Project Report
on

YouTube Comment Analyzer Using NLP

BTech-IT, Sem VI

Prepared By:
Rudra Patel (IT-106)
Rudrakumar Patel (IT-107)

Guided By:
Prof. Sunil Vithlani
Dept. of Information Technology

Department of Information Technology
Faculty of technology,
Dharmsinh Desai University
College road, Nadiad- 387001

April, 2024

CANDIDATE'S DECLARATION

We declare that 6th semester report entitled **"YouTube Comment Analyzer Using NLP"** is our own work conducted under the supervision of the guide **Prof. Sunil Vithlani**.

We further declare that to the best of our knowledge the report for B.Tech. VI semester does not contain part of the work which has been submitted either in this or any other university without proper citation.

Candidate's Signature
Rudra Patel
Student ID: 21ITUOS026

Candidate's Signature
Rudrakumar Patel
Student ID: 21ITUOS011

DHARMSINH DESAI UNIVERSITY
NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project carried out in the subject of Project-I, entitled "YouTube Comment Analyzer Using NLP" and recorded in this report is a bonafide report of work of

1) **Rudra Patel Roll No.IT106 ID No: 21ITUOS026**
2) Rudrakumar Patel Roll No.IT107 ID No: 21ITUOS011

of Department of Information Technology, semester VI. He/She/They was/were involved in Project work during academic year 2023 -2024.

Prof. Sunil Vithlani
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

Prof. (Dr.) V. K. Dabhi
Head, Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

ACKNOWLEDGEMENT

It is indeed a great pleasure to express our thanks and gratitude to all those who helped us during this project. This project has given us a great opportunity to think, implement and interact with various aspects of the Software Development Life Cycle. We would like to acknowledge all the people who have helped us at one stage or another by providing the much-needed support, encouragement, and groundwork to complete our project.

We express a deep sense of gratitude towards our project guide Prof. Sunil Vithlani towards his innovative ideas and earnest effort to make our project a success. It is his sincerity that prompted us throughout the project to do hard work using industry-adopted technologies. Our commitment to the application is the sole result of patience, hard work, and dedication being inspired by him.

A blend gratitude, pleasure, and great satisfaction are what we feel to convey our indebtedness to all those who have directly or indirectly contributed towards the completion of the project.

1. Rudra patel
2. Rudrakumar Patel

DETAILS OF CHAPTERS

1. Introduction

1.1 Project Details: Broad specifications of the work entrusted to you.

The project entails the development of a YouTube comment analyzer, which allows users to input YouTube video links and retrieve comments from those videos. These comments are then subjected to sentiment analysis using machine learning techniques. The aim is to classify comments into positive, negative, or neutral sentiments based on their content. This involves integrating with the YouTube API to fetch comments, preprocessing the text data, training a machine learning model (specifically SVM) on a dataset of labeled comments, and providing a user-friendly interface for users to interact with the system.

1.2 Purpose

The purpose of this project is to offer a tool for analyzing sentiments expressed in YouTube comments. This tool can be beneficial for content creators, marketers, and researchers who seek to understand audience feedback better. By automatically categorizing comments into positive, negative, or neutral sentiments, users can gain insights into the overall sentiment distribution of comments on a particular video.

1.3 Scope

- The scope of the project includes the following:
- Retrieval of comments from a specified YouTube video using the YouTube API.
- Preprocessing of comments to remove noise, such as stop words and punctuation.
- Training a machine learning model, specifically a Support Vector Machine (SVM), on a labeled dataset of comments to perform sentiment analysis.
- Classification of comments into positive, negative, or neutral sentiments based on the trained model.

- Development of a web-based user interface where users can input YouTube video links, view retrieved comments, and see sentiment analysis results.
- Limitations include:
    - The system may not be able to accurately analyze comments in languages other than the language used for training the model.
    - The accuracy of sentiment analysis may be affected by the quality and diversity of the training data.

1.4 Objective (Scope – what it can do and can't do)

The main objective of the project is to accurately classify YouTube comments based on sentiment. Specifically, the system can:

- Retrieve comments from a specified YouTube video.
- Preprocess comments to prepare them for sentiment analysis.
- Classify comments into positive, negative, or neutral sentiments using a pre-trained SVM model.
- Display sentiment analysis results to the user via a web-based interface.

However, the system cannot:

- Guarantee 100% accuracy in sentiment analysis due to the inherent subjectivity and complexity of human language.
- Analyze comments in languages other than the language(s) used for training the model.
- Provide real-time analysis of comments as they are posted on YouTube.

1.5 Technology and Literature Review

The project utilizes various technologies and incorporates insights from relevant literature:

- MERN stack (MongoDB, Express.js, React.js, Node.js) for web development.
- Python programming language for backend processing and machine learning tasks.
- NLP techniques, including tokenization, stop word removal, and TF-IDF vectorization, for preprocessing text data.
- Support Vector Machine (SVM) algorithm for sentiment analysis.
- YouTube API for retrieving comments from YouTube videos.
- Literature review from sources such as Kaggle datasets, academic papers, and documentation for relevant technologies and APIs to inform the design and implementation of the project.

