BSCS-S22-002

03-134191-045 Abdullah Qasim

03-134191-020 Shahroz Amir

Recommendation System for B2C E-commerce in Pakistan

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: Dr Ghulam Mustafa

Department of Computer Sciences

Bahria University, Lahore Campus

January 2023

© Bahria University, 2023

C e r t i f i c a t e

We accept the work contained in the report titled

"**Recommendation System for B2C E-commerce in Pakistan**"

Written by

Abdullah Qasim

Shahroz Amir

As a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: Dr Ghulam Mustafa

(Signature)

Jan 10, 2023

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134191-045	Abdullah Qasim	
03-134191-020	Shahroz Amir	

Date :Jan 10, 2023

Specially dedicated to

My beloved mother and father

(Abdullah Qasim)

My beloved mother and father

(Shahroz Amir)

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to my research supervisor, Dr Ghulam Mustafa for his invaluable advice, guidance and his/her enormous patience throughout the development of the research.

In addition, we would also like to express my gratitude to my/our loving parent and friends who had helped and given me encouragement.

Abdullah Qasim
Shahroz Amir

Recommendation System for B2C E-commerce in Pakistan

ABSTRACT

The internet has changed the way business is conducted in the 21st century. Most importantly it has changed the face and pace of business offering a dispensable opportunity to operate on digital platforms. Businesses find new sources of competitive advantage so they are increasingly adding ecommerce channels. In this case we are creating an ecommerce store which is a business to consumer store with a recommender system that automatically recommends products to the customer depending upon his activity on our Advance Search Engine. As the pandemic has passed people have understood that how you can buy things online without getting into hassle of going to the market to buy things. For B2C Ecommerce store different Businesses can connect to our store and can sell their products on our store and the recommender system will benefit by recommending product which they are looking for and recommend a product with good consumer rating.

TABLE OF CONTENTS

DECLARATION **ii**
ACKNOWLEDGEMENTS **iv**
ABSTRACT **v**
TABLE OF CONTENTS **vi**
LIST OF TABLES **x**
LIST OF FIGURES **xi**

CHAPTERS

1 INTRODUCTION **1**
1.1 Background 1
1.2 Problem Statements 1
1.3 Aims and Objectives 2
1.4 Scope of Project 2

2 SOFTWARE REQUIREMENTS SPECIFICATION **3**
2.1 User Classes and Characteristics 3
2.1.1 Customer 3
2.1.2 Admin: 3
2.1.3 Vendors: 4
2.2 Operating Environment 4
2.3 Design and Implementation Constraints 4
2.4 External Interface Requirements 5
2.5 Use Case Diagrams: 5
2.5.1 System use case diagram: 5

	2.5.2	Register use case diagram	6
	2.5.3	Login use case diagram	7
	2.5.4	Customer use case diagram	8
	2.5.5	Admin use case diagram	8
2.6	Use Case Description Table		9
	2.6.1	Register	9
	2.6.2	Login	10
	2.6.3	Add Products	10
	2.6.4	Update Products	11
	2.6.5	Buy Products	11
	2.6.6	Manage User	11
	2.6.7	Manage Users	12
2.7	Other Non-functional Requirements		12
	2.7.1	Performance Requirements	12
	2.7.2	Security Requirements	13
	2.7.3	Software Quality Attributes	13
	2.7.4	Other Requirements	13
3	**DESIGN AND METHODOLOGY**		**14**
3.1	Domain Model		14
3.2	Sequence Diagram		15
	3.2.1	Customer/User Registration	15
	3.2.2	Customer/User Login	15
	3.2.3	Purchase	16
	3.2.4	Admin Login	16
	3.2.5	Manage Accounts:	17
	3.2.6	Generate Reports:	17
3.3	Class Diagram		18
3.4	Entity Relationship Diagram		19
3.5	Methodology		20
4	**DATA AND EXPERIMENTS**		**21**
4.1	Languages used for Implementation:		21

	4.1.1	Python:	21
	4.1.2	Html:	21
	4.1.3	CSS:	22
	4.1.4	JavaScript:	22
	4.1.5	Node.js:	22
4.2	Framework:		22
	4.2.1	React.js:	22
	4.2.2	Bootstrap:	23
	4.2.3	Express.js:	23
4.3	Database:		23
	4.3.1	MongoDB:	23
4.4	Tools used for Implementation:		23
	4.4.1	VS code:	23
	4.4.2	Abode XD:	24
	4.4.3	Draw.io:	24
4.5	Dataset:		24
5	**RESULTS AND DISCUSSIONS (or USER MANUAL)**		**25**
5.1	Overview		25
5.2	Application Prototype:		26
	5.2.1	Home Page:	26
	5.2.2	Login:	27
	5.2.3	Register user:	27
	5.2.4	Products:	28
	5.2.5	Add products:	28
	5.2.6	Admin Dashboard:	29
6	**CONCLUSION AND RECOMMENDATIONS**		**30**
6.1	Project Achievements		30
6.2	Recommendations:		30
6.3	Future Work		31
6.4	Implementation Issues and Challenges		32
	6.4.1	Conclusion	32

REFERENCES **33**

LIST OF TABLES

TABLE	TITLE	PAGE
Table 1: Operation Environment		4
Table 2:Register		9
Table 3: Login		10
Table 4:Add Products		10
Table 5: Update Products		11
Table 6: Buy Products		11
Table 7: Manage User		11
Table 8: User Management		12

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2. 1: System Use Case		5
Figure 2. 2: Register Use Case		6
Figure 2. 3: Login Use Case		7
Figure 2. 4: Customer Use Case		8
Figure 2. 5: Admin Use Case		9
Figure 3. 1: Domain Model		14
Figure 3. 2: User Registration		15
Figure 3. 3: User Login		15
Figure 3. 4: Purchase		16
Figure 3. 5: Admin Login		16
Figure 3. 6: Manage Account		17
Figure 3. 7: Generate Reports		17
Figure 3. 8: Class Diagram		18
Figure 3. 9:Entity Relationship Diagram		19
Figure 3. 10: FDD[1]		20

Figure 5. 1: Recommendation Model 25

Figure 5. 2: Recommendation Model 26

Figure 5. 3: Home 26

Figure 5. 4: Login 27

Figure 5. 5: Register User 27

Figure 5. 6: Products 28

Figure 5. 7: Add Products 28

Figure 5. 8 Admin Dashboard 29

CHAPTER 1

INTRODUCTION

1.1 Background

Now a day's people are very busy in their lives, they have not enough time to visit the market and buy their things which are important for them. Recommendation system for B2C E-commerce in Pakistan is an online web based ecommerce store for customers where they can buy products online with an additional feature of recommendation system which will recommend consumer products which will suit their interest or system will recommend that product what he will be looking and show him the best product available in that store with a good rating score by consumer as well.

1.2 Problem Statements

E-commerce provides an easy way to sell products to a large customer base but there is a lot of competition among multiple e-commerce sites. When users land on an e-commerce site, they expect to find what they are looking for very quickly and easily. So, here comes a need of some recommendation system which makes the whole process very smooth and hassle free for the customer to select and buy auto recommended products. Also, users are not sure about the brands or the actual products they want to purchase. Many customers nowadays search for their products on Google rather than visiting specific e-commerce sites. They believe that Google will take them to the e-commerce sites that have their product.

1.3 Aims and Objectives

The objectives of the thesis are shown as following:

1. To design system Easy to use.
2. To find relevant products.
3. To search in a quick way with Recommendation system.
4. To Suggest or recommend only relevant products.

1.4 Scope of Project

Recommendation system to assist users to finding the similar products recommender system works as a provider in finding relevant and related products making relevant suggestions to the users. These systems navigate the Data Set through Machine Learning algorithms to come up with relevant suggestions for users based either on their explicitly mentioned preferences or objective behaviours. It is therefore necessary to build high-quality product recommender systems for providing fine-tuned recommendations to users in wide range of daily-life applications. In this regard, researchers and industry practitioners need to come forward and work on some of the prominent issues and challenges the area of recommender systems, which are being presented in this proposal.

CHAPTER 2

SOFTWARE REQUIREMENTS SPECIFICATION

2.1 User Classes and Characteristics

There are three types of user classes within application, the first one is using the UI (user interface) of the application and the second one is managed and control the backend of the application.

Users:

- User Customer
- Admin
- Vendors

2.1.1 Customer

- Customer can Register/sign in into the application
- Customer can view and purchase products
- Customer can have online assistance

Customer can leave a review or rating

2.1.2 Admin:

- Admin can login into the application.
- Admin will have all the control of application.
- Admin can view the orders.
- Admin can block or unblock a user or worker.
- Admin can generate report.
- Admin can monitor the activities online assistance.

2.1.3 Vendors:

- Vendor can login to the system.
- Vendor can add products, view orders and maintain its inventory.
- Vendor can dispatch orders.

