iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: IV Month of publication: April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79380

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 IV Apr 2026- Available at www.ijraset.com

# FinSim: A Deterministic Financial Shock and Decision Simulation System for Personal Finance Decision Support

Rahil Majithia[1], Krishi Oza[2], Janish Dave[3], Dakshesh Khadtare[4], Prof. Kashif Sheikh[5]
*Department of Computer Engineering, Thakur Shyamnarayan Engineering College, Mumbai, India*

***Abstract: FinSim is a personal finance decision support system designed to help individuals examine how financial shocks and major financial decisions affect liquidity, debt burden, and overall resilience. FinSim is designed as a deterministic, rule-based platform with no artificial intelligence, no machine learning, and no probabilistic recommendation logic. Instead, it combines structured user financial profiles, financial formulas, threshold-based classification, and visual analytics to support what-if exploration. The proposed platform integrates three main functional modules: shock simulation, decision impact simulation, and resilience analysis. These modules compute metrics such as savings rate, debt-to-income ratio, emergency fund coverage, net worth, loan affordability, and a composite financial health score. The literature reviewed for this work shows that many existing personal finance tools focus on tracking past spending, while prior decision support research highlights the value of model-driven and knowledge-driven systems for complex personal finance decisions [1]-[4]. FinSim contributes a transparent architecture that connects these ideas to an Indian personal finance context through INR-based modeling, EMI analysis, and explainable rule-driven recommendations. Using the system's implementation and validation results, this paper shows that the platform can identify high-risk cases such as prolonged job loss, weak liquidity buffers, and overextended loan decisions, while also providing interpretable outputs through charts, health scoring, and typed recommendations..***
***Keywords: decision support system, personal finance, financial resilience, shock simulation, rule-based recommendations, EMI analysis.***

## I. INTRODUCTION

Personal financial planning has become more difficult as households increasingly need to manage income allocation, debt obligations, savings, investments, and risk exposure at the same time. Prior work on financial decision support systems argues that computerized support is valuable when users face unstructured or semi-structured decisions involving multiple variables, high stakes, and limited analytical capacity [1]-[4]. Financial vulnerability research also shows that household fragility is shaped not only by income and debt, but also by behavioral factors, liquidity, and the ability to cope with short-term shocks [5], [7], [8].

The literature reviewed in this study points to three recurring themes. First, model-driven and personalized decision support systems can help users evaluate financial alternatives more systematically [1], [2], [4]. Second, household vulnerability often emerges when debt burdens rise, savings buffers are thin, or families cannot mobilize funds quickly during emergencies [5], [7], [8], [9]. Third, financial literacy and perceived understanding strongly influence borrowing choices and can lead to sub-optimal credit behavior when consumers underestimate the cost or risk of decisions [6], [10].

At the same time, the FinSim literature review observes that most mainstream consumer tools are retrospective: expense trackers answer where money went, investment platforms emphasize execution, and isolated calculators answer only one narrow question at a time. They do not normally allow the user to simulate how a six-month job loss, a medical emergency, an inflation spike, or a new loan will cascade through the rest of the household balance sheet. This gap motivates FinSim as a simulation-driven system that treats personal finance as a decision support problem rather than only a record-keeping task.

FinSim is a Financial Shock and Decision Simulator that stress-tests a user's financial position under adverse scenarios and major purchase decisions. The platform adopts a deterministic design philosophy: every score and recommendation must be reproducible from explicit financial formulas and threshold rules. This paper presents the proposed system, its formulas, architecture, implementation methodology, evaluation results, and current limitations.

 

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 IV Apr 2026- Available at www.ijraset.com

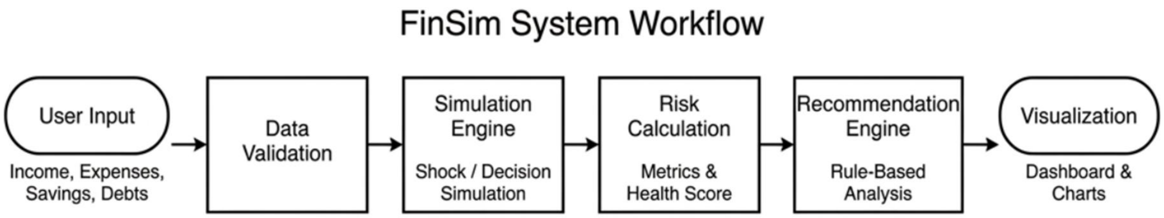


Figure 1. FinSim system workflow: input data are validated, processed through simulation and risk calculation, passed through the recommendation engine, and rendered as dashboard visualizations.

