Prepared by group 5

CampusExpense Manager: A Mobile Expense Tracker for University Students

Lai Duc Anh - BH00650
Hoang Tien Manh - BH00677
Nguyen Manh Cuong - BH00674
Dang Ngoc Chinh - BH00695

9 December, 2024

Project Overview

Brief Description:

CampusExpense Manager is a mobile application designed to help university students track expenses, set budgets, and analyze spending habits.

Goals:

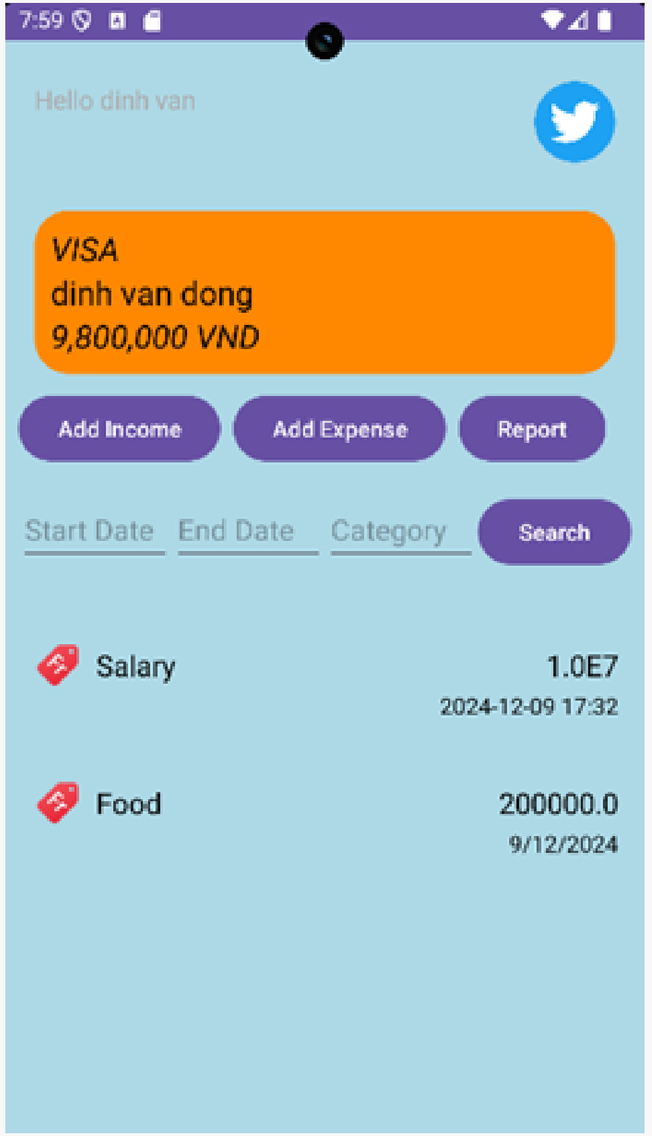
Simplify personal financial management, promote responsible spending, and provide actionable insights.

Target Audience:

University students managing their finances independently.

Objectives of the App

- Track and categorize expenses (e.g., rent, groceries).
- Set and monitor budgets for various categories.
- Provide an intuitive user interface for ease of use.
- Enable offline access to support students in low-connectivity areas.



User Requirements

- Key user needs identified during analysis:
- Expense tracking and categorization.
- Notifications for budget limits.
- Intuitive and user-friendly interface.
- Summary reports and spending analytics.

Systems Investigation and Research

Research Phase:
Competitor apps lack offline functionality, simplicity, and are too ads.

Insights Collected:
Students prefer straightforward, lightweight tools for tracking finances.

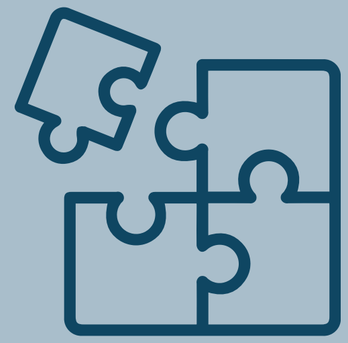
Priority Features:

Offline mode, easy data entry, real-time budget monitoring.

Project Scope and Constraints

Scope:

- Core Features: Expense tracking, budget setting, and notifications.
- Secondary Features: Spending trends, summary reports.



Constraints:

- Limited team experience in mobile app development.
- Strict 12-week development timeline.