2. Project Management

2.1 Feasibility Study

2.1.1 Technical Feasibility

After evaluating the project requirements, we have assessed the technical feasibility of implementing the YouTube comment analyzer using the MERN (MongoDB, Express.js, React.js, Node.js) stack for website development and Python for model training. The chosen technologies and tools are:

- MongoDB for database storage
- Express.js for backend development
- React.js for frontend development
- Node.js for server-side scripting

These technologies are well-suited for building scalable and responsive web applications. Additionally, Python provides robust libraries and tools for machine learning model training, making it suitable for sentiment analysis on YouTube comments.

2.1.2 Time Schedule Feasibility

Our project timeline is structured as follows:

- December 2023: Initial planning, research, and dataset exploration
- January 2024: Backend and frontend development using the MERN stack
- February 2024: Model training and integration with the YouTube API
- March 2024: Testing, debugging, and refinement
- April 2024: Deployment and user acceptance testing

This timeline allows adequate time for each phase of development, ensuring timely completion of the project within the specified timeframe.

2.1.3 Operational Feasibility

The operational feasibility of the YouTube comment analyzer is high, as the system's intuitive design and user-friendly features ensure smooth navigation and interaction for users. The integration with the YouTube API enables seamless retrieval

of comments, enhancing operational efficiency. Additionally, the MERN stack's flexibility allows for easy scalability and maintenance of the system.

2.1.4 Implementation Feasibility

The implementation feasibility of the YouTube comment analyzer is supported by the availability of resources and expertise in the chosen technologies. Leveraging Python for model training and the MERN stack for website development, we can build a robust and efficient system. The MERN stack's modular architecture and extensive community support facilitate rapid development and deployment of web applications. Furthermore, Python's rich ecosystem of machine learning libraries enables accurate sentiment analysis of YouTube comments.

2.2 Project Planning

2.2.1 Project Development Approach and Justification

We have chosen an Agile development approach for the YouTube comment analyzer project. Agile methodology emphasizes iterative development and close collaboration between development teams and stakeholders. This approach allows for flexibility in responding to changing requirements and ensures continuous improvement throughout the project lifecycle. By adopting Agile practices, we aim to deliver a high-quality product that meets user needs and expectations.

2.2.2 Project Plan

Our project plan includes the following key milestones:

1. Project initiation and requirement analysis
2. Backend and frontend development using the MERN stack
3. Model training and integration with the YouTube API
4. Testing and quality assurance
5. Deployment and user acceptance testing

We will adhere to this plan to ensure efficient project execution and timely delivery of the YouTube comment analyzer system.

2.2.3 Roles and Responsibilities

Name	Analysis	Design	Coding	Testing	Documentation	Maintenance
Rudra Patel	✓	✓	✓	✓	✓	✓
Rudrakumar Patel	✓	✓	✓	✓	✓	✓

This distribution of roles ensures that each team member contributes effectively to all aspects of the project, from initial analysis and design to ongoing maintenance and support.

3. System Requirements Study

3.1 Study of Current System

Given the nature of our project, there isn't a direct comparison to an existing system. However, we can draw parallels with sentiment analysis tools and comment monitoring platforms. These systems often lack integration with YouTube's API for real-time comment analysis, which is a unique feature of our project.

3.2 Problems and Weaknesses of Current System

The primary issue with existing systems is the inability to efficiently analyze sentiments of YouTube comments in real-time. Furthermore, many sentiment analysis tools lack specificity for YouTube content, resulting in less accurate analysis. Our system addresses these weaknesses by leveraging machine learning models trained specifically for YouTube comments.

3.3 User Characteristics

- **The user base for our system includes**:

1. **YouTube Content Creators**: Utilize the platform to analyse comments on their videos, gaining insights into audience sentiment.
2. **General Users**: Individuals interested in gauging public opinion on YouTube videos.
3. **Researchers**: Analyse trends and patterns in YouTube comments for academic or market research purposes.

3.4 Hardware and Software Requirements

The minimum hardware and software requirements for running our system are as follows:

- **Hardware**: Standard computer or laptop with internet connectivity.
- **Software**: Web browser (Google Chrome, Mozilla Firefox, etc.).

3.5 Constraints

Several constraints need consideration:

- **Regulatory Policies**: Compliance with YouTube's terms of service and data protection regulations.
- **Hardware Limitations**: System performance may vary based on the user's device specifications.
- **Interfaces to Other Applications**: Integration with YouTube's API for comment retrieval and analysis.
- **Parallel Operations**: The system should handle multiple users accessing it simultaneously.
- **Higher Order Language Requirements**: Utilization of Python for machine learning model training and JavaScript for web development.
- **Reliability Requirements**: Ensuring the system is reliable and available for users when needed.
- **Criticality of the Application**: The importance of accurate sentiment analysis for users relying on the system's insights.
- **Safety and Security Consideration**: Implementing measures to protect user data and ensure secure transactions.

