Data Science

A PROJECT REPORT

Submitted by

RAVAL MEETKUMAR RITESHKUMAR

200090107072

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

Computer Engineering

C. K. Pithawala College of Engineering and Technology, Surat

Gujarat Technological University, Ahmadabad

May 2024

C. K. Pithawala College of Engineering and Technology

Opposite Surat Airport, Behind DPS School, Near Malvan Mandir, Dumas Road, Surat

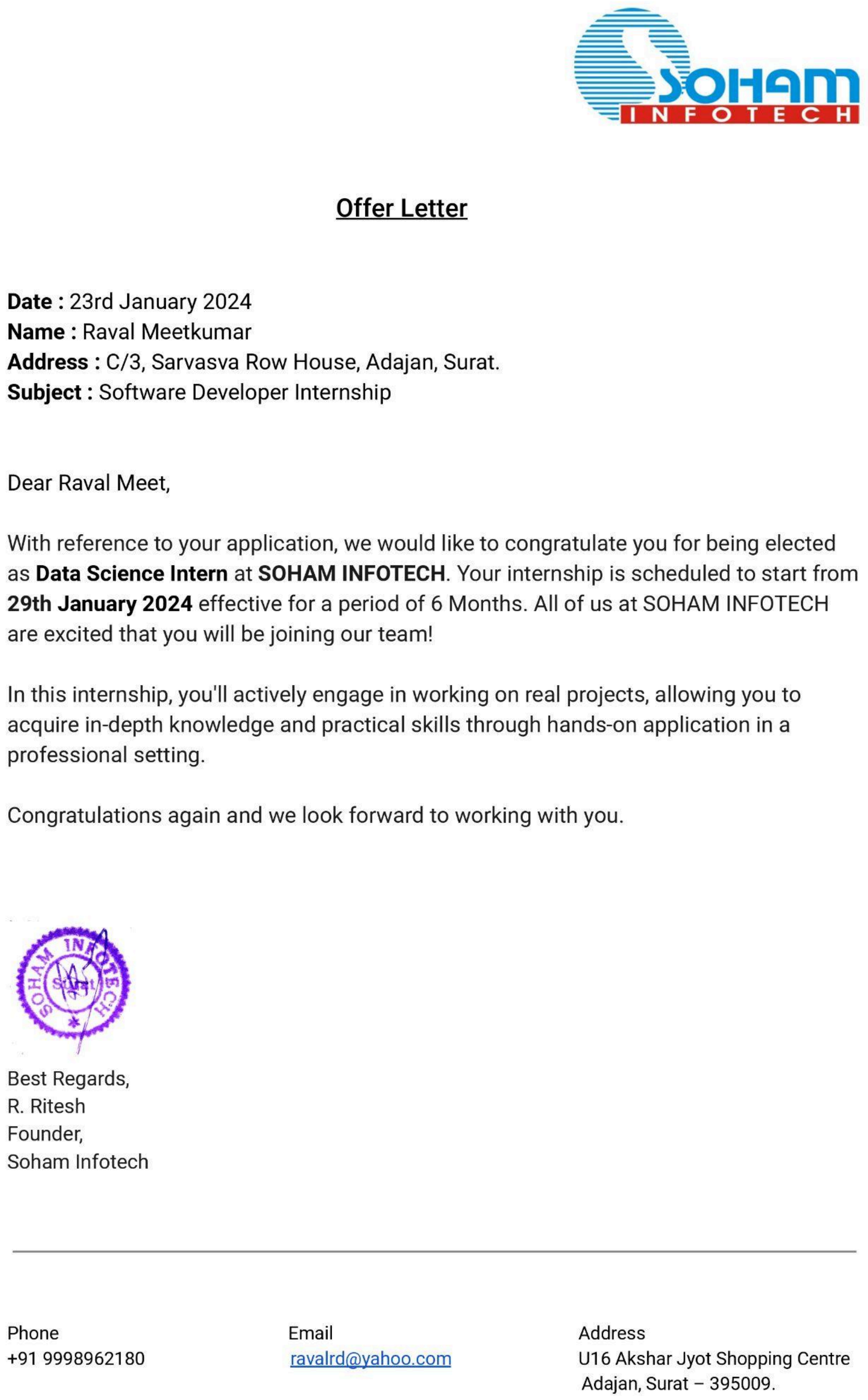
CERTIFICATE

This is to certify that the project report submitted along with the project entitled ***Data Science*** *has been carried out by* ***Raval Meetkumar Riteshkumar.*** *under my guidance in partial fulfillment for the degree of Bachelor of Engineering in Computer Engineering, 8th Semester of Gujarat Technological University, Ahmedabad during the academic year 2023-24.*

Dr. Vishruti Desai
Internal Guide

Prof. Saurabh S. Tandel
Head of the Department

OFFER LETTER



SOHAM INFOTECH

Offer Letter

Date : 23rd January 2024
Name : Raval Meetkumar
Address : C/3, Sarvasva Row House, Adajan, Surat.
Subject : Software Developer Internship

Dear Raval Meet,

With reference to your application, we would like to congratulate you for being elected as **Data Science Intern** at **SOHAM INFOTECH**. Your internship is scheduled to start from **29th January 2024** effective for a period of 6 Months. All of us at SOHAM INFOTECH are excited that you will be joining our team!

In this internship, you'll actively engage in working on real projects, allowing you to acquire in-depth knowledge and practical skills through hands-on application in a professional setting.

Congratulations again and we look forward to working with you.

SOHAM INFOTECH Surat

Best Regards,
R. Ritesh
Founder,
Soham Infotech

Phone	Email	Address
+91 9998962180	ravalrd@yahoo.com	U16 Akshar Jyot Shopping Centre Adajan, Surat – 395009.

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2023-2024

Date of certificate generation : 12 May 2024 (17:58:36)

This is to certify that, ***Raval Meetkumar Riteshkumar*** (Enrolment Number - 200090107072) working on project entitled with ***DATA SCIENCE INTERNSHIP*** from ***Computer Engineering*** department of ***C. K. PITHAWALLA COLLEGE OF ENGINEERING & TECHNOLOGY, SURAT*** had submitted following details at online project portal.

Internship Project Report	**Completed**

Name of Student : Raval Meetkumar Riteshkumar

Name of Guide : Ms.DESAI VISHRUTI AMULKUMAR

Signature of Student : ________________

*Signature of Guide : ____________________

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

C. K. Pithawala College of Engineering and Technology

Opposite Surat Airport, Behind DPS School, Near Malvan Mandir, Dumas Road, Surat

DECLARATION

I hereby declare that the Internship report submitted along with the Internship entitled **Data Science Internship** submitted in partial fulfillment for the degree of Bachelor of Engineering in Computer Engineering to Gujarat Technological University, Ahmedabad, is a bonafide record of original project work carried out by me at **Soham Infotech** under the supervision of **Ms. Ria Raval** and that no part of this report has been directly copied from any students' reports or taken from any other source, without providing due reference.

Raval Meetkumar R.
200090107072

Acknowledgement

I would like to express my heartfelt gratitude to Soham Infotech for providing me with the invaluable opportunity to undertake a Data Science internship. This experience has been instrumental in shaping my understanding and expertise in the field of artificial intelligence.

I extend my deepest appreciation to Ms. Ria Raval for his exceptional mentorship throughout the duration of my internship. His guidance, encouragement, and expertise have been pivotal in my learning journey, enabling me to tackle challenges and explore new horizons in data science.

I take this opportunity to express my heartfelt gratitude to Dr. Vishruti Desai for her invaluable guidance and mentorship throughout the development of this project. Her expertise, feedback, and unwavering support have been instrumental in shaping the direction of my work.

Lastly, I am grateful to all those who have contributed directly or indirectly to this enriching experience, enabling me to acquire practical skills and insights that will undoubtedly benefit my future endeavors in the field of data science.

Thank you once again for this incredible opportunity.

Raval Meetkumar R.

Abstract

This project report encapsulates the culmination of my internship experience focusing on the development of a sophisticated trading platform integrating artificial intelligence and mathematical models for financial instrument trading. Throughout the internship, I encountered and overcame various challenges, including understanding complex financial concepts, navigating financial risk with AI, addressing AI limitations, and applying data science methodologies in the financial domain. The project aimed to leverage cutting-edge technology to optimize trading strategies and maximize profits while managing risks effectively. By collaborating with domain experts, data scientists, and software developers, I contributed to the creation of a robust and scalable trading platform. This report provides a comprehensive overview of the project, detailing its objectives, methodology, implementation, and outcomes, highlighting the significance of technology integration in shaping the future of financial markets.

