Software Requirements Specification

for

MarketMaven

(An Advanced Strategic Investing)

Prepared by

Batch no. 23

Bochkar Nikhith	22N31A6629
Chandarlapati Venkat	22N31A6638
Gazula Joshua Anand	22N31A6656
Guguloth Ravalika	22N31A6661

Under the Guidance of

Mrs. N. Radhika

Assistant Professor

Department of Computational Intelligence

MRCET

Table of Contents

List of Figures

INTRODUCTION

The stock market is a highly complex and volatile system influenced by numerous factors, such as company performance, economic conditions, investor sentiment, and global events. Predicting stock market trends is challenging but crucial for maximizing financial returns and minimizing risk. Traditionally, stock market predictions have relied on **fundamental analysis** and **technical analysis**. While fundamental analysis evaluates a company's intrinsic value, technical analysis examines past price data to predict future trends. However, these methods have limitations, often failing to capture short-term fluctuations and non-linear relationships. Recent advancements in machine learning, particularly deep learning, have provided more sophisticated solutions for stock market prediction.

The **MarketMaven** project leverages **Recurrent Neural Networks (RNNs)**, specifically **Long Short-Term Memory (LSTM)** and **Gated Recurrent Units (GRU)**, to model time-series data for accurate trend forecasting. Unlike traditional methods, MarketMaven incorporates diverse data sources such as financial news sentiment and macroeconomic indicators to provide more comprehensive insights. By combining historical stock data with real-time factors, the system aims to offer data-driven predictions that help investors make informed and strategic decisions.

1. LITERATURE SURVEY

The field of stock market prediction has undergone significant evolution, with a shift from traditional analytical methods to more sophisticated machine learning and deep learning techniques. This literature survey provides a comprehensive overview of existing approaches, their strengths, limitations, and the advancements that have shaped current methodologies. The following points summarize key concepts and findings from various sources relevant to the **MarketMaven** project on stock market prediction, including the data sources for each reference:

- Edwards and Magee in *"Technical Analysis of Stock Trends"* [1] described Traditional approaches to stock evaluation include fundamental and technical analysis. Fundamental analysis assesses a company's financial health, industry position, and economic factors to predict stock performance, as detailed by Graham and Dodd in their 1934 book, *Security Analysis*. In contrast, technical analysis focuses on historical price data and trading volume to identify trends, utilizing tools like Simple Moving Average (SMA) and Exponential Moving Average (EMA), as by Together, these methods provide investors with diverse strategies for analyzing market opportunities.

- Tsai and Wu *"Expert Systems with Applications"* [2] highlighted the advancements study published in the emergence of machine learning has significantly improved stock market predictions, surpassing traditional analytical methods. Techniques such as support vector machines have been effectively utilized to forecast stock prices, demonstrating the potential of machine learning in financial analysis. As machine learning continues to evolve, it offers new avenues for enhancing investment strategies and decision-making.

- Hochreiter and Schmidhuber *"Long Short-Term Memory with Neural Computation"* [3] describing Recurrent Neural Networks (RNNs) are specifically designed to process sequential data, making them well-suited for time-series prediction tasks such as stock forecasting. Despite their strengths, RNNs can encounter challenges like vanishing gradient problems, which hinder their ability to learn long-term dependencies. Which is a type of RNN that mitigates this problem. By enhancing the learning capabilities of RNNs, LSTMs have become a powerful tool in financial modeling and predictions.

- Yoon et al. *"Stock Price Prediction Using LSTM"* [4] demonstrated the efficacy of LSTMs presented at the International Conference on Big Data and Smart Computing. Long Short-Term Memory (LSTM) networks were developed to overcome the limitations of traditional Recurrent Neural Networks (RNNs) by effectively maintaining long-term dependencies in sequential data. This capability makes LSTMs particularly effective for tasks such as stock price prediction. As a result, LSTMs have become a prominent tool for financial forecasting in machine learning applications.