2.2 Operating Environment

While design and developing this platform, there are following hardware and software requirements

Table 1: Operation Environment

Name	Description
Operating system	Windows, Linux
Browsers	Google Chrome 77.0.3865,120, Firefox 94, Microsoft Edge 86.0.622.63 and Internet explorer 11
Language	Html, CSS, Bootstrap, JavaScript Python, React.js, Node.js, Express.js
Tools	Visual Studio Code 1.67.1, Draw.io
Database	Mongo DB

2.3 Design and Implementation Constraints

E-buy is a web-site for both providing and seeking a different products on this platform. For the front-end development we using Html, CSS, Bootstrap and JavaScript. For back-end development Node.js and Express.js is used. Mongo DB database will be used to store the data. React.js framework/Library will be used to develop web application.

2.4 External Interface Requirements

Following are the external interface requirements of this project:

2.5 Use Case Diagrams:

2.5.1 System use case diagram:

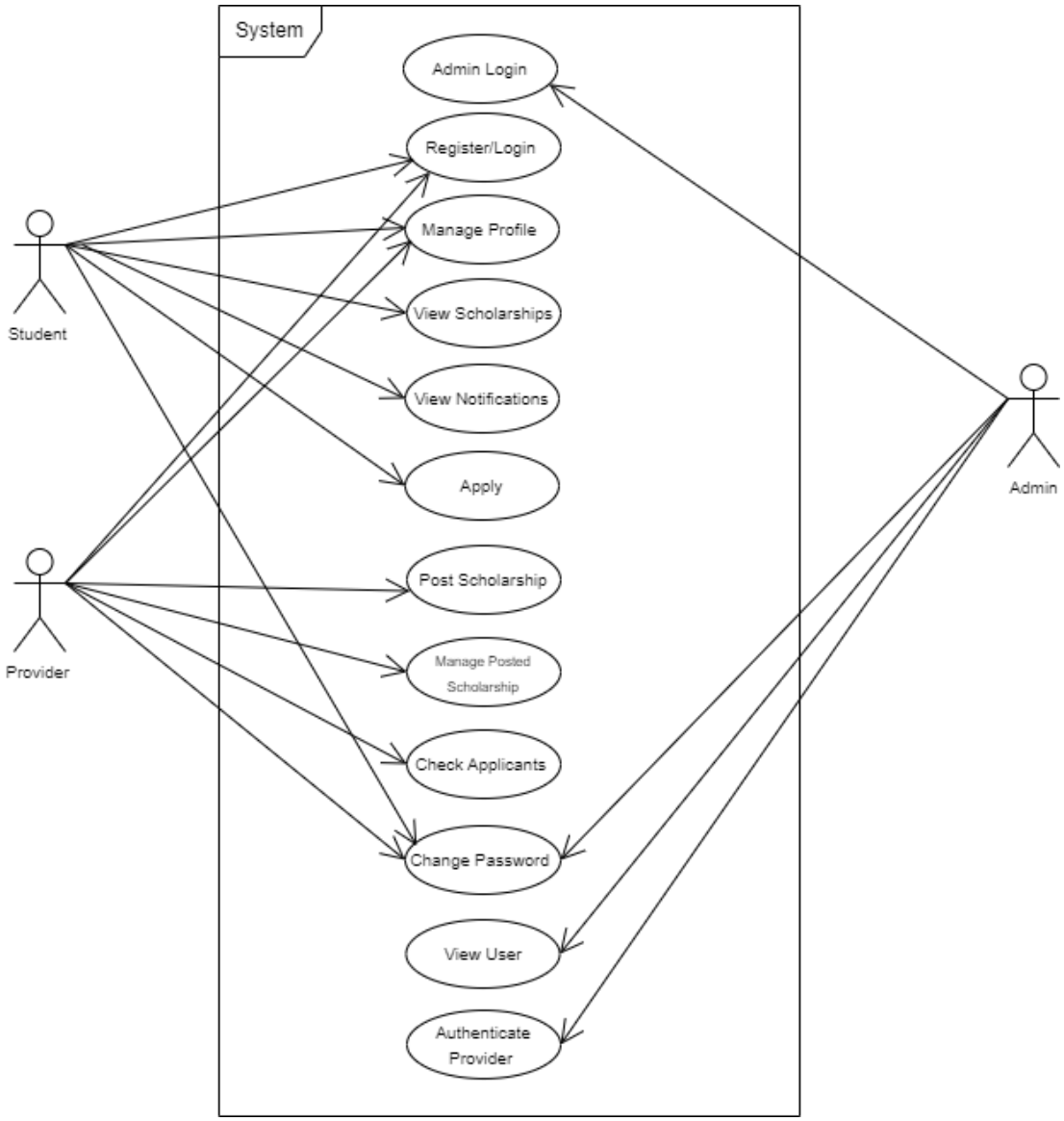

Figure 2. 1: System Use Case

Users and Admin that control the overall system of the application. According to scholarship provider first register in the application, sign-in, post scholarships, view posted scholarships and view applied candidates. According to scholarship seeker first register in the application, sign-in, view scholarships, apply scholarships, check notifications according to its need. According to admin first sign-in, authenticate providers, and manage users.

2.5.2 Register use case diagram

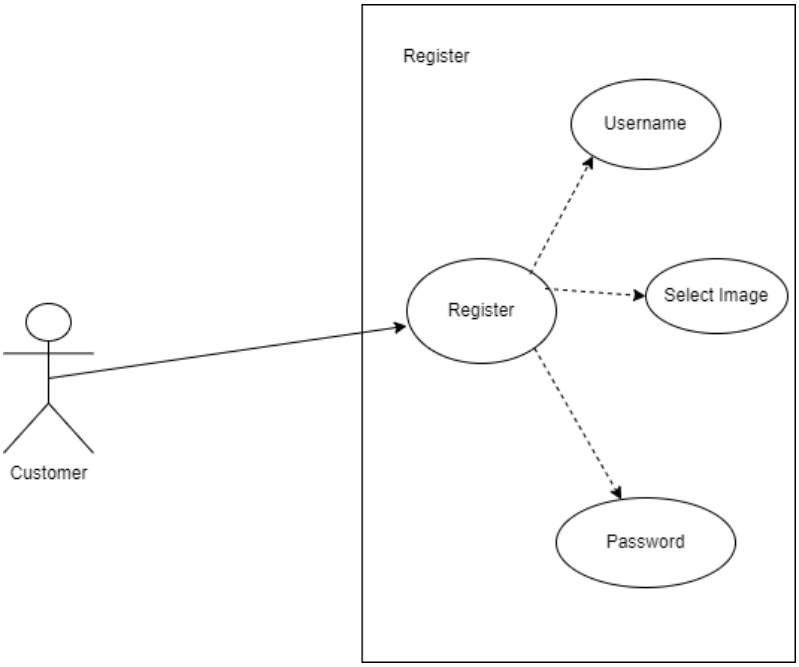

Figure 2. 2: Register Use Case

Sign-up process for the user to enter the required information according to the given sequence to create the profile or account for the application. Sequence:

- User Name
- Email
- Password

2.5.3 Login use case diagram

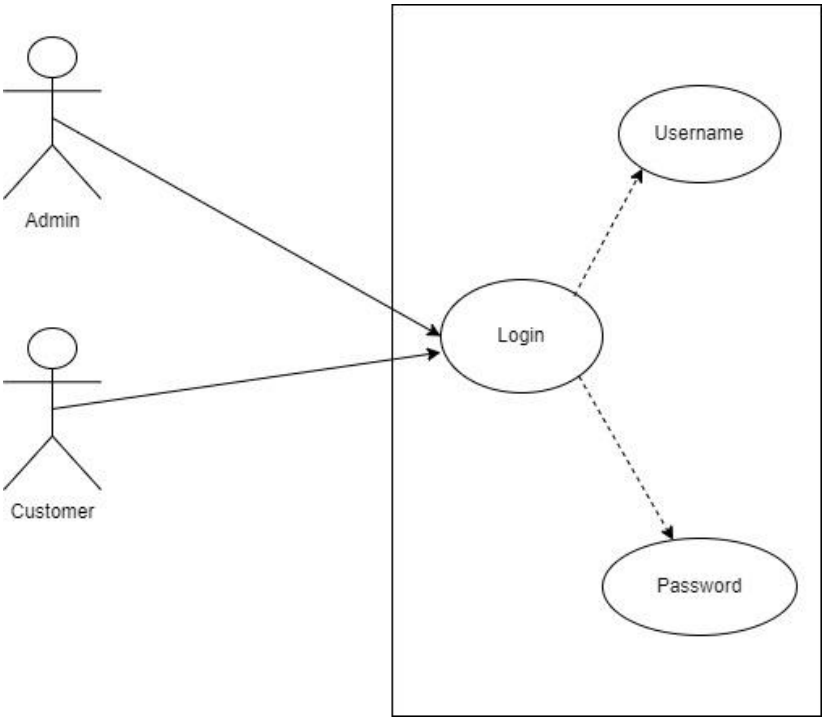

Figure 2. 3: Login Use Case

After the completion of the registration the next part is Sign-In process by enter the correct detail in the two-given section. Sections:

- Email
- Password

2.5.4 Customer use case diagram

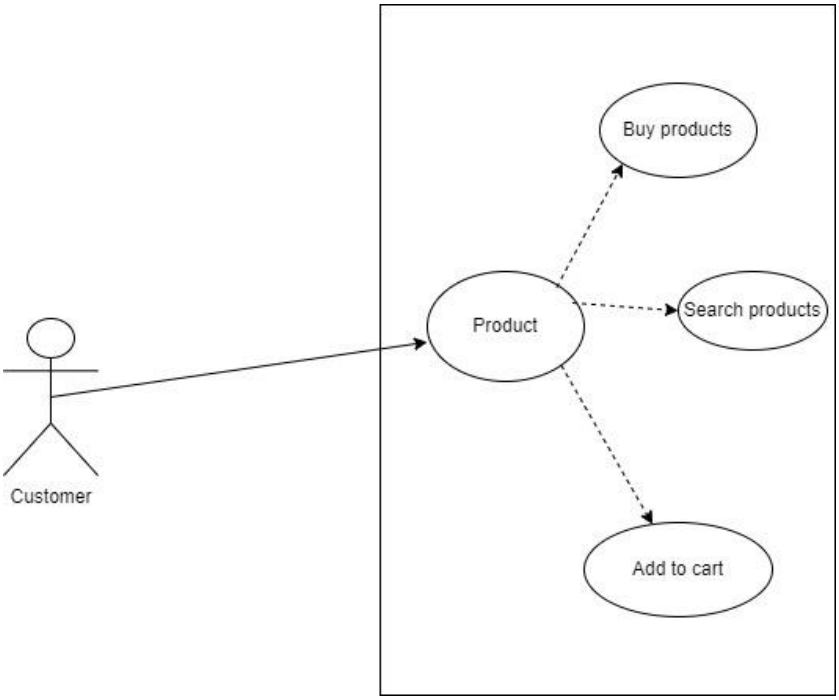

Figure 2. 4: Customer Use Case

Customer can search products and add products into the cart to buy the selected product.

2.5.5 Admin use case diagram

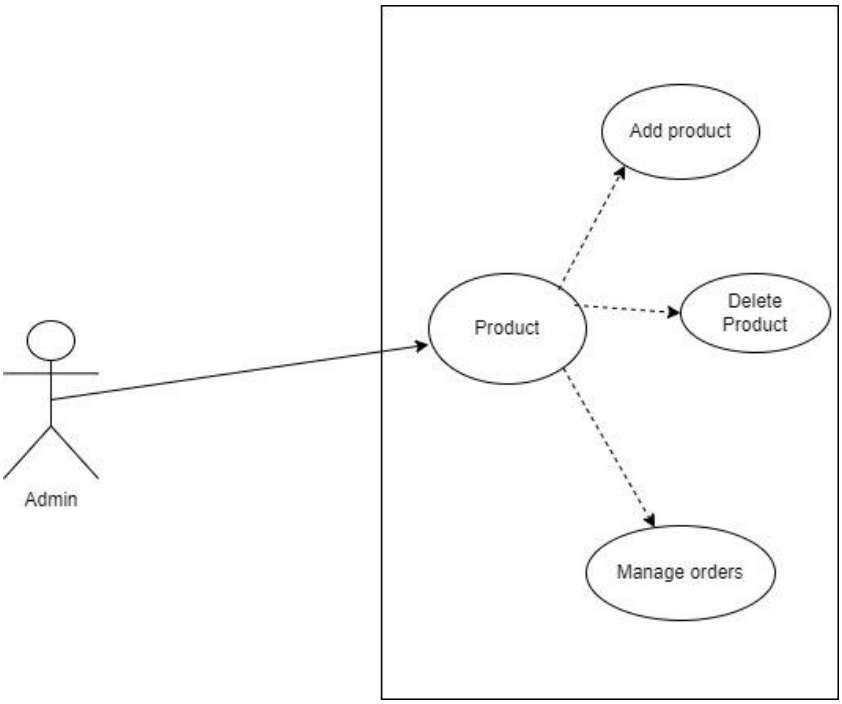

Figure 2. 5: Admin Use Case

2.6 Use Case Description Table

The following tables along with the certain conditions, flow of work and modules to be used in the Application.

2.6.1 Register

Table 2:Register

1	**Unique Identifier**	U1
2	**Objective**	To have users correct information and validate email for maintaining record
3	**Priority**	High
4	**Source**	User/customer
5	**Actors**	User/customer
6	**Flow of events**	Basic flow: • User will fill the required fields • Validating data • User is ready to use app Alternative flow: • Register users cannot register again • If user already exist, then app will move to login screen
7	**Includes**	None
8	**Pre-condition**	User/customer
9	**Post Condition**	Users is successfully registered

2.6.2 Login

Table 3: Login

1	**Unique Identifier**	U2
2	**Objective**	To validate that the user login to their account or are they registered yet or not
3	**Priority**	High
4	**Source**	Admin/User
5	**Actors**	Admin/User
6	**Flow of events**	Basic flow: • Users enter the login credentials • Validation data • User redirected to main screen of their account Alternative flow: • User cannot use same credentials for another account
7	**Includes**	U1
8	**Pre-condition**	User must be connected with internet
9	**Post Condition**	User can easily log-in or log-out

2.6.3 Add Products

Table 4:Add Products

1	**Unique Identifier**	U3
2	**Objective**	To allow seller to add products
3	**Priority**	High
4	**Source**	Admin/Seller
5	**Actors**	Admin/Seller
6	**Includes**	U2
7	**Pre-condition**	Successful login credentials

2.6.4 Update Products

Table 5: Update Products

1	**Unique Identifier**	U4
2	**Objective**	To allow seller to update products
3	**Priority**	High
4	**Source**	Admin/Seller
5	**Actors**	Admin/Seller
6	**Includes**	U3
7	**Pre-condition**	Successful login credentials
8	**Post Condition**	Admin can successfully update his products.

2.6.5 Buy Products

Table 6: Buy Products

1	**Unique Identifier**	U5
2	**Objective**	To allow customer to buy products
3	**Priority**	Moderate
4	**Source**	Customer/user
5	**Actors**	Customer/user
6	**Pre-condition**	Successful login credentials
7	**Post Condition**	Customer can successfully buy products.

2.6.6 Manage User

Table 7: Manage User

1	**Unique Identifier**	U6
2	**Objective**	This use case describes the event of an admin to block or unblock an account
3	**Priority**	High
4	**Source**	Admin
5	**Actors**	Admin
6	**Flow of events**	Basic flow: • Admin will check and manage the accounts

		• Admin will accept or reject the user's account
8	**Pre-condition**	Administrator is logged in
9	**Post Condition**	Account is successfully accepted or rejected

2.6.7 Manage Users

Table 8: User Management

1	**Unique Identifier**	U7
2	**Objective**	This use case describes the event of an admin to block or unblock an account
3	**Priority**	High
4	**Source**	Admin
5	**Actors**	Admin
6	**Flow of events**	Basic flow: • Admin will check and manage the accounts • Admin will accept or reject the user's account
7	**Includes**	U2
8	**Pre-condition**	Administrator is logged in
9	**Post Condition**	Account is successfully accepted or rejected

2.7 Other Non-functional Requirements

Non-functional requirements of this project are:

2.7.1 Performance Requirements

The system should perform all the actions correct and frequently. A slow internet connection may impact the performance of the application. A good and stable internet connection is required

2.7.2 Security Requirements

To provide the users with the best application experience we will update our application from time to time to fix the bugs and errors.

2.7.3 Software Quality Attributes

- System shall permit only to authorize users.
- Users should login to the system to get the facilities.

2.7.4 Other Requirements

Software quality attributes of this project are:

- **Availability:** The application will be available for the user 24/7.
- **Testability:** The application should be easy to test at each level and find the bugs/defects at each level of development and remove the defects easily.
- **Maintainability:** The system should be maintained and configured easily

CHAPTER 3

DESIGN AND METHODOLOGY

The design phase gives an overview of the design and methodology of the Recommendation system in ecommerce store. Design and methodology give a complete view of how the Recommendation system in ecommerce store application operates. This will help developers and users to understand and check the design in detail.