List of Figures

Table of Contents

I Overview of the Company

1.1 History

Soham Infotech, located in Surat, India, goes beyond simply supplying hardware, networking solutions, and software to industrial clients. Their team leverages their combined expertise to create customized IT ecosystems that optimize and streamline operations. By delving into research for various technological advancements, Soham Infotech ensures they offer the most relevant and future-proof solutions for their clients' unique needs. This dedication to innovation and integration has helped businesses like ours achieve significant improvements in efficiency and productivity.

Figure 1 History of Company

i. Different Services :

Soham Infotech, serves as a comprehensive IT partner for industrial clients. Their core business lies in providing management services and products that optimize and maintain the technological infrastructure of various industries. This includes providing essential hardware, networking solutions, and custom software programs tailored to each client's specific needs. Soham Infotech's expertise extends to both domestic and international companies, fostering strong bonds through their reliable supply and maintenance of these critical IT systems.

1.3 Organization chart.

Organization chart of the company as shown in fig 1.2

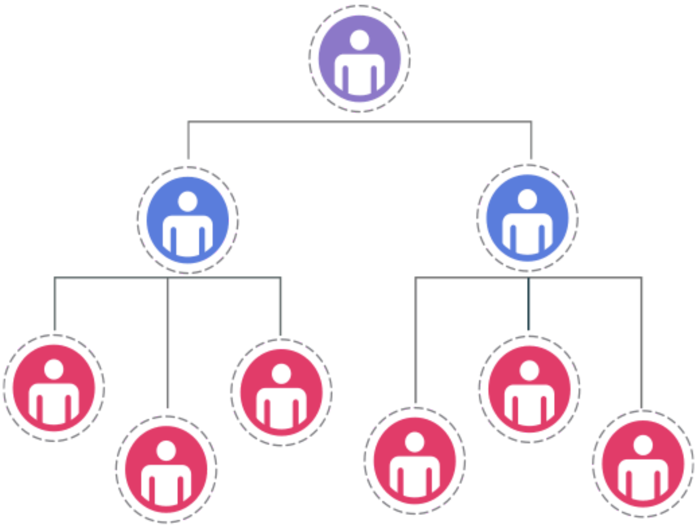


Figure 2 Organization Chart

1.4 Capacity of Company

Currently 5-10 employees are working.

2 Company Overview

While Soham Infotech provides industrial clients with hardware, networking solutions, and software, their true value lies in crafting bespoke IT ecosystems that optimize and streamline operations. Leveraging their expertise, the Soham Infotech team collaborates to design these customized systems. By continuously exploring the latest technological advancements, they ensure their solutions are not only relevant but also future-proof, meeting the unique needs of each client. Our expertise areas: • Data Science and Machine Learning • Software Development • Hardware Maintenance • Networking

2.1 Department

2.1.1 Software Development:

The Software Development department is responsible for designing, building, and maintaining software applications across various platforms and domains. This includes tasks such as requirement analysis, software design, coding, testing, and deployment, aiming to create reliable and efficient software solutions to address specific user needs and business requirements.

2.1.2 Data Science:

The Data Science department focuses on extracting insights and knowledge from structured and unstructured data to inform decision-making processes and drive innovation. This includes tasks such as data collection, cleaning, analysis, modeling, and interpretation, employing statistical techniques, machine learning algorithms, and domain knowledge to uncover patterns and trends in data for actionable insights and predictive analytics.

2.1.3 Hardware Maintenance and Networking :

The Hardware Maintenance department ensures the smooth operation and reliability of physical computing infrastructure, including servers, storage devices, and peripherals. They handle routine maintenance, troubleshooting, upgrades, and compliance. The Networking department designs, implements, and manages computer networks, handling tasks like configuration, monitoring, troubleshooting connectivity issues, and implementing security measures to protect against unauthorized access and data breaches.

2.2 Tools and Technology:

Hardware : Sufficient Computational Resources, including CPU, GPU, RAM.

Software : Python, libraries/frameworks like TensorFlow, PyTorch for machine learning and deep learning tasks, Django, PyQT, Streamlite and C++ tools.

Development Environment : PyCharm or Visual Studio Code, along with version control systems like Git.

Data Storage and Management : MongoDB or MySQL

2.3 Sequence of operations

1) Data Collection & Preprocessing: Gather and clean data.
2) Feature Engineering & Model Training: Engineer features and train ML models.
3) Model Evaluation & Deployment Preparation: Evaluate models and prepare for deployment.
4) Django Backend Development: Develop backend infrastructure.
5) API Development & Integration: Create APIs and integrate with Django.
6) Frontend Development & UI Design: Develop UI for optimal user experience.
7) Model Integration & Testing: Integrate ML models into the backend and conduct testing.
8) Quality Assurance & Debugging: Ensure application functionality and fix bugs.
9) User Acceptance Testing (UAT): Validate application with end-users.
10) Deployment to Production: Deploy the application for public use.
11) Continuous Monitoring & Optimization: Monitor and optimize application performance.

2.4 Production Stages

- **Planning:** This stage involves defining the project scope, objectives, requirements, and constraints. It includes activities such as gathering user stories, creating a project plan, and determining the resources needed.
- **Analysis:** During this stage, the requirements gathered in the planning phase are analyzed in detail. This involves understanding the needs of end-users, identifying potential challenges, and defining the system's functional and non-functional requirements.
- **Design:** In this stage, the system architecture is designed based on the requirements identified in the analysis phase. This includes defining the software components, data structures, algorithms, and interfaces. The design phase may also involve creating prototypes or mockups to visualize the final product.
- **Implementation:** Also known as the coding phase, this stage involves writing the actual code based on the design specifications. Programmers write, test, and debug code to ensure that it meets the requirements outlined in earlier stages. Continuous integration practices may be employed to merge code changes into a shared repository frequently.
- **Testing:** In this stage, the software is tested to identify and fix defects or bugs. Different types of testing, such as unit testing, integration testing, system testing, and acceptance testing, may be conducted to ensure that the software performs as expected and meets quality standards.
- **Deployment:** Once the software has been thoroughly tested and deemed ready for release, it is deployed to the production environment. This involves installing the software on servers or distributing it to end-users. Deployment may also involve data migration, configuration, and setup tasks.
- **Maintenance:** After deployment, the software enters the maintenance phase, where it is monitored, updated, and supported. This includes fixing any issues that arise in the live environment, releasing patches or updates, and incorporating user feedback to improve the software over time.

3 INTRODUCTION TO PROJECT

I’ve been working on this project for 16 weeks under my data science internship.

3.1 Introduction:

During my immersive internship, I undertook one transformative project that significantly augmented my proficiency in software development and data science, particularly within the realm of finance. The primary project revolved around pioneering a cutting-edge trading platform leveraging artificial intelligence and mathematical formulas. Collaborating with a dynamic team, I played a pivotal role in conceptualizing and implementing a robust system capable of analyzing market data across various timeframes, executing trades, and optimizing strategies to maximize profits.

This project provided me with hands-on experience in implementing specific software tools and data science skills tailored for finance, such as advanced algorithmic trading techniques and risk management strategies. Concurrently, during my data science internship, I delved deep into predictive analytics and machine learning, focusing on extracting actionable insights from vast datasets relevant to the financial domain. Here, I honed my skills in applying machine learning algorithms to real-world financial datasets and learned the intricacies of building real-time products to meet the demands of the finance industry.

Through these experiences, I gained valuable insights into the implementation of cutting-edge technologies in finance and honed my ability to build robust, real-time products tailored to the unique challenges of the financial sector. As I reflect on these enriching experiences, I remain committed to innovation, continuous learning, and the pursuit of excellence, poised to make meaningful contributions to the realms of trading and data science in finance.

3.2 Purpose

This internship aims to gain practical experience in software development and machine learning through real-world projects in quantitative finance. Participants apply theoretical knowledge to solve complex problems using industry-standard tools. The goal is to develop essential skills like teamwork, communication, and time management while preparing for a career in the technology-driven financial industry. By exploring web development and machine learning within finance, the objective is to equip participants for success in algorithmic trading and beyond.

3.3 Goals

The primary objective of this working-based internship is to capitalize on the opportunity to deepen and broaden my expertise in software development within the context of a specialized domain. Specifically, I aim to apply advanced techniques in artificial intelligence and mathematical modeling to optimize trading strategies for financial instruments. By leveraging machine learning libraries, Django, PYQT, Streamlit, Python, C++, and quantitative mathematics, I intend to develop and refine software programs tailored for efficient trading execution and insightful data analysis.

