NOVENTIA JOURNAL
Volume I Issue 1 December 2025

Virtual Account Manager

D Yasaswini

Department of Data Science, St. Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

yasaswinidevana@gmail.com

Abstract: This paper introduces the Virtual Account Manager (VAM) — a personal finance system designed to encourage controlled spending through the use of category-based virtual accounts. The framework integrates a Spring Boot (Java 17) backend with a MySQL database and a responsive HTML/CSS/JavaScript interface. Each account category—such as Food, Bills, or Education—is assigned a specific budget limit and expiration period, supported by an automated emergency reserve for fallback transactions. The system validates every transaction against both active and reserve balances, thereby preventing overspending. The integrated analytics dashboard further assists users by visualizing expense trends and category-wise usage. Functional validation confirms accurate rule enforcement and reliable expense tracking, making VAM a practical approach to disciplined digital money management.

Keywords: Virtual Account Manager, Budget Control, Category-Based Spending, Spring Boot, MySQL, Personal Finance Management.

I. INTRODUCTION

Uncontrolled digital spending often leads to poor budgeting and financial stress. Conventional wallet systems allow unrestricted expenditure from a single balance, offering little accountability. The Virtual Account Manager introduces structured financial control by dividing a user's funds into multiple virtual accounts (VAs) such as Food, Bills, and Education, each assigned a fixed limit and validity period. An emergency account acts as a mandatory fallback whenever the primary VA lacks funds, ensuring continuous yet disciplined transaction handling.

Existing personal-finance applications mainly provide analytics without enforcing budgetary constraints. VAM bridges this gap by coupling budgeting with real-time spending control. The system's objectives are:

- to implement automated budget segregation through virtual accounts.
- to prevent overspending using mandatory emergency fallbacks.
- to generate transparent analytics for informed financial decisions.

II. SYSTEM DESIGN AND ARCHITECTURE

The proposed system adopts a three-tier architecture comprising Presentation, Application, and Data layers.

A. Presentation Layer

Developed with HTML, CSS, and JavaScript, this layer provides a secure login interface integrated with JWT-based authentication, an interactive dashboard displaying account balances and expiry statuses, and visual charts representing monthly spending trends. Filters by category, month, and year facilitate historical review.

B. Application Layer

Implemented using Spring Boot, this layer defines RESTful controllers and services that enforce account rules, validate transactions, and ensure secure API communication through JWT. Business logic governs account creation, expiration, and inter-account transfers.

C. Data Layer

A MySQL database maintains user credentials, virtual-account details (category, balance, expiry, status), transactions (ID, amount, mode, timestamp), and the unique emergency account for each user. All transactions undergo verification to maintain data integrity and consistency.

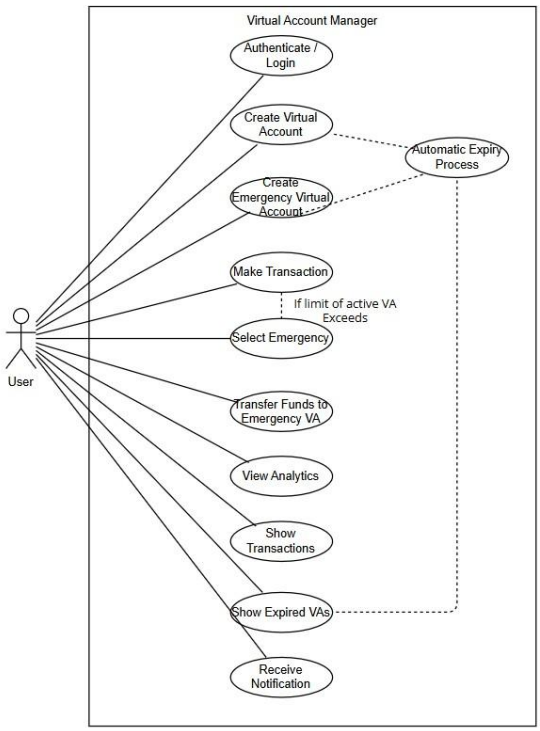

Fig 1: Usecase diagram of Virtual Account Machine

III. FUNCTIONAL FLOW

The system workflow revolves around five key operations:

Account Creation: Users may create one active VA per category and one mandatory emergency account. Upon expiry, accounts automatically shift to the "Expired" section and become read-only.

Transaction Processing: Each payment request validates availability in both normal and emergency VAs. User confirmation is mandatory; no automatic deductions occur.

Transfer Control: Funds can only be transferred from active VAs to the emergency account, ensuring transparent tracking.

Expiry Handling: Expired accounts are archived and accessible through month-, year-, and category-based filters.

