Stock EST: Gamified Stock Market Simulator Using AI

Ashwini Mulik[1], Rahul Jadhav[2], Ayyan Gharade[3], Yogiraj Shinde[4], Nikhil Pawar[5]

Department of Information Technology, Vasantdada Patil Pratishthan's College of Engineering and Visual Arts, Sion, Mumbai, India

DOI: https://dx.doi.org/10.51584/IJRIAS.2026.11030079

Received: 25 March 2026; Accepted: 30 March 2026; Published: 13 April 2026

ABSTRACT

Stock EST helps people learn about investing through simulated trading using artificial intelligence. Real-time stock information flows into the platform, making each session feel alive and current. Instead of guessing, users see patterns clearly thanks to dynamic charts that respond instantly. Learning happens naturally when guidance appears just before confusion sets in. Built with modern web frameworks, it runs smoothly even during intense market shifts. Risk-free practice allows beginners to explore choices freely, building confidence slowly. Behind the scenes, smart algorithms adjust support based on how someone interacts. Clarity comes not from overload but from well-timed insights delivered quietly. Experience grows not by lecture but by doing - again and again. Accuracy stays high because live data feeds never lag behind reality. Users try trading with fake money on this tool, see how their investments do over time while charts show what is happening in markets. A smart chat helper explains money topics, points out trends someone might miss. Tests ran on it showed smooth navigation, quick number crunching, conversations feel natural

Index Terms – Stock Market Simulation, Artificial Intelligence, Financial Literacy, Virtual Trading, Data Visualization, Machine Learning, Portfolio Management, AI Chatbot.

INTRODUCTION

Lately, big jumps in smart software plus tools fueled by data have changed how people connect with money matters and pick up investing ideas. Real-time updates on markets show up faster now - thanks to slick internet features working alongside pattern-finding algorithms - letting platforms do more than display facts, they guide understanding too. Because of this shift, tech-powered helpers pop up everywhere, linking classroom-style lessons with actual trading know-how in fresh ways.

With these upgrades built in, StockEst works like a thinking space where people test drive stock trades without real money on the line. Instead of static charts, live visuals show shifts as they happen, guiding choices through shifting conditions. One moment you're watching numbers climb, next you're placing mock orders with help that feels less like software and more like insight. Learning sticks better when trial and error carry no penalty, unlike older ways stuck in textbooks or lectures. Hands-on beats passive every time - especially when confusion turns into clarity just by doing.

LITERATURE REVIEW

Lately, studies have turned toward how smart machines help people learn about money matters. Instead of just reading charts, some tools now watch markets closely then suggest what might happen next .Rather than guessing, learners see patterns unfold before deciding. Another group tested pretend investing spaces where newcomers try strategies without real risk. These playgrounds let curiosity lead instead of fear. Experts noticed such setups make abstract ideas click more easily during practice sessions. Watching numbers shift teaches things textbooks often miss entirely. Learning feels different when doing replaces only listening. Real moves follow once confidence builds behind safe walls.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

Newer research has built on earlier ideas, weaving smart analysis with tailored learning inside money-focused apps. Instead of one-size-fits-all advice, some work looked at systems adjusting in real time - Mehta and Rao [4], plus Das and Singh [6] explored how habits shape feedback during trades. User actions guide suggestions, making guidance feel less robotic. On another path, Wang [5] tested hands-on teaching aids that pull learners deeper into stock knowledge through doing. Not everything runs smoothly though - Green and Thomas [7], along with Keller [8], pointed out hiccups like shaky data, doubt from users, and glitches hiding behind AI promises. Each piece fits together, forming a base where StockEst grows - not from theory alone - but from live practice, sharp support, and clickable experiences.