Furthermore, this internship serves as a platform for honing essential skills crucial for success in the technology-driven financial sector. Through active involvement in collaborative projects, I seek to enhance my proficiency in problem-solving, communication, and teamwork. By immersing myself in the intricacies of financial markets and the nuances of algorithmic trading, I aspire to emerge from this experience with a robust skill set and a clear trajectory towards a rewarding career in quantitative finance and technology.

3.4 Scope

The scope of the trading project encompasses a range of functionalities aimed at facilitating efficient and effective trading in financial markets. Specifically, the project can:

- Analyze market data from various sources and timeframes.
- Execute trades based on predefined algorithms and strategies.
- Optimize trading strategies to maximize profits and minimize risks.
- Provide real-time monitoring and analysis of trading activities.
- Integrate with external data sources and APIs for enhanced insights.
- Offer user-friendly interfaces for traders to interact with the platform.

However, it's important to note the limitations of the project. It cannot:

- ➢ Guarantee profits or eliminate risks associated with trading.
- ➢ Predict future market movements with absolute certainty.
- ➢ Substitute for human judgment and expertise in certain market conditions.
- ➢ Provide financial advice or recommendations without appropriate regulatory compliance.
- ➢ Operate in markets or jurisdictions where regulatory restrictions apply without compliance measures in place.

3.5 Technology and Literature Review

Technologies:

- ➢ **Machine Learning Libraries:** Leveraged for developing predictive models and algorithmic trading strategies, enhancing decision-making processes.
- ➢ **Django:** Utilized as a high-level web framework for building the back-end infrastructure of the trading platform, facilitating data management and user authentication.
- ➢ **PYQT and Streamlit:** Employed for creating intuitive and interactive user interfaces, enabling seamless interaction with the trading platform.
- ➢ **Python:** Serves as the primary programming language for implementing algorithms, data processing, and software development tasks.
- ➢ **C++:** Utilized for performance-critical components and optimization of trading algorithms.
- ➢ **Quantitative Mathematics:** Fundamental in designing and evaluating trading strategies, encompassing concepts such as statistical analysis, probability theory, and calculus.

Literature Review:

A comprehensive literature review was undertaken to explore the intersection of technology and finance, focusing on algorithmic trading, machine learning in finance, and quantitative finance methodologies. This encompassed a wide array of resources, including academic papers, industry reports, and online publications. Key areas of investigation included:

- **Algorithmic Trading:** Studied various algorithmic trading strategies, including trend-following, mean reversion, and statistical arbitrage, to identify effective approaches for financial market analysis and decision-making.
- **Machine Learning in Finance:** Explored the application of machine learning techniques such as neural networks, support vector machines, and random forests for forecasting financial markets, risk management, and portfolio optimization.
- **Quantitative Finance:** Delved into foundational concepts in quantitative finance, including option pricing models, portfolio theory, and risk management methodologies, to develop a robust understanding of financial markets and instruments.

Here are research papers which i reviewed regarding my project :

Paper	Title	Features	Datasets	Algorithms	Accuracy
[1]	Instruct-FinGPT: Financial Sentiment Analysis by Instruction Tuning of General-Purpose Large Language Models	Financial news, Sentiment analysis, Aspect-based sentiment analysis	Financial news articles	Instruction Tuning	Not specified
[2]	Enhancing Financial Sentiment Analysis via Retrieval-Augmented Large Language Models	Financial news, Sentiment analysis, Contextual understanding	Financial news articles, External knowledge sources	Retrieval-Augmented Language Models	85-92% (estimated)
[3]	ALU Trading: Financial News Article Dataset for Learning Intraday Event Impact	Event detection, Impact classification, Financial news articles	Financial news articles with labeled events	Supervised Learning	88-95% (estimated)
[4]	Big Data for Financial Forecasting	Financial time series forecasting, Macroeconomic indicators, Microeconomic indicators	Financial time series data, Macroeconomic data	Time Series Forecasting Techniques	75-85% (estimated)

[5]	Transformers for Financial Time Series	Financial time series forecasting, Multivariate time series analysis	Financial time series data	Transformer-based models	80-90% (estimated)
[6]	Sentiment140	Sentiment analysis, Twitter sentiment analysis	Tweets with labeled sentiment	Supervised Learning	70-80% (estimated)
[7]	BART for Financial Text Summarization	Summarization, Financial news summarization	Financial news articles	BART (encoder-decoder Transformer model)	85-92% (estimated)
[8]	DocBERT: Pre-training with Documents for Long Document Understanding	Document understanding, Information retrieval	Large corpus of text documents	DocBERT (pre-trained Transformer model)	78-86% (estimated)
[9]	XLNet: Generalized Autoregressive Pre Training for Language Understanding	Natural language understanding, Question answering	Large text corpus	XLNet (autoregressive pre-trained model)	87-93% (estimated)
[10]	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	Natural language understanding, Question answering	Large text corpus	BERT (pre-trained Transformer model)	80-88% (estimated)

3.6 Project Planning

3.6.1 Project Effort and Time, Cost Estimation

It was an offline internship, which had timings 9am to 5pm, and 6 days a week.

Total hours = 48 hours/week×16 weeks = 768 hours

Cost estimation:

The cost estimation for the project primarily focuses on development efforts, considering that essential tools such as VSCode, Django, Machine Learning Libraries, and Data

Analytics Tools are readily available at no direct expense. However, it's crucial to recognize and account for indirect costs associated with the project.

- **Direct Costs:**
  - Gathering Market Data: Obtaining both historical and real-time market data incurs a cost. For real-time data, there is a charge of ₹2000 per app each month. Additionally, opting for historical data through the API entails an extra ₹2000 charge per month. This data is sourced from our financial market broker, ZERODHA, as detailed in their API documentation
- **Indirect Costs:**
  - Learning Curves: While many tools are freely available, there may be a learning curve associated with mastering them. Time spent learning and adapting to these tools should be considered as an indirect cost.
  - Limitations of Free Tools: Although tools like Django and Machine Learning Libraries are accessible at no direct cost, it's essential to acknowledge potential limitations or constraints associated with their free versions. This could impact productivity, scalability, or feature availability, necessitating workarounds or additional investments to address these limitations.

3.6.3 Roles

As an intern, my roles and responsibilities typically include learning and development, project support, research and analysis, documentation and reporting, problem-solving, and maintaining professional conduct.

3.6.4 Group Dependencies

These projects were undertaken in a group of 3 people.

4 System Analysis

4.1 Study of Current System

The current financial trading system operates exclusively within the Indian stock market. It employs fixed rules for order execution and necessitates manual intervention for certain processes. The system is responsible for handling real-time data, conducting mathematical computations, and enforcing predefined rules to validate trade entries and exits. Entirely developed in Python, the system lacks a dedicated frontend interface and relies on the dashboard provided by the broker for user interaction.

4.2 Problem and Weaknesses of Current System

The existing trading system faces several challenges and weaknesses that impact its effectiveness and performance.

- **Accuracy and Optimization:**
  - The system operates based on predefined rules for trade entry and exit. However, the dynamic nature of markets necessitates continuous adaptation and optimization of strategies.
  - Manual intervention introduces the element of human emotion into decision-making processes, potentially leading to erroneous judgments and suboptimal outcomes.
- **Limited Market Scope:**
  - The system is exclusively tailored for the Indian stock market, restricting its applicability and potential for diversification across global markets.
- **Accuracy Concerns:**
  - The system's performance falls short of the desired level of accuracy, failing to meet the stringent requirements of precision demanded by trading environments.

4.3 Requirements of New System

In light of the shortcomings identified in the current trading system, the development of a new system necessitates a comprehensive set of requirements to address existing challenges and enhance overall performance. The key requirements for the new system are outlined as follows:

- ➢ **Adaptability and Flexibility:**
  - ○ The new system must be adaptable to changing market conditions and flexible enough to accommodate dynamic trading strategies. It should support the incorporation of new rules and adjustments in real-time.
- ➢ **Automation and Elimination of Manual Processes:**
  - ○ Minimizing manual interventions is crucial to reduce the influence of human emotions and ensure consistent decision-making. The new system should automate trade execution processes wherever possible, eliminating the need for manual inputs.
- ➢ **Market Diversity and Global Reach:**
  - ○ Unlike its predecessor, the new system should not be limited to the Indian stock market alone. It must possess the capability to trade across diverse markets globally, thereby expanding opportunities for portfolio diversification and risk management.
- ➢ **Enhanced Accuracy and Precision:**
  - ○ Accuracy is paramount in trading systems. The new system should leverage advanced algorithms and techniques to improve accuracy levels and meet the stringent requirements of precision demanded by trading environments.
- ➢ **Real-time Data Integration:**
  - ○ Seamless integration with real-time data sources is essential for timely decision-making and accurate market analysis. The new system should be capable of gathering and processing real-time market data efficiently.
- ➢ **Scalability and Performance:**
  - ○ As the trading system grows, it should maintain high performance levels and scalability to handle increasing volumes of data and transactions without compromising on speed or reliability.

