International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Intelligent Paper-Trading and Portfolio Analysis Platform with Risk-Aware Trading and AI-Assisted Financial Analysis

Ravsaheb Khairnar

Department of Computer Science, Patkar-Varde College, University of Mumbai

***

Abstract— Today, digital trading applications make it easier for users to access financial-market information, check their portfolio, view charts, and perform trading activities from web and mobile applications. However, users often need to use different sections or applications for trading, portfolio tracking, risk analysis, financial research, and performance analysis. Beginners may also find some of the financial information difficult to understand.

This paper presents an **Intelligent Paper-Trading and Portfolio Analysis Platform** that combines paper trading, portfolio management, risk analysis, transaction-cost calculation, and AI-assisted financial analysis in one application. The proposed system provides three main modes: **Invest, Trade, and Analyze**. The Invest mode focuses on portfolio and investment information, the Trade mode provides simulated trading functionality, and the Analyze mode provides portfolio analysis, risk information, research, and AI-based explanations.

The proposed system uses a secure authentication architecture with separate authentication for the main application and socket communication. The application authentication handles login, registration, OTP verification, OAuth authentication, PIN verification, and token refresh. The socket authentication is used for controlled access to real-time stock activity and WebSocket communication. The paper-trading system uses virtual money and simulates orders, transactions, holdings, and portfolio values without performing real financial transactions.

The analysis layer calculates portfolio value, gross and net profit and loss, transaction charges, sector exposure, portfolio concentration, and the possible effect of a new order on the user's portfolio. An **AI Financial Copilot** is also proposed to provide explanations using verified portfolio and market information. Important financial calculations are performed by the application itself instead of depending on the AI model for numerical results.

The evaluation of the proposed system can include functional testing, financial calculation accuracy, order-processing performance, portfolio-risk calculations, AI response grounding, and comparison of different system configurations. Since the complete experimental evaluation is still part of the development process, this paper describes the proposed evaluation method instead of claiming results that have not yet been measured.

Keywords— Paper Trading, FinTech, Portfolio Analysis, Risk Analysis, Artificial Intelligence, Financial Technology, WebSocket, Trading Simulation, Explainable AI, Portfolio Management.

1. INTRODUCTION

Financial markets have become easier to access because of digital trading platforms. Modern trading applications allow users to search for stocks, check market prices, create watchlists, view charts, manage portfolios, and place orders using web or mobile applications.

However, a trading application can become difficult to use when many features are placed together. Different users also need different types of information. For example, a beginner may mainly want to see portfolio value, holdings, and basic stock information. An active trader may need charts, positions, orders, watchlists, and real-time market data. Before placing an order, a user may also want to know how that order will affect the overall portfolio.

Paper trading provides a useful way to solve some of these problems without using real money. In paper trading, the user receives virtual money and can perform simulated buy and sell operations. This makes it possible to test the complete trading process in a controlled environment.

A basic paper-trading system normally focuses on buying and selling. It may not provide enough information about the effect of a transaction on the complete portfolio. For example, it may not clearly show the change in sector exposure, transaction charges, concentration, or portfolio value after an order.

Another problem is that financial information is often available in different places. Market prices, portfolio data, transaction history, financial information, and news may come from different sources. The user therefore needs to combine this information manually to understand the complete situation.

Artificial intelligence can help by providing information in a natural-language format. However, financial applications need to handle AI carefully because an AI model should not create account balances, holdings, transaction values, or other financial information on its own. These values should come from the trusted database and application calculations.

To address these problems, this paper proposes an intelligent paper-trading platform that combines simulated trading, portfolio analysis, risk analysis, and AI-assisted financial information in one application.

The main contributions of this work are:

Integrated paper trading: The system provides a complete simulated trading process from market-data retrieval to order execution, transaction creation, holding updates, and portfolio calculation.

Risk-aware order analysis: The system shows how a proposed order can affect portfolio exposure and concentration before the simulated order is executed.

Transparent P&L calculation: Gross P&L and estimated transaction charges are calculated separately so that the user can understand the actual simulated result.

Portfolio X-Ray: The system provides portfolio information based on holdings, sectors, allocation, and concentration.

AI-assisted financial analysis: An AI-based interface is proposed for explaining portfolio and market information using verified application data.

Multi-mode interface: Invest, Trade, and Analyze modes provide different functionality based on the user's activity.

Secure real-time architecture: Application authentication and socket authentication are handled as separate security contexts.

2. RELATED WORK

2.1 Digital Trading and Investment Platforms

Digital trading platforms provide users with financial-market information and trading features through web and mobile applications. Common features include stock search, market prices, watchlists, charts, portfolios, orders, positions, and transaction history.

