A

PROJECT REPORT

ON

"GADGET RECOMMENDATION SYSTEM"

SUBMITTED TO SHIVAJI UNIVERSITY, KOLHAPUR

IN THE PARTIAL FULFILLMENT OF REQUIREMENT FOR THE AWARD OF DEGREE BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY

MISS. MRUNAL PATIL	**18UCS076**
MR. VRUSHABH RAJMANE	**18UCS094**
MR. ONKAR RANBHARE	**18UCS095**
MR. OMKAR SHINDE	**18UCS109**
MISS. AAYUSHI BALDI	**18UET002**
MISS. SAMRUDDHI SHETE	**18UET058**

UNDER THE GUIDANCE OF

DR. S. K. SHIRGAVE

DKTE

Promoting Excellence in
Teaching, Learning & Research

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DKTE SOCIETY'S TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI

2021-2022

D.K.T.E. SOCIETY'S

TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI
(AN AUTONOUMOUS INSTITUTE)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that, project work entitled

"GADGET RECOMMENDATION SYSTEM"

is a bonafide record of project work carried out in this college by

MISS. MRUNAL PATIL	**18UCS076**
MR. VRUSHABH RAJMANE	**18UCS094**
MR. ONKAR RANBHARE	**18UCS095**
MR. OMKAR SHINDE	**18UCS109**
MISS. AAYUSHI BALDI	**18UET002**
MISS. SAMRUDDHI SHETE	**18UET058**

is in the partial fulfilment of award of degree Bachelor of Technology in Computer Science & Engineering prescribed by Shivaji University, Kolhapur for the academic year 2021-2022.

PROF.(DR.) S. K. SHIRGAVE
(PROJECT GUIDE)

PROF.(DR.) D.V. KODAVADE
(HOD CSE DEPT.)

PROF.(DR.) P.V. KADOLE
(DIRECTOR)

EXAMINER: _____

DECLARATION

We hereby declare that, the project work report entitled “Gadget Recommendation System” which is being submitted to D.K.T.E. Society’s Textile and Engineering Institute Ichalkaranji, affiliated to Shivaji University, Kolhapur is in partial fulfilment of degree B.Tech.(CSE). It is a bonafide report of the work carried out by us. The material contained in this report has not been submitted to any university or institution for the award of any degree. Further, we declare that we have not violated any of the provisions under copyright and Piracy / Cyber / IPR Act amended from time to time.

MISS. MRUNAL PATIL	18UCS076
MR. VRUSHABH RAJMANE	18UCS094
MR. ONKAR RANBHARE	18UCS095
MR. OMKAR SHINDE	18UCS109
MISS. AAYUSHI BALDI	18UET002
MISS. SAMRUDDHI SHETE	18UET058

ACKNOWLEDGEMENT

With great pleasure we wish to express our deep sense of gratitude to Dr. S. K. Shirgave for his valuable guidance, support and encouragement in completion of this project report.

Also, we would like to take the opportunity to thank our head of department Dr. D. V. Kodavade for his co-operation in preparing this project report.

We feel gratified to record our cordial thanks to other staff members of the Computer Science and Engineering Department for their support, help and assistance which they extended as and when required.

Thank you,

MISS. MRUNAL PATIL	18UCS076
MR. VRUSHABH RAJMANE	18UCS094
MR. ONKAR RANBHARE	18UCS095
MR. OMKAR SHINDE	18UCS109
MISS. AAYUSHI BALDI	18UET002
MISS. SAMRUDDHI SHETE	18UET058

ABSTRACT

Customer often feels difficult to describe what he needs when he intends to buy a high-tech product with complex features, such as smartphone, notebook, camera, PC, etc. It is because the most of users are less familiar with the technical features of this product types. While recommender systems are in widespread use, the customers still experience problems. Many recommender systems produce recommendations which the customers find unsatisfactory. Naturally, users often express their needs based on functional requirements (need smartphone for gaming, reading e-book, online activity, etc.). Considering these problems, our project proposes a new approach for products recommendation based purely on user requirements, to produce proper recommendation. We perceive an opportunity for knowledge-based recommender systems with a combination of Euclidean distance to gain leverage on recommendation tasks by using explicit models of both the user of the system and the products being recommended. This differs from previous systems which, when they use a user model, have used one that is inferred from the ratings given by that user (i.e., an implicit user model). The proposed approach also gives a more flexible recommendation process, since it considers the products that have better quality than the quality level that user wants.

INDEX

Contents

1. Introduction

1 Introduction

Recommendation systems are software agents that bring out the interests and preferences of individual users and make recommendations accordingly. Also it aims to predict user's interests and recommend product items that quite likely are interesting for them. These are basically the systems that recommend things like videos, books, music, shopping items and even people. They have the potential to support and improve the quality of the decisions users make while searching for and selecting things online. In Database large volume of information is present, so recommender systems filterers the most important information based on the data provided by a user and other factors that take care of the user's preference and interest. Recommendation Systems have become excessively popular in recent times with their presence and increase in their use on almost every platform. Companies like Netflix, Amazon make use of recommender systems to suggest the best product to their users.

Why a gadget recommendation system?
Today smartphones have become a basic need of individuals, as a communication device across the globe. The advances in smartphone technology and the competitive fight among the smartphone manufacturers created the situation that almost every day a new model of a smartphone is being introduced into the market. The endless increase in the options space presented a tricky challenge in front of the buyers of smartphones. The major factors that influence users in selecting a smartphone to use, which include innovative features, image, price, personal recommendation, durability, portable aspects, influence of media, post-sales service and so on. Though smartphones have a number of features in common, manufacturers still try to carry uniqueness to their products by adding some more new features to the existing features. This made smartphone development a challenge and manufacturers welcomed the challenge with a great set of innovative designs. The growing number of brands and models created fierce market competition. Therefore it is necessary to run with innovations, updates and at the same time it is most desirable to know the trending thoughts of potential customers. So here the recommendation system suggests the best possible outcome for the customers based on the recommendation algorithm.

Past solutions and their drawbacks
Content-Based system -
These systems make recommendations using a user's item and profile features. They hypothesize that if a user was interested in an item in the past, they will once again be interested in it in the future. Similar items are usually grouped based on their features. User profiles are constructed using historical interactions or by explicitly asking users about their interests. There are other systems, not considered purely content-based, which utilize user personal and social data. One issue that arises is making obvious recommendations because of excessive specialization that means user A is only interested in categories B, C, and D, and the system is not able to recommend items outside those categories, even though they could be interesting to them. Another common problem is that new users lack a defined profile unless they are explicitly asked for information. In spite of that it is relatively simple to add new items to the system. We just

need to ensure that we assign them a group according to their features.

Collaborative filtering system - Collaborative filtering is currently one of the most frequently used approaches and usually provides better results than content-based recommendations. Some examples of this are found in the recommendation systems of YouTube, Netflix, and Spotify. Collaborative filtering is a technique that can filter out items that a user might like on the basis of reactions by similar users. It works by searching a large group of people and finding a smaller set of users with tastes similar to a particular user. It looks at the items they like and combines them to create a ranked list of suggestions.

There are two main challenges that come up with these systems:
Cold start: we should have enough information that is user-item interactions for the system to work. If we set up a new e-commerce site, we cannot give recommendations until users have interacted with a significant number of items.

Adding new users/items to the system: whether it is a new user or item, we have no prior information about them since they don't have existing interactions.

Our solution - Hence to overcome above issues of content-based and collaborative filtering a knowledge based recommendation system is used which is based on functional requirements of the product, to produce the more accurate recommendation. In this the users can give their requirements explicitly which they cannot do in content based and collaborating filtering techniques.

1.1 Problem definition

Most people (both with technical and non-technical backgrounds) are in a dilemma about which gadget will be a perfect fit for them as the number of new launches is increasing every day.

1.2 Aim and objective of the project

1.2.1 Aim

- **Data collection -** Using data scraper collecting the specifications of various devices from different e-commerce sites. The data will be in either numeric or string format. Data will be processed in further procedure.
- **Input user requirements** – Collecting the requirements from user by giving web form. Web form will contains check boxes, drop down lists, value selector. Collected requirements will give to algorithm for further procedure
- The output will be product information like product name, its pros and cons, specifications, product image, best buy affiliate link.

1.2.2 Objectives

1. To get a dataset containing the information of the gadgets.

2. To Design and develop a platform that suggests laptops, smartphones, earphones, and other gadgets that meet most of the user's technical requirements.

3. To ensure maximum customer satisfaction by suggesting the product in less time

1.3 Scope and limitation of the project

1.3.1 Scope

As an initial version, the project will be able to generate Smartphone Recommendations.
The problem statement is quite straightforward: recommend gadgets to users from a large availability of gadgets. There are many online recommender systems each following have its own methodology in recommending gadgets. The GSMarena, for example, follows a methodology where it takes into consideration the user activity, the similarity, scores based on specification. Here we are taking a very basic approach of calculating similarities of requirements based on users' requirements.

1.3.2 Limitations

1. As an initial version, the project is starting with only Smartphone Recommendations.

2. Gadget's design, battery life, and camera ratings may vary from person to person because they are relative. This project focuses on the bigger picture where all the gadgets are compared with each and every gadget from that category. The views presented about design, battery life, and camera are with respect to the standard user.

3. Currently the project is only working with the English language

1.4 Timeline of the project

Task Name	Duration	Start Date	End Date
Domain Selection	7 days	01-07-2021	07-07-2021
Domain Finalization	7 days	08-07-2021	14-07-2021
Selection of Problem Statement	14 days	15-07-2021	28-07-2021
Finalization of Problem Statement	7 days	29-07-2021	04-08-2021
Study on Research Paper	14 days	05-08-2021	18-08-2021
Documentation of Synopsis	14 days	19-08-2021	01-09-2021
Requirement Analysis	7 days	02-09-2021	08-09-2021
System Requirement	7 days	09-09-2021	15-09-2021
Module Identification	7 days	16-09-2021	21-09-2021
System Architecture	7 days	23-09-2021	30-09-2021
Implementation 25%	14 days	01-12-2021	14-12-2021
Testing 25%	7 days	15-12-2021	21-12-2021
Implementation 50%	14 days	22-12-2021	04-01-2022
Testing 50%	7 days	05-01-2022	11-01-2022
Implementation 75%	14 days	12-01-2022	25-01-2022
Testing 75%	7 days	26-01-2022	01-02-2022
Implementation 100%	14 days	02-02-2022	15-02-2022
Testing 100%	7 days	16-02-2022	22-02-2022
Report Making	14 days	23-02-2022	09-03-2022