3.6 Assumptions and Dependencies

- **Assumptions**: Users have basic knowledge of YouTube and understand the purpose of sentiment analysis.
- **Dependencies**: Reliance on YouTube's API for comment retrieval and the availability of training data for machine learning model development.

4. System Analysis

4.1 Requirements of New System (SRS)

The Software Requirements Specification (SRS) for the YouTube comment analyser outlines the functional and non-functional requirements of the system, including:

- **Functional Requirements**:

  - Ability to accept YouTube video links from users.
  - Integration with YouTube API to retrieve comments from specified videos.
  - Implementation of sentiment analysis using machine learning models.
  - Classification of comments into positive, negative, and neutral categories.
  - User authentication and authorization for accessing the system.
  - Filtering options for users to view comments based on sentiment.
  - Visualization of comment sentiments in the frontend interface.

- **Non-Functional Requirements**:

  - **Performance**: The system should handle large volumes of comments efficiently.
  - **Reliability**: Ensuring accurate sentiment analysis results for comments.
  - **Security**: Implementing measures to protect user data and system integrity.
  - **Usability**: Providing an intuitive and user-friendly interface for seamless interaction.
  - **Scalability**: Ability to accommodate future growth in user base and data volume.

4.2 Features of New System

The key features of the YouTube comment analyser system include:

1. **YouTube Video Link Input**: Users can input URLs of YouTube videos for comment analysis.
2. **Comment Retrieval**: Integration with the YouTube API to fetch comments associated with the provided video links. **Sentiment Analysis**: Utilization of machine learning models to analyse the sentiment of each comment.

3. **Comment Labelling**: Classification of comments into positive, negative, and neutral categories based on sentiment analysis results.
4. **User Authentication**: Secure user authentication and authorization mechanisms for accessing the system.
5. **Filtering Options**: Filtering comments based on sentiment categories (positive, negative, neutral).
6. **Visualization**: Visualization of comment sentiments through graphical representations for better understanding.
7. **User Management**: Administration panel for managing user accounts and permissions.
8. **Reporting**: Generation of reports summarizing sentiment analysis results for users' reference.

4.3.1 User Requirements

The user requirements for the YouTube comment analyser system include:

- **Content Creators**: Require insights into audience sentiment for their YouTube videos.
- **General Users**: Interested in analysing sentiment trends on specific YouTube content.
- **Researchers**: Need access to sentiment analysis data for academic or market research purposes.

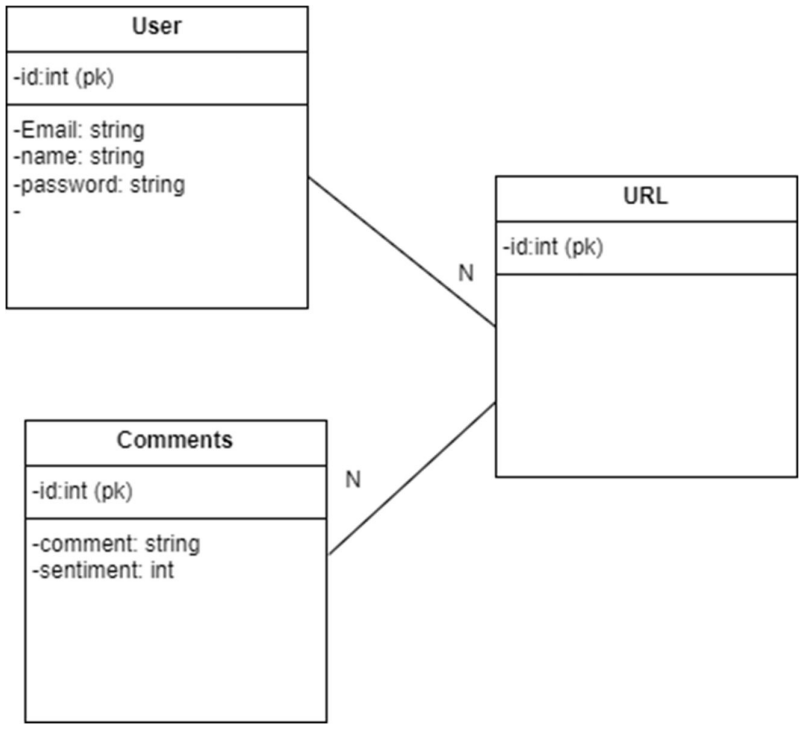
4.3.2 System Requirements

- Integration with YouTube API for comment retrieval.
- Implementation of machine learning models for sentiment analysis.
- Secure user authentication and authorization mechanisms.
- Scalable architecture to handle varying loads and user volumes.

