Multidisciplinary Journal of Research in Engineering and Technology

ISSN: 2348-6953

Volume 13 Issue 2s (2026) | Pages: 191-204

Intelligent Investment Analysis and Asset Allocation System

Nilima Ramteke[1], Jayashree Prasad[2], Sneha Gupta[3], Deepak Yadav[4], Chetanya Rander[5], Aryan Jitender Bhardwaj[6]

[1,3,4,5,6]Department of Computer Science and Engineering, MIT School of Computing, MIT ADT University, Pune, India
[2]Artificial Intelligence and Analytics; Department of Computer Science and Engineering, MIT School of Computing, MIT ADT University, Pune, India
[1]nilima.ramteke@mituniversity.edu.in, [3]snehagupta00887@gmail.com, [4]yadavdeepak4123@gmail.com, [5]chetanyarander7@gmail.com, [6]aryanbhardwaj9054@gmail.com

Peer Review Information

Type: *Article*
Received: *27 March 2026*
Revised: *12 April 2026*
Accepted: *26 May 2026*
Published: *16 June 2026*

Abstract

The fast development of financial tech- nology platforms. has added to financial markets involvement, particularly among. first-time and young investors. Nonetheless, there are numerous people who are lacking. financial literacy to do informed investment. decisions. Through this, investors tend to turn to informal sources. as peer advice like social media or peer advice in investment selection. strategies. In this paper, an Intelligent Investment Analysis is proposed. Portfolio Sys- tem which offers individualized portfolio. user financial profile- based recommendations. The system evaluates the factors like income, age, amount invested, determinable risk tolerance and investment horizon to establish appropriate asset. divestment between such financial instruments as stocks, mutual. debt instruments, funds and exchange traded funds. The suggested system implements diversification principles inferred. as given in the Modern Portfolio Theory [8] to minimize investment risk and enhance sustainability of finances. The system is evaluated through the various investor profiles in terms of safeness, mod- erate, and risky violent types. Results demonstrate consistent, explicable portfolio allocations whose goals are feasible casually. test using Monte Carlo simulation, validation of the system. transparency of recommendation and functional correctness. The system will enhance the financial literacy hence increase data- driven investment decision making by the retail investors within the system. Indian market.

Keywords: Asset Allocation, Investment Analysis, Portfolio Management, Financial Technology, Intelligent Systems, Robo-Advisory, Monte Carlo Simulation.

How to Cite This Article

Ramteke, N., Prasad, J., Gupta, S., Yadav, D., Rander, C., & Bhardwaj, A. J. (2026). Intelligent investment analysis and asset allocation system. *Multidisciplinary Journal of Research in Engineering and Technology*, 13(2s), 191–204.

Introduction

There is a significant role of financial investment in long- term. custodial prosperity and economic stability. With the advancement of digital trading platforms and financial tech- nology applications, now people can have a more convenient access to financial markets, and an extensive variety of investment. Financial applications, online brokers and digital advisory. The investment process of investing in stocks, mutual has been made easy by tools. Money, and other financial instruments [1]. These technological developments notwith- standing, there are a number of investors, deficient comprehen- sion of how to make structured investment. strategies. Novice investors will often be speculative, social media recommenda- tions or peer advice, information. when selecting investments [6]. These methods usually result in lack of diversification in portfolios and higher level of financial risk. One of the most well-known aspects is the portfolio diversification, best mea- sures of mitigating risk in investments. Based on the theory of Modern Portfolio put forward by Markowitz. By, investors can maximize the performance of their portfolio. allocating investments into a variety of asset classes and. trading off portfolio risk and expected return. In recent years, financial technology has started incorporating intelligent systems capa- ble of analyzing financial information automatically. Advanced computational approaches enable systems to assess market indicators, financial indicators and. investment restrictions to help investors make, better financial decisions [14]. Intelligent asset management systems are increasingly being developed to support portfolio construction and asset allocation by us- ing algorithmic analysis of financial parameters. Moreover, according to recent studies, it is important to use. combinations of computational intelligence applications within the financial field. decision support systems [15]. Such intelligent systems are able to, help investors by processing financial features, find diversification possibilities, and create structured investment solutions. As a result, automated financial advisory tools are becoming an important element in the staff of contemporary

financial tech. The Intelligent Investment Analysis is proposed in this research. and Asset Allocation System created to aid amateur, investors in making informed financial decisions. The system evaluates the financial profiles of investors and does a diversification, pre-determined riskthose allocations based on risk in a portfolio. investment objectives. Aurelian capital is an intelligent system that is proposed. investment planning software meant to help users make decisions when planning their finances. It involves the following components of the system, a frontend, made with Next.js, a backend, made with. Fast API, and a data storage layer with Supabase. The frontend is an interactive dashboard in which users, can enter such financial information as income, investment, age. amount, and financial goals. These are relayed to the, backend via API calls. The inputs are processed by the backend. with a hybrid approach of machine learning and rule-based logic techniques. It calculates the risk profile of the user, it creates a diversified portfolio allocation, and assesses the goal feasibility using Monte Carlo simulation. The system also has one-on-one recommendations formulated as Next Best Actions (NBA). The system guarantees explainability of the system through returns. Explain risk, portfolio allocation, and human readable explanations. financial outcomes. The suggested system is not like the conventional system. By combining a hybrid decision making approach based on combining rule-based financial modeling, advisory platforms [14]. Distribution of allocation through machine learning, and probabilistic. Monte Carlo simulation. Further, the system includes. an elucidate recommendation system, which is known as Next Best. Action (NBA), the actionable insights given to the users. and with argument and possible outcomes. This focus on explainability and probability of outcome model improvement, assurance to the users and enhances financial decision making.

Objectives

The rationale of the proposed system is:

- To analyze investor financial profiles including age, in- come, investment amount, and risk tolerance, and com- pute a numeric risk score via a transparent, threshold based scoring function.
- To provide individualized, four-portfolio assignments. By a seven-step hybrid, (equity, debt, gold, cash) pipeline which is rule-based logic and ML-based fine-tuning.
- To minimize the risk of financial loss by investing in more than one area. modern portfolio theory based portfolios [8].
- To predict goal feasibility probabilistically with Monte. Carlo simulation, which is P10, median, and P90 out outcome percentiles over 10,000 simulation paths.
- To help novice investors make well-informed investor choice with an explainable Next Best Action, engine that offers reasons, action consequences and alternative plans.

