iJRASET
International Journal For Research in Applied Science and Engineering Technology



# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 14 **Issue:** III **Month of publication:** March 2026

**DOI:** https://doi.org/10.22214/ijraset.2026.78369

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue III Mar 2026- Available at www.ijraset.com*

# AI - Based Automated Financial Portfolio Management System

Shyam Sundar V.[1], Suthish Kumar M.[2], Thirukumaran C.[3], Viswa B.[4], Gowtham[5]
[1,2,3,4,5]*Department of Computer Science and Engineering, Arunai Engineering College, Tiruvannamalai, Tamil Nadu, India*

***Abstract:*** ***The rapid growth of global financial markets and the continuous generation of real-time trading data have created significant opportunities for intelligent investment decision systems. However, traditional portfolio management methods often require extensive financial expertise, complex market analysis, and continuous monitoring of rapidly changing market conditions—challenges that can be particularly difficult for retail investors and beginner traders. This paper presents FinAI, an AI-based automated financial portfolio management system that integrates real-time stock market data analysis, machine learning–driven predictive modeling, and intelligent trading recommendation mechanisms into a unified decision-support platform. The system is designed as a modular client–server architecture consisting of a data acquisition layer, an analytical processing engine, and an interactive user interface for portfolio monitoring. Real-time market data is collected through financial APIs and processed using advanced feature engineering techniques to generate technical indicators such as moving averages, Relative Strength Index (RSI), and volatility metrics. The predictive analytics module utilizes machine learning algorithms to analyze historical stock price movements and detect patterns that indicate potential investment opportunities. Based on predictive outputs and market trend analysis, the system automatically generates actionable trading signals, including buy, hold, or sell recommendations. Experimental evaluation demonstrates that the proposed system can assist retail investors in making data-driven portfolio management decisions while reducing manual analysis effort and emotional bias in trading behavior. The platform provides a scalable and intelligent framework for next-generation automated financial advisory systems in modern financial technology environments.***

***Keywords: Artificial Intelligence, Portfolio Management, Stock Market Prediction, Machine Learning, Financial Technology, Trading Recommendation System, Algorithmic Trading, Real-Time Market Data, Investment Decision Support, Automated Portfolio Management.***

## I. INTRODUCTION

The global financial ecosystem is undergoing a transformative shift driven by digital technologies, high-speed internet connectivity, and the increasing availability of real-time financial market data. Modern stock markets generate massive volumes of trading information every second, including stock price movements, trading volumes, economic indicators, and market sentiment signals. Investors must continuously analyze these complex datasets to make informed decisions regarding buying, holding, or selling financial assets. However, the traditional investment model—where individual investors manually monitor market conditions and analyze financial indicators—often leads to inefficiencies such as delayed decision-making, emotional bias, and limited access to advanced analytical tools.

The rapid expansion of financial technology (FinTech) has significantly transformed the way individuals interact with financial markets. Digital trading platforms, algorithmic trading systems, and automated advisory services have introduced new possibilities for intelligent investment decision-making. Despite these advancements, many retail investors still face substantial challenges in interpreting financial data and predicting market behavior. Financial markets are highly volatile environments influenced by economic policies, geopolitical events, corporate performance, and investor sentiment, making accurate predictions extremely difficult without advanced analytical methods.

Artificial Intelligence (AI) and Machine Learning (ML) technologies have emerged as powerful tools for addressing these challenges by enabling automated financial analysis and predictive modeling. AI-based systems are capable of processing large-scale historical and real-time financial datasets, identifying hidden patterns, and generating predictive insights that support portfolio management decisions. Early implementations of automated trading systems primarily relied on rule-based algorithms and predefined financial indicators. However, these systems often lacked adaptability and failed to capture the complex, dynamic relationships present in modern financial markets.

©IJRASET: All Rights are Reserved 

1) A scalable system architecture capable of collecting and processing real-time financial market data through external financial APIs.
2) A machine learning–based predictive analytics module that analyzes historical stock price trends and technical indicators to forecast market movements.
3) An automated recommendation engine that generates actionable trading suggestions such as buy, hold, or sell based on predictive analysis results.
4) A portfolio monitoring dashboard that enables investors to visualize market trends, predicted signals, and portfolio performance in real time.
5) An intelligent decision-support framework that assists beginner investors in making data-driven investment decisions while reducing emotional bias in trading activities..

The remainder of this paper is organized as follows: Section II presents the system architecture including the data acquisition module, prediction engine, and recommendation system. Section III describes the methodology and implementation of the machine learning models used for stock market prediction. Section IV reviews existing research related to financial technology, algorithmic trading systems, and AI-based stock prediction models. Section V outlines the hardware and software requirements used during system development. Section VI presents the experimental results and performance evaluation of the proposed system. Finally, Section VII concludes the paper and discusses potential directions for future improvements.

### *A. Problem Statement and Motivation*

The fundamental challenges associated with financial market participation can be broadly categorized into three interconnected dimensions: market complexity, information overload, and lack of accessible analytical tools. Each of these challenges individually impacts investor decision-making, and when combined, they create significant barriers for retail investors attempting to manage their financial portfolios effectively.

Market complexity arises from the dynamic nature of financial markets, where stock prices fluctuate continuously due to multiple influencing factors such as economic policies, global events, and investor sentiment. Retail investors often struggle to interpret these market movements and identify profitable trading opportunities without the assistance of advanced analytical tools.

Information overload represents another major challenge in modern financial markets. Investors must analyze vast amounts of financial data including historical price charts, trading volumes, corporate financial reports, and technical indicators. Without automated data processing systems, analyzing this information manually can be both time-consuming and prone to human error.

Another limitation of traditional investment strategies is the lack of accessible intelligent decision-support systems for beginner investors. While institutional investors utilize sophisticated algorithmic trading systems and predictive analytics tools, similar technologies are often unavailable or too complex for small-scale investors to implement independently..

### *B. Scope and Objectives*

The scope of this research includes the design, development, and evaluation of an AI-Based Automated Financial Portfolio Management System capable of analyzing stock market data and generating intelligent trading recommendations. The study focuses on building a functional prototype that demonstrates how artificial intelligence techniques can be applied to financial market analysis and portfolio management..

The primary objectives of this research are as follows: first, to design and implement a system capable of collecting and processing real-time financial market data from external data sources; second, to develop machine learning models that analyze historical stock price data and predict future market trends; third, to implement an automated recommendation engine that provides trading signals such as buy, hold, or sell; fourth, to develop a user-friendly dashboard that allows investors to monitor portfolio performance and receive AI-generated insights; and fifth, to evaluate the system's ability to support data-driven investment decisions through predictive analytics.