3.1 Domain Model

Domain models represent the set of requirements that are common to systems within a product line. There may be many domains, or areas of expertise, represented in a single product line and a single domain may span multiple product lines. The requirements represented in a domain model include:

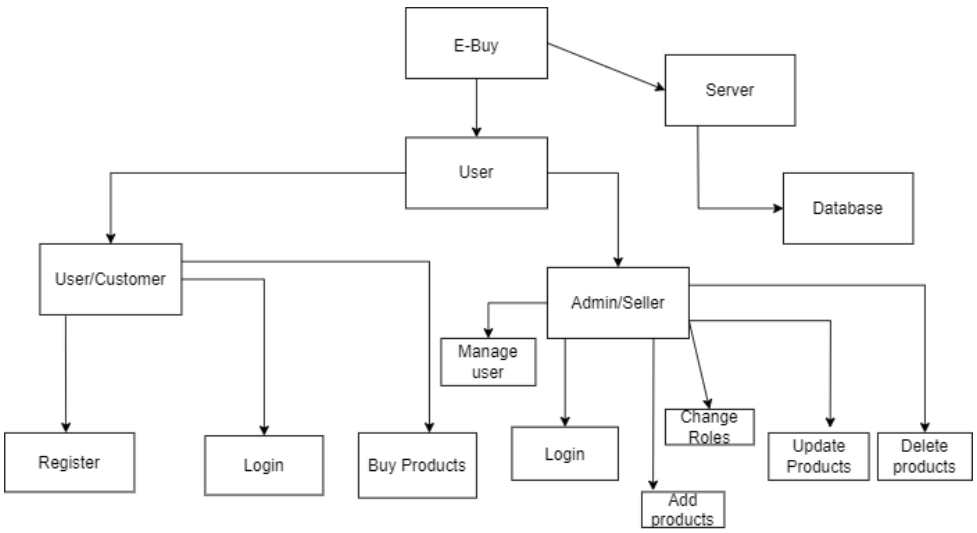

Figure 3. 1: Domain Model

3.2 Sequence Diagram

Sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and as horizontal arrows, the messages exchanged between them, in the orders in which they occur. Sequence diagrams of the "Ecommerce store" given below:

3.2.1 Customer/User Registration

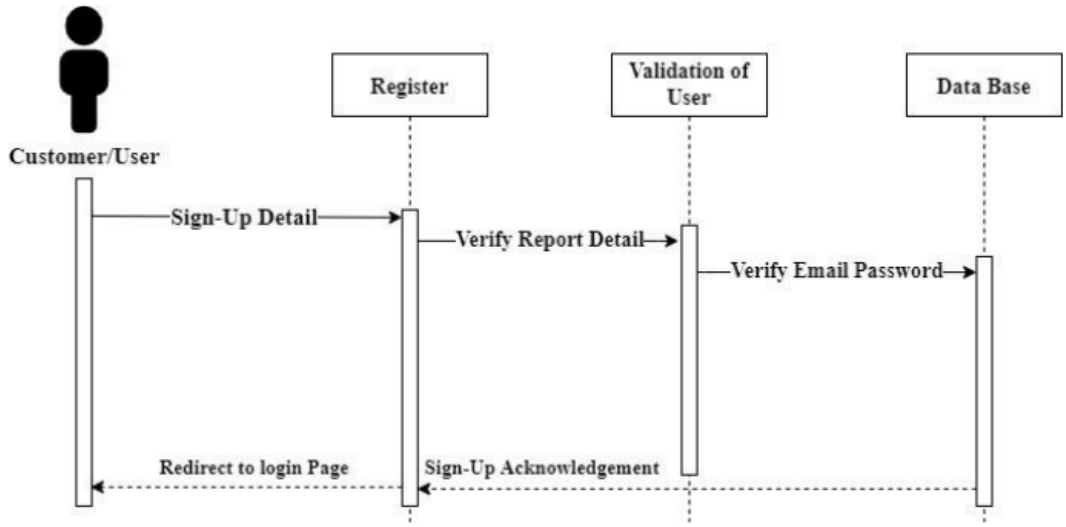

Figure 3. 2: User Registration

3.2.2 Customer/User Login

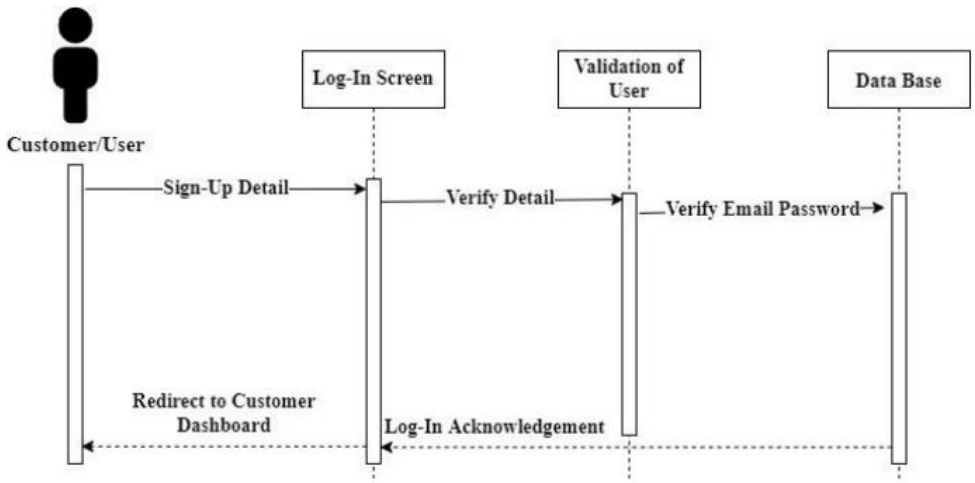

Figure 3. 3: User Login

3.2.3 Purchase

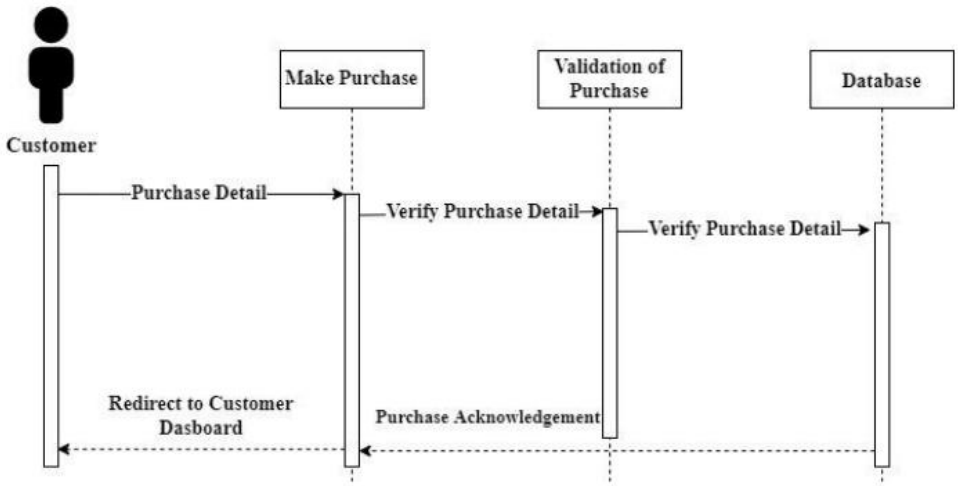

Figure 3. 4: Purchase

3.2.4 Admin Login

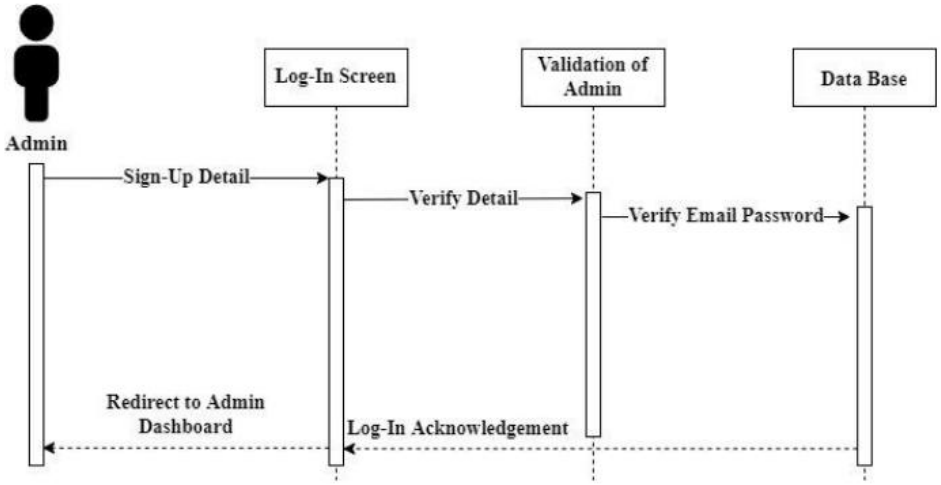

Figure 3. 5: Admin Login

3.2.5 Manage Accounts:

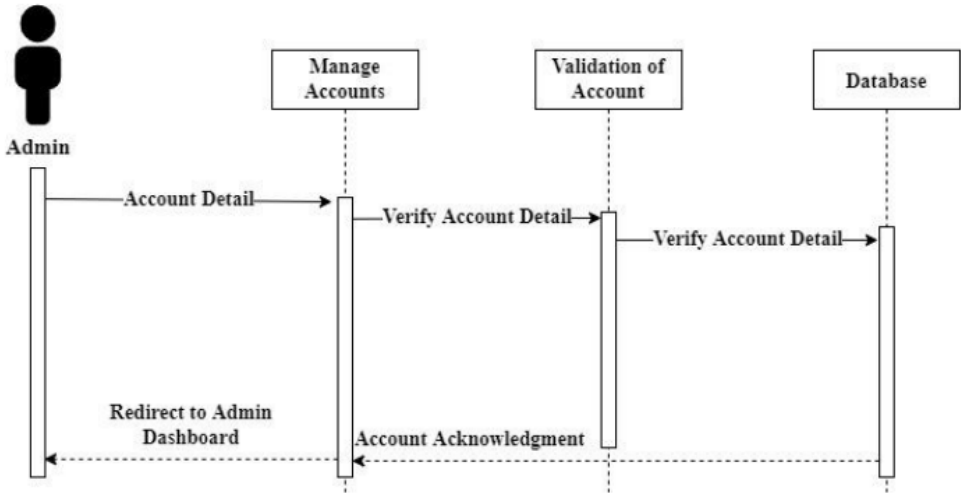

Figure 3. 6: Manage Account

3.2.6 Generate Reports:

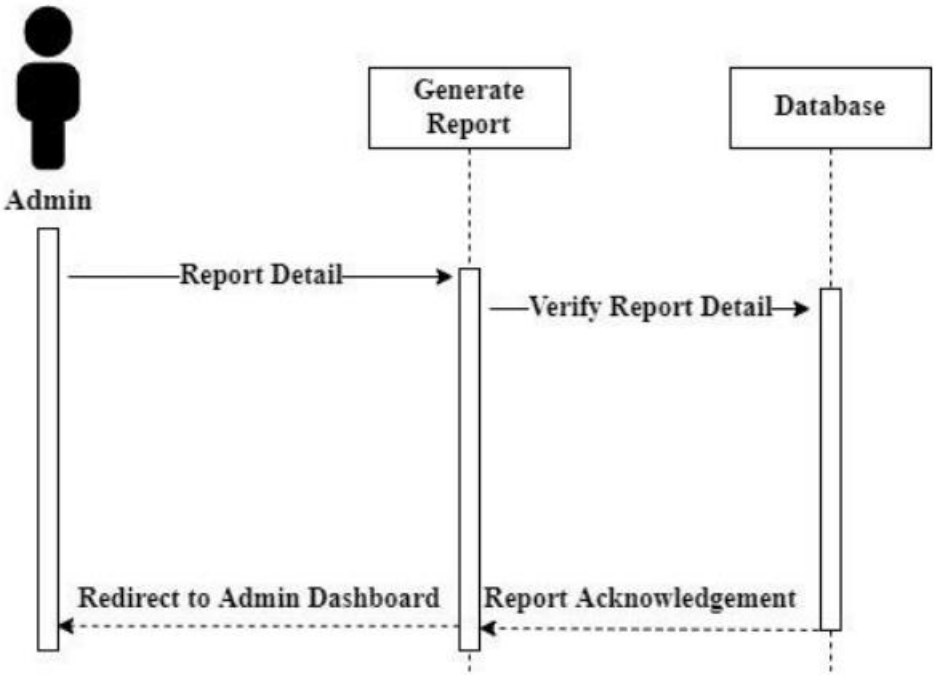

Figure 3. 7: Generate Reports

3.3 Class Diagram

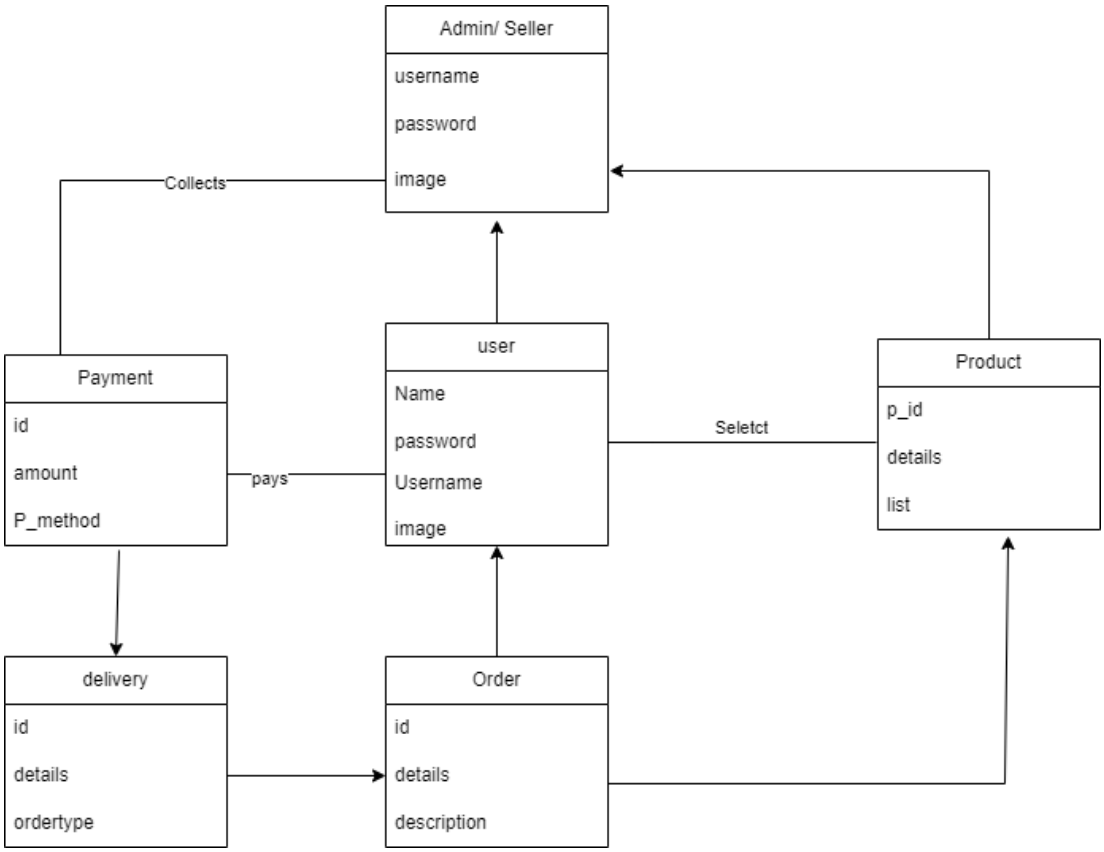

Figure 3. 8: Class Diagram

3.4 Entity Relationship Diagram

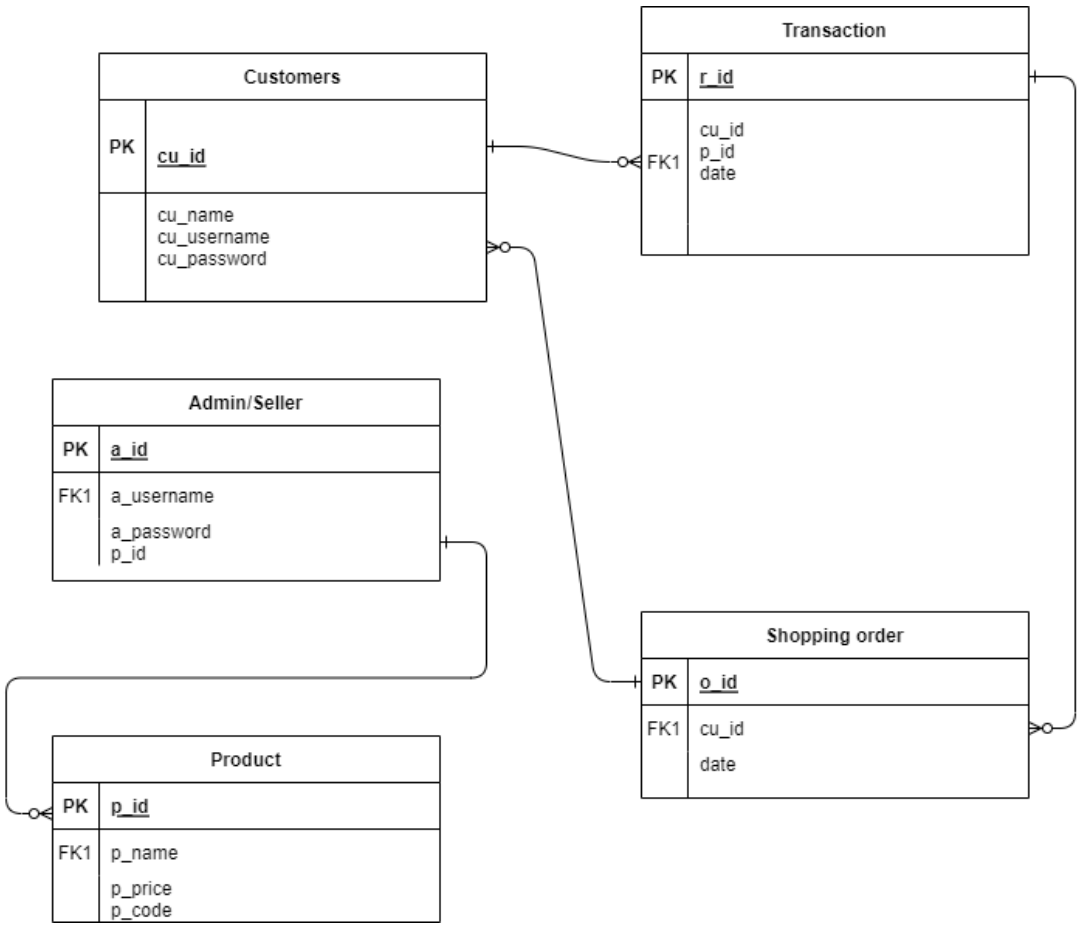

Figure 3. 9:Entity Relationship Diagram

3.5 Methodology

The methodology which will be used in our project is “Feature Driven Development” in Agile. FDD is customer-centric, iterative and incremental, with the goal of delivering tangible software results often and efficiently. The structure of feature driven development easy to adaptable and uses a sensible approach.