- Cho et al. *"Learning Phrase Representations using RNN Encoder-Decoder"* [5] introduced GRUs at the Conference on Empirical Methods in Natural Language Processing. Gated Recurrent Units (GRUs) offer a simpler architecture compared to Long Short-Term Memory (LSTM) networks, requiring fewer parameters while delivering comparable performance. This efficiency makes GRUs a compelling alternative for stock prediction tasks. As a result, GRUs have gained popularity in various applications where effective sequence modeling is essential.

- Bontempi et al. *"Data-driven modeling of the stock market"* [6] highlighted the effectiveness of such data-driven modeling presented at the International Conference on Machine Learning and Applications. Hybrid models that combine traditional methods with machine learning techniques can enhance prediction reliability by leveraging the strengths of both approaches. This integration allows for more robust analyses, capturing various market dynamics. By utilizing hybrid models, investors can achieve more accurate stock market forecasts.

This literature survey provides a foundational understanding of the existing approaches and methodologies in stock market prediction, forming the basis for the proposed advancements in the **MarketMaven** project.

1.1 Existing System:

Existing systems for stock market prediction mainly rely on traditional methods like **fundamental analysis** and **technical analysis**. Fundamental analysis assesses a company's financial health and market conditions but often overlooks short-term fluctuations. Technical analysis uses historical price patterns and trading volumes through tools like the **Simple Moving Average (SMA)** and **Exponential Moving Average (EMA)**, which are slow to respond to sudden changes. Many models also focus on limited features, ignoring external factors such as macroeconomic data and market sentiment. While basic machine learning approaches like **linear regression** and **random forests** are employed, they may fail to capture the complex, non-linear relationships in stock data and are prone to overfitting. Consequently, existing systems struggle to provide accurate predictions and adapt to real-time market conditions.

1.2 Drawbacks of Existing System:

The existing systems for stock market prediction face several significant drawbacks. First, traditional methods like **fundamental analysis** often overlook short-term market fluctuations, leading to missed opportunities for timely trading decisions. **Technical analysis** relies heavily on lagging indicators, such as the **Simple Moving Average (SMA)** and **Exponential Moving Average (EMA)**, which can be slow to react to sudden price changes, making them less effective in volatile markets.

- Limited scope

- Lack of generalizability

- Limited accuracy

- Data quality issues

- Limited scalability

1.3 Proposed System:

The proposed system, **MarketMaven**, aims to enhance the prediction of stock market trends by leveraging advanced deep learning techniques, specifically Recurrent Neural Networks (RNNs) and their variants such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). Unlike traditional models that rely heavily on historical stock prices and simplistic algorithms, MarketMaven focuses on capturing

complex, non-linear patterns in time-series data, offering more accurate and timely predictions. The system integrates diverse data sources to build a comprehensive, real-time predictive model for stock market movements.

