

OPTI-BUDGET ENGINE - AI QUERY-DRIVEN EXPENSE MANAGEMENT AND BUDGET MONITORING SYSTEM

Dr. Vivek Jaladi¹, Abdul Kareem Khan², Md Khustar Ahmed³, Mohammed Afzanuddin⁴, Syed Shah Mustafa Hussaini⁵

¹Professor & Head, Department of Computer Science & Engineering, Lingaraj Appa Engineering College, Bidar, Karnataka, India

^{2,3,4,5}BE Final Year, Department of Computer Science & Engineering, Lingaraj Appa Engineering College, Bidar, Karnataka, India

Abstract - Personal expense management has become increasingly complex due to the growth of digital transactions and diverse spending patterns. Conventional expense tracking systems depend on manual data entry and static interfaces, which often leads to incomplete records, low user engagement, and weak financial insight. This paper presents Opti-Budget Engine, an AI query-driven expense management and budget monitoring system designed to simplify and automate personal finance tracking. The system enables users to manage expenses through natural language interaction, allowing them to add, query, update, and delete financial records conversationally instead of filling traditional forms. An integrated AI agent interprets user intent, extracts key entities such as amount, category and date, and executes verified operations directly on the backend SQLite database, ensuring real-time synchronization between data storage and visual analytics. The platform also offers dynamic dashboards, category-wise and trend-based insights, and professional PDF report generation to support informed financial decision-making. Overall, Opti-Budget Engine transforms conventional expense tracking into an intelligent financial assistance process by reducing user effort, improving record accuracy, and enhancing usability, providing a modern and user-centric solution for personal finance management.

Key Words: Personal finance, Expense management, Budget monitoring, Conversational AI, Large language model, Flask, SQLite, Dashboard.

1. INTRODUCTION

In the contemporary digital era, personal financial management has emerged as a critical yet challenging task due to the increasing volume, diversity, and frequency of daily transactions. Individuals often struggle to maintain accurate expense records, analyze spending behavior, and derive meaningful insights from fragmented financial data. Conventional expense management applications largely depend on manual data entry and static categorization, which not only demands sustained user effort but also increases the likelihood of errors, inconsistencies, and incomplete records. With the rapid advancement of Artificial Intelligence, particularly in natural language processing and

intelligent automation, new opportunities have arisen to transform how users interact with financial systems. The Opti-Budget Engine is introduced as an AI query-driven expense management and budget monitoring system designed to simplify and modernize personal finance handling

1.1 Background Survey

The growing complexity of modern lifestyles has significantly increased the need for efficient personal finance management systems. With the rise of digital payments, online subscriptions, and frequent micro-transactions, individuals generate large volumes of financial data daily. Traditional expense management systems typically require users to input transaction details through structured forms, which is time-consuming and prone to inconsistency. Recent advancements in Artificial Intelligence, particularly in natural language processing and intelligent agents, have introduced new possibilities for improving user interaction with software systems. However, in many existing finance-related applications, AI functionality is confined to providing suggestions without direct authority to modify the underlying financial database. This gap forms the foundational motivation for developing an intelligent, AI-driven expense management platform.

1.2 Problem Statement

Despite the widespread availability of digital expense tracking applications, individuals continue to face significant challenges in effectively managing their personal finances. Existing systems largely depend on manual data entry, rigid input forms, and predefined categories, which require continuous user effort and discipline. This repetitive process often leads to delayed updates, incomplete records, and inaccurate categorization. Additionally, most conventional tools offer limited interaction mechanisms, restricting users to static dashboards and fixed query options. Users are unable to interact with their financial data in a natural manner using everyday language. Another critical limitation lies in the lack of real-time synchronization between user actions, data processing, and visual analytics. Therefore, the

core problem addressed in this project is the absence of an intelligent, user-centric expense management system that can seamlessly interpret natural language inputs, perform direct database operations, and present real-time, meaningful financial insights.

1.3 Motivation

Effective personal financial management has become increasingly important in an era characterized by frequent digital transactions and diverse spending channels. Despite the availability of numerous expense tracking applications, many users struggle to maintain consistent financial records due to the repetitive nature of manual data entry. Another key motivation arises from the gap between how users naturally think about their finances and how traditional systems require them to interact. Users tend to recall expenses in conversational terms rather than filling structured forms. The motivation behind developing the Opti-Budget Engine is to eliminate this disconnect by transforming expense management into an intuitive, conversational experience. By enabling users to interact with their financial data using natural language and allowing an intelligent AI agent to directly execute database operations, the system significantly reduces user effort while improving accuracy and consistency.

