## UNIVERSITY OF MUMBAI



## Master of Computer Application

**In-Semester Capstone Project** **Report on**

## Submitted by –

## Roll no –

(Semester-I)

**Year 2024-25**

Under the Guidance of

**Table of Contents Page no:**

|  |  |
| --- | --- |
| **1. Introduction** | **3** |
| * Introduction of Project | **3** |
| * Project Purpose | **4** |
| * Project Objective | **5** |
| * Scope of Project | **5** |
| **2. System Study** | **6** |
| * [Requirement Analysis](#_30j0zll) | **8** |
| * [Planning and Scheduling](#_1fob9te) | **9** |
| * [Preliminary Product Description](#_3znysh7) | **10** |
| * [Justification of Platform](#_2et92p0) | **11** |
| * [Conceptual Model](#_tyjcwt) | **12** |
| **3. Analysis and Design** | **13** |
| * [Hardware Requirements](#_3dy6vkm) | **14** |
| * [Software Requirements](#_1t3h5sf) | **15** |
| * [Actual Gantt Chart](#_4d34og8) | **15** |
| * [System Design](#_2s8eyo1) | **16** |
| * [UML Diagrams](#_17dp8vu) | **17** |
| * [Flow Chart](#_3rdcrjn) | **18** |
| * [Data Flow Diagram](#_26in1rg) | **18** |
| * [Entity Relationship Diagram](#_lnxbz9) | **19** |
| * [Description of Entities](#_35nkun2) | **20** |
| **4. Testing and Validation** | **21** |
| * Code Efficiency | **22** |
| * [Testing Approach](#_1ksv4uv) | **24** |
| * [Unit Testing](#_44sinio) | **25** |
| * [Integration Testing](#_2jxsxqh) | **25** |
| **5.User Manual** | **26** |
| * User Documentation | **27** |
| * Outputs | **29** |
| * Code Snippets | **30** |
| **6.Conclusion** | **31** |
| * Conclusion | **31** |
| * Limitations | **31** |
| * Future Scope | **32** |

**Introduction to Stock Prediction**

**Project Overview**

The stock market is a cornerstone of the global financial ecosystem, serving as a critical platform for driving economic growth, fostering business expansion, and enabling wealth creation. It connects companies seeking capital to investors eager for opportunities, forming a dynamic environment where trillions of dollars exchange hands daily. However, this dynamic environment is characterized by its inherent volatility, driven by a multitude of factors, including economic indicators, geopolitical events, corporate earnings, and investor sentiment. This volatility makes accurate stock price prediction both a highly desired and inherently challenging task.

For investors and traders, the ability to anticipate market trends and price movements is invaluable. Accurate predictions enable market participants to make informed decisions, minimize risks, and capitalize on opportunities. However, traditional methods of stock analysis, such as fundamental or technical analysis, often fall short in addressing the complexities and nuances of market behavior. To address these challenges, advanced techniques powered by machine learning and artificial intelligence have emerged as game-changing tools.

This project seeks to harness the power of machine learning to develop a robust, efficient, and user-friendly stock prediction application. The application will analyze historical stock market data, leveraging sophisticated forecasting algorithms to provide actionable insights into short-term price movements. By employing cutting-edge technologies such as Long Short-Term Memory (LSTM) networks, Facebook’s Prophet, the project aims to deliver a comprehensive solution for stock market forecasting.

The overarching goal of this project is to empower both novice and experienced investors with actionable intelligence. Whether an individual is new to investing or a seasoned trader, this tool will provide valuable insights to optimize their investment strategies. By offering accurate predictions, the application will help users identify potential investment opportunities, mitigate risks associated with market volatility, and make informed buy, sell, or hold decisions.

**Project Purpose**

The primary purpose of this project is to empower individuals with the knowledge and tools necessary to make informed investment decisions. Key aspects include:

1. **Enhance Decision-Making**

Equip users with insights to make confident buy, sell, or hold decisions.

Provide data-driven analysis for evaluating market conditions effectively.

1. **Identify Potential Opportunities**

Highlight stocks with promising upward trends for profitable investments.

Support both short-term trading and long-term portfolio growth by identifying emerging trends.

1. **Mitigate Risks**

Alert users to potential market downtrends or heightened volatility.

