### A REPORT ON

MACHINE LEARNING INTERNSHIP REPORT – ZAWR INDUSTRIES LLP

***Submitted by,***

## Mr. AHMED PASHA - 20211CSE0358

***Under the guidance of,***

## Mr. ASAD MOHAMMED KHAN

***in partial fulfillment for the award of the degree of***

# BACHELOR OF TECHNOLOGY

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**At**

****

**PRESIDENCY UNIVERSITY, BENGALURU**

**MAY 2025**

**PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that the Internship/Project report **“Machine Learning Internship – Zawr Industries LLP”** being submitted by “Ahmed Pasha” bearing roll number “20211CSE0358” in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

|  |  |
| --- | --- |
| **Mr. ASAD MOHAMMED KHAN**  Professor, School of CSE&IS PSCS, Presidency University  Bangalore - 560054 | **Dr. Asif Mohamed H B** HOD, School of CSE&IS PSCS, Presidency University  Bangalore - 560054 |

|  |  |
| --- | --- |
| **Dr. MYDHILI NAIR**  Associate Dean PSCS  Presidency University | **Dr. SAMEERUDDIN KHAN**  Pro-Vice Chancellor - Engineering Dean –PSCS / PSIS  Presidency University |

# PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

**DECLARATION**

I hereby declare that the work, which is being presented in the report entitled “**Machine Learning Internship - Zawr Industries LLP”** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering**, is a record of my own investigations carried under the guidance of **Dr. Asad Mohammed Khan, Professor, Presidency School of Computer Science and Engineering, Presidency University, Bengaluru.**

I have not submitted the matter presented in this report anywhere for the award of any other Degree.

|  |  |  |
| --- | --- | --- |
| Name | Roll number | signature |
| ahmed pasha | 20211CSE0358 |  |

# INTERNSHIP COMPLETION CERTIFICATE

****

.

# ABSTRACT

As part of my final semester internship, I am currently working as a Machine Learning Intern at Zawr Industries LLP., a multidisciplinary company specializing in Customer Demographics providing services for multiple sectors in IT fields, Healthcare and Biotechnology, Real Estate Ventures and Business Development Consultation. During the internship, I am gaining hands-on experience in Multiple projects building and training Machine learning models to predict the future trends which will help the company to grow in the near future , where my primary responsibilities involve working extensively with Designing and building ML models as front-end developer for building website to show the graphs for better prediction of real-time. This role allows me to understand the architecture and workflow of Machine Learning applications and Customer Demographics while applying best practices in software engineering. The internship has provided me with valuable exposure to the real- world multidisciplinary environment , strengthened my practical and technical skills with diverse project experience , and offered insight by dealing with real unsolved industry problems. Through this opportunity, I am not only enhancing my knowledge of enterprise technologies but also learning how to effectively collaborate within a professional development team.

# ACKNOWLEDGEMENTS

First, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro- VC - Engineering and Dean, Presidency School of Computer Science and Engineering & Presidency School of Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Dean **Dr. Mydhili Nair,** Presidency School of Computer Science and Engineering, Presidency University, and **Dr. Asif Mohamed H B,** Head of the Department, Presidency School of Computer Science and Engineering, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Mr. Asad Mohammed Khan, Professor** and Reviewer **Dr. Jeyanthi Kameleswaran**, Professor, Presidency School of Computer Science and Engineering, Presidency University for his/her inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the internship work.

We would like to convey our gratitude and heartfelt thanks to the PIP4001 Internship/University Project Coordinator **Mr. Md Ziaur Rahman and Dr. Sampath A K,** department Project Coordinators and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

**Ahmed Pasha**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Table Name** | **Table Caption** | **Page No.** |
| 1 | Table 4.2 | A table comparing LSTM model performance  metrics (e.g., MSE loss) for different hyperparameters | 10 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Figure Name** | **Caption** | **Page No.** |
| 1 | Figure 4.1 | Stock Price Predictor System Architecture | 8 |
| 2 | Figure 5.1 | Internship Timeline | 14 |
| 3 | Figure B1 | Sign-up Page | 25 |
| 4 | Figure B2 | Login Page | 25 |
| 5 | Figure B3 | Forecasting | 26 |
| 6 | Figure B4 | Graph afer Forecasting | 26 |
| 7 | Figure B5 | Forecast in Table Format | 27 |
| 8 | Figure B6 | Mongodb Database(Stock Data and User Data) | 28 |

|  |  |  |
| --- | --- | --- |
| **Chapter**  **No.** | **Title** | **Page**  **No.** |
|  | Certificate | ii |
|  | Declaration | iii |
|  | Internship Certificate | iv |
|  | Abstract | v |
|  | Acknowledgement | vi |
|  | List of Tables | vii |
|  | List of Figures | viii |
| 1 | Introduction | 1 |
|  | 1.1 Multidisciplinary Projects in the Modern Era | 1 |
|  | 1.2 About Zawr Industries LLP | 1 |
|  | 1.3 Handwritten Digit Recogntion | 1 |
|  | 1.4 AI for Predicting Business Failures | 2 |
|  | 1.5 Stock Price Predictor | 2 |
|  | 1.6 Learning Outcomes | 2 |
| 2 | Literature Survey | 4 |
| 2.1 Stock Price Prediction a Using Machine Learning | | 4 |
|  | 2.2 Real-Time Data Integration | 4 |
|  | 2.3 Web-Based Financial Tools | 5 |
|  | 2.4 Relevance to the Project | 5 |
| 3 | Objectives | 6 |
|  | 3.1 Real-Time Financial Data Integration | 6 |

|  |  |  |
| --- | --- | --- |
| **Chapter**  **No.** | **Title** | **Page**  **No.** |
|  | 3.2 Data Preprocessing and Scaling | 6 |
|  | 3.3 Deep Learning for Time-Series Forecasting | 6 |
|  | 3.4 Model Evaluation and Optimization | 6 |
|  | 3.5 Data Visualization | 6 |
|  | 3.6 Web Application Development | 6 |
|  | 3.7 Backend Integration and Database Handling | 6 |
| 4 | Architectural Overview of the Stock Price Predictor | 7 |
|  | 4.1 Stock Price Predictor System Architecture | 7 |
|  | 4.1.1 Client Tier (Front-end) | 7 |
|  | 4.1.2 Application Tier (Backend Processing) | 7 |
|  | 4.1.3 Data Storage Tier (Database) | 7 |
|  | 4.1.4 Integration Layer | 8 |
|  | 4.2 Machine Learning Model Design | 9 |
|  | 4.2.1 Input Preparation | 9 |
|  | 4.2.2 Model Architecture | 9 |
|  | 4.2.3 Training Configuration | 9 |
|  | 4.2.4 Prediction Process | 10 |
| 5 | Timeline for Execution of Internship(Gantt Chart) | 11 |
| 6 | Outcomes | 12 |
|  | 6.1 Practical Industry Exposure | 12 |
|  | 6.2 Technical Skill Development | 12 |
|  | 6.3 Soft Skills and Professionalism | 12 |
|  | 6.4 Understanding of Project Lifecycle | 12 |