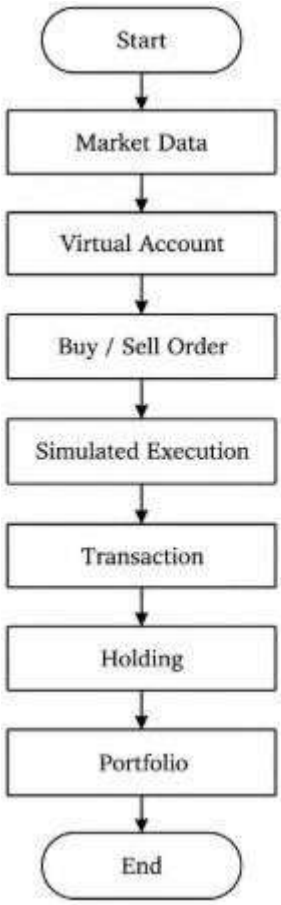
These applications generally need to solve two important problems. The first is providing reliable and timely market information. The second is processing transactions correctly. A trading system needs to maintain the relationship between an order, its execution, transaction record, holding, and portfolio value.

The proposed platform follows this basic architecture but adds portfolio and risk-analysis features around the trading process.

2.2 Paper Trading Systems

Paper trading is a method where users perform financial transactions using virtual money instead of real money.

A basic paper-trading process can be represented as:



Paper trading is useful during application development because the complete order-processing workflow can be tested without placing real orders in the market.

However, a simple paper-trading system normally concentrates on order execution. It may not explain how a particular transaction changes the user's complete portfolio.

2.3 Portfolio Analysis and Risk Management

Portfolio analysis looks at all the holdings together instead of looking at each stock separately.

Some important portfolio measures are:

- Portfolio value
- Individual holding allocation
- Sector allocation
- Portfolio concentration
- Unrealized P&L
- Realized P&L
- Exposure
- Historical performance

A transaction can therefore have two types of effects:

```
Individual Position Effect

          +

Portfolio-Level Effect
```

For example, if a user purchases more shares of a company, the exposure to that company increases. If the company belongs to a particular sector, the sector exposure can also increase.

Because of this, the proposed system adds a risk-analysis stage before the final simulated order.

2.4 Artificial Intelligence in Financial Applications

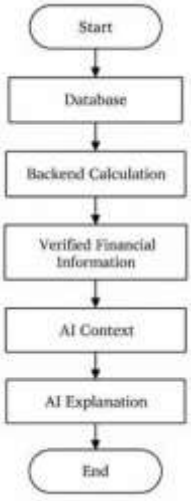
Artificial intelligence can be used to convert structured and unstructured financial information into easy-to-understand explanations.

Some possible uses include:

- Portfolio summaries
- Stock research
- Financial-news analysis
- Trading-journal analysis
- Portfolio explanations
- Risk explanations
- Natural-language financial search

However, AI-generated information can sometimes contain incorrect or unsupported information. Because of this, the proposed system does not use the AI model as the main source for account information.

Instead, the system follows this process:



This keeps the financial calculations separate from the natural-language generation performed by the AI.

2.5 Existing Trading Application Limitations

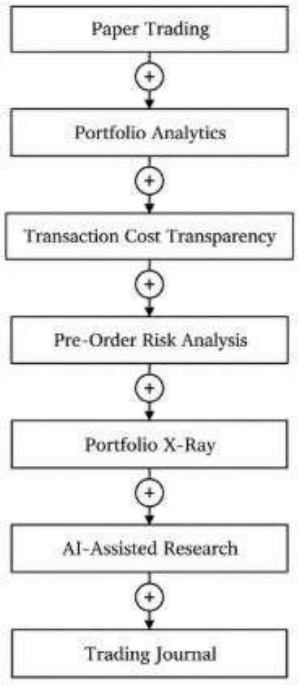
Existing trading applications provide many different features. The proposed system focuses on bringing several of these features together in one paper-trading environment.

Functionality	Traditional Trading Interface	Proposed Platform
Stock information	Yes	Yes
Watchlist	Yes	Yes
Charts	Yes	Yes
Order management	Yes	Yes
Paper trading	—	Yes
Portfolio	Yes	Yes
Simulated trading	Some platforms	Core functionality
Transaction-cost explanation	Varies	Proposed
Pre-order portfolio impact	Limited	Proposed
Portfolio X-Ray	Varies	Proposed
AI portfolio explanation	Limited/varies	Proposed
Trading journal	Varies	Proposed
AI-assisted research	Varies	Proposed

This table describes the scope of the proposed research system. It does not mean that every existing trading platform does not provide these features.