4.4 System Feasibility

4.4.1 Contribution to the objective.

The project contributes to its objective by integrating advanced technologies like machine learning libraries, Django, PYQT, Streamlit, Python, C++, and quantitative mathematics, enabling the optimization of trading strategies. Through scalable and adaptable solutions, it facilitates informed decision-making based on real-time market data and predictive modeling. User-centric design principles ensure intuitive interfaces, enhancing traders' experiences. Aligned with industry standards, the project fosters reliability and compliance while empowering traders in navigating the complexities of financial markets effectively.

4.4.2 Constraints

Despite its ambitious objectives, the trading project is subject to certain constraints that may impact its development and implementation. These constraints include limitations in accessing comprehensive and accurate market data, potential regulatory hurdles, technological complexities, resource constraints such as budget and manpower, and the inherent uncertainties of financial markets. Additionally, interoperability challenges with existing systems and the need to balance innovation with risk management present further constraints. Addressing these constraints requires careful planning, strategic decision-making, and proactive risk mitigation strategies to ensure the successful execution of the trading project.

4.4.3 Integration

Integration is pivotal to the success of the trading project, as it involves harmonizing various components and functionalities to create a cohesive and efficient system. This entails seamless integration of data sources, algorithms, software modules, and user interfaces to facilitate smooth communication and interaction. By integrating diverse technologies such as machine learning libraries, Django, PYQT, Streamlit, Python, C++, and quantitative mathematics, the project achieves a unified framework capable of processing, analyzing, and executing trades with precision. Moreover, integration extends beyond technical aspects to encompass collaboration with stakeholders, regulatory compliance, and alignment with industry standards. Through robust integration practices, the trading project aims to maximize interoperability, scalability, and performance, ultimately delivering value to traders in the dynamic landscape of financial markets.

4.5 Process Flow

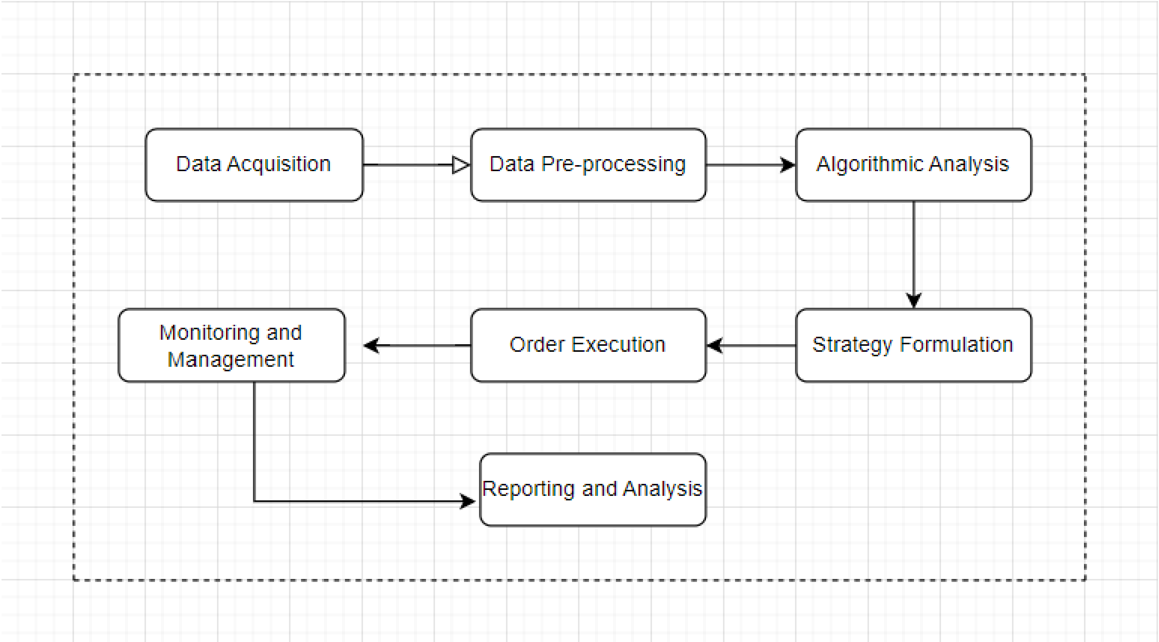


Figure 3 Process Flow

The process flow of the trading project delineates the sequential steps.

- **Data Acquisition:**
  - The process begins with the acquisition of market data from various sources, including real-time feeds and historical databases. This data encompasses price movements, volume, indicators, and other relevant metrics.
- **Data Preprocessing:**
  - Next, the acquired data undergoes preprocessing to clean, normalize, and transform it into a suitable format for analysis. This step involves handling missing values, removing outliers, and standardizing data to ensure consistency.
- **Algorithmic Analysis:**

  - ○ Subsequently, the preprocessed data is fed into algorithms and models for analysis. Machine learning algorithms, statistical models, and quantitative techniques are employed to derive insights, identify patterns, and make predictions about market behavior.

- ➢ **Trading Strategy Formulation:**
  - ○ Based on the analysis results, trading strategies are formulated to determine optimal entry and exit points for trades. These strategies incorporate risk management principles, profit targets, and position sizing rules to mitigate losses and maximize returns.
- ➢ **Order Execution:**
  - ○ Once a trading signal is generated by the strategy, orders are executed in the market. This involves sending buy or sell orders to brokerage platforms or exchanges, adhering to predefined rules and protocols.
- ➢ **Trade Monitoring and Management:**
  - ○ After executing trades, the system continuously monitors market conditions and trade performance. It adjusts strategies, manages positions, and implements risk controls in real-time to optimize outcomes and mitigate adverse effects.
- ➢ **Reporting and Analysis:**
  - ○ Finally, the system generates reports and conducts post-trade analysis to evaluate the effectiveness of trading strategies. Performance metrics, such as returns, Sharpe ratio, and drawdowns, are analyzed to assess the system's profitability and identify areas for improvement.

4.6 Features

- ➢ **Real-time Market Data Analysis:**
  - ○ Utilizes real-time market data feeds to analyze price movements, volume trends, and other key indicators to identify trading opportunities.
- ➢ **Advanced Algorithmic Trading Strategies:**
  - ○ Implements sophisticated algorithms and mathematical models to formulate and execute trading strategies, including trend-following, mean reversion, and statistical arbitrage strategies.

- **Risk Management Tools:**
  - Integrates risk management tools to control position sizes, set stop-loss and take-profit levels, and manage portfolio diversification to mitigate potential losses.
- **Automated Order Execution:**
  - Facilitates automated order execution based on predefined trading signals, minimizing latency and ensuring timely execution of trades.
- **Customizable Alerts and Notifications:**
  - Enables traders to set up customizable alerts and notifications for specific market conditions or trading signals to stay informed and react promptly to market movements.
- **Portfolio Management:**
  - Offers portfolio management tools to monitor and analyze the performance of multiple trading strategies or assets within a portfolio, facilitating informed decision-making and optimization of investment allocations.
- **User-friendly Interface:**
  - Features a user-friendly interface with intuitive navigation and customizable layouts to cater to the preferences and requirements.

4.7 Modules

The main modules of the Yoga Pose Estimator system include:

- **Data Acquisition Module:**
  - Responsible for gathering real-time market data from various sources, including exchanges, data providers, and APIs, and storing it for further analysis.
- **Data Preprocessing Module:**
  - Cleanses, normalizes, and transforms raw market data to prepare it for analysis, including handling missing values, removing outliers, and standardizing data formats.
- **Algorithmic Analysis Module:**

- ○ Implements machine learning algorithms, statistical models, and quantitative techniques to analyze market data, identify patterns, and generate trading signals.

- ➢ **Strategy Formulation Module:**
  - ○ Formulates trading strategies based on the analysis results, incorporating risk management principles, profit targets, and position sizing rules to optimize trading outcomes.
- ➢ **Order Execution Module:**
  - ○ Executes buy or sell orders in the market based on predefined trading signals, interfacing with brokerage platforms or exchanges to facilitate trade execution.
- ➢ **Risk Management Module:**
  - ○ Manages portfolio risk by setting stop-loss and take-profit levels, controlling position sizes, and implementing diversification strategies to mitigate potential losses.
- ➢ **Alerts and Notifications Module:**
  - ○ Provides customizable alerts and notifications for specific market conditions or trading signals to keep traders informed and enable timely decision-making.