|  |  |  |
| --- | --- | --- |
| **Chapter**  **No.** | **Title** | **Page**  **No.** |
| 7 | Results and Discussion | 14 |
|  | 7. 1 Practical Application of Theoretical Knowledge | 14 |
|  | 7.2 Exposure to Modern Software Architecture | 14 |
|  | 7.3 Improved Debugging and Problem-Solving Skills | 14 |
|  | 7.4 Professional Growth and Workplace Readiness | 14 |
| 8 | Conclusion | 15 |
|  | References | 16 |
| A | APPENDIX-A (PSUEDO CODE) | 17 |
|  | A1. Handwritten Digit Recognition using CNN(with image upload and  prediction) | 17-18 |
|  | A2. Business Failures in France (2014-2024) | 19-22 |
|  | A3. Stock Price Forecasting | 22-24 |
| B | APPENDIX-B (SCREENSHOTS) | 25-28 |
| C | APPENDIX-C(ENCLOURES ON SUSTAINABLE DEVELOPMENT  GOALS) | 29-30 |

## Chapter 1 INTRODUCTION

### Multidisciplinary Projects in the Modern Era

In today’s fast-evolving technological landscape, multidisciplinary projects have emerged as the driving force behind innovation and problem-solving. Unlike traditional projects confined to a single domain, multidisciplinary projects bring together expertise from various fields— such as computer science, mechanical engineering, design, finance, healthcare, and artificial intelligence—to address complex, real-world challenges.

### About Zawr Industries LLP.

**Zawr Industries LLP** is a multidisciplinary technology company specializing in developing innovative solutions across domains like smart automation, fintech, logistics, and digital infrastructure. It focuses on combining software, hardware, and data-driven technologies to solve real-world problems. The company engages in product design, IoT systems, and cloud- based applications. Zawr Industries encourages learning through hands-on projects and internships. It is known for offering diverse, industry-relevant exposure to students and professionals. Though not a leading fintech firm, it operates in sectors that intersect with fintech and other modern tech fields.

### Handwritten Digit Recognition

As a part of internship my first project was to building an intelligent system for **handwritten digit recognition** using the popular **MNIST dataset**. It explores and compares the performance of three classification models: **Convolutional Neural Networks (CNN)**, **K- Nearest Neighbors (KNN)**, and **Support Vector Machines (SVM)**. The CNN model is enhanced with data augmentation, dropout, and batch normalization, while the KNN and SVM serve as baseline algorithms. A key highlight of this project is the **real-time digit prediction feature** that allows users to upload images for classification, with added preprocessing techniques for accuracy improvement. This project not only demonstrates core deep learning concepts but also showcases practical problem-solving within the constraints of **Google Colab**, making the implementation both innovative and adaptable.

### AI for Predicting Business Failures

This project focuses on predicting business failures in France using machine learning techniques, based on historical data from 2014 to 2024. The dataset, was taken from Kaggle, which includes the detailed information on business closures across various regions, locations, and capitals. By taking state-wise analysis, feature engineering, and advanced regression models such as XGBoost, the project aims to identify patterns and trends that lead to business failures. The final model helps forecast future risks, providing valuable insights for policymakers and investors to make informed decisions.

### Stock Price Predictor

This was given to me as the final project in my internship .The Stock Price Predictor which is a web-based application developed using Python and Streamlit were I have applied deep learning techniques to forecast future stock prices. By integrating real-time financial data from Tiingo and utilizing LSTM neural networks, the system will predict the highest stock prices for up to 30 future days. By providing interactive visualizations and stores predictions in a MongoDB database for future reference. This tool aids investors and analysts in making informed decisions based on data-driven forecasts.

### Learning Outcomes

The internship has provided me with valuable exposure to the real-world multidisciplinary environment ,the development of the Stock Price Predictor during my internship at Zawr Industries LLP provided significant learning outcomes. During this project I gained hands-on experience in Python programming especially in using libraries like Pandas , Numpy and TensorFlow for data analysis and machine learning tasks. I developed good understanding of deep learning techniques more in LSTM neural networks ,time-series forecasting and on development side , I built interactive web applications using Streamlit and handles backend operations using MongoDB for storing and retrieving data and I did many other projects in my internship first project was to check my ability on how I predict our next customer to the store using retail dataset. As I completed I slowly gained some view on machine learning models on predictive analysis and later the other project given to me was Handwritten Digit Recognition here I got to know how to take an in-built dataset using keras model and using other such ML models like SVM,KNN and CNN for better accuracy and to check the nearest

digit I made it unique by applying user based input that he can take the digit picture from the internet and save it in his device from which I applied the logic of probability and confidence level that it will give the accurate digit by analysing the image. And later the other projects which I had done individually give me an experience and alos sharpened my problem-solving abilities especially when it came to debugging training issues and improving the model’s prediction accuracy. Collaborating with other developers at Zawr Industries improved my Communication skills, giving me valuable exposure to real world fintech challenges.

# Chapter 2

# LITERATURE SURVEY

The task of stock price prediction has progressed considerably in recent years, thanks to the advancements in machine learning and data processing. This review examines key studies and approaches relevant to the Stock Price Predictor project developed during my internship at Zawr Industries LLP, using LSTM neural networks and integrating real-time financial data , and web based tools for end users.

### Stock Price Prediction Using Machine Learning

Historically, stock price forecasting relied on statistical models like ARIMA in 1970 and valued for their structural approach .However, these linear models often fall short when dealing with the highly non-linear and volatile nature of financial time-series data .To overcome this, recent research has turned towards machine learning particularly neural networks. The introduction of LSTM networks by Hochreiter and Schmidhuber in 1997 marked a turning point. LSTMs a specialized type of Recurrent Neural Network (RNN) which are capable of capturing long-term dependencies in sequential data by using memory cells and gating mechanisms. These capabilities have made them very useful for time-series tasks like stock price prediction.

### Real-Time Data Integration

Access to real-time market data has become essential for building robust forecasting models.By widely using APIs such as Yahoo Finance which offers live data, they often have limitations in terms of reliability and granuality. This has prompted a shift toward premium services like Tiingo known for its high-quality , real time financial datasets. In this project the Tiingo API was integrated through pandas allowing seamless retrieval of live stock data .This data was then preprocessed using the MinMaxScaler technique ensuring it was in a suitable format for training the LSTM model. This approach is supported by research from Patel which found that data normalization improves the model convergence and prediction accuracy for more real time results after the forecasting of the stock prices.

