A

Software Requirements Specification

On

AI Based Personalized Electronic Gadget Recommendation System

Mentor:
Ms. Surbhi Sharma
Assistant Prof.

Submitted To:
Dr. Sushila Vishnoi
Associate Prof.

Submitted by:
Mohd. Arshad (19ESKIT056)
Mohit Singhal (19ESKIT057)
Mukul Palol (19ESKIT059)

DEPARTMENT OF INFORMATION TECHNOLOGY

Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Contents

List of Figures

1. Introduction

1.1 Problem Definition

Websites and applications that offer their users or customers an item or a product, have been trying to recommend them relevant products in order to their items/elements which they are interested in. Increasing the time that user spends on the website and increasing the interest of the user to the items in the website are the main reasons why the recommender systems are being used. Accuracy and time-efficiency are the most common problems of recommender systems. We will be trying to design an accurate and fast algorithm which will solve these problems.

People often complain about irrelevant recommendations of the websites they are using. User sometimes even complains about websites like Amazon or Netflix even though they are considered as having the best recommender systems. Since most of the global websites that offer their users an item, the problem can be considered as a worldwide problem.

For people who are totally unaware of how a recommendation system works, we can explain evaluation of the effectiveness of the recommender systems which is one of the most important issues to understand the topic. As it is explained in the Website of Creighton University, recall precision and DCG (Discounted Cumulative Gain) are the common metrics to assess the quality of the recommendation method. Recall is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. It is usually expressed as a percentage. Precision is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved. It is usually expressed as a percentage too. DCG measures the usefulness, or gain, of a document based on its position in the result list. The gain is accumulated from the top of the result list to the bottom with the gain of each result discounted at lower ranks. These are the basic concepts related to calculating recommendations.

1.2 Document Conventions

The format of this document is referred from the standard IEEE guidelines:

- Font face: Times New Roman

- Font size:
    - Heading: 18
    - Sub-heading: 14
    - Description: 12
- Bold face and indentation is used on general topics and or specific points of interest including the heading and sub-heading.

1.3 Intended Audience and Reading Suggestions

This document is mainly intended for project guides. The sequence for reading the document begins with the overview sections and proceeding through the sections that are most pertaining to each reader type.

1.4 Purpose

The purpose of this project is to implement a well-developed AI recommendation system that helps businesses improve their shopper's experience on website and result in better customer acquisition and retention by building an recommendation engine that recommends the customers about the best personalized Electronic Gadgets based on behavioral data and also analyze large sets of historical and real-time data to understand what each customer wants.

1.5 Scope

Neural Networks and Deep Learning have been all the rage the last couple of years in many different fields, and it appears that they are also helpful for solving recommendation system problems.

One of the benefits of Deep Learning is similar to matrix factorization, in that there is an ability to derive latent attributes. Deep Learning, however, can make up for some of the weaknesses of matrix factorization such as the inability to include time in the model — which standard matrix factorization isn’t designed for. Deep Learning, however, can utilize Recurrent Neural Networks which are specifically designed for time and sequence data.

Incorporating time into a recommender system is important, because there are often preference seasonal effects. For example, it is likely that in December, more people are going to be watching holiday-themed movies and buying home decorations.

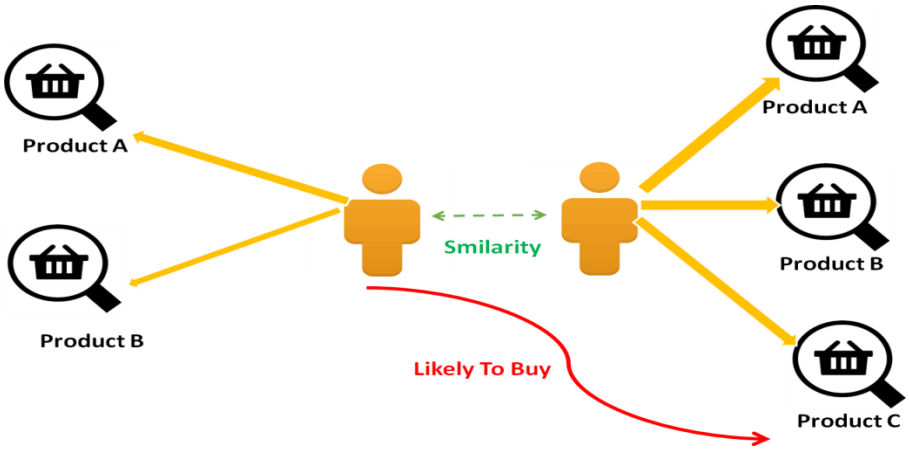

Figure 1- Collaborative Filtering for Personalized Recommendation System

2. Overall Description

2.1 Product Perspective

It is aimed at replacing the overwhelming quantity and choices of products that user face. The system will collect data and store it for fast and easy reference. The system will provide users with a personalized recommendation based on their search input and previous orders. It will also provide recommendations to new user based on content based filtering. The system is thus helpful to reduce the time and complexity of maintaining the records.

The objective of this project is to implement a well-developed AI recommendation system that helps businesses improve their shopper's experience on website and result in better customer acquisition and retention by building an recommendation engine that recommends the customers about the best personalized Electronic Gadgets based on behavioral data and also analyze large sets of historical and real-time data to understand what each customer wants.