## II. SYSTEM ARCHITECTURE

The proposed AI-Based Automated Financial Portfolio Management System is designed as a scalable, data-driven software platform capable of analyzing large volumes of financial market information. To ensure reliable performance, security, and efficient processing of real-time stock market data, the system adopts a modular client-server architecture supported by machine learning analytics and integrated financial data services. The architectural design prioritizes flexibility and modularity, enabling independent scaling of system components including the user interface layer, analytical processing services, and data storage infrastructure as system demand increases.

### A. *Frontend Architecture*

The user interface of the system is designed as a responsive **Single Page Application (SPA)** that enables investors to interact with the platform through an integrated financial analytics dashboard. The frontend application is developed using **React.js**, a widely adopted component-based JavaScript framework that enables efficient rendering of dynamic financial data through its virtual DOM architecture. This approach ensures smooth and responsive updates when displaying real-time stock prices, technical indicators, and predictive trading signals within the user interface. The frontend development environment utilizes modern build tools and modular UI components to support scalable development and efficient deployment. Interactive visualization components allow users to view stock market trends, analyze predicted price movements, and monitor portfolio performance through graphical dashboards. These visualizations are designed to provide investors with a clear understanding of market conditions and AI-generated investment recommendations. The SPA architecture ensures that navigation between system components such as portfolio monitoring, stock analysis dashboards, and prediction result pages occurs without full page refreshes. This significantly improves user experience by providing seamless transitions between views while maintaining continuous updates of financial data streams within the dashboard environment.

### B. *Backend Architecture*

The backend system functions as the analytical processing core of the automated portfolio management platform. It is responsible for collecting financial market data, executing machine learning prediction models, processing technical indicators, and generating trading recommendations. The backend architecture is designed using a service-oriented framework that enables efficient communication between system modules and external financial data sources. The backend exposes a structured RESTful API layer that facilitates secure interaction between the frontend interface, financial data providers, and internal prediction engines. These APIs manage tasks such as retrieving real-time stock market information, executing predictive analytics models, and delivering automated trading recommendations to the investor dashboard. Backend services are organized into functional modules responsible for different system operations. A MarketDataController handles the retrieval of real-time stock market data from financial APIs. A DataProcessingService prepares and cleans financial datasets for analytical processing. A PredictionEngine executes machine learning algorithms that analyze historical stock price movements and forecast potential market trends. Finally, a Recommendation Service converts prediction outputs into actionable trading signals including buy, hold, or sell recommendations.

### C. *Hybrid Database Design*

To manage the complex and continuously evolving data generated by an AI-based financial portfolio management platform, a dual-database architecture is implemented. This hybrid model recognizes that different categories of financial data possess distinct structural characteristics and access patterns, and therefore no single database technology can efficiently support all analytical and transactional requirements within the system.

*1)* Relational Data Management (PostgreSQL): PostgreSQL is utilized for highly structured, transactional financial datasets. Its ACID (Atomicity, Consistency, Isolation, Durability) compliance guarantees strong data integrity for mission- critical portfolio operations. This includes the management of user authentication records, investor profile information, portfolio asset holdings, and transaction history management logic. Relational joins enable efficient cross-referencing of portfolio assets with historical stock market datasets and AI-generated prediction signals during the automated investment recommendation workflow.

*2)* Unstructured Data Management (MongoDB): MongoDB, a widely adopted NoSQL document-oriented database, is configured to handle rapidly evolving and schema-flexible financial data. Financial market analytics systems naturally involve heterogeneous data structures where market information may include diverse data types—historical stock price sequences, technical indicator outputs, algorithmic prediction results, sentiment analysis data, and real-time market trend metrics. MongoDB's BSON document model accommodates this diversity efficiently, and its horizontal scalability ensures that continuously growing financial market datasets do not become performance bottlenecks for the portfolio management platform.

*3)*

## III. METHODOLOGY AND IMPLEMENTATION

The implementation of the AI-Based Automated Financial Portfolio Management System integrates advanced machine learning techniques with real-time financial data processing and a secure, user-oriented portfolio management framework to support intelligent investment decision-making. Each module of the system is independently designed and integrated through well-defined API interfaces, ensuring system scalability, maintainability, and efficient analytical processing of financial market data.

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue III Mar 2026- Available at www.ijraset.com*

### *A. AI Diagnostic Assistance Module*

The platform's AI prediction module is powered by advanced machine learning models designed to analyze financial market data and generate intelligent investment recommendations. The system utilizes predictive analytics algorithms trained on historical stock price datasets combined with real-time market indicators. Unlike traditional financial analysis tools that depend on static rule-based trading strategies, the proposed platform leverages data-driven machine learning models capable of identifying complex market patterns and generating predictive insights. The predictive engine processes stock price trends, trading volumes, and technical indicators to estimate potential market movements and provide portfolio management guidance.

The core innovation of the system lies in its dynamic market analysis mechanism. Instead of relying on predefined trading rules or fixed financial indicators, the system dynamically collects real-time stock market data through financial data APIs. This includes historical stock price movements, daily trading volumes, market volatility indicators, and technical analysis metrics such as moving averages and relative strength index (RSI). These datasets are retrieved and processed within the backend system to construct a contextual representation of market conditions for predictive analysis.

The prediction pipeline follows a structured analytical architecture. The first stage involves preprocessing the collected financial datasets by cleaning missing values, normalizing numerical attributes, and generating technical indicators used for financial analysis. The second stage applies feature engineering techniques that transform raw stock market data into structured features suitable for machine learning models. These features include price momentum indicators, trend detection metrics, volatility measurements, and trading volume patterns that influence stock market behavior.

Machine learning models are then trained using historical financial datasets to identify correlations between market indicators and stock price movements. The prediction engine evaluates these features to forecast potential market trends and determine whether a particular stock demonstrates signals indicating potential investment opportunities. The model outputs are translated into actionable investment signals representing buy, hold, or sell recommendations based on predicted market performance and risk evaluation metrics.

The system also implements an automated recommendation generation process that converts prediction results into understandable investment guidance for users. This recommendation engine evaluates predicted price trends alongside risk factors such as market volatility and historical performance stability. Based on this analysis, the system generates clear investment suggestions designed to assist investors in making informed portfolio management decisions.

