**A PROJECT REPORT ON**

**On**

# PRUS: Product Recommendation System Based on User Specification and Customer Reviews

**Submitted in the partial fulfilment of requirements to**

**CS – 363 – Term Paper**

**By**

**Batch – 19**

**Oruganti Monik Paparao (Y22CS139)**

**Pendyala Skanda Bhagavan (Y22CS145)**

**Tulam Sai Sudheer (Y22CS184)**


**R.V.R. & J.C. COLLEGE OF ENGINEERING (Autonomous)**

**(NAAC 'A+' Grade)**

**Approved by AICTE :: Affiliated to Acharya Nagarjuna University**

**Chandramoulipuram::Chowdavaram**

**GUNTUR – 522 019, Andhra Pradesh, India.**

# Contents:

# Goal of the Project

The research paper titled **"PRUS: Product Recommender System Based on User Specifications and Customers Reviews"** introduces a new and improved way to recommend products to users by understanding their specific needs more accurately. Most existing recommendation systems only focus on the overall sentiment of product reviews or highlight only the positive aspects. However, this approach can overlook important negative feedback that may affect a user's decision. To solve this problem, the proposed system, called **PRUS**, takes into account both **positive and negative sentiments** found in customer reviews.

One of the main features of PRUS is that it allows users to **specify exactly which product features they care about**, such as camera quality, battery life, or screen resolution. Instead of looking at the entire review as one opinion, the system breaks the reviews down into **individual sentences** and identifies opinions related to each feature. It then performs **aspect-level sentiment analysis**, which means it understands the user's opinion on each feature separately, rather than as a whole.

To rank the products effectively, the authors introduce a method called **RANK-ify**, which assigns scores to each product based on how often and how positively or negatively each feature is mentioned. This scoring method also gives users the flexibility to decide whether they want to prioritize positive feedback, negative feedback, or both equally.

Overall, the goal of the paper is to create a more **personalized, accurate, and useful product recommendation system** that helps users make better decisions by understanding detailed customer opinions, not just average ratings or general reviews.

# Mechanism to be Followed

Here's the **step-by-step mechanism** of how the PRUS system works, explained in simple words:

---

**Step 1: Collect Product Reviews**

The system starts by collecting a large number of **customer reviews** from platforms like Amazon, focusing on products such as mobile phones.

---

**Step 2: Clean and Prepare the Reviews**

The reviews are **cleaned** by removing useless data like short or incomplete reviews, stopwords, punctuation, and numbers. Words are also simplified using techniques like **lemmatization** so that similar words are treated as the same.

---

**Step 3: Break Reviews into Sentences**

Each review is **split into sentences** to find specific opinions about different features (like battery, camera, etc.) rather than looking at the whole review.

---

**Step 4: Extract Features and Sentiments**

From each sentence, the system **identifies the product feature** being discussed (e.g., "battery") and the **sentiment** (positive, negative, or neutral) expressed about that feature using tools like **TextBlob**.

---

**Step 5: Match Features with User Query**

The user gives a **search query** specifying the features they care about (e.g., "good camera, long battery life"). The system matches these requested features with the features found in the reviews.

---

**Step 6: Assign Scores to Features**

Each product is assigned **scores** for each feature based on how many times that feature is mentioned positively or negatively. A formula is used to give more or less importance to positive or negative feedback, depending on what the user prefers.

---

**Step 7: Rank Products (RANK-ify Algorithm)**

The **RANK-ify algorithm** calculates a total score for each product by combining all feature scores. Products are then **ranked** from best to worst based on how well they match the user's needs.

---

**Step 8: Display Recommended List**

Finally, the system shows a **ranked list of recommended products** that best fit the user's specified features and preferences, considering both good and bad reviews.

---

# Tools, and Algorithms needed for Implementation

## 🧩 Frontend (React.js)

| Tool / Library | Purpose |
|---|---|
| React.js | Frontend JavaScript library to build UI |
| Axios or Fetch API | To send HTTP requests to backend |
| React Router | For page navigation (e.g., `/products`, `/home`) |
| Tailwind CSS / Bootstrap | UI styling and responsiveness |
| Vite / Create React App | For creating and managing the React app |

## 🧠 Backend (Python)

| Tool / Library | Purpose |
|---|---|
| FastAPI | REST API backend (highly recommended for modern APIs) |
| Flask (alt.) | Lightweight web framework (also suitable for REST APIs) |
| TextBlob / NLTK / spaCy | For sentiment and feature extraction in reviews |
| Pandas / NumPy | Data processing, managing review datasets |
| Uvicorn | ASGI server to run FastAPI |
| Pydantic | Request validation and type-checking in FastAPI |
| scikit-learn (optional) | For ML-based models if needed for ranking |

