

Expense Tracker Application: A Comprehensive Solution for Personal and Group Financial Management

Presented By: Sameer S Deshpande. NUID:[002685440]

Under the Guidance: Prof.Ahmad Rabah

Table of Contents

- 1. Introduction**
 - Problem Definition
 - Objective of the App
- 2. Design/Approach**
 - Key Features
 - Architecture
 - Database and Data Handling
 - Multi-User Support and Syncing
 - Login Screen
- 3. Screenshots and User Flow**
 - Login Screen
 - Adding an Expense
 - Viewing Spending Analytics
 - Currency Conversion in Action
- 4. Techniques Used (Above and Beyond)**
 - Real-Time Currency Conversion
 - Dark Mode and Light Mode Support
 - Efficient Database Management with SQLite
 - Multi-User Support and Syncing
 - Login and User Authentication
- 5. Conclusion and Future Work**
 - Summary of Current Features
 - Future Enhancements
- 6. References**
 - CurrencyLayer API
 - SQLite Documentation
 - MVVM Design Pattern

Problem Definition

In today's world, managing personal finances can be a complex and time-consuming task. With various types of expenses, from daily purchases to group expenses shared with friends or family, it becomes difficult to track and manage everything manually. Additionally, individuals often face challenges in keeping track of currency conversions when traveling abroad or dealing with multiple currencies.

Moreover, many people prefer using digital solutions for organizing and tracking their finances, but existing apps are often too complex, not user-friendly, or lack the ability to handle both personal and shared group expenses seamlessly. There is also a growing need for features like dark mode, which enhances user experience, especially in low-light environments, and multi-user support, which allows multiple individuals to track shared expenses.

This project addresses these challenges by offering a user-friendly, feature-rich app that helps users efficiently manage both their individual and group expenses. The app also includes currency conversion, light/dark mode, and the ability to sync across multiple devices.

Objective of the App

The primary objective of this expense tracker app is to provide users with an efficient and intuitive way to manage their personal and group finances. The app aims to:

1. **Track Individual Expenses:** Allow users to record and categorize their personal expenses, including the option to select the currency they are using, and automatically convert currencies when necessary.
2. **Track Group Expenses:** Enable users to create shared expenses with others, track who paid for what, and automatically calculate the split for each person involved in the group.
3. **Currency Conversion:** Integrate a currency conversion feature to facilitate the tracking of expenses in different currencies, with automatic updates to ensure the correct conversion rates are used.
4. **User Experience:** Provide a visually appealing and user-friendly experience with the option to toggle between dark mode and light mode based on user preferences.
5. **Multi-User Support and Syncing:** Implement multi-user support, allowing multiple users to track and manage shared group expenses, while ensuring that data syncs seamlessly across devices in real-time.
6. **Analytics and Reporting:** Provide users with insightful analytics on their spending patterns, and generate PDF reports summarizing expenses for personal or group financial reviews.

Design/Approach

In this section, we describe the design approach, key features, architecture, and handling of data to develop the expense tracker app. We also delve into the multi-user support, syncing, and the login system that ensures seamless functionality.

Key Features

The app provides several key features aimed at enhancing the user experience and functionality:

1. **Expense Tracking:**
 - Users can record both individual and group expenses.
 - Each expense includes details like amount, description, category, and payer.
 - Group expenses can be split among multiple users, with automatic calculations of who owes what.
 2. **Currency Conversion:**
 - The app allows users to manage expenses in multiple currencies, with real-time currency conversion.
 - It fetches live conversion rates to ensure accurate calculations.
 3. **Dark Mode / Light Mode:**
 - The app provides a toggle between dark and light modes, improving readability and user experience across various lighting conditions.
 4. **Analytics:**
 - Spending analytics are provided to help users visualize their expenses in charts and graphs.
 - Breakdown by categories like "Food", "Transportation", etc., helps users see where their money is going.
 5. **PDF Export:**
 - Users can generate and export PDF reports summarizing their expenses, ideal for reviewing or sharing expense data.
 6. **Multi-User Group Expense Management:**
 - Multiple users can track and manage group expenses. The app automatically calculates the share each user owes based on their contributions to shared expenses.
 7. **Multi-Device Syncing:**
 - Expenses and user data are synced across devices to ensure consistency across all platforms (e.g., iPhone, iPad).
-

Architecture

The app follows the **MVVM (Model-View-ViewModel)** architecture, which separates concerns into three components, making the code more modular, testable, and maintainable.

1. **Model:**
 - Represents the data structures for expenses, users, categories, etc.
 - It interacts with the database and handles data management tasks.
2. **View:**
 - The View layer is responsible for the UI components of the app.
 - SwiftUI is used to build the UI, allowing for declarative views that automatically update when data changes.
3. **ViewModel:**
 - The ViewModel acts as a bridge between the Model and View layers.
 - It contains the logic to fetch, update, and manipulate data (like adding an expense, fetching analytics, etc.).
 - It also handles any business logic related to data presentation, such as calculating totals or currency conversions.

This separation ensures that the UI is kept separate from the business logic, making the code easier to manage and scale in the future.

Database and Data Handling

The app uses **SQLite** for local storage to manage and persist data such as user expenses, categories, and group transactions. SQLite was chosen for its simplicity, performance, and ease of integration with iOS apps.