Also, we use advanced libraries of Deep Learning with the existing framework, such as PyTorch and Keras which gives us the fast computation for the RNN module to run faster. For machine training, we use algorithms like LSTM, GRU.
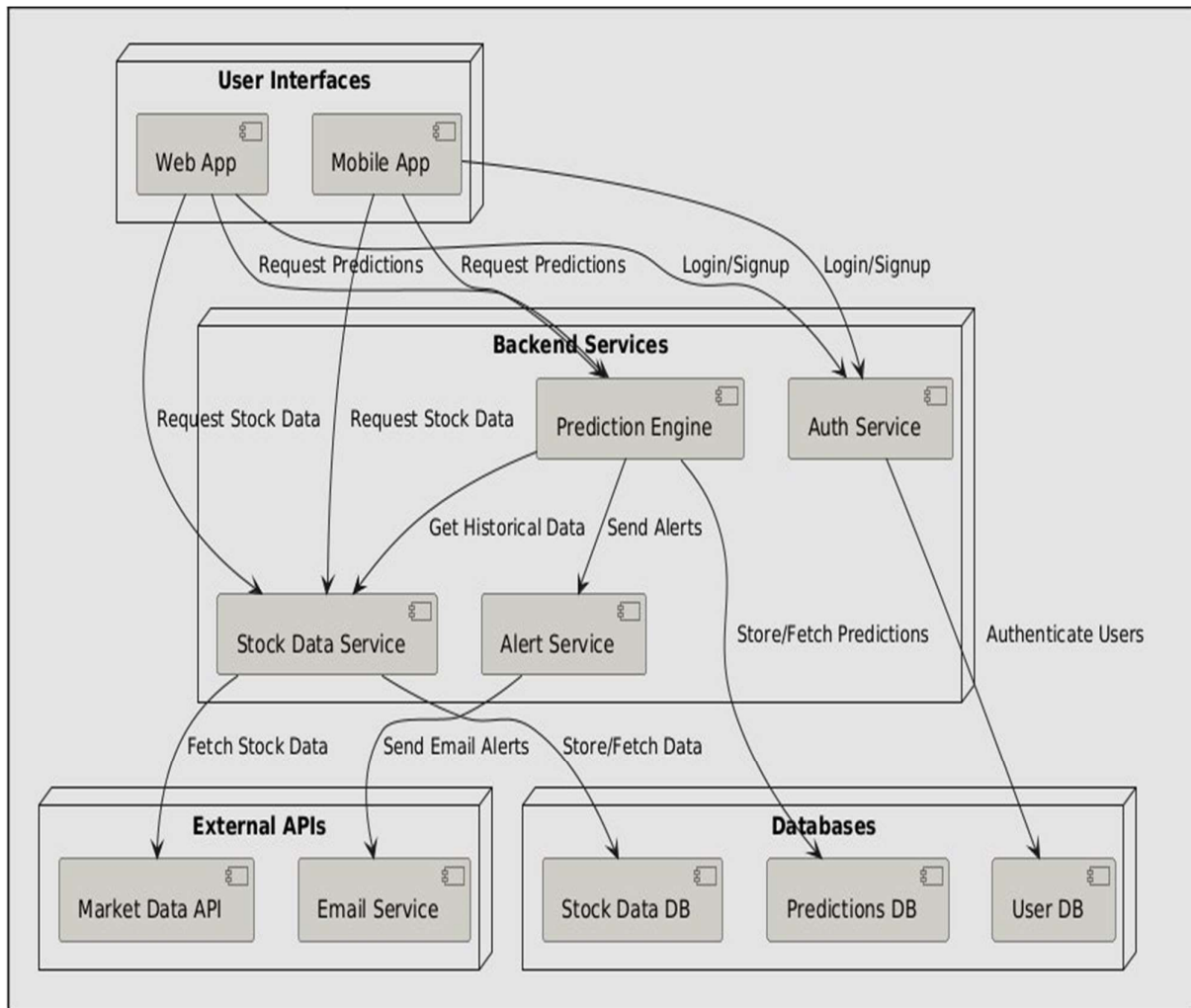
1.4 Advantages:

The proposed **MarketMaven** system offers several advantages over traditional stock market prediction models, addressing many of the limitations that current systems face. These advantages span across prediction accuracy, data handling, user adaptability, and system performance. Below is a detailed exploration of the key advantages that MarketMaven brings to investors and financial analysts:

- Improved Prediction Accuracy
- Adaptability to Market Changes
- Comprehensive Data Integration
- Interpretable
- Personalized inventions
- Cost effective

