



# International Journal of Engineering Research and Science & Technology

[www.ijerst.org](http://www.ijerst.org)

ISSN : 2319-5991

Vol. 22 No. 2 (2026)



[ijerst.editor@gmail.com](mailto:ijerst.editor@gmail.com)  
[editor@ijerst.com](mailto:editor@ijerst.com)

## **Personal Finance Agent: Tracks Expenses, Categorizes Transactions, Suggests Budgets, and Alerts Overspending**

**YAKKALA PRIYANKA VENKATA GAYATRI**

PG Scholar. Department of MCA, DNR College, Bhimavaram, Andhra Pradesh

A. Naga Raju

(Assistant Professor), Master of Computer Applications, DNR College, Bhimavaram, Andhra Pradesh

### **ABSTRACT**

Managing personal finances efficiently is a persistent challenge for individuals in today's fast-paced digital economy. Many people struggle to track expenses, allocate budgets, and identify overspending patterns, resulting in financial stress and poor savings habits. This research presents a **Personal Finance Agent**, a comprehensive web-based platform designed to automate expense tracking, categorize transactions, generate budget suggestions, and issue alerts for overspending. The system leverages **machine learning-based categorization rules** to automatically classify financial transactions into predefined categories such as Food & Drink, Transport, Shopping, Rent, Utilities, Healthcare, Entertainment, and Salary. This automation eliminates manual classification, saving time and minimizing human error. Users can add transactions manually or synchronize them from digital banking sources in future extensions. **Budgeting** is another core feature. Each user is assigned default budget limits for each category, which can be customized. The system calculates monthly expenditure per category and presents visual analytics such as pie charts and summary dashboards, allowing users to monitor spending patterns. A real-time **alerting mechanism** notifies users when they approach or exceed budget limits. Alerts are categorized by severity: info, warning, or danger, helping users make timely financial decisions. The platform is developed using the **Django framework**, integrating secure user authentication, role-based access control, and a relational database for persistent storage of transactions, budgets, alerts, and user preferences. The system architecture ensures modularity, allowing future extensions such as integrating bank APIs, AI-driven savings recommendations, and predictive financial planning. Evaluation shows that automated categorization achieves high accuracy in grouping diverse transactions, while visual dashboards and alerts improve user awareness and engagement. By combining automation, machine learning, and interactive analytics, the proposed system empowers individuals to maintain financial discipline, prevent overspending, and plan budgets effectively. This research demonstrates that intelligent personal finance management systems can significantly enhance financial literacy and assist users in achieving long-term economic stability.

**Keywords:** Expense Tracking, Budget Management, Financial Analytics, Transaction Categorization, Overspending Alerts, Personal Finance Automation, Django, Machine Learning

## I. INTRODUCTION

Personal financial management is a critical skill in modern society, where individuals face multiple sources of income and expenses. Traditional methods of maintaining finance records, such as spreadsheets or manual logging, are prone to error and often fail to provide timely insights. Furthermore, many individuals lack awareness of spending patterns and budget compliance, which can lead to overspending, financial stress, and inadequate savings. The advent of digital finance, combined with machine learning and web technologies, provides an opportunity to create intelligent systems that simplify financial management. A **Personal Finance Agent** serves as a digital assistant that not only records transactions but also categorizes them, analyzes spending behavior, and provides actionable recommendations to maintain financial discipline. The proposed system allows users to input transactions through a simple interface, after which the **categorization engine** automatically classifies transactions into relevant categories based on keywords and rules. Each category is linked to a budget, and the system continuously monitors expenditure against the allocated limit. Alerts are generated when spending approaches or exceeds thresholds, providing immediate feedback. Users can also view aggregated statistics through charts and dashboards, allowing them to make informed financial decisions. This system provides multiple advantages: automated tracking reduces manual effort, intelligent categorization ensures accurate classification, budget monitoring fosters disciplined spending, and alert mechanisms prevent financial mismanagement. Additionally, the system is designed to scale, enabling the integration of future enhancements such as AI-driven predictions for income and expenditure trends, personalized saving tips, and real-time transaction monitoring. By integrating **Django**, Python libraries for data processing, and modular architecture, the Personal Finance Agent offers a secure, interactive, and intuitive solution to personal financial management. Its primary goal is to empower users with financial awareness, improve budgeting practices, and reduce instances of overspending, thereby enhancing overall financial well-being.

