



TITLE: PERSONAL EXPENSE TRACKING APP

NAME: NAMRA RAFIQUE

STUDENT ID: 25MEI10100

COURSE: INTEGRATED M TECH (CYBER SECURITY)

SUBMITTED TO: DR. SANDIP MAL

DATE: NOVEMBER 24, 2025

## **INDEX**

1. INTRODUCTION
2. OBJECTIVES
3. PROJECT SCOPE
4. FUNCTIONAL REQUIREMENT
5. NON-FUNCTIONAL REQUIREMENT
6. IMPLEMENTATION
7. SAMPLE OUT SCREENSHOT
8. CONCLUSION
9. REFERENCE

## 1. INTRODUCTION:

THIS PYTHON PROJECT BUILDS A PERSONAL EXPENSE TRACKER APP. THE TUTORIAL GUIDES VIEWERS THROUGH CREATING AN "EXPENSE" CLASS, STRUCTURING THE APP, AND IMPLEMENTING USER INPUT, SAVING TO A CSV FILE, AND SUMMARIZING EXPENSES. LEARN HOW TO CATEGORIZE EXPENSES AND TRACK A REMAINING BUDGET.

Personal Expense Tracking App. The goal of this project is to create a Python app that lets you (as a user) track and categorise your monthly expenses and help you budget. The app will let users type their expense category and amount directly into the terminal. It'll then save (append) that expense entry to a file. And finally, read the file to summarise the expense totals for that month.

It'll also tell the user how much they can spend for the rest of the month to stay in budget (which is a custom value decided by your app, e.g. \$2000).

⌚ App Requirements 1. Ask the user to add an expense (name, category, amount)

2. Save expense entries to a .csv file.

3. Read the file to summarise the expense totals for that month

4. Show the user how much they can spend for the rest of the month (to stay in budget)

❖ Bonus

1. Show expenses by category

2. Give the user a rough estimate of how much they have left to spend per day

- 💡 Recommended Project Structure The final project will consist of 2 files
- 1.expense.py: A class for creating and storing expense objects.
  - 2.expense\_tracker.py: The main application logic.

## 2. OBJECTIVE:

The objective of a personal tracker app is to help users achieve personal goals by providing tools to set, monitor, and stay motivated, whether those goals are related to fitness, productivity, or other areas of life. These apps help individuals break down large ambitions into manageable tasks, track progress, and maintain focus through features like habit tracking, progress charts, deadline notifications, and reminders. They also provide a centralised platform for accountability and can offer insights into performance over time.

### Key objectives of personal tracker apps

**Goal management:** To provide a structured platform for users to set personal and professional goals, whether for fitness, finances, or skill development.

**Actionable planning:** To help users break down large goals into smaller, more manageable tasks with specific deadlines.

**Progress monitoring:** To visually represent progress through charts and dashboards, giving users a clear overview of their performance over time.

**Motivation and accountability:** To keep users motivated with features like reminders, alerts for approaching deadlines, and the ability to see their own progress, which can reinforce commitment.

**Increased awareness:** To make users more aware of their habits and patterns, which can help them make positive changes and establish new routines.

**Centralised tracking:** To offer a single location for all goals and tasks, simplifying organisation and eliminating the need to use multiple apps or methods.

### 3. PROJECT SCOPE :

The project scope for a personal expense tracking app defines all the work, features, and limitations required to deliver the final product. It ensures all stakeholders are aligned on the project's boundaries, timeline, and budget, helping to prevent scope creep.

#### Core Project Scope Elements

A well-defined project scope statement for a personal expense app typically includes the following components:

**Project Objectives:** The primary goal is to provide users with an intuitive and efficient tool to track income and expenses, manage budgets, and gain insights into their spending habits to foster better financial health.

**Target Audience:** The app could target a wide range of users, from individuals and families to freelancers or small businesses, each with specific needs that will influence the features.

#### Key Features & Functionality (Deliverables):

**User Management:** Secure user registration, authentication (email/password, potentially biometric), and profile management.

