoproject.com/>
2. <https://www.analyticsvidhya.com/blog/2021/06/build-book-recommendation-system-unsupervised-learning-project/>
3. <https://www.kaggle.com/code/hoshi7/goodreads-analysis-and-recommending-books/data>

1. <https://www.kaggle.com/code/midouazerty/book-recommendation-system-with-machine-learning/notebook>
2. M. V. Murali, T. G. Vishnu and N. Victor, "A Collaborative Filtering based Recommender System for Suggesting New Trends in Any Domain of Research," 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), 2019, pp. 550-553, doi: 10.1109/ICACCS.2019.8728409.
3. <https://www.analyticsvidhya.com/blog/2021/07/recommendation-system-understanding-the-basic-concepts/#:~:text=A%20recommendation%20system%20is%20a,suggests%20relevant%20items%20to%20users>.