**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TOPICS** | **PAGE NO** |
|  | **Introduction**  1.1 Introduction of the System  i. Project Title  ii. Category  iii. Overview  1.2 Background  i. Introduction of the Company  ii. Brief note on Existing System  1.3 Objectives of the System  1.4 Scope of the System  1.5 Structure of the System  1.6 System Architecture  1.7 End Users  1.8 Software/Hardware required for the Development  1.9 Software/Hardware required for the Requirements |  |
|  | **SRS**  2.1 Introduction (Brief write-up about SRS)  2.2 Overall Description  i. Product perspective  ii. Product Functions  iii. User characteristics.  iv. General constraints  v. Assumptions  2.3 Functional requirement  i. Module 1  ii. Module 2 ‘  iii. Module 3  iv. Module 4  v. Module 5  2.4 Design Constraints  2.5 System Attributes |  |
|  | **System Design (Functional Design)**  3.1 Introduction (brief write-up about System Design)  3.2 Assumptions and Constraints  3.3 Functional decomposition  3.4 Description of Programs  i. Context Flow Diagram (CFD)  ii. Data Flow Diagrams  iii. Use Case Diagram  iv. Activity diagram  3.5 Description of components  i. Functional Component 1  ii. Functional Component 2 |  |
|  | **Database/Datasets used**  4.1 Introduction (brief write-up about Datasets)  4.2 Purpose and scope |  |
|  | **User Requirement System Planning and Scheduling**  5.1 Pert Chart  5.2 Gantt Chart |  |
|  | **Program code listing** |  |
|  | **User Interface (Screens and Reports)** |  |
|  | **Testing**  8.1 Introduction (brief write-up about Software Testing)  i. Unit Testing  ii. Integrate Testing  iii. System Testing |  |
|  | **Conclusion** |  |
|  | **References** |  |

**1.Introduction**

1. **Introduction of the System**
2. **Project Title -** "Sentimart: Sentiment analysis of product reviews"
3. **Category -** Web-Based Sentiment Analysis Application using Streamlit and NLP
4. **Overview –** The *Sentiment Analysis of Product Reviews* project is a web-based application designed to analyze customer feedback from Amazon product pages using Natural Language Processing techniques. The system classifies customer reviews into Positive, Neutral, and Negative sentiments through VADER Sentiment Analysis and displays interactive visualizations like word clouds, pie charts, and bar charts. Additionally, users can enter their own reviews to receive instant sentiment analysis results. The application not only helps customers make informed purchasing decisions but also assists businesses in understanding product reception and improving their services.

**1.2 Background**

**i. Introduction of the Company**

Amazon is one of the world’s leading e-commerce platforms, offering a vast range of products across various categories. With millions of customer reviews available on its platform, analyzing these reviews is crucial for both consumers and sellers to make informed decisions and optimize product offerings.

**ii. Brief note on Existing System**

Existing sentiment analysis applications have several limitations:

* Lack of Real-Time Data Collection: Most tools cannot fetch live reviews directly from product pages.
* Limited Visualization: Many systems rely on simple sentiment scores without interactive visual insights.
* No User Review Testing: Most platforms do not allow manual input of user reviews for analysis.
* Difficult User Interface: Some systems are complex and challenging for non-technical users.

**1.3 Objectives of the System**

* To classify customer reviews into Positive, Neutral, and Negative sentiments using VADER NLP.
* To provide real-time sentiment analysis by scraping reviews from Amazon product URLs.
* To visualize sentiment distribution through interactive charts and word clouds.
* To enable users to input their own reviews for instant sentiment feedback.

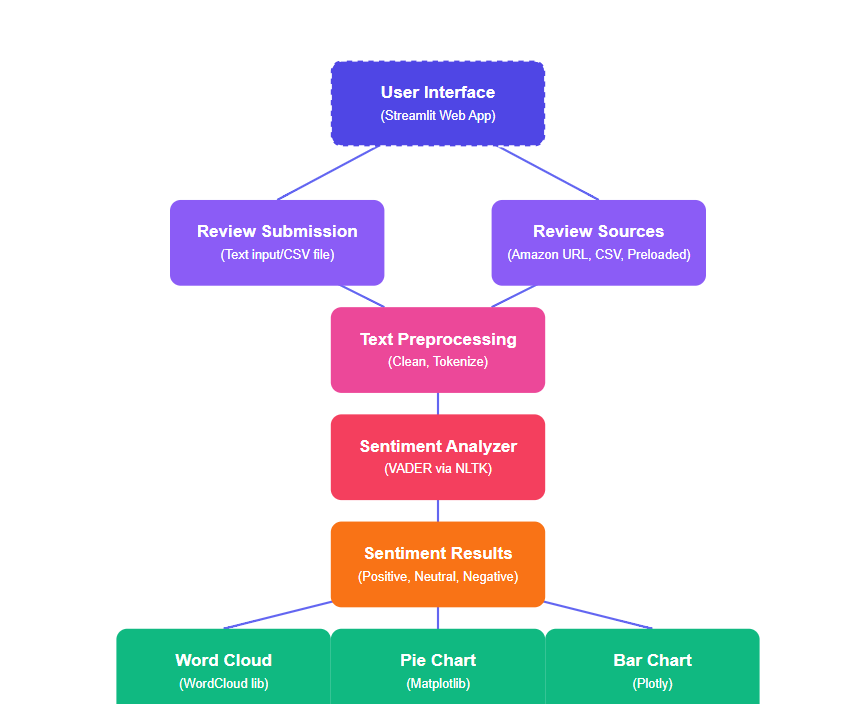
**1.4 Scope of the System**

* The system helps **e-commerce shoppers** analyze product reviews and make informed purchasing decisions.
* It allows **product sellers and marketers** to evaluate customer feedback and improve their products and services.
* The application serves as a **practical tool for data science learners and researchers** working on sentiment analysis projects.
* It provides **real-time sentiment analysis** by scraping live reviews from Amazon product pages

**1.5 Structure of the System**

The system comprises modules for:

* Amazon Review Scraping
* Sentiment Analysis using VADER
* Data Visualization (Word Clouds, Pie Charts, Bar Charts)
* User Review Input and Feedback
* Interactive Web Interface

**1.6 System Architecture**

**1.7 End Users**

The primary end users of the *Sentiment Analysis of Product Reviews* system include e-commerce shoppers, sellers, and data science enthusiasts. E-commerce shoppers can utilize the system to analyze customer reviews and sentiments before making purchasing decisions, ensuring a better and more informed shopping experience.