**2.3 Web-Based Financial Tools**

In the fintech space, delivering machine learning insights through user friendly web applications is becoming increasingly important. Streamlit a lightweight Python framework, has emerged as a popular tool for building such interactive dashboards, Its ease of use and responsiveness have led to its adoption in various data science projects emphasize how web based platforms help make complex financial models more accessible to non technical users The Stock Price Predictor project reflects this trend ,featuring a Streamlit interface with user controls such as dropdowns, sliders and real time Plotly graphs for visualizing predictions. Whereas backend storage MongoDB was used to manage user inputs and the forecast data it aligns with the modern NoSQL approaches for handling unstructured data for more scalability and flexibility in web based systems.

## Chapter 3 OBJECTIVES

This project aims to build an intelligent stock price prediction system using LSTM models for time series forecasting .It integrates real time data APIs and offers a user friendly web interface for easy access to predictions.

### 3.1.Real-Time Financial Data Integration

The Tiingo API is used to fetch historical and real time stock data automatically and it ensures the model stays updated with the current market trends for accurate predictions.

### Data Preprocessing & Scaling

Pandas and NumPy handle large datasets efficiently while maintaining chronological structure and MinMaxScaler is used for normalization and missing values are addressed to support LSTM learning.

### Deep Learning for Time-Series Forecasting

An LSTM model built with Tensorflow captures sequential patterns in stock price data.The architecture is trained, validated and tuned for optimal forecasting performance.

### Model Evaluation & Optimization

Model accuracy is evaluated using MAE, RMSE, and MSE metrics. Hyperparameter tuning and debugging are performed to address issues like overfitting .

### Data Visualization

Plotly is used to create interactive charts comparing actual vs predicted stock prices. These visualizations help users understand trends through dynamic, real-time graphs.

### Web Application Development

I had developed responsive web interface using Streamlit for ease of use. Users can select stocks, view predictions, and interact with visual insights.

### Backend Integration & Database Handling

In the backend MongoDB stores user inputs, historical data and prediction results efficiently. APIs ensure smooth data retrieval and updates between frontend and backend systems.

**Chapter 4**

# ARCHITECTURAL OVERVIEW OF THE STOCK PRICE PREDICTOR

### Stock Price Predictor System Architecture

The Stock Price Predictor is architected as a modular, web-based application that integrates front-end, backend, and machine learning components to deliver stock price forecasts. The system follows a client-server model, with the following architectural layers:

### Client Tier (Front-End):

* + - * Built using Streamlit, a Python-based framework for creating interactive web applications.
      * Provides a user interface with dropdown menus for selecting ticker symbols (e.g., from NASDAQ), a slider for specifying forecast days (1–30), and a "Forecast" button to initiate predictions.
      * Displays results through Plotly visualizations, including line charts, area charts, and an indicator gauge comparing the predicted high price to the last known value.
      * The front-end is styled with custom CSS for enhanced user experience, ensuring accessibility and responsiveness.

### Application Tier (Backend Processing):

* + - * Handles data retrieval, preprocessing, model training, and prediction generation.
      * Data Retrieval:Utilizes the Tiingo API to fetch historical stock data for a selected ticker symbol, including features like open, high, low, close, and volume.
      * Data Preprocessing: Employs Pandas for data manipulation and MinMaxScaler from Scikit-learn to normalize data into a [0, 1] range, preparing it for LSTM training.
      * Machine Learning Model: Implements a Sequential LSTM model using TensorFlow/Keras, consisting of two LSTM layers, a Dropout layer , and a Dense output layer. The model is trained on 100 timesteps of historical data to predict the next day's high price, with 10 epochs and a batch size of 65.
      * Prediction Generation: Iteratively generates 30-day forecasts by feeding the model's output back into the input sequence, updating the input window for each prediction.

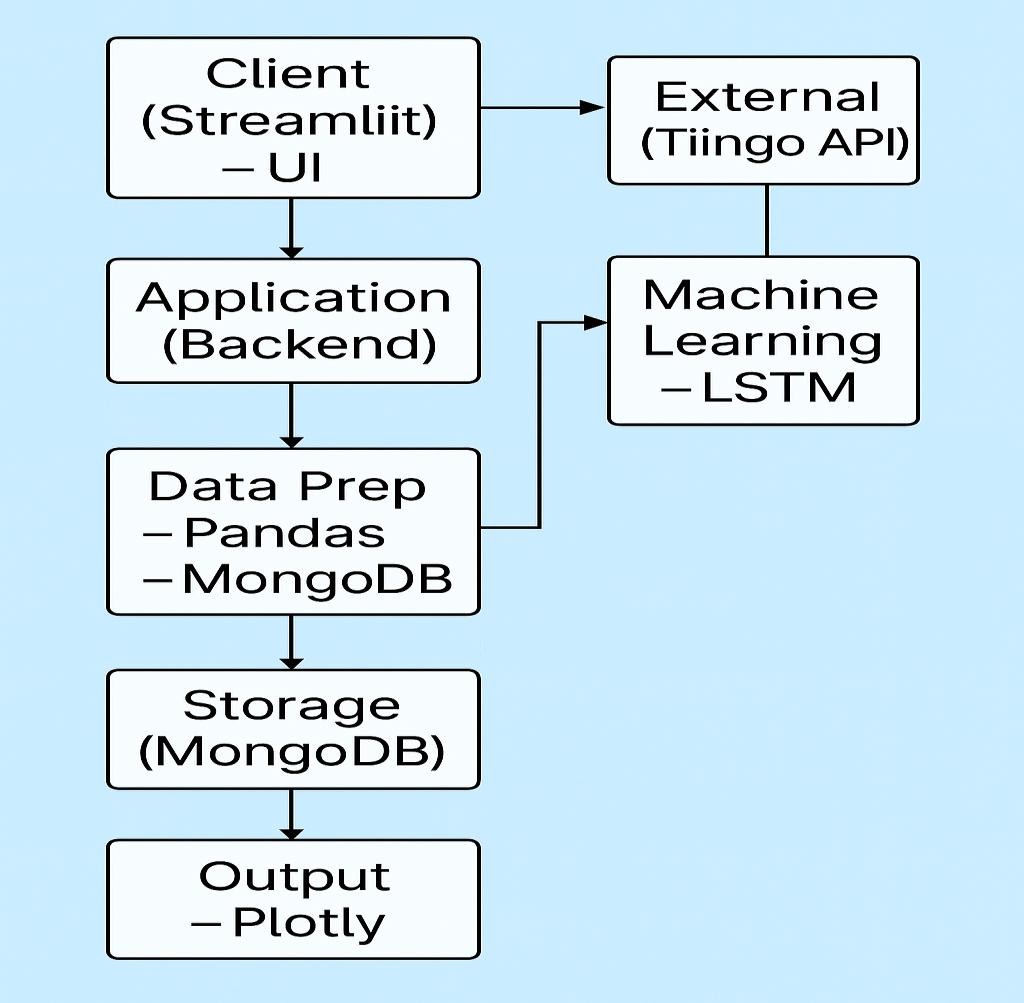
### Data Storage Tier (Database):

