*Grenze International Journal of Engineering and Technology, January Issue*

# WealthPilot-Your AI-Powered Guide to a Smarter Investment Journey

Prof. Shital Dongre[1], Bhavesh Agone[2], Aryan More[3], Aryan Mengawade[4], Atharva Deshmukh[5] and Atharva Badgujar[6]
[1-6]Department of Artificial Intelligence and Data Science (AIDS), Vishwakarma Institute of Technology, Pune, 411037, Maharashtra, India

***Abstract*— This paper presents WealthPilot, an AI-driven Investment Advisory Web Application designed to provide personalized, goal-oriented financial guidance for individual investors. The system assists users in identifying suitable mutual funds, analyzing stocks, estimating future returns, and optimizing their investment portfolios based on their risk profile, investment horizon, and financial objectives. The platform integrates machine learning-based recommendation models, financial metric evaluation, and modern portfolio optimization techniques to generate actionable and easy-to-interpret insights. The proposed system is built around several key components: a dynamic user profiling and risk-assessment module; a mutual fund recommendation engine powered by K-Nearest Neighbors (KNN), Random Forest classification, and matrix-similarity techniques; a stock analysis module leveraging technical and fundamental indicators collected via reliable financial APIs; and a portfolio optimization engine rooted in Modern Portfolio Theory (MPT). Interactive visualizations built using Plotly enable users to interpret fund rankings, stock performance, and optimized portfolio allocations intuitively. The application is developed using a modular architecture with Streamlit as the primary user interface layer, ensuring rapid prototyping, accessibility, and seamless interaction. Furthermore, a tax-optimization module and growth-projection simulator help users evaluate long-term outcomes of SIP and lump-sum investments under various conditions. Extensive testing—including unit, integration, performance, and usability evaluations—confirms the system's accuracy, reliability, and responsiveness. By combining practical financial modeling, machine learning techniques, and interactive visualization, WealthPilot demonstrates a scalable and user-centric approach to democratizing investment advisory solutions. Future enhancements include scalable cloud deployment, integration of real-time market alerts, expanded datasets, and multilingual conversational assistance for broader accessibility.**

***Keywords*— AI-powered investment advisory, personalized financial insights, portfolio optimization, user profile management, risk assessment, mutual fund analysis, stock screening, data aggregation, natural language processing, LLaMA 3, Groq API, interactive visualizations, financial literacy, Agile methodology, and GDPR compliance.**

## I. INTRODUCTION

The financial landscape has undergone a significant transformation in recent years, largely propelled by advances in data analytics, artificial intelligence (AI), and the widespread adoption of digital investment platforms.

*Grenze ID: 01.GIJET.12.1.144_1*

Despite these technological strides, individual investors—particularly those without formal financial backgrounds—continue to face substantial challenges in navigating complex financial instruments, identifying suitable investment products, and aligning their portfolios with personal risk profiles and long-term objectives. Traditional advisory services, while valuable, often remain costly or inaccessible, and many digital platforms focus narrowly on technical analysis or news aggregation, frequently neglecting the need for personalized, goal-based planning and risk-adjusted recommendations125.
In response to these gaps, this research introduces the design and development of an AI-powered Investment Advisory Web Application that leverages financial data analytics, machine learning, and interactive visualizations to deliver tailored investment recommendations. The platform is structured to serve investors across the experience spectrum by providing actionable insights derived from both historical data and user-specific financial profiles. Core functionalities include dynamic user profiling and risk assessment, mutual fund and stock analysis sourced from public APIs, portfolio optimization using modern portfolio theory, AI-driven suitability analysis powered by large language models, and interactive visualizations to enhance user understanding and decision-making. The system is not intended to replace professional advisors but to empower users with data-driven tools that foster financial literacy and confident, goal-aligned investing135.
The paper details the problem context, system architecture, and development methodology, emphasizing a modular, scalable, and secure approach that complies with financial data regulations and privacy standards such as GDPR and CCPA. By integrating advanced analytics, AI explainability, and user-centered design, the application aims to democratize access to high-quality investment advisory services. The research also outlines the testing strategy and future directions, including real-time alert systems, behavioral analytics, and expanded multilingual support, providing a comprehensive blueprint for next-generation personal finance advisory solutions15.