## 🗃️ Database (if needed)

| Tool | Purpose |
|---|---|
| MongoDB | If you prefer NoSQL-style document storage |

## 🛠️ Development Tools

| Tool | Purpose |
|---|---|
| VS Code | Recommended IDE for both React and Python |
| Postman | To test API endpoints during backend development |
| Git + GitHub | Version control and code hosting |
| Node.js + npm | Required to run React.js and install packages |
| Python 3.9+ | For backend development |

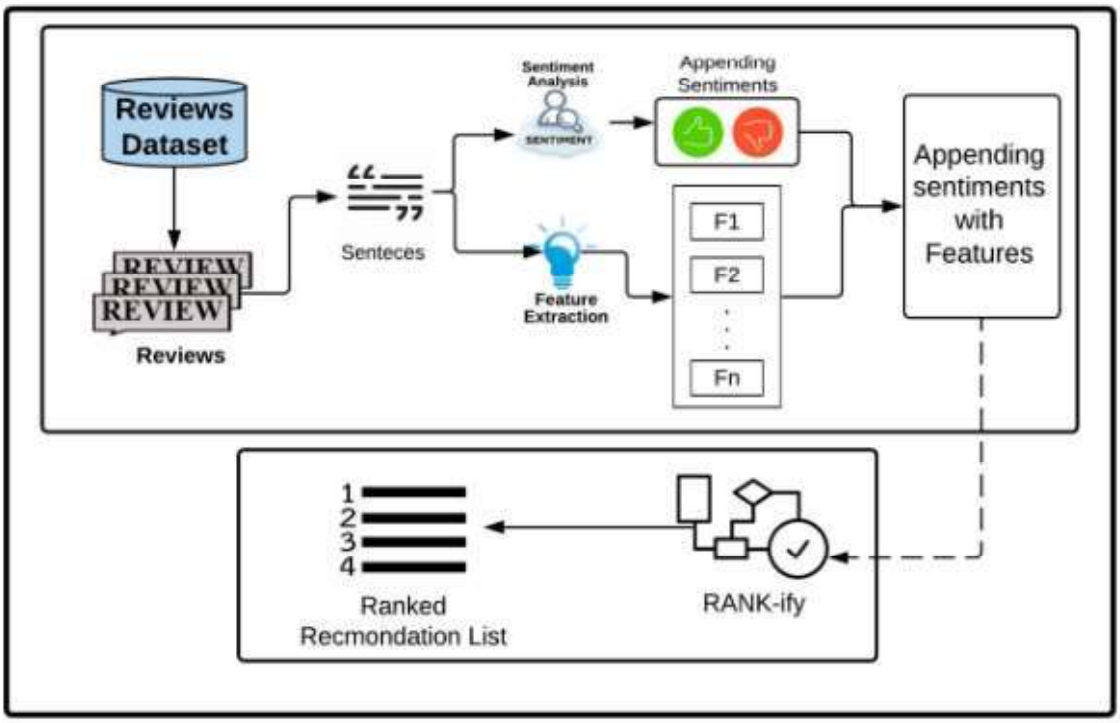
## 🌐 Deployment Tools (Later Stage)