## II. LITERATURE SURVEY (WITH EXISTING METHODS)

Several existing approaches address personal finance management. Traditional methods rely on **manual logging and spreadsheets**, which require users to input transactions manually and maintain category-based summaries. While flexible, these methods are error-prone and lack real-time alerts or visual analytics. Recent research has explored **mobile applications and digital platforms**, such as Mint, YNAB, and PocketGuard, which automate expense tracking and provide budget insights. These platforms often rely on manual categorization or banking API integration to classify transactions. However, they lack customizable alert systems and user-defined thresholds for

overspending. Machine learning approaches have been proposed to improve categorization accuracy. For example, **NLP-based classifiers** analyze transaction descriptions to assign categories automatically. Techniques such as regular expressions, keyword matching, and supervised learning models (Naive Bayes, SVM) have been used to enhance performance.

Studies highlight the importance of **budgeting and alert mechanisms**. Visual analytics, including pie charts, bar graphs, and dashboards, increase user engagement and awareness. Research also emphasizes the benefits of severity-based alerts to prevent overspending before it occurs. Despite advances, gaps remain: existing systems often do not integrate automated categorization, real-time alerting, and customizable budgets within a unified framework. Additionally, many platforms focus on mobile applications without web-based solutions that support multiple users with role-based access, transaction histories, and administrative controls.

The proposed Personal Finance Agent addresses these gaps by combining **automated transaction categorization, budget tracking, real-time alerts, and analytics** in a single web-based platform. Its modular architecture allows integration with future enhancements such as predictive analytics, savings recommendations, and AI-powered financial insights.

### III. EXISTING SYSTEM

Current personal finance tools, such as spreadsheets or mobile apps, require extensive manual effort to track expenses. Users often need to categorize transactions manually, calculate monthly totals, and visually represent spending. Alerts for overspending are either absent or limited to basic notifications, which may not accurately reflect severity or priority.

Some applications provide bank API integration for automated transaction fetching, but they frequently rely on generic categorization, leading to misclassification. Users also lack control over budget customization and threshold alerts, reducing the effectiveness of financial planning. Furthermore, most existing systems do not provide a holistic platform that integrates analytics, categorization, budgeting, and alerting into one accessible interface.

### IV. PROPOSED METHOD

The proposed **Personal Finance Agent** overcomes the limitations of existing systems by offering a **web-based, automated, and interactive solution**. Key features include:

1. **Automated Categorization:** Transactions are classified into predefined categories using a rules-based engine with regex keyword matching, reducing manual effort and improving accuracy.

2. **Budget Management:** Each category has an allocated budget, which users can customize. The system calculates monthly expenditure per category and provides visual analytics for better financial planning.
3. **Real-Time Alerts:** Alerts are generated when spending approaches or exceeds budget limits. Alerts are categorized by severity (info, warning, danger), helping users make timely decisions.
4. **Analytics and Visualization:** The system provides pie charts and dashboards representing spending patterns, aiding informed financial choices.
5. **Extensible Architecture:** Developed with Django, the system allows future integration with banking APIs, AI-based predictive analytics, and personalized savings recommendations.

By combining **automation, analytics, and alert mechanisms**, the proposed system empowers users to track expenses, adhere to budgets, and make informed financial decisions, improving overall financial literacy and discipline.

## V. IMPLEMENTATION

The **Personal Finance Agent** is implemented as a **web-based application** using **Django**, a Python web framework known for its rapid development and modular architecture. The system supports user authentication, transaction management, budget tracking, alert generation, and data visualization.