* E-commerce shoppers
* Product sellers and marketers
* Data science learners
* General public interested in product feedback analysis

**1.8 Software/Hardware required for the development**

**Software Requirements**

* Python
* Streamlit
* CSS
* Pandas, NumPy
* NLTK (VADER Sentiment Analysis)
* BeautifulSoup, Requests
* Matplotlib, WordCloud

**Hardware Requirements**

* Intel Core i3/i5 or AMD equivalent
* Minimum 4GB RAM
* Minimum 500MB storage
  1. **Software/Hardware required for the implementation**

**Software:**

* Python 3.x and required libraries
* Streamlit environment setup
* Visual Studio Code or Anaconda for development

**Hardware:**

* Minimum 4GB RAM system
* Internet connectivity for review scraping and application access

**2. SRS**

**2.1 Introduction (Brief write-up about SRS)**

This Software Requirements Specification (SRS) document outlines the functional and non-functional requirements for the project titled **Amazon Product Review Sentiment Analysis Web App**. The system uses Natural Language Processing (NLP) techniques, specifically **VADER** Sentiment Analysis, to classify product reviews as Positive, Negative, or Neutral.

The application enables users to either write their own reviews, upload CSV files containing reviews, or scrape real-time reviews from Amazon product URLs. It provides sentiment insights through interactive visualizations like word clouds, pie charts, and bar charts.

This project follows the Waterfall Model methodology, a linear and sequential development approach where each phase is completed before moving to the next. The methodology was chosen for its structured process and suitability for academic projects with well-defined requirements. This SRS acts as a reference document throughout the software development lifecycle, ensuring that all project requirements are clear, documented, and agreed upon.

* 1. **Overall Description**

1. **Product Perspective**

The product is a standalone, web-based application built using Streamlit framework and Python libraries. It integrates with Amazon product review pages via web scraping (BeautifulSoup) and processes data locally for sentiment analysis**.**

1. **Product Functions**

* Accept user-written product reviews and classify their sentiment.
* Fetch and analyze reviews from Amazon product URLs.
* Upload CSV files containing reviews for bulk analysis.
* Visualize sentiment distributions using pie charts, bar charts, and word clouds.
* Display product images, details, and star ratings.

1. **User Characteristics**

* Target audience: Online shoppers, small business owners, and students interested in product review sentiment analysis.
* User roles: Single-user access with the ability to analyze personal reviews, scrape reviews, upload CSV files, and view visualizations.
* User expectations: Easy-to-use interface, accurate sentiment results, quick processing, and clear visual representation of sentiment data.

1. **General Constraints**

* **Technical constraints:** The application has certain technical constraints such as requiring a stable internet connection for performing live review scraping and downloading necessary Python libraries. Additionally, it relies on the existing structure of Amazon’s review pages, and any changes to the website’s HTML layout may disrupt the scraping functionality.
* **Business constraints:** From a business perspective, the application is strictly designed for academic and educational purposes only. It is also limited by Amazon’s anti-scraping policies and fair usage restrictions, which may prevent continuous or large-scale data extraction from their platform.

1. **Assumptions**

* Amazon review pages maintain a consistent HTML structure.
* Uploaded CSV files have predefined columns like review and rating.
* Users will input valid product URLs or appropriate text reviews.

**2.3 Functional requirement**

1. **Module 1:** Application Overview & Navigation

Objective: Introduce the app and provide seamless navigation.  
Functions:

* + - Display homepage with app introduction.
    - Sidebar navigation to access different modules.

1. **Module 2:** User Review Submission & Instant Feedback

Objective: Allow users to type a review and receive immediate sentiment feedback.  
Functions:

* Text input field for reviews.
* Sentiment classification using VADER.
* Display of sentiment and compound score.

1. **Module 3:** Product Review Lookup & Sentiment Breakdown

Objective: Analyze existing product reviews from a preloaded dataset.  
Functions:

* Category and product selection.
* Display of product details and images.
* Visualize review sentiment breakdown.
* Review-wise sentiment classification and star ratings.

1. **Module 4:** Live Review Scraping & Sentiment Analysis

Objective: Scrape Amazon product reviews in real-time and classify sentiment.  
Functions:

* Accept product URL input.
* Scrape reviews and ratings using BeautifulSoup.
* Sentiment classification and visualization.

1. **Module 5:** CSV Review Import & Sentiment Insights

Objective: Analyze product reviews from a CSV file.  
Functions:

* CSV upload with review and rating columns.
* Sentiment analysis and visualization of uploaded data.

**2.4 Design Constraints**

* Streamlit’s UI limitations restrict complex user interface design.
* Amazon’s review scraping restrictions can limit data availability.
* System performance limitations for large review datasets.
* Internet dependency for accessing live product pages and external packages.

**2.5 System Attributes**

* **Reliability:** Provides accurate sentiment classification using VADER.
* **Availability:** Available 24/7 on supported local or cloud environments.
* **Maintainability:** Code modularity makes it easy to update scraping logic or improve UI.
* **Portability:** Compatible across Windows, Linux, and macOS systems.
* **User-friendliness:** Intuitive navigation and visually appealing interface.
* **Performance:** Efficient processing for moderate-sized datasets; real-time scraping might vary based on internet speed and product review count.

