
Software Requirement Specification For AI Based Personalized Electronic Gadgets Recommendation System

**Project Guide:**

Mr. Praveen Kumar Yadav
(Assistant Professor)

Submitted to:

Mrs. Sanju Choudhary
(Associate Professor)

Prepared by:

Sneha Khandelwal (19ESKIT090)
Sonali Kumawat (19ESKIT091)
Vaibhav Jain (19ESKIT096)
Mishu Jain (19ESKIT304)

Department of Information Technology

**Swami Keshvanand Institute of Technology, Management and Gramothan, Jaipur
Rajasthan Technical University, Kota**

Table of Contents

1. Introduction.....	1
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions.....	1
1.4 Project Scope.....	2
1.5 Aim.....	2
2. Overall Description.....	3
2.1 Product Perspective.....	3
2.2 Product Functions.....	3
2.3 User Classes and Characteristics.....	4
2.4 Design and Implementation Constraints.....	4
2.5 Assumptions and Dependencies.....	4
3. System Design.....	5
3.1 System Analysis.....	5
3.2 Use Case Diagram.....	5
3.3 Class Diagram.....	7
3.4 Sequence Diagram.....	7
3.5 Activity Diagram.....	9
3.6 State Transition Diagram.....	10
4. External Interface Requirements.....	11
4.1 User Interfaces.....	11
4.2 Hardware Interfaces / Software Interfaces.....	11
4.3 Block Diagram.....	13
4.4 Communication Interfaces.....	13
5. Functional Requirements.....	14
5.1 UI/UX requirement:.....	14
5.2 Personalized recommendation system:.....	14
6. Other Nonfunctional Requirements.....	15
6.1. Non-functional requirement:.....	15
6.2 Usability requirement:.....	16
7. References.....	17

1. Introduction

1.1 Purpose

The purpose of the AI-based Personalized Electronic Gadgets Recommendation System is to provide personalized recommendations for electronic gadgets, such as smartphones, laptops, and tablets, to users based on their preferences, usage patterns, and other relevant factors. The system aims to improve the shopping experience for users and increase sales for the online store.

The system will collect and store data on electronic gadgets, including specifications, features, prices, and so on. By analyzing the content features of the gadgets, such as brand, operating system, processor, RAM, storage, display size, and other relevant factors, the system will be able to understand the preferences and needs of individual users and provide them with relevant and tailored recommendations for electronic gadgets that align with their preferences.

The system will utilize content-based filtering algorithms to generate recommendations. These algorithms will analyze the content features of the gadgets and match them with the user preferences to recommend gadgets that are most likely to meet their requirements. The recommendations will be displayed to users in a user-friendly interface, providing detailed product information to help users make informed decisions.

1.2 Document Conventions

Bold-faced text has been used to emphasize section and subsection headings. Highlighting is to point out words in the glossary and italicized text is used to label and recognize diagrams.

1.3 Intended Audience and Reading Suggestions

This project is a prototype for the Gadget Recommendation system for electronic items and it is restricted within the college premises. This has been implemented under the guidance of college professors. This project is useful for various users in recommending the best possible gadgets according to their interests.

1.4 Project Scope

The purpose of the system is to provide personalized recommendations for electronic gadgets, such as smartphones, laptops, tablets, etc., to individual users based on their preferences using content-based filtering techniques.

The system will focus on providing recommendations for a wide range of electronic gadgets and utilize content-based filtering techniques to generate recommendations based on the content features of the gadgets, such as brand, operating system, processor, RAM, storage, camera quality, display size, and other relevant factors. The system will have a user-friendly interface that displays detailed product information, comparison features, and other relevant information to help users make informed decisions.

1.5 Aim

The main aim of this developed project is to:

- Provide individual users with personalized recommendations for electronic gadgets based on their preferences.
- Utilize content-based filtering techniques to capture user preferences from browsing history, and other relevant data.
- Generate recommendations based on content features of the gadgets, such as brand, operating system, processor, RAM, storage, camera quality, display size, etc.
- Present users with a user-friendly interface displaying detailed product information, and other relevant information.
- Enhance the decision-making process for users when it comes to purchasing electronic gadgets.
- Offer a seamless and personalized experience to users, helping them make informed decisions about the right gadget for their needs.