There are different modules in our recommendation system:

- **Recommendation System Part I:** Product popularity based system targeted at new customers.

- **Recommendation System Part II:** Model-based collaborative filtering system based on customer's purchase history and ratings provided by other users who bought similar items.
- **Recommendation System Part III:** When a business is setting up its e-commerce website for the first time without any product rating.

2.2 Product Functions

With Product Recommender, the user will be able to see electronic gadget recommendations made especially for him/her. The recommendations will be based on the user's previous actions and the actions of the other users who have a similar taste of products as the user, who will get the recommendations. Since the recommendations are made based on the user, they are likely to be unique.

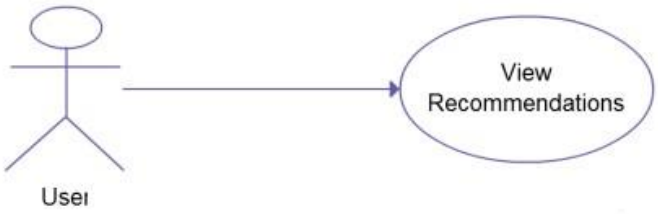

Figure 2- View Recommendation Use Case

Recommendation System is composed of the following fundamental features:

Users:

- Generate Data
- Get Recommendation

Inter-agent:

- Get Recommendation
- Provide Dataset
- Update Dataset

2.3 Constraints

- The Internet connection is a constraint for the application. Since the application fetches data from the server over the Internet, it is crucial that there is an Internet connection for the application to function.
- The web portal will be constrained by the capacity of the database. Since the database is shared with the larger system, it may be forced to queue incoming requests and as a result, increase the time it takes to fetch data.
- The computers must be equipped with web browsers such as Internet explorer.
- Execution time for the algorithm should take no longer than one second.
- All Java code shall conform to the Java Code Convention standards.

- Electronic Gadget Recommender will be a sub-component of an Electronic Gadget commercial website.
- Users shall be required to log in to the website to get recommendations.
- Recommendation system shall be available to users 99.9% of the time when the Electronic Gadget commercial website is available.
- The system must be operational for each user. It also needs to give unique recommendations for each user.

2.4 Assumptions and Dependencies

Every system requires some certain parameters to work, to work as per the requirement, our system also requires some parameters, and we assume them as fulfilled before using this system, which is as:

- Customers will have a username and password; else, they'll have to register themselves on our website.
- This software needs user to have complete knowledge of recommendation system and its working.
- Software is dependent on access of Internet, as it is a remote application, it is necessary to have internet access.
- Assume that all the information entered by the user will be correct. If any wrong information is found then the system will notify an alert.
- The system is required to save the generated reports.

3. Specific Requirements

3.1 Interface Requirements

The user needs to click the link to the website. Then he/she needs to register to the system by providing a password and an email, otherwise he/she won't be able to use the Recommender System. Then, to benefit from the Recommender System he/she needs to be active on the website by listening and downloading music, adding his/her favorites to the list or sharing them on Facebook/Twitter.

3.2 Functional Requirements

This section outlines the use cases for each user registered to the website (larger system).