**3. System Design**

**3.1 Introduction (brief write-up about System Design)**

System Design is a crucial phase in the software development life cycle where the overall architecture, data flow, and module structure of the system are planned based on the requirements collected during the analysis phase. It involves defining how various components of the system will interact, the processing of data, and the structure of each module. This ensures that the system’s development proceeds in an organized, efficient, and logical manner.

The proposed system consists of the following key modules:

* **Module 1: Application Overview & Navigation**  
  Introduces the purpose of the application and provides easy navigation through a sidebar menu.
* **Module 2: User Review Submission & Sentiment Feedback**  
  Allows users to write their own product reviews and receive immediate sentiment classification using the VADER Sentiment Analysis technique.
* **Module 3: Product Review Lookup & Sentiment Breakdown**  
  Enables users to select a product from a dataset, view its details, and analyze stored customer reviews with visual sentiment breakdowns.
* **Module 4: Live Review Scraping & Sentiment Analysis**  
  Accepts an Amazon product URL, scrapes real-time customer reviews from the web, and performs sentiment analysis on the fetched reviews.
* **Module 5: CSV Review Import & Sentiment Insights**  
  Allows users to upload a CSV file containing customer reviews, processes the data, performs sentiment analysis, and presents results through visualizations like word clouds, pie charts, and bar charts.
  1. **Assumptions and Constraints**

**Assumptions**

* + - * The Amazon website structure remains consistent during the scraping process.
      * Users provide valid product URLs, properly structured CSV files, and meaningful review text inputs.

**Constraints**

* + - * Requires a reliable internet connection for scraping and data processing.
      * Scraping and review analysis limited by Amazon’s website policies and review page structure.
  1. **Functional Decomposition**

The system is divided into five functional modules, each responsible for a specific set of tasks:

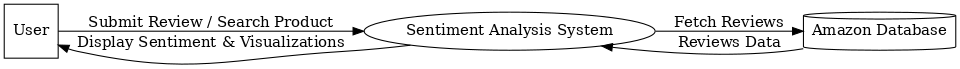
1. Application Overview & Navigation — For displaying app features and navigation controls.
2. User Review Submission — For submitting personal reviews and obtaining instant sentiment feedback.
3. Product Review Lookup — For selecting products from a dataset and analyzing preloaded reviews.
4. Live Review Scraping — For fetching and analyzing live Amazon reviews based on product URLs.
5. CSV Review Import — For uploading and analyzing a CSV file of customer reviews.
   1. **Description of Programs**

**i. Context Flow Diagram (CFD)**

The Context Flow Diagram provides a high-level view of the system and its interactions with external entities. It shows the system as a single process with its inputs and outputs.

**Key Features:**

* Represents the entire system as one process.
* Shows external entities (users, other systems) interacting with the system.
* Displays the data flow between the system and external entities.
* Helps understand system boundaries and high-level information exchange.

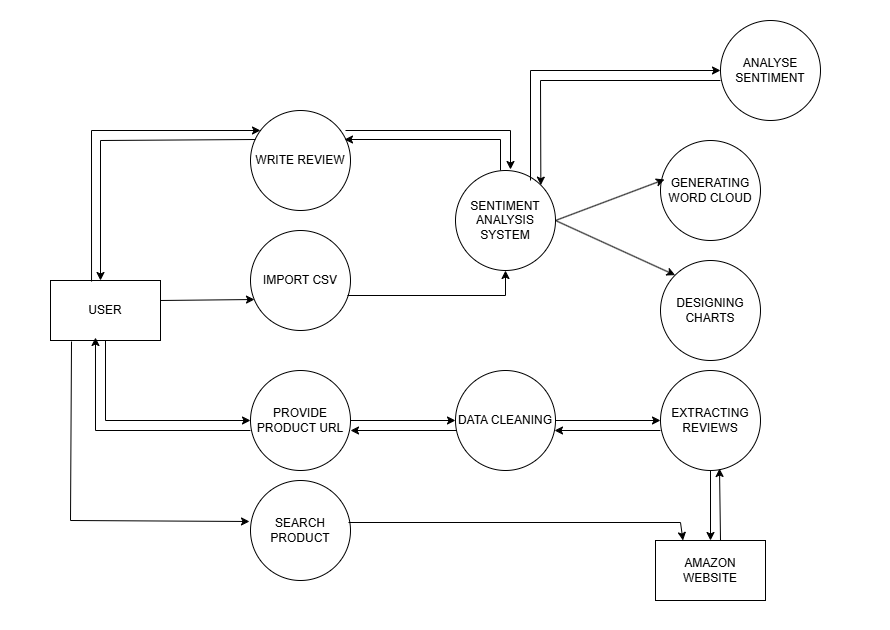
****

**ii. Data Flow Diagrams (DFD)**

A data flow diagram is a way of representing a flow of data of a process or system. It includes data inputs and outputs, data stores, and the various sub processes the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

**Key Features:**

* Uses symbols like processes, data stores, data flows, and external entities.
* Can be created at different levels (Level 0, Level 1, Level 2, etc.) for detailed system analysis.
* Highlights how input is transformed into output through different processes.
* Helps in analyzing, designing, and documenting a system’s data processing.



**iii. Use Case Diagram**

A use case diagram is a dynamic or behaviour diagram in UML. Use case diagrams model the functionality of a system using actors and use cases.