Literature Review

Investment advice and financial portfolio. systems have proved to be widely researched in academia and use in industry. Initial preparatory work by The Modern Portfolio Theory (MPT), was introduced by Markowitz [8]. which brought about the principle of diversification and risk-return trade-off. This theory showed that. balanced portfolios based on expected optimal portfolios can be built. with variation, and gives the foundation of the greatest collection of portfolio. allocation models used today.

Sharpe [11] further built on this work with the Capital. Asset Pricing Model (CAPM) which is a mathematical, correlation between expected and systematic risk. CAPM came up with the use of beta to measure market, risk and is a popular financial analysis tool. Nonetheless, MPT and CAPM assume that there are assumptions, including: unusual distributed re- turns and efficient markets, which may but this not be true in practice.

Other studies have since examined the shortcomings of conventional models and made suggestions. Black and Lit- terman [13] proposed a Bayesian model of optimal trading on the markets that is a mixture of the market equilibrium and investor views, allowing for more flexible and realistic allocation strategies. On the same note, Fama [12] suggested the Efficient. Market Hypothesis (EMH) that asserts that the prices of the assets are determined by the laws of supply and demand record all information in the system and hence restrict the capacity. To continuously make excess returns.

Financial technology has over the past years facilitated the creation of automated systems of advisory on invest- ments which is normally known as having automatic elements known as robo-advisors. Studies such as D'Acunto et al.

[14] underscore the applicability of robo-advisors in terms of algorithmic decision-making and user profiling in offering personalized portfolio. recommendations. Accessibility is en- hanced by these systems and minimise human bias yet tend to be not transparent on decision making processes.

Recent research in machine learning-based portfolio op- timization explores techniques to make models such as re- gression models, neural networks and reinforcement learning better. allocation performance [16]. Although these solutions provide flexibility and they need high-compute resources, large datasets, and data-driven. are frequent victims of interpretabil- ity problems.

Unlike strictly data-driven systems, rule-based financial advisory systems to improve transparency and reliability, have been suggested [17]. These Predefined financial heuristics and domain knowledge are used by systems to produce consistent and explainable recommendations. Rule-based systems might not be very adaptable, they are appropriate compared to machine learning models to provide explicit reasoning, for decision-support applications.

Moreover, Monte Carlo-based methods of probabilistic sim- ulation have also been. common in financial planning to model uncertainty in investment returns. [10]. Future portfolio can be estimated by Monte Carlo simulation. Performance in different market conditions, which enables investors to assess the likelihood. of achieving financial goals.

According to a study on behavioral finance, retail investor highlights the fact that retail investors are not able to change their behavior as per the study on behavioral finance when it comes to retailing. Examples of cognitive biases that often impair outcomes include loss aversion, performance. chasing and SIP discontinuation in a market correction. This result encourages the. The design of NBA engine, which explicitly considers the context of behavior generation, in the generation process.

Although new systems have made major strides, the current systems have various limitations such as poor explainability, need to work with very big data, insufficient customization, and inadequate Management of uncertainty [17]. Furthermore, no existing Indian retailer facing consumer platform combines all four elements of it– rule-based risk profiling, ML allocation tuning, Monte Carlo simulation, and an explainable recom- mendation engine - within one system only.

Thus, something is needed, an intelligent investment ad- visory system that is a combination probabilistic simula- tion in structured rule-based decision-making with additional maintenance. openness and user-friendliness. The proposed system is a solution to these gaps by incorporating. Monte Carlo simulation, rule-based allocation of portfolios and risk profiling to get clear and trustworthy advice on investment to novice investors.

Table 1. *Comparative Literature Review*

Study	Method	Contribution	Limitation
Markowitz [8]	Mean-Variance Optimization	Introduced diversification and risk-return trade-off	Assumes static market conditions
Sharpe [11]	CAPM Model	Defines relation between return and systematic risk	Oversimplifies real-world markets
D'Acunto et al. [14]	Robo-advisory algorithms	Automated portfolio recommendation using profiling	Limited transparency and explainability
Jung et al. [15]	ML-based robo-advisor	Improved portfolio performance via machine learning	Interpretability and user trust challenges
Proposed System	Rule-based + ML + Monte Carlo	Explainable, hybrid advisory system for Indian retail investors	Simulated evaluation; no real-data backtesting

Proposed Model

The system that was proposed is Aurelian Capital, a smart investment advisory platform that will help retail people. investors, especially first time and novice investors, in devel- oping evidence-based, structured, financial decisions. Rather as opposed to informal peer advice or even hypothetical so- cial media recommendations, users can get personal portfolio recommendations based on a rigorous multi stage analytical, rule-based financial logic, machine pipeline that is a combi- nation of refinement of learning, probabilistic simulation and deterministic recommendation engine.

This system is based on a three-level cloud-based architec- ture. The frontend front is created in Next.js and stored with Vercel, gives the user interface that investors can interact with. supply their financial profiles and interact with the results. The backend, which was created in Fast API and was hosted on Render. performs all essential financial calculations and yields wholesomely. explainable JSON responses. Supabase is the authentication and storage layer, carrying on user profiles, financial. planning records, goals, and past planning.

Two major planning flows through the system dedicated REST endpoints. The flow of the Investment Plan, which can be found through the /predict endpoint, receives core investor. generates attributes and, based on them, age, monthly income, monthly SIP amount and expected annual return as well as the duration of the investment are generated portfolio allocation that is a risk-classified and confidence-scoring recommendations Next Best Action and. The Goal Plan flow, available at the endpoint /plan, also takes a. financial (name, target corpus, target years, and simulation, priority), conducts a Monte Carlo simulation to analyze goal feasibility, and gives out an estimate of the probability-of-success with actionable guidance. Investors are categorized, depending on the calculated risk score.