1.4 Aim and Objectives

Aim: The primary aim of the Opti-Budget Engine is to develop an intelligent, AI-driven expense management and budget monitoring system that simplifies personal finance tracking by minimizing manual intervention and enabling natural, conversational interaction with financial data.

Objectives: The key objectives are:

- To automate the process of expense recording and management by interpreting user intent expressed through natural language commands, thereby reducing dependency on traditional form-based data entry.
- To enable conversational querying of financial data, allowing users to retrieve summaries, analyze spending patterns, and modify records through an interactive chat interface.
- To provide real-time visual analytics through dynamic dashboards and charts to support better financial awareness and informed decision-making.
- To ensure data portability and usability by generating structured, professional financial reports in PDF format.
- To include a manual fallback mechanism to maintain data accuracy and user control, ensuring reliability in scenarios where conversational input may require precise adjustments.

2. LITERATURE REVIEW

Johri et al. proposed an expense management system that digitizes daily financial transactions using category-based logging, centralized storage and summary-oriented financial reports [1]. While it improves accuracy over manual bookkeeping, the system still depends on explicit user-driven data entry and static workflows, with no intelligent intent handling or automated database actions. Bhatele et al. introduced TrackEZ, a web-based categorized expense tracker that combines structured input forms with dashboard visualizations to support short-term financial analysis [2]. However, its reliance on manual form filling and predefined categories limits scalability toward intelligent, adaptive automation.

Sabab et al. presented eExpense, an early digital platform that simplifies everyday expense tracking through ease of entry and periodic summaries [3]. The approach reduces reliance on physical logs but remains deterministic and retrospective, lacking adaptive intelligence or natural language interaction. Naik et al. explored automatic expense tracking systems using semi-automated, rule-based pipelines to capture and process transactions [4]. Their framework reduces manual intervention but remains system-triggered rather than user-intent-driven, with no conversational layer controlling the database.

Sahel et al. developed a web-based expense tracker focused on clean interface design, categorized expenditure views and improved accessibility [5]. Despite its usability, the system behaves mainly as a passive digital ledger, requiring continuous user effort and offering no intelligent querying. Singla et al. designed a daily expense tracker that enhances financial insight through categorized visualization and summary analytics [6], yet its intelligence is confined to post-processing, without real-time conversational control or AI-driven database actions.

Kamra et al. proposed a full-stack expense monitoring system that integrates frontend, backend and database layers to achieve modular, end-to-end architecture [7]. The solution nonetheless operates on predefined, user-triggered operations and does not employ large language models or agentic AI to execute financial commands. Girdhar et al. focused on a mobile expense app emphasizing portability and ease of use [8], but the interaction remains static, without intelligent automation or learning from user behavior.

Raihan et al. investigated user readiness for AI-powered personal expense tracking and reported strong demand for intelligent, automated financial assistants [9]. Their findings support the transition from manual or semi-automated tools to conversational, AI-driven platforms. Finally, an enterprise expense reimbursement system based on association rule

mining demonstrates how structured data mining can optimize financial workflows in organizations but lacks conversational adaptability and user-level personalization [10]. Together, these works highlight a gap for systems like Opti-Budget Engine that combine conversational AI, direct database control and real-time analytics for personalized, intent-driven expense management.

Therefore, existing systems lack a unified platform that supports conversational expense entry, direct AI-driven database execution, and synchronized real-time analytics, which motivates the proposed Opti-Budget Engine.

3. SYSTEM DESIGN AND METHODOLOGY

The Opti-Budget Engine is designed as a layered, web-based application that tightly integrates conversational AI, backend business logic and structured financial data. The system follows an end-to-end methodology where user commands, given through natural language or uploaded bills, are translated into concrete database operations and real-time visual analytics. The overall design aims to minimize manual effort, maintain data consistency and provide immediate insight into personal spending behavior.

3.1 System Architecture

The system architecture adopts a four-layer structure: presentation layer, application layer, data layer and visualization and reporting layer. At the presentation layer, users interact through a web interface built with HTML, CSS and JavaScript. This interface provides a login page, dashboard, conversational chat window, manual expense entry forms and report download options, ensuring a responsive and user-friendly experience.