## II. RELATED WORK AND RESEARCH GAP

The project literature positions FinSim at the intersection of decision support systems, personal financial planning, household vulnerability, and financial literacy. Palma-dos-Reis and Zahedi describe personalized financial DSS as systems that should integrate multiple models while also taking user preferences and characteristics into account [1]. Vahidov and He extend this thinking by arguing for situated personal finance DSS that actively link decision logic to the problem environment, rather than offering passive support only [2]. Songsangyos and Iamamporn focus on what-if analysis and goal-seeking in personal financial analysis [3], while Gao et al. show how agent-assisted family financial planning can improve coordination across complex financial decisions [4].

The vulnerability literature contributes the economic rationale for simulation. Noerhidajati et al. emphasize that household financial vulnerability is affected by income factors, behavior, and socio-economic conditions, not only by debt volumes [5]. Anderloni, Bacchiocchi, and Vandone broaden vulnerability beyond loan default and define it through difficulties in meeting monthly expenses, balancing the household budget, and coping with unexpected events [7]. Lusardi, Schneider, and Tufano show that many households remain financially fragile because they cannot easily come up with emergency funds within a short time frame [8]. Sikarwar, Goyal, and Mathur further highlight the macroeconomic importance of household debt and its relationship with inclusive growth in India [9].

Behavioral research also supports the need for explainable guidance. Balasubramnian and Sargent show that inflated perceptions of financial literacy are associated with weaker financial decisions [6]. Disney and Gathergood similarly find that lower financial literacy is linked to more expensive consumer credit portfolios and poorer understanding of credit terms [10]. Together, these studies suggest that a practical personal finance tool should not merely compute outputs; it should present transparent, interpretable reasoning that helps the user understand the implications of a decision.

| Dimension | What existing literature/tools provide | Gap addressed by FinSim |
|---|---|---|
| DSS foundations | Model-driven, personalized, or situated decision support for finance [1]-[4] | A consumer-oriented system that combines simulation, health scoring, and explainable recommendations in one workflow |
| Household vulnerability | Measures of fragility, debt burden, and emergency resilience [5], [7], [8], [9] | Direct scenario simulation of job loss, medical expense, market decline, and other shocks using user-specific financial data |
| Financial literacy | Evidence that weak or inflated literacy can worsen financial decisions [6], [10] | Transparent formulas and typed recommendations instead of black-box outputs |
| Consumer tools | Budget trackers, calculators, or investment apps documented in the literature review | Integrated before/after analysis across savings rate, DTI, disposable income, emergency buffer, and net worth |

Table 1. Research gap synthesized from the uploaded literature review and cited personal finance DSS papers.

From these sources, the central research gap becomes clear: there is a need for a transparent personal finance system that uses structured user data, standard financial formulas, and scenario analysis to support decisions before they are made. FinSim responds to that gap through deterministic modeling, modular architecture, and full visibility into the rules that drive its outputs.

## III. PROPOSED SYSTEM OVERVIEW AND CONTRIBUTIONS

FinSim is designed as a model-driven decision support system whose value is derived from computation over a structured financial profile. The system is organized around four core elements: a shock simulator, a financial decision impact simulator, a resilience analyzer, and a cross-cutting recommendation engine. Together, these modules allow a user to ask three practical questions: What happens if my finances face a shock? What happens if I take a major loan or purchase decision? How resilient is my current profile before any additional decision is taken?

| Module | Primary purpose | Representative outputs |
|---|---|---|
| Shock Simulator | Estimate the effect of job loss, salary reduction, medical emergency, market crash, or inflation spike | Remaining savings, emergency months after shock, new net worth, savings timeline, risk level |
| Decision Simulator | Evaluate major purchase or loan affordability using amortization logic | EMI, total repayment, disposable income before/after, DTI before/after, decision label |
| Resilience Analyzer | Assess baseline financial strength using weighted scoring | Emergency fund score, debt score, savings score, composite health score, risk class |
| Recommendation Engine | Convert computed metrics into explainable advice using thresholds | Typed recommendations in savings, debt, expense, and investment categories |

Table 2. Core system modules and their outputs, summarized from the project overview and specification documents.