2.6 Research Gap

Existing trading and investment applications provide many individual features. The main focus of this research is to combine these features in one paper-trading environment.
The proposed system combines:



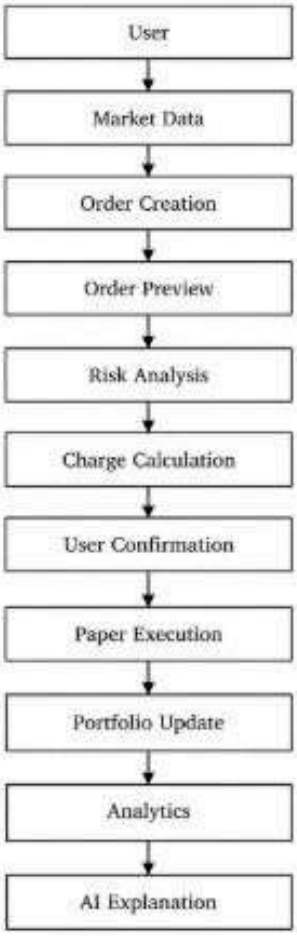
The main idea is that trading and analysis should not be treated as completely separate features. The effect of a simulated trade should also be connected with portfolio analysis and risk information.

3. PROPOSED FRAMEWORK

3.1 Overview

In a normal paper-trading application, a user selects a stock, enters the quantity, submits an order, and receives a simulated execution result.

The proposed system adds some additional analysis before and after the simulated transaction.



The purpose of this architecture is to give the user more information about a transaction. The system does not automatically make the trading decision for the user.

3.2 Three Operating Modes

The proposed platform contains three main operating modes.

Invest Mode

The Invest mode provides simple investment and portfolio-related features.

- Portfolio
- Stocks
- ETFs
- Watchlist
- Investment Information
- Goals

Trade Mode

The Trade mode provides the main paper-trading features.

- Watchlist
- Charts
- Market Data
- Orders
- Positions
- Buy / Sell
- Alerts

Analyze Mode

The Analyze mode provides advanced analysis features.

- AI Financial Copilot
- Portfolio X-Ray
- Risk Analysis
- Stock Research
- News Analysis
- Trading Journal
- Performance Analytics

These three modes allow the application to show different levels of information depending on what the user is currently doing.

3.3 Framework Components

The main components of the proposed framework are described below.

Authentication and Authorization:
The system provides authentication features such as login,

registration, OTP verification, OAuth, PIN verification, biometric verification, and token refresh.

Market Data Layer:

The market-data layer provides stock and index information needed by charts, watchlists, portfolio valuation, and paper trading.

Paper-Trading Engine:

The trading engine receives simulated buy and sell requests. It then updates orders, transactions, holdings, and virtual balances.

Portfolio Engine:

The portfolio engine calculates holdings, average cost, portfolio value, allocation, and profit and loss.

Risk Engine:

The risk engine calculates the possible effect of a proposed order on the existing portfolio.

Portfolio X-Ray:

The X-Ray module provides information about the portfolio based on holdings, sectors, allocation, and concentration.

AI Financial Copilot:

The AI layer provides natural-language explanations using verified information from the application.

Trading Journal:

The journal stores completed paper trades and provides historical information about the user's trading activity.

3.4 Portfolio Analysis Outputs

Instead of giving only one risk value for the complete portfolio, the proposed system provides different types of portfolio information.

Output	Meaning
Portfolio Value	Current simulated value of holdings and available funds
Gross P&L	Profit/loss before estimated charges
Net P&L	Profit/loss after estimated charges
Sector Exposure	Portfolio allocation by sector
Holding Concentration	Exposure to individual securities
Order Impact	Expected portfolio change after a proposed order
Historical Performance	Previous portfolio and trading performance
Trading Statistics	Historical user trading behavior

These outputs are intended to provide useful information to the user rather than automatically deciding whether a transaction should be performed.

4. METHODOLOGY

4.1 Problem Formulation

Consider a paper-trading account U containing:

- Virtual balance B
- Holdings H
- Orders O
- Transactions T

A proposed order can be represented as:

$$o = (s, q, p, d)$$

where:

- s = security
- q = quantity
- p = estimated execution price
- d = order direction, such as buy or sell

The system needs to process the order and calculate the resulting simulated portfolio state.