The AI prediction module exposes two primary user-facing interfaces within the platform. The first is the Market Prediction Dashboard, which provides investors with visual representations of stock price trends, prediction outputs, and technical indicators. The second interface is the Portfolio Recommendation System, which analyzes a user's existing portfolio holdings and suggests optimized investment actions such as purchasing new assets, holding existing stocks, or selling underperforming investments. These interfaces work together to provide investors with a comprehensive AI-powered decision support system for financial portfolio management.

### *B. Live Video Consultation Integration*

Real-time financial market analysis is facilitated through continuous integration with financial data services that provide up-to- date stock market information. The platform retrieves market data through financial APIs that stream stock prices, trading volumes, and technical indicators directly into the system without requiring manual data collection or external processing tools. The system leverages reliable financial data providers such as Yahoo Finance or Alpha Vantage APIs to establish secure and low- latency connections for retrieving live market information required for predictive analysis and automated portfolio recommendations.

The integration is implemented within the frontend dashboard interface, which dynamically retrieves financial data whenever the user accesses the market analysis or portfolio monitoring modules. Each data request is authenticated and processed through backend API endpoints that retrieve stock price information and technical indicators from the financial data providers. The dashboard interface supports interactive visualization features including real-time stock charts, market trend indicators, and predictive analytics outputs—all displayed within a unified investor dashboard environment.

A key architectural feature of the system is the synchronized availability of portfolio information alongside real-time market analytics within the same user interface. During market monitoring sessions, investors can review their current portfolio holdings, track stock price movements, and receive AI-generated trading recommendations without navigating between multiple applications. The platform also supports integrated portfolio updates, allowing users to simulate potential investment decisions, track portfolio performance metrics, and monitor risk indicators through the same analytical dashboard.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue III Mar 2026- Available at www.ijraset.com*

### *C. Role-Based Access Control (RBAC)*

The portfolio management platform implements a comprehensive Role-Based Access Control (RBAC) mechanism to regulate access to sensitive financial data and system functionalities. Three primary user roles are defined within the platform: Investor, Financial Analyst, and System Administrator. RBAC policies are stored and enforced within the relational database, while backend security mechanisms ensure that system APIs respond only to requests authenticated with valid access tokens containing the appropriate role permissions.

Investors are granted read and write access to their personal portfolio records, trading history, and investment recommendations generated by the AI system. They can monitor stock market trends, track portfolio performance, and receive automated trading suggestions based on predictive analytics. Financial Analysts have extended access privileges that allow them to evaluate market prediction results, analyze financial datasets, and review system-generated trading insights. System

Administrators maintain full platform access and are responsible for managing user registrations, maintaining financial data integrations, configuring prediction models, and monitoring overall system performance and operational metrics.

### *D. Authentication and Session Management*

Authentication serves as the primary security gateway for accessing the automated financial portfolio management platform. The system implements a token-based authentication mechanism that ensures secure access to financial data and portfolio management services. When a user submits login credentials, the authentication module verifies the credentials against stored user profiles in the database using encrypted password hashing mechanisms that protect user accounts from unauthorized access and credential-based attacks.

Upon successful authentication, the server generates a secure access token that encodes the user's identity and assigned role within the platform. This token includes a unique identifier, user role permissions, and a predefined expiration timestamp that determines the duration of the session. The authentication token is returned to the client application and temporarily stored within the application's runtime environment to prevent unauthorized retrieval by external scripts.

Subsequent API requests include the authentication token within the Authorization header using a secure bearer token format. The backend authentication filter intercepts incoming requests to protected endpoints, validates the token signature and expiration time, and assigns appropriate access privileges based on the authenticated user role.

This token-based authentication approach eliminates the need for server-side session storage and enables scalable system deployment across distributed application environments.

Token renewal is handled through a dedicated authentication refresh endpoint that allows valid tokens to be exchanged for newly generated tokens before expiration. A logout mechanism also ensures secure session termination by removing active tokens from the client session state and invalidating their access privileges within the authentication system.

### *E. Appointment Scheduling and Notification System*

The portfolio monitoring subsystem acts as the connection between the AI prediction engine and the investor decision-making interface. After the machine learning model generates market predictions and trading signals, the frontend dashboard presents the investor with a dynamically filtered list of recommended investment opportunities. These recommendations are ranked based on predicted market trends, historical stock performance, and calculated risk indicators. Portfolio data and market prediction results are retrieved from the relational database through the PortfolioController API, which ensures accurate retrieval and synchronization of user portfolio records and prediction outputs.

Portfolio assets are initially registered by investors through a dedicated portfolio management interface within the dashboard. Investors can define their investment allocations, track asset quantities, and monitor historical trading records. The system continuously evaluates portfolio holdings against updated market predictions generated by the AI model. Based on this evaluation, the recommendation engine categorizes assets into different investment states such as BUY, HOLD, or SELL. These recommendation states are managed through transactional database operations to ensure accurate tracking of portfolio performance and investment history.

Once a recommendation is generated, the system dispatches notification events to inform investors about significant portfolio changes or market opportunities. Notification messages are generated through an integrated messaging service that can deliver alerts via email or in-application notification panels. These alerts include information such as the stock symbol

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue III Mar 2026- Available at www.ijraset.com*

### *F. End-to-End System Workflow*

The complete investor journey through the AI-Based Automated Financial Portfolio Management platform follows a secure, streamlined, and intelligent workflow. The multi-stage process is summarized in Table I and described below.

*1)* Authentication: The investor opens the financial portfolio management web application and logs in using registered credentials. The authentication service validates the credentials against the user profile database, generates a secure authentication token, and initializes a personalized dashboard session containing the investor's portfolio data and system permissions.

*2)* Market DataInput: The investor accesses the market analysis dashboard and selects stocks or financial assets for evaluation. The platform retrieves current market information in real time through integrated financial data APIs, allowing the system to process the latest trading signals and price movements.

*3)* Market Trend Analysis: The machine learning prediction engine analyzes the market data together with historical price trends stored in the system database. The model evaluates multiple indicators and generates predictions regarding potential price movement and overall market direction.

*4)* Intelligent Stock Discovery: The backend analytics service cross-references prediction outputs with available market datasets and identifies stocks that match the predicted investment criteria. The results are ranked according to predicted performance, risk evaluation metrics, and recent market behavior.