Table 1: Timeline of the project

2. Background study and literature overview

2 Background study and literature overview

2.1 Literature overview

Shakila Shaikh [1] proposed techniques such as content-based, collaborative, and hybrid methods. In Collaborative Filtering the algorithm first finds the customers who are similar to the user and then gives recommendations to the user. Technique used in collaborative filtering is that each item is ranked first according to how many similar customers purchase that particular item. In Content-based recommender systems the recommendations are provided by considering the properties of the product or services which are already taken by the user. The properties of a product or services which are previously taken by the user are analyzed and matched with other properties of the products present in the database. Then the products in the database which have similar properties are displayed as the recommended product.
Pradeep Kumar Singh [2] proposed techniques such as Content based, collaborative, knowledge based, hybrid, demographic, etc. The recommendation systems discussed in these papers are used by many e-commerce sites but the user priorities are not much taken into the consideration as these methods make recommendations based on user similarity, user history, product properties, searches made by users, etc.
Greg Linden, Brent Smith, and Jeremy York [3] proposed cluster model. Cluster models divide the customer base into many segments and treat the task as a classification problem to find customers who are similar to the user. The algorithm assigns the user to the particular segment containing the most similar customers. It then uses the purchases and ratings of the customers in that assigned segment to generate recommendations. The item-to-item collaborative algorithm's online component looks up similar items for the user's purchases and ratings scales which is not dependent on the catalog size or the total number of customers; it is dependent only on how many items the user has purchased or rated. Thus, the algorithm works fast even for extremely large data sets. Because the algorithm recommends the items which are highly correlated to the purchased or rated item so the quality of the recommendation is excellent. The item-to-item collaborative algorithm also performs well with limited user data, producing high-quality recommendations based on as few as two or three items. The proposed system is able to successfully integrate recommendations.
Ken Arnett DS, Z.K.A. Baizal, Adiwijaya [4] proposed knowledge based recommender system for gadget recommendation which is based on functional requirements of the product, to produce the more proper recommendation. Proposed approach uses mapping between functional requirements, components product and supporting features of components. Here the Euclidean fuzzy concept is utilized for calculating the similarity between user requirements and product features. By using Euclidean fuzzy concept the system only recommends a product which is exactly fit with the user's requirements. The recommendation techniques such as collaborative filtering, content based filtering, hybrid filtering are based on history, especially rating history. Therefore, the user cannot describe their needs explicitly. In a knowledge-based recommender system, the system gets users' requirements on a product then uses the knowledge base to decide the most suitable products. The proposed approach based on Euclidean Fuzzy, produces a good accuracy performance, with precision value 92.6

2.2 Critical appraisal of other people's work

Collaborative filtering works on the similarity between the items or the users. This algorithm recommends the items that are liked by the similar type of user. The item-based approach gives the suggestions using the properties of the similar products. The content-based filtering is suitable for e-commerce sites. This method uses the different attributes of the items and generates the recommendations based on these attributes. The products with closer properties are recommended. The query suggestion technique helps users to shrink the scope for searching items on the internet.
In [3] discussed about Amazon.com and its recommendation algorithms. It states how the new item to item based collaborative filtering approach is better than the old collaborative filtering.
In [2] a variety of recommendation algorithms like content based , collaborative, knowledge based, hybrid, demographic, etc. Problems of recommendation systems like scalability, limited content analysis, synonymy, abbreviations, over-specialization are discussed in this paper. The recommendation systems discussed in these papers are used by many e-commerce sites but the user priorities are not much taken into the consideration as these methods make recommendations based on user similarity, user history, product properties, searches made by users, etc.

2.3 Investigation of current project and related work

There are many approaches to implement a recommendation system. We can use algorithms like collaborative filtering, cluster-based filtering, content-based filtering and many more of them. There are many ecommerce sites using these algorithms. Various methods have been tested to implement the new recommendation engine. System which uses the Euclidean Distance algorithm to find the similarity between the items and also the collaborative filtering method which uses the cosine similarity formula to get the similarity between the products are tested. After trying some of those methods it is found that the user priorities are not being considered and thus it is decided to implement a system which uses the user requirements and priorities to generate the recommendations.
The system is built on a web app platform where users will get recommendations. Users can log in and explore the variety of gadgets. There is a form with some questions about the requirements of the user for the specific gadget. Users can answer the questions in the form of numeric values using sliders. The questions asked in the form are simple and easy to understand so that the users without any technical background can give the precise answers. The data of the gadgets is collected from the ecommerce sites using the web scraping methodology and stored in the database. Then according to the user requirements the system checks for the products in the database using the algorithm and gets the list of the products which are best fit for the user. The recommended products are shown with their image and corresponding features.

3. Requirement Gathering

3 Requirement analysis

3.1 Requirement Gathering

1. As an admin, I want to add new gadgets, so that I can use those newly added gadgets for recommendation and customers also get the best gadget.
2. As a customer, I want a recommendation about a gadget and also the specifications of that gadget, so that I can see the configurations of the gadget.
3. As a customer, I want the pros and cons of the recommended gadget, so that I can find whether the gadget is suitable for me or not.
4. As a customer, I want an alternative recommendation along with the previous recommendation, so that I can get one more gadget to choose.
5. As a customer, I want a comparison chart between the recommended gadget and alternative recommended gadget, so that it is easy for me to choose the gadget between these two.
6. As a customer, I want the make-in-India alternative recommendation, so that I can purchase the gadget which is made in India rather than purchasing another country's gadget
7. As a customer, I want a link or site name where the recommended gadget is available, so that it is easy for me to go there and purchase the gadget

3.1.1 User Stories

1. An admin can add new gadgets in the database.
   Note: Avoid duplicate gadget entry.
2. A customer can see the recommended gadget and also the specifications of that gadget
3. A customer can check the pros and cons of recommended gadget.
4. A customer can get an alternative recommendation of a gadget.
5. A customer can easily compare the recommended gadget and alternative recommended gadget by watching the comparison chart.
6. A customer can get the make-in-India alternative recommendation.
7. A customer can get a link from where they can purchase the gadget.

3.2 Requirement Specification

Sr No.	Requirements	Essential or Desirable	Description of requirement	Remark
RS1	The System should have a login and signup page	Essential	Home page should have a login/signup option for the authentication of user	In progress
RS2	The system should have a help page	Desirable	After clicking on help option system should redirect to help page	In progress
RS3	The system should have facility of forget password	Desirable	After clicking on forgot password the system should give facility to set new password to the user.	In progress
RS4	The system should have form which collects the user requirements about gadget	Essential	After the successful login, form should ask certain questions about specifications of gadget to the user.	Completed
RS5	The system should recommend the product with their specifications	Essential	After submitting the form the system should display the recommended gadget along with their specifications.	Completed
RS6	The system should also provide the alternative recommendation	Essential	After submitting the form along with the recommended gadget the system should also provide an alternative recommendation of the gadget.	Completed
RS7	The system should have facility to see the comparison between recommended gadget and alternative recommended gadget	Essential	After clicking on the 'show comparison' button the system will show the comparison between the recommended gadget and an alternative recommended gadget.	Completed
RS8	The system should provide a make-in-India alternative recommendation	Essential	After submitting the form along with the recommended gadget the system should also provide make-in-India alternative recommendation of the gadget.	In progress
RS9	The system should have a facility to provide a link from where the user can purchase the gadget	Essential	After clicking on the 'show links' button, the system should display the best links or site names from where the user can purchase the recommended gadget.	Completed

Table 2: Requirement Specifications

3.3 Use case Diagram

Figure 1 shows the Use Case diagram where the two actors are user and admin. It shows the graphical representation of their possible interactions with the system.

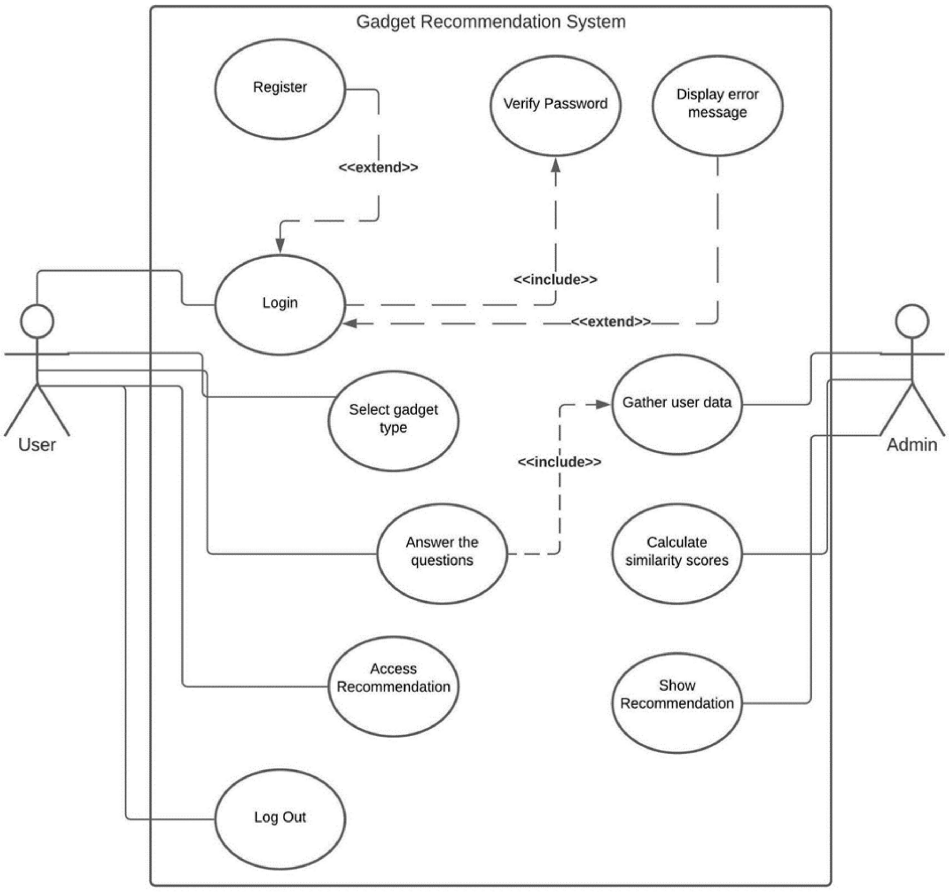

Figure 1: Use Case Diagram

4. System Design

4 System design

4.1 Architectural Design

Figure 2 shows the Architectural design where the system is decomposed into interacting components. It shows relationship between the components and how the components communicate with each other to share data.

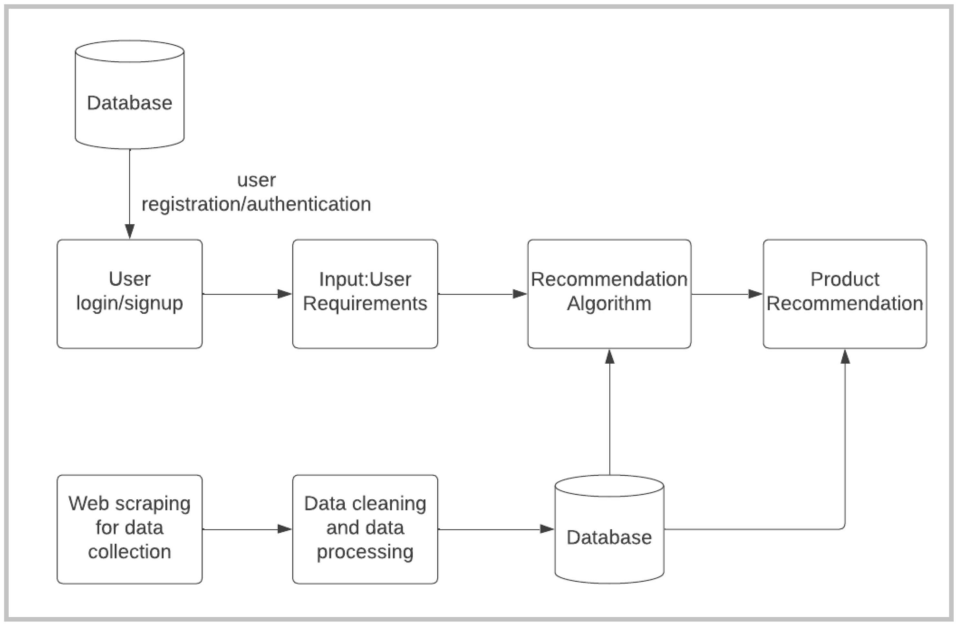

Figure 2: Architectural Design

4.2 User Interface Design

Figure 3 shows the User Interface Design. It shows how user interact with the system.