**Use Case Diagram Notations**

#### Actors

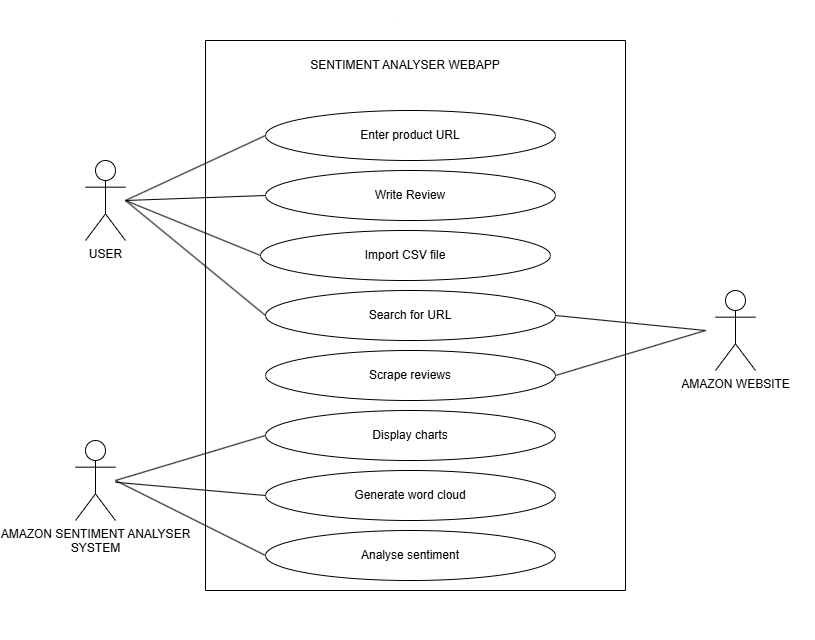
Actors are the entities that interact with a system. Although in most cases, actors are used to represent the users of system, actors can actually be anything that needs to exchange information with the system. So, an actor may be people, computer hardware, other systems, etc.Actors are represented by the following symbol,

#### Use Case

Use cases are a set of actions, services, and functions that the system needs to perform. Use Cases are represented by oval.

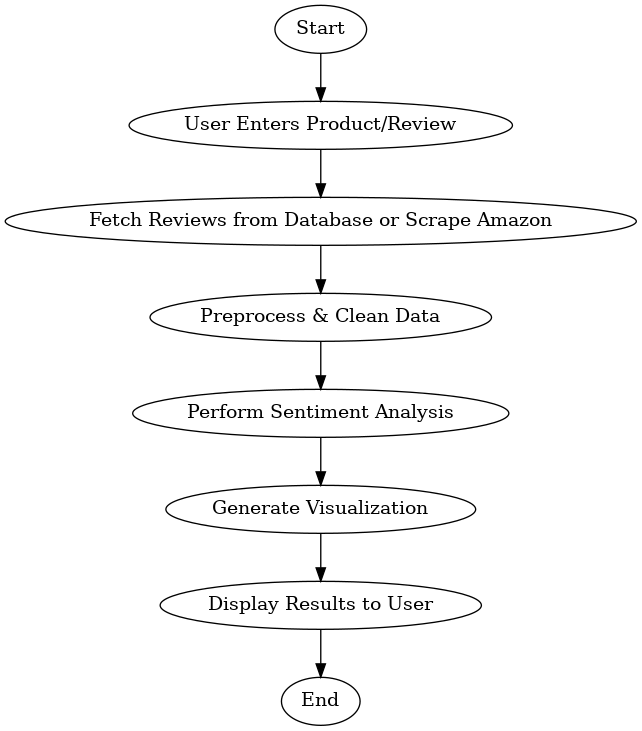
#### Relationships

It illustrates relationships between an actor and a use case. Relationships are simply illustrated with a line connecting actors to use cases.

****

**iv. Activity Diagram**

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modelling. They can also describe the steps in a use case diagram. Activities modelled can be sequential and concurrent. In both cases an activity diagram will have a beginning (an initial state) and an end (a final state).

****

**3.5 Description of components**

1. **Functional Component 1:** Sentiment Analysis Module

* Analyzes customer reviews using VADER Sentiment Analysis.
* Categorizes reviews as Positive, Negative, or Neutral based on compound scores.
* Displays sentiment results with visualizations like pie charts, bar charts, and word clouds.

1. **Functional Component 2**: Review Scraping and Data Handling Module

* Accepts product URLs and scrapes reviews using BeautifulSoup and Requests.
* Supports CSV file uploads and reads data using Pandas.
* Cleans and processes review data before analysis.
* Integrates data fetching, cleaning, and displaying review sentiments in a user-friendly format.