2. SYSTEM ARCHITECTURE AND UML DIAGRAMS:

2.1 System Architecture:

Fig 2.1: System Architecture

2.1 UML Diagrams:

2.2.1 Use-case Diagram:

MarketMaven

View Stock Prices

Receive Alerts

Analyze Trends

Generate predictions

Investor

Analyst
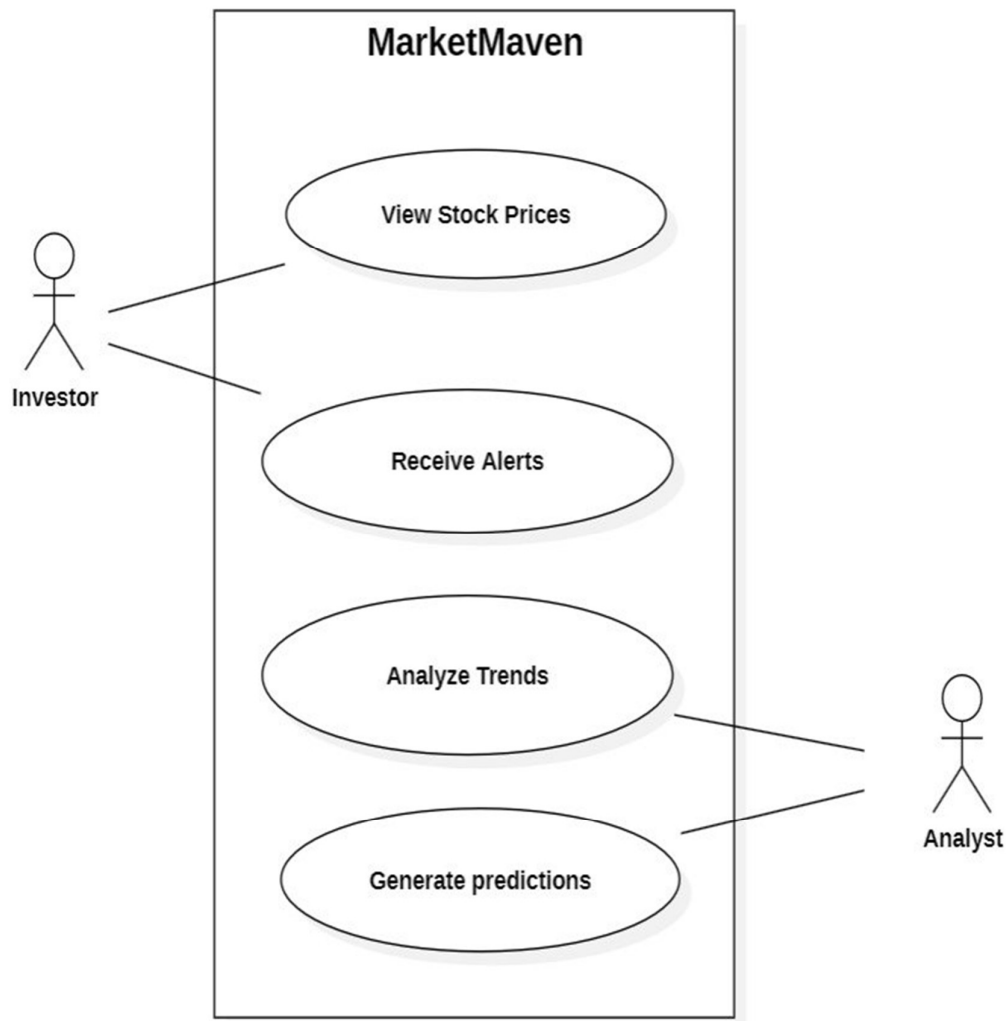
Fig 2.2: Use-case Diagram:

2.2.2 Class Diagram:

Fig 2.3: Class Diagram:

2.2.3 Sequence Diagram:

Fig 2.4: Sequence Diagram

2.2.4 Activity Diagram:

Fig 2.5: Activity Diagram

3. OVERALL DESCRIPTION

3.1 Feasibility Study:
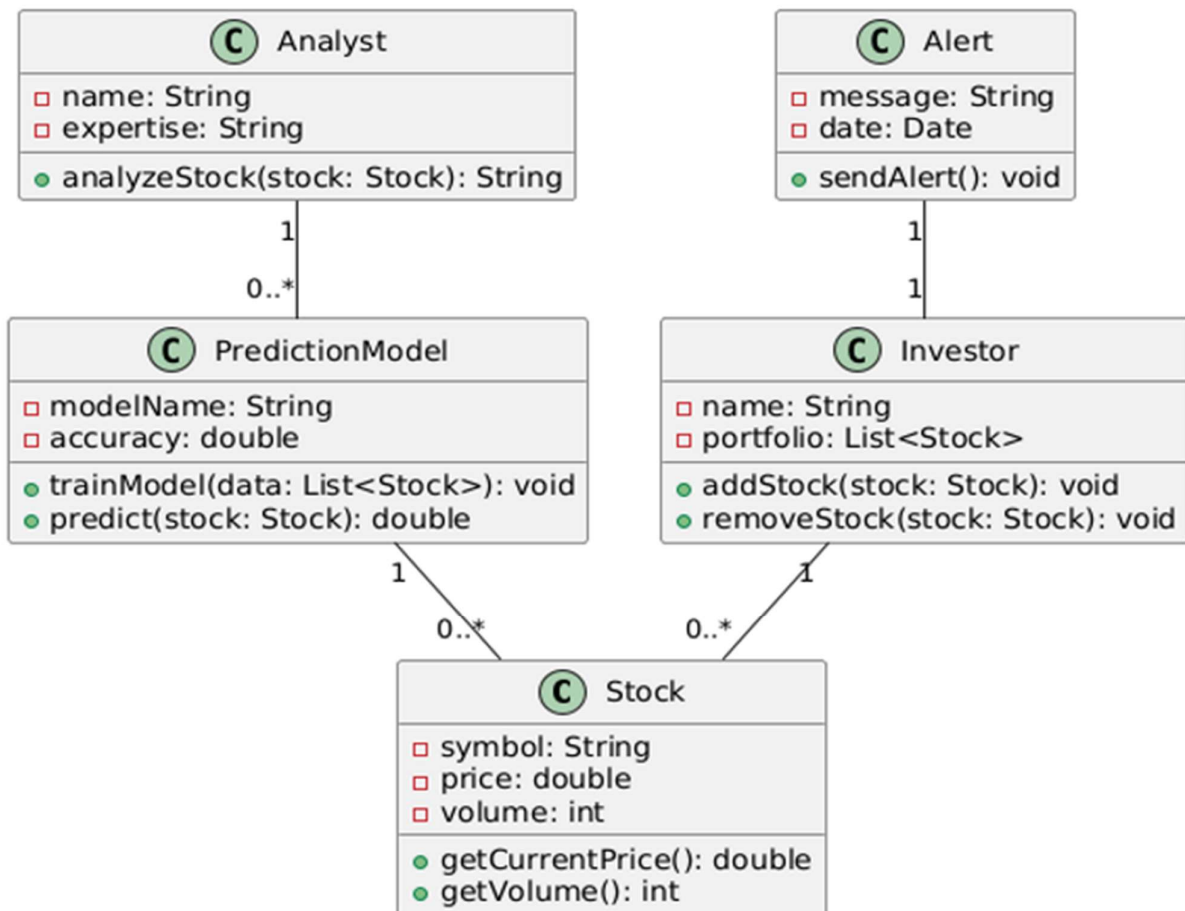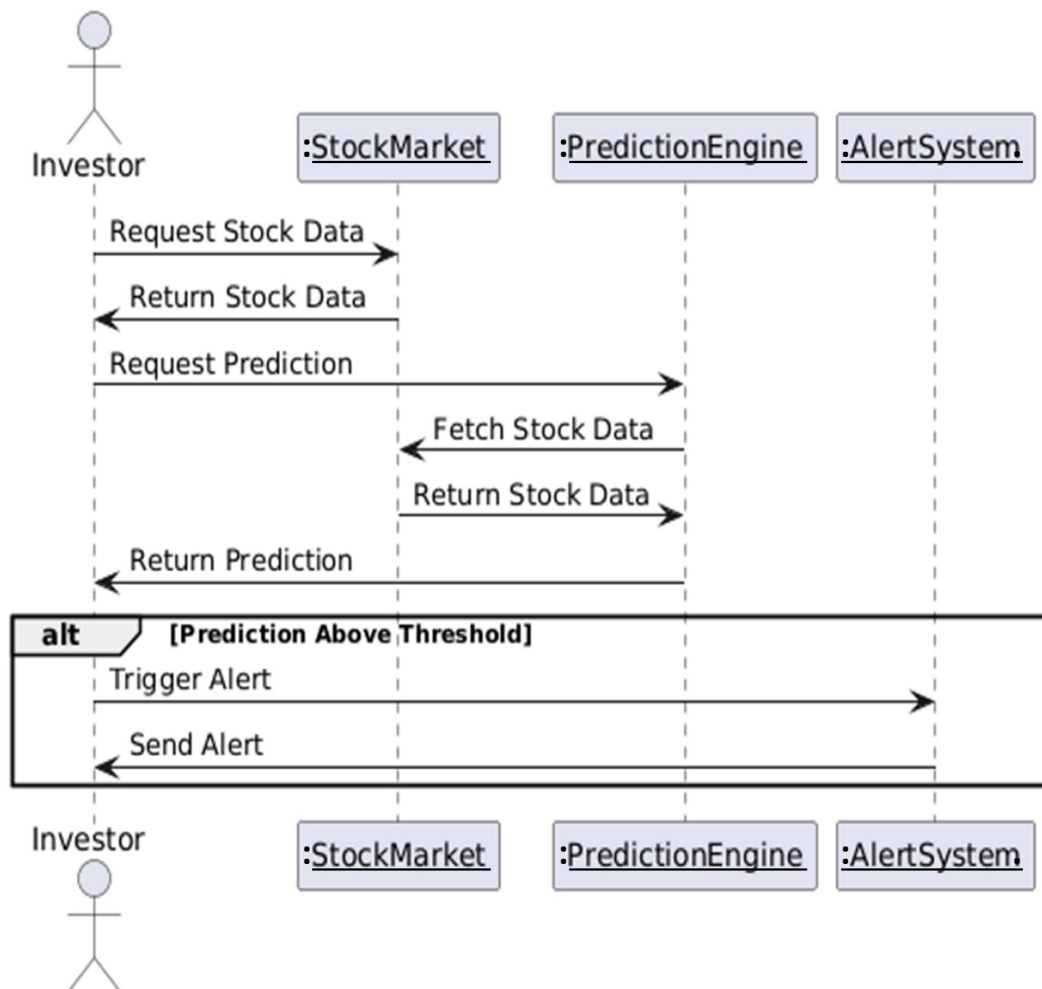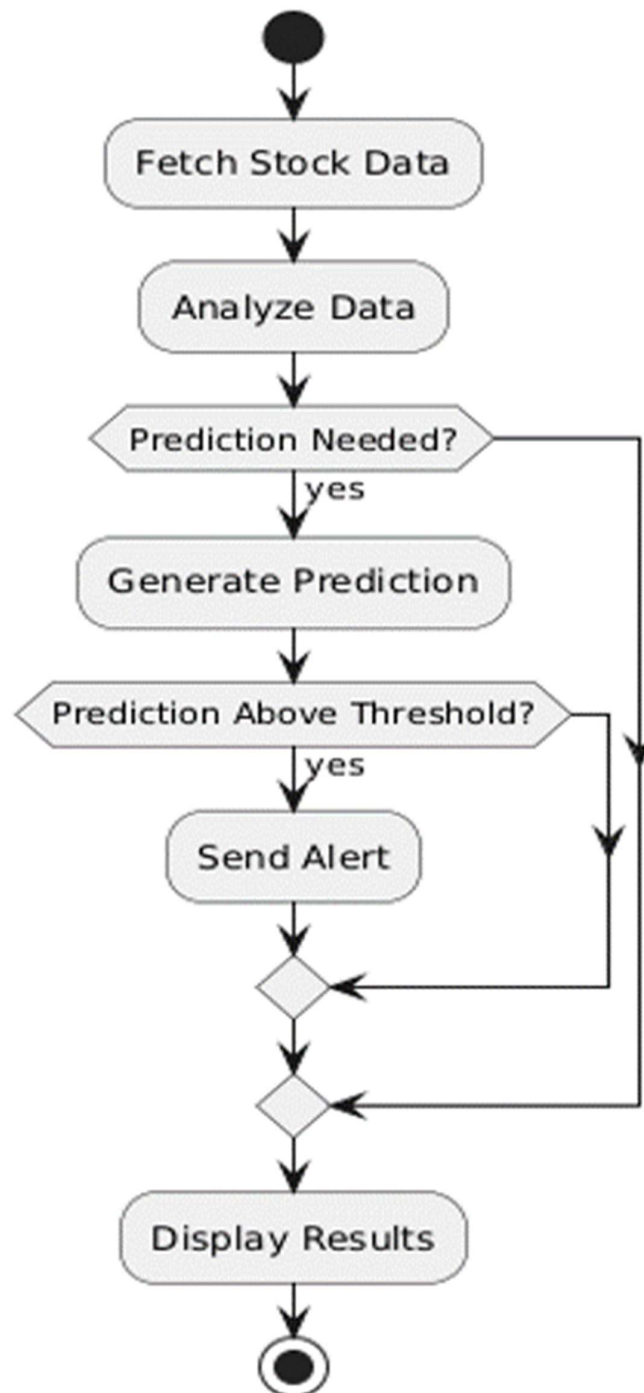
The feasibility study evaluates the practicality of the **MarketMaven** project in terms of operational, economic, technical, and legal factors. This ensures the project can be implemented and deployed successfully while complying with legal and ethical standards.