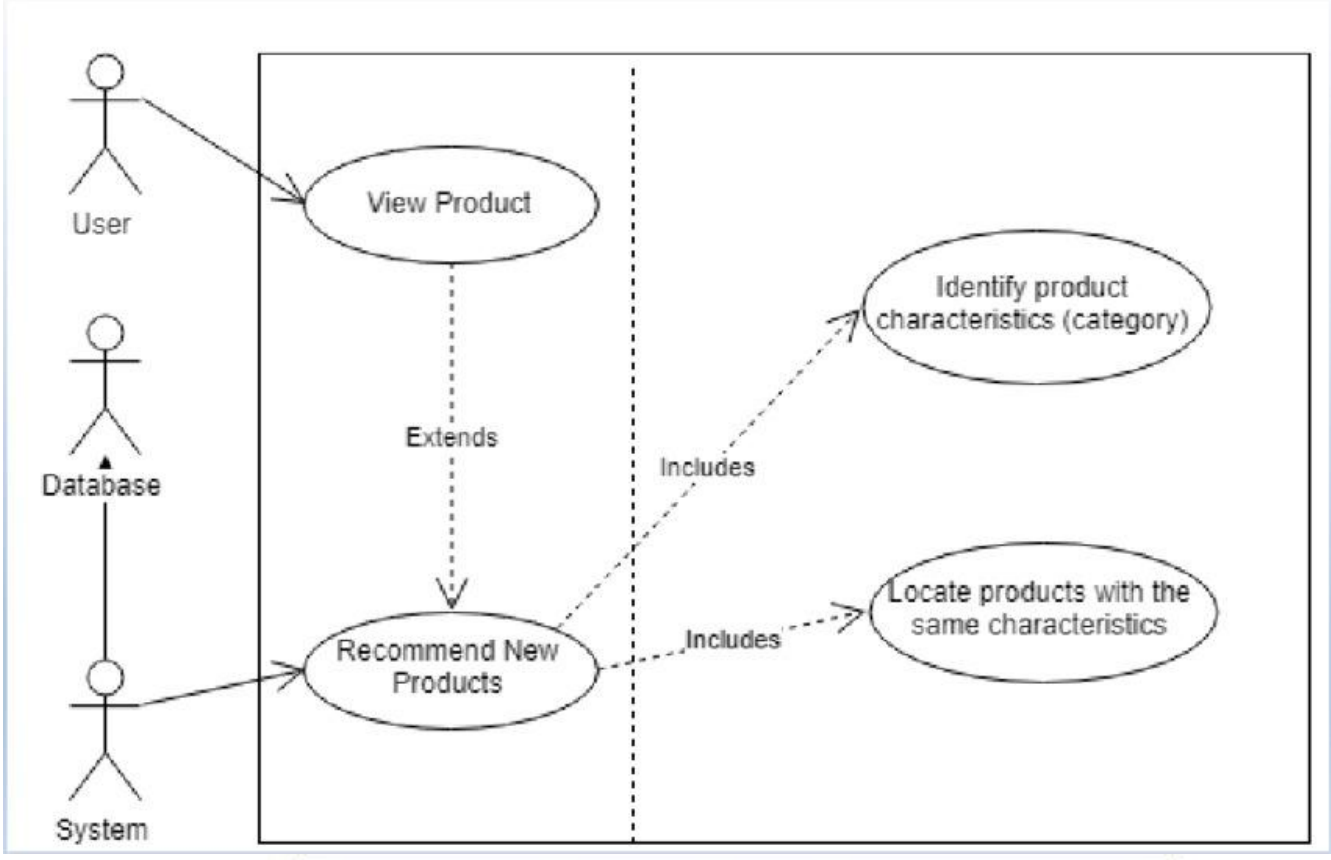

Figure 3- Use Case Diagram

3.2.1 User Use Case

3.2.1.1 Use Case: Generate Data

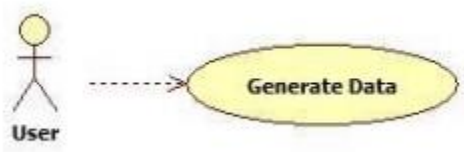

Figure 4- Generate Data

User can search or buy gadgets from the e-commerce website. Product information is collected according to order information, price range, user information, time of action and rating value. This information will fill the database. Every product has a unique order, price, time of action, user and rating value. So, this process will generate practical data (Figure 4).

3.2.1.2 Use Case: Get Recommendation

Figure 5- Get Recommendation

System can suggest product(s) as a recommendation to user based on the dataset which is refined by users' collaborative approach. The main function of our system shows these products based on recommendation algorithms. When a user chooses recommendation part in application, he/she will get the most important point of the project recommendation and project gives user a chance to choose product through recommended tracks according to his/her own previous choices (Figure 5).

3.2.2 Inter-Agent

3.2.2.1 Use Case: Provide Dataset

Figure 6- Provide Dataset

Inter-agent provides dataset, in other words the big data, in cooperation with product purchasing and viewing application (Figure 6).

3.2.2.2 Use Case: Update Dataset

Figure 7- Update Dataset

An e-commerce web application collects millions of data every day. Inter-agent also delivers this data to our web service. These updates are necessary for making more accurate recommendations (Figure 7).

3.2.2.3 Use Case: Integrate Web Service

Figure 8- Integrate Web Service

After the recommendation system project is completed, inter-agent integrates this web service to recommending chatbot application. Then, users will access to our web service and receive recommendations through the application (Figure 8).

3.3 Non-Functional Requirements

3.3.1 Performance Requirements

Performance of making recommendation and updating this recommendation is very important issue because we are aiming to make the system real-timed. In other words, the system should have enough speed that users of the system cannot realize the processing of data. In order to make system real-timed, at the end of viewing a product or rating a product or after purchasing a product, the system shall update recommendations. Besides, our web service should handle multiple users at the same time.

3.3.2 Design Constraints

In the implementation process of this system, Python Programming Language will be the main development language. Since Python is selected to be the main development language, Python Programming Language Code Convention published by Oracle is chosen as a standard for the development process of the system. In the process of the documentation of the system, IEEE standards will be used and UML standard will be used while designing the diagrams.

Since this system will be a part of much larger system, it must be portable to this larger system. That's why portability is one of the most important attributes of this system. Since the larger system is a website that has the potential of increasing its number of users, user traffic and number of products, this system needs to be scale up with the website in the correct order.

Therefore, scalability must be the number one attribute that system will have.

3.3.3 Security

Database has to be reached securely and its data should not be broken. It also should not change except inter-agent updates. Moreover, since our dataset contain some personal information of user such as user id, product he/she bought, security design is important in the web service.

3.3.4 Usability

The scope of the product is widespread. The only requirement is using e-commerce website and downloading web application. Besides, people from every age shall easily use the system.