**Expense & Income Tracking:** Manual entry of transactions with details like date, amount, category, and description.

**Categorisation:** Predefined and user-customizable expense and income categories for organised tracking.

**Budgeting Tools:** Ability to set spending limits for specific categories (e.g., food, transportation) and track progress toward those limits.

**Reports & Analytics:** Visualised reports using charts and graphs (daily, weekly, monthly summaries) to help users understand spending patterns.

**Notifications & Alerts:** Push notifications to remind users to log expenses, alert them when they are nearing or exceeding a budget limit, or notify them of recurring bills.

**Data Security & Privacy:** Robust security measures, including data encryption (both in transit and at rest) and compliance with relevant data protection regulations.

**Multi-Device Synchronisation:** Secure cloud synchronisation to allow users to access their data seamlessly across multiple devices (iOS, Android, web).

#### Exclusions (What's Out of Scope):

Integration with bank accounts for automatic transaction import (this is an advanced feature often excluded from a basic Minimum Viable Product, or MVP, due to complexity and security requirements).

Investment tracking or management of complex financial portfolios.

AI-powered personalised financial advice or predictive analytics (these can be considered for future enhancements).

In-app payment processing functionality (e.g., direct bill payment).

#### Constraints & Assumptions:

The project must be completed within a specific budget and timeline.

The app will initially be developed for a specific platform (e.g., Android or iOS) or as a cross-platform application using frameworks like Flutter or React Native.

**Users will have an internet connection to use features requiring data synchronisation or cloud storage.**

**Milestones:** Key dates for project phases, such as design completion, beta testing, and final launch in app stores.

**Team & Resources:** The personnel involved, including project managers, UI/UX designers, front-end and back-end developers, and QA testers.

### **Future Scope Possibilities**

Once the core app is built, future iterations could include advanced features like OCR (Optical Character Recognition) for receipt scanning, bank integrations, shared family budgeting, gamification, and AI-driven insights.

#### 4. Functional requirements:

User Management: Allow users to register, log in, manage profiles, and securely access their data.

Income and Expense Tracking: Enable users to manually input income and expense details, including amounts, dates, and descriptions.

Receipt Management: Provide the ability to attach images of receipts to expense entries, with optional Optical Character Recognition (OCR) for automatic data extraction.

Categorisation: Offer pre-defined and customizable categories for organising expenses (e.g., groceries, travel, utilities).

Budgeting: Allow users to set budgets for specific categories and receive alerts when nearing or exceeding their limits.

Reporting and Analytics: Generate visual reports, such as charts and graphs, to show spending patterns and financial trends over various time periods (e.g., weekly, monthly, yearly).

Account Integration: (Advanced feature) Integrate with banking and payment systems to automatically track transactions, with robust security measures.

Bill Reminders: Notify users of upcoming bill due dates to help them avoid late fees.

Data Export: Allow users to export their financial data for personal records.

## 5. Non-functional requirements:

**Security:** Safeguard sensitive financial data with strong security measures, including data encryption, multi-factor authentication, and biometric access (e.g., fingerprint, face recognition).

**Usability:** Ensure the app is intuitive and easy to use, with clear instructions and a user-friendly interface for seamless navigation.

**Performance:** The app should have fast response times for data entry, report generation, and other functions, even with a high volume of data.

**Reliability:** The app should be highly available with minimal downtime and have backup and recovery procedures to prevent data loss.

**Scalability:** The system should be able to handle growth in the number of users and the volume of data without a decline in performance.

**Compatibility:** The app should function consistently across different mobile devices, screen sizes, and operating systems.

**Maintainability:** The system should be designed to allow for easy updates, troubleshooting, and future enhancements

## 6. IMPLEMENTATION:

### User Management

Implement a secure sign-up and login system.

Use JSON Web Tokens (JWTs) for handling user sessions and authentication securely.

### Expense Tracking:

Allow manual entry for each transaction, including amount, date, and a description.

Develop features to add custom tags and notes for more context.

Implement an option to scan receipts using AI for automated entry, says MakeMyTrip MyBiz.

Provide functionality to delete or update existing expense records.