The main contribution claimed by the project documentation is not a machine learning model but a transparent analytical framework. The system avoids AI-generated advice and instead uses threshold-based rules, reproducible formulas, and direct mapping between metrics and recommendations. This is important in finance because explainability, repeatability, and auditability are often more valuable to end users than opaque prediction alone. The project also explicitly localizes the system to an Indian context through INR-based amounts, EMI-centric loan evaluation, and financial examples expressed in lakhs.

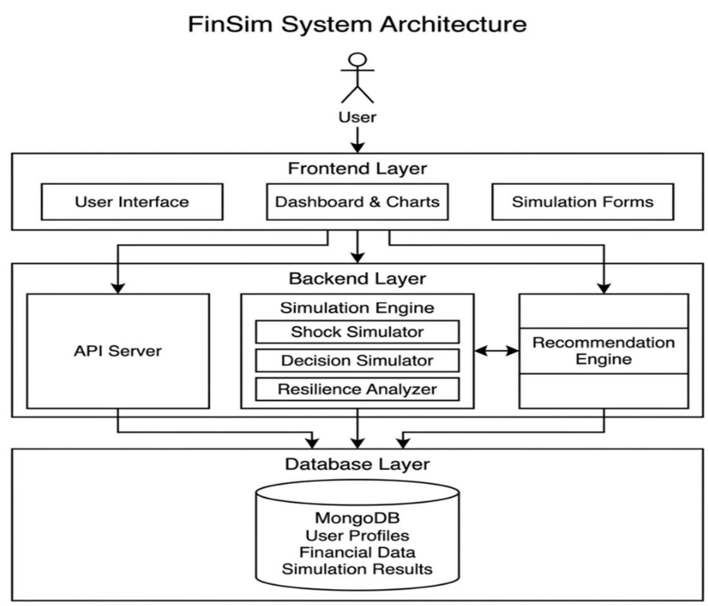


Figure 2. FinSim system architecture organized into frontend, backend, and database layers with a dedicated simulation engine and recommendation engine.

 

## IV. ARCHITECTURE, DATA MODEL, AND PROCESSING PIPELINE

FinSim follows a three-tier MERN architecture with a React-Vite client, a Node.js and Express server, and MongoDB persistence. On the client side, the interface includes authentication pages, profile setup, simulation forms, dashboard metrics, and chart rendering. On the server side, routing, controllers, services, isolated simulation modules, middleware, and shared financial utilities are separated by responsibility. This layered design helps keep formulas independent of HTTP concerns and database access.

The data model includes five collections: users, financial_profiles, simulations, recommendations, and audit_logs. The financial profile is the single source of truth for user inputs. It stores income sources, expenses, assets, liabilities, an existing EMI amount, and an inflation rate, while derived totals such as monthly income, monthly expenses, total assets, total liabilities, savings, and investments are computed during profile updates. The simulations collection records executed analyses, recommendations store the typed rule outputs associated with those analyses, and audit logs capture security-relevant actions such as registration, login, and analysis requests.

The implemented system uses protected routes for profile retrieval, dashboard metrics, decision simulation, and resilience analysis, with authenticated requests carrying a JWT bearer token. In the current prototype, shock simulation is computed on the frontend from the stored financial profile rather than being executed through a persisted backend API workflow. The platform also includes bcrypt password hashing, Joi input validation, Helmet headers, CORS controls, rate limiting, database access restrictions, and structured audit logging. Taken together, these design choices define a system that is not only computationally deterministic but also organized with clear operational boundaries.

| Collection | Relationship rule | Purpose in the platform |
|---|---|---|
| users | One document per registered user | Authentication, account status, and identity fields |
| financial_profiles | One profile per user | Baseline financial data consumed by all modules |
| simulations | Many per user | History of shock, decision, and resilience runs |
| recommendations | Many per simulation | Stored rule outputs linked to a specific analysis |
| audit_logs | Many per user | Security and activity tracking |

Table 3. Core MongoDB collections defined in the database schema.

The end-to-end data flow moves user input from the browser to the Redux store and, for API-backed features, through Axios requests into Express controllers and the simulation layer before responses are returned and visual components are re-rendered. In the current prototype, shock simulation follows a client-side computation path using the stored profile, while decision simulation and resilience analysis use the backend flow. This structure is central to the system's decision support role because it ensures that the same structured data feeds all metrics, visualizations, and recommendations.