2. Overall Description

2.1 Product Perspective

The tech world has become more competitive than ever with a lot of manufactures trying to produce the best for their customers. The endless increase in the options space left the customers in a dilemma of choosing the right gadget for them. The major factors that influence customers in selecting an electronic gadget to use include: Brand, RAM, battery life, price, year of release, innovative features, personal recommendation and so on.

Gadget recommendation system is a software tool designed to give recommendations to things or substances a particular user would like to purchase or engage with. We propose an AI based recommendation system specially designed for electronic gadgets. It has a simple UI with many functions. It is a user-friendly app with profile free interaction. It provides a superior customer experience that can deliver the level of personalization which will impress consumers with individualized options. The user searches for gadgets according to his preference by entering certain specifications in the filter option. If the user is satisfied with the recommended gadget he submits a response. Else he repeats the step.

Item details:

The UI consists of the set of details about the gadgets which needs to be entered by the user to get the basic idea of the user's interest. It includes the name, brand, internal specifications, camera, etc. The system responds with the results containing the specific recommendations

2.2 Product Functions

The purpose of the AI-based Personalized Electronic Gadgets Recommendation System is to provide personalized recommendations for electronic gadgets, such as smartphones, laptops, and tablets, to users based on their preferences, usage patterns, and other relevant factors. The system aims to improve the shopping experience for users and increase sales for the online store.

The system will collect and store data on electronic gadgets, including specifications, features, prices, and so on. By analyzing the content features of the gadgets, such as brand, operating system, processor, RAM, storage, display size, and other relevant factors, the system will be able to understand the preferences and needs of individual users and provide them with relevant and tailored recommendations for electronic gadgets that align with their preferences. The system will utilize content-based filtering algorithms to generate recommendations. These algorithms will analyze the content features of the gadgets and match them with the user preferences to recommend gadgets that are most likely to meet their requirements. The recommendations will be displayed to users in a user-friendly interface, providing detailed product information to help users make informed decisions.

- If the recommended item is not up to the expected quality of the user, then the user can make a complaint in order to get better suggestions from the AI about the gadgets. Further, response is to be given by the AI authority to his complaint.
- Users can assist in the better functioning of the product.
- Users can give their feedback for the services provided to them.

2.3 User Classes and Characteristics

Users of the system should be able to get the recommendations of the various devices according to their interest. The recommendations should not be improper data processed by the fault of the engine. The system will support two types of user privileges, users and Admin (AI Authority). Users will have access to user functions, and the admin will have access to both user and item management functions.

The user should be able to do the following functions:

- Access to the items present already in the system.
- Check for other items of his choice.
- If the recommendation is not properly optimized a report can be issued containing all the issues of the user.
- Can personalize the results of the recommendation

2.4 Design and Implementation Constraints

- Data required in CSV format
- Implement AI and Machine Learning algorithms

2.5 Assumptions and Dependencies

- A data sheet containing details of the electronic gadgets
- Another input from the user using the service.

3. System Design

Now we have to design our system before implementing it in such a way that it can execute all the work we want without any loss of data. For this purpose, we have different kinds of diagrams to represent the system interaction flow of the data storage.

3.1 System Analysis

The system analysis phase is considered to be one of the most important phases in the system development life cycle. It is immensely important that the software developer makes thorough study of the existing system. Thorough study of the system is made and needs i.e. features that are critical to system success and users wants (i.e. features that would be good but not essential) are brought out. The study will enable the developer to know the intricacies of the existing system.

Requirement analysis is done in order to understand the problem which the S/W system is to solve e.g., the problem could be automating the existing manual system or developing a completely new automated system or a combination of the two. For large systems having a large number of features and the need to perform many different tasks, understanding the requirements of the system is a major task. The emphasis in requirement analysis is on identifying what is needed from the system, and not how the system achieves its goal.

3.2 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the framework that demonstrates the relationship between the user and the different use cases in which the user is involved. A use case diagram can recognize the different types of users of a system and the different use cases and will frequently be accompanied by other types of diagrams as well.

The following figure is a use case diagram which illustrates the interactions between the user and various functionalities that are present in the application and how the user accesses the information.