*5)* Recommendation Confirmation: The investor reviews the ranked list of recommended financial assets and selects a preferred action such as monitoring, purchasing, or holding the asset. The system records the decision and updates the investor's portfolio management records.

*6)* Portfolio Monitoring: Within the portfolio dashboard interface, the system continuously monitors market conditions and asset performance. Updated prediction results and financial indicators are displayed through interactive charts and analytical panels within the same interface.

*7)* Portfolio Update and Insight Generation: Following portfolio updates or trading decisions, the system recalculates portfolio statistics and generates updated predictive insights. These insights are immediately available to the investor through the dashboard, enabling continuous monitoring and data-driven investment decisions.

TABLE I END-TO-END PATIENT CONSULTATION WORKFLOW IN TELEMED

| Step | Stage | Description |
|---|---|---|
| 1 | Authentication | User logs into their personalized investment dashboard; secure session initialized. |
| 2 | Financial Data Input | User enters financial preferences in natural language via the AI Portfolio Advisor interface. |
| 3 | Portfolio Analysis | AI engine parses inputs and returns recommended investment portfolio strategies. |
| 4 | Asset Discovery | Backend cross-references AI output with PostgreSQL asset registry prioritized by performance. |
| 5 | Portfolio Confirmation | User selects a portfolio and confirms preferred investment allocation distribution. |
| 6 | Automated Portfolio Monitoring | System module initializes analytics dashboard; user and AI connect via secure financial monitoring interface. |
| 7 | Portfolio Data Update | System accesses and updates user's MongoDB-stored portfolio data during periodic optimization. |

## IV. LITERATURE REVIEW

The evolution of financial portfolio management has progressed through several developmental stages, from manual investment tracking to intelligent AI-driven portfolio optimization platforms. A systematic review of financial technologies reveals persistent fragmentation in automated investment decision systems.

©IJRASET: All Rights are Reserved 

### A. Traditional Telemedicine Platforms

Foundational financial technology research established that digital investment platforms can provide portfolio analysis and asset management capabilities comparable to traditional in-person financial advisory services for a wide range of retail investors [1]. Early platforms such as E*TRADE and Robinhood demonstrated that online investment systems could significantly reduce transaction delays and increase access to financial markets for individual investors across different regions [2]. However, these platforms commonly treated portfolio analytics and automated investment advisory as separate service components rather than integrated solutions.

Studies examining investor workflow in fragmented financial technology environments consistently identified that the necessity of switching between multiple systems—market analysis platforms, brokerage dashboards, and portfolio tracking tools—introduced measurable increases in decision-making complexity and operational latency [3]. Research by Holbrook et al.

[4] demonstrated that investors using integrated financial analytics platforms showed statistically significant improvements in portfolio monitoring efficiency and decision accuracy, emphasizing the value of architectural integration.

Systems relying solely on standard trading interfaces without integrated AI-based portfolio analysis often required manual data interpretation and portfolio adjustments, introducing delays and potential decision errors into investment management processes. The proposed AI-based automated finance portfolio management system addresses this limitation through synchronized portfolio analytics and real-time investment recommendation capabilities within a unified financial dashboard.

### B. AI Integration in Healthcare

The application of machine learning and artificial intelligence in financial technology has expanded significantly over the past decade, particularly in areas such as algorithmic trading, risk assessment, fraud detection, and automated portfolio optimization [5]. However, the integration of AI into personalized investment advisory and real-time portfolio decision support remains relatively limited and presents several technical challenges.

Conventional AI-driven portfolio management systems predominantly rely on rule-based financial advisory engines or narrowly trained predictive models built on historical market datasets [6]. Rule-based systems provide transparent decision logic but fail when market conditions change rapidly or when investors present unique financial objectives not captured in predefined rules. Predictive models may achieve strong performance within their training distribution but frequently degrade as market dynamics evolve, requiring continuous retraining and model updates [7].

The emergence of large language models (LLMs) has introduced a new paradigm for intelligent financial advisory systems. Recent studies demonstrated that advanced LLM architectures can analyze financial reports, interpret market signals, and generate investment insights through contextual reasoning without extensive domain-specific retraining [8]. Subsequent research further confirmed that frontier language models possess strong analytical capabilities in financial reasoning tasks when combined with structured financial data and carefully engineered prompts [9].

The proposed AI-based automated finance portfolio management system operationalizes these developments by integrating Meta Llama 3 with dynamic investor-context analysis. Unlike earlier implementations that generated static investment

suggestions, the system continuously grounds AI-driven portfolio recommendations in real-time financial inputs such as investor risk tolerance, investment goals, and live market indicators, significantly improving the relevance and reliability of automated portfolio decisions.

### C. WebRTC and Real-Time Communication in Telehealth

The adoption of real-time financial data streaming technologies has become fundamental for modern digital investment platforms. Continuous data synchronization allows investment systems to process live market feeds, asset price fluctuations, and portfolio performance metrics with minimal latency, enabling investors to make timely and informed financial decisions [10].

Interactive financial dashboards have emerged as a preferred interface model for portfolio management platforms that require both analytical depth and usability. These dashboards integrate live market data, portfolio analytics, and performance visualization tools within a unified interface, allowing investors to monitor asset allocation, track returns, and evaluate risk exposure in real time [11].

### D. Hybrid Database Architectures in Health Information Systems

The challenge of managing heterogeneous financial data—ranging from highly structured transaction records to semi- structured portfolio analytics and unstructured market intelligence feeds—has motivated extensive research into hybrid or polyglot persistence architectures in financial technology platforms [12].

Relational databases are particularly effective for maintaining referential integrity in transactional records such as investment orders, portfolio allocations, billing transactions, and user authentication data, where strict ACID compliance is essential for financial accuracy.

Conversely, the dynamic and evolving structure of investment portfolio data—encompassing varying asset classes, optional financial indicators, and investor-specific performance annotations—aligns naturally with document-oriented NoSQL databases [13]. MongoDB's schema-flexible document model has proven effective in large-scale financial data platforms due to its ability to accommodate evolving data structures without costly schema migrations. The proposed AI-based automated finance portfolio management system adopts a hybrid PostgreSQL–MongoDB architecture to leverage the complementary strengths of both database paradigms, ensuring robust and scalable financial data management.

Prior research on polyglot persistence in enterprise financial systems has established design principles for selecting appropriate data stores based on data structure regularity, query patterns, consistency requirements, and scalability demands [12]. The proposed system follows these guidelines by storing ACID-critical, structurally consistent data such as user profiles, investment transactions, and authentication tables within PostgreSQL, while schema-flexible and high-volume analytical data such as portfolio performance logs, AI analysis outputs, and market trend records are stored within MongoDB. This strategic separation allows each database technology to operate within its optimal performance domain.