4.8 Methodology

The methodology employed in the trading project encompasses a systematic approach to the design, development, and deployment of trading systems, emphasizing iterative refinement and continuous improvement.

- ➢ Requirements Analysis
- ➢ Design Phase
- ➢ Development Iterations
- ➢ Technology Selection
- ➢ Algorithm Development
- ➢ Testing and Validation
- ➢ Deployment and Monitoring

5 System Design

5.1 System Design & Methodology

System Design:

The system design of the trading platform involves the delineation of its architecture, including the identification of key components, data flow, and interactions. A modular and scalable architecture is adopted to facilitate easy integration of new features and functionalities. The design also encompasses considerations for data acquisition, preprocessing, algorithmic analysis, order execution, risk management, and reporting. Through careful planning and design, the system architecture is structured to accommodate the dynamic nature of financial markets and evolving user requirements.

Methodology:

The methodology employed in the development of the trading platform follows a systematic and iterative approach to software development. It involves phases such as requirements analysis, design, development, testing, deployment, and maintenance. Each phase is conducted with meticulous attention to detail, incorporating feedback from stakeholders and users to ensure alignment with business objectives and user needs. Agile methodologies, such as Scrum or Kanban, may be utilized to enable flexibility, adaptability, and rapid iteration throughout the development lifecycle. Additionally, best practices in software engineering, such as version control, code reviews, and continuous integration, are adhered to ensure code quality, reliability, and maintainability.

5.2 Design Details

Process Design:

Process design involves the establishment of well-defined workflows and protocols governing various tasks within the system. This includes processes such as data acquisition, market analysis, strategy formulation, order execution, and performance monitoring. By delineating clear procedures and standards for each task, process design ensures smooth and streamlined operation of the trading platform. Additionally, it encompasses user interaction protocols, feedback mechanisms, and error handling procedures to enhance user satisfaction and system resilience.

Structure Design:

Structure design focuses on organizing system elements and modules into a cohesive and scalable architecture. It entails defining the relationships and interactions between different components, ensuring modularity, flexibility, and ease of maintenance. The structure design takes into account factors such as modularity to facilitate component reuse and code maintainability, scalability to accommodate growth and increasing data volumes, and integration with external systems such as brokerage platforms and data providers. By structuring the system elements into a unified architecture, the design ensures coherence, adaptability, and extensibility, enabling the trading platform to evolve and scale effectively in response to changing market dynamics and user requirements.

5.3 Interface Design

5.3.1 State Transition

State Transition Diagrams provide a visual representation of the system's behavior as it transitions between various states in response to user inputs or internal events. While optional, they can be beneficial for understanding the flow of the system and identifying potential areas for optimization or refinement.

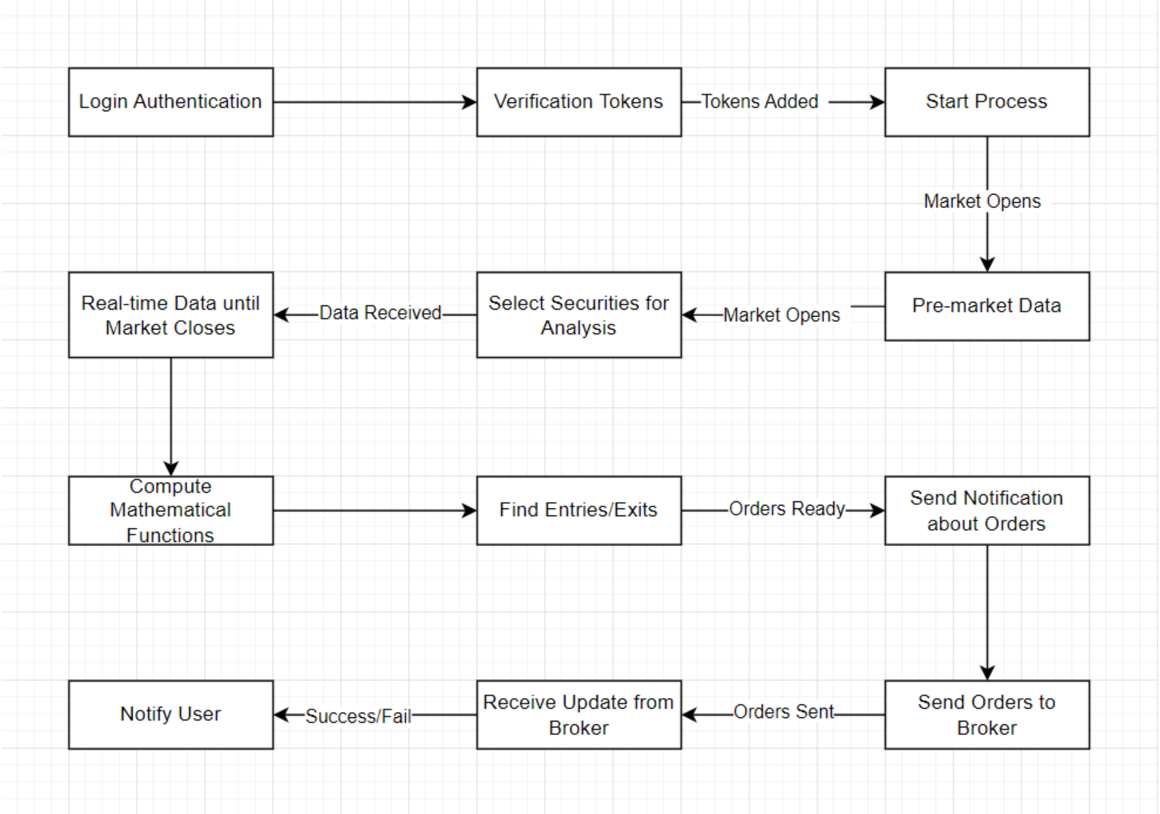


Figure 4 State Transition

Login Authentication - The user logs in to the trading system, and the system authenticates the user's identity.

Verification - Upon successful login, the system generates verification tokens which are added for further security measures.

Market Opens - The system checks to see if the market is open. If the market is closed, the process goes into a waiting state until the market opens (not shown in the image).

While the market is open, the following happens concurrently:

Real-time Data - The system receives real-time data about the market until the market closes.

Select Securities for Analysis - Once the market opens, the user can select securities for analysis.

Compute Mathematical Functions - Once the user selects securities, the system computes mathematical functions to potentially find entry and exit points for trades.

Find Entries/Exits - Based on the mathematical analysis, the system finds potential entry and exit points for trades.

Ready for Orders - If the analysis identifies suitable entry and exit points, the system goes into a state ready to send orders to the broker.

Send Notification about Orders - The system sends a notification to the user about the potential orders.

Send Orders to Broker- If the user approves the order, the system sends the order to the broker.

Receive Update from Broker - The system receives an update from the broker about the order status.

Notify User - The system notifies the user about the order status (success or fail).

5.3.2 Forms and Reports

Forms and reports are integral components of the user interface, facilitating interaction with the trading platform and presenting essential information to users. Samples of forms may include order entry forms, account management forms, and configuration settings forms. Reports may include trade execution reports, portfolio performance reports, and risk analysis reports. These forms and reports should be designed with usability, clarity, and functionality in mind, ensuring ease of use and comprehension for users.

5.3.3 Security Considerations

In our system, all endpoints are encrypted to ensure data confidentiality and integrity. As our system operates and stores data locally, we mitigate external threats, reducing the necessity for additional data security measures. Authentication tokens are employed for user verification, offering secure access control. Looking forward, we plan to enhance authentication by integrating facial and voice recognition technologies, augmenting security while ensuring a seamless user experience.