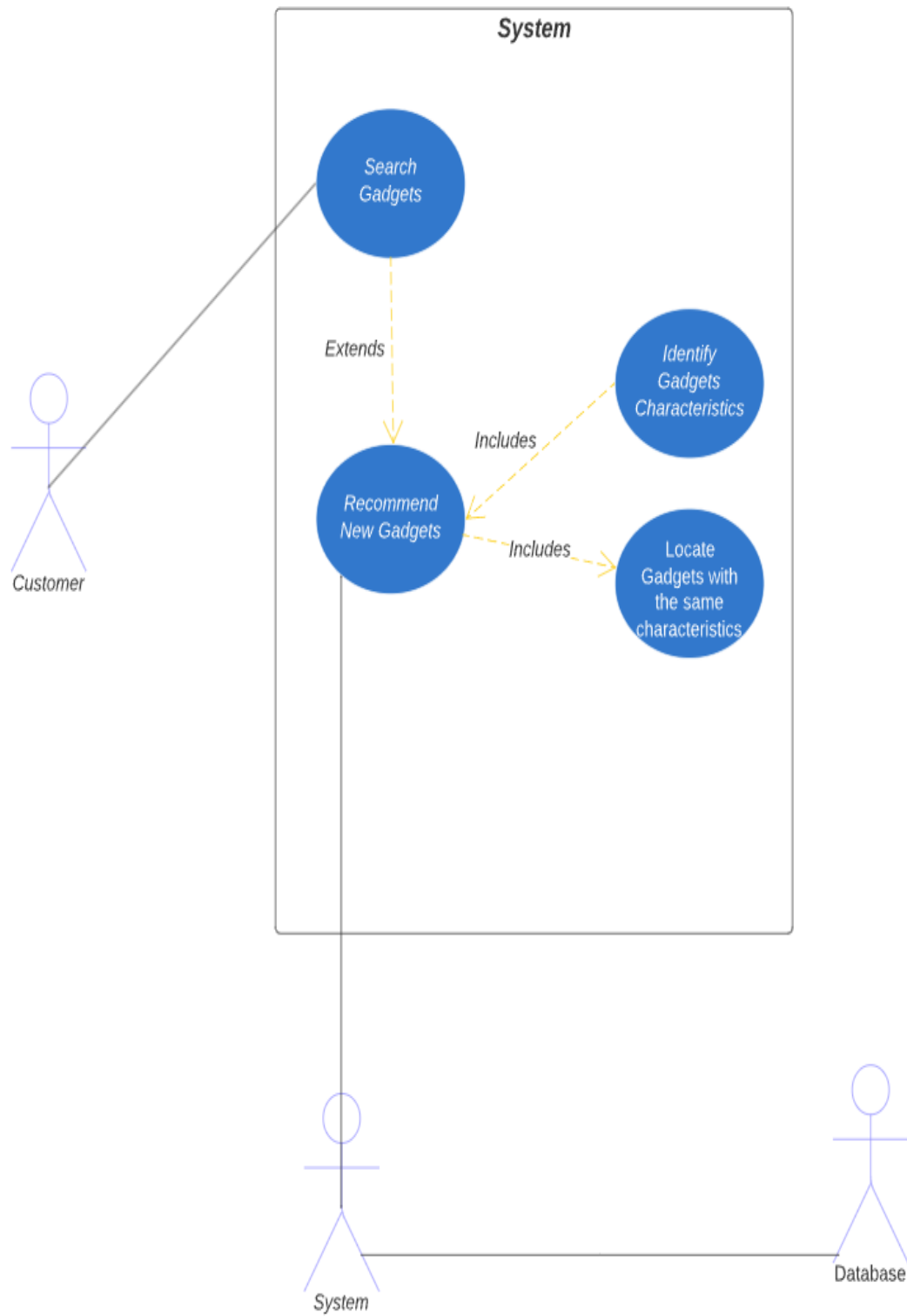


Fig. - 1 Use Case Diagram

3.3 Class Diagram

A class diagram is an outline of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class characterizes the methods and variables in an object, which is a particular entity in a program or the unit of code representing that entity.

