



Personal Finance Management Application

Gian Carlo Bata

College of Computer Studies
Camarines Sur Polytechnic Colleges
gibata@my.cspc.edu.ph

Mark Wayne Cleofe

College of Computer Studies
Camarines Sur Polytechnic Colleges
marcleofe@my.cspc.edu.ph

Kaye Khryсна Olores

College of Computer Studies
Camarines Sur Polytechnic Colleges
kaolores@my.cspc.edu.ph

1 Executive Summary

The Personal Finance Management Application (PFM) is a handy tool that will help people manage their finances more easily. With PFM, users will be able to track their daily expenses, set goals for saving, manage loans and debts, and create budgets for each month. The app is intended to be simple to use but powerful enough to cover all the important parts of managing personal finances.

What sets PFM apart is its use of Artificial Intelligence (AI) to provide personalized financial advice. By analyzing users' financial data, AI will help them make smarter financial decisions, offer customized budgeting tips, and predict future expenses. This makes the app ideal for young professionals, freelancers, and digital nomads who are looking for intelligent, user-friendly tools to manage their finances. With AI-driven insights, PFM aims to differentiate itself from other finance apps by offering users a more tailored and efficient way to manage their money.

2 Market Analysis

The Personal Finance Management Application (PFM) targets young professionals, freelancers, and tech-savvy individuals, students, families, and small businesses who need smarter tools for managing finances. Unlike competitors like Mint, YNAB, and Pocketguard, which focus on basic budgeting and expense tracking, PFM integrates AI for personalized financial advice. This unique combination caters to users seeking advanced, automated financial insights, positioning PFM as a standout in a market where AI-driven finance tools are becoming increasingly popular.

3 Technical Description

The Personal Finance Management Application is designed to empower users to take full control of their finances while enjoying a secure, intuitive experience. Below are the core functionalities and how artificial intelligence (AI), as an emerging technology, will enhance its features.

Core Features

- **Expense Tracking:** Users can manually log expenses or connect their bank accounts to automate tracking. Transactions are automatically categorized, giving users a clear picture of their spending.
- **Budget Management:** PFM helps users set realistic budgets and sends notifications when they are nearing or exceeding their limits. What makes this feature smarter is the integration of AI, which analyzes spending patterns and suggests ways to optimize the budget.
- **Debt Management:** Users can track their loans, credit cards, and other forms of debt. AI-driven algorithms will offer suggestions for efficient repayment strategies, helping users pay down debt faster.
- **Savings Goals:** Whether it's for an emergency fund or a major purchase, users can set up and track their savings goals. The app will provide actionable tips on how to reach these goals more quickly.
- **Data Visualization and Reports:** Users can generate reports and visual insights to better understand their financial health. This could be through income vs. expense graphs, savings progress charts, and more.

4 Development Plan

Project Overview

The finance management app aims to help users track their income, expenses, savings, and financial goals. It will feature an intuitive UI with interactive visualizations, secure user authentication, intelligent suggestions, and data storage. The app will support user profiles, transactions, and financial goal tracking.

Development Phases

Phase 1: Project Setup and Initial Features (1 Week)

Objective: Set up the development environment and implement basic features.

Tasks:

Set up a virtual environment and project structure

Plan the database structure and implement a database manager

Create a basic Flask app that allows users to sign in and sign up. Includes: username, email, and password

Implement password hashing using SHA-256

Create the basic homepage

Implement error handling and input validation for signing in and up.

Plan the AI/ML part of the application

Phase 2: Core Features (2 Weeks)

Objective: Develop the core functionality of the app.

Tasks:

- Build the database schema (tables for users, transactions, categories)
- Implement user account management (profile editing, password reset).
- Add data input functionality (income, expenses, and categories).
- Implement a system for users to set financial goals (e.g., savings target, debt payoff).
- Develop a live line graph to show real-time financial trends.
- Store transaction data in a database.
- Look for suitable datasets

Phase 3: Financial Reporting and Visualization (2 Weeks)

Objective: Add visualization and reporting features to enhance user experience.

Tasks:

- Design and implement financial reports:
 - Monthly income/expense summaries.
 - Category-wise breakdown (e.g., food, rent, entertainment).
- Add interactive graphs (e.g., pie charts, bar graphs) for financial data visualization.
- Implement budget vs. actual expense comparison graphs.
- Implement export functionality for reports (e.g., CSV or PDF).
- Clean found AI/ML datasets

Phase 4: Security, Data Integrity, and Performance Enhancements (2 Weeks)

Objective: Ensure app security and performance optimizations.

Tasks:

- Implement additional security measures:
 - Encrypt sensitive data (e.g., using stronger hash algorithms for passwords).
 - Add password strength validation and multi-factor authentication (MFA).
- Improve database query efficiency.
- Add backup functionality for user data (e.g., automatic database backup).
- Implement data validation (prevent incorrect data entry).
- AI/ML model planning and research

Phase 5: User Experience and Polishing (2 Weeks)

Objective: Refine the UI/UX and fix any bugs.