4. External Interface Requirement

4.1 Hardware Interfaces

- Minimum Requirements:

Client Side			
	Processor	**RAM**	**Disk Space**
Google Chrome v84	Intel Pentium III or AMD -800 MHz	128 MB	100 MB
Server Side			
	Processor	**RAM**	**Disk Space**
NPM v6	Intel Pentium III or AMD -800 MHz	1 GB	3.5 GB
MongoDB- v4+	Intel Pentium III or AMD -800 MHz	256 MB	500 MB (Excluding Data Size)

- Recommended Requirements:

Client Side			
	Processor	**RAM**	**Disk Space**
IE10, Google Chrome 90	All Intel or AMD - 1 GHZ	256 MB	100 MB

Server Side			
	Processor	**RAM**	**Disk Space**
NPM v7	All Intel or AMD - 2 GHZ	2 GB	3.5 GB
MongoDB v4.4	Intel or AMD - 2 GHZ	512 MB	500 MB (Excluding Data Size)

4.2 Software Interfaces

- **Client on Internet**

  Web Browser (Google Chrome v84, Internet Explorer v10)

- **Operating System**

  Microsoft Windows 7+, Linux, Ubuntu v20+, MacOSX

- **Data Base Server**

  MongoDB atlas, supported on any OS

- **Development End**

  Jupyter, IBM Watson, NPM script engine for serving React 16+ application.

4.3 Communication Interfaces

Our system is a web-based application and hence it does not require much. This system supports Google Chrome & Mozilla Firefox web browsers.

5. Data Model and Description

5.1 Data Description

5.1.1 Data Objects

This subsection of the document explains system's classes and their relations with each other. Most of the system functionalities are represented in Figure 9.

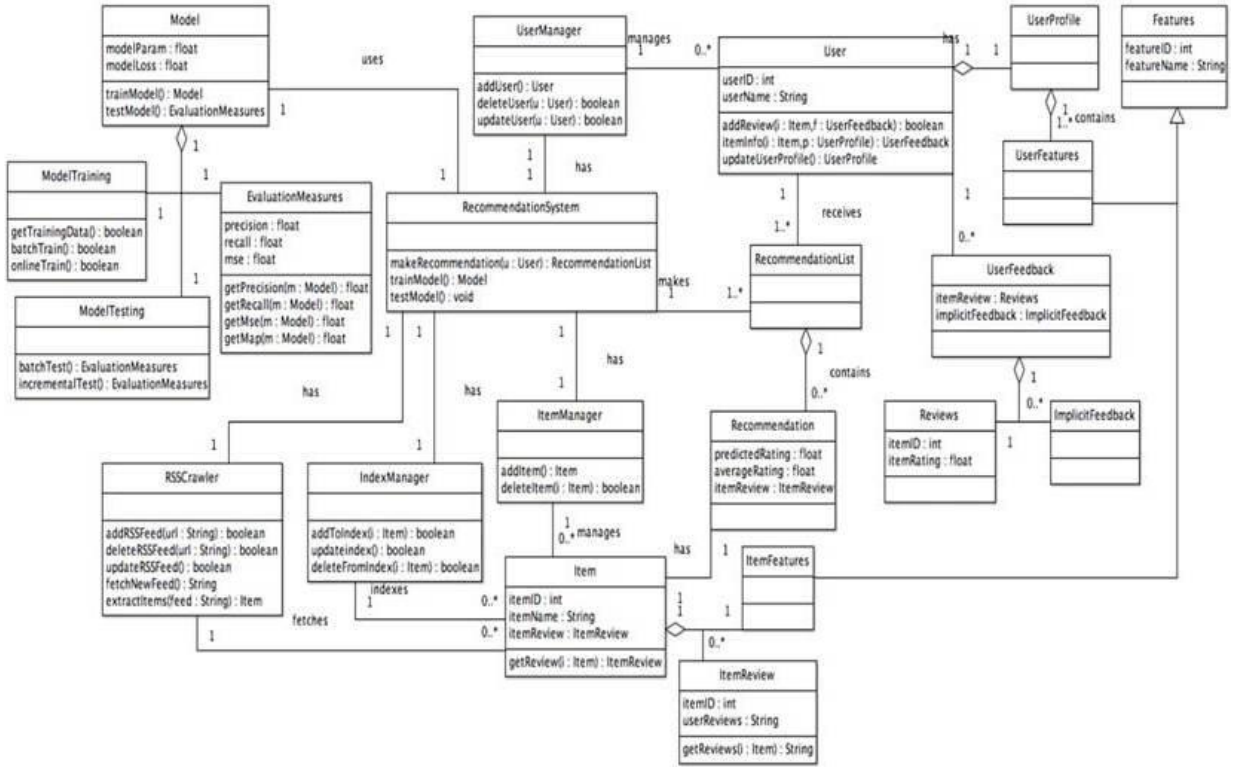

Figure 9- Class Diagram

6. Behavioral Model and Description

6.1 Description for Software Behavior

This subsection describes the major events and states of our software. When the user open the webpage at first, user will see a login screen. In this screen user will required to provide username and password. If the information is valid the user will be able to move on to the main page. On this page user will be able to view and purchase products. Every product that the user purchased will be added to the user history. By checking the user history, every user will be able to view recommendations. Users will be able to logout of the system.