## V. FINANCIAL MODELING, HEALTH SCORING, AND RECOMMENDATION LOGIC

The financial logic of FinSim is based on established primitives such as EMI amortization, debt-to-income ratio, savings rate, emergency fund months, compound growth, and net worth. For loan decisions, the EMI is computed using the standard annuity-based formula, where P is the principal, r is the monthly interest rate, and n is the tenure in months. For resilience analysis, the system converts three core metrics - emergency fund coverage, debt burden, and savings rate - into normalized scores and then aggregates them using a weighted composite formula.

| Metric / formula | Definition used by FinSim | Interpretive purpose |
|---|---|---|
| EMI | P × r × (1 + r)^n / ((1 + r)^n - 1) | Measures monthly repayment burden for a purchase or loan |
| Debt-to-income ratio | Existing or total EMI / monthly income | Indicates debt stress and borrowing capacity |
| Savings rate | (Income - expenses - EMI) / income | Captures capacity to build reserves |
| Emergency fund months | Savings / monthly expenses (or expenses plus EMI in scoring) | Represents liquidity buffer against shocks |
| Net worth | Total assets - total liabilities | Summarizes the balance-sheet position |
| Health score | 0.4 × emergency + 0.3 × debt + 0.3 × savings | Combines liquidity, debt burden, and savings strength into a 0-100 scale |

Table 4. Summary of the main financial formulas and metrics specified in the FinSim project documents.

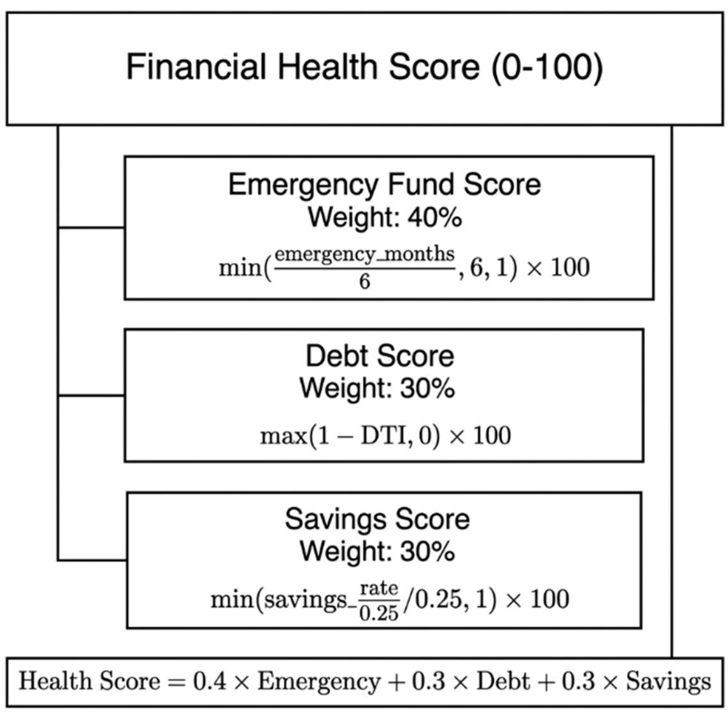


Figure 4. Financial health score structure with weighted emergency fund, debt, and savings components.

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 IV Apr 2026- Available at www.ijraset.com

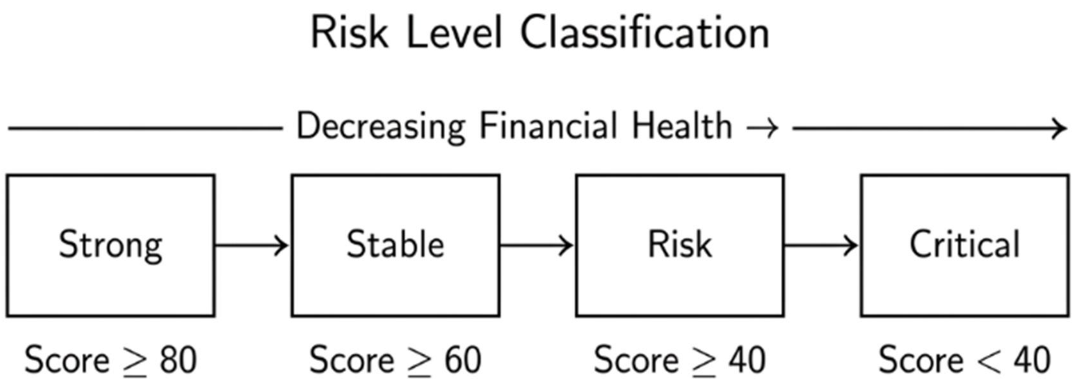


Figure 5. Risk level classification used to map the financial health score into Strong, Stable, Risk, and Critical categories.