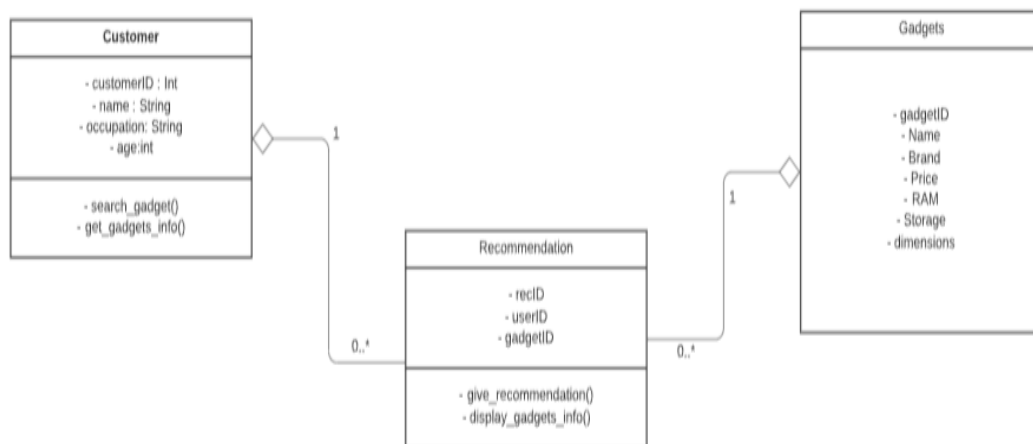


Fig. -2 Class Diagram

3.4 Sequence Diagram

A Sequence diagram is a connection chart that shows how procedures work with each other and in what request. It is a build of a Message Sequence Chart. A sequence diagram shows object collaborations orchestrated in time arrangement. It portrays the articles and classes included in the situation and the succession of messages traded between the items expected to do the situation's usefulness. Arrangement graphs are normally connected with use cases acknowledged in the Logical View of the framework, a work in progress. Succession outlines are now and then called occasion charts or occasion situations.

The following figure shows the sequence diagram showing the data sequence of the application.