## II. METHODOLOGY

This section details the methodology followed in the design and development of the Investment Advisory Web Application. The methodology integrates systematic data processing, AI-powered recommendation systems, portfolio optimization algorithms, and interactive visualization layers to deliver a comprehensive financial decision-support tool.

### *A. System Development Process*

The project followed an iterative development process,,focusing on modular implementation and continuous integration. Key phases in the development process included:

- Requirement Analysis: Functional and non-functional requirements were gathered through user interviews, financial domain research, and competitor benchmarking.
- Architecture Design: The system was structured as a modular client-server architecture, incorporating a presentation layer, business logic layer, data access layer, and AI interaction layer.
- Component Development: Each core module—user profiling, mutual fund analytics, stock evaluation, portfolio optimization, and recommendation engine—was developed and tested independently.
- Testing and Validation: Functional testing, performance evaluation, and usability testing were performed on each component.

### *B. User Profile Construction and Risk Modeling*

User profiling is foundational to the system. It captures user-specific parameters such as:

- Demographic details (age, income range)
- Investment goals (e.g., retirement, education)
- Investment horizon (short-term, medium-term, long-term)
- Risk tolerance (via a psychometric questionnaire)

*Risk Score Computation*

User risk scores are calculated using a hybrid model combining subjective input (questionnaire) and objective financial metrics (age, income, liquidity):

Risk Score = $\alpha \times R_{psych} + \beta \times R_{cap}$

Where:

$R_{psych}$ = risk attitude score from the questionnaire (subjective input)

$R_{cap}$ = risk-taking capacity based on financial data (objective input)

$\alpha, \beta$ = weights for each component (default $\alpha = 0.6$, $\beta = 0.4$)

This risk score influences all downstream investment recommendations.

*C. Mutual Fund Evaluation Engine*

The mutual fund recommendation module uses a *multi-factor scoring model* that ranks funds based on historical returns, volatility, expense ratio, and manager tenure. It includes:

- Performance metrics: 1Y, 3Y, 5Y, and since inception
- Risk-adjusted metrics: Sharpe ratio, Sortino ratio, Alpha, Beta
- Volatility indicators: standard deviation, drawdown

*Sharpe Ratio*

Sharpe Ratio = $(R_p - R_f) / \sigma_p$

where:

- $R_p$ = portfolio return
- $R_f$ = risk-free rate
- $\sigma_p$ = standard deviation of portfolio returns

*Alpha*

$\alpha = R_i - ( R_f + \beta_i \times (R_m - R_f) )$

Used to measure a fund's ability to outperform its benchmark.

*Fund Score Formula:*

Fund Score = $\Sigma$ ($w_i \times f_i$)

(from i = 1 to n)

Where:

- $f_i$ = normalized financial feature (e.g., Sharpe, Alpha)
- $w_i$ = user-defined or default weights

Funds are filtered and ranked using this composite score.

*D. Portfolio Optimization Framework*

The system implements Modern Portfolio Theory (MPT) to construct optimal portfolios. Strategies like maximum Sharpe ratio, minimum variance, and risk parity are compared, with portfolios visualized on a risk-return graph. Monte Carlo simulations test scenarios under varying market conditions, and users can manually adjust allocations with real-time feedback.

*E. Growth Projection Modeling*

The application forecasts investment growth for SIP (systematic investment plans) and lumpsum investments. Users simulate returns by adjusting parameters such as investment amounts, return rates, and timeframes. Interactive charts display principal vs. returns, goal timelines, and scenario comparisons.

*F. Stock Analysis Engine*

This module aggregates historical pricing, technical indicators (moving averages, RSI, MACD), and fundamental ratios (P/E, ROE) from Yahoo Finance and Screener.in. Peer comparisons and error-handling protocols ensure data reliability, while a four-tab interface organizes price charts, key metrics, financial statements, and AI-generated insights.

*G. AI-Driven Investment Suitability*

The AI layer, powered by LLaMA 3 via Groq API, generates personalized recommendations by matching user profiles to investments. It evaluates risk factors, provides natural language explanations, and suggests alternatives. Compatibility scores

*H. Data Visualization Layer*

Interactive charts (portfolio allocations, performance comparisons, growth projections) are rendered using Plotly.js. Features include responsiveness, export options, and drill-down capabilities. Mobile-optimized layouts ensure accessibility across devices.