The application layer is implemented using a Flask-based backend that handles request routing, authentication, session management and coordination between the AI module and the database. Within this layer, an AI interaction module receives natural language inputs, performs intent recognition and entity extraction and maps each request to a valid financial operation such as add, query, update or delete.

The data layer consists of an SQLite database that stores structured expense records, including fields such as date, category, amount and description, along with user details and metadata. All CRUD operations originate from the backend logic, ensuring controlled access and data integrity. Above this, the visualization and reporting layer aggregates the latest expense data to generate real-time dashboards, charts and PDF reports so that analytical views stay synchronized with the underlying records

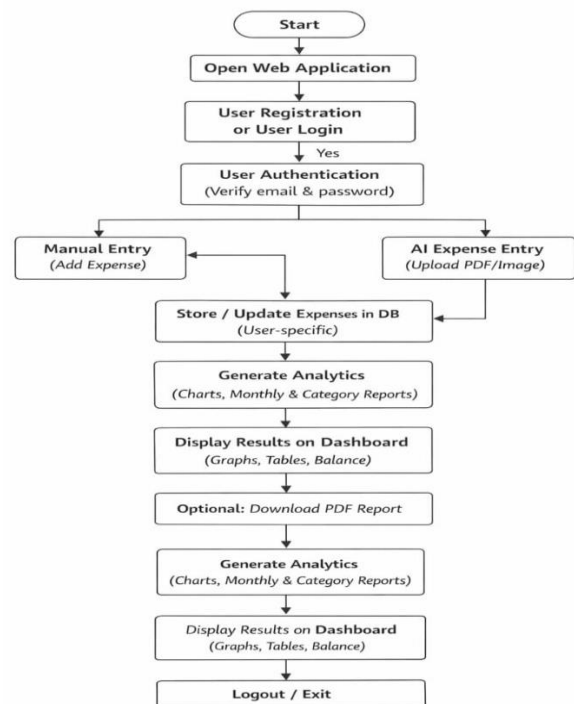


Fig -1: System Architecture and Operational Flow of Opti-Budget Engine

3.2 Data Flow and Sequence

The data flow of Opti-Budget Engine begins when an authenticated user issues a command through the web interface or uploads a bill/document. The frontend forwards this input to the Flask backend, which determines whether the request is conversational or manual.

Conversational inputs are routed to the AI processing module, where the language model interprets user intent and extracts key entities such as amount, category and date. Once identified, the backend translates the extracted information into database operations on the SQLite expense store, performing create, read, update or delete actions as required.

The updated expense records are then processed by the analytics component to recompute totals, category-wise distributions and monthly spending trends. These results are immediately reflected on the dashboard for real-time visualization. When the user requests documentation, the processed data is passed to the report generation module, which produces structured PDF reports. Overall, each request follows the sequence:

User Input → Backend Routing → AI Interpretation → Database Operation → Analytics Update → Dashboard/Report Output

This ensures that conversational actions result in persistent changes to the financial dataset and synchronized visual insights.

3.3 Use Cases

The primary actor in Opti-Budget Engine is the end user managing personal finances. After logging in, the user can add expenses either through natural language commands or by filling a manual form. In both cases, the transaction is stored securely in the expense database.

The user can issue conversational queries to obtain total spending, category-wise breakdowns or monthly summaries, and the system responds by fetching and aggregating relevant records. Additional use cases include editing and deleting expenses through chat commands or record selection, allowing users to correct mistakes and maintain accurate data.

The platform also provides real-time dashboards with key indicators and trend charts, along with professional PDF report generation for selected time periods. A manual fallback mechanism is available whenever conversational input is ambiguous, ensuring full user control over financial records.

4. IMPLEMENTATION

The Opti-Budget Engine was implemented as a full-stack web application that integrates conversational AI, backend database control, and real-time financial analytics. The system was developed using a modular architecture to ensure secure expense storage, intelligent automation, and responsive visualization. The implementation focuses on enabling users to manage personal expenses through both manual forms and AI-driven natural language interaction.

4.1 Technology Stack

The Opti-Budget Engine was developed using a lightweight full-stack web technology framework to support intelligent expense tracking and real-time analytics. The frontend interface was implemented using **HTML, CSS and JavaScript**, providing responsive pages for login, dashboard, expense forms and AI chat interaction. The backend was built using the **Flask framework in Python**, which manages routing, authentication and coordination between the AI module and the database. Expense records are stored in an **SQLite database**, ensuring structured and user-specific transaction management. The conversational AI component is integrated through the **HuggingFace API**, enabling natural language understanding and automated expense handling. Interactive dashboards and charts are generated using visualization libraries, and a PDF export module is included to produce professional financial reports.