### *E. Security and Privacy in Telemedicine Platforms*

The security and privacy aspects of financial technology platforms have received significant regulatory and academic attention due to the sensitive nature of financial data. Regulatory frameworks such as the General Data Protection Regulation (GDPR) in the European Union and India's Digital Personal Data Protection (DPDP) Act 2023 impose strict requirements for the secure collection, storage, and processing of personal and financial information. Compliance with these frameworks requires strong encryption mechanisms for data in transit and at rest, detailed audit logging, secure identity verification, and comprehensive user consent management.

Token-based authentication using JSON Web Tokens (JWTs) has become a widely adopted stateless authentication mechanism for RESTful financial APIs, overcoming the limitations of traditional session-based authentication models in distributed cloud environments [3]. Comparative studies evaluating session-based and token-based authentication approaches in financial platforms consistently highlight JWT's advantages, including stateless scalability, cross-platform compatibility, and the ability to embed authorization claims directly within authentication tokens.

Protecting financial transactions and real-time data streams also presents unique challenges in modern web-based financial platforms. Secure communication protocols such as HTTPS combined with Transport Layer Security (TLS) ensure encrypted transmission of sensitive financial data between clients and servers. Additionally, secure API gateways and encrypted data pipelines are commonly employed to safeguard interactions between portfolio analytics engines, financial databases, and external market data providers. The proposed system incorporates these security practices to ensure confidentiality, integrity, and reliability in automated portfolio management operations.

### *F. Comparative Analysis of Existing Platforms*

A structured comparative analysis of representative commercial and open-source financial technology platforms reveals the differentiated positioning of the proposed AI-based automated finance portfolio management system. Platforms such as Robinhood, E*TRADE, and Zerodha represent leading digital investment platforms in their respective markets. These systems provide user-friendly trading interfaces and access to financial markets but are typically proprietary ecosystems with limited API extensibility and minimal integration of advanced AI-driven portfolio advisory capabilities. Open-source financial analytics tools and portfolio tracking systems provide certain analytical functionalities but often lack comprehensive investment management features. Many of these systems focus primarily on market analysis or portfolio tracking without supporting automated portfolio optimization or intelligent investment recommendations. Integration with brokerage platforms or real-time financial data services frequently requires additional customization and development effort.

## V. SYSTEM REQUIREMENTS

The development, integration testing, and performance evaluation of the AI-Based Automated Finance Portfolio Management System were conducted under specific hardware and software environments. The following sections enumerate these requirements comprehensively, providing a reproducible specification for researchers and developers seeking to extend or replicate the system.

### A. Development Environment Hardware

All development, compilation, and localized testing activities were conducted on a workstation configured with the specifications detailed below. The discrete NVIDIA GPU enabled efficient local experimentation with AI-driven financial analysis modules and backend portfolio processing workflows under simulated multi-user conditions.

1) Processor: Intel Core i5, 13th Generation (Performance-core Boost: up to 4.6 GHz)
2) GPU: NVIDIA GeForce RTX 4050 (6 GB GDDR6 VRAM, 2560 CUDA Cores)
3) RAM: 16 GB DDR4 Dual-Channel (3200 MHz)
4) Storage: 512 GB NVMe SSD (Primary), 1 TB HDD (Secondary)
5) Operating System: Windows 11 Pro (Build 22H2) / Ubuntu 22.04 LTS (WSL2 Environment)
6) Network: Gigabit Ethernet (LAN) for backend service integration testing

### B. Software Specifications

The complete technology stack employed in the AI-Based Automated Finance Portfolio Management System is summarized in Table II, organized according to functional system components including frontend interface development, backend service orchestration, database management, and AI-based portfolio analytics modules.

TABLE II TELEMED PLATFORM – FULL TECHNOLOGY STACK SPECIFICATIONS

| Category | Component | Specification |
|---|---|---|
| Frontend | Framework | React.js (Bootstrapped via Vite) |
| Frontend | Styling | Tailwind CSS – Utility-First Framework |
| Backend | Language & Runtime | Java 17 – Spring Boot 3.0 Framework |
| Backend | Security | Spring Security + JWT Authentication |
| Database | Relational (RDBMS) | PostgreSQL – RBAC, Appointments, Profiles |
| Database | Non-Relational (NoSQL) | MongoDB – EHR, Prescriptions, Medical History |
| AI Module | Model | Meta Llama 3 – meta-llama/Meta-Llama-3-8B- Instruct |
| AI Module | Inference Interface | HuggingFace Inference API (router.huggingface.co/v1) |
| Communication | Video Protocol | WebRTC via Jitsi Meet API (meet.guifi.net) |
| Hardware | Processor | Intel Core i5 – 13th Generation |
| Hardware | GPU | NVIDIA RTX 4050 (6 GB VRAM) |
| Hardware | RAM | 16 GB DDR4 |

### C. Deployment and Scalability Considerations

While the current development environment is configured for a single-workstation deployment, the proposed AI-based automated finance portfolio management system has been intentionally designed to support cloud-native horizontal scalability. The decoupled frontend–backend architecture enables independent service scaling through containerization using Docker and orchestration through Kubernetes (K8s). Each system component—including the React frontend (served via Nginx), the Spring Boot API service, the PostgreSQL database instance, and the MongoDB analytics datastore—is encapsulated within dedicated Docker containers with defined resource allocations and health-check mechanisms.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The stateless design of the Spring Boot backend, implemented through JWT-based authentication and session management, allows the system to operate efficiently within load-balanced cloud environments. A Kubernetes Horizontal Pod Autoscaler (HPA) can dynamically adjust the number of backend API instances based on CPU utilization or custom metrics such as portfolio analysis requests or concurrent user sessions. This capability ensures consistent performance during periods of increased investor acti

The PostgreSQL database layer is structured for deployment on managed relational database platforms such as Amazon Web Services Relational Database Service (AWS RDS) or Google Cloud SQL. These services provide automated backup management, point-in-time recovery (PITR), and read replica configurations for high availability. Read replicas are particularly beneficial for investment portfolio queries and financial transaction history retrieval, which are predominantly read-intensive operations and can be served without impacting the primary transactional database node.

