**A HIGH LEVEL DESIGN**

**On**

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

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

**CS - 454 - Project Work**

**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.**

**Smt. N. Zareena Dr. S. J. R. K. Padmini Valli Dr. M. Sreelatha**

**Project Guide Project In-Charge Prof. & HOD**

**Contents:**

**1. Problem Statement**

**2. Functional Requirements**

**3. Basic Architecture**

**a. Major components**

**b. Diagram**

**c. Workflow**

**d. Pattern**

**4. Non Functional Requirements**

**5. Technology Stack**

**6. Summary**

1. **Problem Statement**

**Manual product recommendation systems often rely on overall product ratings or general customer reviews, which can overlook important details about specific product features. Most existing systems do not allow users to specify which features matter most to them, resulting in less personalized recommendations.**

Additionally, many approaches focus only on positive sentiments, ignoring valuable negative feedback that could influence buying decisions. PRUS addresses these challenges by applying aspect-level sentiment analysis to customer reviews, breaking them down into feature-specific opinions such as camera quality, battery life, or screen resolution. This system enables users to specify their preferences and ranks products using a novel scoring method called RANK-ify, which considers both positive and negative sentiments for each feature.

Key goals include:

* User-specified feature-based recommendations.
* Aspect-level sentiment analysis of customer reviews.
* Flexible ranking with emphasis on positive, negative, or balanced feedback.
* Enhanced personalization and decision support for buyers.

1. **Functional Requirements**

|  |  |
| --- | --- |
| Functionality | Description |
| User Input of Preferences | Allow users to enter specific product features (e.g., battery life, camera) via a web form. |
| Submit Query to Backend | Send user preferences and sentiment weights to backend API using Axios for processing. |
| Sentiment Analysis of Reviews | Backend performs sentence-level sentiment analysis and identifies feature-specific sentiments using NLP tools like TextBlob. |
| Feature Matching and Scoring | Match user-specified features with review data and calculate product scores using the RANK-ify algorithm. |
| Generate Ranked Product List | Produce a ranked list of products based on user input and sentiment analysis results. |
| Display Results on Frontend | Show the ranked product recommendations clearly using cards, lists, or tables on the frontend. |
| Handle Errors and Invalid Input | Validate user input and handle missing data, API failures, or incorrect queries gracefully |

**4. Non Functional Requirements**

|  |  |
| --- | --- |
| Category | Description |
| Performance | The system should return ranked results within a few seconds of receiving the user input. |
| Scalability | The system should send the user preferences and sentiment weights (positive/negative) to the backend API using Axios. |
| Security | The backend should be designed to handle large datasets (e.g., thousands of reviews) and many users simultaneously. |
| Usability | The frontend should be intuitive, responsive (mobile-friendly), and easy to use for non-technical users |
| Maintainability | The codebase should be modular and well-documented to allow future updates or improvements to the algorithm or UI. |
| Reliability | The system should remain available and correctly function even if one part (e.g., review sentiment) encounters errors. |
| Portability | The application should be deployable across various environments (local, cloud platforms like Vercel/Render). |

**6. Summary**

**This project develops a personalized product recommendation system where users can specify important product features like battery life or camera quality through a simple web form. Users can also choose how much weight to give positive or negative feedback, making the recommendations more tailored to their needs.**

The backend analyzes customer reviews using natural language processing to perform aspect-level sentiment analysis, breaking reviews into sentences and identifying sentiments related to specific features. This detailed approach helps the system understand how different product aspects are perceived.

Using the RANK-ify algorithm, the system matches user preferences with the sentiment data to score and rank products based on how well they meet the user’s criteria. This allows for flexible recommendations that consider both positive and negative opinions.

Finally, the ranked products are displayed clearly on the frontend, with input validation and error handling to ensure a smooth experience even if issues arise.