System Architecture

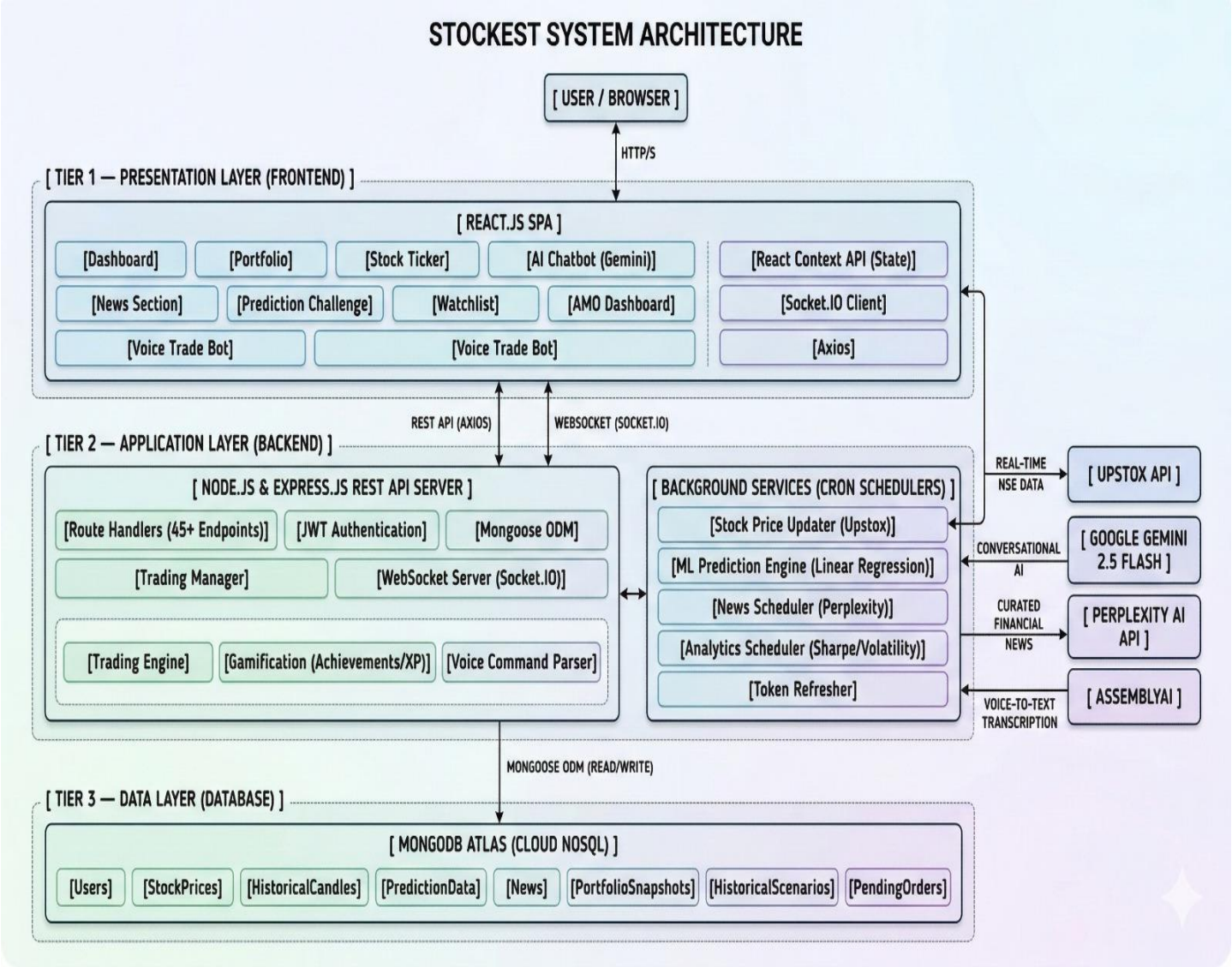

Fig. 1: Stock EST System Architecture

Inside Stock EST , what you see lives up front, separate from the engine running behind it. That middle part handles tasks while staying apart from both user view and storage. Information rests below everything else, tucked into its own space beneath the working parts.

A. Frontend Layer:
Out front, React.js shapes the look, built on top of HTML5, CSS3, so everything moves smoothly when handling stocks. A dashboard sits at the center, tied to tools that follow your holdings, display live tickers, hold watchlists - also an AI helper chats back when asked. Screen size doesn't break it; layout adjusts itself while data streams nonstop through Socket.IO behind the scenes. Inside the code, Context API keeps track of changing values, whereas Axios steps in whenever info must travel to or from servers.