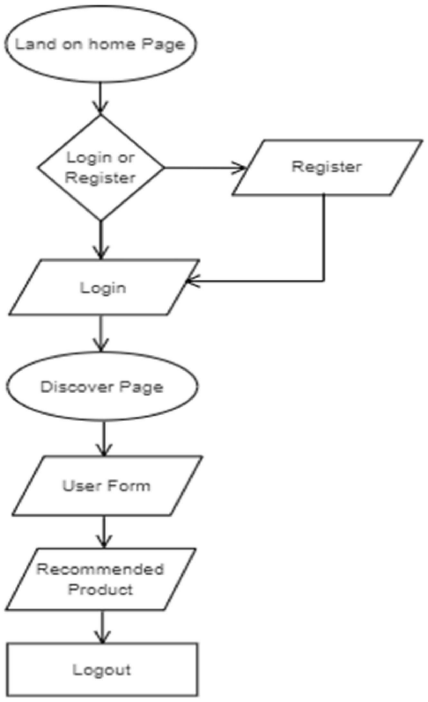

Figure 3: User Interface Design

4.3 Algorithmic description of each module

- User Login
  1. Input email ID or username and password
  2. Check if the input data is following the database constraints.
  3. If yes, then redirect to the home page
  4. If no,print error and return to step '1'
- User Registration
  1. Input email ID or username, password, and other user's information.
  2. Check if the input data is following the database constraints.
  3. If yes, then store that data in the database and redirect to the user login page.
  4. If no, then ID, password

5. Validate entered email ID and password with the data from the database.
6. If the input matches, then redirect to the home page.
7. If the data authentication is failed, then print the error and return to step '1'.

- Input Forms

1. Input the data from the user.
2. Check if all the required fields from the form are filled and pass the constraints.
3. If yes, then send that data to the knowledge-based algorithm.
4. If no, then print an error and return to step '1'.

- Web Scraper

1. Input the website's URL
2. Fetch the HTML elements from the given webpage.
3. Identify the required elements using element ID.
4. Start an element pointer from the first element.
5. Check if the data from that HTML element passes the constraint.
6. If yes, then save that data to a CSV file.
7. If not, then print the error and move to the next element.
8. Increment the element pointer.
9. If the element pointer reaches the end, then return the CSV file to the admin and exit the web scraper.

- Gadgets specification's dataset

1. Open the Product Specifications CSV file generated using a web scraper.
2. Perform data cleaning and data processing on each column.
3. Save the changes to the CSV file and close the file.

- Recommendation algorithm for gadget recommendation

1. Get user input from the input form.
2. Select the products from the CSV file in the given price range.
3. Compare user input with all the product specifications column-wise from step 'b' and store the difference in columns named such as a1,a2,a3 and so on.
4. Multiply a1, a2, a3,... column values with their respective priorities (weights) and store it again to a1, a2, a3,...columns.
5. Take row-wise summation of a1,a2,a3, a2, a3,... values and store it in column B.
6. Sort the CSV file in descending order with respect to values from column B.

7. Share the first 2 rows addressed to the recommendation view component.

- Product recommendation view
  1. Get the JSON objects of the top two recommended gadgets from the recommender algorithm.
  2. Using that these objects fetch details from Gadgets specification's dataset.
  3. Print those details using the predefined layout.

4.3.1 Dataflow Diagram

Figure 4 shows level 0 DFD. It is a basic overview of the whole system.

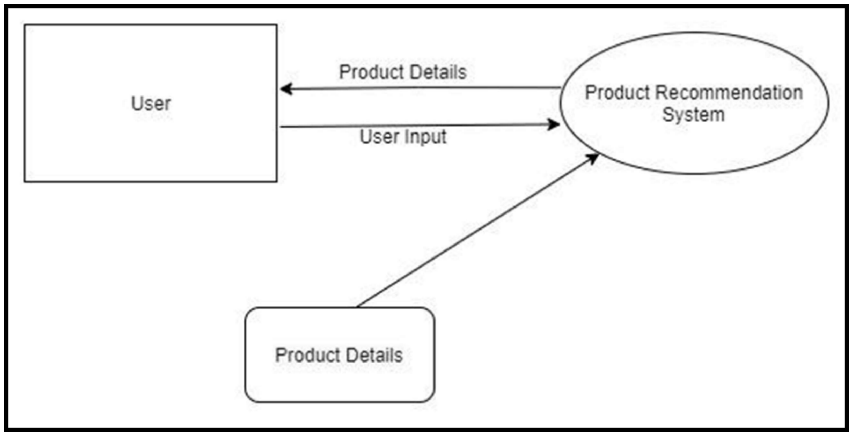

Figure 4: Data flow Diagram level 0

Figure 5 shows DFD level 1 and it mentions sub processes that together form the complete system.

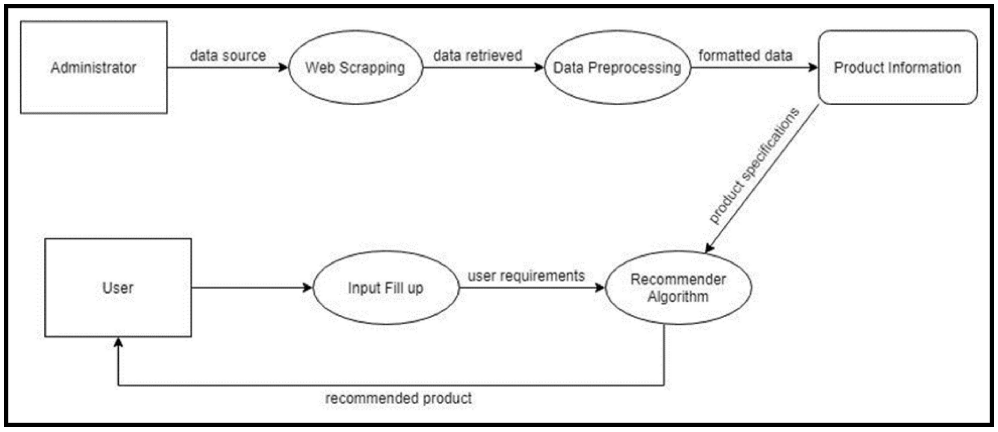

Figure 5: Data flow Diagram level 1

4.3.2 Sequence Diagram

Figure 6 illustrates the sequence of messages between objects in an interaction. It consists of a group of objects that are represented by lifelines, and the messages that they exchange over time during the interaction.

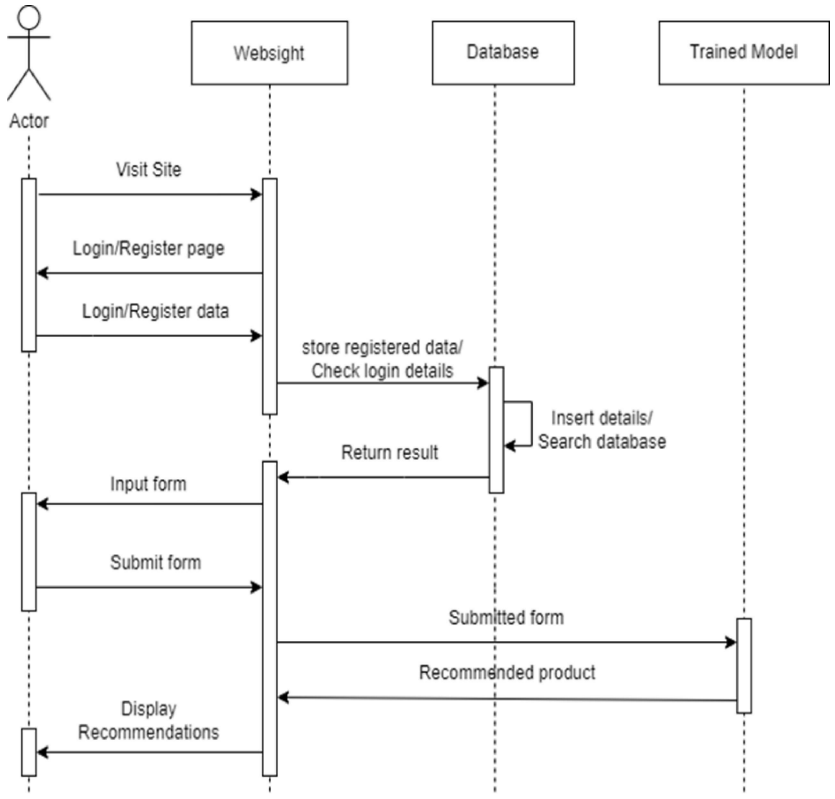

Figure 6: Sequence Diagram

4.3.3 Activity Diagram

Figure 7 shows activity diagram for the gadget recommendation system which depicts graphical representation of step wise activities and actions in the system.

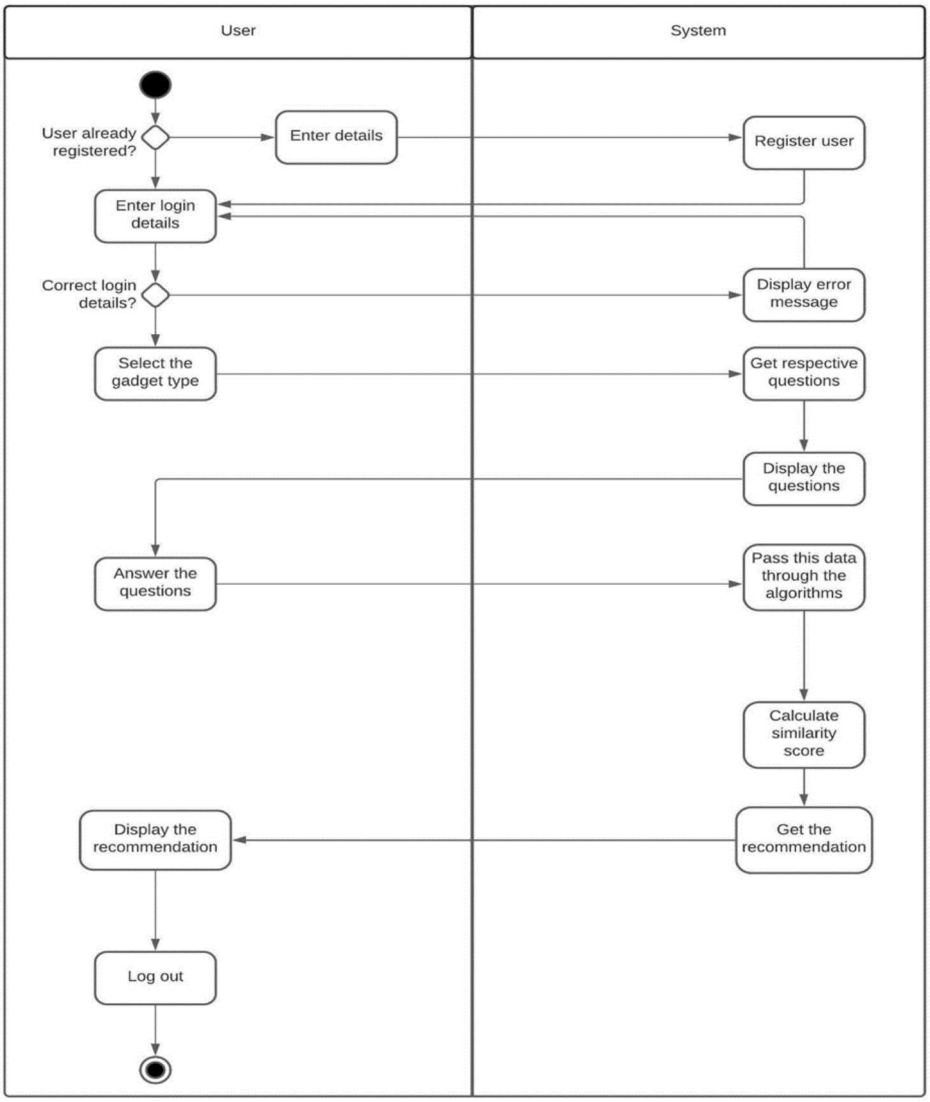

Figure 7: Activity Diagram

4.3.4 Component Diagram

Figure 8 describes the organization and wiring of the physical components in a system