Analytics Module: The system computes total spending per category, monthly trends, and emergency-fund usage frequency, visualized through simple charts for decision support.

IV. IMPLEMENTATION

The backend uses Spring Boot 3 with Spring Data JPA for modularity, scalability, and security. Primary endpoints include:

/login – User authentication

/api/virtualaccounts – Account creation and retrieval

/api/transactions – Transaction recording and history

/api/analytics/monthly – Category-wise monthly spending

Frontend communication employs Axios for API calls, modular JavaScript for state updates, and responsive CSS styling. Error handling provides descriptive alerts such as "Insufficient Balance" or "Emergency Account Required."

V. RESULTS AND DISCUSSION

Functional testing validated rule enforcement and data consistency. Transactions exceeding available balances were blocked, while reverse transfers (Emergency → Normal) were correctly restricted. The analytics dashboard accurately reflected monthly spending trends. Expired accounts were archived with fully functional filtering.

Although the system effectively enforces discipline, future work should include quantitative performance evaluation—such as response latency, concurrent user load, and resource utilization—to strengthen empirical validation.

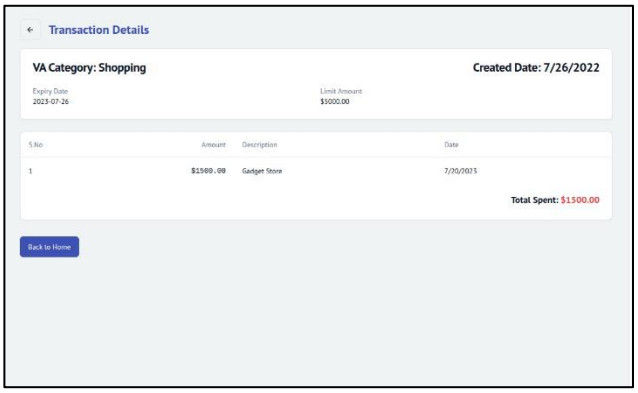

Fig 2: Expired Virtual Account Transaction

NOVENTIA JOURNAL
Volume I Issue 1 December 2025

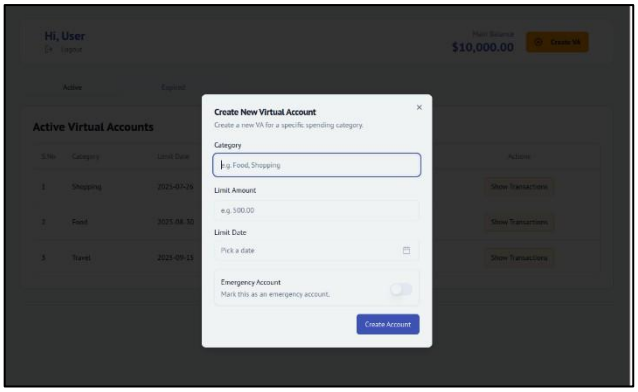

Fig 3: Creating Virtual Account

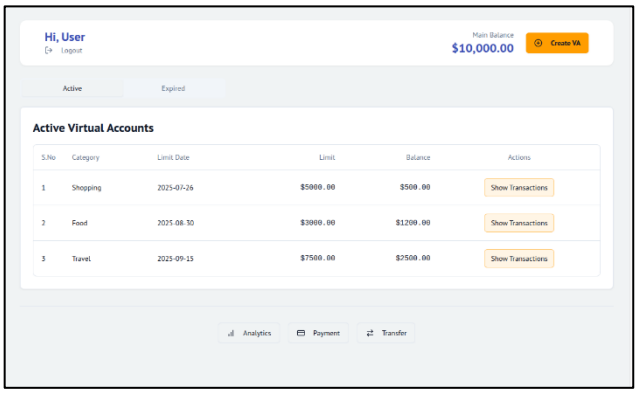

Fig 4: Home Page

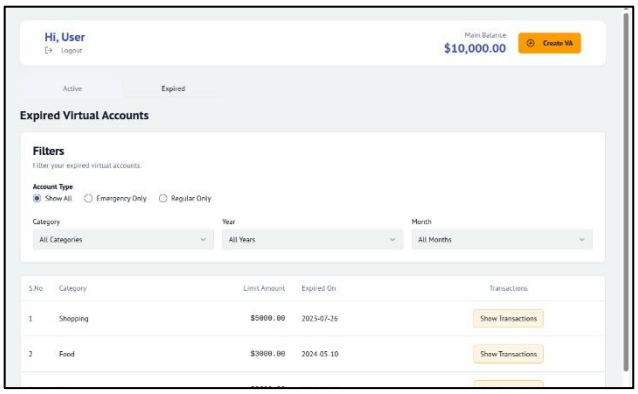

Fig 5: Expired Virtual Accounts page

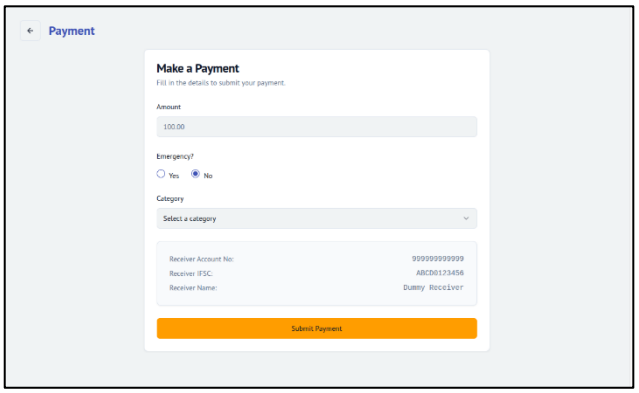

Fig 6: Payment Page

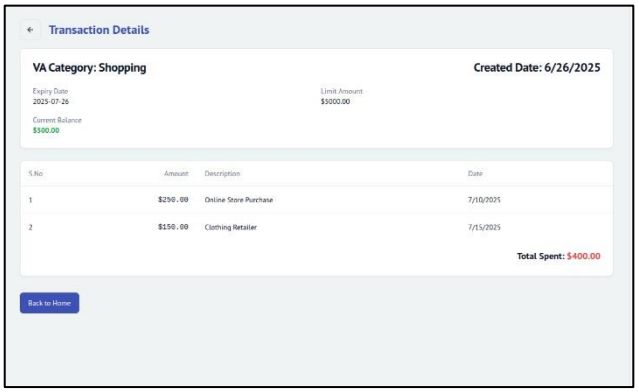

Fig 7: Transaction Page

VI. CONCLUSION

The Virtual Account Manager offers an integrated method for responsible financial management by combining budgeting, spending control, and analytics into a unified framework. Through category-based segregation and emergency-fund validation, the platform ensures that transactions stay within predefined limits while maintaining accessibility when required. The prototype successfully enforces financial discipline through user-defined constraints and real-time validation. Future work will extend this foundation with live banking integration, mobile compatibility, and predictive expenditure analysis, transforming VAM into a comprehensive personal finance companion.