* + - * Uses MongoDB, a NoSQL database, to store prediction results and metadata.
      * Stores documents containing the ticker symbol, last known date, last high value, and a list of forecasted prices for future reference.
      * Connects to a MongoDB Atlas cluster using the PyMongo library, ensuring scalability and reliability.
      * If predictions for a ticker exist in the database, the system retrieves them to avoid redundant model training, improving efficiency.

### Integration Layer:

* + - * Facilitates communication between the front-end, backend, and external services.
      * The Tiingo API is accessed using the `pandas\_datareader` library with an API key for real- time data.
      * MongoDB interactions are managed via PyMongo, enabling seamless data storage and retrieval.
      * Plotly Express and Graph Objects are used to render interactive visualizations, integrated into the Streamlit interface.

This architecture ensures modularity, scalability, and user-friendliness. The loosely coupled design allows independent updates to the front-end, backend, or machine learning components. The use of MongoDB enables persistent storage, while the LSTM model provides robust time-series forecasting. The system is extensible, supporting additional ticker symbols or forecasting features with minimal changes.



**Figure 4.1 Stock Price Predictor System Architecture**

### Machine Learning Model Design

The core of the Stock Price Predictor is its LSTM-based machine learning model, designed for time-series forecasting of stock prices. The model architecture is as follows:

### Input Preparation:

* + - * Historical stock data is fetched from Tiingo and processed using Pandas.
      * The data is normalized using MinMaxScaler to scale features to [0, 1], ensuring stable model training.
      * The dataset is structured into sequences of 100 timesteps (n\_past = 100), with each sequence containing 10 features (data.shape[1]). The target is the high price of the next day.

### Model Architecture:

* + - * LSTM Layer 1:64 units, ReLU activation, returns sequences to feed into the next layer. This layer captures long-term dependencies in the time-series data.
      * LSTM Layer 2:32 units, ReLU activation, does not return sequences, summarizing the learned features.
      * Dropout Layer:0.2 dropout rate to prevent overfitting by randomly deactivating 20% of neurons during training.
      * Dense Layer:Outputs a single value, representing the predicted high price for the next day.
      * Input shape: (100, 10), where 100 is the timestep length and 10 is the number of features.

### Training Configuration:

* + - * Optimizer: Adam, known for efficient gradient descent in deep learning.
      * Loss Function: Mean Squared Error (MSE), suitable for regression tasks like price prediction.
      * Training Parameters: 10 epochs, batch size of 65, 10% validation split to monitor performance.
      * The model is trained on the prepared dataset, with progress displayed via a Streamlit spinner indicating a 2–3 minute training time.

### Prediction Process:

* + - * The last 100 days of scaled data are reshaped into (1, 100, 10) for model input.
      * The model predicts the next day's high price, which is appended to the predictions list.
      * The input sequence is rolled forward, incorporating the predicted value, and the process repeats for 30 days.
      * Predictions are inverse-transformed using MinMaxScaler to convert them back to the original price scale.

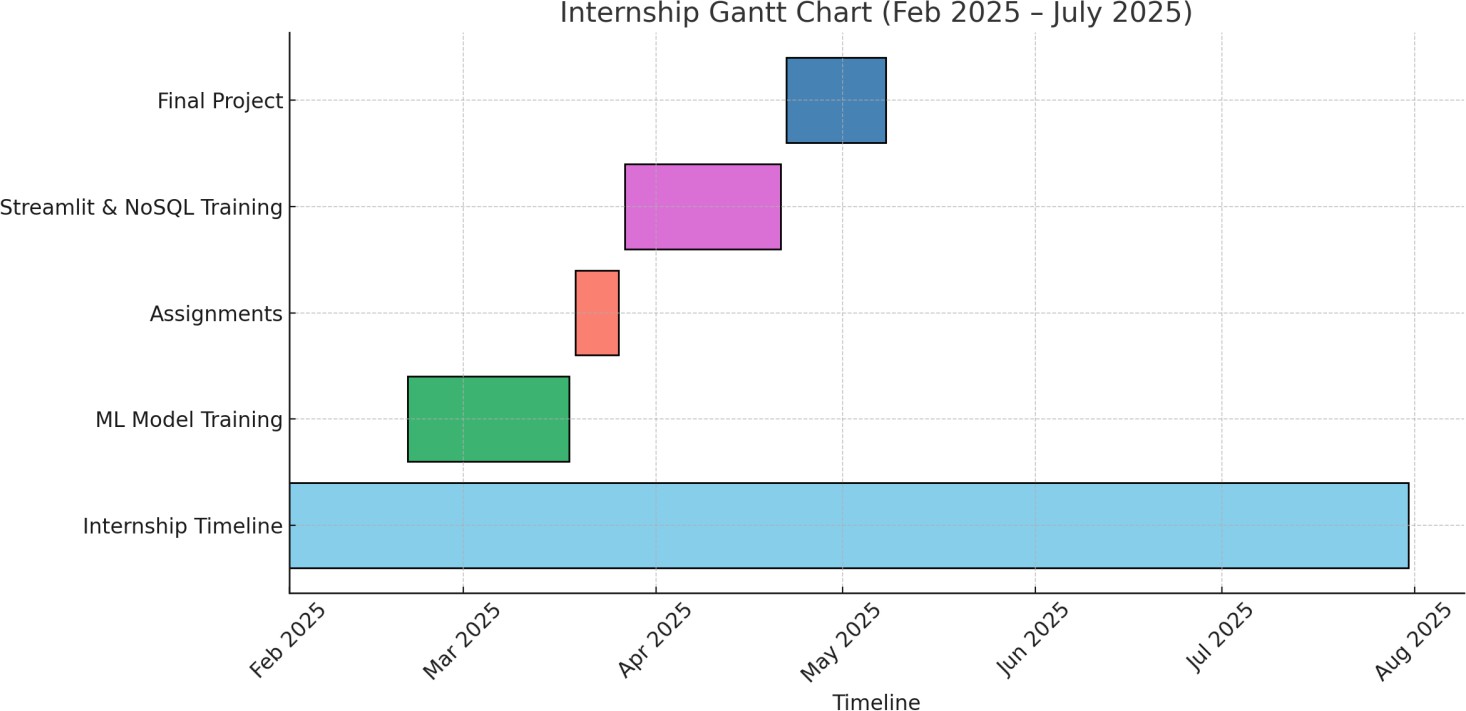
This design leverages LSTM's ability to model temporal dependencies, making it ideal for stock price forecasting. The use of dropout and validation split enhances model generalization, while the iterative prediction approach enables multi-day forecasts.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model No**. | **Epochs** | **Batch Size** | **Learning Rate** | **Units(LSTM)** | **Dropout** | **MSE loss** |
| **1** | **50** | **32** | **0.001** | **50** | **0.2** | **0.00128** |
| **2** | **100** | **64** | **0.001** | **100** | **0.3** | **0.00194** |
| **3** | **100** | **32** | **0.0005** | **100** | **0.3** | **0.00176** |
| **4** | **150** | **32** | **0.0001** | **128** | **0.4** | **0.00163** |
| **5** | **200** | **16** | **0.0001** | **150** | **0.5** | **0.00147** |