Investors are categorized, depending on the calculated risk score. when making the choice we have to transform all the risks into one of the three categories of risk, each of which will carry the symbol a. distinct portfolio allocation strategy:

- Low Risk (Conservative): More to debt. capital preser- vation instruments with moderate level of income genera- tion. Appropriate with short term investors who are older. horizons and conservative expectations of returns.
- Moderate (Medium Risk): equally distributed allocation across. equity and debt, and small amounts of gold and cash. Suit suited to mid-career investors who have medium horizons. and temperate reversion wishes.
- Aggressive (High Risk): long-term wealth generation by the use of growth assets. Suitable to younger long- horizon and high-return investors.

Hybrid Decision Framework

The presented system incorporates a hybrid decision- making. construct which intentionally fuses two complemen- tary. paradigms: deterministic rule-based financial modelling and refinement of assignments by machine learning. The selection of this architecture was to solve a core conflict of intelligent. advisory systems- the trade off of interpretability and adaptability [14].

The rule-based layer defines consistent, auditable base allo- cations which are based on classical financial planning princi- ples. [8]. All the allocations made by this layer are complete. explainable: the system is able to explain the technical reasons why a specific equity-debt split was suggested in terms of the user. age, investment horizon, and the risk level. This transparency is key to gaining user confidence, especially with novice users. investors who are unfamiliar with portfolio construction concepts.

The trained AllocationTuner—the machine learning layer. model — adds flexibility by forecasting constrained adjust- ments of perassets which are overlaid on the rule-based foundation. The - model taking four input features: investment duration, the formal equity percentage which is the calculated percentage of confidence, and investor age. It produces tiny adjustment values of each. asset category that change the distribution to patterns learned. based on training data as well as staying within specified limits. floors. Importantly, the ML model can be considered as a fine-tuning layer instead of the key decision-maker. This design ensures that the suggestions on the system can be interpreted and consistently profitable even when the ML manipulations are. applied.

The system also includes two analytical modules which run without depending on the allocation pipeline. The Monte Carlo simulation engine is used to assess the feasibility of goals through a simulation stochastic path of investments in market (thousands in total). Uncertainty [10], offers a probabilistic instead of a deterministic. than deterministic outlook on their financial future. The Next Best Action (NBA) engine produces profile-based highly deterministic. financial advice in using a rule of sequence to their calculated risk profile, SIP parame- ters, the likelihood of success, and the behavioral context [6]. Both modules are able to provide human readable explanations with their quantitative results.

Mathematical Model

The following section More formally establishes the quantitative basis of the Aurelian Capital system. Each component of the processing pipeline is based on a mathematical definition, enabling the reproducibility, auditing, of all recommendations. and uniform to the same inputs.

Portfolio Allocation Function

Portfolio Allocation Function. The fundamental aim of the system is to establish an optimal. portfolio allocation **P** of a particular investor profile. The allocation is given in a formal manner as a function of four main. investor attributes:

$$P = f(A, I, R, H) \quad (1)$$

where A is the age of the investor, I is his monthly income, R is the constant is the risk tolerance and H is the investment horizon in years.

The portfolio allocation vector P is formulated on four. asset classes:

$$P = (w1, w2, w3, w4) \quad (2)$$

where w1 is the equity weight, w2 is the debt weight, w3 is the gold weight, and w4 is the cash weight. The portfolio weights must satisfy the unity constraint:

$$\sum_{i=1}^{4} w_i = 1, \quad w_i \geq 0 \quad \forall i \qquad (3)$$

The anticipated portfolio yield is calculated as the weighted combination of asset-class performance [8]:

$$E[R_p] = \sum_{i=1}^{4} w_i \cdot R_i \qquad (4)$$

where Ri is the expected annual return of asset class i.

Risk Score Calculation

The risk rate **S** is a figures metric in the range **[0, 100]** which is calculated by adding together. The contribution of factors being points based on four dimensions of the financial of the user profile. The overall shape of risk score is:

$$S = \min\left(\sum_{j=1}^{4} \phi_j(x_j),\ 100\right) \qquad (5)$$

and where the contribution to the point at x of factor j is φj(xj) user's input value xj. The network factors and their threshold-wise contribution functions are defined in Table 2.

Table 2. *Risk Score Factor Contributions*

Factor	**Condition**	**Points**
Age (A)	A < 30 years	+20
	30 ≤ A < 45 years	+10
	A ≥ 45 years	+5
Duration (H)	H > 10 years	+20
	5 ≤ H ≤ 10 years	+10
	H < 5 years	0
Expected Return (Re)	Re > 12%	+25
	8% ≤ Re ≤ 12%	+15
	5% ≤ Re < 8%	+8
	Re < 5%	0
SIP/Income Ratio (Ir)	Monthly SIP > Monthly Income	+20
	Monthly SIP 0.5–1× Monthly Income	+10
	Monthly SIP < 0.5× Monthly Income	0
Maximum Cap	Total score	≤ 100

The risk score S is subsequently mapped to a discrete risk level via the function R(S):

$$R(S) = \begin{cases} \text{Low} & \text{if } S < 30 \\ \text{Medium} & \text{if } 30 \leq S < 65 \\ \text{High} & \text{if } S \geq 65 \end{cases} \quad (6)$$

The intuition behind this scoring scheme is that younger investors (A < 30) with longer horizons (H > 10 years), aggressive return targets (Re > 12%), and high SIP-to-income ratios accumulate the most points and are correctly classified as high-risk. On the other hand, aged investors having short terms, low-risk returns expectations, and simple SIP commitments receive lower scores and can be categorized as low- or medium-risk.

Confidence Score Calculation

The internal consistency and realism of the user is measured by a confidence score *C*. input combination. The risk score, which is indicative of the aggressiveness of the investor, is not used in a way that is analogous to the stress-test. confidence score quantifies on how the inputs are coherent and financial plausible. The score is set to 1. 0 and penalized by the deductions of the penalties, denoted by, δ_k. given any inconsistency found *k*:

$$C = \max\left(0.0,\ \min\left(1.0,\ 1.0 - \sum_k \delta_k\right)\right)$$

(7)

The penalty conditions and their magnitudes are defined in Table 3.