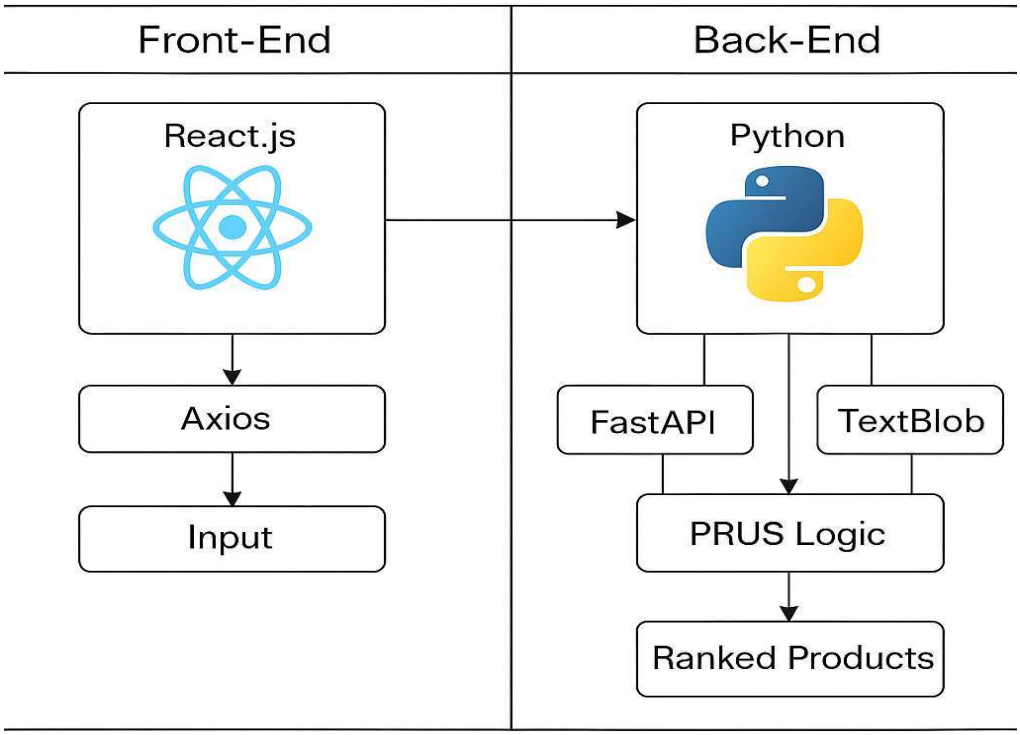
| Tool | Purpose |
|---|---|
| Vercel / Netlify | To deploy React frontend |
| Render / Railway / Heroku | To deploy FastAPI/Flask backend |

**Algorithm 1** RANK-Ify Algorithm

$k$ =Number of Products in Dataset
$L_p$ =List of Products in Dataset
$Q$ =Features in User Query
**for** $p \in L_p$ **do**
  **for** $U_f \in Q$ **do**
    **if** $U_f \in p$ **then**
      $FS(U_f) = FeaSco(U_f)$
    **end if**
  **end for**
  $RS(p) = \sum_{U_f \in Q} FS(U_f)$
**end for**
Sort $L_p$ w.r.t. $RS$
Return top-k from $L_p$