**Table 4.2: A table comparing LSTM model performance metrics (e.g., MSE loss) for different hyperparameters**

# Chapter 5

# TIMELINE FOR EXECUTION OF INTERNSHIP (GANTT CHART)

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Name** | **Start Date** | **End Date** | **Duration** |
| 1 | Internship Timeline | Feb 10, 2025 | July 30, 2025 | 171 days |
| 2 | Machine Learning Models Training | Feb 20, 2025 | March 18, 2025 | 26 days |
| 3 | Assignments | March 19, 2025 | March 26, 2025 | 8 days |
| 4 | Streamlit and NoSQL Training | March 27, 2025 | April 21, 2025 | 26 days |
| 5 | Final Project | April 22,2025 | May 1,2025 | 10 days |

**Fig 5.1 Internship Timeline**

# Chapter 6

# OUTCOMES

### Practical Industry Exposure

These projects offered me real-world experience in fintech by inetgrating live stock data , predicting new digits and by enhancing the clarity by removing the noise from the images. And other such projects using machine learning. It gave me a productive-like environment with API usage, NoSQL databases and UI development challenges.

### Technical Skill Development

I strengthened my skills in Python, TensorFlow, Pandas and NumPy for data science and LSTM modeling. Additionally, I worked with Streamlit, Plotly, PyMongo, and APIs enhancing full-stack development expertise.

### Soft Skills and Professionalism

I improved time management and communication through regular updates and teamwork with Zawr Industries. Presenting insights cleary and following daily practices refined my professionalism towards my colleagues and adaptability through working on my projects.

### Understanding of Project Lifecycle

The Project exposed me to the entire SDLC-from gathering requirements to development, testing, and deployment

Working in modular components taught me how to align with agile workflows and deliver functional solutions. .

**Chapter 7**

# RESULTS AND DISCUSSIONS

### Practical Application of Theoretical Knowledge

Through this project, I applied LSTM neural networks to real-time stock price forecasting using financial data. Concepts like data normalization, sequential modeling and modular design were put into direct use in building the project.

### Exposure to Modern Software Architecture

The system architecture used a client server approach with Streamlit, TensorFlow and MongoDB. I worked with RESTful APIs and NoSQL databases to build an extensible and scalable prediction app. This hands-on exposure deepened my understanding of how the fintech systems are structured in practice.

### Improved Debugging and Problem-Solving Skills

I debugged LSTM model issues such as data shape mismatches and model convergence in predicting and forecasting the actual real time stock price. Database connection erros and API integration bugs were resolved through testing and error handling. These tasks strengthened my ability to troubleshoot problems and understand the complexity of the errors.

### Professional Growth and Workplace Readiness

Completing this project with Zawr Industries helped me adapt to real development timelines and responsibilities to complete the work in the given timeline and the projects. I also worked in a team in the beginning stage of my internship improved my soft skills by communicating with them. It prepared me to contribute confidently to fintech projects in a professional environment.

## Chapter 8 CONCLUSION

The on-going six-month internship at Zawr Industries has been a valuable learning experience, It helped me a move from textbook knowledge and understand how real-world problems are solved using tech. As a fresher, I gained solid foundation in applying algorithms and ML models to solve practical challenges. The Stock Price Predictor project was a cornerstone of my internship at Zawr Industries LLP, encapsulating the technical and professional skills gained over the six-month period. By developing a web-based application that leverages LSTM neural networks, real-time financial data, and MongoDB, I contributed to a practical fintech solution for stock price forecasting. The project enhanced my proficiency in Python, deep learning, web development, and database management, while exposing me to modern software architecture and API-driven workflows. Professionally, I developed skills in collaboration, time management, and agile development, preparing me for a career in fintech. This internship bridged the gap between academic knowledge and industry applications, equipping me with the confidence and expertise to tackle enterprise-level challenges in digital finance.

# REFERENCES

1. Tiingo Financial Market Data API: ([https://api.tiingo.com](https://api.tiingo.com/))
2. Yahoo Finance API & yfinance Library: ([*https://github.com/ranaroussi/yfinance*](https://github.com/ranaroussi/yfinance)*)*
3. LSTM for Stock Price Prediction — Research Paper: (<https://ieeexplore.ieee.org/document/8614252>)
4. Time Series Forecasting using TensorFlow: (<https://www.tensorflow.org/tutorials/structured_data/time_series>)
5. Plotly for Financial Charting: (<https://plotly.com/python/)>
6. Kaggle Notebook — Handwritten Digit Recognition using CNN, KNN, and SVM: ([https://www.kaggle.com/code/spscientist/mnist-handwritten-digit-recognition-cnn-knn-](https://www.kaggle.com/code/spscientist/mnist-handwritten-digit-recognition-cnn-knn-svm) [svm](https://www.kaggle.com/code/spscientist/mnist-handwritten-digit-recognition-cnn-knn-svm))
7. Kaggle Dataset — Prediction of Business Failures in France: (<https://www.kaggle.com/datasets/fedesoriano/company-bankruptcy-prediction>)

# APPENDIX-A PSUEDOCODE

### A1. Handwritten Digit Recognition using CNN (with image upload and prediction)

**Step 1: Load and Preprocess Dataset**

-Load MNIST dataset (images and labels)

-Normalize pixel values to range [0, 1]