Table 3. *Confidence Score Penalty Deductions*

Condition	**Penalty δ_k**
Expected return > 15%	−0.25
Expected return 10%–15%	−0.15
Duration < 3 years AND expected return > 8%	−0.25
Risk = Low BUT expected return > 7% (mismatch)	−0.20
Risk = High BUT expected return < 6% (mismatch)	−0.15
Income < INR 25,000 AND SIP > INR 5,00,000	−0.15

Confidence score C can be used in downstream in two ways it is a feature of the AllocationTuner ML model as an input feature allows the model to come up with conservative allocations of different profiles and the NBA engine uses it as a filtering criteria to prompt rebalancing suggestions where confidence drastically decreases to a set minimum.

Portfolio Allocation — Seven-Step Pipeline

The portfolio allocation pipeline produces a normalized four- asset weight vector through seven sequential steps. Let $B = (b_e, b_d, b_g, b_c)$ denote the rule-based base allocation for equity, debt, gold, and cash respectively.

Step 1 — Base Allocation by Risk Level. The base allocation is determined by the computed risk level $R(S)$:

$$B = \begin{cases} (0.30, 0.50, 0.10, 0.10) & \text{if } R(S) = \text{Low} \\ (0.50, 0.30, 0.10, 0.10) & \text{if } R(S) = \text{Medium} \\ (0.70, 0.15, 0.10, 0.05) & \text{if } R(S) = \text{High} \end{cases} \quad (8)$$

Step 2 — Duration Tilt. For long-horizon investors, equity allocation is increased to reflect greater capacity to absorb short-term volatility [10]. If $H \geq 10$ years:

$$b_e \leftarrow b_e + 0.05, \quad b_d \leftarrow b_d - 0.05 \quad (9)$$

Step 3 — Machine Learning Adjustment. The AllocationTuner model fML receives a feature vector x = [A, H, Re, C] and outputs bounded per-asset adjustments:

$$\Delta_{ML} = f_{ML}(x) = (\Delta_e, \Delta_d, \Delta_g, \Delta_c) \quad (10)$$

Step 4 — Apply Adjustments

The ML adjustments are added to the duration-tilted base allocation.

$$P' = B + \Delta ML \quad (11)$$

Step 5 — Caps and Floors. Each weight is clipped to prevent extreme or degenerate allocatons

$$p_i' \leftarrow \max(\epsilon_i^{\min}, \min(p_i', \epsilon_i^{\max})) \quad \forall i \qquad (12)$$

Step 6 — Normalization. All four weights are normalized to sum exactly to 1.0:

$$w_i = \frac{p_i'}{\sum_{j=1}^{4} p_j'} \quad \forall i \qquad (13)$$

Step 7 — Explainer Generation. A human-readable portfolio_explained string is created explaining the reasoning behind the individual asset weights, with references to the tilt on a particular duration, and the direction in which the ML is adjusted.

Monte Carlo Simulation

Monte Carlo Simulation, The Monte Carlo simulation finds approximate probability that the current SIP of a user, which is invested within the period of the goal return, will yield the target goal amount-accounting. arbitrarily on volatility of markets [10]. The portfolio value is a stochastic, evolves month-by-month:

$$V_{t+1} = V_t(1 + r_t) + \text{SIP} \qquad (14)$$

where r_t is a random monthly return drawn from a normal distribution:

$$r_t \sim \mathcal{N}(\mu_m, \sigma_m^2) \qquad (15)$$

The mean monthly return μ_m and monthly volatility σ_m are derived from the annualized inputs as:

$$\mu_m = \frac{r_{\text{annual}}}{12}, \quad \sigma_m = \frac{\sigma_{\text{annual}}}{\sqrt{12}} \qquad (16)$$

where σ^{annual} is a risk-level-calibrated annual volatility value. A total of $N^{total} = 10{,}000$ independent simulation paths are run, each iterating over $T = H \times 12$ monthly steps. The probability of achieving the financial goal G is:

$$P_{\text{success}} = \frac{N_{\text{success}}}{N_{\text{total}}} = \frac{\#\{V_T^{(i)} \geq G : i = 1, \ldots, 10000\}}{10000} \qquad (17)$$

The simulation calculates in addition to the probability of success, a set of outcome statistics in totality based on $\boldsymbol{N^{total}}$ terminal values:

$$\text{Median} = Q_{0.50}(\{V_T^{(i)}\}), \quad P_{10} = Q_{0.10}(\{V_T^{(i)}\}), \quad P_{90} = Q_{0.90}(\{V_T^{(i)}\} \qquad (18) \quad (18)$$

where Qp denotes the p-th quantile. Goal feasibility is mapped to a discrete label based on the probability of success:

$$\text{Feasibility} = \begin{cases} \text{HIGH} & \text{if } P_{\text{success}} \geq 0.75 \\ \text{MODERATE} & \text{if } 0.50 \leq P_{\text{success}} < 0.75 \\ \text{LOW} & \text{if } P_{\text{success}} < 0.50 \end{cases} \qquad (19)$$

Next Best Action Engine

The NBA engine is implemented using six rules, which are deterministic to create individual financial recommendations [6]. Let $p = P_{\text{success}}$, S be the monthly SIP, and C be the confidence score.

1. Stay the Course: If $p \geq 0.80$, the engine emits a single "Stay the course" movements, breaks. No further rules are evaluated.

2. Increase SIP: If $p < 0.70$ and $S > 0$:
   - Variable income: suggest $S' = S \times 1.05$
   - Stable income: suggest $S' = S \times 1.10$
   - Constrained by: $S' \leq 0.30 \times I_{\text{monthly}}$
3. Extend Time Horizon: When $p < 0.50$, propose extension the objective within 2-3 years.
4. SIP Rebalancing: If equity weight w_e exceeds the high- equity threshold AND $H < H_{\text{short}}$, OR if $C < C_{\min}$, suggest rebalancing equity toward debt and gold.
5. Risk Alignment: When the anticipated return is signifi- cant less than expected not in accordance with the designated Low or Medium risk level, flag the misalignment and recom- mend realignment of expectations.
6. Portfolio Drift: When the current portfolio allocation is different by the user notably below the suggested allocation, evaluate rebalancing with target allocations.