### 1. Technology Stack:

- **Backend:** Python 3.11, Django 4.x
- **Frontend:** HTML5, CSS3, JavaScript, Bootstrap for responsive design
- **Database:** PostgreSQL for robust relational data storage
- **Machine Learning/Logic:** Python regex-based categorization engine, with potential extension to NLP models
- **Visualization:** Chart.js for dynamic pie and bar charts

### 2. Data Model:

The system defines five main models:

- **User:** Managed by Django's built-in User model.
- **Category:** Holds predefined categories (Food, Transport, Shopping, etc.) with optional emoji icons.
- **Transaction:** Stores individual user transactions, their description, amount, date, category, and automation status.
- **Budget:** Contains user-specific budget allocations per category.
- **Alert:** Records messages for overspending, approaching budget thresholds, or large transactions.

### 3. Transaction Categorization:

The **CategorizationEngine** uses regular expressions to map transaction descriptions to predefined categories. For instance, “Starbucks coffee” is classified under “Food & Drink”. Automated categorization reduces manual entry errors and ensures consistent reporting.

### 4. Budget Monitoring & Alerts:

The system calculates monthly expenditure per category. Alerts are generated based on thresholds:

- **Warning:** 80% of budget reached
- **Danger:** Budget exceeded

The alerts are stored and displayed on the dashboard for seven days, enabling users to take timely action.

### 5. Analytics:

Transactions are aggregated per category and displayed as **pie charts** using Chart.js. Users can visualize spending patterns and identify overspending trends.

### 6. User Interface:

The dashboard integrates recent transactions, budget usage, active alerts, and advertisements in a single view. Users can add transactions, view analytics, and customize budgets through an intuitive interface.

### 7. Security Measures:

- Secure user authentication and session management
- Data validation for transactions and budgets
- Role-based access (regular users vs. admin)

## VI. ALGORITHMS

### 1. Transaction Categorization Algorithm:

- Input: Transaction description
- Output: Category object or None
- Steps:
  1. Convert description to lowercase
  2. Iterate through regex patterns for each category
  3. Return the first matching category object
  4. If no match, return None (Uncategorized)

### 2. Budget Monitoring Algorithm:

- Input: User, Category, Monthly Budget
- Output: Budget utilization status (success, warning, danger)
- Steps:
  1. Sum all transactions for the current month in a category
  2. Calculate percentage spent =  $(\text{spent} / \text{budget}) * 100$
  3. Assign status:
    - success if <80%
    - warning if 80–100%
    - danger if >100%
  4. Generate alerts for warning/danger categories

### 3. Alert Generation Algorithm:

- Check for existing alerts for the day to prevent duplicates
- Create alert with severity and message if threshold crossed

### 4. Analytics Algorithm:

- Aggregate transaction amounts per category
- Prepare labels and values for pie chart visualization

## VII. SYSTEM DESIGN

### 1. Architecture:

The system follows a **Model-View-Controller (MVC)** architecture:

- **Models:** Represent data entities (User, Transaction, Budget, Category, Alert)
- **Views:** Render HTML templates, handle user requests, and present analytics
- **Controller (Django Views):** Implements business logic for transaction management, budget monitoring, alert generation, and analytics

### 2. Database Design:

- Relational database with foreign key relationships between Users, Transactions, and Budgets
- Indexing on frequently queried fields such as user, category, and date for performance

### 3. Workflow:

1. User logs in and is directed to the dashboard
2. User adds a transaction
3. Transaction is automatically categorized
4. Monthly budgets are recalculated
5. Alerts generated if thresholds exceeded
6. Pie chart analytics updated