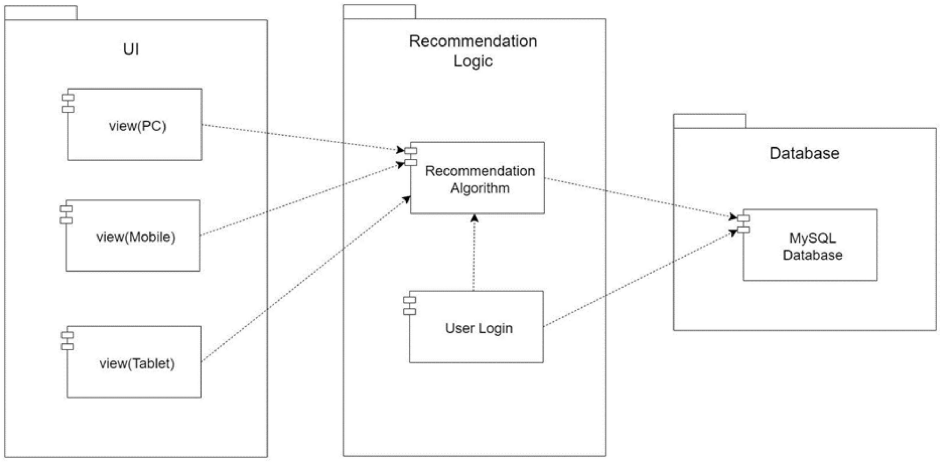

Figure 8: Component Diagram

4.3.5 Deployment Diagram

Figure 9 shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

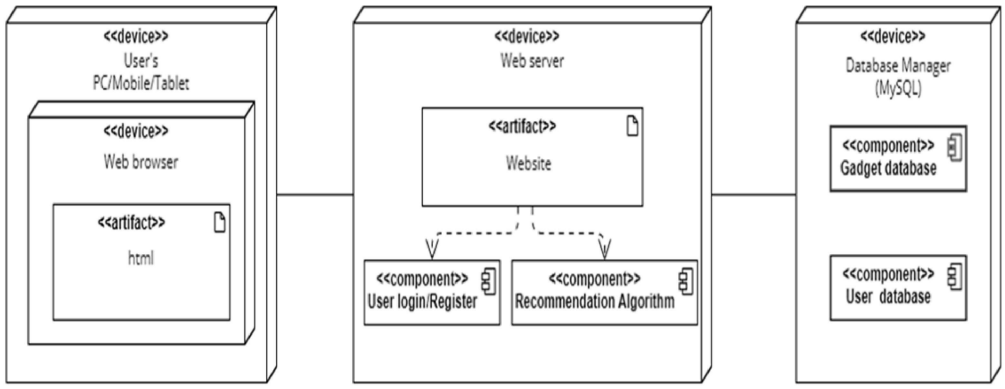

Figure 9: Deployment Diagram

5. Project Management and Cost

5 Project Management and cost

5.1 Project Scheduling Plan

Task Name	Duration	Start Date	End Date	Assigned To	Priority
Domain selection	7 days	01-07-2021	07-07-2021	All	High
Domain finalization	7 days	08-07-2021	14-07-2021	All	Medium
Selection of problem statement	14 days	15-07-2021	28-07-2021	Mrunal,Omkar	High
Finalization of problem statement	7 days	29-07-2021	04-08-2021	Vrushabh, Onkar	High
Study on Research paper	14 days	05-08-2021	18-08-2021	Samruddhi	High
Documentation of synopsis	14 days	19-08-2021	01-09-2021	Vrushabh, Onkar, Samruddhi	Medium
Requirement analysis	7 days	02-09-2021	08-09-2021	Omkar,Aayushi	Medium
System requirements	7 days	09-09-2021	15-09-2021	Mrunal,Aayushi	High
Module identification	7 days	16-09-2021	21-09-2021	Mrunal,Omkar,Aayushi	High
System architecture	7 days	23-09-2021	30-09-2021	Mrunal, Omkar	High
Implementation 25%	14 days	01-12-2021	14-12-2021	Omkar, Aayushi	High
Testing 25%	7 days	15-12-2021	21-12-2021	Mrunal, Aayushi	High
Implementation 50%	14 days	22-12-2021	04-01-2022	Vrushabh, Onkar, Samruddhi	High
Testing 50%	7 days	05-01-2022	11-01-2022	Vrushabh, Onkar, Samruddhi	High
Implementation 75%	14 days	12-01-2022	25-01-2022	Mrunal,Omkar,Aayushi	High
Testing 75%	7 days	26-01-2022	01-02-2022	Mrunal,Omkar,Aayushi	High
Implementation 100%	14 days	02-02-2022	05-02-2022	Mrunal,Omkar,Aayushi	High
Testing 100%	7 days	16-02-2022	22-02-2022	Mrunal,Omkar,Aayushi	High
Report Making	14 days	23-02-2022	09-03-2022	Vrushabh, Onkar, Samruddhi	High

Table 3: Project Schedule

5.2 Project Cost

The following is the pricing as per the Google Firebase Blaze plan:

- Hosting

Storage	INR 1.95/GB
Data Transfer	INR 11.25/GB

Table 4: Hosting charges

At the initial stage hosting will use less than 5GB storage.

- Real Time Databases

GB Stored	INR 375/GB
GB Downloaded	INR 75/GB

Table 5: Database charges

At the initial stage databases will use less than 2GB storage

- Domain charges : INR 800/year

At the initial stage the total costing will be around **INR 2000/year**

COCOMO RESULTS for Gadget Recommendation System								
MODE	**"A" variable**	**"B" variable**	**"C" variable**	**"D" variable**	**KLOC**	**EFFORT, (in person-months)**	**DURATION, (in months)**	**STAFFING, (recommended)**
semi-detached	3	1.12	2.5	0.35	1.900	6.156	4.723	1.304

Explanation: The coefficients are set according to the project mode selected on the previous page, (as per Boehm). Note: the decimal separator is a period.

The final estimates are determined in the following manner:

effort = a*KLOC^b, in person-months, with KLOC = lines of code, (in thousands), and:

staffing = effort/duration

where a has been adjusted by the factors:

Product Attributes	
Required Reliability	1.00 (N)
Database Size	1.00 (N)
Product Complexity	1.00 (N)
Computer Attributes	
Execution Time Constraint	1.00 (N)
Main Storage Constraint	1.00 (N)
Platform Volatility	1.00 (N)
Computer Turnaround Time	1.00 (N)
Personnel Attributes	
Analyst Capability	1.00 (N)
Applications Experience	1.00 (N)
Programmer Capability	1.00 (N)
Platform Experience	1.00 (N)
Programming Language and Tool Experience	1.00 (N)
Project Attributes	
Modern Programming Practices	1.00 (N)
Use of Software Tools	1.00 (N)
Required Development Schedule	1.00 (N)
New (Values are probably wrong)	
Required reusability	1.00 (N)
Documentation match to life-cycle needs	1.00 (N)
Personnel continuity	1.00 (N)
Multisite development	1.00 (N)

For further reading, see Boehm, "Software

Figure 10: Cost estimate using COCOMO model

6. Implementation

6 Implementation

6.1 Environmental Setting for Running the Project

1. Installation of IDE(Visual Studio code)
   (a) Visual Studio code is downloaded from official website:- `https://code.visualstudio.com/`
2. Installation of node.js :-
   (a) Node.js is downloaded from official website:- `https://nodejs.org/en/download/`
   (b) Open the Command prompt run as a administrator.To check the version of node type the command **node -v**
3. Installation of typescript :-
   (a) Open the Command prompt in administrator mode and type the command **npm install -g typescript**
4. Installation of Angular CLI
   (a) Open the Command prompt in administrator mode and type the command :- **npm install -g @angular/cli**
   (b) Command to check the angular version - **ng v**

6.2 Detailed Description of Methods

1. User Login / Signup
   - Users will be asked to fill a form for both login and signup.
   - If the user is new then the system allows the user to create an account.
   - If the user is signing up the data will be stored in the database.
   - If the user is logging in the data will be authenticated using the database.
2. User form
   - After the user visits the platform, they will have to answer certain questions.
   - These questions will be according to the gadget they want to be recommended.
   - There requirements will be further sent to the recommendation algorithm
3. Web Scraping
   - Using a web scraper the specifications of various devices from GSMarena, Flipkart or Amazon will be collected.
   - It will be implemented using Beautiful Soup or Scrapy Python library.
   - The data collected will be further processed and cleaned

4. Data pre-processing and Cleaning
    - The data obtained from web scraping will be processed and cleaned.
    - After the data is in a useful format it will be stored in a csv file in the database
5. Recommendation Algorithm
    - The user requirements will work as input for the algorithm.
    - Using the similarity metrics along with the priorities of the specifications, it will find out the product that matches most of the user requirements from the database.
    - It will be sent to the product recommendation module
6. Product recommendation
    - The product specifications, image, pros and cons, best buy links, etc will be displayed

6.3 Implementation Details

The **takeInput** function consists of the logic that takes input from the user and recommends the product according to the requirements entered by the user. Input to this function is parameters taken from the user and the respective product category's dataset. The function returns the top two recommended products

function takeInput(data: array of JSON objects, input_obj: JSON object)
Output - Returns an array of objects of products

```
BEGIN
    param_list = ['Design','Display','Software','Performance','Battery','Camera']
    user_req = []
    FOR i = 0 to param_list.length-1
        user_req[i] = parseInt(input_obj[param_list[i]])
    END FOR
    priority_ids =['Design_p','Display_p','Software_p','Performance_p','Battery_p','Camera_p']
    priority_list = []
    FOR i = 0 to priority_ids.length-1
        priority_list[i] = parseInt(input_obj[priority_ids[i]])
    END FOR
    user_price = parseInt(input_obj.Price)
    affordable=[];
    FOR i=0 to data.length-1
        IF parseInt(data[i].price) ¡= user_price) THEN
            affordable.push(data[i]);
        END IF
    END FOR
    FOR i = 0 to data.length-1
        data[i]['Design_diff'] = (parseInt(data[i].design)-user_req[0])*priority_list[0]
        data[i]['Display_diff'] = (parseInt(data[i].display)user_req[1])*priority_list[1]
```

```
        data[i]['Software_diff']=(parseInt(data[i].software)-user_req[2])*priority_list[2]
        data[i]['Performance_diff']=(parseInt(data[i].performance)-user_req[3])*priority_list[3]
        data[i]['Battery_diff']=(parseInt(data[i].battery_life)-user_req[4])*priority_list[4]
        data[i]['Camera_diff']=(parseInt(data[i].camera)-user_req [5])*priority_list[5]
      END FOR
      FOR i = 0 to data.length-1
        data[i]["main_diff"] = data[i].Camera_diff+ data[i].Performance_diff+
        data[i].Display_diff+ data[i].Battery_diff+ data[i].Software_diff+data
        [i].Design_diff
      END FOR
     data.sort((a,b) =¿ b.main_diff-a.main_diff)
     obj=[]
     obj.push(data[0])
     obj.push(data[1])
     return obj
END
```

7. Integration and Testing

7 Integration and Testing

7.1 Description of the Integration Modules

The login module avails the user with forms for the login and signup. If the user has already created an account then login form is displayed to the user. The data(Email ID, Password) taken through the signup form is checked if it follows the database constraints. After validation it is stored in the database. If the user is new then the system allows the user to create an account. The login details(Email ID, Password) will be authenticated using the database. If the data authentication fails then an error message is shown on the login page. The user chooses the type of the gadget on the discover page. Then the user has to answer the certain questions asked in the form according to the gadget needed to be recommended. After checking that the required fields are filled and passing the constraints, the inputs are provided to the recommendation algorithm for further processing.