6.2 State Transition Diagrams

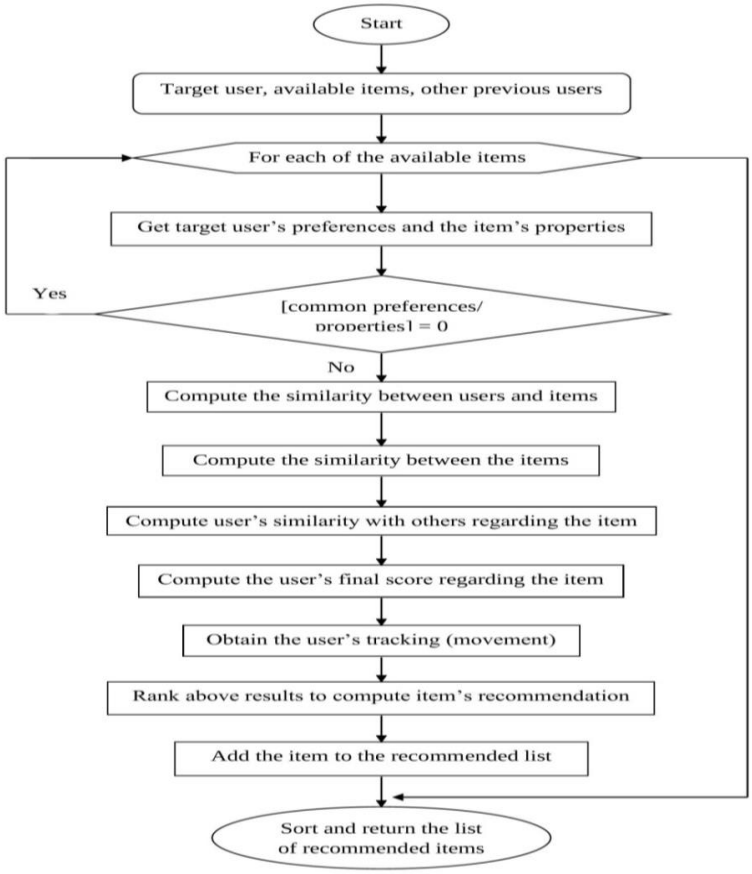

Figure 10- State Transition Flowchart

6.2 Sequence Diagrams

This is the UML sequence diagram of Personalized Recommendation System which shows the interaction between the objects of Users, Customers, Inter-agent, and Databases.