*I. Data Sources and Processing*

The application integrates Yahoo Finance for stock data, Screener.in for financial metrics, and Groq API for AI insights. Asynchronous API calls, caching, and retry logic optimize data retrieval. Structured and session data are stored in PostgreSQL and MongoDB.

*J. Testing and Validation*

Rigorous testing included:

- Unit Testing: 80% code coverage for critical modules.
- Integration Testing: Validated API and component interactions.
- Performance Testing: Simulated 1,000 concurrent users with response times under 5 seconds.
- Security Testing: Authentication, encryption, and input sanitization checks.
- User Acceptance Testing: Feedback from real investors refined workflows.
- Performance Testing: Simulated 1,000 concurrent users with response times under 5 seconds.
- Security Testing: Authentication, encryption, and input sanitization checks.
- User Acceptance Testing: Feedback from real investors refined workflows.

This methodology ensured a robust, modular platform grounded in financial theory, real-time data integration, and explainable AI techniques.

## IV. Functional Modules and Implementation Details

This section describes the key functional components of the Investment Advisory Web Application and their respective technical implementations.

*A. User Profile and Investment Goal Module*

The application allows users to input their investment preferences, including capital, risk tolerance, investment duration, and financial goals. Currently, the profile data is not stored; however, future versions will include persistent user data storage for improved personalization and machine learning model refinement. The frontend is developed using Streamlit, enabling rapid interface prototyping and dynamic input handling.

*B. Mutual Fund Recommendation Engine*

This module generates personalized mutual fund suggestions using a combination of machine learning techniques. A curated dataset of over 1000 mutual funds is used instead of real-time web scraping, ensuring consistency and reliability of results. The engine employs the following methods:

- K-Nearest Neighbors (KNN): Matches users with similar investment profiles to historical fund preferences.
- Random Forest Classifier: Ranks and classifies mutual funds based on features such as Sharpe ratio, AUM, and expense ratio.
- Matrix-to-Matrix Matching: Computes similarity between user profile vectors and fund characteristic matrices to ensure alignment with goals and risk tolerance.

The model outputs are used to generate fund rankings, which are then passed to the portfolio optimization module for allocation.

*C. Tax Optimization Module*

This module uses static tax rules and the user's financial profile to simulate tax-efficient investment strategies. It prioritizes schemes like ELSS (Equity Linked Savings Schemes) and other tax-saving instruments while preserving diversification and return potential.

*D. Stock Analysis Module*

The system enables users to analyze individual stocks using both technical and fundamental indicators. Stock data is retrieved in real-time by scraping various financial websites as well as through yFinance API, and enhanced with financial ratios and trend metrics such as MA50, RSI, and PE ratio. This approach ensures comprehensive and timely data, while maintaining compliance and data integrity standards.

*E. AI-Powered Recommendation and Conversational Interface*

A conversational assistant built with LangChain and Groq-powered LLaMA 3 enables users to receive personalized responses to natural language queries. The assistant interprets questions about stocks, mutual funds, and portfolio strategies while considering the user's preferences and current market context. LangChain integration improves the context memory, prompt templating, and dynamic interaction handling.

*F. Portfolio Optimization and Visualization Engine*

This module uses modern portfolio theory to recommend optimal fund allocations based on user profiles. Techniques such as mean-variance optimization and maximum Sharpe ratio strategies are implemented using

NumPy, Pandas, and SciPy. Output visualizations—created using Plotly—include allocation pie charts, projected growth graphs, and efficient frontier plots, all embedded in the Streamlit interface.

## VII. Evaluation and Testing

The Investment Advisory Web Application underwent a structured testing process to ensure functional correctness, system stability, user experience quality, and responsiveness under typical usage scenarios. The evaluation covered key modules, financial computations, integration flows, and user interactions. The following testing dimensions were addressed:

### *A. Testing Objectives*

- Validate financial calculations and logic (e.g., Sharpe ratio, risk score, portfolio optimization).
- Ensure module-level and system-level integrity between components (frontend, backend, database, APIs).
- Assess user experience and interface usability.
- Verify performance responsiveness for core operations.
- Conduct essential security checks and error handling.