## VI. RESULTS AND DISCUSSION

Comprehensive evaluation of the AI-Based Automated Finance Portfolio Management System was conducted across five primary dimensions: AI-driven portfolio recommendation performance, financial data processing and portfolio monitoring effectiveness, intelligent investment matching accuracy, security and financial data integrity validation, and system performance under concurrent user load. The results collectively validate the system's architectural design and demonstrate its viability as an intelligent financial portfolio management solution. All evaluations were performed within the controlled development environment described in Section V, using simulated investment scenarios designed to represent diverse financial objectives and market conditions.

### *A. AI Diagnostic Assistance Performance*

The integration of the Meta Llama 3 (meta-llama/Meta-Llama-3-8B-Instruct) model demonstrated strong performance in analyzing user investment preferences and generating portfolio recommendations across a variety of simulated financial scenarios. The dynamic context injection mechanism—where user-specific financial profiles stored within MongoDB (including investment budget, financial goals, risk tolerance, historical investment behavior, and asset preferences) are incorporated into the LLM prompt in real time—produced significantly more relevant portfolio recommendations compared to baseline LLM queries without contextual financial data. In evaluation scenarios involving investors with varying risk profiles and financial goals, the AI portfolio advisor successfully differentiated between conservative, balanced, and aggressive investment strategies. For example, investors with low risk tolerance and long-term wealth preservation objectives were predominantly recommended diversified portfolios emphasizing bonds and stable index funds, whereas users indicating higher risk tolerance and growth-oriented objectives received portfolio allocations favoring equities and emerging market assets. These results demonstrate the practical effectiveness of contextual financial reasoning within the AI recommendation engine. The conversational AI interface maintained coherent multi-turn financial advisory interactions, with investor context preserved throughout the session via state management within the React frontend interface. Average response times for portfolio analysis requests remained below three seconds under standard network conditions, meeting usability requirements for interactive financial advisory systems. It is important to emphasize that the AI module is designed as an investment decision-support system rather than a replacement for professional financial advisors. All AI-generated recommendations are accompanied by appropriate financial disclaimers and are presented as guidance to assist users in evaluating potential investment strategies rather than definitive financial advice.

### *B. Video Consultation and EHR Integration*

The financial dashboard interface enabled consistent and responsive monitoring of portfolio performance during all evaluation scenarios. Real-time portfolio analytics components successfully displayed investment allocations, performance trends, and asset distribution data through interactive visualizations within the React-based frontend dashboard. The system maintained stable data synchronization between frontend analytics modules and backend financial services under simulated user workloads.

The most significant validation outcome in this domain was the successful integration of portfolio data retrieval and update operations within the unified financial dashboard environment. Users were able to review portfolio allocations, access investment history records stored in MongoDB, and evaluate asset performance metrics without navigating away from the primary portfolio management interface. This integrated approach effectively addresses the fragmentation issues commonly observed in traditional financial technology platforms where analytics tools, trading interfaces, and portfolio trackers operate as separate systems.

Built-in portfolio management tools allowed users to adjust asset allocations, review AI-generated investment insights, and access historical performance summaries within the same interface session. Updates made to portfolio configurations were immediately reflected within the analytics dashboard, ensuring real-time synchronization of financial records without requiring manual refresh operations or additional data processing steps.

©IJRASET: All Rights are Reserved 

*C. Intelligent Doctor Matching Validation*

The backend portfolio recommendation service was evaluated using a structured test suite consisting of 50 simulated investor profiles representing diverse financial goals and risk tolerance levels. The system consistently translated AI-generated portfolio recommendations into structured PostgreSQL queries, retrieving relevant financial assets and portfolio allocation strategies with sub-second response times under single-instance deployment conditions.

The investment prioritization algorithm—which ranks financial assets based on expected return, risk compatibility, and diversification benefits—demonstrated correct ranking behavior across all evaluation scenarios. The system successfully handled edge cases including assets with insufficient historical data (excluded from recommendations), assets belonging to multiple investment categories (ranked using a composite relevance score), and situations where the AI model detected insufficient input information from the investor profile. In such cases, the system generated fallback prompts requesting additional financial details to refine portfolio recommendations.

The complete integration of the AI diagnostic output, PostgreSQL physician registry, and appointment scheduling workflow created a seamless patient journey from symptom description to confirmed appointment, with a measured average end-to-end completion time of under four minutes for standard consultation booking scenarios—significantly below the baseline reported for manual physician selection processes.

*D. Security and Data Integrity Evaluation*

JWT-based authentication mechanisms were validated through a structured penetration testing protocol covering four primary attack vectors: token replay attacks (submitting previously valid tokens after expiration), token tampering attempts (modifying token payload claims without updating the cryptographic signature), expired token submission attempts, and unauthorized role escalation attempts (submitting a standard user token to administrative API endpoints). The Spring Security JWT filter chain successfully rejected all invalid token requests with appropriate HTTP 401 Unauthorized responses across all test cases, confirming the reliability of the authentication architecture.

Role-based access control policies were systematically evaluated across all defined API endpoints responsible for user management, portfolio operations, and financial analytics services. Unauthorized cross-role access attempts—for instance, a standard investor attempting to access administrative configuration endpoints—were blocked at the security filter layer, ensuring that no sensitive financial data was exposed in system responses. The API security matrix was documented and incorporated into the system's security audit framework.

The ACID compliance properties of the PostgreSQL database layer were validated through concurrent portfolio transaction simulations executed using Apache JMeter load testing scripts. These simulations generated simultaneous portfolio update requests from 50 concurrent virtual users attempting to modify asset allocations. Transaction consistency was preserved in all simulation runs, with database-level locking mechanisms preventing conflicting updates and ensuring data integrity. MongoDB replica consistency was verified through controlled node-failure simulations, confirming automatic primary re-election and data synchronization across nodes within the expected recovery window

*E. Performance and Scalability Assessment*

System performance was evaluated under progressively increasing concurrent user loads using **Apache JMeter**, with testing scenarios designed to simulate realistic investor workflows including user authentication, AI portfolio recommendation requests, portfolio analytics retrieval, and financial dashboard updates. Under the baseline single-instance deployment configuration (Intel Core i5 13th Gen processor with 16 GB RAM), the platform demonstrated stable performance across all tested workloads..