The recommendation engine extends the quantitative layer into explainable decision support. Recommendations are generated by evaluating a metrics object against threshold conditions. For shock simulations, the rules examine remaining savings, emergency fund coverage after the event, income loss, unexpected expenses, market decline, and net worth change. For decision simulations, the rules check post-decision DTI, savings-rate decline, disposable income, and the proportion of income committed to the new EMI. For resilience analysis, the rules summarize current emergency buffer, savings rate, and debt burden.

This architecture makes each recommendation traceable to a specific quantitative trigger. For example, an emergency fund below three months is classified as highly vulnerable, a DTI ratio above 40% is flagged as high debt burden, and disposable income falling below the documented threshold is treated as tight post-loan cash flow. The result is a knowledge-driven layer that remains deterministic while still producing concise user guidance.

## VI. METHODOLOGY AND IMPLEMENTATION APPROACH

FinSim was developed using an iterative incremental process supported by detailed upfront specifications. Requirements were gathered from domain analysis, competitive analysis, user scenario exploration, and financial planning literature. The development process produced dedicated specifications for product requirements, software requirements, system design, API behavior, database schema, recommendation logic, charts, UI design, testing, security, and research positioning. This specification-first approach reduced ambiguity while still allowing iterative refinement during implementation.

From a technical standpoint, the backend was built bottom-up through configuration, models, shared mathematical utilities, simulation modules, services, middleware, controllers, and route assembly. The frontend was assembled feature-by-feature through authentication, financial profile setup, overview metrics, charts, shock simulation, decision simulation, resilience analysis, layout, and design polish. The methodology also emphasizes separation of concerns: presentation logic lives in React components, orchestration lives in controllers and Redux thunks, business logic stays in services and simulation modules, and persistence is isolated in Mongoose models.

Validation is organized as a multi-layered process consisting of input validation, formula verification, edge-case testing, integration testing, and security validation. The validation design includes worked examples for EMI calculation, financial health score calculation, shock simulation, decision simulation, and low-income resilience assessment. These test cases are important because the value of the system depends on numerical correctness and stable interpretation rather than on subjective output generation.

| Validation layer | What is checked | Representative examples from the testing document |
| --- | --- | --- |
| Input validation | Schema constraints before computation | Joi rules for shock type, tenure limits, rate limits, and non-negative financial fields |
| Formula verification | Correctness of core finance equations | EMI, savings rate, DTI, emergency months, and health score examples |

 

| Validation layer | What is checked | Representative examples from the testing document |
|---|---|---|
| Edge-case testing | Boundary and zero-value behavior | 0% interest, 0 tenure, 100% income loss, zero savings, maximum shock duration |
| Integration testing | End-to-end application flow | Register, create profile, fetch dashboard, run simulations, retrieve resilience analysis |
| Security validation | Protection of user data and routes | JWT verification, inactive-user checks, password hashing, and error sanitization |

Table 5. Multi-layer validation strategy summarized from the testing and security documents.

## VII. RESULTS AND DISCUSSION

The implementation results show that the major modules were completed: user authentication, financial profile management, financial overview, shock simulation, decision simulation, resilience analysis, and the recommendation engine. The prototype includes approximately 80 total source files, about 1,200 backend lines of code, about 3,500 frontend lines of code, 10 API endpoints, 5 MongoDB collections, 11 chart components, and more than 38 recommendation rules. These numbers indicate that FinSim is not simply a calculator but a full-stack prototype with persistent user state, historical simulation records for API-backed analyses, and multiple visual analysis layers.

Scenario testing provides the clearest evidence of system behavior. In the documented job-loss matrix, a profile with a monthly income of INR 95,000, monthly expenses of INR 37,000, existing EMI of INR 20,000, and savings of INR 250,000 reaches a critical condition when complete income loss continues for six months. The testing notes show that savings are exhausted by around month five, while the post-shock emergency fund coverage falls to zero. Less severe reductions, such as a 30% income cut over six months, remain comparatively stable because the emergency fund stays above the critical threshold. This confirms that the simulator captures the non-linear practical difference between manageable stress and full depletion.