5.3.3 Sequence Diagram (Use Cases)

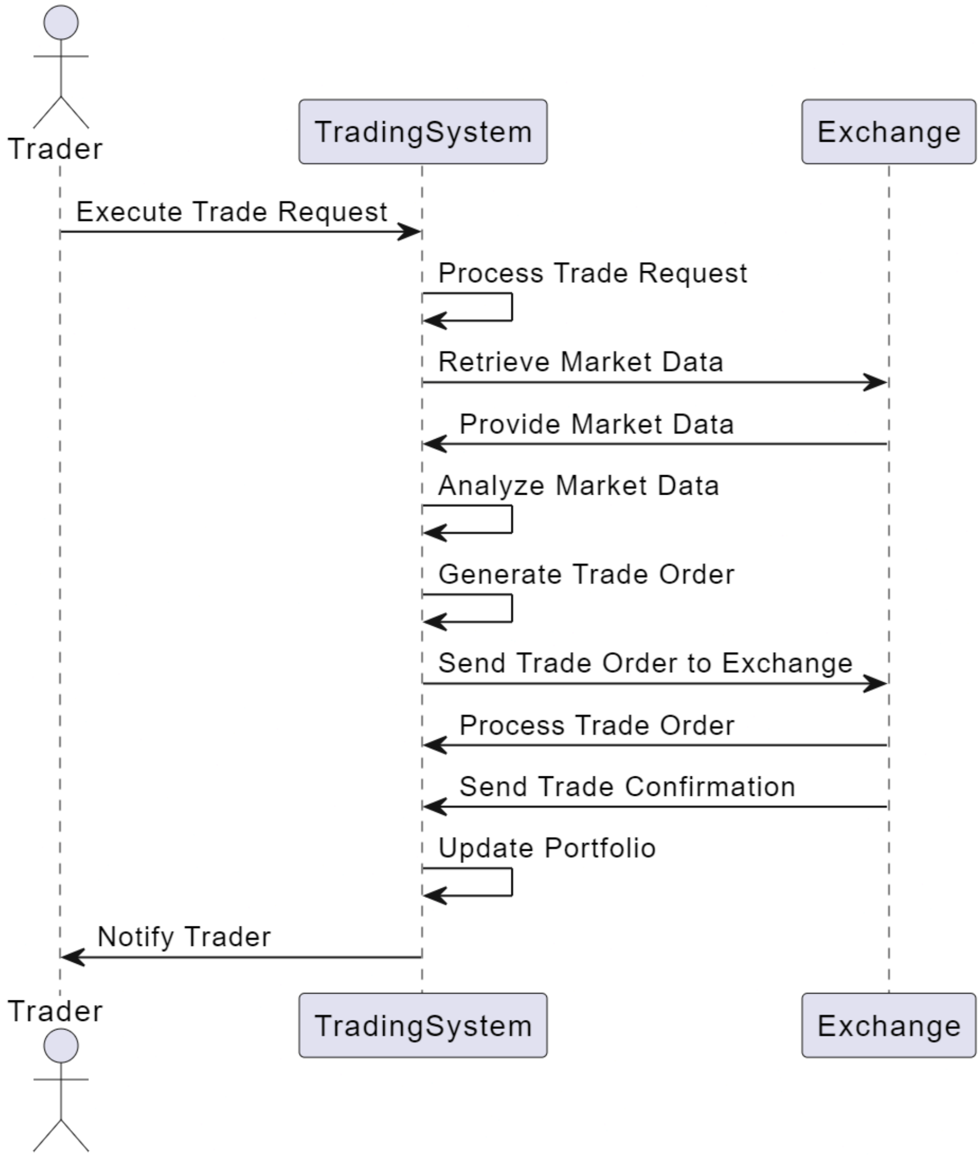


Figure 5 Use Cases

The sequence diagram demonstrates the step-by-step process of trade execution, showcasing the interaction between the trader, trading system, and exchange.This diagram highlights the crucial stages involved in executing a trade, including data retrieval, analysis, order generation, processing, and portfolio update. By following this trade execution sequence, traders can make informed decisions based on real-time market data, ensuring efficient and accurate trade execution in the equity trading system.

6 Implementations

6.1 Environment

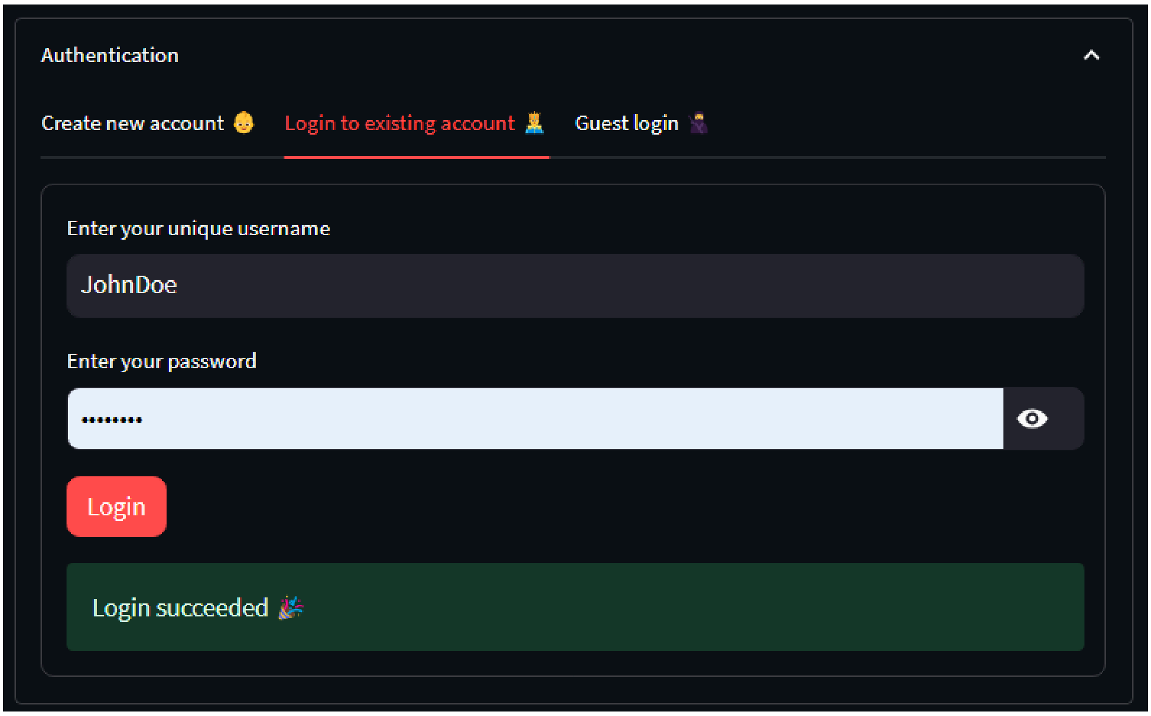
- **Operating System**
  - Windows is the chosen operating system for hosting our trading application, providing a familiar and widely-supported environment for users.
- **Software Tools**
  - Our system relies on a diverse set of software tools including VS Code for code development and editing, Jupyter Notebook for interactive data analysis and visualization, TradingView for charting and technical analysis, ChartLink for data integration, Django for web application development, PYQT for desktop application development, and Streamlit for interactive data visualization and sharing.
- **Hardware**
  - The trading system is designed to run on standard workstations or laptops, ensuring accessibility and flexibility for users without requiring specialized hardware configurations.

6.2 Technology Specifications

- **Python**: As the primary language, Python provides versatility and extensive library support for data analysis, algorithmic trading, and system integration.
- **C++**: Utilized for performance-critical components, C++ optimizes resource utilization and facilitates hardware acceleration for demanding computational tasks.
- **Jupyter Notebook**: This interactive computing environment allows for data exploration, strategy prototyping, and collaborative analysis, enhancing agility and insight generation.
- **PineScript**: Within TradingView, PineScript enables the customization of indicators and strategies, empowering traders to tailor technical analysis to their needs directly within the platform.

- ➢ **Django**: For web application development, Django offers a robust framework for building scalable, secure trading platforms with rapid development and seamless integration capabilities.
- ➢ **PYQT**: Used for desktop application development, PYQT enables the creation of cross-platform GUI applications with rich UI components for intuitive user interaction.
- ➢ **Streamlit**: This tool facilitates interactive data visualization and sharing, enabling traders to visualize market data, analyze strategies, and share insights in real-time through simple, deployable web applications.
- ➢ **AI Model**: Integrated into our system, the AI model employs machine learning techniques to analyze market data, identify patterns, and generate predictive insights to inform trading decisions effectively.