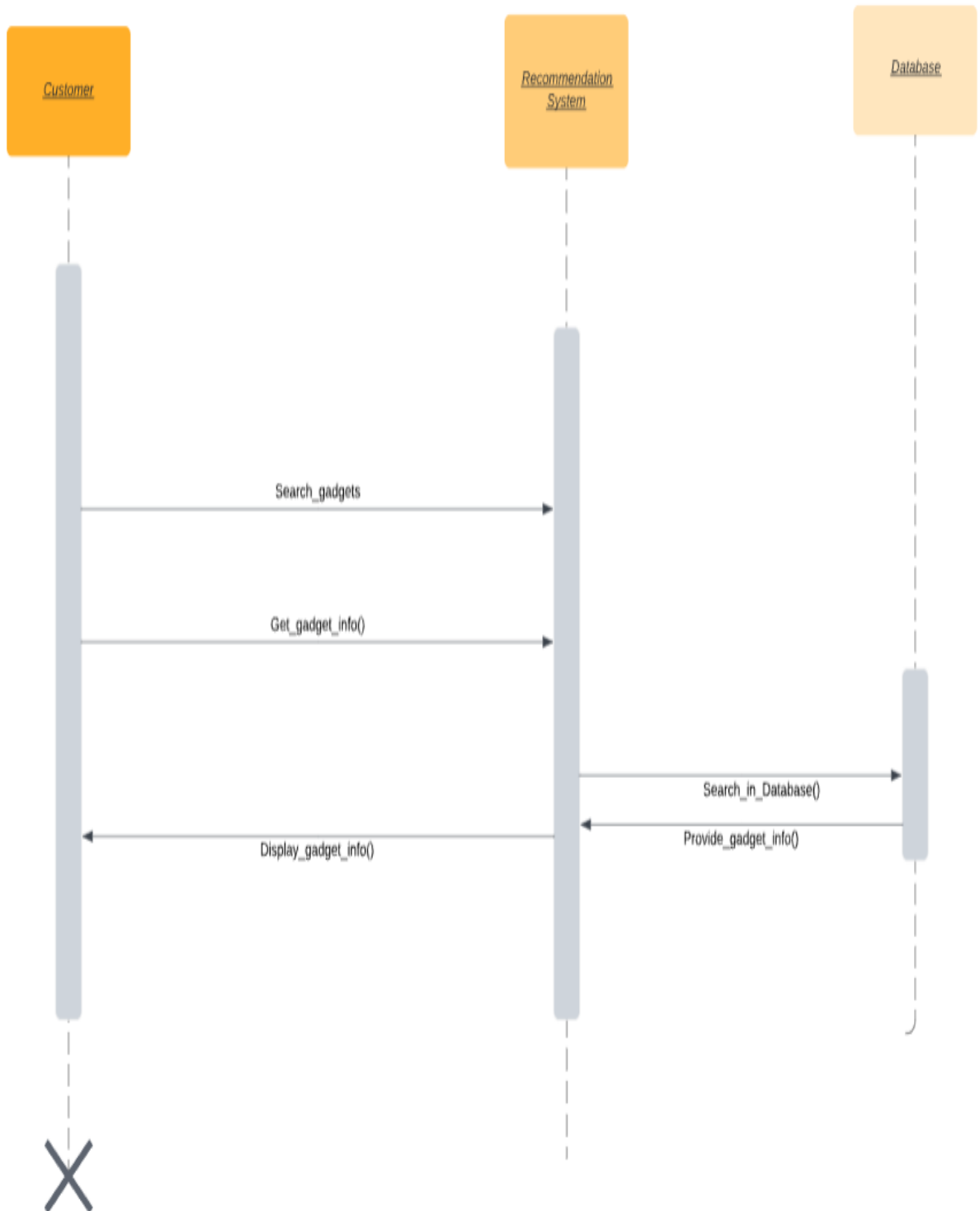


Fig. - 3 Sequence Diagram

3.5 Activity Diagram

Activities diagrams are graphical representations of work processes of stepwise exercises and actions with backing for decision, emphasis and simultaneousness. In the Unified Modeling Language, movement outlines are planned to display both computational and authoritative procedures (i.e., workflows). Activity charts demonstrate the general stream of control. The following figure shows how the data is sent and collected in the central server using an activity diagram.