For a buy transaction:

$$OrderValue = q \times p$$

The new balance is:

$$B' = B - OrderValue - Charges$$

The holding quantity becomes:

$$Q' = Q + q$$

For a sell transaction:

$$B' = B + SaleValue - Charges$$

and:

$$Q' = Q - q$$

The system needs to maintain consistency between these values.
The methodology mainly focuses on five tasks:

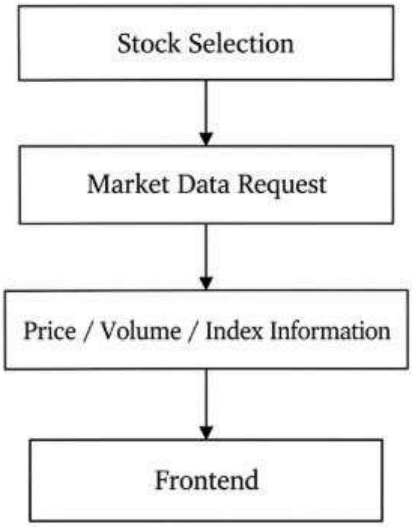
1. Processing simulated orders.
2. Maintaining the correct portfolio state.

3. Calculating transparent P&L.
4. Calculating portfolio impact and risk.
5. Generating AI explanations using verified information.

4.2 Market Data Retrieval

The first step is to retrieve the market information required by the application.

The basic process is:



For real-time applications, price updates can be provided using WebSocket communication.

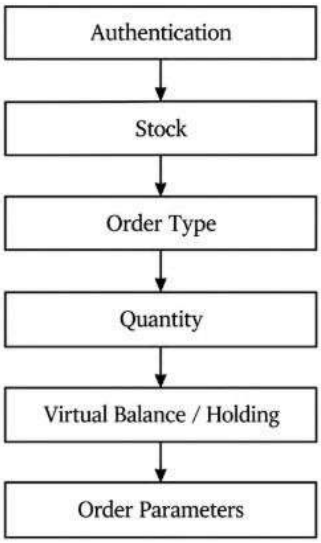
The system should distinguish between:

- Current price
- Previous close
- Historical prices
- Simulated execution price
- Portfolio valuation price

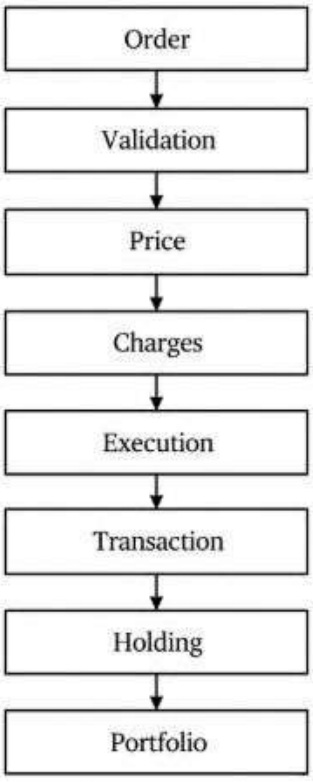
This is important because the market price and the simulated execution value may not always be the same.

4.3 Paper Order Processing

When a user submits an order, the system first validates the order.



After validation, the order is sent to the simulation engine.



The transaction is then stored as a historical record.

4.4 Average Cost Calculation

When a user purchases the same security multiple times, the average acquisition cost can be calculated as:

$$\text{Average Cost} = \frac{\sum_{i=1}^{n} Q_i P_i}{\sum_{i=1}^{n} Q_i}$$

Where:

Q_i = Quantity purchased in the i^{th} transaction

P_i = Purchase price per share in the i^{th} transaction

n = Total number of purchase transactions

The unrealized gross P&L can then be calculated as:

GrossP&L = (CurrentPrice - AverageCost) × Quantity

The system should store enough transaction information so that these calculations can be reproduced.

4.5 Transaction Cost Calculation

The proposed system keeps the transaction value and simulated charges separate.

TotalCharges =

Brokerage + Taxes + ExchangeCharges + OtherCharges

Therefore:

NetP&L = GrossP&L - TotalCharges

The exact charge model should be configurable because charges can be different depending on the market, broker, instrument, and transaction type.

In this paper-trading system, these charges are simulated and do not represent actual charges from a live broker.

4.6 Portfolio Risk Calculation

Before the user confirms a simulated order, the system calculates the possible change in portfolio exposure.

Current exposure to security s:

$$E_s = \frac{Value_s}{PortfolioValue} \times 100$$

After the proposed order:

$$E'_s = \frac{Value'_s}{PortfolioValue'} \times 100$$

The change in exposure is:

$$\Delta E_s = E'_s - E_s$$

The same method can also be used to calculate changes in sector exposure.