The following are the steps of FDD that will be considered in my project:

1. Develop an overall model: This is the first phase of FDD in which project’s scope will be decided initially. It will decide what this project is, what functionalities this project is going to perform.
2. Build a feature list: This phase provides a foundation for upcoming phases. In this phase we will identify features by inspecting model.
3. Plan by features: This third phase is the most crucial phase where all the planning of the project starts. All the data gathered in previous phase will be documented. Estimate time of completion for each module will be given. Gantt chart and other charts can be used.
4. Design by feature: In this phase more precision is required on how a system by feature is designed. Sequence diagrams for each module will be made so if there is any error in functionality it can be altered.
5. Build by feature: This phase is the last phase where the technical part comes in action. After designing phase features one after another will be made. After completion, this project be tested and debugged.

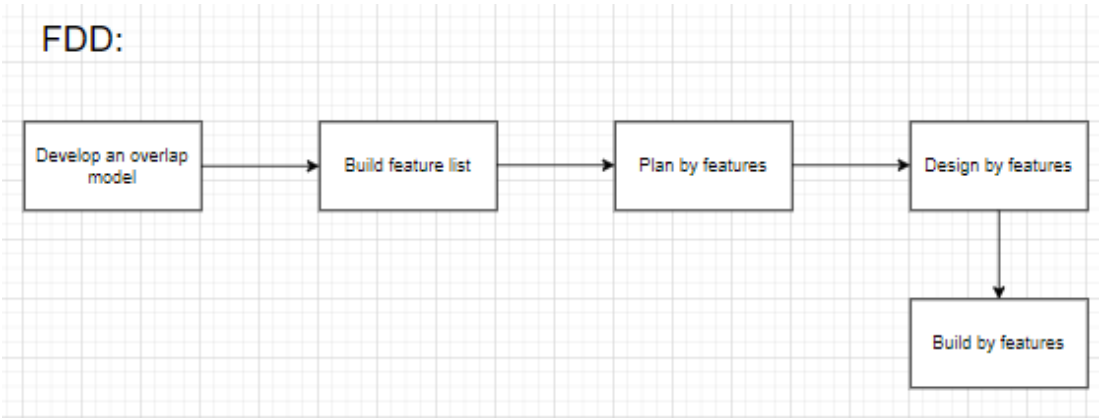

Figure 3. 10: FDD[1]

CHAPTER 4

DATA AND EXPERIMENTS

4.1 Languages used for Implementation:

4.1.1 Python:

Python is an open-source high-level programming language. It is basically used for web development, game development, language development. It has a very huge developer community. Python is programming language that lets you work quickly and integrate systems more efficiently. Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

4.1.2 Html:

Html is the set of markup symbols or codes used in a file to show on a World Wide Web browser page. It is also known as Hypertext Markup Language. The markup instructs the web browser on how to show the user's text and picture content on a web page. Some people refer to each markup code as a tag, while others refer to it as an element.

4.1.3 CSS:

Cascading Style Sheets (CSS) are a set of rules that we use to define and modify web pages. We can alter the font, colour, size, spacing and styling in our content by using it. CSS allow web designers to have much more control over their pages look and layout. The position and appearance of text and other items on your website can both be controlled using CSS.

4.1.4 JavaScript:

JavaScript is the most popular programming language that use for web. The programmers use JavaScript create interactive web content. It is use as a client-side programming language. It allows to implement the complex feature on the website. JavaScript also show the error message before any information transmitted to the server.

4.1.5 Node.js:

Node.js is a platform which we run on different operating systems like Linux, Windows and macOS. It is used for backend. Node is a java script runtime environment. Node.js is mostly used in server side environment.

4.2 Framework:

4.2.1 React.js:

React is a frontend JavaScript library used for creating attractive interfaces. It provides efficient working with less code.

4.2.2 Bootstrap:

Bootstrap is a free, open-source front-end development framework for the creation of websites and web apps. Designed to enable responsive development of mobile-first websites, Bootstrap provides a collection of syntax for template designs.

4.2.3 Express.js:

Node.js is a server side runtime environment of java script, we do most of the changes on backend using node but there are some part which are not easily handle by node for this we use express.js. Express.js is the framework of the node.js

4.3 Database:

4.3.1 MongoDB:

MongoDB is a document based database which makes use of collections and documents which consist of key value pairs which is the basic unit of data in MongoDB. Collection contain some sets of documents and functions which is equivalent to relational database tables.

4.4 Tools used for Implementation:

4.4.1 VS code:

Microsoft Visual Studio Code is a standalone source code editor that runs on multiple operating systems. It is used to develop computer programs, as well as websites, web apps, web services, and mobile apps. It supports many programming languages and a

set of features that differs per language. VS Code provides initial support including syntax highlighting etc.

4.4.2 Abode XD:

Abode XD is a prototyping tool for user experiences and interaction designers. Adobe XD features are useful for creating wireframes prototypes UI designing for digital products such as mobile apps and websites.

4.4.3 Draw.io:

Draw.io is a free diagramming application that allows users to create and share diagrams within a web browser. It helps to draw diagrams like sequence diagrams, collaboration diagrams, ER diagrams and class diagrams.

4.5 Dataset:

The dataset we used for recommendation of products are collected from kaggle.com[2].

The dataset have following categories:

- Product ID
- Gender
- Category
- Subcategory
- Product Type
- Color
- Usage
- Product Title
- Image
- Image URL
- Rating

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 Overview

E-Buy is a B2C ecommerce website where a customer can buy and view different products. The site also recommend different products on the bases of similarity.