Table 1: Technology Stack Used

Module	Technology
Frontend	HTML, CSS, JavaScript
Backend	Python Flask
Database	SQLite
AI Model Access	HuggingFace LLM API
Visualization	Chart-Based Dashboards
Reporting	PDF Report Generator

4.2 AI Agent and Database Control

A major contribution of Opti-Budget Engine is the integration of an AI expense agent that performs direct database-grounded operations instead of functioning as a simple chatbot. Users can interact with the system through conversational commands such as *“Add ₹500 spent on food today”* or *“Show my monthly expenses.”* The AI module interprets the user’s intent and extracts key entities including amount, category and date. These extracted details are validated by the Flask backend and translated into corresponding CRUD operations on the SQLite expense database.

Once an expense is added, updated or deleted, the system immediately recalculates totals, category-wise spending and monthly trends, ensuring real-time synchronization between stored data and dashboard analytics. This agentic AI-database integration reduces manual effort, improves accuracy and enables an intelligent, user-centric finance management experience.



Fig -2: AI Expense Assistant Interface for Conversational Queries

5. RESULTS AND DISCUSSION

The Opti-Budget Engine was successfully implemented and tested as an AI query-driven expense management platform. The system demonstrates effective integration of conversational AI, real-time database control and interactive financial dashboards. The results confirm that the platform

reduces manual effort while improving usability and accuracy in personal finance tracking.

5.1 Functional Results

The developed system provides complete expense management through both manual and AI-assisted workflows. Users can add expenses using structured forms or natural language commands through the AI assistant interface. All transactions are stored securely in the SQLite database and can be retrieved instantly through conversational queries or list views.

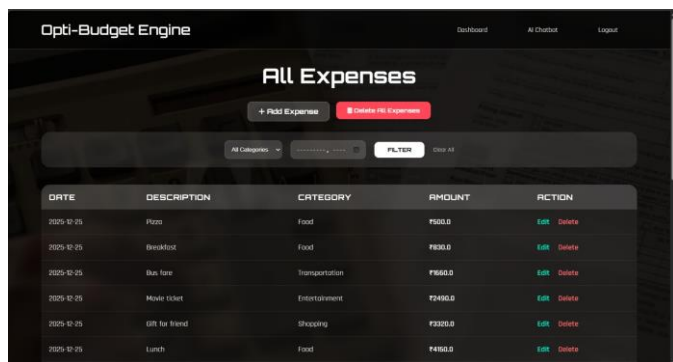


Fig -3: Expense History View with Filtering and CRUD Operations

The dashboard module displays real-time analytics including total spending, balance estimation, category-wise expense distribution and monthly spending trends. CRUD functionality was verified through successful expense editing and deletion, ensuring data consistency. Additionally, the system supports PDF report generation, allowing users to export structured summaries of their financial activity. Overall, the functional testing confirms that Opti-Budget Engine delivers an intelligent, responsive and user-centric solution for modern expense monitoring.



Fig -4: Real-Time Expense Dashboard with Category Breakdown

5.2 Comparative Analysis

A comparison with conventional expense tracking systems highlights the advantages of the proposed approach. Traditional applications rely heavily on manual data entry and static dashboards, whereas Opti-Budget Engine introduces conversational AI with direct database execution and real-time synchronization.

Table 2: Comparative Analysis of Expense Trackers

Feature	Traditional Trackers	Opti-Budget Engine
Expense Entry	Manual form	Manual + AI
Natural Language Query Support	Not available	Fully supported
Database Control	User-driven	AI-controlled CRUD
Dashboard Updates	Static	Real-time
Report Generation	Limited	PDF export
Interaction Style	Fixed UI	Conversational

6. CONCLUSION AND FUTURE WORK

6.1 Conclusion

This paper presented the Opti-Budget Engine, an AI query-driven expense management and budget monitoring system designed to simplify personal finance tracking. Unlike traditional expense trackers that rely on repetitive manual data entry, the proposed system enables users to manage expenses through natural language interaction supported by an integrated AI agent. The system performs verified database operations directly on an SQLite backend and provides real-time dashboards, category-wise analytics and professional PDF report generation. The results demonstrate that Opti-Budget Engine improves usability, reduces user effort and enhances financial awareness through intelligent automation and synchronized visualization.