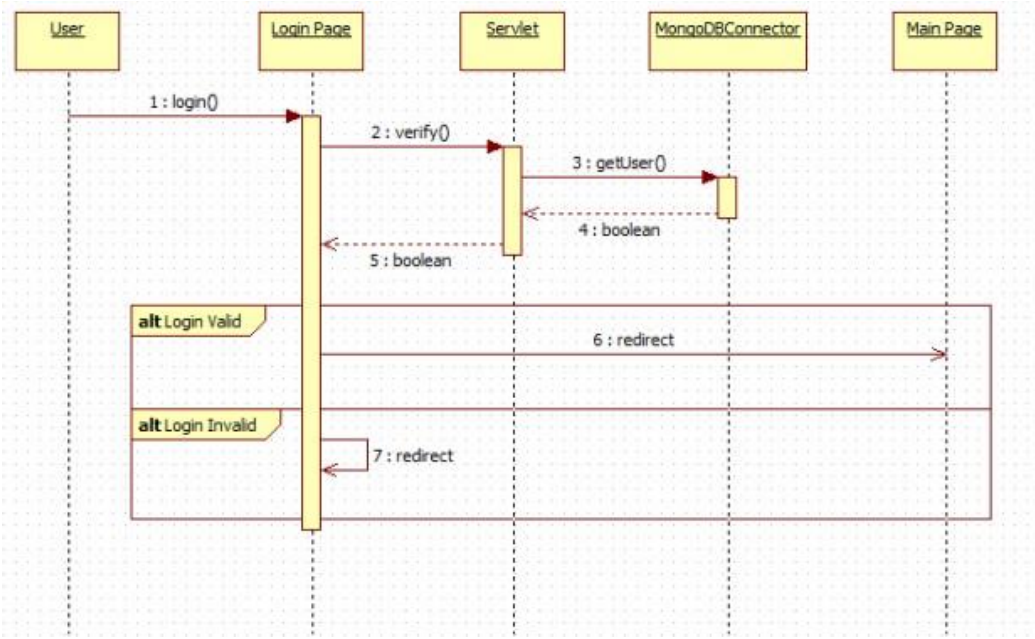

Figure 11- Login Sequence Diagram

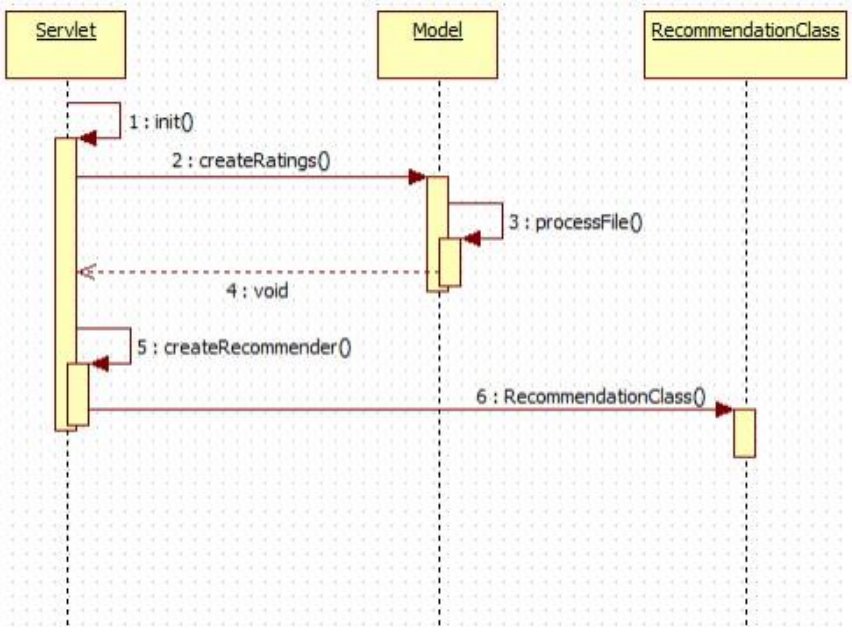

Figure 12- Generate Recommendations Sequence Diagram

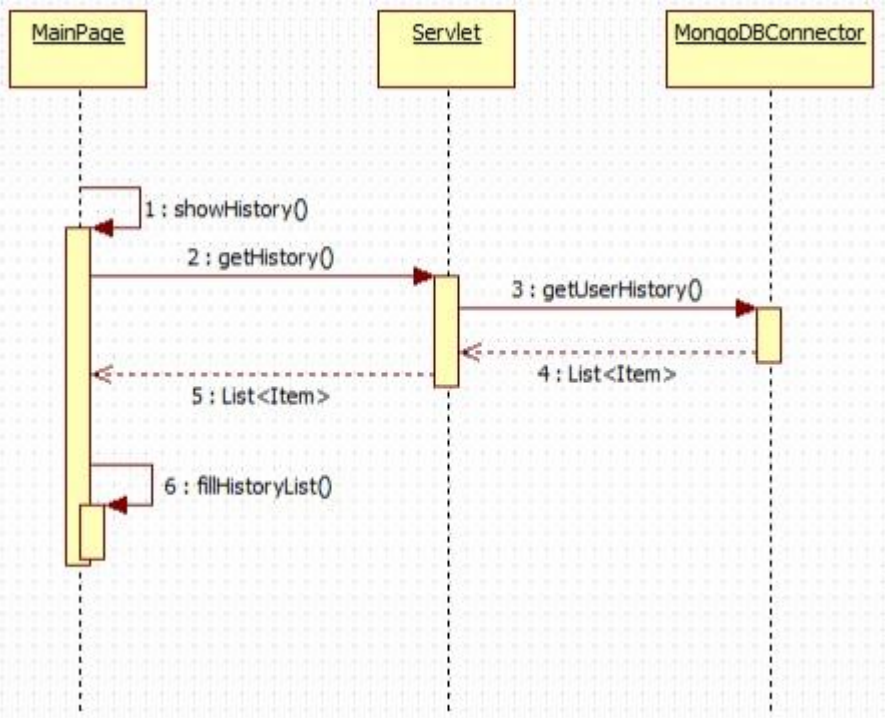

Figure 13- View History Sequence Diagram

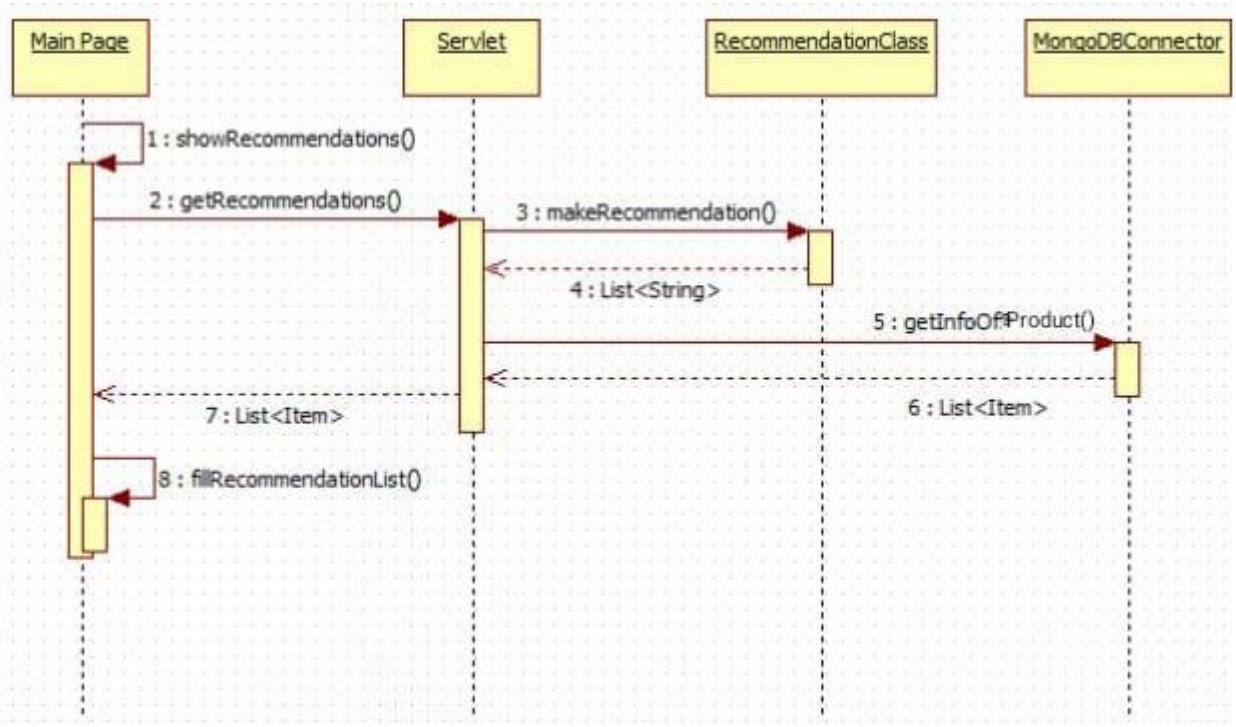

Figure 14- View Recommendation Sequence Diagram

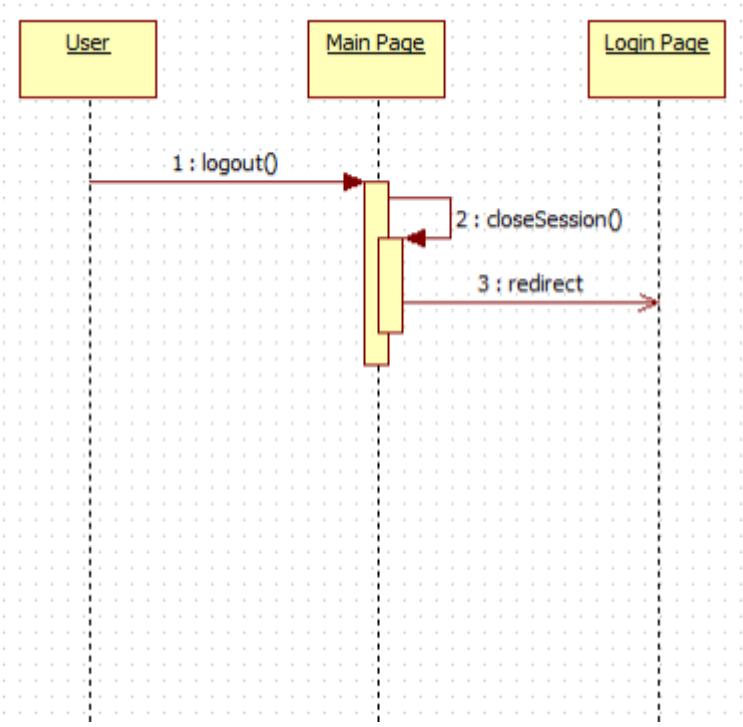

Figure 15- Logout Sequence Diagram

6.3 Block Diagram

6.3.1 Level 0 Data Flow Diagram

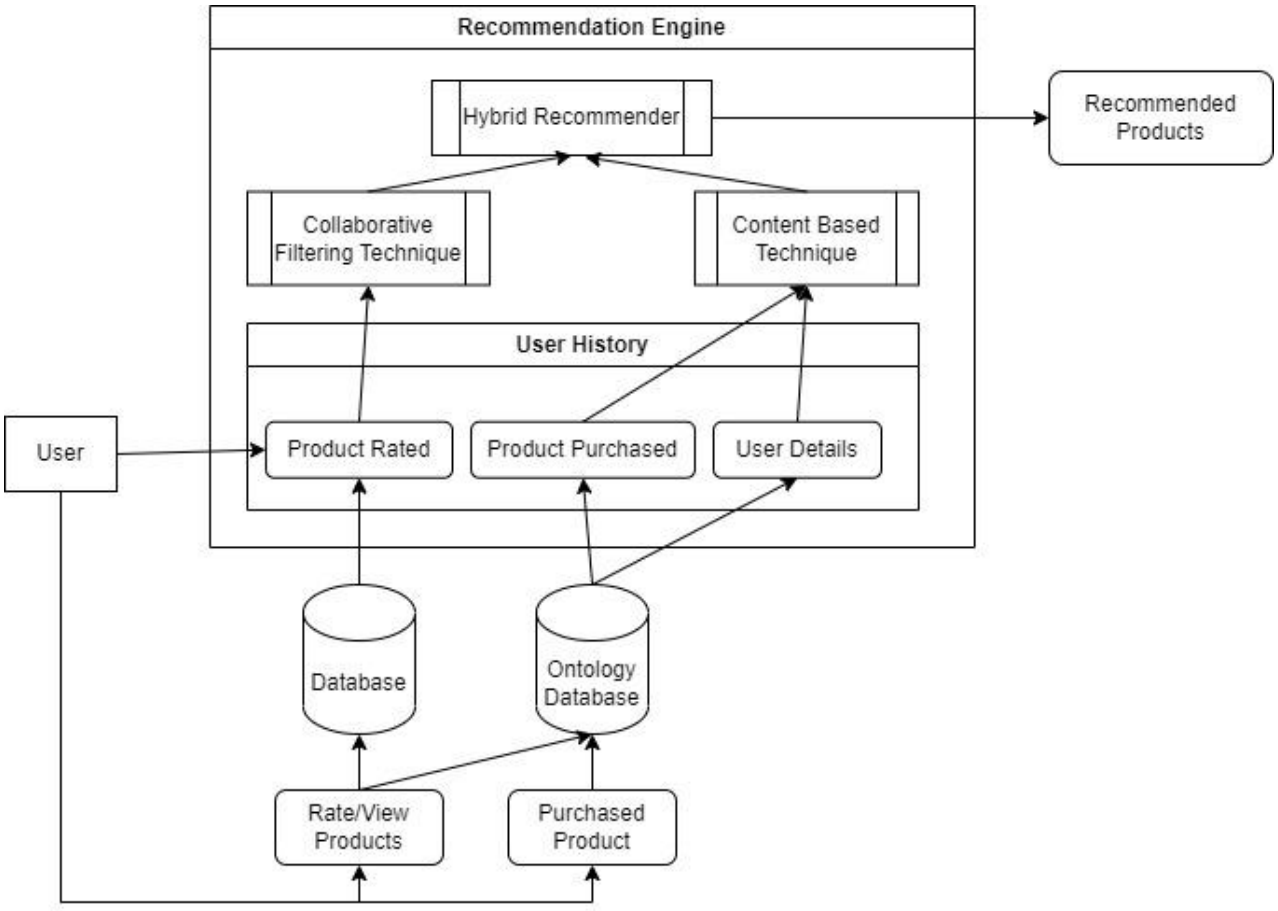

Figure 16- Level 2 Block Diagram

7. Conclusion & Future Scope

Electronic gadgets have become vital in basic need of individual. The advances in technology, makes it necessary to satisfy different functional needs of end-users. Therefore, it is significant to suggest gadgets for the customers based on their personal preference. Recently with rapid development of technology, smart devices and communication networks spring up to cover all the aspect of customer's activity. These data can also be trained and modeled for future use to cope with the upcoming technological innovations.

The goal of this recommender system is to provide the user with items or data of possible interests of gadgets that will help the user make a more informed decision. Recommendations are made after gathering (explicitly or implicitly), processing, and analyzing user or item data. However, developing a recommendation system is a manual and labor-intensive task for programmers, because of the apparent heterogeneity of data available to be processed. One way to facilitate requirements gathering, and consequently recommendation