**4.Database &**

**Datasets Used**

#### 4.1 Introduction (brief write-up about Database design)

#### 4.2 Purpose and Scope

**Purpose**

**Scope**

**5. User Requirement**

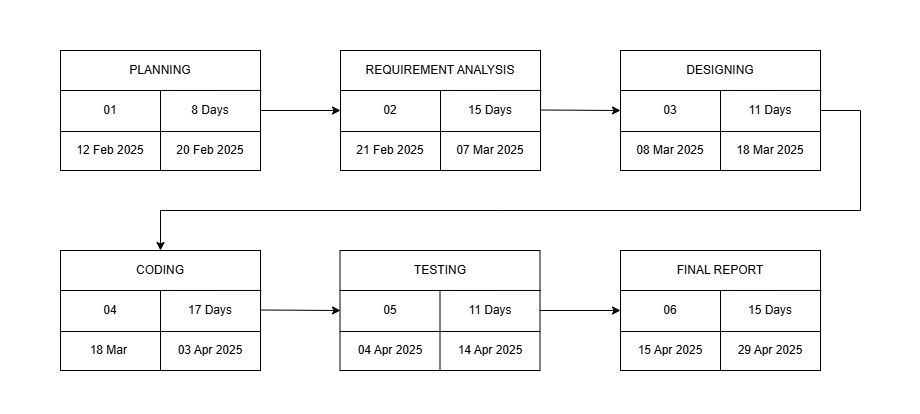
**System Planning**

**& Scheduling**

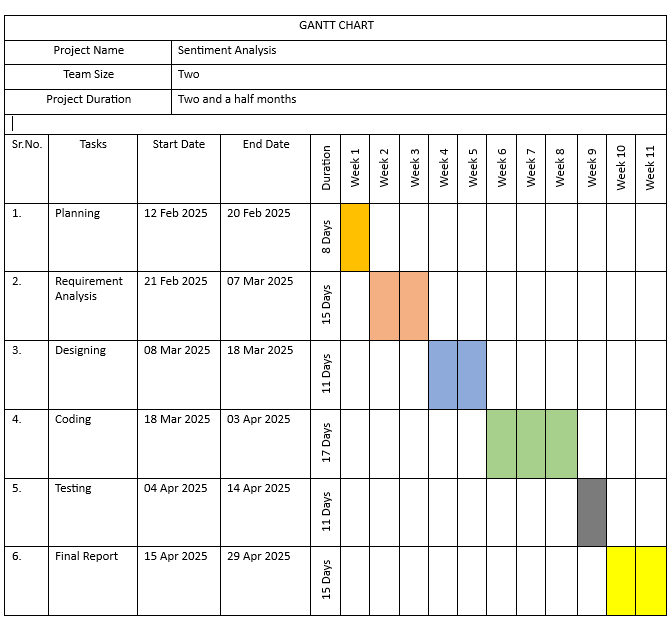
**5.1 PERT Chart**

* PERT Chart is acronym for “Program Evaluation and Review Technique”.
* A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project. It is basically a method to analyse the tasks involved in completing a given project, especially the time needed to complete each task, and to identify the minimum time needed to complete the total project.
* The main objective of PERT is to facilitate decision making and to reduce both the time and cost required to complete a project. PERT is intended for one-time, nonroutine, complex projects with a high degree of inter-task dependency, projects which require a series of activities, some of which must be performed sequentially and others that can be performed in parallel with other activities.

#### PERT Chart Notations

* Task Name
* Task ID
* ****Duration  Start and End Date

### 5.2 GANTT Chart

* A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar, the position and length of the bar reflects the start date, duration and end date of the activity.
* This allows you to see at a glance:
  1. What the various activities are
  2. When each activity begins and ends
  3. How long each activity is scheduled to last
  4. Where activities overlap with other activities, and by how much
  5. ****The start and end date of the whole project