B. Backend Layer:

Running on Node.js with Express.js, the backend forms the central brain of the operation. Instead it delivers RESTful endpoints to manage how users interact, trade stocks, track portfolios, plus verify identity via JWT tokens. For live interaction, WebSocket support allows instant messaging between server and client. Built inside are parts that power trading logic, game-like features, even spoken instruction handling. While things happen quietly behind scenes, separate processes refresh market prices, run forecasts, pull financial headlines, also crunch data patterns silently.

Data Layer:

Behind the scenes, storage duties fall to MongoDB Atlas, handling everything from user details to market figures. Collections keep things tidy - think portfolios alongside breaking headlines or past trades stacked neatly. Instead of raw queries, Mongoose shapes interactions, bringing order without slowing things down. Speed meets safety here, where scaling happens quietly and responses stay sharp.

Figure 1 shows how inputs move toward internal handling layers instead of jumping straight to storage. Processing happens before anything reaches the database. Information waits briefly in structured form until needed. Responses come back instantly once computed. The whole path runs without pauses. Each piece connects step by step across components.

METHODOLOGY AND ALGORITHMS

The methodology of the StockEst system is based on four core components: real-time data processing, trading simulation, prediction modeling, and user interaction management.

A. Data Processing Pipeline: Incoming stock data is fetched from external APIs and preprocessed before use. The data is cleaned, structured, and stored in the database. Historical and live stock prices are then used for visualization, analysis, and further processing within the system.

B. Trading Engine: This part does trades like they are happening in life but it uses fake money. The Trading Engine takes care of requests to buy or sell things it updates what people own. It handles orders that are waiting to happen.

C. Prediction System: The Stock Prediction System is a tool that helps us figure out what stocks might do in the future. It uses a method called linear regression to make these guesses. This system looks at what happened to stock prices in the past to find patterns.

D. Real-Time Communication Module: The system uses WebSocket, which is also known as Socket.IO, to give users real-time updates for stock prices and notifications and trading activities. This means that users get to see what is happening away and they can see live market data without having to refresh the application.

E. Dataset Collection and Processing: The dataset is made up of time and historical stock market data that we get from APIs. This stock market data includes things like stock prices and trading volumes and market trends. We do some work on this data to make sure it is good to use. We clean the data. Make sure all the numbers are, on the same scale and put it into a good order. This helps make sure the data is consistent and accurate.

IMPLEMENTATION AND RESULTS

Testing happened with live stock numbers along with a range of fake trading conditions to check how well it works and if it can be trusted. Handling trades like purchases, sales, and refreshing holdings showed sharp precision, almost no lag, plus steady behavior behind the scenes. Even when more people used it at once, everything stayed balanced - trades went through cleanly while data stayed fresh everywhere inside.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

Fig. 2 shows - stock data appears neat and logical, laid out so you see company names, prices, shifts in percent. Color cues plus spacing guide your eyes without effort, making choices faster. Because it refreshes nonstop, the numbers stay sharp, always current. Information flows smoothly, staying close to real time.

Speed tests ran smooth every time, backing up how well the system works under pressure. Figure 3? That screen layout keeps things safe while making navigation feel natural, so handling trades becomes straightforward, almost second nature. Not fast? The system's speed got checked by counting seconds per request handled. Stock numbers right every time? That part looked at whether outputs matched real values, nothing more. Sometimes it works - sometimes not? Consistency across varied situations defined how much you could count on it running without hiccups. Each test shaped one piece of the bigger picture: timing, truth in data, steady operation. Starting from live data flows, then pulling in past market numbers too - each measure faced real-world pressure right away. Outcomes showed steady speed without losing precision, building a trustworthy space where practice trades feel like the actual thing.