For example:

```
Current Portfolio

IT                  28%

Banking             22%

Energy              18%

Other               32%

Proposed Order

IT                  39%

Banking             21%

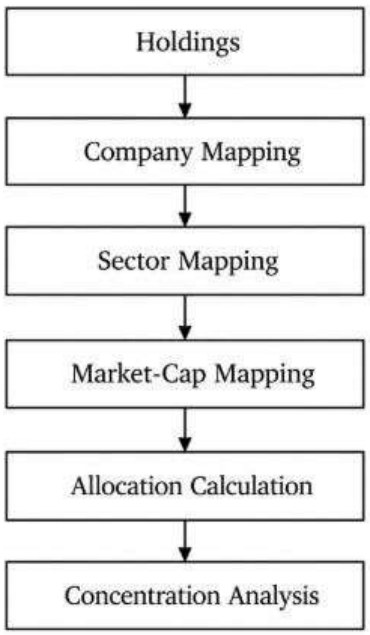
Energy              17%

Other               23%
```

The system displays this change to the user instead of automatically blocking the transaction.

4.7 Portfolio X-Ray Methodology

The Portfolio X-Ray converts individual holdings into portfolio-level information.



The allocation can be calculated as:

$$Allocation_i = \frac{Value_i}{TotlPortfdioVlue} \times 100$$

The resulting information can then be shown using tables and charts.

4.8 AI Financial Analysis

The AI layer receives verified information from the application before generating an explanation.

For example:

```
User:

Why is my portfolio down today?
Backend:
Portfolio P&L = -₹2,450


Top contributors:

Stock A = -₹1,200
Stock B = -₹750
Stock C = -₹500


Sector exposure:

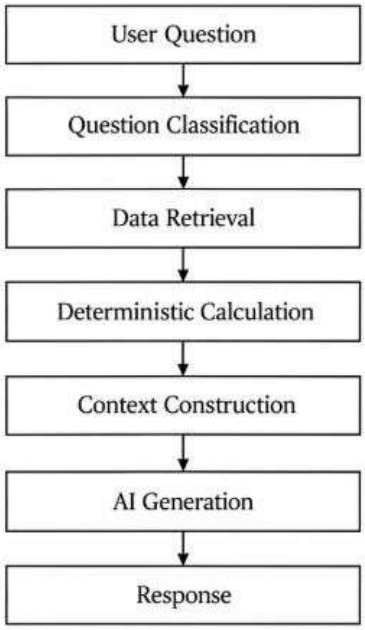
IT = 34%

Banking = 29%

Energy = 18%
```

The AI then generates a natural-language explanation using this information.

The intended process is:

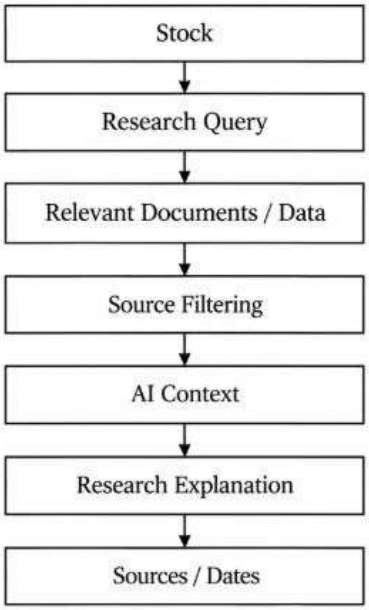


The AI should not independently calculate the user's account balance or other important financial values.

4.9 Source-Aware Financial Research

For stock research, the system can retrieve information from configured financial sources.

The process can be represented as:



The final response should clearly separate:
Facts obtained from sources
Calculations obtained from the application

AI-generated explanations

Information for which enough evidence is not available

This can make the financial information easier for the user to understand and verify.

5. SYSTEM IMPLEMENTATION

5.1 Architecture

The proposed system is implemented as a full-stack application containing a frontend, backend API, database, WebSocket layer, trading engine, portfolio engine, and AI services.

The high-level architecture is:

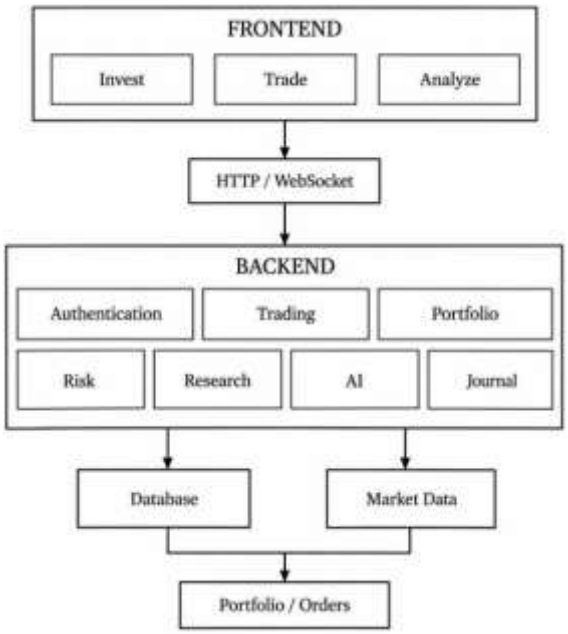


Figure 1. Proposed system architecture showing the frontend, backend services, database, market-data layer, and real-time communication.