**6. Program**

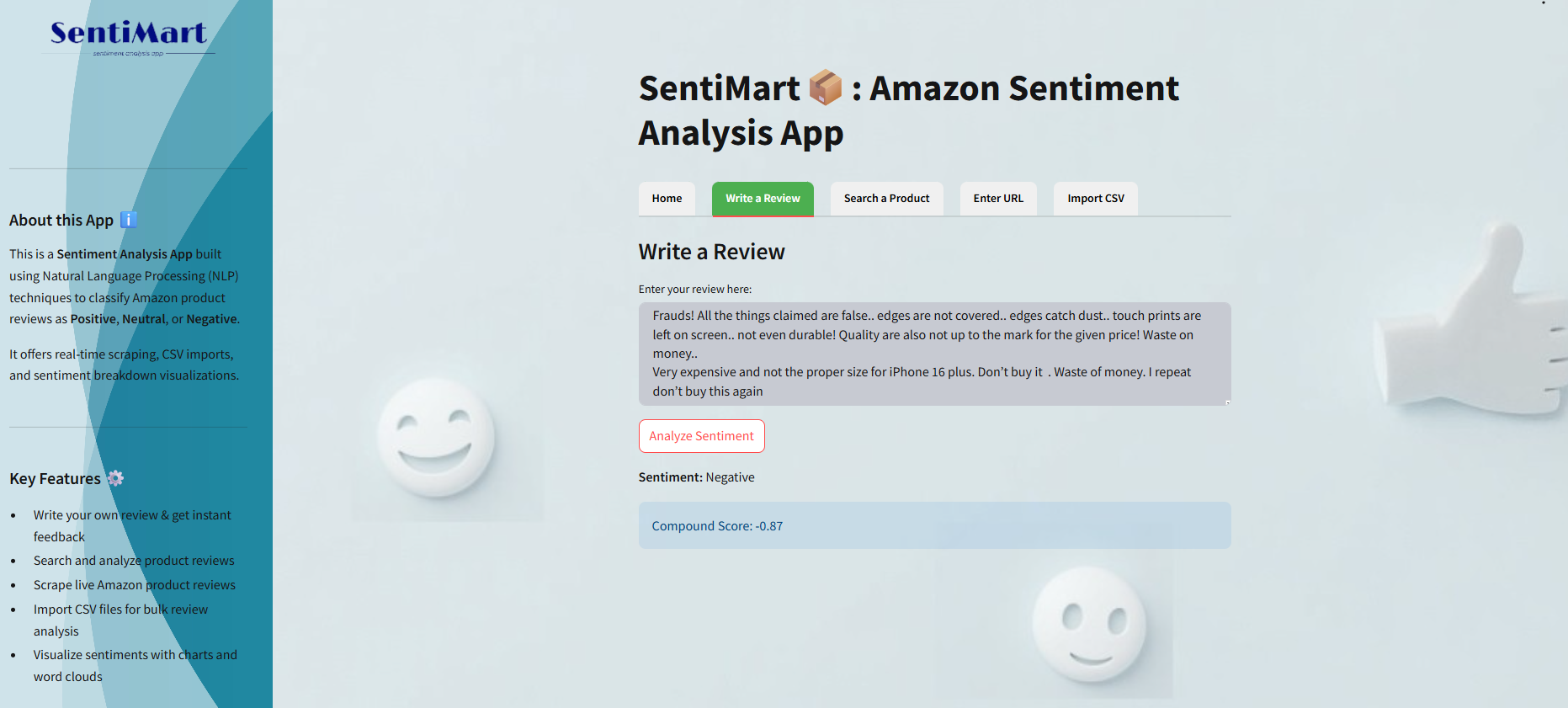
**Code Listing**