-Reshape images to 28x28x1 for CNN input

### Step 2: Data Augmentation

Initialize ImageDataGenerator with random transformations:

- rotation, zoom, shift

Fit the generator on training images

### Step 3: Build and Train CNN Model

Create Sequential CNN model:

* Conv2D + ReLU + BatchNorm + MaxPool
* Conv2D + ReLU + BatchNorm + MaxPool
* Conv2D + ReLU + BatchNorm
* Flatten
* Dense (ReLU) + Dropout
* Output Dense layer (Softmax)

Compile model with:

* Adam optimizer
* Sparse categorical cross-entropy loss
* Accuracy as metric

Train model on augmented data:

* Use generator flow
* Validate using test data Save trained model to file

### Step 4: Upload and Load Image

-Prompt user to upload image

-Get image file path

-Load previously saved model from file

### Step 5: Preprocess Uploaded Image

Function to preprocess image:

* Convert to grayscale (L mode)
* Resize to 28x28
* Apply Gaussian blur
* Apply binary threshold (invert image)
* Normalize to [0, 1]
* Reshape to CNN input shape (1, 28, 28, 1)

### Step 6: Predict Digit

Feed preprocessed image to model Get predicted probabilities

Extract highest confidence score Get predicted digit (argmax)

If confidence < 0.5:

* Choose second-best prediction

### Step 7: Display Results

Plot the uploaded image with:

* Predicted digit and confidence score in title

Plot a bar chart showing:

* Prediction probabilities for digits 0 to 9

## A2. Business Failures in France (2014-2024)

### Step 1: Load Data

-Load CSV data file into DataFrame (df)

**Step 2: Data Exploration & Visualization** Display dataset info and check for missing values Plot line graph: Business Failures over Years

Group by:

* Region → Sum of FailureUnit
* Location → Sum of FailureUnit
* Capital → Sum of FailureUnit

Plot bar charts for top 10:

* Regions with most failures
* Locations with most failures
* Capitals with most failures

Print top 10 tables for regions, locations, and capitals

### Step 3: Data Preprocessing

For each categorical column: Encode using LabelEncoder Store the encoder for future use

Split DataFrame into:

* Features (X)
* Target (y = FailureUnit)

Split into training and testing sets (80/20 split) Scale features using StandardScaler

### Step 4: Model Training & Initial Evaluation

Initialize models:

* Random Forest Regressor
* XGBoost Regressor

For each model:

Train on training data Predict on test data Evaluate using:

* + MAE
  + MSE
  + R2 Score

Store performance results

### Step 5: Hyperparameter Tuning (XGBoost)

Define parameter grid for XGBoost:

* n\_estimators
* learning\_rate
* max\_depth

Perform GridSearchCV with R2 scoring

Train best XGBoost model from grid search Predict and evaluate best model:

* MAE
* MSE
* R2 Score

Add results to comparison table

### Step 6: Visualizing Model Performance

Plot bar chart of R2 Score comparison for all models including Tuned XGBoost

### Step 7: Analyze Failure Trends by Capital

Plot line chart: Business Failures over Time (by Capital)

Print unique combinations of Capital, Location, and ZoneCategory

### Step 8: Predict Future Failures

Use best XGBoost model to predict failures on X\_test Add predictions and Capital names to X\_test

Sort by predicted failure count (descending)

Print top 10 capitals predicted to undergo the most failures

## A3. Stock Price Forecasting

START

IMPORT necessary libraries for:

* Data processing (pandas, numpy)
* Visualization (plotly)
* Machine Learning (keras, sklearn)
* Data source (pandas\_datareader)
* Web interface (streamlit)
* Database (MongoDB via pymongo)

DEFINE API key for Tiingo

LOAD NASDAQ ticker symbols from CSV

SET Streamlit page configurations

DISPLAY main title: "Stock Price Predictor ⬛#/"

IN SIDEBAR:

SELECT between 'Signup', 'Login', 'Forecast'

IF user selects 'Forecast': DISPLAY input columns:

* Ticker symbol selection
* Forecast days (1 to 30)
* Forecast button

CONNECT to MongoDB cluster

ON button click:

FETCH model from database using selected ticker symbol

IF model/data not found in database:

FETCH latest stock data from Tiingo using API CLEAN and preprocess data

SCALE data using MinMaxScaler CREATE input sequences for LSTM model BUILD LSTM model with:

* LSTM layers
* Dropout
* Dense layer

COMPILE model and train with data

MAKE predictions iteratively for 30 days using sliding window INVERSE SCALE predictions

STORE prediction in MongoDB ELSE:

LOAD stored predictions

GENERATE future dates starting from last known date DISPLAY:

* Line plot of predictions
* Area plot of predictions
* Number+Delta indicator comparing to last day value
* Dataframe of predictions
* Link to Yahoo Finance page

ELSE IF user selects 'Signup':

DISPLAY signup form (username, password, confirm password, email) IF passwords match:

CONNECT to MongoDB INSERT new user details

SHOW success message ELSE:

SHOW password mismatch error

ELSE IF user selects 'Login':

DISPLAY login form (username, password)