Enable proactive risk management to safeguard investments, especially for novice investors.

1. **Democratize Financial Analysis**

Make advanced stock prediction techniques accessible to a broader audience.

Simplify complex algorithms and present results in a user-friendly format.

1. **Promote Education and Awareness**

Help users understand stock market dynamics and predictive analytics.

Foster a more informed and confident investor community.

1. **Level the Playing Field**

Reduce barriers to entry in financial markets by providing accessible tools.

Empower users of all backgrounds to succeed in the stock market.

1. **Empower Strategic Investment**

Enable users to act strategically by anticipating market trends.

Equip both novice and experienced investors with actionable intelligence.

This project ultimately aims to revolutionize stock market participation by equipping individuals with the resources they need to make informed, strategic, and confident investment decisions.

**Project Objectives**

To achieve these goals, the project focuses on the following specific objectives:

1. **Data Acquisition and Preprocessing:**
   * Collect historical stock data from reliable sources such as Yahoo Finance
   * Clean and preprocess the data to handle missing values, outliers, and inconsistencies.
2. **Model Development and Training:**
   * Implement and train multiple machine learning models, including:
     + **LSTM:** Long Short-Term Memory networks for capturing temporal dependencies in time series data.
     + **Prophet:** Facebook's Prophet for time series forecasting, particularly suited for trend analysis.
   * Optimize model hyperparameters to achieve optimal performance.
   * Evaluate model performance using appropriate metrics (e.g., Mean Squared Error, Mean Absolute Error, Root Mean Squared Error).
3. **Model Deployment and User Interface:**
   * Deploy the selected models on a user-friendly platform (e.g., Streamlit).
   * Develop an intuitive user interface that allows users to:
     + Search for specific stocks.
     + Visualize historical price trends and predicted future prices.
     + Access detailed model performance metrics.
     + Receive personalized recommendations based on the predicted trends.

**Project Scope**

The scope of this project encompasses the following:

* **Data:** Historical stock data from Yahoo Finance.
* **Models:** LSTM, Prophet.
* **Features:** Technical indicators, fundamental metrics, and time-based features.
* **User Interface:** A web-based application built using Streamlit.
* **Predictions:** Short-term price predictions (e.g., 1-month).

By successfully achieving these objectives, this project aims to provide a valuable tool for investors and traders, empowering them to make informed decisions in the complex world of stock market investing.

**System Study**

**Requirement Analysis**

1. **Functional Requirements:**

* **Input:**Users will input the stock symbol or company name to retrieve the relevant stock data for the selected company. The system will fetch historical data for the specified stock from a reliable source, such as Yahoo Finance**.**
* **Output:**The application will provide predicted stock prices for the next 2-3 days based on the data processed by the machine learning models. The predicted prices will be displayed in dynamic charts to give users a clear understanding of the trends. Additionally, the system will show performance metrics (e.g., Mean Squared Error, Root Mean Squared Error) to evaluate the accuracy of the predictions.
* **Features:**

1. **Model Comparison:**

* The application will integrate multiple machine learning models, such as LSTM (Long Short-Term Memory) and Prophet, to generate stock predictions. Users will have the ability to select which model they want to use for predictions or allow the system to automatically compare multiple models.
* The system will display both the results of the selected model and the comparative analysis of other models to provide a broader view of potential price movements. The models will be evaluated side by side, showing their respective predictions in the same graph for easy comparison.
* Each model will be paired with a performance evaluation, showcasing its strengths and weaknesses based on the historical data, allowing users to make an informed decision about which model best suits their needs.

1. **Prediction Accuracy:**

* The system will offer a detailed overview of the prediction accuracy. It will compute and display performance metrics such as error rates, comparing the actual stock prices against the predicted values to highlight the model's efficiency.
* A model performance dashboard will be available, where users can see at a glance which model has provided the most accurate predictions. The dashboard will include statistics such as error rates, the variance between predictions and actual prices, and other relevant metrics.
* For each model, there will be a clear breakdown of the different types of errors (e.g., underprediction vs. overprediction) and visual cues that help users identify when and why a model may have failed to deliver optimal results.