3.1.1 Operational Feasibility:

MarketMaven is designed to provide real-time stock predictions by analyzing vast amounts of historical and real-time data using advanced deep learning algorithms.

- **Scalability**: MarketMaven can leverage cloud platforms (AWS, GCP, etc.) for real-time stock predictions, ensuring scalable and responsive operations.
- **Data Integration**: The system can integrate various data sources (stock prices, financial news, etc.) using cloud services to provide continuous, up-to-date information.

In summary, the operational aspects of **MarketMaven** are feasible due to advancements in cloud computing and data integration technologies, which enable high-speed, real-time processing.

3.1.2 Economic Feasibility:

MarketMaven can be economically feasible due to several factors:

- **Cost Efficiency**: Cloud infrastructure allows scalable, cost-effective resource management, reducing operational expenses.
- **ROI Potential**: Accurate stock predictions can deliver a high return on investment, making the initial deployment cost economically feasible.

Thus, the project is cost-effective if managed efficiently with cloud-based infrastructure and proper resource allocation.

3.1.3 Technical Feasibility:

The technical feasibility of **MarketMaven** is ensured by the availability of cutting-edge technologies for data processing and deep learning:

- **Advanced Frameworks**: Utilizing TensorFlow, Keras, and PyTorch ensures the system can handle complex data and model architectures.
- **Cloud Resources**: High-performance cloud computing (GPUs, TPUs) allows fast training and real-time prediction processing.

Overall, the availability of advanced deep learning frameworks, cloud-based data management tools, and integration APIs ensures that the **MarketMaven** project is technically feasible.

3.1.4 Legal and Ethical Feasibility:

Legal and ethical compliance is crucial for **MarketMaven**, particularly because it handles potentially sensitive financial data and user information:

- **Data Privacy Compliance**: MarketMaven must comply with regulations like GDPR and CCPA, ensuring user consent and data protection.
- **Data Security**: Strong encryption and secure access control will be employed to safeguard sensitive financial data and prevent breaches.

By adhering to these legal and ethical guidelines, **MarketMaven** ensures compliance with both regulatory standards and best practices in data privacy and financial market operations.

4. SYSTEM ANALYSIS

4.1 Software Requirement Specifications:

A software requirements specification (SRS) is a comprehensivedescription of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform. An SRS minimizes the time and effort required by developers to achieve desired goals and minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs, and human users in a wide variety of real-world situations. Hence, westart the SRS with the feasibility study.