5.2 Recommendation Model:

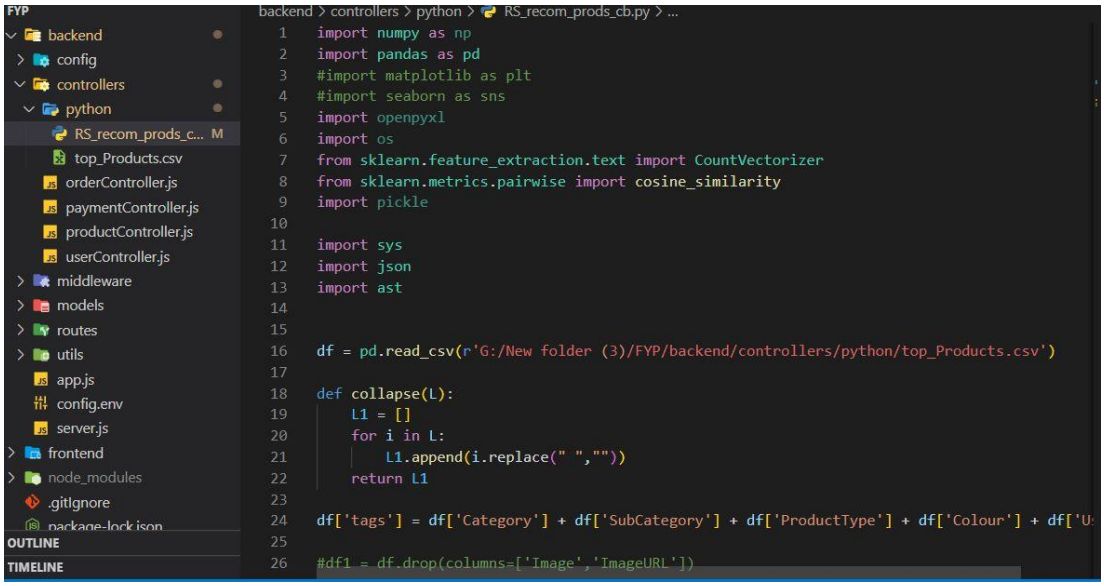

Figure 5. 1: Recommendation Model

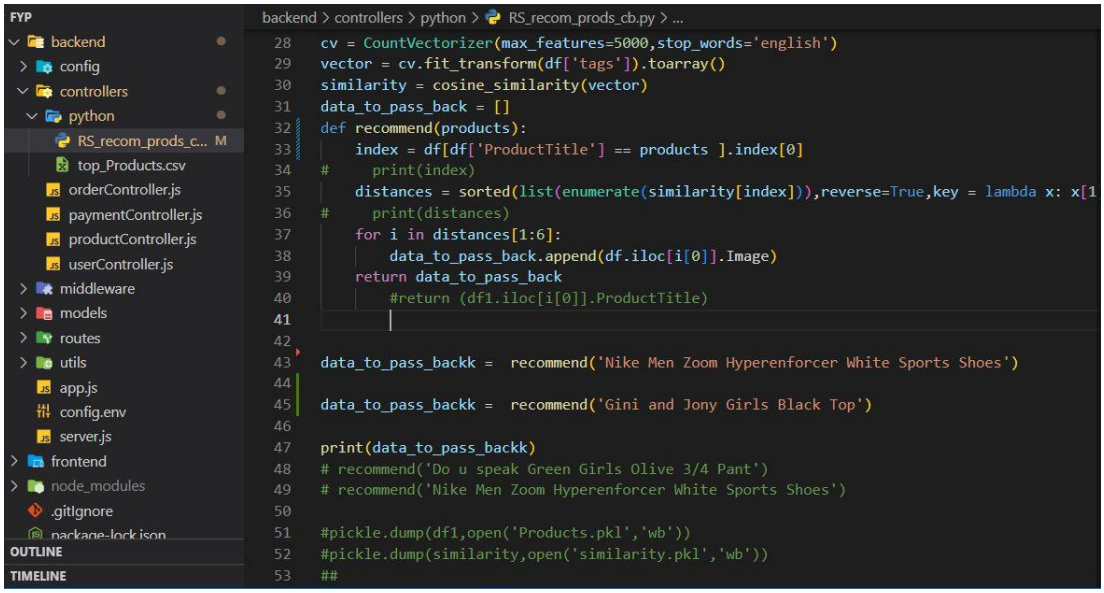

Figure 5. 2: Recommendation Model

5.3 Application Prototype:

5.3.1 Home Page:

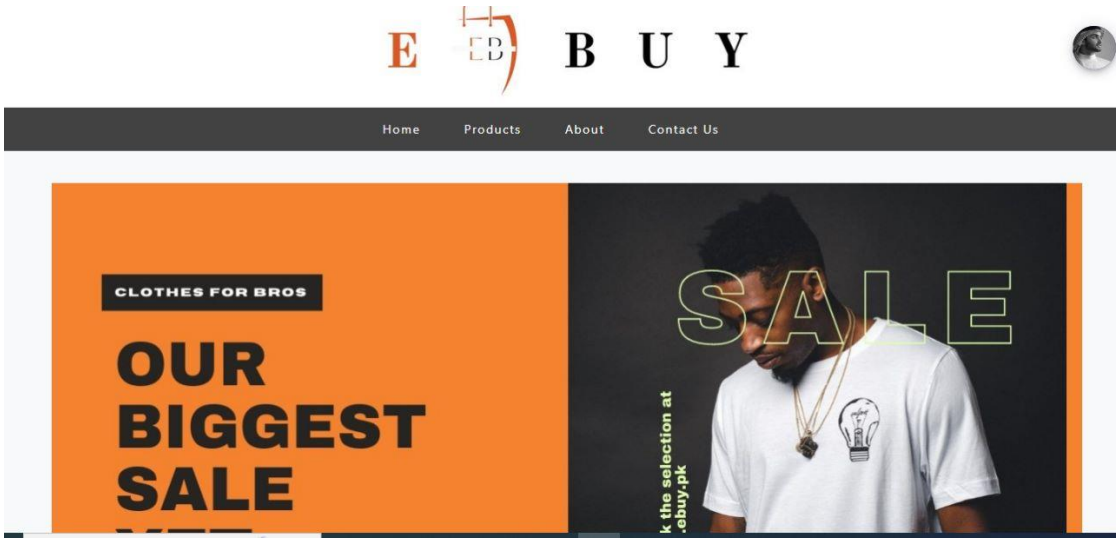

Figure 5. 3: Home

5.3.2 Login:

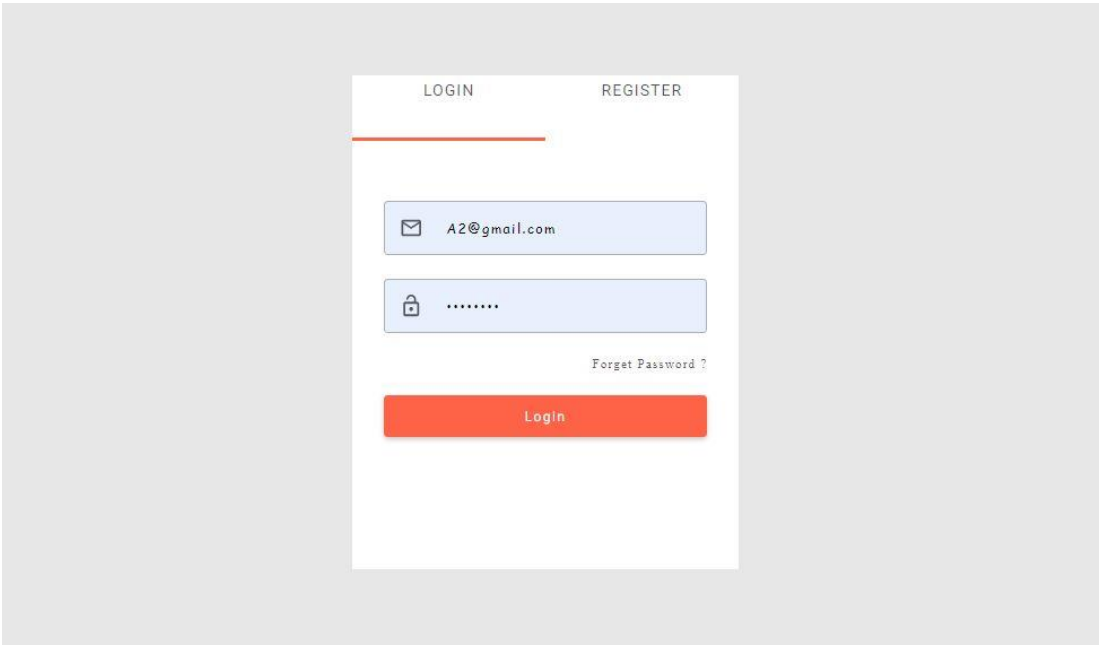

Figure 5. 4: Login

5.3.3 Register user:

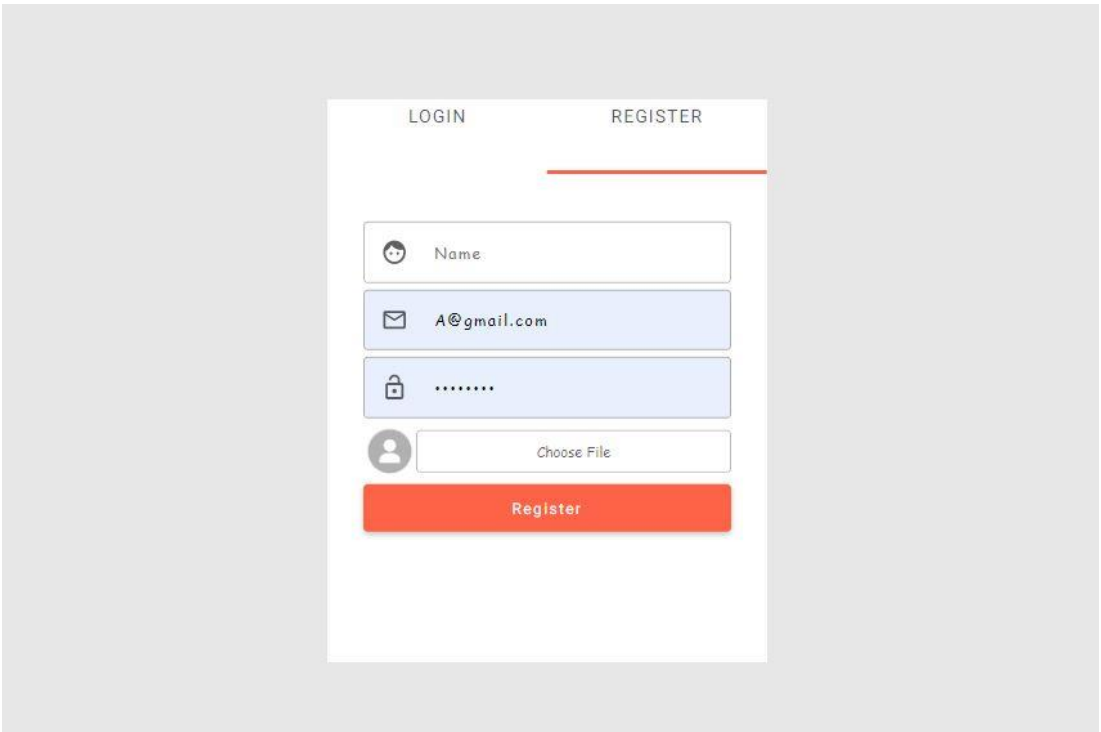

Figure 5. 5: Register User

5.3.4 Products:

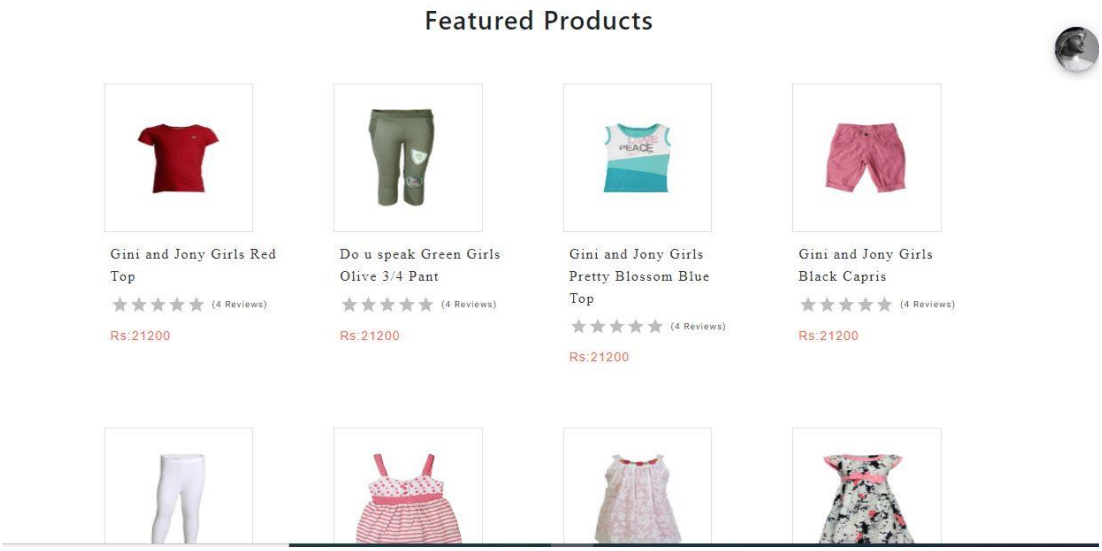

Figure 5. 6: Products

5.3.5 Add products:

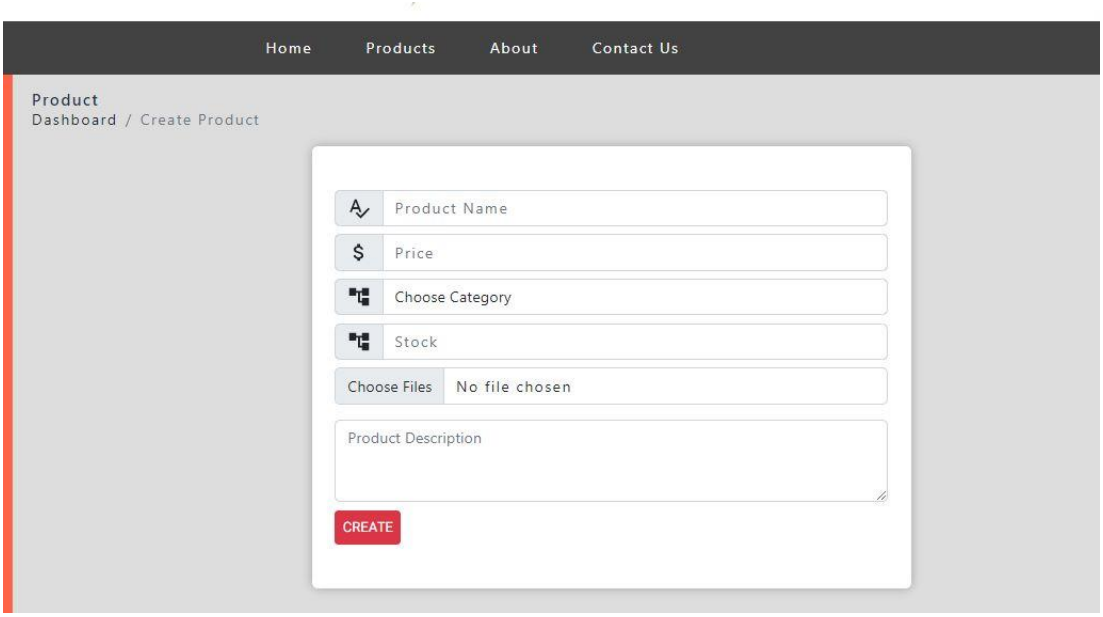

Figure 5. 7: Add Products

5.3.6 Admin Dashboard:

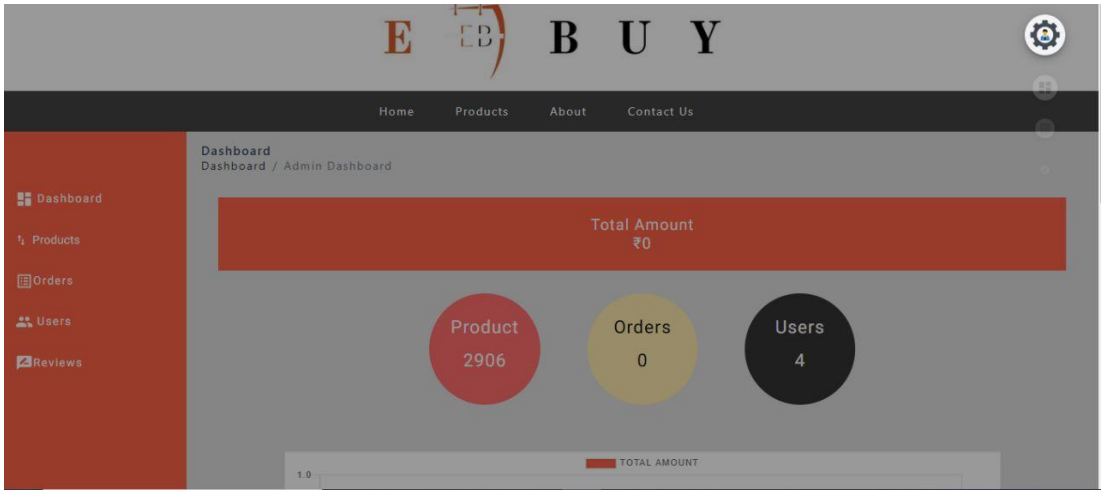

Figure 5. 8 Admin Dashboard

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 Project Achievements

From this project we have learned several things that were not known prior from this project. We got to know the whole structure of the web development how things are linked together to become a whole ecommerce store. As we integrated top end Recommendations in our store it would also benefit the user by having a look in the top products which are rated by the customers. The system was built using Python as the programming language. Python was chosen because of its ease of use and lot of libraries are available inbuilt, which makes the implementation part convenient and less stressful. All features implemented in the ecommerce store are useful features required by the customers of these days for online buying of goods.

6.2 Recommendations:

These recommendation systems have become the integral part of the web applications. This is mainly because the e-commerce websites have a large variety of product inventory's available the data which is to be displayed to the user can be overwhelming. As a result the user may face difficulties to search for the product that they may be looking for. At this point the recommendation system's comes to usage. Recommender systems are a tool to cope with this challenge, through product recommendation it is possible to fulfil customers' needs and expectations, helping maintaining loyal

customers while attracting new customers. In this project we have tried to use the ratings given to a product by other users to make recommendations to the current user. Python libraries like NumPy, pandas, matplotlib, and Seaborn, Scikit learn. These libraries are part of the system design as they work on the data set and provide the appropriate results[3].

The pipeline of a recommendation system has the following five phases:

1).Pre-processing

2).Model Training

3).Hyper Parameter Optimization

4).Post Processing

5).Evaluation.