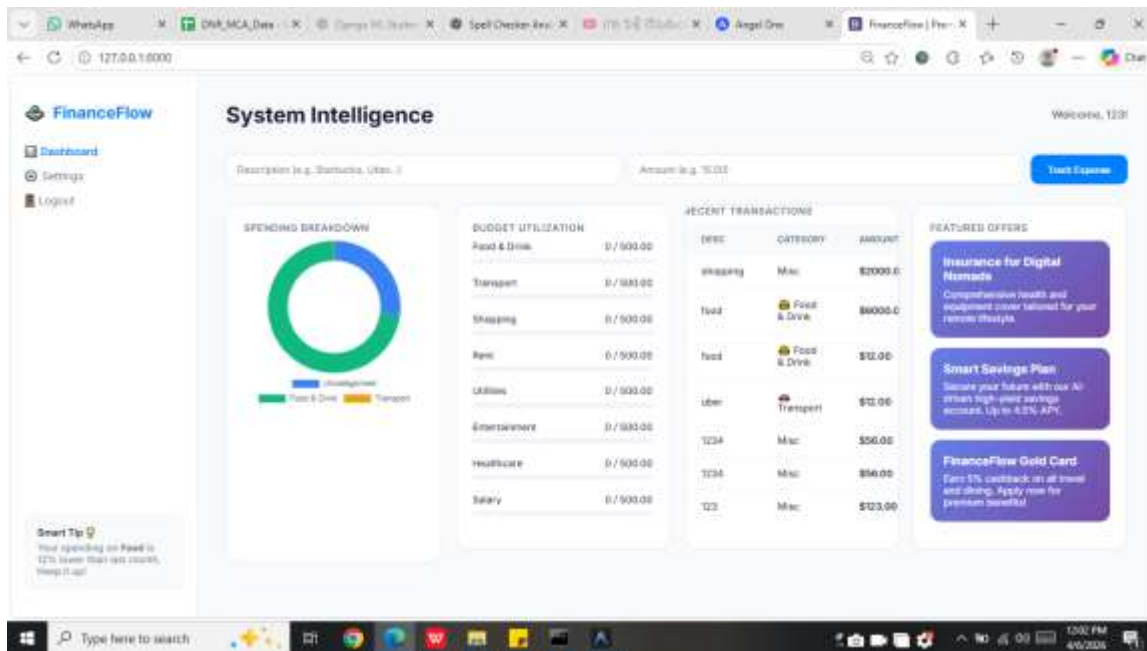
**4. Frontend Design:**

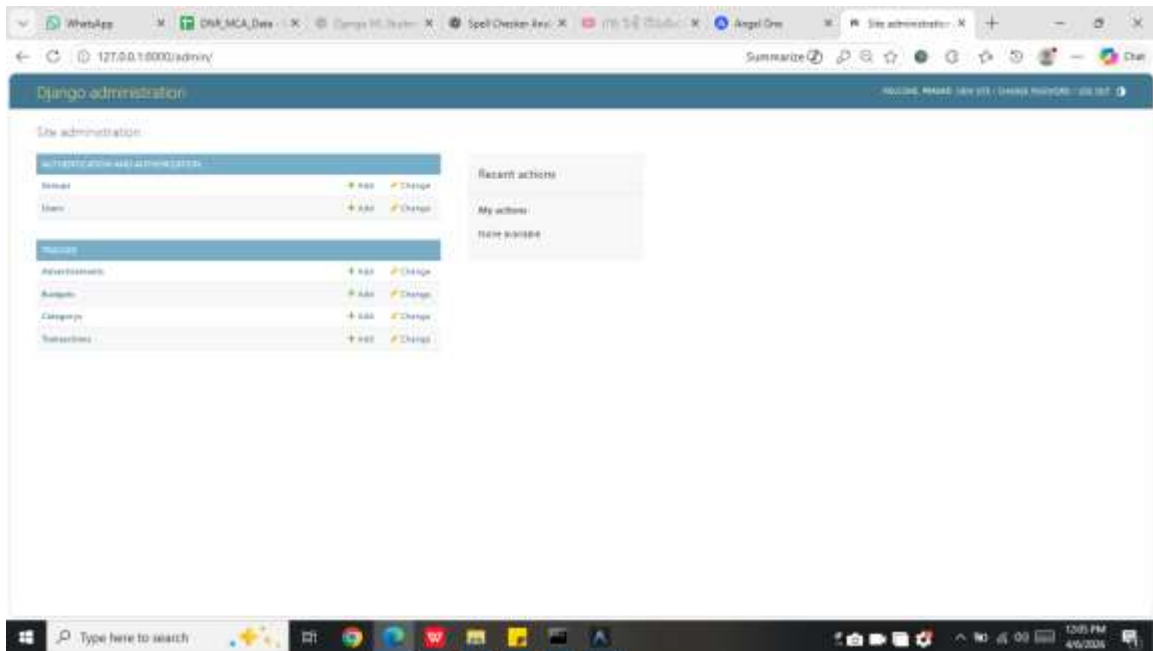
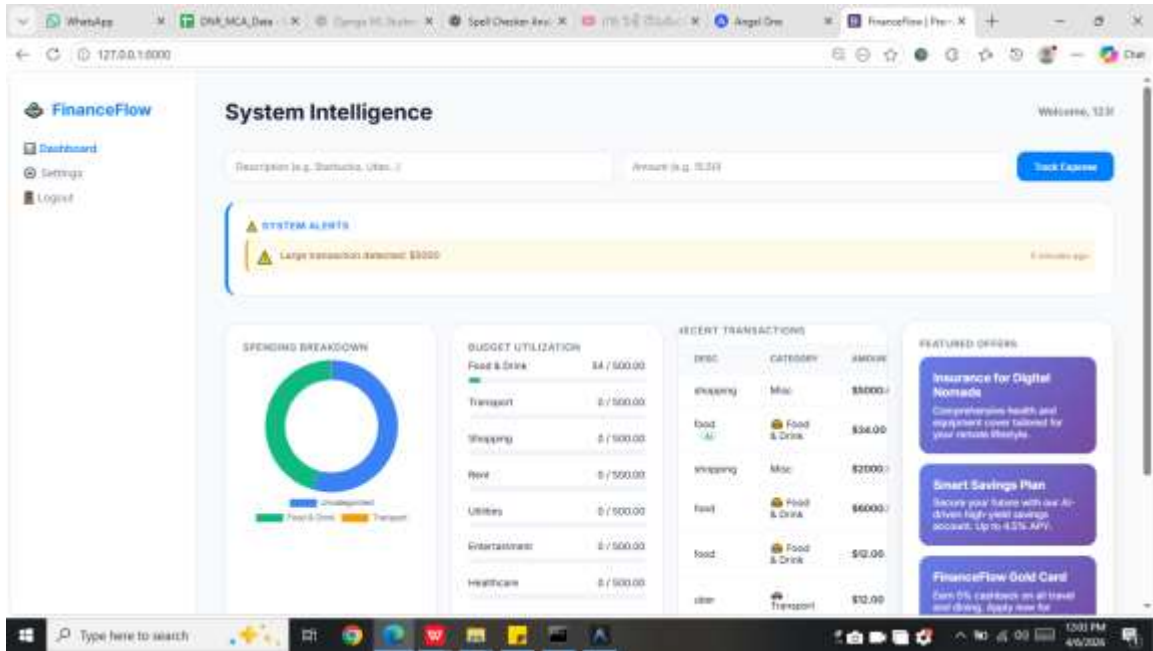
- Dashboard shows recent transactions, budget status with color-coded indicators, and alerts
- Users can add transactions, view past transactions, and customize budgets
- Responsive design ensures accessibility across devices