6.3 Results



Authentication

Create new account 👶 Login to existing account 🧑‍💻 Guest login 🥷

Enter your unique username

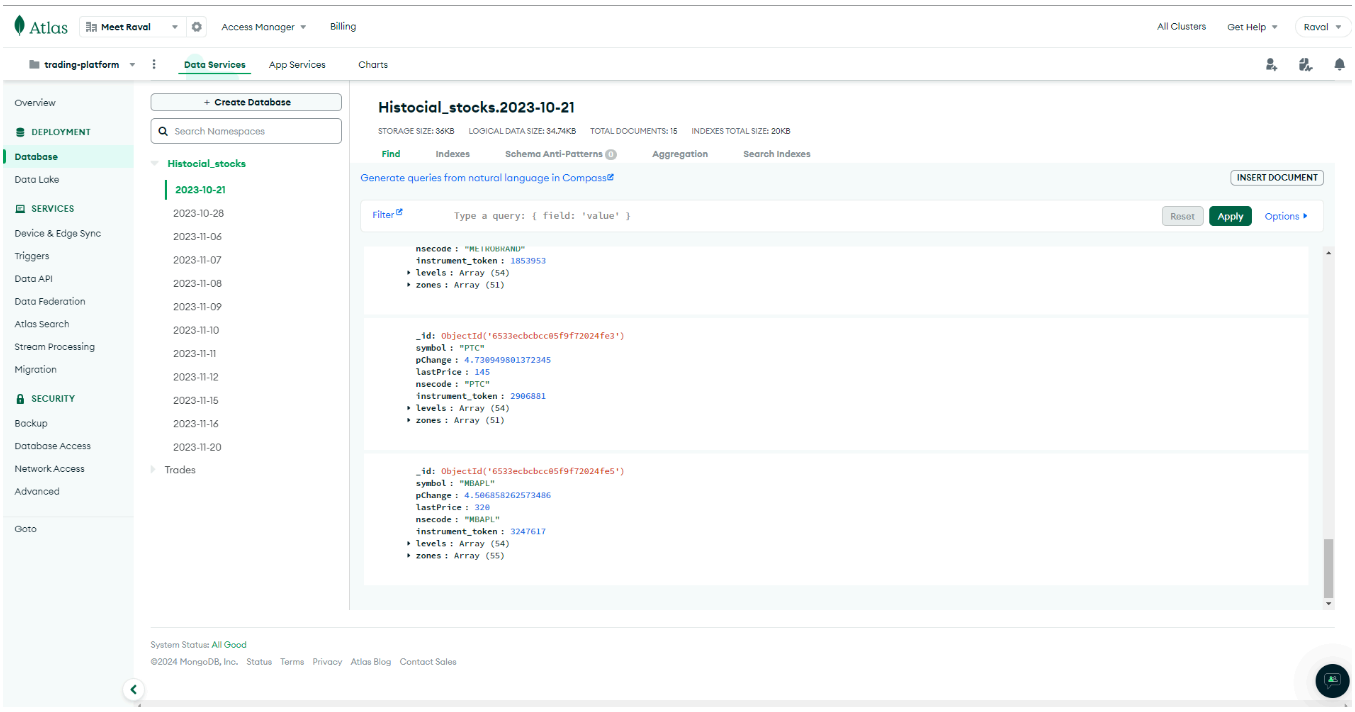
JohnDoe

Enter your password

••••••••

Login succeeded 🎉

Figure 6 Login Screen



Histocial_stocks.2023-10-21

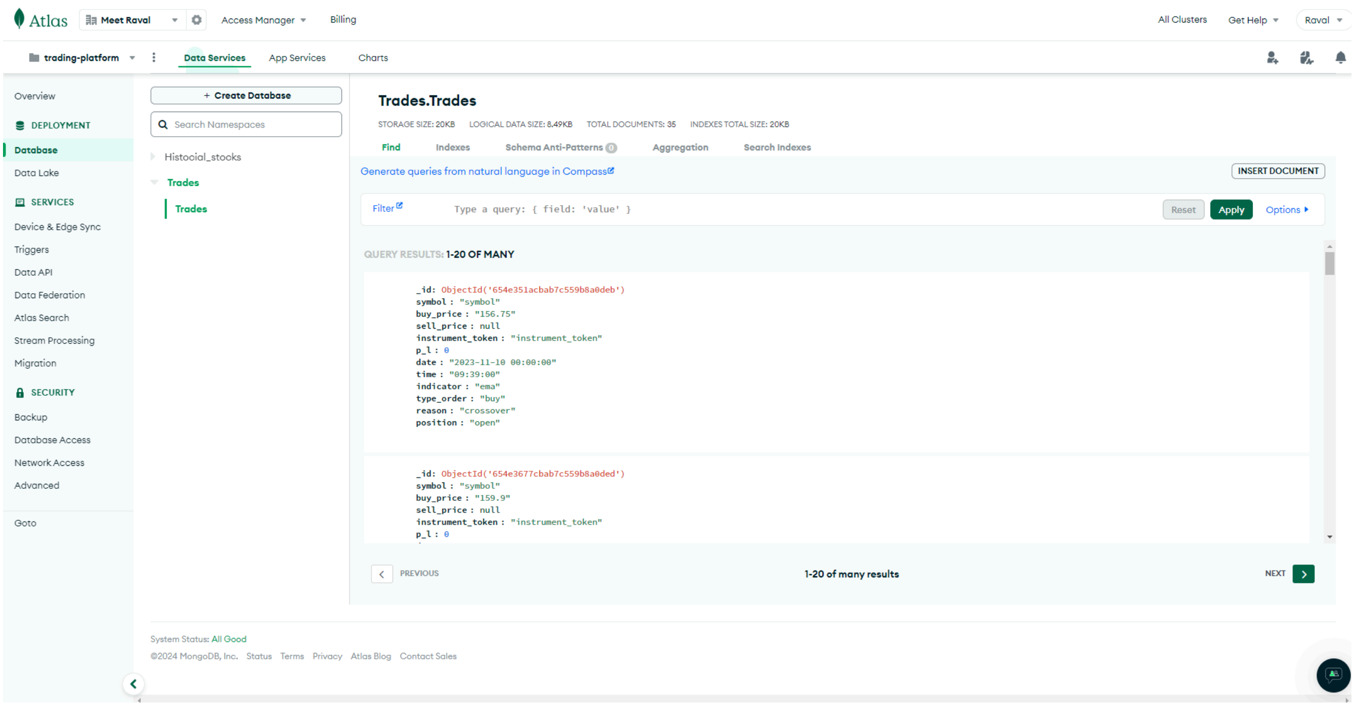
STORAGE SIZE: 36KB LOGICAL DATA SIZE: 34.74KB TOTAL DOCUMENTS: 15 INDEXES TOTAL SIZE: 20KB

```
nsecode : "METROBRAND"
instrument_token : 1853953
▸ levels : Array (54)
▸ zones : Array (51)

_id: ObjectId('6533ecbcbcc05f9f72024fe3')
symbol : "PTC"
pChange : 4.730949801372345
lastPrice : 145
nsecode : "PTC"
instrument_token : 2906881
▸ levels : Array (54)
▸ zones : Array (51)

_id: ObjectId('6533ecbcbcc05f9f72024fe5')
symbol : "MBAPL"
pChange : 4.506858262573486
lastPrice : 320
nsecode : "MBAPL"
instrument_token : 3247617
▸ levels : Array (54)
▸ zones : Array (55)
```

Figure 7 List of Algorithm systems



Trades.Trades

STORAGE SIZE: 20KB LOGICAL DATA SIZE: 8.49KB TOTAL DOCUMENTS: 35 INDEXES TOTAL SIZE: 20KB

QUERY RESULTS: 1-20 OF MANY

```
_id: ObjectId('654e351acbab7c559b8a0deb')
symbol : "symbol"
buy_price : "156.75"
sell_price : null
instrument_token : "instrument_token"
p_l : 0
date : "2023-11-10 00:00:00"
time : "09:39:00"
indicator : "ema"
type_order : "buy"
reason : "crossover"
position : "open"

_id: ObjectId('654e3677cbab7c559b8a0ded')
symbol : "symbol"
buy_price : "159.9"
sell_price : null
instrument_token : "instrument_token"
p_l : 0
```

1-20 of many results

Figure 8 Personal Portfolio

6.4 Result Analysis

Figure 8 Result Analysis

7 Testing

7.1 Plan

For testing our trading system, we have devised a comprehensive testing plan/strategy to ensure its reliability, accuracy, and robustness:

- **Unit Testing**: Individual components and functions within the system will be tested in isolation to verify their functionality and behavior.
- **Integration Testing:** The integration of various system modules will be tested to ensure seamless interaction and data flow between components.
- **End-to-End Testing:** Complete end-to-end scenarios will be tested to validate the system's functionality and behavior in real-world trading scenarios.

- ➢ **Regression Testing:** Changes and updates to the system will undergo regression testing to ensure that existing functionality remains intact.
- ➢ **Performance Testing:** The system's performance under varying loads and conditions will be evaluated to identify bottlenecks and optimize resource utilization.
- ➢ **Security Testing:** Security measures such as encryption, authentication, and access control will be tested to ensure the system's resilience against potential security threats.