Every action produced has an associated canonical schema. which includes six fields: action (short title), category (SIP / REBALANCE / RISK / TIMELINE), priority (HIGH / MEDIUM). LOW), rationale (profile-conscious explana- tion), danger of omission.(consequence of inaction), and al- ternatives (other ways of handling the same issue). The limit_actions_by_mode() function filters the entire action list according to the choice of the user.

System Architecture

The Aurelian Capital system is adopted as a stratified Cloud-native architecture comprises of four major levels: the Frontend Presentation Layer, User Interaction Layer, The Data Storage Layer, and Backend Processing Layer. This separation of concerns guarantees that every component of the system could be created, enhanced, and developed in isolation. There is another layer of External Services, which unites third-party platforms to authenticate, discover, and analyze.

User Interaction Layer

The user interface layer is the layer where investors access the system using conventional web or mobile browsers. No native needs to be installed. The system aims at be available on a wide variety of devices to users.conditions of connectivity, in accordance with its target audience of new and novice retail investors.

Frontend Presentation Layer

It uses Next.js to construct the frontend which is run on Ver- cel.edge network, delivering the user interface with very low latency. Authentication of the users is done through the fron- tend.Supabase Auth; display of two main planning interfaces.

Identify the API calls to the REST API to (Investment Plan and Goal Plan forms).the FastAPi back-end; and display of the results dashboard, with interactive portfolio allocation graphs, risk and confidence scoreboard, Monte Carlo results graphics, and Next Best Action cards, with rationale expansions. All core financial computation logic is only in the backend.

Backend Processing Layer

FastAPI is used to implement the backend and is deployed on.Render. It reveals two main REST endpoints -/predict in- vestment planning /plan for goal-based planning and manages all requests, which come in with a completeness of nine stages, by the complete ninestage advice route in Section VII. The backend gives a versioned, fully explainable, JSON reply to each request.

Data Storage Layer

Supabase (PostgreSQL) Supabase (PostgreSQL) is the au- thentication and storage layer. It provides user sign-up, log- in and session management.issue tokens, and maintains user profiles, financial goals and historical plan records to trace longitudinally. The core advice logic is processed in full on the FastAPI back end on every API request, it is necessary to make sure the advice engine is not horizontally scalable and stateless.

External Services

The system combines Supabase Auth to carry out safe user authenentiating *Google Analytics* to oversee usages, and Google, Search engine discoverability with search console. The frontend interacts with these seravelengths and has no interface with them. the advise engine at the back.

Table 4. *System Architecture Layer Summary*

Layer	Technology	Primary Responsibility
User Interaction	Web / Mobile Browser	Investor access point

Frontend	Next.js on Vercel	Authentication, dashboards, visualizations, charts, and NBA recommendation cards
Backend	FastAPI on Render	Monte Carlo simulation, risk engine, asset allocation, and NBA generation
Data Storage	Supabase (PostgreSQL)	Authentication, user profiles, goal history, and investment plan history
External Services	Supabase Auth, Google Analytics	Authentication, analytics, and platform discoverability

Workflow of the System

The processes of the workflow convert raw investor financials to financial advice based on an individual, customizable portfolios via. nine processing stages.

Step 1 — User Interaction and Data Input. The investor makes use of the system via web or mobile browser In the case of Investment Plan flow, the user inputs: age (18-) Age (in years), 70, monthly earnings (INR), SIP amount monthly at least 500), long-run(1-40 years), projections of investment. annual return and an optional profile context that captures. level of experience, risk behavior, stability of earnings, and financial.flexibility. In the case of the Goal Plan flow, the user

also. gives a name of the goal, target corpus in INR, target years, and priority ranking.

Step 2 — Input Validation and Transmission. The frontend transmits inputs to the FastAPI backend as a JSON payload via HTTPS POST. The backend validates against the Pydantic InvestmentRequest or GoalPlanRequest schema. Invalid payloads are rejected before any financial computation is initiated.

Step 3 — Risk Score Computation and Classification. The calculate_risk_score() function produces S [0, 100] per Equation (5). The map_risk() function maps S to Low, Medium, or High per Equation (6).

Step 4 — Confidence Score Computation. The calculate_confidence() function applies penalty de- ductions per Table III and clamps the result to [0.0, 1.0] per Equation (7).

Step 5 — Portfolio Allocation. The generate_portfolio_allocation() function executes the full seven-step pipeline (Equations 8– 14), producing normalized asset weights with a portfolio_explained rationale string in advanced mode.

Step 6 — SIP Breakup Calculation. In advanced mode, calculate_sip_breakup() distributes the user's total monthly SIP across the four asset classes in INR proportionally to the normalized weights.

Step 7 — Monte Carlo Simulation (Goal Plan Flow Only). run_monte_carlo() executes 10,000 stochastic paths per Equations (15)–(16), computing $P_{success}$, median, P10, P90, and feasibility label per Equations (17)–(19). explain_monte_carlo() generates plain-language summary bullets.

Step 8 — Next Best Action Generation. The generate_next_best_actions() function evaluates the six NBA rules described in Section V-F. The limit_actions_by_mode() function filters the generated actions according to the selected mode and the investor's experience level.

Step 9 — Response Assembly and Frontend Visualization. The backend assembles a versioned JSON response containing all computed outputs and transmits it to the fron- tend, which renders interactive allocation charts, Monte Carlo visualizations, and expandable NBA cards.

Experimental Results

A series of simulated investor profiles to the entire spec-trum of risk were used to test the proposed system.tolerance categories. All profiles were made to work out.A unique mix of input on all four risk score factors: age, investment years, expected income and SIP-toincome ratio. The system calculated the risk on each profile. score and classification, confidence score, portfolio allocation,Monte Carlo goal feasibility estimate, and Next Best Action recommendations. Consistency of all results was validated.with repeated identical inputs, establishing the deterministic assuring the deterministic behavior of the other rule-based components and the stability of the ML allocation tuner.