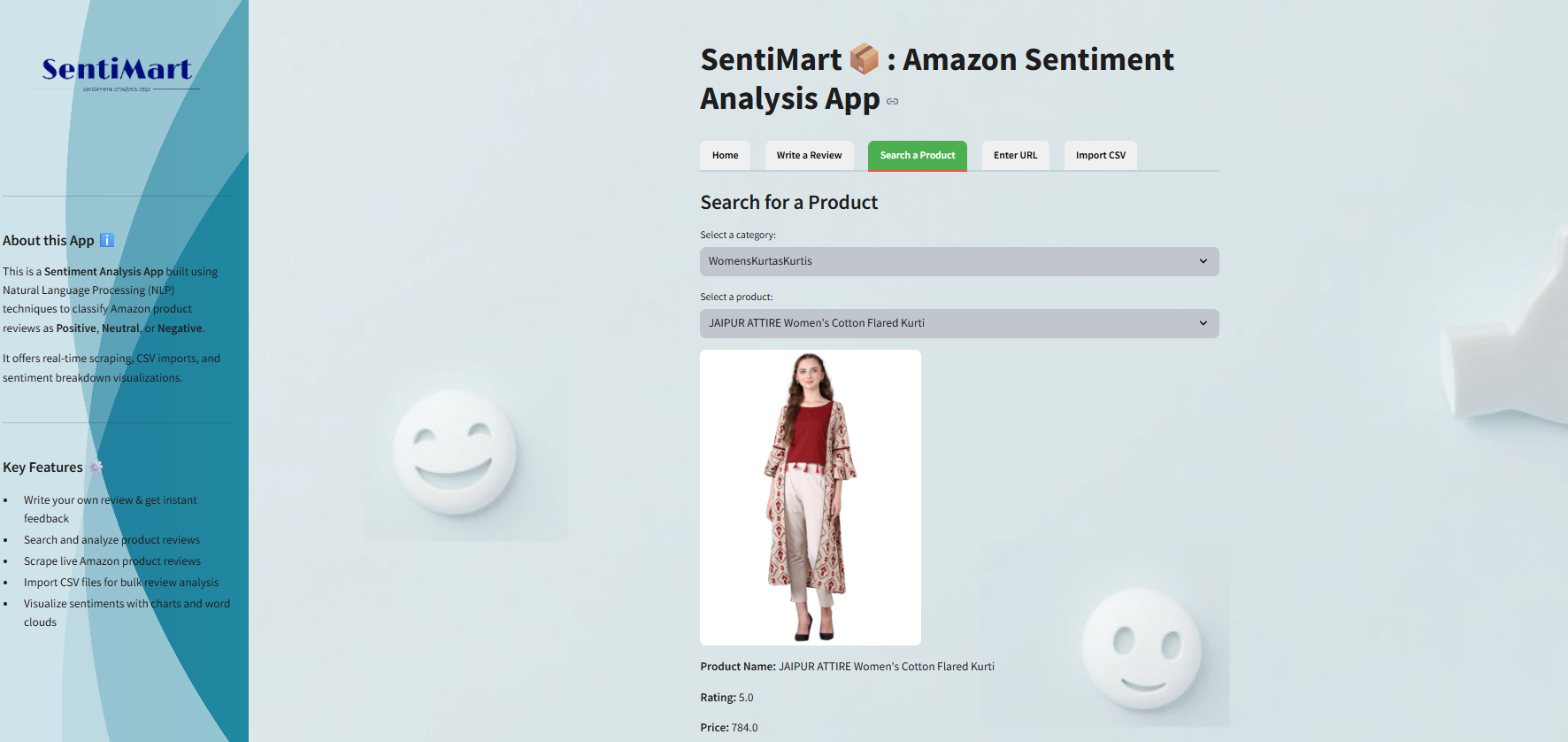
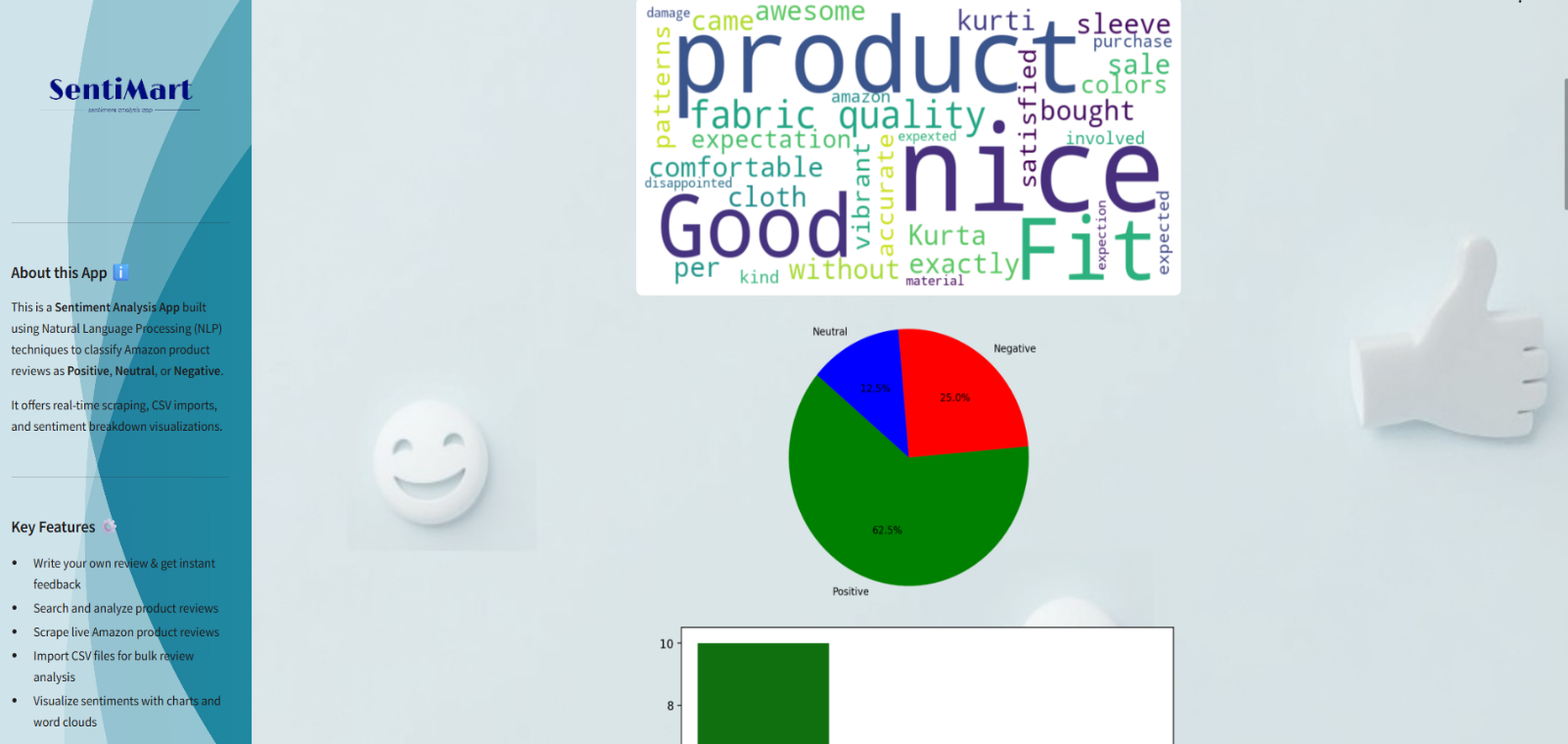
**7. USER**