5.2 Authentication Architecture

The authentication system uses separate authentication contexts for the main application and socket communication.

App Token

The application uses:

- Access Token
- Refresh Token

Socket Token

The socket connection uses:

- Access Token
- Refresh Token

The architecture can be represented as:

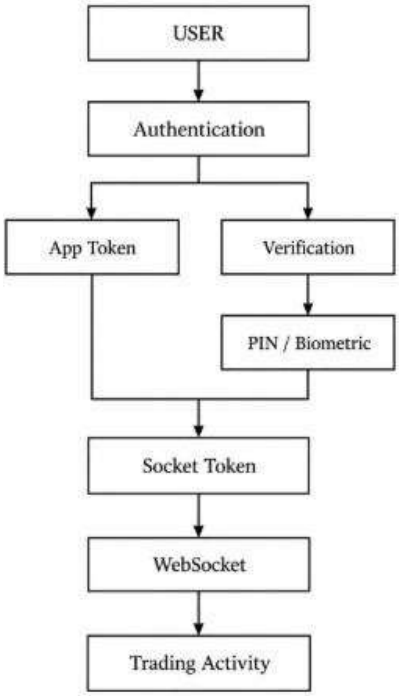


Figure 2. Application and socket authentication architecture.

5.3 Technology Stack

Table II. Technology Stack

Component	Technology
Frontend	React / React Native
Backend	Node.js
API	Express.js
Database	MongoDB / Mongoose
Real-time communication	WebSocket / Socket.IO
Authentication	JWT
OAuth	Google / Apple
Validation	Backend validation middleware
AI Layer	LLM-based service
Charts	Trading/financial chart library
API Testing	Postman
Version Control	Git / GitHub
Deployment	Cloud deployment platform

The exact versions and AI model should be mentioned in the final paper according to the actual versions used in the project.

5.4 Procedural Flow

The complete application workflow can be divided into five major workflows.

A. Authentication Workflow

1. Enter Email
2. Check Email
3. Login / Registration
4. OTP / OAuth
5. Create or Verify Password
6. Create PIN
7. Verify PIN / Biometric
8. Generate Socket Token

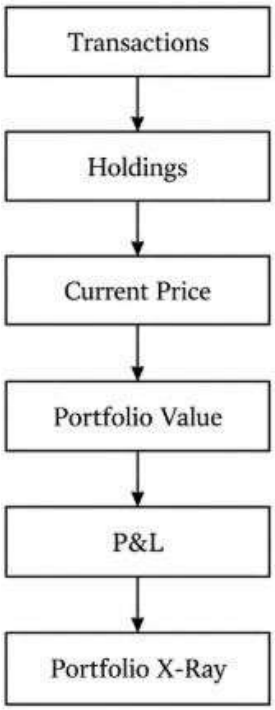
B. Market Workflow

1. Select Stock
2. Retrieve Market Data
3. Display Price
4. Display Chart
5. Add to Watchlist

C. Paper-Trading Workflow

1. Select Buy/Sell
2. Enter Quantity
3. Calculate Order Value
4. Calculate Charges
5. Calculate Portfolio Impact
6. Show Order Preview
7. Confirm
8. Simulate Execution
9. Create Transaction
10. Update Holding
11. Update Portfolio

D. Portfolio Workflow



E. AI Workflow

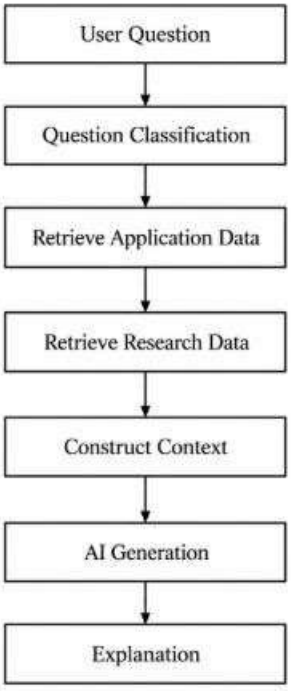


Figure 3. Procedural flow of the main workflows in the proposed system.

6. EVALUATION METHODOLOGY

The proposed platform should be evaluated using multiple test configurations. This helps to understand the contribution of the individual components instead of only testing the complete system.

The following configurations can be used.

Configuration A

Basic paper-trading system.

Configuration B

Paper trading + transparent P&L.

