USER REQUIREMENT DOCUMENT

Introduction

This project is focused on creating a platform that helps users understand how news affects stock prices. The platform will gather historical stock data and news articles, analyze the sentiment of those articles, and show users how that news impacted stock prices over time. By using Natural Language Processing (NLP), the system will determine whether news articles have a positive, negative, or neutral effect on a company's stock. The primary users of this platform will be students, and the teaching assistant (TA) will act as the company representative. They will guide the development of the project by providing feedback on features and priorities.

Scope:

The project will focus on gathering historical stock price data and news articles, performing sentiment analysis, and displaying this information to users via an interactive and user-friendly dashboard. Users will be able to log in, analyze stock trends, view sentiment graphs, and leave comments on the data presented.

Purpose:

The purpose is to provide a tool that allows users to understand how news affects stock prices, helping them in making better investment decisions.

User Requirements

Features:

1. NLP-Based Sentiment Analysis:
   The system will automatically analyze news articles about a company and show whether the news is good, bad, or neutral for its stock.
2. User Authentication:
   Only users who sign up and log in can access the platform's main features, ensuring secure and personalized access.
3. Graphical Data Display:
   Users will see graphs showing how stock prices changed over time alongside the sentiment of related news articles.
4. Like/Dislike Functionality:
   Users will be able to like or dislike the sentiment analysis results for each news article.
5. Commenting:
   Users can leave comments on news articles or stock trends, and there will be an option to expand and see more comments.

Integration Requirements:

1. Stock Price API (e.g., Alpha Vantage): The platform will integrate with third-party stock price APIs to gather historical and real-time stock data.
2. News API (to be determined): The platform will need to fetch news articles via a reliable news API, particularly those relevant to specific companies and time periods.

User Interface (UI):

1. The interface should be clean, simple, and easy to navigate without too many pictures or distractions. The goal is for users to focus on the data and insights.

2. A dashboard will let users choose a company, display the news of the company and view stock data affected by the news article, and also see the results of the sentiment analysis.
3. The user will also be able to filter the news articles according to the greatest impact or the lowest impact. The user will have many filters too choose from for each company.

Constraints:

1. Timeline: The project needs to be developed within a constrained timeline, requiring clear prioritization of essential features.
2. Technology: The back end will be built using Flask and Python, with Python libraries used for NLP. These technologies are flexible and ideal for this project.
3. Resources: The APIs used will be free or low-cost, and data storage will be designed for efficient use of resources.

Use Cases

1. Viewing Stock Trends and Sentiment Analysis
   Title: Checking stock price and sentiment trends.
   Description: A user logs in to view stock prices for a company and how news articles affected those prices.
   Actors: The user (someone interested in stock market trends).
   Steps:
   - User logs in.
   - User selects a company and time frame.
   - The platform shows stock price changes and news sentiment over that period.
   - User reviews the data and adds a comment.

   Expected Outcome: The user can easily view stock price trends and the sentiment of related news articles.
2. Commenting on Stock Trends
   Title: Leaving a comment on stock trends.
   Description: A user leaves a comment on stock price data and news analysis.
   Actors: The user (someone who wants to share their thoughts).
   Steps:
   - User logs in.
   - User views the stock and news sentiment data.
   - User leaves a comment under the data.
   - User clicks “See More” to view additional comments.

   Expected Outcome: The comment appears under the stock data, and others can see it.
3. Liking or Disliking Sentiment Analysis
   Title: Liking or disliking the sentiment analysis.
   Description: A user interacts with the sentiment analysis by liking or disliking it.
   Actors: The user (someone who wants to provide feedback on the analysis).
   Steps:
   - User logs in.
   - User views a sentiment analysis for a news article.

- User clicks the like or dislike button.

Expected Outcome: The platform records the user's feedback on the sentiment analysis.

Priorities and Milestones

Priorities:

1. Set Up the Basic UI: Start with building a clean, simple user interface for the dashboard so users can easily access and view stock data.
2. NLP Functionality: Design the model to analyze news articles and set-up the basic algorithm for NLP.
3. User Authentication: Ensure users can sign up and log in, restricting access to core features for authenticated users only.

Milestones:

1. UI Design Completed: Create a basic interface with placeholder data to show how the dashboard will look.
2. NLP Set Up and Tested: Integrate and test the NLP model to analyze news sentiment.
3. Authentication System Working: Implement a sign-in and log-in system that only allows registered users to access the features.
4. Stock Trend Insights: Enable the system to track and display how news affects stock prices, helping users determine how the news has affected the stock prices previously.

Tentative TimeLine:

End of September:

1. MongoDB set up and tested
2. NLP model set up and tested
3. APIs finalized
4. Literature review completed
5. Basic dashboard set up

Mid-October:

1. Data collected from at least 5 companies for APIs
2. NLP models compared to find the best fit
3. Data displayed on the dashboard
4. Best libraries identified for trend visualization

End of October:

1. Analysis of 2 companies completed
2. Data from 2 companies displayed on the dashboard

Mid-November:

1. Analysis completed for the remaining companies

End of November:

1. Documentation finished
2. Project ready for delivery