Portfolio Allocation Results

Table 5 shows portfolio allocations that have been produced by the system for Conservative, Moderate, and Aggressive investor profiles with all the adjustments duration tilt and ML tuner have been used and weights equalized

Table 5. *Sample Portfolio Allocations by Investor Risk Category*

Investor Type	Equity	Debt	Gold	Cash
Conservative (Low Risk)	30%	50%	10%	10%
Moderate (Medium Risk)	50%	30%	10%	10%
Aggressive (High Risk)	70%	15%	10%	5%

The conservative profile stresses on capital preserva- tion.with an allocation probability (50% debt) in the form of a debt allocation. a 30% equity exposure to really generate returns and a 10% gold hedging as an anti-inflationary. The aggressive profile maximizes the wealth generation in the long term with 70% equity.allocation, the cash (5%) is minimized at the expense of gold (10 percent) remaining position to cater a certain level of market volatility.

Monte Carlo Simulation Results

Monte Carlo simulation results are given in table VI by a representative profile of investor. Findings confirm that there is an increase in. monthly SIP and extending investment horizon are the two best treatments to adopt as per goal achievement probability Risks aggressive profiles had stronger median terminal portfolio values and also much larger between. P10 and P90 results, proving the basic risk-return trade-off. The distributions of outcome around lower median values were tighter with conservative profiles, which makes sense capital-preservation orientation of high-debt allocations.

Table 6. *Monte Carlo Simulation Outcomes by Investor Profile*

Profile	SIP	Horizon	Target	Psuccess	P10	P90
Conservative	5,000	10 yr	8L	61%	6.1L	10.2L
Moderate	10,000	15 yr	40L	78%	22.4L	41.8L
Aggressive	15,000	20 yr	1Cr	82%	48.3L	1.14Cr

Note: Values derived from system simulation runs. L = Lakh, Cr = Crore (INR).

Confidence Score Validation

The score mechanism of confidence was tested across.profiles that were created to evoke each condition of the penalty. Profiles specifying anticipated returns of more than 15% were heavily endowed.penalties ($\delta = -0.25$), which represents the unrealistic. nature such expectations in the Indian market setting.Profiles with risk- return differences such as Low- risk classification plus above 7% return expectations- were rightfully punished ($\delta = 0.20$) and the resulting less-so confidence scores also led to suitable NBA recommendations to reset the expectations of returns.

Next Best Action Engine Validation

NBA engine was tested on determinism, profileawareness and rule sequencing correctness. Across all tested profiles, the same inputs made the same action lists on. recursive appeals, attesting to the engine.behavior. An awareness of profiles was confirmed that variable-income profiles were recommended increased at a more conservative rate (on the upside capped at 5% increase) than stableincome profiles (10% increase) and Beginner experiencelevel profiles were provided with simpler action language than Expert profiles. Rule sequencing was confirmed by verifying that profiles of Psuccess = 0.80 or above raised the Stay-the-Course rule and yielded no more actions.

Explainability Output Validation

Human-readable explainer outputs—risk explained, monte caro generated portfolio explaineds and plain language Monte Carlo summarys of all the evaluated profiles were created.The explainers in all instances have talked about the specific ap- propriately.input values that did the calculated outputs the ex- plainability layer not only captures the underlying computation well, but also generates boilerplate-free text instead of generate boilerplate. This affirms that the system meets its design intention to achieve provision of clear user-understandable financial advice.

Discussion

The Aurelian Capital is confirmed to be in accordance with the experiment results. system meets its main design goals: to create structured, individualized and explainable investment. suggestions ranging across the spectrum of retail investors. of levels of risk tolerance.

- Effectiveness of the Hybrid Framework: The Hybrid Framework is effective. The hybrid decision- making model resolves a historic conflict of smart financial advisory systems. Purely rule-based systems have full inter- pretability as well as limited. adaptability; purely data-driven systems contain adaptability however. sacrifice transparency [17]. The given architecture addresses. this tension by giving each paradigm to that task which it best performs. suited for. High-level structurally is dealt with by rule-based layer. significant distribution choices; the ML layer is concerned with finergrained corrections. This design option is in line with the findings. by D'Acunto et al. [14], who note that lack of transparency a major obstacle to purely algorithmic advisory systems is. to user trust.
- Probabilistic Modeling and User Expectation Setting: Probabilistic Modeling and expecting users. Monte Carlo simulation is also a form of integration. better material pre- diction than the deterministic projection models. appearance in numerous consumer-friendly financial planning instruments [10], which project a single value of corpus as though it were. a certainty. The three level classification of feasibility of the system. High Goal + Low Investor Expectations (HIGH, MODerate and Low): gives investors a practical, visually intuitive frame of the feasibility of their goal in the absence of. soggifying them with uncultured statistical output.
- Explainability as a First-Class Design Principle: A First-Class Design Principle. Another difference with the proposed system is that explainability is not an after consid- eration but the first-class design requirement that is embedded across the whole processing pipeline. [17]. A human readable explanation of every quantitative output can be found and is combined with the specific input values. Moving the calculated value. This multi-layer explainability architecture develops trust on the part of users and enhances financial literacy. aiding users to comprehend the connection between their. financial decisions and performance.
- Scalability and Architectural Separation: The layer structure enhances scalability and maintainability. The stateless Fast API back-end can be horizontally. indepen- dently scaled without depending on Supabase storage layer. But the present single-thread Monte Carlo implementation (10,000 paths × up to 480 monthly steps in a Python loop) has a scalability bottleneck with simultaneous user load. Imple- menting NumPy vectorized matrix operations would decrease computation time by replacing the loop-based implementation. an estimated factor of 50–100× [10].
- Alignment with Modern Portfolio Theory: Consensus with the Modern Portfolio Theory. The portfolio allocation strategy in discussion of the system generates on the basis of the. principles of diversification started by Markowitz [8] in Modern Portfolio Theory. The manner in which they spread investments over four. balanced asset classes: where a balance is struck between expected return and weight. based on the system and profile of the investor, risk. the essence of the MPT that a portfolio risk decreases with diversification. and without proportionate loss of expected return.
- Limitations and Implications for Recommendation Quality: Limitations and Implications for the Quality of the Rec- ommendation. The greatest constraint is the lack of inflation. goal modeling adjustment. At the structural inflation in India. rate of 5-6about 89 lakhs over a decade to continue with similar. purchasing power [5]–a loophole that the existing system lacks. surface. Equally, post-tax corpus projections are not given. means that the output of the system overestimates feasible wealth, especially to investors with higher income tax rates in which case Long-term capital gain (equity) and slab-rate tax of debt returns cause significant decreases in the corpus investable [9].