In the dataset there are users who are over reactive and provide high ratings or least ratings. We need to normalize the ratings from such users. For this we find the average rating of the product and average rating of the user and add them to the global average An evaluation metric is to be chosen before the parameters are being tuned. In this we choose the value of K which indicates the number of recommendations to be generated as suggestion to the user. In our work conducted we have chosen the value of K as twenty five 5 so it recommends the top twenty five products to the user.

6.3 Future Work

The future work of the project includes improving the efficiency of the system. And it should also be able to give appropriate recommendations to the users who don’t have any previous purchase history or to the new users. In future we can try to use recurrent neural networks and deep learning. The future work will also include to have an android or IOS application for our E-commerce store.

6.4 Implementation Issues and Challenges

In this project, we encounter serval challenges throughout the project life cycle. First and foremost, development tools and environment need to be determined before the project can proceed. Choosing the suitable features for an ecommerce store was also a challenge to begin with while our store is all written in JSON and the model that was created for recommending product was through python the main challenge we faced was of integration of a python file with JavaScript and send information over to get our required result. Integration of backend with frontend was also a difficult task to play with and last to embed recommendation system with backend and creating an API for it.

6.4.1 Conclusion

The primary goal of this project is to provide recommendations to the user in a e-commerce website by making use of machine learning algorithms. We have designed and implemented the system using collaborative filtering and Pearson correlation coefficient. The dataset considered has the ratings given by the other users to a specific product and depending on the similarity between the rated products we try to recommend the products to our current user. With the help of deep learning techniques we can overcome some of the drawbacks of the matrix factorization technique. Deep learning uses recurrent neural networks to accommodate time in the recommender system which is not possible in the matrix factorization method. We can also work on providing sub-optimal recommendations to the user and record the reaction of the user and it can be used in the future by the system.

REFERENCES

[1] J. M. D. Santos, “XP, FDD, DSDM, and Crystal Methods of Agile Development,” *Project-Management*, 2022. [Online]. Available: https://project-management.com/xp-fdd-dsdm-and-crystal-methods-of-agile-development/. [Accessed: 26-Mar-2022].

[2] “Fashion Product Images Dataset | Kaggle.” [Online]. Available: https://www.kaggle.com/datasets/paramaggarwal/fashion-product-images-dataset. [Accessed: 14-Dec-2022].

[3] M. Kanagala, “Name of Committee Member PRODUCT BASED RECOMMENDATION SYSTEM USING MACHINE LEARNING TECHNIQUES . Name : MUKHUL KANAGALA,” 2020.