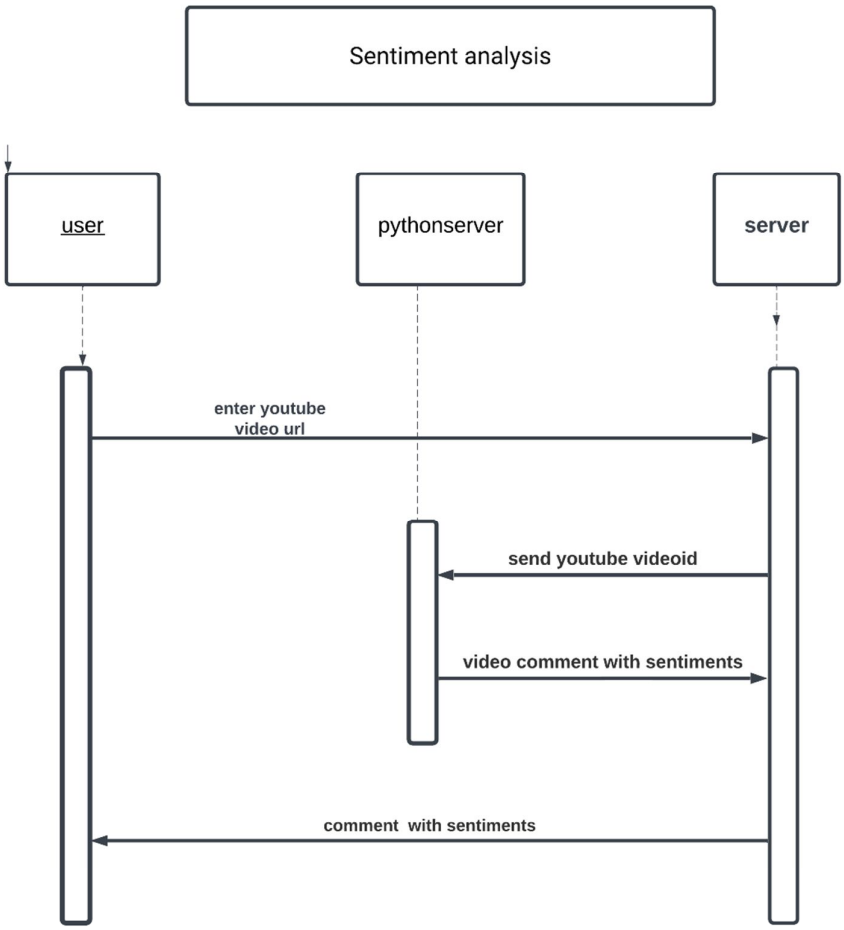
5. System Design

5.1 System Architecture Design

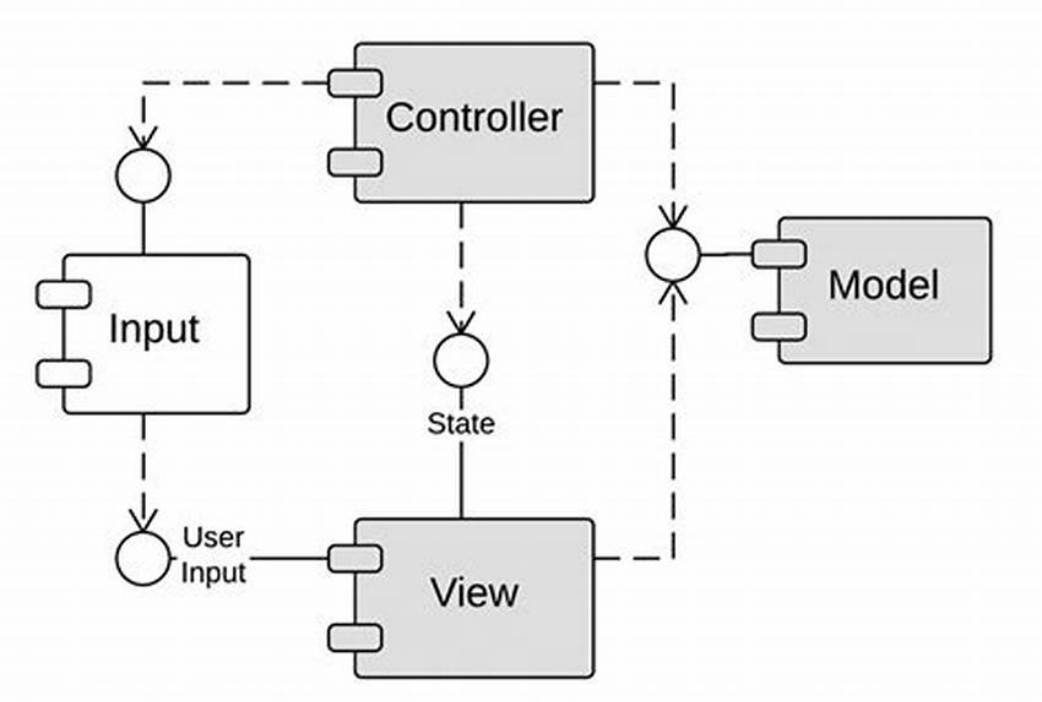
5.1.1 Class Diagram:



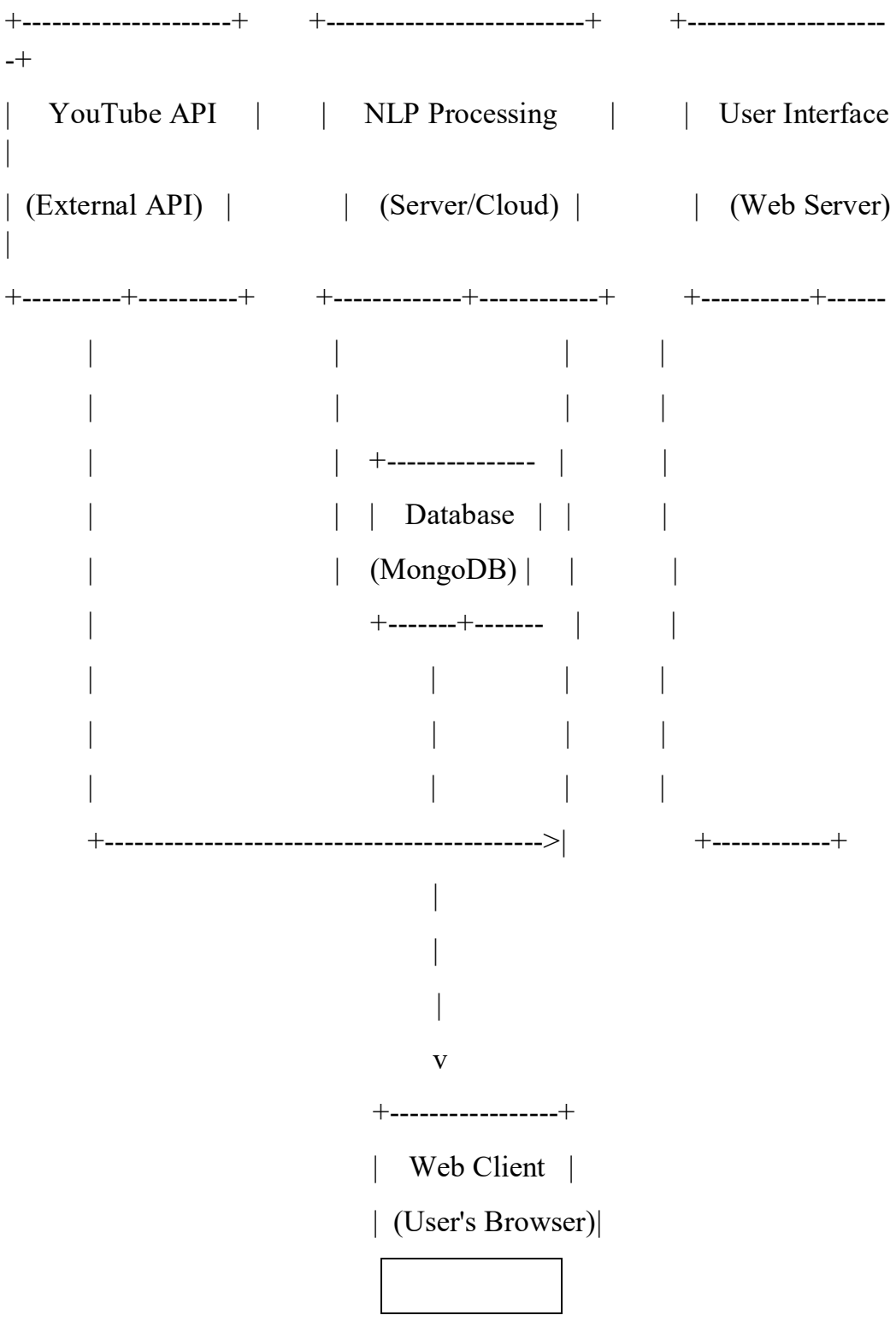
5.1.2 Sequence Diagram:



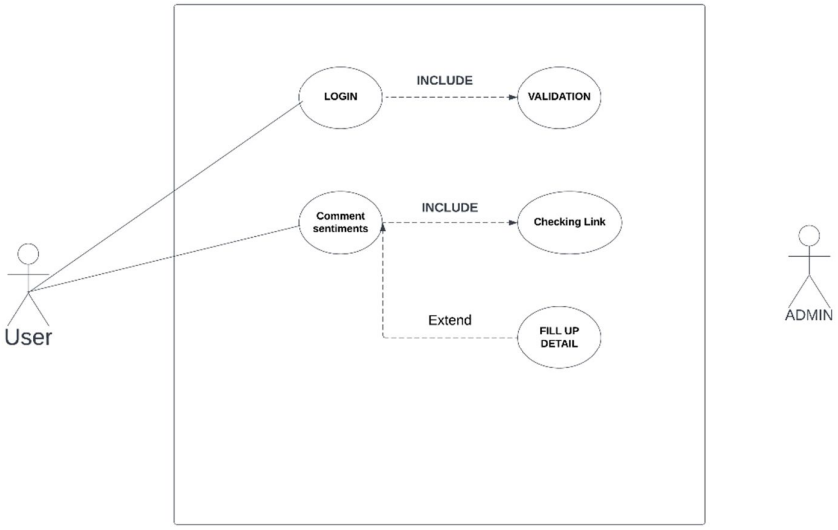
5.1.3 Component Diagram:



5.1.4 Deployment Diagram:



5.1.5 Use-case Diagram:



6. Implementation Planning

6.1 Implementation Environment (Single vs Multi-User, GUI vs non-GUI)

For the implementation of our YouTube Comment Analyzer project, we've adopted the following environment:

- **Single vs Multi-User**: The system is designed to support multiple users concurrently, allowing multiple users to access the platform simultaneously.
- **GUI vs Non-GUI**: The project features a web-based graphical user interface (GUI) for ease of interaction with users.

6.2 Program/Modules Specification

The system comprises the following key modules:

- **User Interface (UI):**
  - Provides an intuitive interface for users to input YouTube video links and view sentiment analysis results.
  - Allows users to filter and visualize comment sentiments.
- **Backend (Server-side):**
  - Developed using Node.js to handle user authentication, authorization, and server-side logic.
  - Integrates with Python scripts for machine learning model training and YouTube comment fetching.
- **Python Scripts:**
  - Utilized for training machine learning models (e.g., SVM, Random Forest) on sentiment analysis.
  - Fetch comments from YouTube using the YouTube API.
- **Database:**
  - Stores user data, including login credentials and preferences.
  - Stores analysed comments and their sentiment labels.

6.3 Coding Standards

To ensure consistency and maintainability of our codebase, we adhere to the following coding standards:

- Consistent indentation and spacing for improved readability.
- Comprehensive commenting to explain the purpose and functionality of each code segment.
- Descriptive variable and function names for clarity and understanding.
- Consistent naming conventions following best practices in both Node.js and Python.
- Adherence to PEP 8 guidelines for Python code to ensure consistency and readability.

7. Testing

7.1 Testing Plan

Our testing plan includes a comprehensive approach to ensure the functionality, reliability, and performance of the YouTube Comment Analyzer system. It outlines the scope, objectives, resources, and schedule for testing activities.

7.2 Testing Strategy

Our testing strategy encompasses various methodologies to validate the system's behaviour under different scenarios. It includes unit testing, integration testing, system testing, and acceptance testing to ensure the system meets the specified requirements and user expectations.

7.3 Testing Methods

We employ the following testing methods:

Unit Testing: Individual components such as functions and modules are tested in isolation to verify their correctness.

Integration Testing: Combined modules are tested together to ensure they work seamlessly as a whole system.

System Testing: The entire system is tested to validate its compliance with functional and non-functional requirements.

Acceptance Testing: End-to-end testing is performed to evaluate the system's compliance with user requirements and acceptance criteria.

7.4 Test Cases

7.4.1 Purpose

The purpose of test cases is to validate the functionality and behaviour of the YouTube Comment Analyzer system under different conditions.

7.4.2 Required Input

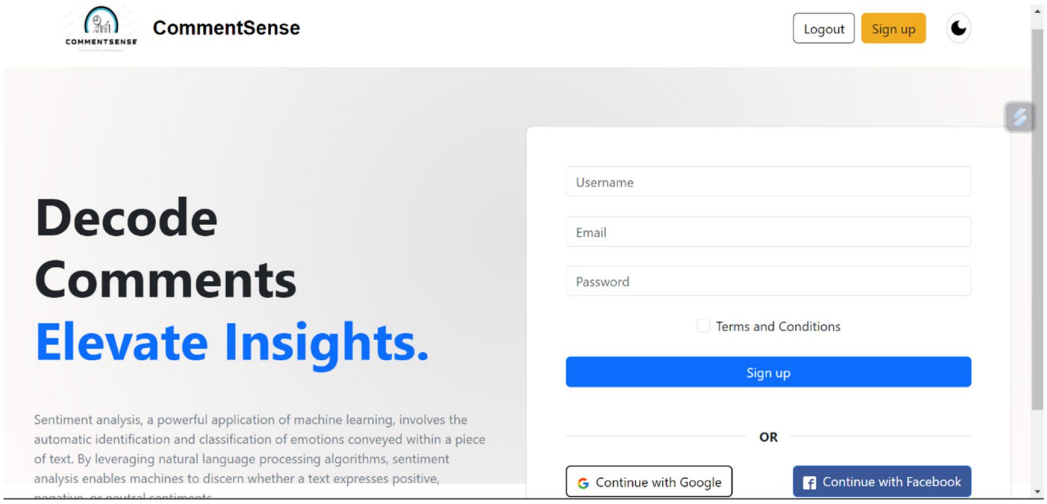
Test cases include various input scenarios such as valid YouTube video links, invalid inputs, positive/negative sentiment comments, and different user interactions with the system.