4.2 Hardware and Software Requirements

4.2.1 Hardware Requirements

To support the computational needs of deep learning and real-time data processing, **MarketMaven** requires high-performance hardware infrastructure. Below are the specific hardware requirements:

- **Processor (CPU)**: Minimum: Intel Core i7 or AMD Ryzen 7 (8 cores, 3.0 GHz or higher)
- **Graphics Processing Unit (GPU)**: GPUs with at least 16 GB VRAM for deep learning tasks.
- **Memory (RAM)**: Minimum: 16 GB DDR4 RAM for development environments.
- **Storage**: **SSD (Solid State Drive)**: Minimum 512 GB SSD for faster data access and to store essential project files.
- **Power Supply and Backup**: Uninterruptible Power Supply (UPS) to ensure system stability during extended training sessions and data processing.
- **Networking**: **Internet Speed**: High-speed broadband (100 Mbps or higher) for real-time data streaming and API interactions.

4.2.2 Software Requirements

To ensure optimal performance for developing, training, and deploying deep learning models for stock prediction, the following software components are required:

- **Operating System**: Windows 10/11 or Ubuntu Linux for development;

  Ubuntu 20.04 LTS for production.

- **Languages**: Python 3.x for models; JavaScript, HTML/CSS for front-end.

- **Libraries**: TensorFlow, Keras, PyTorch for deep learning; NumPy, Pandas.

- **Tools**: Jupyter Notebook, VS Code for development;

  Git, Docker for version control and deployment.

- **Cloud**: AWS, GCP, or Azure for scalable deployment.

- **Database**: PostgreSQL/MySQL for structured, MongoDB for unstructured data.

- **Security**: SSL/TLS and OAuth 2.0 for data and user protection.

5. FUNCTIONAL REQUIREMENTS

5.1 Data Collection:

The system shall collect a dataset of defective images and classification with the following features: detent coordinates, types of physical or locomotive damage, duration and possible cause.

5.2 Data Preprocessing:

The system shall preprocess the dataset by handling missing values, outliers,and categorical variables, and by normalizing the numerical features.

5.3 Feature Selection:

The system shall select the most relevant features for exterior aircraft defect detection using deep learning and machine learning techniques such as PyTorch, Keras MobileNet framework, etc.

5.4 Model Training:

The system shall train machine learning models such as logistic regression, decision tree, random forest, support vector machine, and neural network on the preprocessed dataset, using cross-validation and hyperparameter tuning to optimize the performance metrics.

5.5 Model Evaluation:

The system shall evaluate the performance of the trained models on a test dataset, using various metrics such as accuracy, precision, recall, F1 score, and area under the curve (AUC), and visualizing the results with confusion matrices andreceiver operating characteristic (ROC) curve

6. NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are important considerations in the development and implementation of a heart disease prediction system. Here are some potential non-functional requirements for a heart disease prediction research paper

6.1 Performance:

The system must perform efficiently and process tasks in a reasonable amount of time, with minimal latency and response time.

6.2 Accuracy:

The system must produce accurate results for object detection and machine translation tasks, with a high level of precisionand recall

6.3 Robustness:

The system must be robust and resilient to handleunexpected errors or inputs, and recover gracefully from failures.

6.4 Compatibility:

The system must be compatible with various hardware and software configurations, and support different operating systems and platforms.

6.5 Maintainability

The system should be easy to maintain and update as needed to ensure ongoing effectiveness.

6.6 Scalability:

The system should be able to scale up or down as needed to accommodate changing patient populations and healthcaresystems.

7. REFERENCES:

1. Edwards, R., & Magee, J. (1948). Technical Analysis of Stock Trends. 5th Edition. New York Institute of Finance.

2. Tsai, C. F., & Wu, H. C. (2008). Predicting stock prices with support vector machines. Expert Systems with Applications, 34(4), 2281-2289.

3. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735-1780.

4. Yoon, J., et al. (2018). Stock Price Prediction Using LSTM Recurrent Neural Network. Proceedings of the 2018 International Conference on Big Data and Smart Computing (BigComp).

5. Cho, K., et al. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).

6. Bontempi, G., et al. (2012). Data-driven modeling of the stock market. Proceedings of the 2012 International Conference on Machine Learning and Applications (ICMLA).