Decision analysis results are equally illustrative. For a purchase of INR 1,000,000 with a down payment of INR 200,000, interest of 9%, and a 60-month tenure, the system computes an EMI of roughly INR 16,607, post-decision DTI of 38.5%, and a moderate affordability label. In contrast, a much larger loan without down payment produces a critical DTI above 90%. The system therefore does more than calculate EMI: it embeds repayment into the wider financial profile and shows how the same loan changes disposable income, savings rate, and resilience.

The resilience analyzer results also support the weighting strategy used in the health score. A strong salaried profile can reach a score of 94, while low-savings or high-debt profiles fall into the Risk or Critical ranges even if one dimension remains healthy. Because emergency fund coverage carries 40% of the total weight, the score prioritizes liquidity as the first defense against financial stress. This weighting is consistent with the fragility literature, which emphasizes that the inability to cover unexpected expenses is one of the clearest signs of household vulnerability [5], [7], [8].

| Scenario | Key documented inputs | Representative outputs |
|---|---|---|
| Severe job loss shock | 6 months; 100% income loss; savings INR 250,000; monthly expenses INR 37,000; EMI INR 20,000 | Savings depleted by month 5; emergency fund after shock = 0; risk level = Critical |
| Moderate salary reduction shock | 6 months; 30% income reduction | Emergency fund remains above critical threshold; risk level reported as Stable |

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 IV Apr 2026- Available at www.ijraset.com

| Scenario | Key documented inputs | Representative outputs |
|---|---|---|
| Loan decision example | Purchase INR 1,000,000; down payment INR 200,000; 9% interest; 60 months | EMI ~ INR 16,607; DTI after = 38.5%; decision label = Moderate |
| Low-income resilience profile | Income INR 25,000; expenses INR 22,000; savings INR 10,000; EMI = 0 | Health score = 47; risk level = Risk due to weak emergency buffer |
| High-debt profile | Income INR 80,000; expenses INR 30,000; EMI INR 45,000 | Health score = 38; classified as Critical in results summary |

Table 6. Representative results extracted from the documented evaluation and testing scenarios.

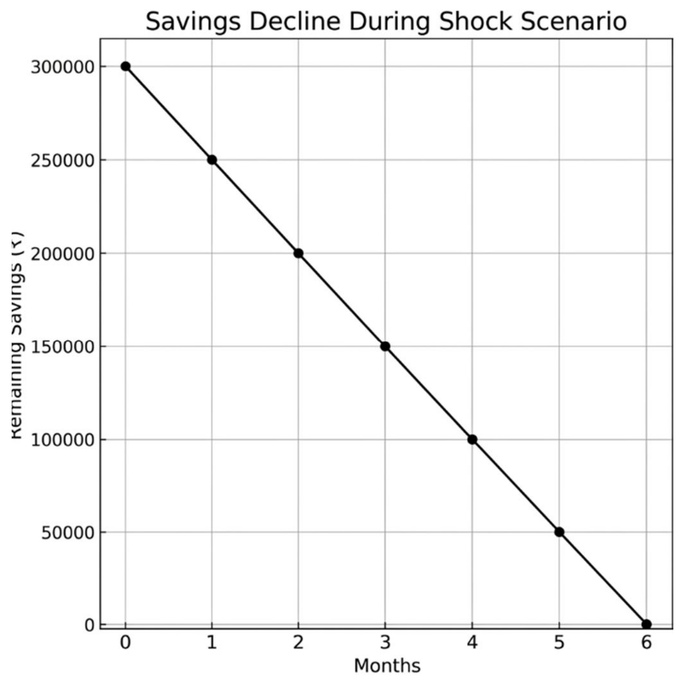


Figure 7. Savings decline during a severe shock scenario, illustrating the month-by-month depletion pattern documented in the test cases.

From a system design perspective, three strengths stand out. First, the outputs are reproducible because the same inputs always produce the same metrics and recommendations. Second, the platform combines multiple financial concepts - liquidity, debt burden, net worth, and savings behavior - rather than isolating each one in a separate calculator. Third, the visual layer makes the results more interpretable: line charts highlight depletion, bar charts emphasize before/after change, and the health score provides a compact summary metric that is easy to compare across profiles. These traits make FinSim particularly suitable as an educational decision support prototype.