Before the recommendation algorithm runs the dataset about the gadgets is created using the web scraping module. A website URL is taken as the input then the HTML elements are fetched. Required elements are identified using the element ID. Data of the elements that pass the constraints is stored in the CSV file. Once the CSV file is created using the web scraper, it needs to be processed. Certain operations like data cleaning and data processing are performed on the data and the changes made are saved to the CSV file.

The algorithm then takes the user inputs. The products are selected from the CSV file which are in the given price range. Then these products are compared with all the product specifications provided by the user. The difference is taken column wise and it is stored in corresponding rows. Then the difference is multiplied with the priorities (weights) and these values are stored. Then row wise summation is taken and the values are stored in corresponding rows. These are the final values that are used for the recommendations.

The CSV file is then sorted using the values calculated by the algorithm in descending order and the first two values from the list are shared with the view component.

To view the products , the pointer from the algorithm is taken and using that pointer the gadget specifications are fetched from the dataset. Then the product details are displayed on the predefined layout.

7.2 Testing

Sr No.	Test Case Title	Description	Expected Outcome
1	Successful User Registration The Registration to the system is successful because all the Mandatory fields and proper data is filled by the user. Registration should be successful	The Registration to the system is successful because all the Mandatory fields and proper data is filled by the user.	Registration should be successful
2	Unsuccessful user Registration due to Mandatory fields not filled.	Registration to the system with unfilled mandatory fields.	Registration should fail with an error 'missing required field'.
3	Unsuccessful user Registration due to Invalid data	Registration to the system with Invalid data. For ex., • Email address is not valid • Mobile number is not 10 digit	Registration should fail with an error 'Invalid Data'
4	Successful User Verification	Successful User Verification.The login to the system should be tried with the correct admin and password. Login should be successful and the user successfully enter in to the system	Login should be successful and the user successfully enter in to the system
5	Unsuccessful User Verification due to wrong password	Wrong password is typed while login.	Unsuccessful User Verification due to wrong password Wrong password is typed while login.Login should fail with an error message 'Invalid Password'
6	Successful gadget Recommendation.	After entering into the system the user gives all the answers properly.	Recommendation should be successful and users can access the recommendation.
7	Unsuccessful Recommendation due to wrong device name or other data	After entering into the system if the user gives the wrong device name(device names that are not available) or gives wrong data.	Recommendation should fail with an error 'Wrong device name'

Table 6: Test Cases and their description

8. Performance Analysis

8 Performance Analysis

- **User interface**
  To interact with the system, the interface provided is much simpler, minimal and responsive. As the user interface is web based it is platform independent and anyone with internet connection can access it from anywhere. It is created using Angular.js with a single page design so that users can have an experience like they are using an application on a device. The questions in the form are easy to understand for a typical user without any technical background. This makes the system user friendly and provides better usability.

- **User Time**
  Once the user visits the website it will take hardly 1 to 2 minutes to fill the login details. After login, users can explore the discover page and choose the gadget from the categories section in about 3 or 5 minutes. To fill the form, about 6 to 7 minutes of average time is needed. Then the recommendations are shown and the user can take as much time as he/she requires to select a product. Hence it will take about 15 to 20 minutes to decide the desired product. The system makes the decision making process fast and convenient, saving the user's valuable time.

- **Correctness**
  The gadgets that are recommended by the system are based on the user requirements and priorities in order to fulfill all the requirements of the user. The algorithm selects only those products which have a higher similarity score that aligns with the user's expectations. As inputs are taken from the actual user, the results generated are the best fit in the user's perspective which implies the correct execution of the system.

- **Response Time**
  As Angular.js is used to create the web app, major events like authentication and routing are executed on the client side. This results in reducing the response time as there is no need to send requests to the server for loading the pages

9. Future Scope

9 Future Scope

Nowadays digital devices are part of the day-to-day life of every person. Digital devices have come to be essential in basic needs of individuals. As the technology developed rapidly, it is essential to meet different functional needs of end-users. Therefore it is necessary to suggest gadgets to the customers by knowing their interest. Digital devices and communication networks are increasing to cover all the aspects of customer's activity. These data can also be trained and modeled for future use to cope with the upcoming technological innovations.

10. Applications

10 Applications

1. **Primary consumer**
   This system will give the best suitable suggestions to the primary consumers by just taking their requirements. So while buying any electronic smart gadget they can use this system for taking suggestions. By using this system the primary consumer can meet their technical requirements and also get satisfied.

2. **Gadget sellers**
   Both online sellers and offline sellers can use this system to give the best suggestions about gadgets to their customers. By giving best suggestions the gadget seller helps customers to fulfill their needs. Due to this the customer also gets satisfied and gadget sellers can easily sell their gadgets. So by using this system the gadget sellers can make more and more customers and can sell more gadgets. So it will increase their profit

3. **Gadget Companies**
   By using this system gadget companies will know about which gadgets the customer likes most and what are the requirements of the customer. This system will help companies to study the product market. After studying the product market, gadget companies can easily make more and more decisions about gadgets.

11. Installation Guide and User Manual

11 Installation Guide and User Manual

1. Link of the website is given to the user.
2. Users need a good internet connection and any web browser like Google chrome, Microsoft edge, etc
3. Users should have to click on the given link or just copy paste that link on to the available web browser.
4. After clicking on the link the home page will open.
5. Users should have to click on the registration button if he/she is using the site for the first time, otherwise choose the login option. Fill the correct details and also fill all the mandatory fields. Then click on register button.
6. If User is already registered, choose the login option and fill in the correct username and password which is given at the time of registration and then click on the login button.
7. After landing on home page, the system will show one form which is based on user requirements. User have to fill that form and submit the form.
8. After successfully submitting the form the user will get the recommended gadget along with specifications of that gadget. User will also get an alternative recommended gadget and make-in-India alternative recommended gadget.
9. If a user wants to see the comparison between recommended gadget and alternative recommended gadget, User has to click on 'show comparison' button.
10. If a user wants to see the best links from where they can purchase the recommended gadget, users have to click on the 'show links' button.
11. After completion of use, users have to logout by clicking on the logout button.

12. Plagiarism Report

12 Plagiarism Report

INDEX Contents 1

Introduction 1.1 Problem definition . . . . . **1.2** Aim **and** objective **of** 11 **the** project . . . . . **1.2.1 Aim** . . . . . **1.2.2 Objectives** **1.3**

Scope and limitation of the project . . . . . 1.3.1 Scope . . . . . 1.3.2 Limitations . . . . . 1.4 Timeline of the project . . . . . 2 Background study and literature overview 2.1 Literature overview . . . . . 2.2 Critical appraisal of other people's work . . . . . 2.3 Investigation of current project and related work . . . . . 3 Requirement analysis 3.1 Requirement Gathering . . . . . 3.1.1 User Stories . . . . . 3.2 Requirement Specification . . . . . 3.3 Use case Diagram . . . . . 4 System design 4.1 Architectural Design . . . . . 4.2 User Interface Design . . . . . 4.3 Algorithmic description of each module . . . . . 4.3.1 Dataflow Diagram . . . . . .

4.3.2 Sequence **Diagram** . . . . . **4.3.3 Activity Diagram** . . . . . 12 **. 4.3.4** Component **Diagram** . . . . . **4.3.5** Deployment **Diagram**

. . . . . 5 Project Management and cost 5.1 Project Scheduling Plan . . . . . 5.2 Project Cost . . . . . 6 Implementation 6.1 Environmental Setting for Running the Project . . . . . 6.2 Detailed Description of Methods . . . . . 6.3 Implementation Details . . . . . 3 4 4 4 4 5 5 5 6 7 7 8 8 9 9 9 10 11 12 12 13 13 16 17 18 19 19 20 20 20 22 22 22 23 1 7 Integration and Testing 7.1 Description of the Integration Modules . . . . . 7.2 Testing . . . . . 8 Performance Analysis 9 Future Scope 10 Applications 11 Installation Guide and User Manual 12 Bibliography 25 25 26 27 28 29 30 31 2 1 Intro duction Recommendation systems are software agents that bring out the interests and preferences of individual users and make recommendations accordingly. Also it aims to predict user's interests and recommend product items that quite likely are

recommendations: item- to-item collaborative filtering," IEEE Internet Computing, vol. 7, no. 1, pp. 76–80, 2003. [4] D. S. Ken Arnett, Z. K. A. Baizal, and Adiwijaya, "Recommender system based on user functional requirements using euclidean fuzzy," in 2015 3rd International Conference on Information and Communication Technology (ICoICT), 2015, pp. 455–460. 31

sources:

1 278 words / 4% - Internet from 27-Oct-2021 12:00AM
techvaluetrends.com

2 155 words / 2% - Internet from 19-Oct-2019 12:00AM
paper.ijcsns.org

3 98 words / 1% - Internet from 05-Sep-2017 12:00AM
ijarcsse.com

4 63 words / 1% - Internet from 01-Feb-2021 12:00AM
blog.knoldus.com

5 55 words / 1% - Internet from 23-Jun-2017 12:00AM
adiwijaya.staff.telkomuniversity.ac.id

6 45 words / 1% - Crossref
Pushpalatha A, Harish Sanmugam J, Jeya Pradeepa, Madhu Bala S. "A Gadget Recommendation System using Data Science", Journal of Information Technology and Digital World, 2020

7 25 words / < 1% match - Internet from 12-Apr-2022 12:00AM
www.coursehero.com

8 22 words / < 1% match - Internet from 30-Nov-2020 12:00AM
www.slideshare.net

9 20 words / < 1% match - Crossref
Rabi Narayan Behera, Sujata Dash. "A Particle Swarm Optimization based Hybrid Recommendation System", International Journal of Knowledge Discovery in Bioinformatics, 2016

10 18 words / < 1% match - Crossref
Nikos Tsianos, Panagiotis Germanakos, Zacharias Lekkas, Constantinos Mourlas. "chapter 1 Assessment of Human Factors in Adaptive Hypermedia Environments", IGI Global, 2009

11 18 words / < 1% match - Internet from 07-Feb-2022 12:00AM
www.grin.com

You can find detailed plagiarism report at the end of the document

13. Bibliography

13 Bibliography

References

[1] S. Shaikh, S. Rathi, and P. Janrao, "Recommendation system in e-commerce websites: A graph based approached," in *2017 IEEE 7th International Advance Computing Conference (IACC)*, 2017, pp. 931–934.

[2] P. Singh, P. Dutta Pramanik, A. Dey, and P. Choudhury, "Recommender systems: An overview, research trends, and future directions," *International Journal of Business and Systems Research*, vol. 15, p. 14–52, 01 2021.