1. **Non-Functional Requirements:**

* **Performance:**The system must have quick response times, providing stock predictions almost instantly after user input. The time taken for fetching data, processing it, training the models, and displaying the results should be minimized to ensure a smooth user experience.
* **Usability:**The application will be designed with a user-friendly interface to ensure it is accessible for both technical and non-technical users. The UI will be simple and intuitive, with clear instructions and easy-to-understand visualizations. Non-technical users should be able to easily input stock symbols, view predicted trends, and compare results without needing any prior knowledge of machine learning.
* **Scalability:**The system should be scalable to handle a growing number of users. As the user base increases, the platform should be able to process multiple requests simultaneously without significant performance degradation.

**Planning and Scheduling**

**Project Timeline:**

* **Week 1-2: Data Collection and Preprocessing**
* Collect historical stock data from sources like Yahoo Finance, BSE APIs, or Alpha Vantage.
* Clean and preprocess the data to handle missing values, outliers, and inconsistencies.
* Extract relevant features, including technical indicators and other stock metrics.
* **Week 3-4: Model Development and Evaluation**
* Develop and train multiple machine learning models like LSTM, Prophet.
* Fine-tune hyperparameters for optimal model performance.
* Evaluate model performance using appropriate metrics such as Mean Squared Error and Mean Absolute Error.
* **Week 5: Integration with Streamlit UI**
* Build a user-friendly interface using Streamlit.
* Integrate the trained models with the UI to allow users to input stock symbols and view predictions.
* Implement visualization of predictions and model performance.
* **Week 6: Testing and Deployment**
* Thoroughly test the application to ensure it functions as expected.
* Test the prediction accuracy and model comparison features.
* Deploy the application on a cloud platform for scalability and accessibility.

**Preliminary Product Description**

The Stock Prediction Application is designed to help users forecast short-term stock price trends, providing insights into potential market movements for the next 2-3 days. The core functionality of the application includes:

1. **Stock Price Predictions:**The application uses machine learning models, such as LSTM (Long Short-Term Memory), Prophet, to analyze historical data and predict future stock prices. The predictions are based on patterns found in the historical price data, providing users with a forecast of price movements over a short-term horizon (2-3 days).
2. **Dynamic Visualizations:**The predicted stock price trends are displayed through interactive, dynamic charts. These charts allow users to visualize the expected price changes, making it easier to understand potential market fluctuations. The charts will show the predicted prices along with historical trends, providing users with a clear comparison between past and forecasted data.
3. **Model Comparison:**The application will allow users to compare predictions made by different models, enabling them to evaluate which model provides the most reliable and accurate forecast based on performance metrics. Users can select the model they prefer based on its historical accuracy or their specific investment needs**.**
4. **Performance Metrics:**The system will display performance metrics for each model, such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). These metrics will inform users about the accuracy of the predictions, helping them assess the reliability of the forecast before making investment decisions.
5. **User-Friendly Interface:**The application will feature an easy-to-navigate interface built using Streamlit, allowing users to input stock symbols or company names. The interface will be designed for both beginners and experienced investors, with intuitive controls for searching stocks, viewing predictions, comparing models, and reading related news.

By providing these features, the application aims to equip users with the necessary tools to make more informed and confident investment decisions in the stock market.

**Justification of Platform**

* **Frontend:** Streamlit for its simplicity and rapid development capabilities.
* **Backend:** Python, leveraging libraries like TensorFlow, Scikit-Learn, and Prophet.

**Conceptual Model**

The application is structured with the following layers:

* **Data Layer:** Collects and preprocesses stock data.
* **Model Layer:** Executes predictive models and generates outputs.
* **Presentation Layer:** Visualizes predictions and accepts user input.

**Analysis and Design**

**Hardware Requirements**

* Minimum Requirements:

Processor: Intel Core i3 (or equivalent).

RAM: 4 GB.

Storage: 128 GB HDD/SSD.

GPU: Integrated Graphics (e.g., Intel UHD Graphics).

* Recommended Requirements:

Processor: Intel Core i5 or AMD Ryzen 5 (or equivalent).

RAM: 8 GB.

Storage: 256 GB SSD.

GPU: Dedicated GPU with at least 2 GB VRAM (e.g., NVIDIA GeForce MX250, GTX 1050).

VRAM (e.g., NVIDIA GeForce RTX 3060, 3070, or AMD Radeon RX 6700 XT).