# Architecture Diagram



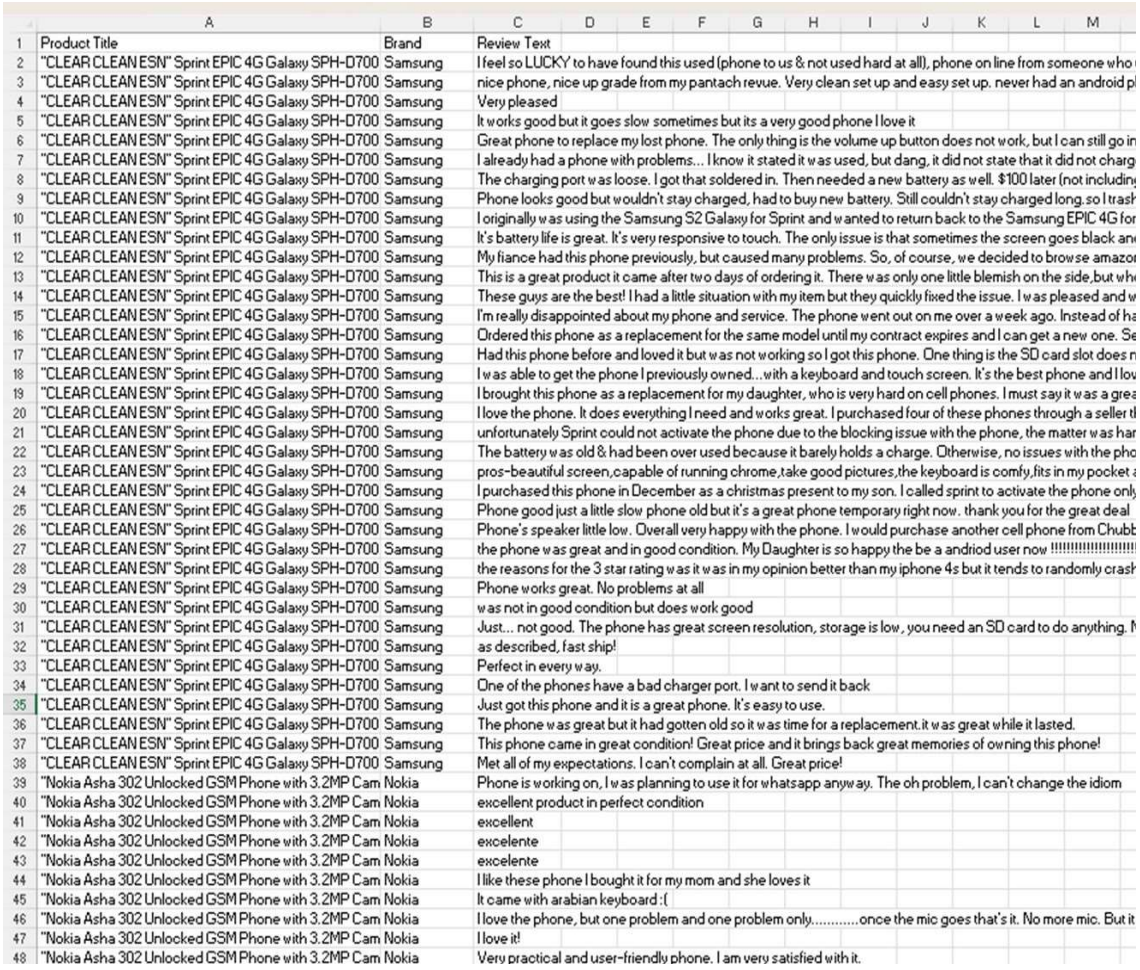



# Architecture Overview

The architecture of the proposed product recommendation system is divided into two main components: the **frontend** developed using **React.js**, and the **backend** built with **Python** using frameworks like **FastAPI**. On the frontend, users interact with a clean and responsive interface built in React. They enter specific product preferences, such as “long battery life” or “high-resolution screen,” through input fields or forms. Once the user submits their preferences, the React application uses a library called **Axios** to send an HTTP POST request to the backend API. This request contains the user-specified features and optionally assigned weights that indicate the importance of positive or negative sentiments. The React app is also responsible for displaying the final list of recommended products returned by the backend in a user-friendly format.

On the backend side, **FastAPI** receives the incoming request and forwards the data to the **core PRUS logic**, which is implemented in Python. The backend system uses **TextBlob** (or similar NLP tools) to process a large number of customer reviews, breaking them into sentences, extracting product features, and assigning sentiment polarity (positive or negative) to each feature. This information is then fed into the **RANK-ify algorithm**, which calculates a ranking score for each product based on how well it matches the user’s query, considering both the frequency and sentiment strength of each feature. The products are sorted accordingly, and a top-N list is created. This ranked list is then sent back through the FastAPI response to the React frontend, where it is rendered for the user. This architecture ensures a modular, scalable, and highly personalized product recommendation experience based on actual customer feedback.