Tasks:

Improve the UI based on user feedback (enhance aesthetics and layout).
Add tooltips, help menus, and onboarding instructions for new users.
Ensure smooth user navigation and interaction within the app.
Test the app on different screen sizes and resolutions.
Perform bug fixes and code optimization.
Train and integrate the AI

Phase 6: Final Testing and Deployment (1 Week)

Objective: Test the app thoroughly and prepare for deployment.

Tasks:

Perform comprehensive testing:
Unit testing for individual components.
Integration testing to ensure all parts of the app work together.
User acceptance testing (beta test with real users).
Final bug fixing and performance enhancements.
Package the app for distribution.
Prepare a user guide and technical documentation.

Timeline

Total Timeframe: Approx. 12-16 weeks

Phase 1: 1 week

Phase 2: 2 weeks

Phase 3: 2 weeks

Phase 4: 2 weeks

Phase 5: 2 weeks

Phase 6: 1 weeks

Tools & Technologies

Development Environment: Flask

Data Storage: sqlite database

Security: Password hashing, input validation, MFA

Graphs/Visualization: data visualization libraries

Testing: Unit testing with pytest, manual UI testing

Future Enhancements

Mobile Version: Consider polishing the app to make it mobile-friendly.

Cloud Sync: Enable users to sync data across devices.

AI/ML Enhancement: Enhance AI-powered financial insights or predictions.

Transactions on behalf of users.

5 Financial Projection

- Revenue Model
- Revenue Projections (User growth)
- Cost Projections (Initial development and ongoing operating costs)
- Profitability Analysis(Expected profits and losses)

6 Marketing and Promotion Strategy

To capture our target audience, we will use a combination of digital marketing strategies.

- **Social Media Campaigns:** The team will use platforms like Instagram, TikTok, and Facebook to target younger users, focusing on financial tips and success stories.
- **Content Marketing:** Educational blog posts, videos, and webinars on topics like budgeting, savings, and financial planning will establish the app as a trusted resource in personal finance.

7 Legal and Regulatory Considerations

Data privacy, cybersecurity, and prevention of malicious actions are the primary legal and regulatory considerations when developing a finance management application. Since the application doesn't handle transactions on behalf of users, the developers can largely avoid the complexities of financial and anti-money laundering regulations by the **Banko Sentral ng Pilipinas (BSP)**. The following specifies the legal and regulatory requirements to keep in mind when developing the application:

Data Privacy

The legal framework governing Philippine data privacy is the *Data Privacy Act of 2012 (Republic Act No. 10173)*[\[1\]](#). The **National Privacy Commission (NPC)** oversees its implementation. The following are the main concerns of the act:

- **Consent and Data Collection:** One must obtain explicit consent from users before collecting, processing, or sharing any personal data (e.g., financial transactions, personal information).
- **Data Minimization:** Only collect data that are necessary for the operation of the app. Avoid collecting excessive personal information that is not essential for finance management functions.
- **User Rights:** Users have the right to access, correct, or delete their data. You need to provide a clear and easy way for users to exercise these rights.
- **Data Security:** One must implement organizational, physical, and technical measures to protect data from unauthorized access, breaches, or loss. Encryption, secure data storage, and user authentication protocols (e.g., two-factor authentication) are crucial.
- **Data Breach Reporting:** In the event of a data breach, the company is required to notify both the NPC and the affected individuals within 72 hours of becoming aware of the breach.

Cybersecurity and Prevention of Malicious Actions

Given the sensitive nature of financial data, one must take serious measures against malicious actions, both from external actors and internal developers. The *Cybercrime Prevention Act of 2012 (Republic Act No. 10175)*[\[2\]](#) lays the legal framework on what to and what not to do when it comes to cybersecurity in the Philippines.

- **Secure Software Development:** Implement best practices for secure coding, regular audits, and vulnerability assessments to prevent exploits or backdoors that could be used maliciously.
- **Prevention of Insider Threats:** Ensure that only authorized personnel have access to sensitive data and use role-based access controls (RBAC) to limit the scope of actions developers can take.
- **User Authentication:** Strong authentication mechanisms, such as multi-factor authentication (MFA), are crucial to protect against unauthorized access to accounts.
- **Encryption:** All sensitive data, including financial information, must be encrypted both in transit and at rest to prevent data breaches.
- **Audit Trails:** Maintain a detailed log of all actions within the app, especially those related to financial transactions, access to personal data, and administrative functions. This will help detect and prevent fraud or malicious activity from either developers or users.

References

- [1] National Privacy Commission (2012) *Data Privacy Act of 2012*, <https://privacy.gov.ph/data-privacy-act/>
- [2] Official Gazette of the Republic of the Philippines (2012) *Cybercrime Prevention Act of 2012*, <https://www.officialgazette.gov.ph/2012/09/12/republic-act-no-10175/>