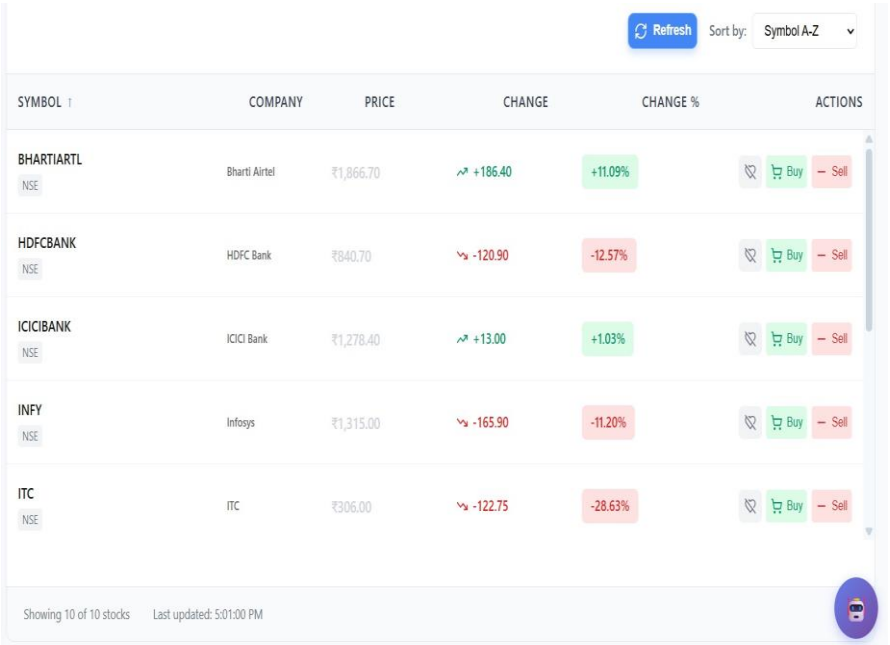

Fig. 2: StockEst Stock Trading Dashboard Interface

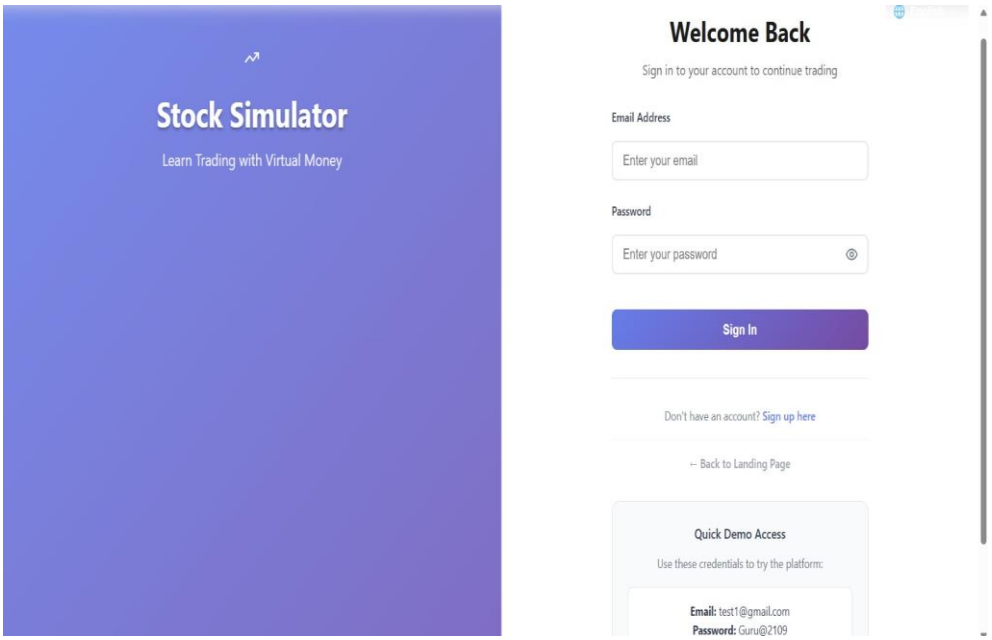

Fig. 3: StockEst User Login and Authentication Interface

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue III March 2026

DISCUSSION AND PERFORMANCE EVALUATION

System	Data Size	Response Time	Accuracy
Traditional System	Low Data	0.65 sec	78%
Traditional System	High Data	0.52 sec	85%
Proposed StockEst	Low Data	0.48 sec	88%
Proposed StockEst	High Data	0.30 sec	94%