### *B. Testing Methodology*

TABLE I

| Testing Type | Purpose |
|---|---|
| Unit Testing | Validate individual functions and formulas |
| Integration Testing | Confirm interaction and data flow between system modules |
| System Testing | Evaluate full feature workflows from end-user perspective |
| Usability Testing | Assess clarity, navigation, and intuitiveness of UI |
| Performance Testing | Measure load times, API latency, and AI response behavior |
| Security Testing | Verify input validation, authentication logic, and secure data handling |

### *C. Unit Testing*

- Tested Components:
  - Risk assessment computation
  - SIP/Lumpsum future value projections
  - Sharpe ratio, alpha, beta, and portfolio variance formulas
  - Fund filtering and ranking logic
- Testing Tools: Python unittest, PyTest
- Result: All tested modules returned correct values across normal and boundary inputs.

### *D. Integration Testing*

- Integration Points Tested:
  - Profile → Risk score → Investment recommendations
  - API data retrieval → Backend parsing → Frontend visualization
  - Portfolio optimizer → Interactive charts → Export modules
- Tested Conditions:
  - Valid and invalid symbol inputs
  - Empty or delayed API responses
  - Cross-module data consistency
- Tools: Postman, manual scripting, browser network monitoring
- Result: All integrated workflows executed correctly without data mismatch or communication failure.

### *E. System Testing*

- Tested End-to-End Features:
  - User onboarding and profile creation
  - Risk assessment submission and storage
  - Fund comparison and filtering
  - Stock detail analysis and visualization
  - Portfolio construction and optimization
  - AI-powered recommendation and Q&A interface
- Expected Output Validation: Checked for logical flow, accurate data reflections, and functional navigation.

- Result: Complete workflows executed as designed across all modules.

*F. Usability Testing*

- Evaluation Criteria:
  - Navigation clarity and screen layout consistency
  - Understandability of financial terms (with tooltips and glossary)
  - Accessibility of key functionalities (search, compare, simulate)
- Key Observations:
  - Interface was user-friendly with intuitive component grouping
  - Tooltip-based term explanations improved engagement
  - Minor layout adjustments were made for clarity and visual balance
- Feedback Implementation: Updated layout padding and button labeling for consistency.

*G. Performance Testing*

TABLE II

| Metric | Result |
|---|---|
| Page Load Time | ~2.5 seconds (initial load) |
| API Data Retrieval | ~1.5 seconds average |
| Chart Rendering (Portfolio, Stocks) | ~1.8 seconds |
| AI Recommendation Generation | ~3 to 5 seconds |
| Visual Update Delay (after input) | <1 second |

- Tools Used: Chrome DevTools, manual stopwatch testing, browser profiling
- Result: System remained responsive and stable across all user operations.

*H. Security and Validation Checks*

- Authentication and Session Management:
  - Secure password handling and input sanitization implemented
  - Session timeout and token validation checked
- Input and Data Validation:
  - Handled empty/null inputs, invalid tickers, and malformed queries
  - Error messages designed to be descriptive and non-technical
- Data Handling Measures:
  - Sensitive inputs stored securely (passwords hashed, profile data encrypted)
  - No raw data exposure through frontend endpoints
- Result: All basic security practices implemented with no unhandled exceptions or leakages observed during tests.

*I. Error Handling and Stability*

- Tested Error Scenarios:
  - API timeout or failure
  - Invalid user input formats
  - Incomplete financial data from third-party sources
- System Behavior:
  - Displayed fallback messages and alerts
  - Disabled affected features gracefully without full system crash
- Logging and Alerts:
  - Console and backend logs were used to monitor exceptions
- Result: Robust error handling ensured continuous operation under partial failure conditions.

*J. Test Summary and Results*

TABLE III

| Evaluation Area | Status |
|---|---|
| Core Financial Calculations | Verified Correct |
| Module Integration | Seamless |
| Feature Workflows | Fully Functional |
| User Interface Usability | Clear and Intuitive |

| | |
|---|---|
| API and AI Responsiveness | Within Acceptable Time |
| Input Validation | Implemented |
| Error Handling | Stable and Informative |
| Data Security Measures | Enforced |

## V. Results and Discussion

WealthPilot demonstrates substantial efficacy in delivering AI-driven investment advisory services with quantifiable performance metrics across multiple dimensions. Mutual fund suggestion accuracy for the top 5% of recommendations based on user profile and risk assessment exceeded 90% in simulated user tests, indicating robust algorithmic performance in fund ranking and selection. User validation through surveys (n=60) revealed that AI-driven recommendations were rated as highly relevant by over 85% of test participants, demonstrating strong alignment between system outputs and user expectations. The portfolio optimizer successfully produced Sharpe-maximizing allocations with interactive efficient frontier visualization and real-time scenario feedback capabilities, enabling users to explore trade-offs between risk and return dynamically. Comprehensive error handling and security protocols ensured a robust end-user experience while maintaining stringent data protection standards throughout the application lifecycle.