Limitations

Through the card the following limitations have been de- termined. systematic architectural review, analysis of user behaviour and comparing with accepted financial modelling standards. dards. They are categorised into five domains.

Inflation Modelling: All goal amounts entered by users are treated as nominal figures at the time of input. The system does not adjust target corpus values for the erosive effect of inflation during the period of investment. At the structural inflation of India. interest of 5-6, an amount goal of 50 lakh today will be equal to an actual buying demand of some89 lakh in ten years[5].By running the Monte Carlo simulation due to the nominal target as opposed to the inflation-adjusted target, the system it underestimates the corpus needed in invirtually every. long-duration goal scenario.

Regulatory Compliance: As the platform transitions from an advisory tool toward execution, anea, it will be subject to. Know Your Customer (KYC) obligations under the Prevention of Money Laundering Act (PMLA) 2002 and SEBI's KYC Registration Agency (KRA) framework[9]. The current system has no KYC infrastructure, which will represent a significant compliance and engineering challenge at the monetisation stage.

Tax Modelling: In India, there is a subject to investment returns. to the asset-class-specific taxation: Long-Term Capital Gains. (LTCG) on equity above 1 lakh at 10Gains (STCG

)on equity at 15at the slab of the income of the investor [9]. The system presents pre-tax corpus estimates that are not adjusted post tax, estimating realisedwealth especially when approaching investors of an investment. higher taxbrackets.

Engineering Scalability: The Monte Carlo engine executes 10,000 a pure Python loops in simulation paths, with yield. up to 4.8 million arithmetic operations per API call. Under concurrent user load, this would exceed the capacity of Ren- der' s standard tier, resulting in request time outs. The current implementation does not utilise NumPy vectorization, which could reduce computation time by a factor of 50–100×[10].

User Engagement: The platform currently lacks any mech- anism to re-engage users after their initial planning session. There are no scheduled plan review reminders, no alerts when the portfolio of a user becomes out of its target, and no notifications where market conditions have there is material impact on goal feasibility. This means the platform functions as a one-time calculator rather than an ongoing financial planning companion directions are proposed in direct re- sponse to the limitations identified in the preceding sections, organised by research horizon.

Near-Term Engineering Enhancements (0–3 Months)

1. Log-Normal Monte Carlo and Vectorisation: The imme- diate priority is to replace the Gaussian return assumption with a log-normal model, which better captures the asymmetric, fat-tailed nature of Indian equity returns [4]. The corrected monthly return is drawn as:

$$r_t = \exp\big(\mu - \tfrac{1}{2}\sigma^2 + \sigma\varepsilon_t\big) - 1, \quad \varepsilon_t \sim \mathcal{N}(0,1) \tag{20}$$

Simultaneously, the simulation loop should be refactored to use NumPy matrix operations [10], estimated to reduce computation time from approximately 2.1 seconds to under 40 milliseconds.

2. Behavioural Input Integration into Risk Score: The risk scoring function should be extended to incorporate the four behavioural dimensions currently collected but unused [6], as shown in Table 7.

Table 7. *Proposed Behavioural Adjustments to Risk Score*

Variable	**Conservative**	**Aggressive**
Risk Behavior	−15 pts	+15 pts
Income Stability	−10 pts (Variable Income)	+5 pts (Stable Income)
Experience Level	−5 pts (Beginner)	+8 pts (Advanced)
Flexibility	−8 pts (Inflexible)	+5 pts (Flexible)

3. Inflation-Adjusted Goal Modelling: The goal plan end- point should compute an inflation-adjusted target corpus before running the Monte Carlo simulation [5]:

$$\text{adjusted\_target} = \text{goal\_amount} \times (1 + r_{\text{inflation}})^T \tag{21}$$

4. Fix Hardcoded Age Placeholder and Dimensional Error: Two implementation errors identified during development have been corrected in the current version of the system. First, the AllocationTuner ML model previously received a hardcoded age value of 35 regardless of actual user age, rendering the ML adjustment layer functionally age-agnostic. This has been corrected: the model now receives the actual investor age A as the first element of the feature vector $\mathbf{x} = [A, H, b_e, C]$, as defined in Equation (10). Second, the SIP-to-income ratio in the risk score computation previously divided the monthly SIP by annualised income (monthly SIP (monthly income 12)), producing a ratio approximately 12 times smaller than the economically meaningful monthly savings rate. This has been corrected: now the denominator will be monthly income directly, shown in Table 2.

Medium-Term Product Enhancements (3–9 Months)

1. Market Regime Detection Engine: Market Regime De- tection Engine: A market regime detection module is an example of pre-processing layer that should be incorporated. to the allocation engine, based on multi-signal frameworks. [3]. The four-regime taxonomy proposed (Calm Bull, Volatile Bull, Mild bear) is summarised in Table VIII. ×

2. Tax-Aware Return Projections: Future versions shall. add post-tax corporate projection with the use of LTCG (10on equity gains more than 1 lakh yearly), STCG (15slab-rate taxation of debt, and less than 12 months held. instruments [9].
3. Three-Path Feasibility Resolution: When a human one notices a. Monte Carlo less than 50, the system should auto. mathematically calculate: (a) the minimum SIP to be increased to. 60success; (b) maximum extension of the horizon to be made. The probability of success 60; and (c) the greatest possible goal corpus at. 60success probability [10].