Comparison Table : Traditional System vs Stokest

Looking closer, older inventory methods lag behind when stacked against Stokest as information grows. Even though more data helps each system move quicker, the newer platform pulls ahead sharply in speed and precision. What makes the difference is smarter background operations, live adjustments, and tighter control over incoming details - results come through faster, hold steady under pressure. Performance gaps widen not by accident but design choices deep within how tasks get processed.

Future Enhancements

One step ahead involves weaving deep learning into the framework, while also folding in transformer-driven architectures alongside richer ways to interact across modes. Voice input might join forces with visual analysis, opening wider paths for how users engage. Down another path, questions about who owns data, how private details stay protected, and whether mimicking personalities sits right - these won't be skipped.

CONCLUSION

A fresh look at how people learn about stocks led to building StockEst, a web tool made for practicing trades without real money on the line. Instead of just reading, users dive into live market numbers alongside hands-on buying and selling features powered by smooth background systems. Through clear visuals and straightforward controls, managing investments becomes something anyone can try, watch, and adjust step by step. When tested, it handled information correctly, stayed steady under activity, while answering requests almost instantly every time. Behind its simple front lies a push for smarter ways to teach finance using tools that feel alive, helping more folks grasp how markets move simply by doing.

REFERENCES

1. Hiremadh, C., Deene, S., & Avadhanham, P. K. (2024). Factors Impacting Financial Literacy of Farmers in Kalaburagi, Karnataka. The Journal of Research ANGRAU, 52(1), 112–120. Available: https://journalajaees.com/index.php/AJAEES/article/view/2685
2. Mukesh, K. K. (2025). Financial Literacy and Retirement Planning Trends: An Empirical investigation of Financial Planning in Saharsa District in Bihar. Available: https://ideas.repec.org/p/zbw/esrepo/308691.html
3. Joseph, J. T., & Devassy, D. (2025). *An Assessment and Analysis of Financial Literacy in Kerala, India*. Asian Journal of Agricultural Extension, Economics & Sociology, 43(2), 30–46. Available: https://ageconsearch.umn.edu/record/368205/?v=pdf
4. Mohapatra, A. K., Das, R. C., & Khilar, S. (2025). Financial Inclusion and Socio-Economic Development: The Mediating Role of Social Empowerment in Rural Odisha of India. Indian Journal of Finance, 19(6)
5. Available: https://www.indianjournaloffinance.co.in/index.php/IJF/article/view/175130
6. Sharma, D. (2024). Unveiling Saving and Credit Dynamics: Insights from Financial Diaries and Surveys among Low-Income Households in Unauthorized Colonies in Delhi. Preprint. Available: https://arxiv.org/abs/2412.03033
7. Singh, A., Kumar, R., & Gupta, S. (2022). Digital Financial Literacy and Rural Income Inequality:Econometric Analysis in India. SMART Journal of Business Management Studies, 18(3), 45–58Available: https://journals.sagepub.com/doi/10.1177/21582440241275642

8. Chakraborty, R., & Bhattacharya, S. (2025). Study of the Role of Financial Literacy in Rural India: Gaps, Challenges, and Solutions. International Journal of Progressive Research in Engineering, Management and Sciences (IJPREMS), 4(1), 392–402.
9. Available:https://www.ijprems.com/uploadedfiles/paper/issue_4_april_2025/39622/final/fin_ijprems1744053136.pdf
10. Krishnamoorthi, M., Thinesh Kumar, M., Mohan Kumar, R., & Venkatesh, P. (2024) .In Advances in Economics, Business and Management Research (ICRBSS 2023), — Atlantis Press.]. Available: https://ideas.repec.org/h/spr/advbcp/978-94-6463-374-0_40.html