7.4.3 Expected Result

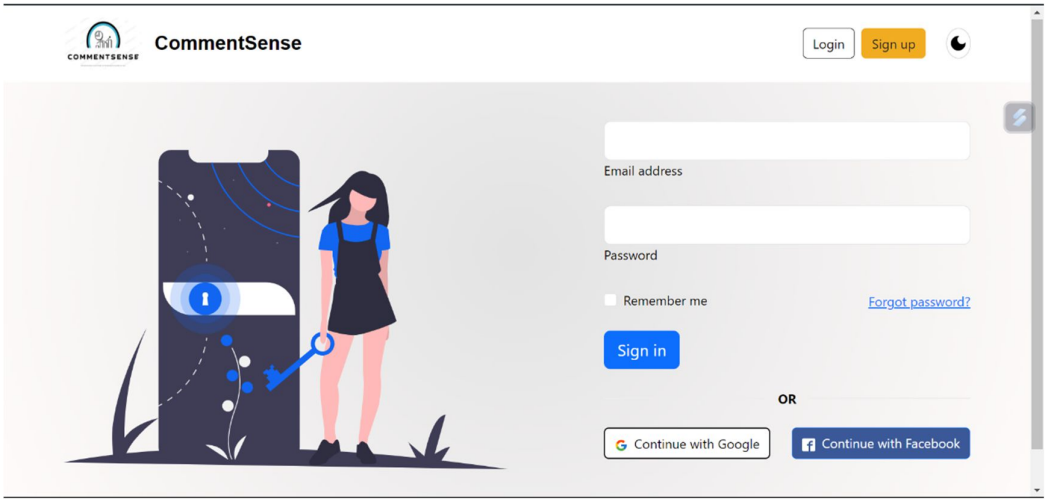
Expected results for each test case are defined to ensure that the system behaves as expected. This includes verifying correct sentiment analysis results, appropriate handling of user inputs, and proper system responses under different scenarios.

8. User Manual (Screen Shots with description)

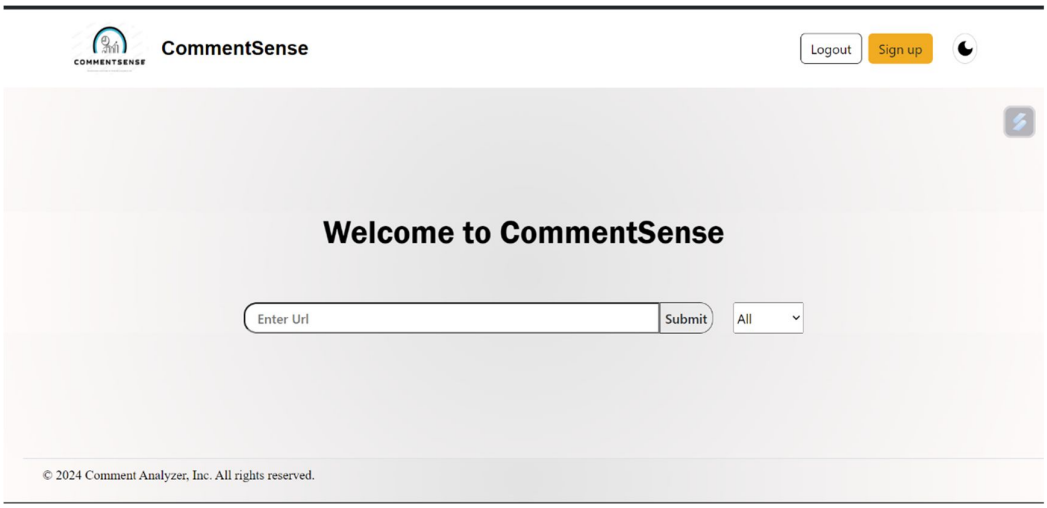
Signup Page: The signup page for a Commentsense website allows users to create accounts to access the features and functionalities of the platform.