Table 8. *Proposed Market Regime Taxonomy with Allocation Adjustments*

Regime	**VIX**	**Trend**	**Equity Δ**	**Debt Δ**	**Gold Δ**
Calm Bull	<15	Up	+2%	−2%	0%
Volatile Bull	15–25	Up	0%	0%	+2%
Mild Bear	15–25	Down	+3%	−3%	0%
Deep Bear	>25	Down	−3%	+1%	+2%

4. Portfolio Aggregation via Account Aggregator Frame- work: Has RBI Account Aggregator frame. connect with veri by work via Sahamati [7] would provide users with a connection, gamified investment portfolios, changing the platform into a tool planning to a portfolio monitoring system which runs on. ground-truth data

Long-Term Research Directions (9–24 Months)

1. Hidden Markov Model-Based Regime Detection: A more rigorous direction involves fitting a Hidden Markov Model (HMM) [2] to Nifty 50 daily return sequences to identify latent market states. Mamon et al. [4] demonstrated strong HMM performance in capturing COVID-19-induced regime shifts in Indian equities from 2017 to 2023.
2. Reinforcement Learning for Dynamic SIP Allocation: A long-term direction involves training a Reinforcement Learn- ing (RL) agent using Proximal Policy Optimisation (PPO) to dynamically adjust SIP allocation based on observed market regime and goal proximity [3].
3. Personalised Return Expectation Calibration: A calibra- tion module should derive a personalised expected return from the user's risk profile, proposed asset allocation, and current market regime [3], [10], replacing the user-provided figure with a financially grounded estimate.
4. Goal Conflict Resolution via Constrained Optimisation: For users with multiple goals, multi-goal SIP allocation can be formulated as a constrained stochastic optimisation problem [8], [10]:

$$\max_{\{s_k\}} \sum_{k=1}^{K} \lambda_k \cdot P_{\text{success}}^{(k)} \quad \text{s.t.} \quad \sum_{k=1}^{K} s_k = S_{\text{total}}, \quad s_k \geq 0 \qquad (22)$$

5. Behavioural Bias Detection and Intervention: A future direction involves gearing the platform to sense be. signals of behaviour like lost SIP at bear stages and instigating evidence- based interventions based on the behavioural finance literature [6].

Conclusion

Aurelian Capital is an enterprise that is technically based. to spread strict financial planning methodology to the Indian mass-market investor. Its architecture—combining rule linear- based risk profiling, ML-adapted allocation, Monte Carlo simu. more deterministic next-best-action generation, notion, and deterministic next-best-action generation, complex as most similar to consumers tools already in the Indian market.

The weaknesses that are determined in this article–cut across absent mathematical modelling errors, behavioral in- puts, unused. The modelling of inflation and tax, regulatory exposure and engineering scalability gaps–reflect fundamen- tally both risks. quality of outcome to the user and platform feasibility. Several of these Breaches, the worst being the hardcoded age place holder, the dimensional error in the computation of risk ratio and the. Gaussian return assumption, have the potential to generate materially misleading advice to users and should be. treated as engineering corrections (high priority).

The future directions proposed—market regime detects tax- aware projection, inflationad goal modelling, ting, multi-goal optimisation and behavioral intervention- define together a product and research road map that, in case done, would place Aurelian Capital in a requisite situation. when compared to existing platforms. The greatest long term opportunity in the construction of what is nothing of an Indian robo- advisor already provides: a behaviorally, financially rigorous. conscious and Indian retail sensitive planning layer. investor.

Reference

1. AMFI India, "SIP Monthly Data – April 2024," Association of Mutual Funds in India, 2024.
2. J. D. Hamilton, "A new approach to the economic analysis of nonstationary time series and the business cycle," *Econometrica*, vol. 57, no. 2, pp. 357–384, 1989.
3. A. Mulliner, C. Harvey, C. Xia, and E. Fang, "Regimes," Duke University Financial Economics Working Paper, 2025.
4. R. Mamon et al., "Market regime identification and variable annuity pricing: Analysis of COVID-19-induced regime shifts in the

Indian stock market," *Mathematical and Computational Applications*, vol. 30, no. 2, p. 23, 2025.
5. SEBI, "SEBI (Investment Advisers) Regulations, 2013," Securities and Exchange Board of India, Gazette Notification, 2013.
6. D. Kahneman, *Thinking, Fast and Slow*. New York, NY, USA: Farrar, Straus and Giroux, 2011.
7. Sahamati Foundation, "Account Aggregator Ecosystem Overview," 2024.
8. H. Markowitz, "Portfolio selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.
9. SEBI, "Prevention of Money Laundering Act (Maintenance of Records) Rules and KYC Registration Agency Framework," 2021.
10. S. Benninga, *Financial Modeling*, 4th ed. Cambridge, MA, USA: MIT Press, 2014.
11. W. F. Sharpe, "Capital asset prices: A theory of market equilibrium under conditions of risk," *The Journal of Finance*, vol. 19, no. 3, pp. 425–442, 1964.
12. E. F. Fama, "Efficient capital markets: A review of theory and empirical work," *The Journal of Finance*, vol. 25, no. 2, pp. 383–417, 1970.
13. F. Black and R. Litterman, "Global portfolio optimization," *Financial Analysts Journal*, vol. 48, no. 5, pp. 28–43, 1992.
14. F. D'Acunto, N. Prabhala, and A. G. Rossi, "The promises and pitfalls of robo-advising," *The Review of Financial Studies*, vol. 32, no. 5, pp. 1983–2020, 2019.
15. D. Jung, H. Dorner, C. Glaser, and S. Morana, "Robo-advisory: Digitalization and automation of financial advisory," *Business & Information Systems Engineering*, vol. 60, no. 1, pp. 81–86, 2018.
16. Lim and S. Zohren, "Time-series forecasting with deep learning: A survey," *Philosophical Transactions of the Royal Society A*, vol. 379, p. 20200209, 2021.
17. S. K. Nanda and P. K. Panda, "Explainability and transparency in AI-based financial advisory systems: A systematic review," *Expert Systems with Applications*, vol. 213, p. 119224, 2023.