### Categorization:

Allow users to create custom expense categories (e.g., "Groceries," "Transportation," "Entertainment").

Implement a default set of categories for new users.

### Budgeting:

Enable users to create budgets for different categories.

Implement notifications and alerts for when users are approaching or exceeding a budget limit.

#### Data Management and Security:

Ensure robust data security, as financial information is sensitive.

Use secure storage solutions that offer scalability, such as IBM Cloud.

Implement multi-factor authentication and biometric access (fingerprint, face ID) to protect user accounts.

#### Reporting and Insights:

Provide visualizations like pie charts or graphs to show spending patterns.

Generate summary reports that filter by time period (e.g., weekly, monthly) and category.

Offer analysis of spending habits to help users identify areas for savings.

#### User Interface (UI) and User Experience (UX):

Design an intuitive and easy-to-navigate interface, notes

Use clear layouts for data entry and report viewing.

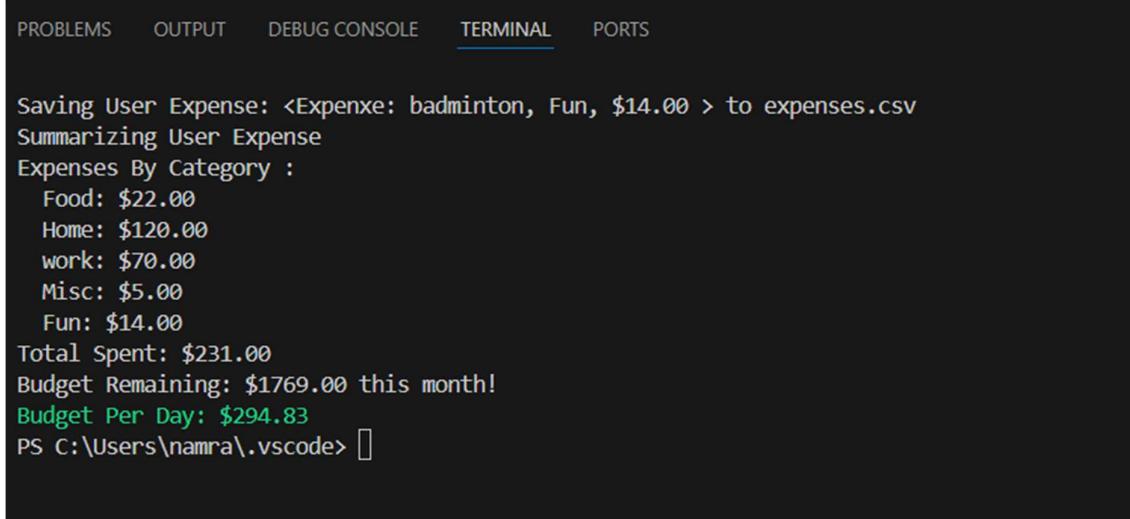
#### Additional Features:

Support multiple currencies.

Allow users to sync with bank accounts or credit cards for automatic transaction imports.

Integrate a customer support feature for user assistance.

## 7. SAMPLE OUTPUT SCREENSHOT:



The screenshot shows a terminal window with the following output:

```
Saving User Expense: <Expenze: badminton, Fun, $14.00 > to expenses.csv
Summarizing User Expense
Expenses By Category :
Food: $22.00
Home: $120.00
work: $70.00
Misc: $5.00
Fun: $14.00
Total Spent: $231.00
Budget Remaining: $1769.00 this month!
Budget Per Day: $294.83
PS C:\Users\namra\.vscode>
```

## 8. CONCLUSION :

Personal expense tracking provides financial clarity and control by revealing spending habits, helping to identify areas for saving, and reducing financial stress. Its value lies in empowering individuals to create and stick to realistic budgets, manage debt, and achieve long-term financial goals, such as saving for a major purchase or retirement. The modern landscape offers an array of tools, from manual spreadsheets and notebooks to automated, secure apps, that simplify the process and minimise the risk of human error associated with traditional methods.

## 9. REFERENCE:

[Flask documentation](#)

[SQLAlchemy documentation](#)

[Python official documentation](#)  
Abstract