Configuration C

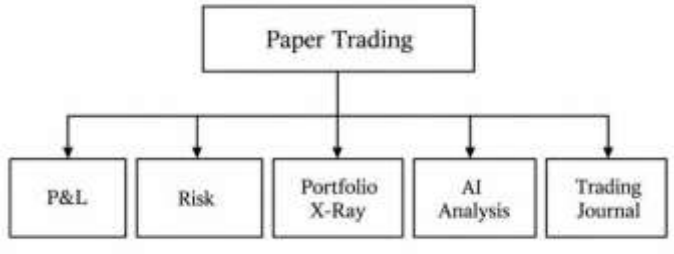
Paper trading + P&L + portfolio risk.

Configuration D

Paper trading + P&L + risk + Portfolio X-Ray.

Configuration E

Complete system containing:



6.1 Metrics

The following metrics can be used to evaluate the system.

Transaction Accuracy

This measures whether simulated transactions produce the expected changes in balance and holdings.

$$TransactionAccuracy = \frac{CorrectTransactions}{TotalTestTransactions}$$

P&L Accuracy

The system result can be compared with an independently calculated expected value.

$$P\&L_{Error} = |ExpectedP\&L - SystemP\&L|$$

Portfolio Calculation Accuracy

This measures whether the portfolio value and allocation calculated by the system match the expected values.

Risk Calculation Accuracy

The expected exposure after a proposed order is compared with the result produced by the system.

Order Processing Latency

This measures the time between order submission and simulated execution.

AI Response Grounding

AI responses can be checked to see whether the financial information in the response is supported by application data or retrieved sources.

AI Response Latency

The average response time and percentile response time can be measured for AI queries.

6.2 Dataset Construction

A custom dataset can be created to evaluate the analytical and AI components.

The dataset can include:

- Synthetic user portfolios
- Historical stock-price data
- Simulated transactions
- Portfolio allocation scenarios
- Risk scenarios
- Financial research questions
- Portfolio-related questions
- Intentionally incorrect financial statements

For example:

```
Question:

What is my current portfolio value?
Expected:
₹5,42,30

AI Response:

The portfolio value is ₹5,42,300.
Result:
Supported
```

Another test can provide an incorrect value:

```
Expected:

₹5,42,300


Generated:

₹6,42,300


Result:

Incorrect financial claim
```

This type of testing helps evaluate the AI component separately from the paper-trading engine.

6.3 Financial Calculation Testing
Separate tests should be created for:

- Buy orders
- Sell orders
- Multiple purchases
- Partial selling
- Average cost
- Gross P&L
- Net P&L
- Charges
- Portfolio value
- Sector allocation
- Portfolio concentration

For each test:
Error = |Expected - System|

The expected result is that deterministic calculations remain within the numerical tolerance defined for the system.

6.4 AI Evaluation

AI responses can be evaluated using the following criteria:

- Numerical accuracy
- Data grounding
- Source correctness
- Unsupported claims
- Response completeness
- Response latency

A basic grounding metric can be defined as:

$$GroundingRate = \frac{SupportedClaims}{TotalFinancialClaims} \times 100$$

The unsupported-claim rate can be calculated as:

$$UnsupportedRate = \frac{UnsupportedClaims}{TotalFinancialClaims} \times 100$$

These metrics should be calculated using a manually labeled evaluation dataset.

6.5 Ablation Study

The ablation study is used to check the contribution of the major components of the proposed system.

Table III. Proposed Ablation Configurations

Configuration	Paper Trading	P&L	Risk	X-Ray	AI	Journal
A	✓	—	—	—	—	—
B	✓	✓	—	—	—	—
C	✓	✓	✓	—	—	—
D	✓	✓	✓	✓	—	—
E	✓	✓	✓	✓	✓	✓

The purpose of this study is to understand the contribution of each major component instead of assuming that every additional feature automatically improves the system.

---

7. LIMITATIONS AND FUTURE WORK

7.1 Limitations

Although the proposed system combines paper trading, portfolio analysis, risk analysis, and AI features, there are still some limitations.

Market Simulation

Paper trading cannot completely reproduce real market conditions.

Real trading may involve:

- Slippage
- Liquidity
- Market impact
- Partial execution
- Order-book changes
- Network delays

Because of this, the result of the simulation may be different from what happens in an actual market.

Dependence on Market Data

The accuracy of portfolio valuation and paper trading depends on the quality and freshness of the market-data source.

AI Model Dependence

The quality of AI-generated explanations depends on the selected language model, prompt design, retrieved information, and context provided to the model.

The AI is therefore intended to help with explanations and analysis rather than acting as an independent financial decision-maker.

Risk Model Limitations

The proposed risk engine mainly focuses on portfolio exposure and concentration. It does not represent every type of financial risk.