**INTEERFACE**

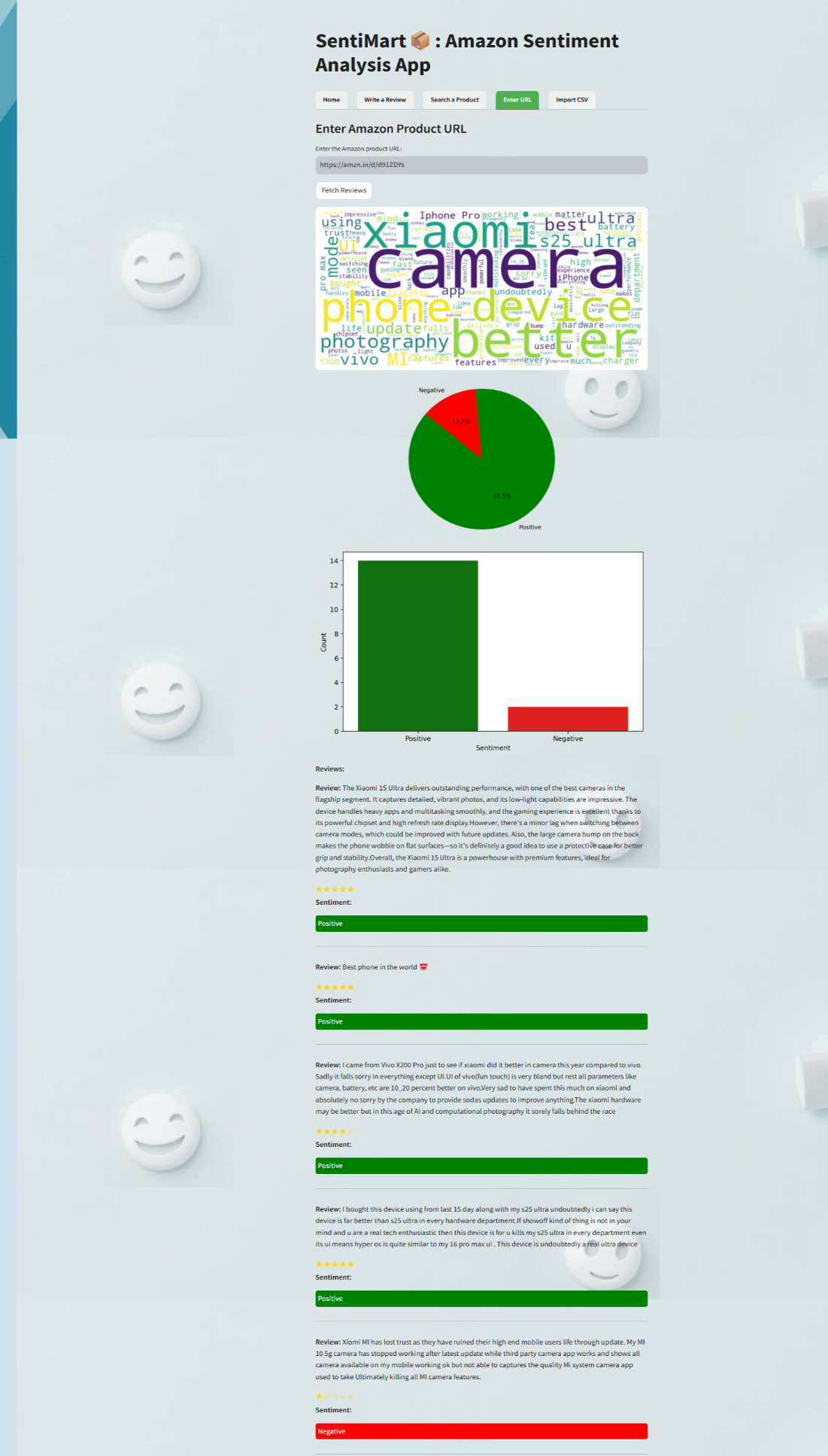
**(SCREEN & REPORTS)**

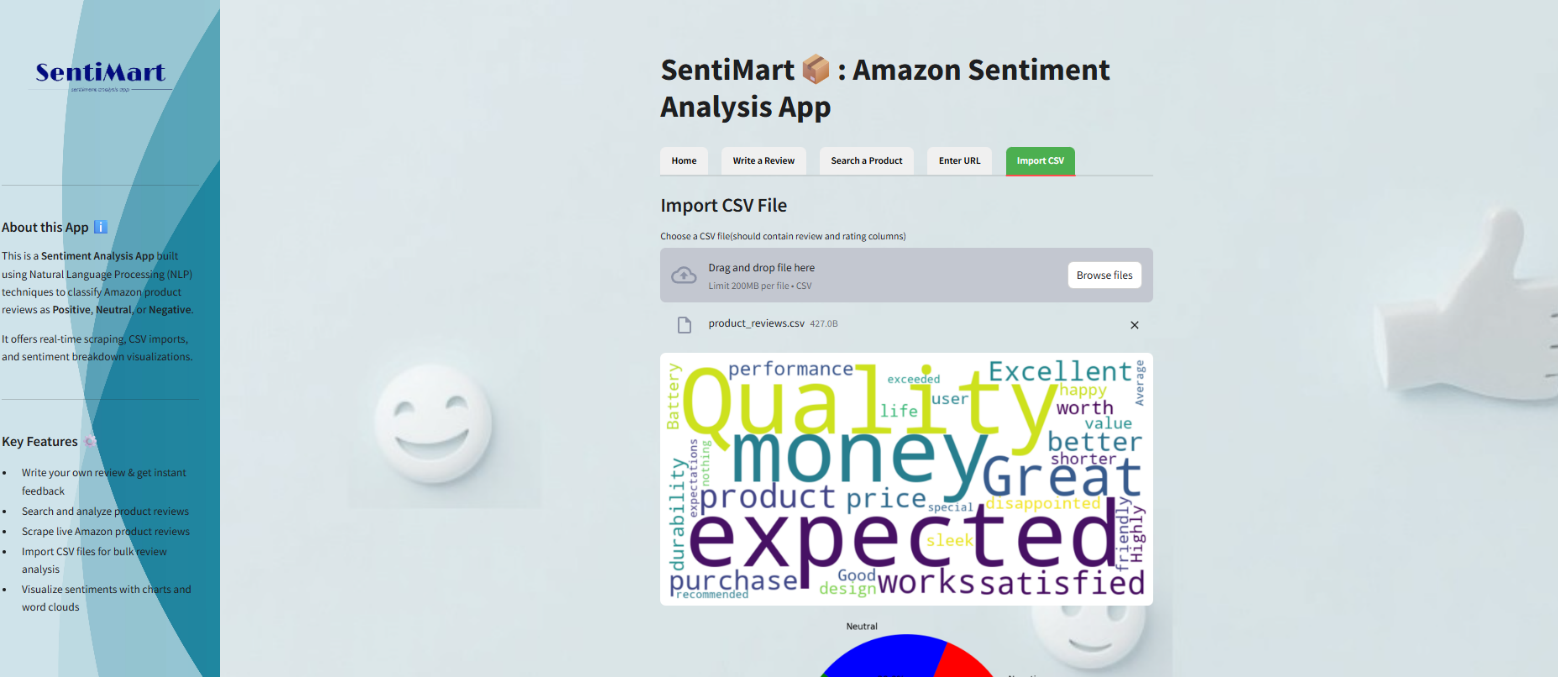
****

****

****

****

****

****

**8.Testing**

**8.1 Introduction (brief write-up about Software Testing)**

### Brief Write-up about Software Testing

* + 1. **Unit testing**

# Integration Testing

* + 1. **System Testing**

**8.2 Test Report**

**9.Conclusion and future scope**

1. **Conclusion**

Opinion mining/Sentiment analysis means piece of text is computed, categorized and identified whether writer/user attitude towards product. It helps to detect spam reviews. We made our own dictionary having sentiment words along with weight. This method is proposed to calculate sentiment score and reviews from neutral language text. This web application will be capable of understanding the deviation of positive and negative in review and rate accordingly.

**Future Enhancement/Scope**

* Multilingual Sentiment Analysis

Enable analysis of reviews in multiple languages using NLP models like BERT or multilingual transformers.

* Voice Search Integration

Add a voice-based input system for product search to enhance accessibility.

* Mobile App Version

Extend functionality with a mobile-friendly version or native app for on-the-go access.

* Marketplace Expansion

Extend support beyond Amazon to other e-commerce platforms like Flipkart, eBay, or Walmart.

* Real-Time Review Monitoring

Implement scheduled scraping and alerting for newly posted reviews to provide up-to-date sentiment trends.

* Emotion Detection

Go beyond polarity (positive/neutral/negative) to detect emotions like joy, anger, trust, surprise, etc.

**10.References**

### VADER Sentiment Analysis Tool Hutto, C.J., & Gilbert, E.E. (2014). *VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text*. [*https://github.com/cjhutto/vaderSentiment*](https://github.com/cjhutto/vaderSentiment)

**Matplotlib** & **Seaborn** – Visualization tools  
[*https://matplotlib.org/*](https://matplotlib.org/)  
[*https://seaborn.pydata.org/*](https://seaborn.pydata.org/)

**Draw.io / diagrams.net** – Used to create system diagrams  
[*https://www.diagrams.net/*](https://www.diagrams.net/)

### Youtube Links:

1.Python:

https://youtu.be/UrsmFxEIp5k?si=JbzBzQkVJATfrfYR

2.Streamlit:

<https://youtu.be/yKTEC1Y5bEQ?si=56oSNLGMcA7js4FW>

3.Web Scraping :

https://youtu.be/8NOdgjC1988?si=R6OHa31-r4Vh7r3d

BOOKS:

*1.Internet and Web Technologies*, Tata McGraw Hill Education

Covers basics of web technologies including web scraping foundations and data extraction.

Author: Raj Kamal

2.Introduction to Machine Learning with Python: A Guide for Data Scientists

Published By: O’Reilly Media

Author : Andres C. Muller and Sarah Guido