// (Login logic not fully shown in code) END

# APPENDIX-B SCREENSHOTS

**STOCK PRICE FORECASTING**

****

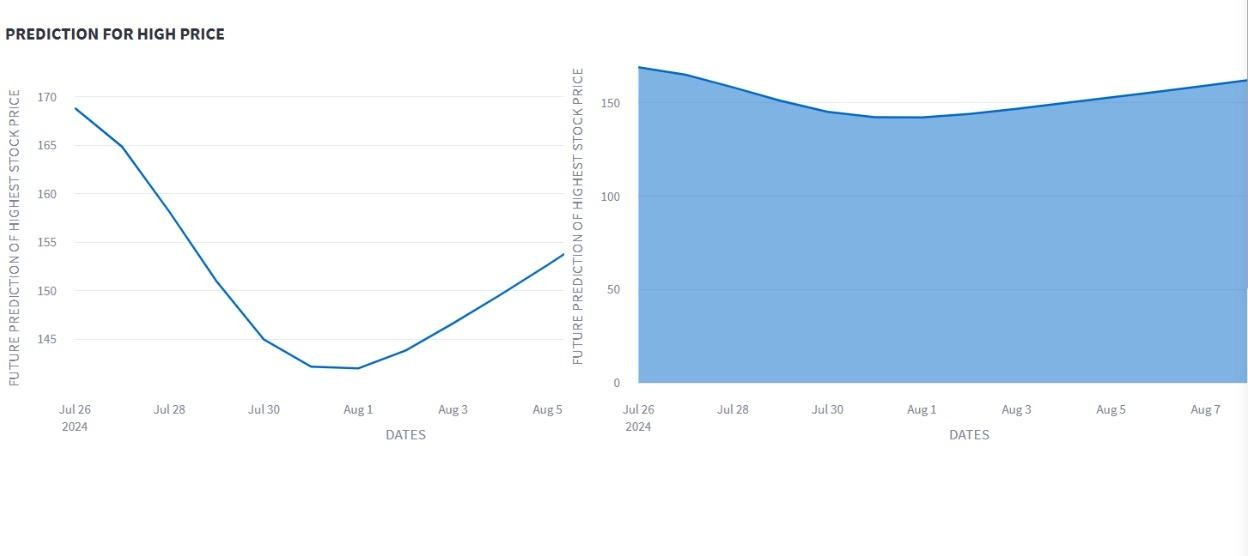
**Fig B1: Sign-up page**

****

**Fig B2: Login page**



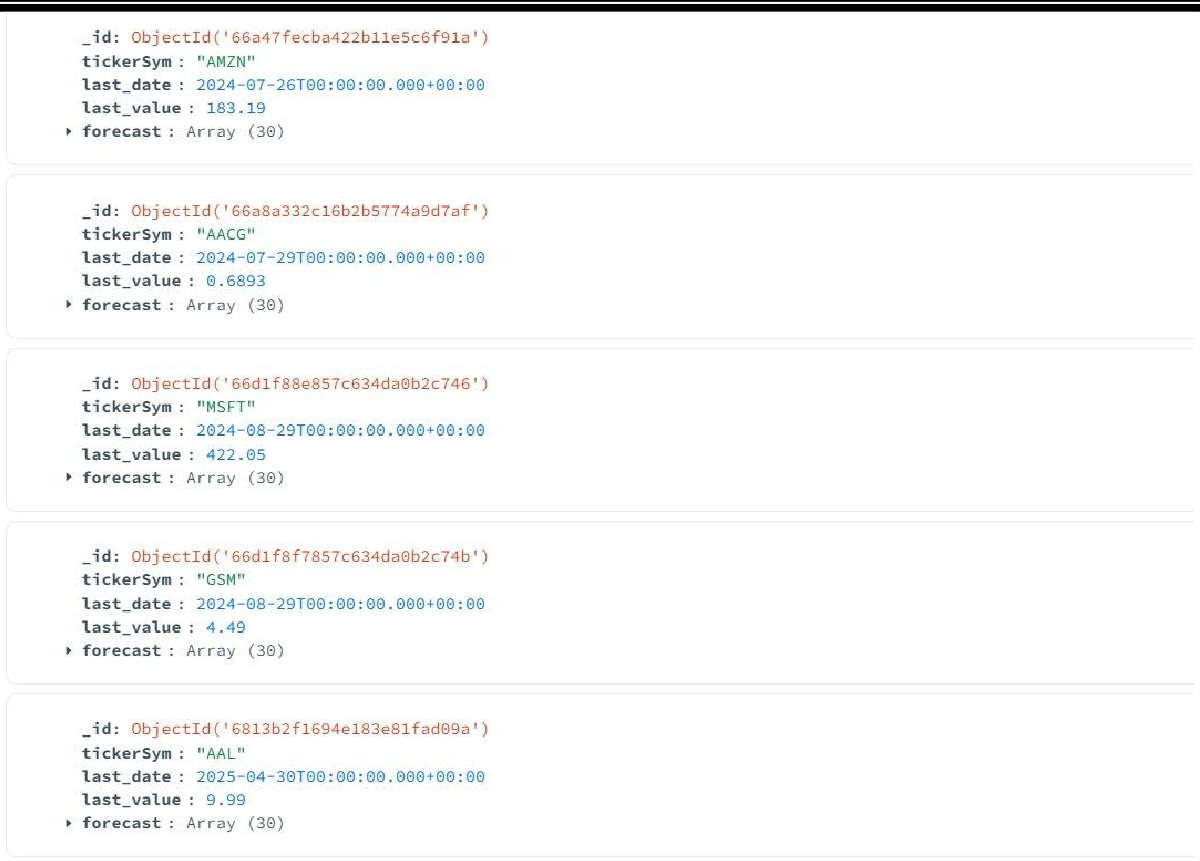
**Fig B3: Forecasting**

****

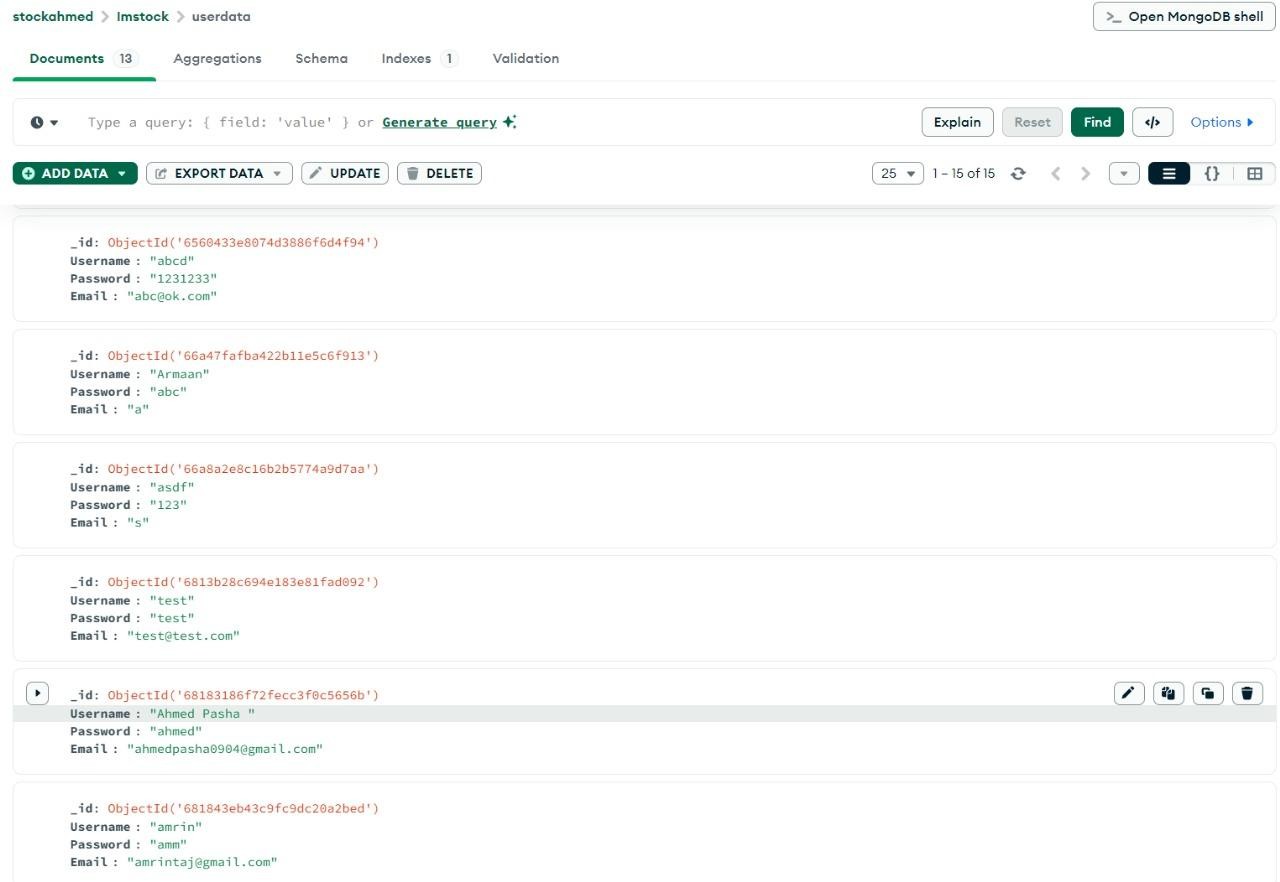
**Fig B4: Graph after forecasting**



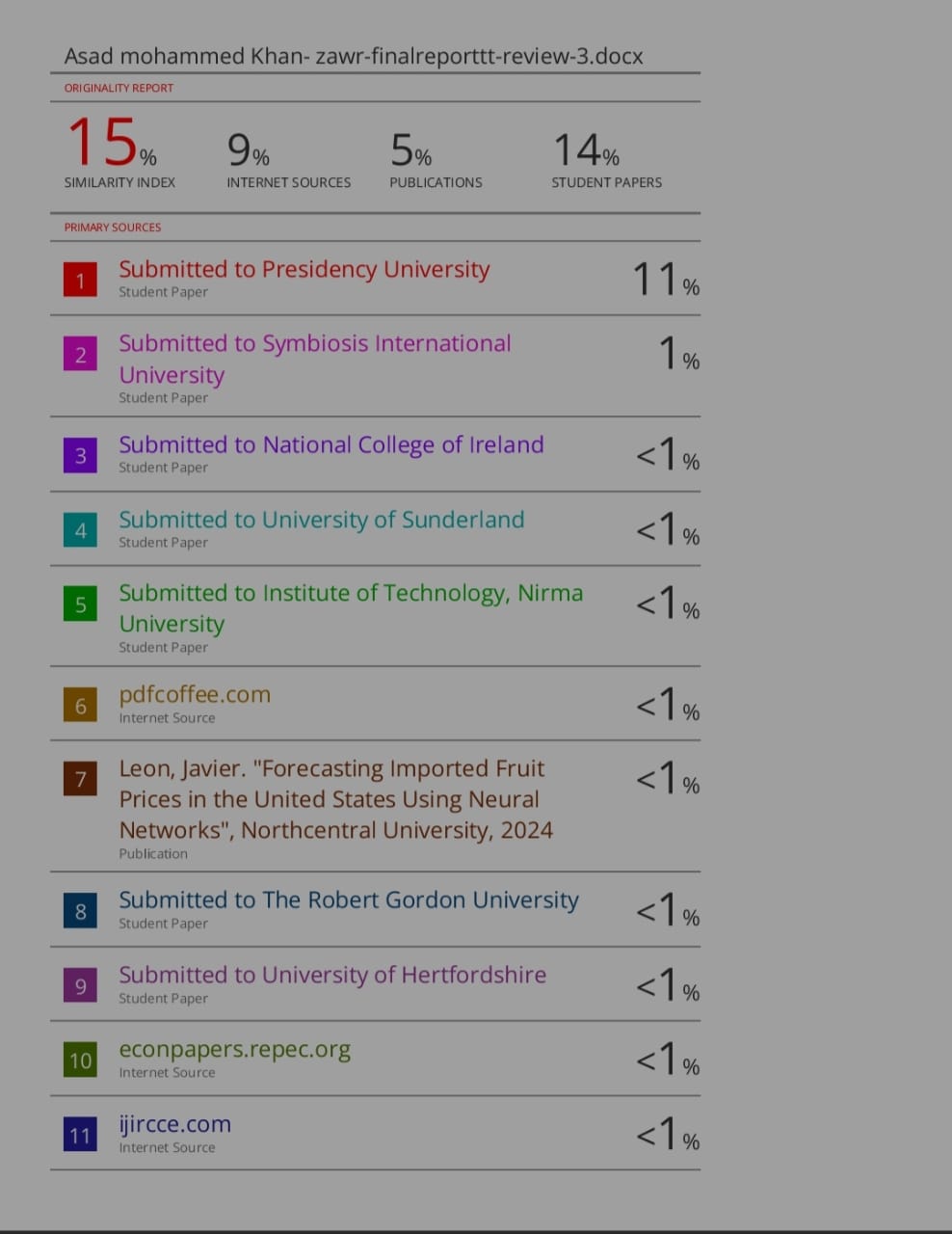
**Fig B5: Forecast in Table Format**



**Fig B6: Mongodb database(Stock Data and User Data)**

****

# APPENDIX-C PLAGIARISM REPORT

****

**APPENDIX-D ENCLOSURES**

## Sustainable Development Goals:

****

**C1. Goal 1 – No Poverty**

Accurate stock predictions can help investors (especially small investors) make informed financial decisions, increasing income and reducing financial loss.Promotes financial literacy and inclusion for individuals trying to build wealth or escape poverty through investments.

**C2. Goal 4 – Quality Education**

Your web app can include **educational tools** such as explanations of predictions, visualizations, or stock market learning resources.Empowers users to learn about finance, machine learning, and economics while using the tool.

**C3. Goal 8 – Decent Work and Economic Growth**

Encourages responsible investment, entrepreneurship, and participation in the financial markets.Promotes innovation and informed investment strategies, which contribute to economic growth.

**C4. Goal 9 – Industry, Innovation, and Infrastructure**

Developing this app involves AI , machine learning , data analysis and ccloud infrastructure . Supports digital innovation in fintech and contributes to smarter economic infrastructures.

**C5. Goal 10 – Reduced Inequalities**

By making stock prediction tools accessibe to the public not just financial elites. Reduces the information and opportunity gap in investment access.

**C6. Goal 17 – Partnerships for the Goals**

Your app could integrate with financial education platforms, stock market APIs, or government financial literacy programs.Builds strong partnerships in the fintech ecosystem and beyond.