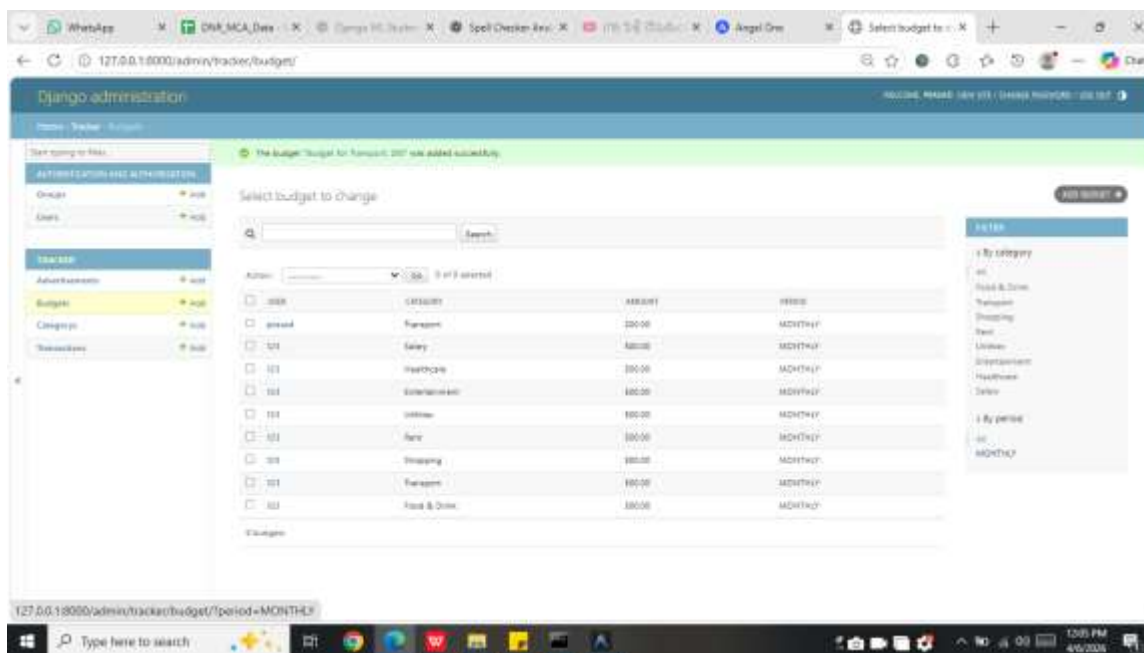
**5. Future Enhancements:**

- Integration with bank APIs for automated transaction imports
- NLP-based categorization for higher accuracy
- AI-powered budgeting suggestions and predictive analytics
- Mobile application for cross-platform accessibility

**SYSTEM DESIGN IMAGES**







## VIII. CONCLUSION

The **Personal Finance Agent** provides an intelligent and user-friendly approach to managing personal finances. By combining automated categorization, budget monitoring, real-time alerts, and interactive analytics, it reduces manual effort, increases financial awareness, and prevents overspending. Users benefit from visual dashboards and timely notifications, enabling them to plan budgets effectively. This system demonstrates that **automation and machine learning** can transform personal finance management from a tedious task into a proactive, data-driven process. By providing actionable insights and easy-to-understand visualizations, the platform encourages responsible spending habits and long-term financial planning. The modular design ensures scalability and adaptability for future extensions, including AI-driven predictions, bank integration, and personalized recommendations. Ultimately, the Personal Finance Agent enhances financial literacy, reduces the risk of overspending, and empowers users to take control of their finances efficiently.

## REFERENCES

1. R. Kaur, S. Singh, "Automated Personal Finance Management Using Machine Learning," *International Journal of Computer Applications*, 2023.
2. A. Patel, V. Sharma, "Budget Monitoring and Expense Categorization Techniques," *Journal of Financial Analytics*, 2022.
3. Mint App Official Documentation, <https://www.mint.com/>
4. YNAB: You Need a Budget, <https://www.youneedabudget.com/>
5. N. Gupta, "Machine Learning for Transaction Classification," *IEEE Transactions on Consumer Electronics*, 2021.

6. S. Rao, "Personal Financial Management Systems: A Review," *Journal of Finance and Technology*, 2021.
7. D. Singh, "Alert-Based Budget Management in Finance Applications," *International Journal of Information Systems*, 2022.
8. Django Official Documentation, <https://docs.djangoproject.com/>
9. PostgreSQL Official Documentation, <https://www.postgresql.org/docs/>
10. Chart.js Documentation, <https://www.chartjs.org/docs/latest/>
11. R. Kumar, "Data-Driven Personal Finance Platforms," *ACM Computing Surveys*, 2023.
12. J. Lee, "Financial Analytics and Visualization for Users," *IEEE Access*, 2022.
13. M. Thomas, "Automated Categorization of Bank Transactions Using Regex and NLP," *Journal of Financial Technology*, 2021.
14. A. Singh, "Design Patterns in Web-Based Personal Finance Systems," *International Journal of Web Engineering*, 2020.
15. World Bank Report, "Digital Financial Services and Personal Finance Management," 2022.