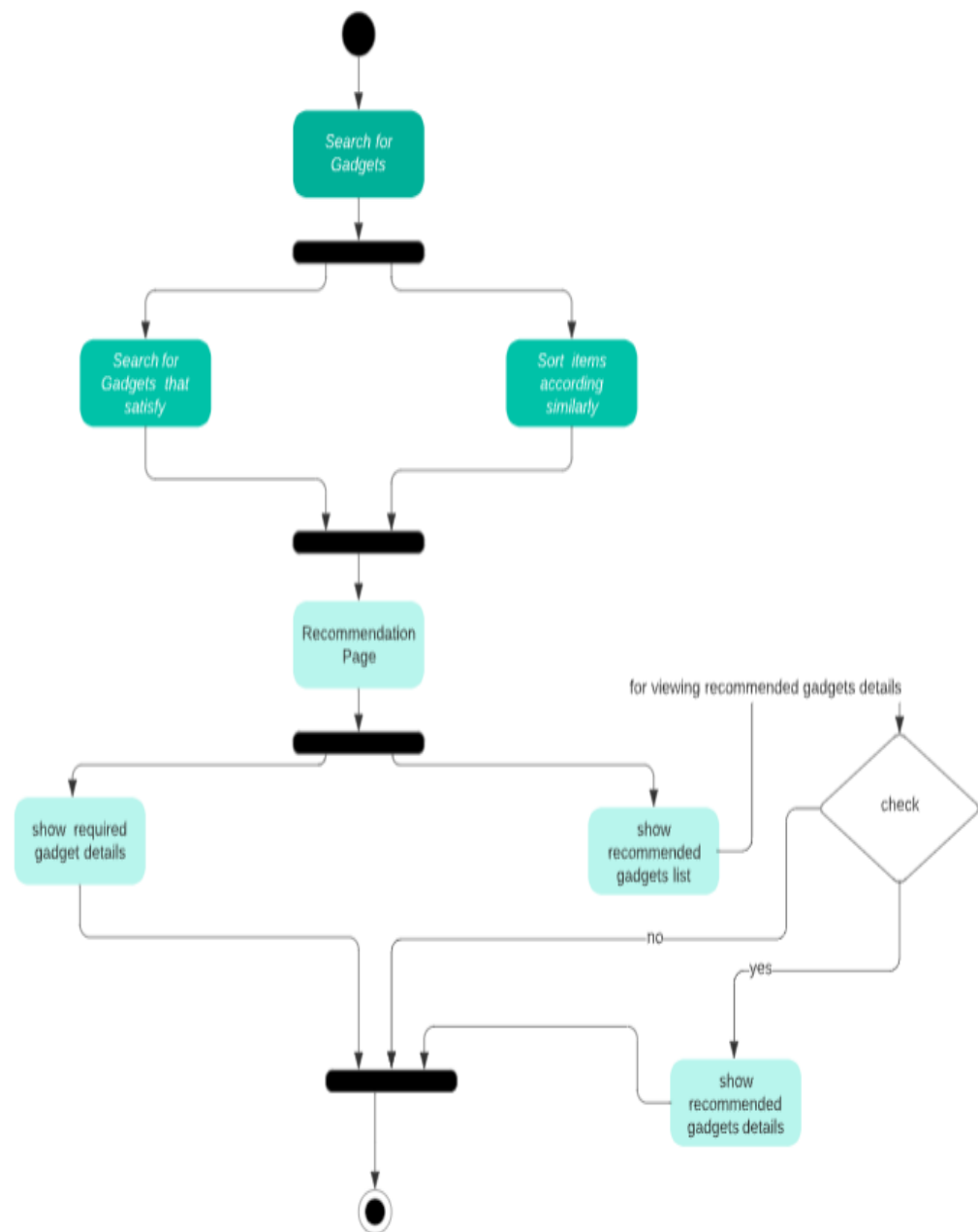


Fig. - 4 Activity Diagram

3.6 State Transition Diagram

The state transition diagram provides an overview of the different states and transitions in the system, showcasing the flow of interactions between the user and the recommendation system, and highlighting the dynamic nature of the system as it adapts to user inputs and updates its recommendations.

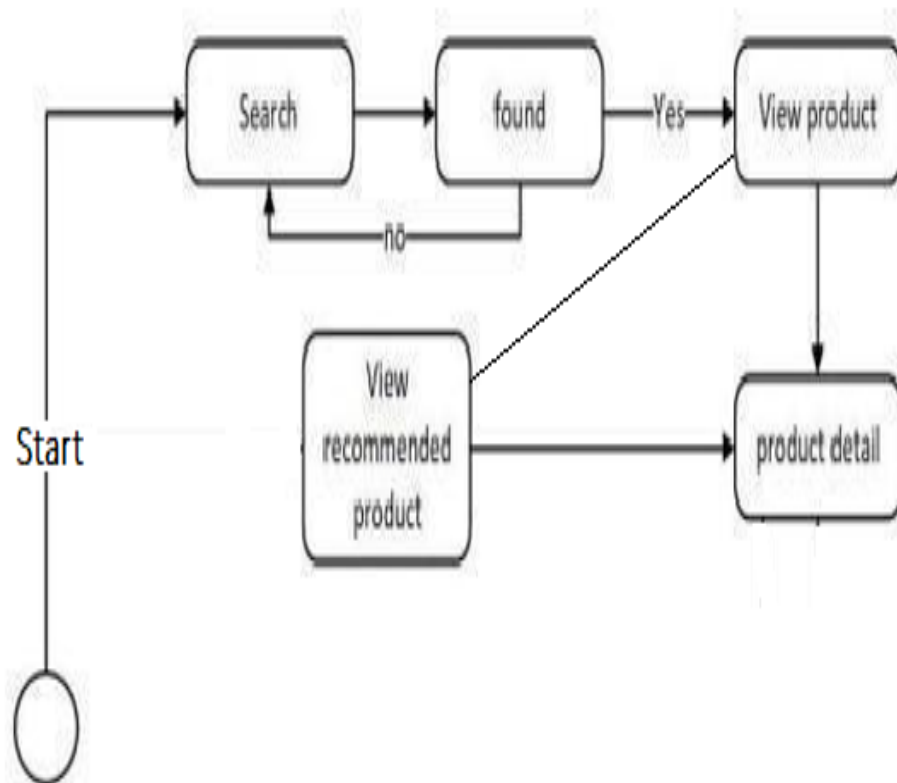


Fig. - 5 State Transition Diagram

4. External Interface Requirements

4.1 User Interfaces

The Recommendation System user interface has been specifically designed with their customers in mind, giving them convenience to get the item recommended to them.