ACKNOWLEDGEMENT

The author expresses sincere gratitude to the Department of Data Science, St. Ann's College of Engineering & Technology, for providing laboratory facilities and academic guidance. Appreciation is also extended to peers for their collaboration and to family for continuous encouragement throughout this research.

REFERENCES

[1] H. G. M., P. Kumar, and R. M., "AI-Driven Financial Insights for Personal Budget Planning: A Smart Approach to Future Expense Prediction," in Proc. 2025 3rd Int. Conf. on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, Feb. 5–7, 2025, pp. 1-6, doi: 10.1109/IDCIOT64235.2025.10914721.

[2] J. K. Kim, W. C. Kwon, I. Y. Choi, H. Heo, and H. S. Moon, "A Group Travel Recommender System Based on Group Approximate Constraint Satisfaction," IEEE Access, vol. 12, pp. 96113 – 96125, Jul. 2024, doi: 10.1109/ACCESS.2024.3427122.

[3] N. Chhillar, A. Gupta, and S. Raj, "Exploring the Role of Digital Financial Literacy and Personal Financial Management," Journal of Business Research, vol. 169, pp. 106213, 2025, doi: 10.1016/j.jbusres.2025.106213.
Relevance: Examines how digital financial literacy influences individual budgeting, saving, and spending control — reinforcing your system's goal of promoting disciplined financial behaviour.

[4] B. Savitha, M. Rao, and P. Kumar, "What Motivates Individuals to Use FinTech Budgeting Apps," FinTech Research Letters, vol. 8, no. 4, pp. 512–520, 2022, doi: 10.1080/23322039.2022.2127482.
Relevance: Investigates user adoption drivers of budgeting apps in the FinTech ecosystem, useful for strengthening your Introduction and Motivation sections.

[5] Y. Y. Lee, "Insights from Collaborative Financial Technologies," ACM Transactions on Financial Technologies, vol. 4, no. 3, pp. 1–18, 2025, doi: 10.1145/3706598.3713119.
Relevance: Discusses digital collaboration frameworks in modern finance applications, aligning with the technological backbone of your VAM architecture (Spring Boot + MySQL + REST APIs).