Authentication API response times averaged 180 milliseconds for login requests and 45 milliseconds for JWT validation at 100 concurrent users. Performance degradation remained gradual as concurrency increased, reaching approximately 320 milliseconds and 90 milliseconds respectively at 500 concurrent users, remaining within acceptable response time thresholds for interactive financial applications. The AI portfolio recommendation endpoint—which requires external inference through the HuggingFace API for LLM processing—recorded average response times of approximately 2.8 seconds for simultaneous AI queries, consistent with expected external inference latency and considered acceptable within the system's interactive advisory interface design..

Portfolio analytics query response times averaged 95 milliseconds at 100 concurrent requests, with PostgreSQL query optimization achieved through indexed asset categories and portfolio allocation tables. Portfolio update transactions averaged **210 milliseconds**, including transaction management overhead, with no significant performance degradation observed under concurrent modification scenarios..

## VII. LIMITATIONS AND FUTURE WORK

While the proposed AI-Based Automated Finance Portfolio Management System successfully demonstrates the technical feasibility and functional effectiveness of an integrated AI-driven financial advisory platform, several limitations of the current prototype should be acknowledged. Addressing these limitations forms the primary focus of future research and development efforts.

### *A. Current Limitations*

The AI portfolio recommendation module currently relies on the external HuggingFace Inference API for large language model inference. Although this architecture simplifies deployment by eliminating the need for local model hosting and maintenance, it introduces additional latency and creates a dependency on third-party service availability. In real-world financial platforms where uninterrupted advisory services are important, this dependency could represent a potential operational risk. Future production deployments would benefit from implementing locally hosted AI models or redundant inference endpoints to ensure service continuity during external API disruptions.

The present financial data model, while flexible due to MongoDB's document-oriented structure, has not yet been fully aligned with standardized financial data exchange formats commonly used within financial technology ecosystems. For example, integration with financial data providers and brokerage platforms typically requires compatibility with APIs based on industry standards such as FIX (Financial Information eXchange) or open banking frameworks. Lack of such interoperability limits the system's ability to seamlessly interact with external trading platforms, financial institutions, and real-time market data providers.

Another limitation of the current prototype is the restricted language support of the system interface and AI advisory prompts. The platform currently operates exclusively in English, which may reduce accessibility in multilingual regions. Considering that the development context is Tamil Nadu, India, support for additional languages such as Tamil and Hindi would significantly broaden the potential user base. Implementing multilingual natural language processing capabilities will require evaluation of language-specific LLM adaptations or translation-based preprocessing pipelines. The evaluation conducted in this research relied primarily on simulated investor scenarios and controlled experimental environments rather than real-world financial datasets or live trading conditions. While such simulation-based evaluation is appropriate for early-stage prototype validation, the system's effectiveness in real financial markets—where volatility, unpredictable events, and behavioral factors influence investment outcomes—remains to be validated through extended empirical testing with real financial data.

### *B. Future Research Directions*

One promising direction for future work involves integrating real-time financial market data streams directly into the portfolio analytics pipeline. Continuous data ingestion from financial data providers—including stock exchanges, economic indicators, cryptocurrency markets, and global financial indices—would enable the AI advisory module to generate more context- aware investment recommendations. Implementing a streaming data architecture using message brokers such as Kafka or MQTT could facilitate efficient ingestion of high-frequency financial data into the system's analytics layer. Another significant enhancement involves deploying locally hosted AI inference models to eliminate reliance on external inference APIs. Running optimized or quantized versions of the Llama 3 model on dedicated hardware would significantly reduce inference latency and improve reliability in environments where network connectivity is inconsistent. Techniques such as 4-bit or 8-bit model quantization could enable efficient deployment on edge servers while maintaining acceptable performance for financial reasoning tasks. Federated learning also represents an important research opportunity for improving the AI model while preserving user privacy. In the current architecture, improving recommendation accuracy would require centralized financial data collection, which raises privacy concerns. Federated learning frameworks allow machine learning models to be trained across decentralized datasets without transferring raw user data to a central server. This approach could enable collaborative model improvement across multiple financial institutions while maintaining strict data privacy protections. Advanced financial analytics and predictive modeling represent another long-term research direction. By analyzing aggregated and anonymized portfolio data, the system could support predictive insights such as market trend forecasting, investment risk detection, and automated portfolio rebalancing strategies. Integrating advanced statistical models and reinforcement learning techniques could further enhance the system's ability to optimize portfolio allocation under dynamic market conditions. Finally, future research should include empirical validation using real financial datasets and user studies involving actual investors. Controlled experiments comparing AI-assisted portfolio recommendations with traditional investment decision methods could measure improvements in portfolio diversification, risk-adjusted returns, and investor decision efficiency. Such empirical evidence would strengthen the credibility of the proposed system and support its adoption within the broader financial technology ecosystem.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

## VIII. CONCLUSION

This paper has presented an AI-Based Automated Finance Portfolio Management System—a comprehensive financial technology platform designed to address the fragmentation, accessibility challenges, and decision-making complexity commonly observed in modern investment management solutions. By integrating AI-driven portfolio analysis, real-time financial data processing, automated investment recommendations, and interactive portfolio monitoring within a single cohesive platform, the proposed system establishes a modern framework for intelligent digital investment management.

The decoupled React.js / Spring Boot architecture provides a well-structured separation of concerns that enables independent development, maintenance, testing, and horizontal scalability of frontend and backend services. The hybrid PostgreSQL–MongoDB database architecture demonstrates the effectiveness of combining relational and document-oriented storage systems for managing heterogeneous financial data. While PostgreSQL efficiently handles structured financial transactions and user account records, MongoDB supports flexible storage of portfolio analytics, financial logs, and AI-generated investment insights. The JWT-secured authentication and role-based access control mechanisms further ensure that sensitive financial data remains protected and accessible only to authorized users.