The home screen offers a menu with a list of functions that the device performs. The user can select one of the options on the menu, and is taken to the respective screen. Every screen displays the menu on the bottom. The user can click on any one of the options and is taken to the screen of their choice. In addition, clicking on the power button displays the home screen with the menu options.

The device offers easy scroll options to navigate the screens efficiently. To scroll down any screen, simply touch the scroll bar on the screen, and roll down. If the user does not know how to use any functionality or has any queries, the help option can be used. The help screen contains a text field to enter search terms. A list of search results pertaining to the query is displayed.

4.2 Hardware Interfaces / Software Interfaces

Front-end:

For the front-end we are using react.js framework. React is an open source, JavaScript library for developing user interface (UI) in web applications. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Back-end:

For the back-end we are using python. Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. It is also the high demand language for machine learning and data science-based projects. The libraries like flask can be used to connect web pages with python-based machine learning models. Flask is an API of Python that allows us to

build up web-applications. It was developed by Armin Ronacher. Flask's framework is more explicit than Django's framework and is also easier to learn because it has less base code to implement a simple web Application. Flask is based on the WSGI (Web Server Gateway Interface) toolkit and Jinja2 template engine. Also, to run the model we are using a jupyter notebook. The Jupyter Notebook is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience. program used to mix code, comments, and visualizations in an interactive document called notebook that can be shared, reused, and reworked in a web browser. Jupyter Notebook (formerly I Python Notebooks) is a web-based interactive computational environment for creating notebook documents. A Jupyter Notebook document is a browser-based REPL containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots.

Following are the software used for online application.

Software used	Description
Operating system	We have chosen the Windows operating system.
Database	To save the item records, we have chosen a CSV File.
Tools/IDE	To implement the project, we have chosen Visual Studio Code for its more interactive support.
Platform	Web Based Application
Technologies Used	HTML, CSS, JS, FLASK, Python

Fig. - 6 Software Applications

4.3 Block Diagram

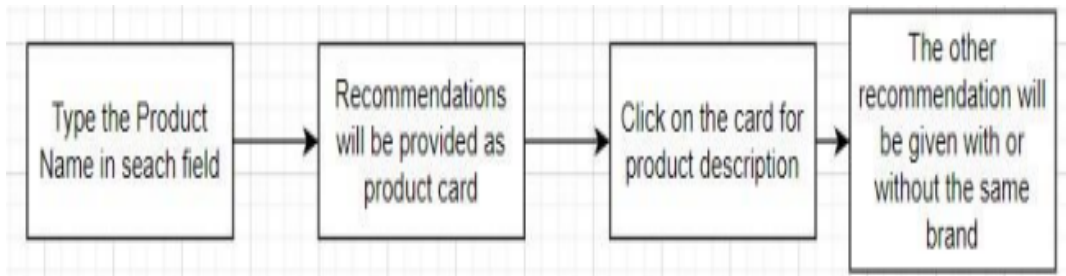


Fig. - 7 Block Diagram

4.4 Communication Interfaces

The communication interface for the AI-based Personalized Electronic Gadgets Recommendation System involves the interaction between various components and modules of the system. These components may include the frontend user interface, backend server, recommendation engine, and any external APIs or services used for data retrieval or processing.

The communication interface can be categorized into two types:

User-System Interface: This interface involves the interaction between the system and the end user. It includes the input and output mechanisms through which the user interacts with the system. The user provides inputs such as preferences, feedback, or gadget selection through the frontend user interface, which may include web pages, mobile apps, or other interactive interfaces. The system processes the user inputs and generates personalized gadget recommendations, which are displayed to the user as output through the same user interface.