engine development, is the definition of a general user and item model that can adapt to any domain. This project describes a systematic review whose goal is to identify all user and item information used in modern and implemented personalized recommendation system.

In addition, this study resulted in the creation of a user and an item model to be used in the development of RSs. The user model, fully described in the last section, classifies user information into 8 categories: profile, education, professional, medical, social, personality, ratings, and system. The category names clarify what information they depict. The last two categories (ratings and system) concern user ratings of items and implicit information about the user behavior, such as a log of mouse clicks. Similarly, the item model separates information into two main categories: identification and attributes. The first contains information that identifies the item, such as its id or name. The second category is divided into several types of data, such as number, text, or lists that contain the data (quantity, words, and references).

8. References

As Internet is an Ocean of knowledge, we, too, has been helped by the same inter network of system. We've referenced from many a site to get Information/ for Knowledge Gathering to understand the current scenario of the market, below are the references we have got helped from, and we acknowledge the same:

1. Shapira B, Ricci F, Kantor PB, Rokach L (2011) Recommender systems handbook. Springer, New York
   https://link.springer.com/chapter/10.1007/978-0-387-85820-3_1
2. Ben Schafer J, Konstan J, Riedl J (1999) Recommender systems in e-commerce. In: Proceedings of the 1st ACM Conference on Electronic Commerce, 1999, pp 158–166
   https://dl.acm.org/doi/pdf/10.1145/336992.337035
3. Vala Ali Rohani, Zarina Mohd Kasirun and Kuru Ratnavelu, "An Enhanced Content-Based Recommender System for Academic Social Networks", 2014 IEEE Fourth International Conference on Big Data and Cloud Computing.
   https://ieeexplore.ieee.org/abstract/document/7034825/
4. Leo Iaquinta, Marco de Gemmis, Pasquale Lops, Giovanni Semeraro, Michele Filannino and Piero Molino, "Introducing Serendipity in a Content-based Recommender System", Eighth International Conference on Hybrid Intelligent Systems.
   https://ieeexplore.ieee.org/abstract/document/4626624/
5. Zeinab Sharifi, Mansoor Rezghi and Mahdi Nasiri, "A New Algorithm for Solving Data Sparsity problem Based-On Non Negative Matrix Factorization in Recommender Systems", 2014 4th International Conference on Computer and Knowledge Engineering(ICCKE).
   https://ieeexplore.ieee.org/abstract/document/6993356/
6. Cunningham P, Bergmann R, Schmitt S, Traphoner R, Breen S, Smyth B. "WebSell: Intelligent sales assistants for the World Wide Web. In: Proceedings CBR in ECommerce, Vancouver BC; 2001
   http://www.wi2.uni-trier.de/shared/publications/1999_cunninghambergmann.pdf
7. Yograj Meena, Monika, Parveen Kumar, Ambalika Sharma, "Product Recommendation

System Using Distance Measure of Product Image Features", 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS) https://ieeexplore.ieee.org/abstract/document/8663113/

8. Chisnall, P. (1973) "Marketing Research: Analysis and Measurement." McGraw-Hill