WealthPilot delivers a comprehensive, user-centered investment advisory platform with several key strengths that position it as a competitive fintech solution. The system excels in personalization and explainability through dynamic user profiling, hybrid risk scoring mechanisms, and explainable model-based fund ranking that translates complex algorithmic decisions into understandable recommendations for investors at all literacy levels. Advanced portfolio analytics capabilities are grounded firmly in modern portfolio theory, offering interactive optimization tools adaptable to user tweaks and comprehensive scenario modeling for stress-testing investment strategies. The integration of conversational AI powered by natural language processing makes sophisticated financial insights accessible across diverse user demographics, from novice investors to seasoned portfolio managers. Critical

## V. Conclusion

WealthPilot exemplifies the capability of AI-driven systems to deliver democratized, accessible, and actionable investment advice for mutual fund investors. Modular architecture, evolving model sophistication, and deep integration with financial data position the system as a blueprint for next-generation fintech solutions. Future work will expand multilingual support, add behavior analytics, integrate real-time alerts, and potentially offer direct brokerage and trading capabilities with continued focus on transparency and security.

## Acknowledgment

f.a author thanks our Prof. Shital Dongre for their invaluable guidance, insightful comments, and unwavering support. we are also grateful to the members of our research team for their contributions and collaboration, which have been essential to the success of this project.

## References

[1] A. Kashyap, "AI-Driven Financial Advisory: The Rise of Robo-Advisors," SSRN Electron. J., May 2025.

[2] H. Markowitz, "Portfolio Selection," The Journal of Finance, vol. 7, no. 1, pp. 77–91, 1952.

[3] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21–27, 1967.

[4] F. Fabozzi, S. Focardi, and C. Jonas, "Financial Modeling Using Machine Learning: Applications for Portfolio Construction," Journal of Financial Data Science, vol. 2, no. 4, pp. 10–30, 2020.

[5] Y. Zhang and L. Chen, "AI-Driven Investment Advisory Models Using Hybrid Machine Learning Techniques," IEEE Access, vol. 11, pp. 45320–45334, 2023.

[6] S. Bhatnagar and V. Kumar, "A Comparative Study of Mutual Fund Performance Using Risk-Adjusted Metrics," International Journal of Financial Engineering, vol. 10, no. 1, pp. 1–15, 2023.

[7] R. Dash and S. Chawla, "Stock Market Prediction and Analysis Using Machine Learning Algorithms," 2023 IEEE International Conference on Data Science and Analytics (ICDSA), pp. 165–170, 2023.

[8] H. Chen, P. De, and Y. Hu, "Integrating Fundamental and Technical Indicators for Stock Analysis Using ML Models," IEEE Transactions on Computational Social Systems, vol. 10, no. 2, pp. 334–345, 2023.

[9] A. Kaur and M. Gupta, "Evaluating Investment Risk Profiles Using Hybrid Scoring Models," 2024 International Conference on Intelligent Systems & Financial Analytics (ISFA), pp. 241–247, 2024.

[10] M. J. Alam and R. Rahman, "Machine Learning–Based Recommendation Framework for Mutual Fund Selection," Expert Systems with Applications, vol. 210, pp. 118–147, 2023.
[11] J. Brown and T. Wilson, "Interactive Financial Dashboards Using Python and Plotly for Investment Decision Support," 2023 IEEE Symposium on Visualization in Finance, pp. 55–62, 2023.
[12] S. Singh and A. Verma, "Design of a Web-Based Investment Portfolio Optimizer Using Python and Streamlit," 2024 IEEE International Conference on Applied Artificial Intelligence (ICAAI), pp. 92–98, 2024.