[3] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[4] D. S. Ken Arnett, Z. K. A. Baizal, and Adiwijaya, "Recommender system based on user functional requirements using euclidean fuzzy," in *2015 3rd International Conference on Information and Communication Technology (ICoICT)*, 2015, pp. 455–460.

Gadget Recommendation System Report

By: Omkar Shinde

As of: May 2, 2022 1:05:18 PM
6,544 words - 20 matches - 12 sources

Similarity Index

12%

Mode: Similarity Report

paper text:

INDEX Contents 1

Introduction 1.1 Problem definition . . . . . **1.2** Aim **and** objective **of** 11 **the** project . . . . . **1.2.1 Aim . . . . . 1.2.2 Objectives 1.3**

Scope and limitation of the project . . . . . 1.3.1 Scope . . . . . 1.3.2 Limitations . . . . . 1.4 Timeline of the project . . . . . 2 Background study and literature overview 2.1 Literature overview . . . . . 2.2 Critical appraisal of other people's work . . . . . 2.3 Investigation of current project and related work . . . . . 3 Requirement analysis 3.1 Requirement Gathering . . . . . 3.1.1 User Stories . . . . . 3.2 Requirement Specification . . . . . 3.3 Use case Diagram . . . . . 4 System design 4.1 Architectural Design . . . . . 4.2 User Interface Design . . . . . 4.3 Algorithmic description of each module . . . . . 4.3.1 Dataflow Diagram . . . . . .

4.3.2 Sequence **Diagram . . . . . 4.3.3 Activity Diagram . . . . . .** 12 **4.3.4** Component **Diagram . . . . . 4.3.5** Deployment **Diagram**

. . . . . 5 Project Management and cost 5.1 Project Scheduling Plan . . . . . 5.2 Project Cost . . . . . 6 Implementation 6.1 Environmental Setting for Running the Project . . . . . 6.2 Detailed Description of Methods . . . . . 6.3 Implementation Details . . . . . 3 4 4 4 4 5 5 5 6 7 7 8 8 9 9 9 10 11 12 12 13 13 16 17 18 19 19 20 20 20 22 22 22 23 1 7 Integration and Testing 7.1 Description of the Integration Modules . . . . . 7.2 Testing . . . . . 8 Performance Analysis 9 Future Scope 10 Applications 11 Installation Guide and User Manual 12 Bibliography 25 25 26 27 28 29 30 31 2 1 Intro duction Recommendation systems are software agents that bring out the interests and preferences of individual users and make recommendations accordingly. Also it aims to predict user's interests and recommend product items that quite likely are

interesting for them. These are basically the systems that recommend things like videos, books, music, shopping items and even people.

They have the potential to support and improve the quality of the decisions users **make while searching for and selecting** things **online** 9

. In Database large volume of information is present, so recommender systems filterers the

most important information based on the data provided by a user and other factors that take care of the user's preference and interest . Recommendation **Systems** 7

have be- come excessively popular in recent times with their presence and increase in their use on almost every platform. Companies like Netflix, Amazon, etc. use recommender systems to give the best product to their users. Why a gadget recommendation system? Today smartphones have

become a basic need of individuals, **as a** communica- tion device **across the globe. The advances in** smartphone **technology and the competitive fight among the** smartphone **manufacturers created the situation that almost every day a new model of a** smartphone **is being introduced** into **the market. The endless increase in the options space presented a tricky** chal- lenge **in front of the** users **of** smartphones. **The major factors that influence** users **in selecting a** smartphone **to use** , which **include innovative features** , im- age, **price, personal recommendation, durability, portable aspects, influence of media, post-sales service and so on** 2

. Though smartphones

have a number of fea- tures **in common, manufacturers still try to** carry **uniqueness to their products by adding some more new features to the existing features. This made** smart- phone **development a challenge and manufacturers** welcomed **the challenge with a great set of innovative designs. The growing number of brands and models created fierce market competition. Therefore it is** necessary **to run with** inno- vations, **updates and at the same time it is** most **desirable to know the trending thoughts of potential customers** 2

. So here the recommendation system suggests the best possible outcome for the customers based on the recommendation al- gorithm. Past solutions and their drawbacks

Content-Based system - These systems make recommendations using a user's item and profile features. They hypothesize that if a user was interested in an item in the past, they will once again be interested in it in the future. Similar items are usually grouped based on their features. User profiles are constructed using historical interac- tions or by explicitly asking users about their interests. There are other systems, not considered purely content-based, which utilize user personal and social data. One issue that arises is making obvious recommendations because of excessive specialization that means user A is only interested in categories B, C, and D, and the system is not able to recommend items outside those categories, even though they could be interesting to them. Another common problem is that new users lack a defined profile unless they are explicitly asked for information 1

. In spite of that

it is relatively simple to add new items to the system. We just 3 need to ensure that we assign them a group according to their features. Collaborative filtering system - Collaborative filtering is currently one of the most frequently used approaches and usually provides better results than content-based recommendations. Some examples of this are found in the rec- ommendation systems of YouTube, Netflix, and Spotify 1

.

Collaborative filtering is a technique that can filter out items that a user might like on the basis of reactions by similar users. It works by searching a large group of people and finding a smaller set of users with tastes similar to a particular user. It looks at the items they like and combines them to create a ranked list of suggestions 4

.

There are two main challenges that come up with these systems: Cold start: we should have enough information that is user-item interactions for the system to work. If we set up a new e-commerce site, we cannot give recommendations until users have interacted with a significant number of items. Adding new 1

users/items to the system: whether it is a new user or item, we have no prior information about them since they don't have existing

inter- actions. Our solution - Hence to overcome above issues of content-based and collab- orative filtering a knowledge based recommendation system is used which is based on functional requirements of the product, to produce the more proper recommendation. In this the users can give their requirements explicitly which they cannot do in content based and collaborating filtering techniques. 1.1 Problem definition Most people (both with technical and non-technical backgrounds) are in a dilemma about which gadget will be a perfect fit for them as the number of new launches is increasing every day. 1.2 Aim and ob jective of the pro ject 1.2.1 Aim • Data collection - Using data scraper collecting the specifications of var- ious devices from different e-commerce sites. The data will be in either numeric or string format. Data will be processed in further procedure. • Input user requirements – Collecting the requirements from user by giving web form. Web form will contains check boxes, drop down lists, value selector. Collected requirements will give to algorithm for further procedure • The output will be product information like product name, its pros and cons, specifications, product image, best buy affiliate link. 1.2.2 Ob jectives 1. To get a dataset containing the information of the gadgets. 4 2. To Design and develop a platform that suggests laptops, smartphones, earphones, and other gadgets that meet most of the user's technical re- quirements. 3. To ensure maximum customer satisfaction by suggesting the product in less time 1.3 Scope and limitation of the project 1.3.1 Scope As an initial version, the project will be able to generate Smartphone Rec- ommendations. The problem statement is quite straightforward: recommend gadgets to users from a large availability of gadgets. There are many online recommender sys- tems each following have its own methodology in recommending gadgets. The GSMarena, for example, follows a methodology where it takes into considera- tion the user activity, the similarity, scores based on specification. Here we are taking a very basic approach of calculating similarities of requirements based on users' requirements. 1.3.2 Limitations 1. As an initial version, the project is starting with only Smartphone Rec- ommendations. 2. Gadget's design, battery life, and camera ratings may vary from person to person because they are relative. This project focuses on the bigger picture where all the gadgets are compared with each and every gadget from that category. The views presented about design, battery life, and camera are with respect to the standard user. 3. Currently the project is only working with the English language 5 1.4 Timeline of the pro ject Task Name Duration Start Date End Date Domain Selection 7 days 01-07-2021 07-07-2021 Domain Finalization 7 days 08-07-2021 14-07-2021 Selection of Problem Statement 14 days 15-07-2021 28-07-2021 Finalization of Problem Statement 7 days 29-07-2021 04-08-2021 Study on Research Paper 14 days 05-08-2021 18-08-2021 Documentation of Synopsis 14 days 19-08-2021 01-09-2021 Requirement Analysis 7 days 02-09-2021 08-09-2021 System Requirement 7 days 09-09-2021 15-09-2021 Module Identification 7 days 16-09-2021 21-09-2021 System Architecture 7 days 23-09-2021 30-09-2021 Implementation 25% 14 days 01-12-2021 14-12-2021 Testing 25% 7 days 15-12-2021 21-12-2021 Implementation 50% 14 days 22-12-2021 04-01-2022 Testing 50% 7 days 05-01-2022 11-01-2022 Implementation 75% 14 days 12-01-2022 25-01-2022 Testing 75% 7 days 26-01-2022 01-02-2022 Implementation 100% 14 days 02-02-2022 15-02-2022 Testing 100% 7 days 16-02-2022 22-02-2022 Report Making 14 days 23-02-2022 09-03-2022 Table 1: Timeline of the project 6 2 Background study and literature overview 2.1 Literature overview Shakila Shaikh [1]; Proposed techniques such as content-based, collabora- tive, and hybrid methods. In Collaborative Filtering the algorithm first finds the customers who are similar to the user and then gives recommendations to the user. Technique used in collaborative filtering is that

each item is ranked first according to how many similar customers purchase that particular item. In Content-based recommender systems the recommendations are provided by considering the properties of the product or services which are already taken by the user. The properties of a product or services which are previously taken by the user are analyzed and matched with other properties of the products present in the database. Then the products in the database which have similar properties are displayed as the recommended product. Pradeep Kumar Singh [2]; Proposed techniques such as Content based, col- laborative, knowledge based, hybrid, demographic, etc. The recommendation systems discussed in these papers are used by many e-commerce sites but the user priorities are not much taken into the consideration as these methods make recommendations based on user similarity, user history, product prop- erties, searches made by users, etc. Greg Linden, Brent Smith, and Jeremy York [3]; Proposed cluster model.

Clus- ter models divide the customer base into many segments and treat the task as a classification problem 10

to find customers who are similar to the user. The algorithm assigns the

user to the particular **segment containing the most** simi- lar **customers. It then uses the purchases and ratings of the customers in** that assigned **segment to generate recommendations. The** 3

item-to-item collaborative

algorithm's online component looks **up similar items for the user's purchases and ratings scales** 3

which is not dependent on

the catalog size or the total number of customers; it is dependent only on how many items **the user has purchased or rated. Thus, the algorithm** works **fast even for extremely large data sets. Because the algorithm recommends** the **items** 3

which are highly correlated to the purchased or rated item so the quality of the recommendation is excellent. The item-to-item collaborative

algorithm also performs well with limited user data, producing high-quality recommendations based on as few as two or three items . The proposed system **is** 3

able to successfully integrate recommendations. Ken Arnett DS, Z.K.A. Baizal, Adiwijaya [4] Proposed knowledge based recom- mender system for gadget recommendation which is

based on functional require- ments **of** the **product, to produce the more** proper **recommendation. Proposed approach uses mapping between functional requirements, components product and supporting features of** components. Here the **Euclidean fuzzy concept** is utilized **for calculating the similarity between user requirements and product features** 5

.