## VIII. LIMITATIONS AND FUTURE SCOPE

FinSim remains a working prototype rather than a complete real-time consumer finance platform. The most important limitation is that the system depends on self-reported data. Users must manually enter and update income, expense, asset, and liability information, and the system has no automated transaction or bank account integration. The model is therefore only as accurate as the current profile snapshot. Other analytical limitations include no tax modeling, no insurance treatment, no inflation compounding over long shock horizons, no adaptive spending reductions during stress, and no support for floating-rate or hybrid EMI structures.

| Limitation area | Documented issue | Implication for the current prototype |
|---|---|---|
| Data collection | Self-reported and static profile data | Accuracy depends on user input; there is no real-time aggregation |
| Analytical scope | No taxes, insurance, or inflation compounding over long durations | True disposable income and resilience may differ from simulated outputs |
| Recommendation logic | Fixed thresholds and no contextual adaptation | Users with different life stages receive the same rule output for the same metrics |
| Technical platform | No offline mode, no export workflow, relaxed demo rate limit, and localStorage JWT strategy | The current system is suited to prototype validation rather than hardened production use |
| Visualization | Two radar axes are placeholders in the documented implementation | Not every displayed dimension is derived from time-series evidence |

Table 7. Limitations explicitly identified in the FinSim limitations and future scope document.

Several directions for future work follow naturally from the present design. Short-term enhancements include PDF reporting, simulation history pages, profile versioning, dynamic radar values, and automated tests. Medium-term directions include goal-based planning, multi-scenario comparison, insurance and tax integration, inflation compounding, and family-level profiles. Longer-term ambitions include account-aggregator integration, machine-learning-enhanced personalization, expense prediction, retirement planning, and an open API. These ideas extend the platform while preserving the central design goal of explainable financial decision support.

## IX. CONCLUSION

This paper presents FinSim as a deterministic personal finance decision support system that combines simulation, rule-based interpretation, and visualization. The reviewed literature establishes the need for such a system: existing studies show the importance of model-driven financial support, the prevalence of household fragility, and the role of literacy and debt structure in financial decision quality [1]-[10]. FinSim operationalizes those concerns through a structured financial profile, simulation modules, weighted health scoring, threshold-based rules, and persistent analysis records for its API-backed modules.

The main value of FinSim lies in transparency. Rather than predicting behavior through opaque models, it allows users to inspect how formulas and thresholds produce outputs such as EMI, DTI, emergency months, decision labels, and health categories. The implementation and testing results further show that the system can distinguish between stable and high-risk cases in ways that are financially meaningful and visually interpretable. As a prototype, it still has clear limitations, but the implemented system provides a credible foundation for future work in personal finance simulation, explainable financial guidance, and consumer-facing decision support.

## REFERENCES

[1] A. Palma-dos-Reis and F. Zahedi, "Designing personalized intelligent financial decision support systems," Decision Support Systems, vol. 26, 1999, pp. 31-47.
[2] R. Vahidov and X. He, "Situated DSS for personal finance management: Design and evaluation," Information & Management, vol. 46, 2009, pp. 453-462.
[3] P. Songsangyos and S. Iamamporn, "Decision support system for personal financial analysis," The 2014 WEI International Academic Conference Proceedings, 2014.
[4] S. Gao, H. Wang, D. Xu, and Y. Wang, "An intelligent agent-assisted decision support system for family financial planning," Decision Support Systems, vol. 44, 2007, pp. 60-78.
[5] S. Noerhidajati, A. B. Purwoko, H. Werdaningtyas, A. I. Kamil, and T. Dartanto, "Household financial vulnerability in Indonesia: Measurement and determinants," Economic Modelling, 2020.

[6] B. Balasubramnian and C. S. Sargent, "Impact of inflated perceptions of financial literacy on financial decision making," Journal of Economic Psychology, vol. 80, 2020.
[7] L. Anderloni, E. Bacchiocchi, and D. Vandone, "Household financial vulnerability: An empirical analysis," Research in Economics, vol. 66, 2012, pp. 284-296.
[8] A. Lusardi, D. Schneider, and P. Tufano, "Financially Fragile Households: Evidence and Implications," Netspar Discussion Paper 03/2011-013, 2011.
[9] T. Sikarwar, A. Goyal, and H. Mathur, "Household debt, financial inclusion, and economic growth of India: Is it alarming for India?" Asian Economic and Financial Review, vol. 10, no. 2, 2020, pp. 229-248.
[10] R. Disney and J. Gathergood, "Financial literacy and consumer credit portfolios," Journal of Banking & Finance, vol. 37, 2013, pp. 2246-2254.