Tools and Technologies Used

Programming Language:

Java

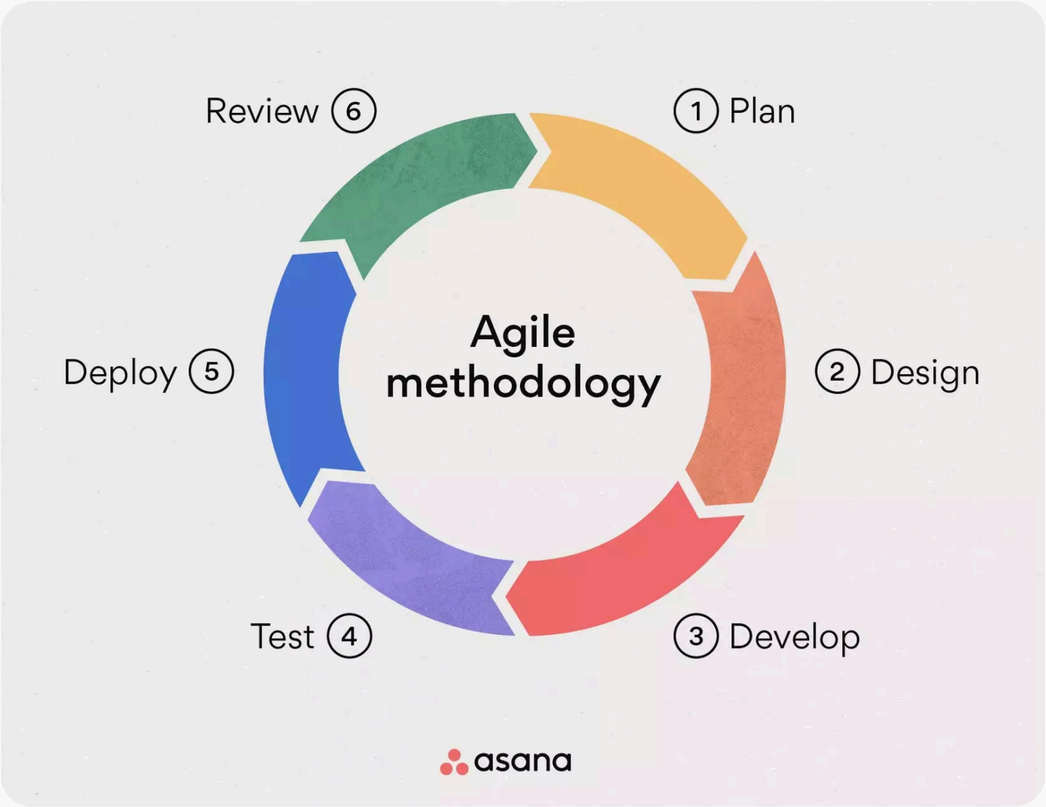
Database:

SQLite (local data storage for offline mode)

Development Environment:

Android Studio

Development Methodology



Methodology Chosen: Agile

- Iterative development with weekly sprints.
- Regular feedback from team and stakeholders.

Why Agile?

- Flexibility to adapt to changes.
- Continuous testing and improvement during development.

Initial Design and Prototyping

Process:

- Created wireframes using Figma.
- Focused on a minimalist and clean layout.

Feedback:

- Students suggested larger buttons for easier input and clearer category labels.

User Interface (UI) Design

Design Principles:

- Simplicity: Minimal clicks to add expenses.
- Consistency: Uniform design across all screens.

Example Screenshots:

- Login Screen
- Expense Entry,
- Summary Dashboard.

Backend Architecture

- SQLite Database:

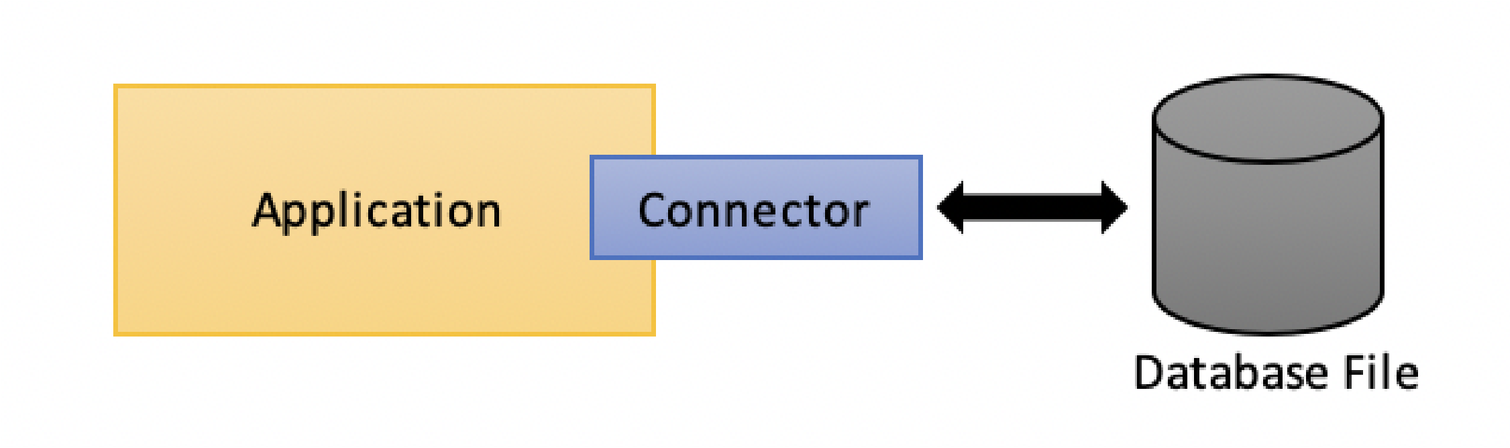
Used for offline data storage.

- Data Flow:

User inputs → Processed in Java → Stored in SQLite.

- Scalability:

Future potential to sync with a cloud-based server.

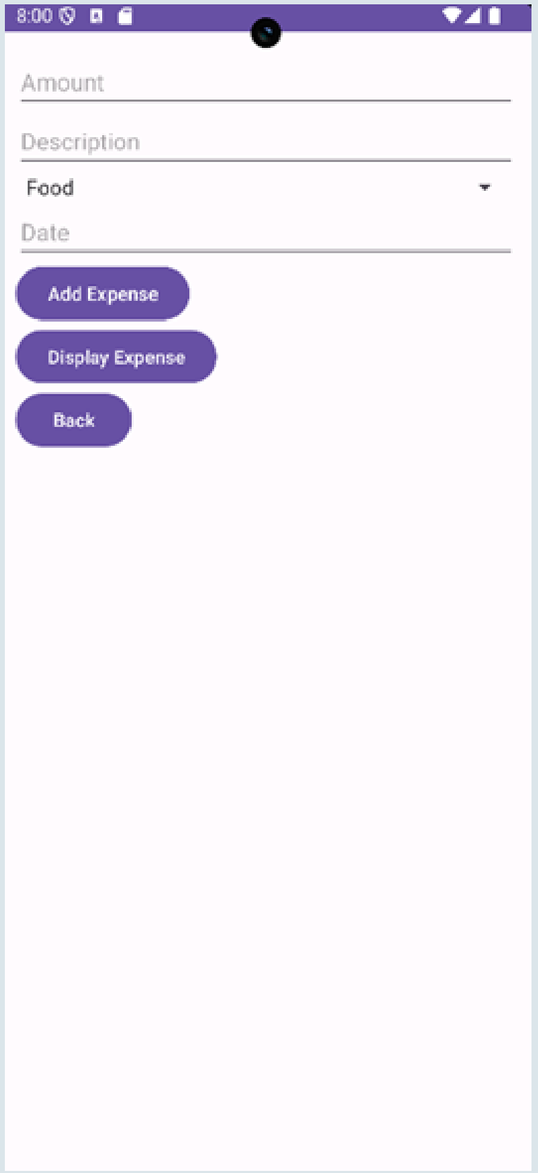


Core Features Implemented

- Expense Logging: Add, edit, and delete expenses.
- Budget Monitoring: Set monthly budgets with notifications for overspending.
- Summary View: Visual breakdown of spending by category.

Additional Features

- Manual Categorization: Users can create and edit expense categories.
- Spending Alerts: Notifications when a category approaches its budget.
- Prioritization: Based on user feedback, these features were secondary to core functionalities.



Data Management and Security

Data Management:

- Local storage in SQLite for offline functionality.

Security:

- Secure user authentication using encrypted passwords.
- Adherence to data privacy standards.

User Testing and Feedback

Testing Phase:

- Conducted with 10 university students.

Feedback Received:

- Positive: Ease of use, useful summary reports.
- Suggestions: Improved categorization and more intuitive navigation.

Changes Made:

- Adjusted dashboard layout and simplified workflows.

Challenges Faced

Technical Difficulties:

Integrating SQLite with Java for offline data storage.

Time Constraints:

Balancing feature implementation with testing phases.

Solutions:

Focused on core functionalities and deferred advanced features.

Thank you