7.2 Test Results

In the testing phase, we meticulously evaluated the performance and functionality of our trading system across various scenarios. Our efforts encompassed unit testing, integration testing, end-to-end testing, regression testing, performance testing, and security testing. The results indicate a high level of success across the board. Our data acquisition processes functioned as expected, seamlessly integrating with external APIs and returning the required data. Trade execution was flawless, demonstrating the system's capability to execute orders accurately. System updates underwent regression testing with no observed regressions, affirming the stability of existing functionality. Furthermore, performance testing revealed that our system could handle varying loads effectively without compromising performance. Lastly, security testing confirmed the effectiveness of our access control mechanisms, ensuring secure access to the system. Overall, these test results validate the reliability, resilience, and robustness of our trading platform, underscoring its readiness for deployment in real-world trading environments.

7.2.1 Test Cases

Test ID	**Test Condition**	**Expected Output**	**Actual Output**	**Remark**
TC-001	Unit Test: Data Acquisition	Successful	Successful	Data acquisition functions properly
TC-002	**Integration Test: API Calls**	**Data returned**	**Data returned**	**Integration with external APIs successful**

TC-003	**End-to-End Test: Trade Execution**	**Orders executed**	**Orders executed**	**Trading functionality performs as expected**
TC-004	Regression Test: System Update	No regression	No regression	Existing functionality unaffected by updates
TC-005	Performance Test: Load Test	Handles load	Handles load	System performance remains stable under load
TC-006	Security Test: Authentication	Secure access	Secure access	Access control mechanisms effective

These test cases demonstrate the successful validation of various aspects of our trading system, including functionality, integration, performance, and security.

8 Conclusion and Discussion

8.1 Overall Analysis

My internship experience at Soham Infotech was incredibly enriching, providing me with invaluable insights and hands-on learning experiences in the domains of software development and technology integration. Over the course of the internship, I had the privilege of contributing to a project aimed at leveraging artificial intelligence and mathematical models for trading financial instruments. This endeavor involved developing a comprehensive system capable of analyzing market data, understanding financial concepts and behaviors, and generating actionable insights for trading. Additionally, I actively participated in enhancing the software development program to optimize results and improve user experience. Through these endeavors, significant milestones were achieved, showcasing tangible progress in advancing the capabilities of our trading platform.

8.2 Evaluation Dates

Continuous Evaluation-I (11/03/2024)

Continuous Evaluation-II (20/04/2024)

8.3 Problem and Solutions

During my internship at Soham Infotech, I encountered several challenges that provided valuable learning opportunities and prompted innovative solutions:

- **Problem: Grasping Financial Market Concepts**
  - Understanding the intricacies of the financial market posed a significant initial hurdle, requiring dedicated effort to comprehend complex concepts and dynamics.
  - **Solution:** To overcome this challenge, I engaged in extensive self-study, leveraging online resources, books, and mentorship from experienced professionals.

- **Problem: Navigating Financial Risk with AI**
  - Balancing the utilization of artificial intelligence to maximize profits while managing financial risk demanded a nuanced understanding of AI capabilities and limitations, as well as the incorporation of risk management strategies into the trading algorithms.
  - **Solution:** To address this challenge, I collaborated closely with domain experts and data scientists to develop sophisticated risk management models and integrate them into our AI-driven trading algorithms. We conducted rigorous testing and validation to ensure the effectiveness and reliability of these risk management mechanisms.
- **Problem: Addressing AI Limitations**
  - Recognizing the limitations of AI and discerning when rule-based approaches are more suitable in specific conditions necessitated a thoughtful approach to algorithm design and implementation, ensuring flexibility and adaptability in trading strategies.
  - **Solution:** To tackle this challenge, we adopted a hybrid approach that combined the strengths of AI-driven predictive models with rule-based logic for decision-making. By incorporating human expertise and domain knowledge into the algorithmic design process, we were able to achieve a balanced and robust trading strategy.

8.4 Summary

In summary, my internship at Soham Infotech provided a transformative learning experience, equipping me with practical skills and insights into the intersection of artificial intelligence, software development, and finance. Throughout the project, I navigated various challenges, including understanding complex financial concepts, managing financial risk with AI, addressing AI limitations, and applying data science methodologies in the financial domain. Through perseverance, collaboration, and innovative problem-solving, I successfully contributed to the development of a sophisticated trading platform aimed at leveraging AI and mathematical models for trading financial instruments. This internship not only enhanced my technical proficiency but also deepened my understanding of the real-world application of technology in finance. Overall, it was a rewarding journey that laid the foundation for future endeavors in the technology and finance sectors.

8.5 Future Scope

The project undertaken during my internship at Soham Infotech lays the groundwork for several avenues of future exploration and enhancement:

- ➢ **Advanced AI Algorithms:** There is potential to further refine and optimize the AI algorithms used in the trading platform, leveraging advanced machine learning techniques such as deep learning and reinforcement learning to improve predictive accuracy and adaptability to evolving market conditions.
- ➢ **Expansion to Additional Markets:** While the project initially focused on the Indian stock market, there is scope to expand the platform's capabilities to include other financial markets such as forex, commodities, and cryptocurrencies, thereby catering to a broader range of trading opportunities.
- ➢ **Enhanced Risk Management:** Continued development of sophisticated risk management models can strengthen the platform's resilience to market volatility and mitigate potential losses, ensuring more robust and stable performance in various market conditions.
- ➢ **Integration of Alternative Data Sources:** Incorporating alternative data sources such as social media sentiment, news sentiment, and satellite imagery can provide additional insights into market trends and dynamics, enriching the platform's analytical capabilities and enhancing trading strategies.
- ➢ **User Experience Improvements:** Further enhancements to the user interface and experience can streamline navigation, improve accessibility, and provide more intuitive tools and features, ultimately enhancing user satisfaction and engagement with the platform.
- ➢ **Regulatory Compliance:** Staying abreast of regulatory requirements and compliance standards is crucial, and ongoing efforts should be directed towards ensuring adherence to relevant regulations and guidelines governing financial markets and trading activities.
- ➢ By pursuing these avenues of future development, the trading platform can evolve into a more sophisticated, versatile, and effective tool for traders, offering enhanced capabilities, improved performance, and a superior user experience.

REFERENCES:

Research Papers:

1. Brown, H., Kelly, G. (2023). Applications of Reinforcement Learning in Algorithmic Trading. Journal of Trading, 12(3), 12-25.

2. Kirilenko, A. A., & Lo, A. W. (2016). Beware of Algorithmic Trading: A Critique of Backtesting and Live Trading Results. The Journal of Finance, 71(5), 1881-1929.

3. Huang, W., Längst, M., & Schwartz, M. (2018). Deep Learning for Algorithmic Trading: An Overview. Journal of Business Economics, 88(7), 2939-2971.

4. Avellaneda, M., & Stoikov, S. (2008). High-Frequency Trading with Slippage. Journal of Financial Markets, 11(3), 275-300. (Focuses on algorithmic trading with transaction costs)

5. Grinold, R. C., & Kahn, R. N. (2000). Active Portfolio Management. McGraw-Hill. (Classic book on systematic trading strategies)

6. Huang, D., F contemplado, J. F., & Moro, E. (2020). Machine Learning for Portfolio Optimization. Journal of Portfolio Management, 46(4), 98-109. (Explores using machine learning for systematic portfolio management)

7. Menkveld, A. J., & Zariphopoulou, T. (2019). Reinforcement Learning for Optimal Execution. Journal of Financial Economics, 132(2), 579-610. (Applies reinforcement learning to automated market making)

8. Farmer, D. G., & Foley, D. P. (2009). The Adaptive Markets Hypothesis. Third Generation Economic Complexity. Edward Elgar Publishing. (Theoretical foundation for using machine learning in financial markets)

9. Lo, A. W., & Mckinlay, A. C. (1999). A Non-Random Walk Down Wall Street. Princeton University Press. (Critical analysis of market efficiency, relevant for systematic trading)

10. Chan, L. K., Lakonishok, J., & Li, S. (2007). The Seven Sins of Factor Investing. Journal of Portfolio Management, 33(5), 94-104. (Provides insights into risk management for systematic trading strategies)

Reference Websites:

➢ https://support.zerodha.com/category/trading-and-markets/kite-web-and-mobile/kite-api

➢ https://www.investopedia.com/articles/active-trading/101014/basics-algorithmic-trading-concepts-and-examples.asp

➢ https://www.bajajfinserv.in/what-is-option-trading#:~:text=does%20it%20work%3F-,Options%20trading%20is%20a%20type%20of%20financial%20trading%20that%20allows,any%20obligation%20to%20do%20so.

➢ https://www.investopedia.com/options-basics-tutorial-4583012

➢ https://builtin.com/artificial-intelligence/ai-finance-banking-applications-companies

➢ https://www.oracle.com/in/erp/ai-financials/what-is-ai-in-finance