For example, it may not fully handle:

- Liquidity risk
- Counterparty risk
- Changes in volatility
- Market-wide shocks
- Complex derivatives risk

Limited Evaluation Dataset

If the first evaluation uses a custom dataset, it may not represent every possible user behavior or financial situation.

System Scale

A research prototype may be tested with a limited number of users and transactions. Its performance may therefore be different from a production application with many users accessing the system at the same time.

Real Trading Restrictions

The proposed system is a paper-trading platform. Connecting it to real financial markets would require additional work related to brokerage APIs, exchange connectivity, security, compliance, regulatory requirements, and financial safeguards.

7.2 Future Work

The following improvements can be considered for future versions of the system:

- Integration with authorized real-time market-data providers.
- Improved simulated execution using slippage and liquidity models.
- Addition of advanced technical indicators.
- Addition of advanced portfolio-risk measures.
- Correlation analysis between different holdings.
- Historical portfolio backtesting.
- Strategy-based paper trading.
- Advanced trading-journal analytics.
- Improved AI retrieval and source grounding.
- Claim-level verification of AI-generated financial information.
- Document-based financial research using annual reports and company filings.
- Multi-user role-based access control.
- Improved WebSocket scalability.
- Comprehensive audit logs.
- Evaluation using larger datasets.
- Comparison of different AI models.
- Study of confidence calibration for AI-generated financial explanations.

---

8. CONCLUSION

This paper presented an **Intelligent Paper-Trading and Portfolio Analysis Platform** that combines simulated trading, portfolio management, risk analysis, transaction-cost calculation, and AI-assisted financial analysis in one system.

The proposed platform is divided into three main modes: **Invest, Trade, and Analyze**. Invest mode provides basic investment and portfolio features. Trade mode provides simulated market and order functionality. Analyze mode provides portfolio analysis, risk analysis, financial research, and AI-assisted explanations.

The system also separates application authentication from real-time socket activity. The application handles login and token-related operations, while a separate socket authentication mechanism is used for WebSocket-based activity.

The paper-trading engine maintains the relationship between orders, transactions, holdings, and portfolio values. The portfolio engine calculates average cost, portfolio value, gross P&L, and net P&L. The risk engine checks how a proposed order can change portfolio exposure before the simulated order is executed.

One important part of the proposed system is the separation of financial calculations from AI-generated explanations. Values such as portfolio value, holdings, transaction values, and P&L should be calculated by the application services. The AI layer can then use these verified values to generate explanations in natural language.

The proposed evaluation includes functional testing, financial calculation accuracy, portfolio-risk accuracy, order-processing latency, AI grounding, and ablation testing of the major components.

At the current stage, the complete experimental evaluation is still part of the development process. Therefore, this paper does not claim specific improvements in trading performance or AI accuracy without actual experimental results. The next step is to complete the implementation, prepare the evaluation dataset, perform the planned experiments, and analyze the obtained results.

ACKNOWLEDGMENT

I would like to thank the **Department of Computer Science, Patkar-Varde College, University of Mumbai**, for providing the academic environment and resources required for this project.

I would also like to thank the faculty members involved in the project and research methodology work for their guidance and support during the development and documentation of this system.

REFERENCES

[1] F. Allen, J. McAndrews, and P. Strahan, "E-Finance: An Introduction," *Journal of Financial Services Research*, vol. 22, 2002.

[2] R. C. Merton, "The Theory of Rational Option Pricing," *The Bell Journal of Economics and Management Science*, vol. 4, no. 1, 1973.

[3] H. Markowitz, "Portfolio Selection," *The Journal of Finance*, vol. 7, no. 1, 1952.

[4] W. F. Sharpe, "Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk," *The Journal of Finance*, vol. 19, no. 3, 1964.

[5] A. López de Prado, *Advances in Financial Machine Learning*. Wiley, 2018.

[6] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2009.

[7] T. Brown et al., "Language Models are Few-Shot Learners," *Advances in Neural Information Processing Systems*, 2020.

[8] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *Advances in Neural Information Processing Systems*, 2020.

[9] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," *Proceedings of EMNLP-IJCNLP*, 2019.

[10] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv:1907.11692, 2019.

[11] C. Niu et al., "RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models," *Proceedings of ACL*, 2024.

[12] J. Li et al., "HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models," *Proceedings of EMNLP*, 2023.

[13] P. Manakul, A. Liusie, and M. Gales, "SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models," *Proceedings of EMNLP*, 2023.

[14] OpenAPI Initiative, *OpenAPI Specification*, documentation.

[15] Socket.IO, *Socket.IO Documentation*, documentation and implementation resources.