By using Euclidean fuzzy concept **the system only recommends a product which is exactly fit with the user's requirements** 5

. The recommendation techniques such as collaborative filtering, content based filtering, hybrid filter- ing are based on history, especially rating history. Therefore, the user cannot describe their needs explicitly. In a knowledge-based recommender system, the system gets users' requirements on a product then uses the knowledge base to decide the most suitable products. The proposed approach based on Euclidean Fuzzy, produces a good accuracy performance, with precision value 92.6 7 2.2 Critical appraisal of other people's work Collaborative filtering works on the similarity between the items or the users. This algorithm recommends the items that are liked by the similar type of user. The item-based approach gives the suggestions using the properties of the sim- ilar products. The content-based filtering is suitable for e-commerce sites. This method uses the different attributes of the items and generates the recommen- dations based on these attributes. The products with closer properties are recommended. The query suggestion technique helps users to shrink the scope for searching items on the internet. In [3] discussed about Amazon.com and its recommendation algorithms. It states how the new item to item based collaborative filtering approach is better than the old collaborative filtering. In [2] a variety of recommendation algorithms like content based , collabora- tive, knowledge based, hybrid, demographic, etc. Problems of recommenda- tion systems like scalability, limited content analysis, synonymy, abbreviations, over-specialization are discussed in this paper. The recommendation systems discussed in these papers are used by many e-commerce sites but the user prior- ities are not much taken into the consideration as these methods make recom- mendations based on user similarity, user history, product properties, searches made by users, etc. 2.3 Investigation of current pro ject and related work There are many approaches to implement a recommendation system. We can use algorithms like collaborative filtering,

cluster-based filtering, content-based filtering and many more of them. There are many ecommerce sites using these algorithms. Various methods have been tested to implement the new recommen- dation engine. System which uses the Euclidean Distance algorithm to find the similarity between the items and also the collaborative filtering method which uses the cosine similarity formula to get the similarity between the products are tested. After trying some of those methods it is found that the user priorities are not being considered and thus it is decided to implement a system which uses the user requirements and priorities to generate the recommendations. The system is built on a web app platform where users will get recommenda- tions. Users can log in and explore the variety of gadgets. There is a form with some questions about the requirements of the user for the specific gadget. Users can answer the questions in the form of numeric values using sliders. The questions asked in the form are simple and easy to understand so that the users without any technical background can give the precise answers. The data of the gadgets is collected from the ecommerce sites using the web scraping method- ology and stored in the database. Then according to the user requirements the system checks for the products in the database using the algorithm and gets the list of the products which are best fit for the user. The recommended products are shown with their image and corresponding features. 8 3 Requirement analysis 3.1 Requirement Gathering 1. As an admin, I want to add new gadgets, so that I can use those newly added gadgets for recommendation and customers also get the best gadget. 2. As a customer, I want a recommendation about a gadget and also the specifications of that gadget, so that I can see the configurations of the gadget. 3. As a customer, I want the pros and cons of the recommended gadget, so that I can find whether the gadget is suitable for me or not. 4. As a customer, I want an alternative recommendation along with the previous recommendation, so that I can get one more gadget to choose. 5. As a customer, I want a comparison chart between the recommended gadget and alternative recommended gadget, so that it is easy for me to choose the gadget between these two. 6. As a customer, I want the make-in-India alternative recommendation, so that I can purchase the gadget which is made in India rather than pur- chasing another country's gadget 7. As a customer, I want a link or site name where the recommended gadget is available, so that it is easy for me to go there and purchase the gadget 3.1.1 User Stories 1. An admin can add new gadgets in the database. Note: Avoid duplicate gadget entry. 2. A customer can see the recommended gadget and also the specifications of that gadget 3. A customer can check the pros and cons of recommended gadget. 4. A customer can get an alternative recommendation of a gadget. 5. A customer can easily compare the recommended gadget and alternative recommended gadget by watching the comparison chart. 6. A customer can get the make-in-India alternative recommendation. 7. A customer can get a link from where they can purchase the gadget. 9 3.2 Requirement Specification Sr No. Requirements Essential or Desirable Description of requirement Remark RS1 The System should have a login and signup page Essential Home page should have a lo- gin/signup option for the au- thentication of user In progress RS2 The system should have a help page Desirable After clicking on help option system should redirect to help page In progress RS3 The system should have facility of forget pass- word Desirable After clicking on forgot pass- word the system should give fa- cility to set new password to the user. In progress RS4 The system should have form which collects the user requirements about gadget Essential After the successful login, form should ask certain questions about specifications of gadget to the user. Completed RS5 The system should rec- ommend the product with their specifications Essential After submitting the form the system should display the rec- ommended gadget along with their specifications. Completed RS6 The system should also provide the alternative recommendation Essential After submitting the form along with the recommended gadget the system should also provide an alternative recommendation of the gadget.

Completed RS7 The system should have facility to see the com- parison between recom- mended gadget and al- ternative recommended gadget Essential After clicking on the 'show comparison' button the system will show the comparison be- tween the recommended gad- get and an alternative recom- mended gadget. Completed RS8 The system should pro- vide a make-in-India al- ternative recommenda- tion Essential After submitting the form along with the recommended gadget the system should also provide make-in-India alternative rec- ommendation of the gadget. In progress RS9 The system should have a facility to provide a link from where the user can purchase the gadget Essential After clicking on the 'show links' button, the system should display the best links or site names from where the user can purchase the recommended gad- get. Completed Table 2: Requirement Specifications 10 3.3 Use case Diagram Figure 1 shows the Use Case diagram where the two actors are user and admin. It shows the graphical representation of their possible interactions with the system. Figure 1: Use Case Diagram 11 4 System design 4.1 Architectural Design Figure 2 shows the Architectural design where the system is decomposed into interacting components. It shows relationship between the components and how the components communicate with each other to share data. Figure 2: Architectural Design 12 4.2 User Interface Design Figure 3 shows the User Interface Design. It shows how user interact with the system. Figure 3: User Interface Design 4.3 Algorithmic description of each module • User Login 1. Input email ID or username and password 2. Check if the input data is following the database constraints. 3. If yes, then redirect to the home page 4. If no,print error and return to step '1' • User Registration 1. Input email ID or username, password, and other user's information. 2. Check if the input data is following the database constraints. 3. If yes, then store that data in the database and redirect to the user login page. 4. If no, then ID, password 13 5. Validate entered email ID and password with the data from the database. 6. If the input matches, then redirect to the home page. 7. If the data authentication is failed, then print the error and return to step '1'. • Input Forms 1. Input the data from the user. 2. Check if all the required fields from the form are filled and pass the constraints. 3. If yes, then send that data to the knowledge-based algorithm. 4. If no, then print an error and return to step '1'. • Web Scraper 1. Input the website's URL 2. Fetch the HTML elements from the given webpage. 3. Identify the required elements using element ID. 4. Start an element pointer from the first element. 5. Check if the data from that HTML element passes the constraint. 6. If yes, then save that data to a CSV file. 7. If not, then print the error and move to the next element. 8. Increment the element pointer. 9. If the element pointer reaches the end, then return the CSV file to the admin and exit the web scraper. • Gadgets specification's dataset 1. Open the Product Specifications CSV file generated using a web scraper. 2. Perform data cleaning and data processing on each column. 3. Save the changes to the CSV file and close the file. • Recommendation algorithm for gadget recommendation 1. Get user input from the input form. 2. Select the products from the CSV file in the given price range. 3. Compare user input with all the product specifications column-wise from step 'b' and store the difference in columns named such as a1,a2,a3 and so on. 4. Multiply a1, a2, a3,... column values with their respective priorities (weights) and store it again to a1, a2, a3,...columns. 5. Take row-wise summation of a1,a2,a3, a2, a3,... values and store it in column B. 6. Sort the CSV file in descending order with respect to values from column B. 14 7. Share the first 2 rows addressed to the recommendation view com- ponent. • Product recommendation view 1. Get the JSON objects of the top two recommended gadgets from the recommender algorithm. 2. Using that these objects fetch details from Gadgets specification's dataset. 3. Print those details using the predefined layout. 15 4.3.1 Dataflow Diagram Figure 4 shows level 0 DFD. It is a basic overview of the whole system. Figure 4: Data flow Diagram level 0 Figure 5 shows DFD level 1 and it mentions sub processes that together form the complete system. Figure 5: Data flow Diagram level 1 16 4.3.2

Sequence Diagram Figure 6 Shows object interactions arranged in time sequence in the field of software engineering Figure 6: Sequence Diagram 17 4.3.3 Activity Diagram Figure 7 shows activity diagram for the gadget recommendation system which depicts graphical representation of step wise activities and actions in the system. Figure 7: Activity Diagram 18 4.3.4 Component Diagram Figure 8 describes the organization and wiring of the physical components in a system Figure 8: Component Diagram 4.3.5 Deployment Diagram Figure 9

shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them . Figure 9: **Deployment** 8

Diagram 19 5 Pro ject Management and cost 5.1 Pro ject Scheduling Plan Task Name Duration Start Date End Date Assigned To Priority Domain selection 7 days 01-07-2021 07-07-2021 High Domain finalization 7 days 08-07-2021 14-07-2021 Medium Selection of problem statement 14 days 15-07-2021 28-07-2021 High Finalization of problem statement 7 days 29-07-2021 04-08-2021 High Study on Research pa- per 14 days 05-08-2021 18-08-2021 High Documentation of synopsis 14 days 19-08-2021 01-09-2021 Medium Requirement analysis 7 days 02-09-2021 08-09-2021 Medium System requirements 7 days 09-09-2021 15-09-2021 High Module identification 7 days 16-09-2021 21-09-2021 High System architecture 7 days 23-09-2021 30-09-2021 High Implementation 25% 14 days 01-12-2021 14-12-2021 High Testing 25% 7 days 15-12-2021 21-12-2021 High Implementation 50% 14 days 22-12-2021 04-01-2022 High Testing 50% 7 days 05-01-2022 11-01-2022 High Implementation 75% 14 days 12-01-2022 25-01-2022 High Testing 75% 7 days 26-01-2022 01-02-2022 High Implementation 100% 14 days 02-02-2022 05-02-2022 High Testing 100% 7 days 16-02-2022 22-02-2022 High Report Making 14 days 23-02-2022 09-03-2022 High Table 3: Caption 5.2 Project Cost The following is the pricing as per the Google Firebase Blaze plan: • Hosting Storage INR 1.95/GB Data Transfer INR 11.25/GB Table 4: Hosting charges At the initial stage hosting will use less than 5GB storage. 20 • Real Time Databases GB Stored INR 375/GB GB Downloaded INR 75/GB Table 5: Database charges At the initial stage databases will use less than 2GB storage • Domain charges : INR 800/year At the initial stage the total costing will be around INR 2000/year 21 6 Implementation 6.1 Environmental Setting for Running the Project 1. Installation of IDE(Visual Studio code) (a) Visual Studio code is downloaded from official website:- https://code.visualstudio.com/ 2. Installation of node.js :- (a) Node.js is downloaded from official website:- https://nodejs.org/en/download/ (b) Open the Command prompt run as a administrator.To check the version of node type the command node -v 3. Installation of typescript :- (a) Open the Command prompt in administrator mode and type the command npm install -g typescript 4. Installation of Angular CLI (a) Open the Command prompt in administrator mode and type the command :- npm install -g @angular/cli (b) Command to check the angular version - ng v 6.2 Detailed Description of Methods 1. User Login / Signup • Users will be asked to fill a form for both login and signup. • If the user is new then the system allows the user to create an account. • If the user is signing up the data will be stored in the database. • If the user is logging in the data will be authenticated using the database. 2. User form • After the user visits the platform, they will have to answer certain questions. • These questions will be according to the gadget they want to be recommended. • There requirements will be further sent to the recommendation algo- rithm 3. Web Scraping • Using a web scraper the specifications of various devices from GS- Marena, Flipkart or Amazon will be collected. • It will be implemented using