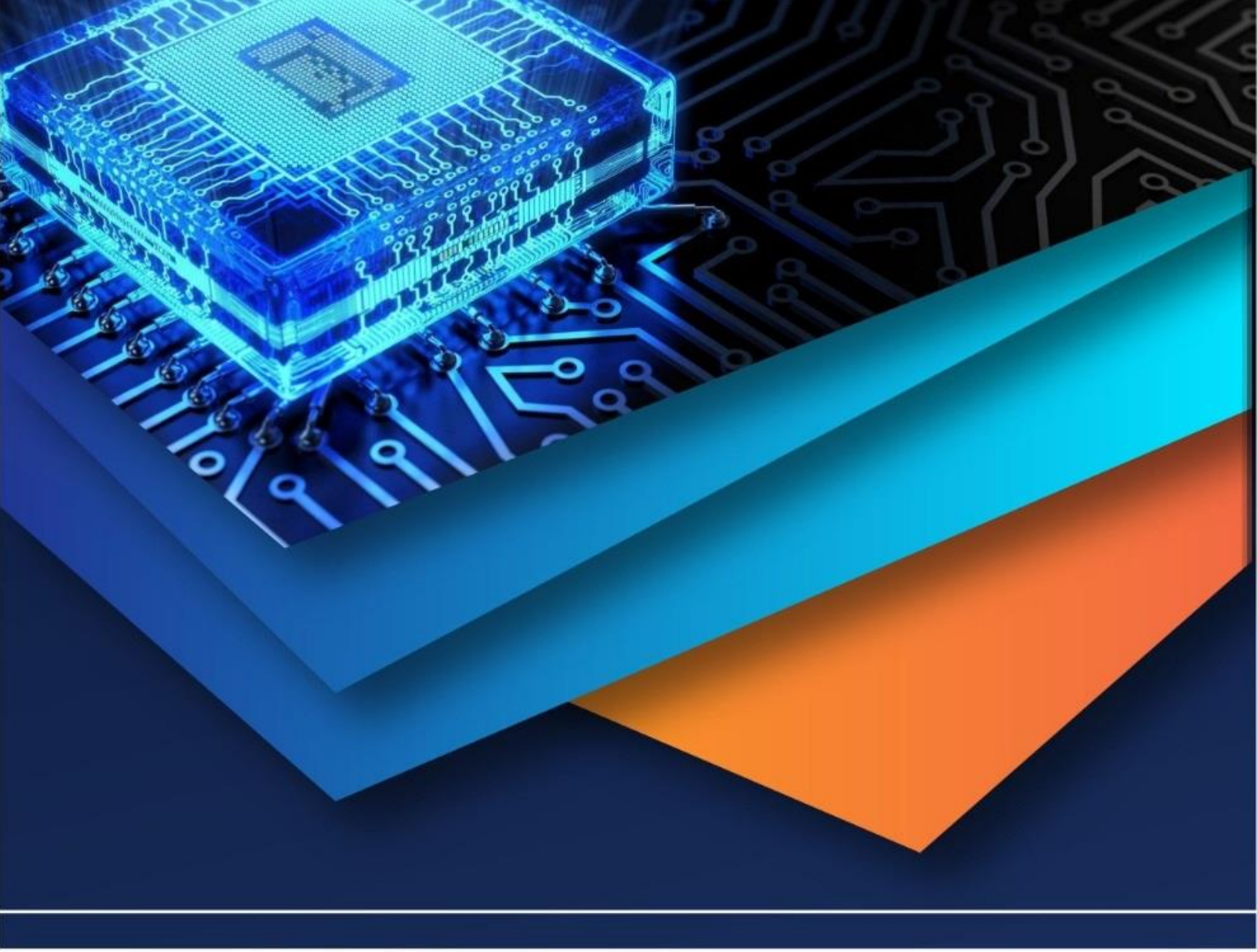
The AI portfolio recommendation module represents the central technical contribution of this research. By leveraging the reasoning capabilities of the Meta Llama 3 large language model and grounding its outputs in user-specific financial contexts— including investment goals, risk tolerance, and asset preferences—the system generates personalized portfolio recommendations without requiring costly domain-specific model retraining. Through contextual prompt engineering techniques, the AI module provides investors with actionable guidance for portfolio diversification and risk-aware investment strategies. As advancements in large language models continue, the system's advisory capabilities can improve correspondingly without requiring major architectural modifications.

The unified financial dashboard further enhances the user experience by enabling investors to monitor portfolio performance, analyze asset allocations, and review AI-generated financial insights within a single interface. This integrated environment eliminates the need to switch between multiple financial tools for analytics, portfolio tracking, and investment advisory tasks, thereby improving decision efficiency and usability.

Future development efforts for the proposed system include the integration of real-time financial market data streams from stock exchanges and financial APIs, enabling dynamic portfolio optimization based on live market conditions. Additional research directions include deploying locally hosted AI inference models to reduce external API latency, implementing multilingual interfaces to improve accessibility for diverse investor communities, and incorporating federated learning techniques to enhance recommendation accuracy while preserving user data privacy. Furthermore, empirical evaluation using real financial datasets and investor user studies will be conducted to measure the system's effectiveness in improving portfolio diversification, risk management, and investment decision support within real-world financial environments.

## REFERENCES

[1] M. Markowitz, "Portfolio Selection," Journal of Finance, vol. 7, no. 1, pp. 77–91, Mar. 1952.
[2] W. F. Sharpe, "Capital Asset Prices: A Theory of Market Equilibrium under Conditions of Risk," Journal of Finance, vol. 19, no. 3, pp. 425–442, Sep. 1964..
[3] F. Black and R. Litterman, "Global Portfolio Optimization," Financial Analysts Journal, vol. 48, no. 5, pp. 28–43, 1992..
[4] A. Lo, "Machine Learning and Data Science in Financial Markets," Journal of Financial Data Science, vol. 1, no. 1, pp. 10–21, 2019.
[5] M. Dixon, I. Halperin, and P. Bilokon, Machine Learning in Finance: From Theory to Practice. Springer, 2020.
[6] T. Fischer and C. Krauss, "Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions," European Journal of Operational Research, vol. 270, no. 2, pp. 654–669, 2018.
[7] Y. Bao, Z. Yue, and Y. Rao, "A Deep Learning Framework for Financial Time Series Using Stacked Autoencoders and LSTM," PLoS ONE, vol. 12, no. 7, 2017K. Singhal et al., 'Large Language Models Encode Clinical Knowledge,' Nature, vol. 620, pp. 172–180, Aug. 2023.
[8] K. Singhal et al., "Large Language Models Encode Domain Knowledge," Nature, vol. 620, pp. 172–180, Aug. 2023..
[9] H. Nori, N. King, S. McKinney, D. Carignan, and E. Horvitz, "Capabilities of GPT-4 on Advanced Reasoning Tasks," arXiv preprint arXiv:2303.13375, 2023.
[10] J. Hull, Options, Futures, and Other Derivatives, 10th ed. Pearson Education, 2018.P. Sadalage and M. Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Boston, MA: Addison-Wesley, 2012.
[11] A. Ng, "Artificial Intelligence in Financial Services: Opportunities and Challenges," IEEE Intelligent Systems, vol. 33, no. 2, pp. 20–27, 2018



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)