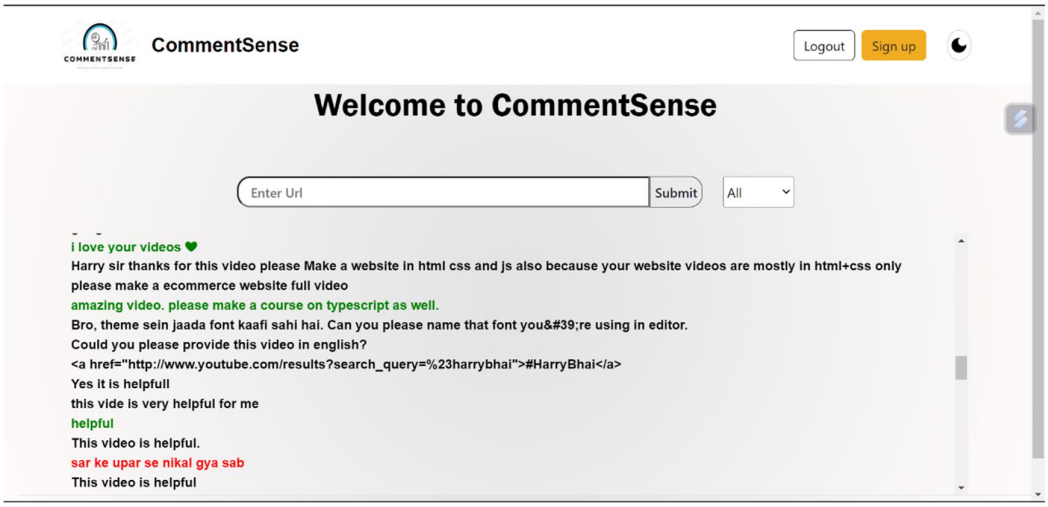
Login Page: The login page for a Commentsense website allows registered users to access their accounts and utilize the platform's features. Users can log in by entering their username or email address along with their password



Home Page: The home page of a Commentsense website can feature a functionality where users can input a YouTube video URL to initiate the comment analysis process.



Comments Output: This is simple sentiment output green is positive, red is negative and black is neutral. also provide filter to show any of the three types of comments.



9. Limitations and Future Enhancements

9.1 Limitations

- Limited Training Data: The sentiment analysis models may suffer from limited training data, affecting their ability to generalize to diverse comment types.
- Dependency on YouTube API: The system's functionality relies on the availability and reliability of the YouTube API, which may introduce potential disruptions.
- Performance Overhead: Processing a large volume of comments in real-time may impose performance overhead, leading to delays in sentiment analysis results.
- Language Support: The current models may have limitations in accurately analyzing comments in languages other than English.

9.2 Future Enhancements

- Enhanced Training Data: Acquiring a more extensive and diverse dataset for training the sentiment analysis models can improve their accuracy and generalization capabilities.
- Integration of Advanced NLP Techniques: Implementing advanced natural language processing techniques, such as deep learning models, can enhance the models' performance in analyzing complex linguistic structures.
- Multilingual Support: Extending the sentiment analysis models to support multiple languages can broaden the system's applicability and user base.
- Real-time Feedback Mechanism: Implementing a feedback mechanism to collect user feedback on sentiment analysis results can facilitate model refinement and continuous improvement.

10. Conclusion and Discussion

10.1 Conclusions and Future Enhancement

In conclusion, the YouTube Comment Analyzer system presents a valuable tool for analyzing sentiments in English YouTube comments. Despite its limitations, the system demonstrates promising capabilities in categorizing comments into positive, negative, and neutral sentiments. Future enhancements, including improvements in training data quality, integration of advanced NLP techniques, and multilingual support, can further enhance the system's effectiveness and usability.

10.2 Discussion

10.2.1 Self-Analysis of Project Viabilities

- **Strengths:** The project leverages machine learning techniques and NLP to provide valuable insights into English YouTube comments' sentiments.
- **Weaknesses:** Limitations in training data quality and dependency on external APIs pose challenges to the system's reliability and performance.
- **Opportunities:** There are opportunities for future enhancements, including the integration of advanced NLP techniques and multilingual support, to address current limitations and expand the system's capabilities.
- **Threats:** Potential disruptions in external APIs and evolving user behavior on YouTube pose threats to the system's sustainability and relevance.

10.2.2 Problems Encountered and Possible Solutions

- **Problem:** Limited training data affects the accuracy of sentiment analysis models.
- **Solution**: Acquire a more extensive and diverse dataset for model training to improve accuracy and generalization.
- **Problem:** Dependency on external APIs introduces potential disruptions.
- **Solution**: Implement fallback mechanisms or alternative APIs to mitigate the impact of API failures.
- **Problem:** Performance overhead in processing a large volume of comments in real-time.
- **Solution:** Optimize system architecture and implement caching mechanisms to improve performance.

10.2.3 Summary of Project Work

Overall, the YouTube Comment Analyzer project has made significant progress in implementing sentiment analysis functionality for analyzing English YouTube comments. While facing challenges such as limited training data and dependency on external APIs, the project has identified opportunities for future enhancements and improvements to address these challenges and deliver a more robust and effective system.