6.2 Future Work

Future enhancements of the Opti-Budget Engine can focus on expanding automation and scalability. Integration with real-time banking and digital payment APIs can enable automatic transaction import without manual input. Advanced machine learning models may be incorporated for expense forecasting, anomaly detection and personalized budgeting recommendations. Additional improvements such as multilingual conversational support, mobile application development and enhanced document-based receipt understanding can further increase accessibility and adoption. These extensions would strengthen the system as a complete intelligent financial assistant for modern users.

REFERENCES

- [1] E. Johri, P. Desai, P. Soni, H. Jain, and N. Sangneria, "Expense Management System," Proceedings of the 4th IEEE Global Conference on Computing, Power, and Communication Technologies, IEEE, 2023.
- [2] P. Bhatele, D. Mahajan, B. Mahajan, N. Mahajan, and P. Mahajan, "TrackEZ Expense Tracker," Proceedings of the 4th International Conference on Intelligent Computing and Communication, IEEE, 2023.
- [3] S. A. Sabab, S. S. Islam, Md. J. Rana, and M. Hossain, "eExpense: A Smart Approach to Track Everyday Expense," Proceedings of the 4th International Conference on Electrical Engineering and Information & Communication Technology, IEEE, 2018.
- [4] S. K. Lokesh Naik, G. Ravi Kumar, A. Kiran, D. Ganesh, N. K, and M. Madgi, "Automating Financial Management: An Exploration of Automatic Expense Tracking Systems," Proceedings of the International BIT Conference on Emerging Trends in Engineering and Technology, IEEE, 2024.
- [5] M. Sahel, S. Ismail, S. Sohail, S. Srilekha, and F. Unnisa, "Streamlining Personal Finances: An Expense Tracker Website Study," Proceedings of the International Conference on Advances in Computing and Communication, IEEE, 2024.
- [6] S. Singla, A. Kaur, Anju, A. Soni, R. Dhaiya, and G. K. Kaur, "Unveiling Financial Insights: The Daily Expense Tracker System Approach," Proceedings of the International Conference on Intelligent Systems and Applications, IEEE, 2024.
- [7] V. Kamra, T. Saini, N. Shishodia, M. Singh, and P. Sehrawat, "A Novel Approach Towards Full Stack Comprehensive Expense Monitoring System," Proceedings of the International Conference on Emerging Smart Technologies, IEEE, 2025.
- [8] G. Girdhar, S. Kumar, A. Bhardwaj, and M. Sharma, "Design and Development of Expense App," Proceedings of the International Conference on Smart Computing and Informatics, IEEE, 2024.
- [9] S. M. R. Raihan, I. Asad, K. W. Rahman, Md. A. Rahman, S. Banik, and M. Hasan, "From Manual Logs to Smart Finance: Assessing User Readiness for AI-Powered Personal Expense Tracking," Proceedings of the International Conference on Artificial Intelligence and Data Engineering, IEEE, 2025.
- [10] "Design of Enterprise Financial Sharing Expense Reimbursement System Based on Data Association Rule

Algorithm," Proceedings of the International Conference on Data Science and Business Analytics, IEEE, 2022.

BIOGRAPHIES



Professor with 12 years of experience in Electrical, Electronics, and Communication Engineering, specializing in curriculum development, innovative teaching, and academic leadership. Served in key roles including Principal Incharge, Head of Department, and Controller of Examinations, while fostering critical thinking and interactive learning among students.



Final-year Computer Science Engineering student with a strong foundation in C, Java, Python, and SQL. Interested in artificial intelligence, web and app development, with a focus on building optimized solutions to real-world problems. Seeking opportunities to apply technical skills, contribute to impactful projects, and gain practical experience in dynamic, innovative environments.



Final-year Computer Science Engineering student, dedicated and sincere with a foundational understanding of programming and software concepts. Eager to learn new technologies and begin a professional journey, with a strong interest in gaining real-world experience through internships or entry-level opportunities. Focused on applying knowledge, growing as a developer, and contributing to impactful projects.



Final year Computer Science Engineering student with skills in Python full stack development and data science using Python, with contributions to advanced typing machines and movie recommender systems. Focused on applying knowledge, growing as a

developer, and contributing to impactful projects



Final year Computer Science Engineering student with research interests in Data Structures and Algorithms, software development, and web-based systems. Has practical experience in building a movie recommender system and is motivated toward solving real-world problems through computing.