# Data Set Description



| | A | B | C |
|---|---|---|---|
| 1 | Product Title | Brand | Review Text |
| 2 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I feel so LUCKY to have found this used (phone to us & not used hard at all), phone on line from someone who |
| 3 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | nice phone, nice up grade from my pantach revue. Very clean set up and easy set up. never had an android p |
| 4 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Very pleased |
| 5 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | It works good but it goes slow sometimes but its a very good phone I love it |
| 6 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Great phone to replace my lost phone. The only thing is the volume up button does not work, but I can still go in |
| 7 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I already had a phone with problems... I know it stated it was used, but dang, it did not state that it did not charg |
| 8 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | The charging port was loose. I got that soldered in. Then needed a new battery as well. $100 later (not includin |
| 9 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Phone looks good but wouldn't stay charged, had to buy new battery. Still couldn't stay charged long so I trash |
| 10 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I originally was using the Samsung S2 Galaxy for Sprint and wanted to return back to the Samsung EPIC 4G for |
| 11 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | It's battery life is great. It's very responsive to touch. The only issue is that sometimes the screen goes black an |
| 12 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | My fiance had this phone previously, but caused many problems. So, of course, we decided to browse amazo |
| 13 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | This is a great product it came after two days of ordering it. There was only one little blemish on the side,but wh |
| 14 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | These guys are the best! I had a little situation with my item but they quickly fixed the issue. I was pleased and w |
| 15 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I'm really disappointed about my phone and service. The phone went out on me over a week ago. Instead of ha |
| 16 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Ordered this phone as a replacement for the same model until my contract expires and I can get a new one. Se |
| 17 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Had this phone before and loved it but was not working so I got this phone. One thing is the SD card slot does n |
| 18 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I was able to get the phone I previously owned...with a keyboard and touch screen. It's the best phone and I lov |
| 19 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I brought this phone as a replacement for my daughter, who is very hard on cell phones. I must say it was a grea |
| 20 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I love the phone. It does everything I need and works great. I purchased four of these phones through a seller t |
| 21 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | unfortunately Sprint could not activate the phone due to the blocking issue with the phone, the matter was har |
| 22 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | The battery was old & had been over used because it barely holds a charge. Otherwise, no issues with the pho |
| 23 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | pros-beautiful screen,capable of running chrome,take good pictures,the keyboard is comfy,fits in my pocket |
| 24 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | I purchased this phone in December as a christmas present to my son. I called sprint to activate the phone only |
| 25 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Phone good just a little slow phone old but it's a great phone temporary right now. thank you for the great deal |
| 26 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Phone's speaker little low. Overall very happy with the phone. I would purchase another cell phone from Chubb |
| 27 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | the phone was great and in good condition. My Daughter is so happy the be a andriod user now !!!!!!!!!!!!!!!!!!! |
| 28 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | the reasons for the 3 star rating was it was in my opinion better than my iphone 4s but it tends to randomly crash |
| 29 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Phone works great. No problems at all |
| 30 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | was not in good condition but does work good |
| 31 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Just... not good. The phone has great screen resolution, storage is low, you need an SD card to do anything. |
| 32 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | as described, fast ship! |
| 33 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Perfect in every way. |
| 34 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | One of the phones have a bad charger port. I want to send it back |
| 35 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Just got this phone and it is a great phone. It's easy to use. |
| 36 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | The phone was great but it had gotten old so it was time for a replacement.it was great while it lasted. |
| 37 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | This phone came in great condition! Great price and it brings back great memories of owning this phone! |
| 38 | "CLEAR CLEAN ESN" Sprint EPIC 4G Galaxy SPH-D700 | Samsung | Met all of my expectations. I can't complain at all. Great price! |
| 39 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | Phone is working on, I was planning to use it for whatsapp anyway. The oh problem, I can't change the idiom |
| 40 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | excellent product in perfect condition |
| 41 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | excellent |
| 42 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | excelente |
| 43 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | excelente |
| 44 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | I like these phone I bought it for my mom and she loves it |
| 45 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | It came with arabian keyboard :( |
| 46 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | I love the phone, but one problem and one problem only.....once the mic goes that's it. No more mic. But it |
| 47 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | I love it! |
| 48 | "Nokia Asha 302 Unlocked GSM Phone with 3.2MP Cam | Nokia | Very practical and user-friendly phone. I am very satisfied with it. |

- **Name:** Amazon Unlocked Mobile Review Dataset
- **Total Entries (Rows):** 413,840
- **Total Columns (Features):**3
- *Source*: **Kaggle Data set**
- *Data_set_Link*: **https://www.kaggle.com/datasets/PromptCloudHQ/amazon-reviews-unlocked-mobile-phones/data**

# Functional Requirements

These define **what the system should do**:

1. **User Input of Preferences**
   Users should be able to enter specific product features (e.g., "battery life", "camera") via a web form.

2. **Submit Query to Backend**
   The system should send the user preferences and sentiment weights (positive/negative) to the backend API using Axios.

3. **Sentiment Analysis of Reviews**
   The backend should process reviews, perform sentiment analysis on each sentence, and identify feature-level sentiment using NLP tools like TextBlob.

4. **Feature Matching and Scoring**
   The backend should match user-specified features with review data and calculate ranking scores using the RANK-ify algorithm.

5. **Generate Ranked Product List**
   Based on user input and sentiment analysis, the system should return a list of ranked products.

6. **Display Results on Frontend**
   The ranked list of recommended products should be displayed in a clear and user-friendly way (e.g., cards, list, table).

7. **Handle Errors and Invalid Input**
   The system should validate input and gracefully handle missing data, API failures, or incorrect user queries.

# Non-FunctionalRequirements

These define **how the system should behave**:

1. **Performance**
   The system should return ranked results within a few seconds of receiving the user input.

2. **Scalability**
   The backend should be designed to handle large datasets (e.g., thousands of reviews) and many users simultaneously.

3. **Security**
   Input should be sanitized to prevent injection attacks. Secure communication (e.g., HTTPS) should be used in production.

4. **Usability**
   The frontend should be intuitive, responsive (mobile-friendly), and easy to use for non-technical users.

5. **Maintainability**
   The codebase should be modular and well-documented to allow future updates or improvements to the algorithm or UI.

6. **Reliability**
   The system should remain available and correctly function even if one part (e.g., review sentiment) encounters errors.

7. **Portability**
   The application should be deployable across various environments (local, cloud platforms like Vercel/Render).