Beautiful Soup or Scrapy Python library. • The data collected will be further processed and cleaned 22 4. Data pre-processing and Cleaning • The data obtained from web scraping will be processed and cleaned. • After the data is in a useful format it will be stored in a csv file in the database 5. Recommendation Algorithm • The user requirements will work as input for the algorithm. • Using the similarity metrics along with the priorities of the specifi- cations, it will find out the product that matches most of the user requirements from the database. • It will be sent to the product recommendation module 6. Product recommendation • The product specifications, image, pros and cons, best buy links, etc will be displayed 6.3 Implementation Details The takeInput function consists of the logic that takes input from the user and recommends the product according to the requirements entered by the user. Input to this function is parameters taken from the user and the respective product category's dataset. The function returns the top two recommended products function takeInput(data: array of JSON objects, input obj: JSON object) Output - Returns an array of objects of products BEGIN param list = ['Design','Display','Software','Performance','Battery','Camera'] user req = [] FOR i = 0 to param list.length-1 user req[i] = parseInt(input obj[param list[i]]) END FOR priority ids =['Design p','Display p','Software p','Performance p','Battery p','Camera p'] priority list = [] FOR i = 0 to priority ids.length-1 priority list[i] = parseInt(input obj[priority ids[i]]) END FOR user price = parseInt(input obj.Price) affordable=[]; FOR i=0 to data.length-1 IF parseInt(data[i].price) ¡= user price) THEN affordable.push(data[i]); END IF END FOR FOR i = 0 to data.length-1 data[i]['Design diff'] = (parseInt(data[i].design)-user req[0])*priority list[0] data[i]['Display diff'] = (parseInt(data[i].display)user req[1])*priority list[1] 23 data[i]['Software diff']= (parseInt(data[i].software)-user req[2])*priority list[2] data[i]['Performance diff']=(parseInt(data[i].performance)-user req[3])*priority list[3] data[i]['Battery diff']=(parseInt(data[i].battery life)-user req[4])*priority list[4] data[i]['Camera diff']= (parseInt(data[i].camera)-user req [5])*priority list[5] END FOR FOR i = 0 to data.length-1 data[i]["main diff"] = data[i].Camera diff+ data[i].Performance diff+ data[i].Display diff+ data[i].Battery diff+ data[i].Software diff+data [i].Design diff END FOR data.sort((a,b) =¿ b.main diff-a.main diff) obj=[] obj.push(data[0]) obj.push(data[1]) return obj END 24 7 Integration and Testing 7.1 Description of the Integration Modules The login module avails the user with forms for the login and signup. If the user has already created an account then login form is displayed to the user. The data(Email ID, Password) taken through the signup form is checked if it follows the database constraints. After validation it is stored in the database. If the user is new then the system allows the user to create an account. The login details(Email ID, Password) will be authenticated using the database. If the data authentication fails then an error message is shown on the login page. The user chooses the type of the gadget on the discover page. Then the user has to answer the certain questions asked in the form according to the gadget needed to be recommended. After checking that the required fields are filled and passing the constraints, the inputs are provided to the recommendation algorithm for further processing. Before the recommendation algorithm runs the dataset about the gadgets is created using the web scraping module. A website URL is taken as the input then the HTML elements are fetched. Required elements are identified using the element ID. Data of the elements that pass the constraints is stored in the CSV file. Once the CSV file is created using the web scraper, it needs to be processed. Certain operations like data cleaning and data processing are performed on the data and the changes made are saved to the CSV file. The algorithm then takes the user inputs. The products are selected from the CSV file which are in the given price range. Then these products are compared with all the product specifications provided by the user. The difference is taken column wise and it is stored in corresponding rows. Then the difference is multiplied with the priorities (weights) and these values are stored. Then row wise summation is taken and the values are stored in corresponding rows. These are the final values

that are used for the recommendations. The CSV file is then sorted using the values calculated by the algorithm in descending order and the first two values from the list are shared with the view component. To view the products , the pointer from the algorithm is taken and using that pointer the gadget specifications are fetched from the dataset. Then the product details are displayed on the predefined layout. 25 7.2 Testing Sr No. Test Case Title Description Expected Outcome 1 Successful User Regis- tration The Registra- tion to the system is successful because all the Mandatory fields and proper data is filled by the user. Registra- tion should be success- ful The Registration to the sys- tem is successful because all the Mandatory fields and proper data is filled by the user. Registration should be successful 2 Unsuccessful user Reg- istration due to Manda- tory fields not filled. Registration to the system with unfilled mandatory fields. Registration should fail with an error 'missing required field'. 3 Unsuccessful user Reg- istration due to Invalid data Registration to the system with Invalid data. For ex., • Email address is not valid • Mobile number is not 10 digit Registration should fail with an error 'Invalid Data' 4 Successful User Verifi- cation Successful User Verifica- tion.The login to the system should be tried with the correct admin and password. Login should be successful and the user successfully enter in to the system Login should be suc- cessful and the user suc- cessfully enter in to the system 5 Unsuccessful User Ver- ification due to wrong password Wrong password is typed while login. Unsuccessful User Ver- ification due to wrong password Wrong pass- word is typed while login.Login should fail with an error message 'Invalid Password' 6 Successful gadget Rec- ommendation. After entering into the system the user gives all the answers properly. Recommendation should be successful and users can access the recommendation. 7 Unsuccessful Recom- mendation due to wrong device name or other data After entering into the system if the user gives the wrong device name(device names that are not available) or gives wrong data. Recommendation should fail with an error 'Wrong device name' Table 6: Test Case Table 26 8 Performance Analysis • User interface To interact with the system, the interface provided is much simpler, min- imal and responsive. As the user interface is web based it is platform independent and anyone with internet connection can access it from any- where. It is created using Angular.js with a single page design so that users can have an experience like they are using an application on a de- vice. The questions in the form are easy to understand for a typical user without any technical background. This makes the system user friendly and provides better usability. • User Time Once the user visits the website it will take hardly 1 to 2 minutes to fill the login details. After login, users can explore the discover page and choose the gadget from the categories section in about 3 or 5 minutes. To fill the form, about 6 to 7 minutes of average time is needed. Then the recommendations are shown and the user can take as much time as he/she requires to select a product. Hence it will take about 15 to 20 minutes to decide the desired product. The system makes the decision making process fast and convenient, saving the user's valuable time. • Correctness The gadgets that are recommended by the system are based on the user requirements and priorities in order to fulfill all the requirements of the user. The algorithm selects only those products which have a higher sim- ilarity score that aligns with the user's expectations. As inputs are taken from the actual user, the results generated are the best fit in the user's perspective which implies the correct execution of the system. • Response Time As Angular.js is used to create the web app, major events like authentica- tion and routing are executed on the client side. This results in reducing the response time as there is no need to send requests to the server for loading the pages 27 9 Future Scope Nowadays digital devices are part of the day-to-day life of every person. Digital devices have come to be essential in basic needs of individuals. As the technology developed rapidly, it is essential to meet

different functional needs of end-users. Therefore it is necessary **to suggest gadgets** to **the customers** by knowing **their** 6

interest. Digital

devices and communication networks are increasing **to cover all the** aspects **of customer's activity. These data can also be trained and** modeled **for future use to cope with the upcoming technological innovations** . 28 10 Applications **1** 6

. Primary consumer This system will give the best suitable suggestions to the primary con- sumers by just taking their requirements. So while buying any electronic smart gadget they can use this system for taking suggestions. By using this system the primary consumer can meet their technical requirements and also get satisfied. 2. Gadget sellers Both online sellers and offline sellers can use this system to give the best suggestions about gadgets to their customers. By giving best suggestions the gadget seller helps customers to fulfill their needs. Due to this the customer also gets satisfied and gadget sellers can easily sell their gad- gets. So by using this system the gadget sellers can make more and more customers and can sell more gadgets. So it will increase their profit 3. Gadget Companies By using this system gadget companies will know about which gadgets the customer likes most and what are the requirements of the customer. This system will help companies to study the product market. After studying the product market, gadget companies can easily make more and more decisions about gadgets. 29 11 Installation Guide and User Manual 1. Link of the website is given to the user. 2. Users need a good internet connection and any web browser like Google chrome, Microsoft edge, etc 3. Users should have to click on the given link or just copy paste that link on to the available web browser. 4. After clicking on the link the home page will open. 5. Users should have to click on the registration button if he/she is using the site for the first time, otherwise choose the login option. Fill the correct details and also fill all the mandatory fields. Then click on register button. 6. If User is already registered, choose the login option and fill in the correct username and password which is given at the time of registration and then click on the login button. 7. After landing on home page, the system will show one form which is based on user requirements. User have to fill that form and submit the form. 8. After successfully submitting the form the user will get the recommended gadget along with specifications of that gadget. User will also get an alter- native recommended gadget and make-in-India alternative recommended gadget. 9. If a user wants to see the comparison between recommended gadget and alternative recommended gadget, User has to click on 'show comparison' button. 10. If a user wants to see the best links from where they can purchase the recommended gadget, users have to click on the 'show links' button. 11. After completion of use, users have to logout by clicking on the logout button. 30 12 Bibliography References [1] S. Shaikh, S. Rathi, and P. Janrao, "Recommendation system in e-commerce websites: A graph based approached," in 2017 IEEE 7th International Ad- vance Computing Conference (IACC), 2017, pp. 931–934. [2] P. Singh, P. Dutta Pramanik, A. Dey, and P. Choudhury, "Recommender systems: An overview, research trends, and future directions," International Journal of Business and Systems Research, vol. 15, p. 14–52, 01 2021. [3] G. Linden, B. Smith, and J. York, "Amazon.com

recommendations: item- to-item collaborative filtering," IEEE Internet Computing, vol. 7, no. 1, pp. 76–80, 2003. [4] D. S. Ken Arnett, Z. K. A. Baizal, and Adiwijaya, "Recommender system based on user functional requirements using euclidean fuzzy," in 2015 3rd International Conference on Information and Communication Technology (ICoICT), 2015, pp. 455–460. 31

sources:

1 278 words / 4% - Internet from 27-Oct-2021 12:00AM
techvaluetrends.com

2 155 words / 2% - Internet from 19-Oct-2019 12:00AM
paper.ijcsns.org

3 98 words / 1% - Internet from 05-Sep-2017 12:00AM
ijarcsse.com

4 63 words / 1% - Internet from 01-Feb-2021 12:00AM
blog.knoldus.com

5 55 words / 1% - Internet from 23-Jun-2017 12:00AM
adiwijaya.staff.telkomuniversity.ac.id

6 45 words / 1% - Crossref
Pushpalatha A, Harish Sanmugam J, Jeya Pradeepa, Madhu Bala S. "A Gadget Recommendation System using Data Science", Journal of Information Technology and Digital World, 2020

7 25 words / < 1% match - Internet from 12-Apr-2022 12:00AM
www.coursehero.com

8 22 words / < 1% match - Internet from 30-Nov-2020 12:00AM
www.slideshare.net

9 20 words / < 1% match - Crossref
Rabi Narayan Behera, Sujata Dash. "A Particle Swarm Optimization based Hybrid Recommendation System", International Journal of Knowledge Discovery in Bioinformatics, 2016

10 18 words / < 1% match - Crossref
Nikos Tsianos, Panagiotis Germanakos, Zacharias Lekkas, Constantinos Mourlas. "chapter 1 Assessment of Human Factors in Adaptive Hypermedia Environments", IGI Global, 2009

11 18 words / < 1% match - Internet from 07-Feb-2022 12:00AM
www.grin.com

12 17 words / < 1% match - Internet from 12-Jun-2017 12:00AM
repository.uinjkt.ac.id