System-System Interface: This interface involves the communication between different components and modules of the system. For example, the frontend user interface communicates with the backend server to send user inputs and receive recommendation results. The backend server communicates with the recommendation engine, which may include algorithms, databases, or external APIs, to generate personalized recommendations based on user inputs and other factors. The recommendation engine may also communicate with external data sources or services for data retrieval or processing purposes.

The communication interface may utilize various protocols, APIs, or standards for data

exchange, such as HTTP, RESTful APIs, JSON, XML, or other formats depending on the system architecture and technologies used. It is essential to ensure that the communication interface is efficient, secure, and reliable to facilitate seamless interaction between the different components of the system and deliver accurate recommendations.

5. Functional Requirements

5.1 UI/UX requirement:

Attractive design, easy to navigate and speedy page loading are three features ecommerce site must have. Apart from that, use of color, fonts and images also plays a vital role to engage the user. These issues must be resolved. Mobile devices are the great source of users nowadays. So, this system must be mobile friendly and scalable to different screen sizes. This system must have features of search and filter products based upon categories and subcategories. Customers always want to hear from other customers how the shopping suits their needs and expectations. To solve this problem review and ratings must be used. This system must have features of shopping cart and checkout to both logged in user and anonymous user.

User-friendly interface: The system should have a simple and intuitive interface that is easy for users to navigate.

Responsive design: The system should have a responsive design that works well on a variety of devices and screen sizes.

Visual aids: The system should use visual aids such as images, videos, and interactive graphics to help users better understand recommended gadgets.

5.2 Personalized recommendation system:

Most popular systems are dynamic and personalized to the user. It is one of the key factors to engage the user with the system. The products are recommended to users based upon their preferences, user history and ratings. The already rated product must not be recommended to the user.

Search Functionality: The home page should allow users to search for gadgets based on keywords, filters, and sorting options, and display accurate and relevant search results.

Product catalog: The system should have a comprehensive catalog of gadgets that includes information such as product features, specifications, and prices.

Recommendation engine: The system should use machine learning algorithms to analyze user data and recommend products that are likely to be of interest to the user.

Personalization: The system should be able to customize recommendations based on user preferences and behavior.

Multi-platform accessibility: The system should be accessible across multiple platforms, such as mobile devices and desktop computers.

Product Details: The recommendation page should display detailed information about each recommended gadget, including product name, image, and other specifications.

6. Other Nonfunctional Requirements

6.1 Non-functional requirement:

Nonfunctional requirements specify whether a particular task will be performed or not. Some of the non-functional requirements are listed below :

1. Speed:

Speed is actually much more complicated than it seems. Speed is one of the nonfunctional requirements that must be considered. The speed may lag because of many reasons. Some of the common reasons for slow load is due to high resolution unoptimized images, complex order entry process, use of many API calls to render, unclean code etc. All of these issues must be solved.

2. Security and privacy:

Since this web app uses the most sensitive data of the user. The system must be protected from vulnerabilities. The authentication must be secured from data breach. To protect the system from code manipulation, the software must be coded with much complexity. Privacy must be protected. Policies, terms and conditions must be able to update from the admin side.

6.2 Usability requirement:

1. Ease of use:

The system must be easy to use and understand the system or the casual user. The navigation and function must be simple.

2. Satisfaction and memorability:

The design of the system must be pleasant to view and the user must be able to recall the site when the user returns after a certain period of time.

3. Graphics:

Graphics is considered as one of the components that helps users to better connect to the system. It mostly deals with data rendering and displaying high quality data. The code must be optimized and should be able to show similar quality in any hardware device.

6.3 Software Quality Attributes

- **Availability**-the item should be available to the user.
- **Correctness**-the list of the items to be recommended should be correct according to the details provided by the user.
- **Usability**-the recommended items should satisfy a maximum number of user needs.
- **Maintainability**-admin should maintain the database and store in updated form.
- **Portability**- the application should be portable to android, iOS and Windows.
- **Reliability**- the system should give 98% correct search results out of 1000

searches during testing.

- **Extendibility**-the application should be easy to extend, code should be written in such a way that it favors implementation of new functions.

7. References

<https://www.researchgate.net>

<https://indatalabs.com/blog/ai-based-recommender-system>

<https://thingsolver.com/introduction-to-recommender-systems/>