1. **Expense Data:**
 - Expenses are stored in the `expenses` table with fields like ID, amount, description, payer ID, category ID, etc.
 - Data is fetched from the database and displayed to users in real-time.
 2. **User Data:**
 - User-related information, including login credentials and user ID, is stored securely.
 - Multiple users are supported, and the app keeps track of each user's expenses.
 3. **Currency Data:**
 - Currency conversion is handled using live API calls to fetch real-time exchange rates.
 - A caching mechanism ensures that the data is updated periodically while minimizing external API calls.
 4. **Syncing:**
 - When users add or modify expenses, the changes are synced across devices using iCloud, ensuring that all users in a shared group have the most up-to-date information.
-

Multi-User Support and Syncing

One of the key features of the app is its ability to support multiple users for group expense management. This is achieved by:

1. **Shared Group Expenses:**

- Users can create group expenses, and the app tracks which user paid for which expense.
- Each user's share of the group expense is automatically calculated based on their contribution.

2. **Syncing Across Devices:**

- The app uses **iCloud** for data synchronization. Expenses added or updated on one device are reflected on all devices that are linked to the same user account.
- This ensures that multiple users can collaborate in real-time and track shared expenses efficiently.

3. **Real-Time Collaboration:**

- Group members can see updates instantly when a new expense is added or when an expense is settled, without needing to manually refresh the data.

Login Screen

The app includes a secure login screen that allows users to create an account, log in, and manage their personal finances. Here's an overview of the login system:

1. **User Authentication:**

- The app supports account creation via email/password, and uses **Firestore Authentication** or a similar secure method for login.
- Passwords are securely hashed and stored, ensuring that user data is protected.

2. **Session Management:**

- Once logged in, users can access their expenses and settings, and the app keeps track of the active session until the user logs out.

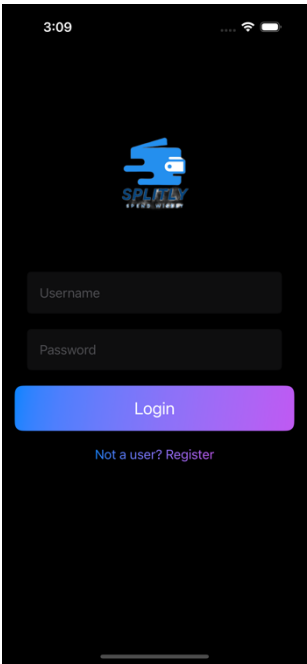
3. **Multiple User Support:**

- The login system allows multiple users to maintain separate accounts while sharing group expenses.
- Each user's data (expenses, categories, settings) is isolated and synced across devices using their user ID.

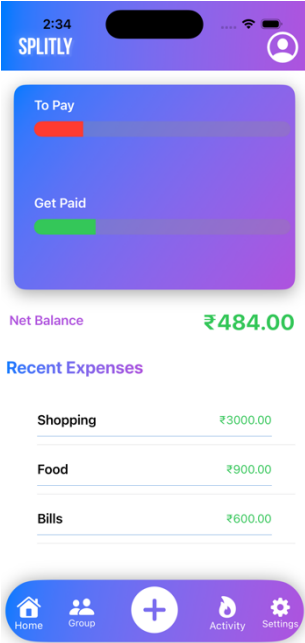
4. **User Profile:**

- Users can manage their profile settings, such as preferred currency and language, directly from the app.

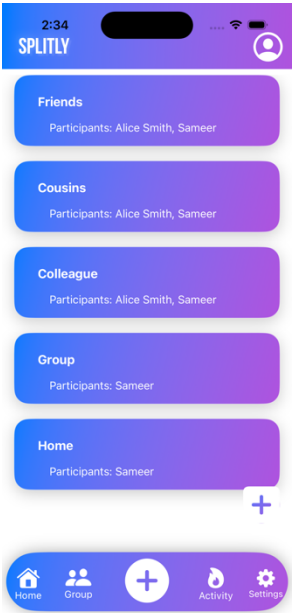
Screenshots and User Flow



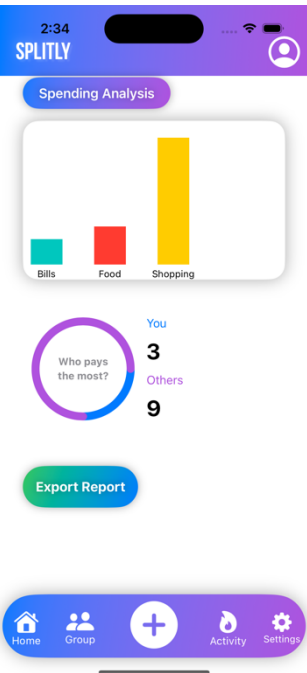
a. Login Screen



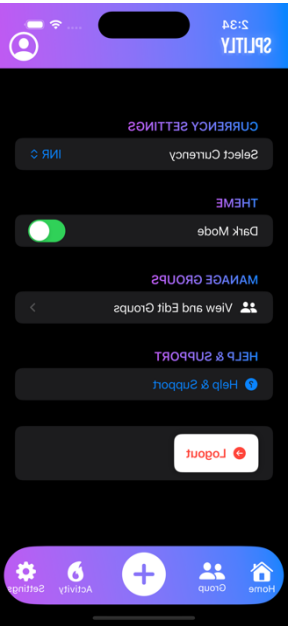
b. Dashboard



c. Adding groups



d. Analytics



e. Settings Page

Techniques Used (Above and Beyond)

- **Real-Time Currency Conversion:**
The integration of real-time currency exchange rates from a third-party API (e.g., Open Exchange Rates or CurrencyLayer) allows users to automatically convert their expenses based on current rates.
- **Dark Mode and Light Mode Support:**
The app respects system-wide theme settings (dark or light mode) and allows for a better user experience.
- **Efficient Database Management with SQLite:**
SQLite is used for fast and offline data storage. It supports efficient query execution and data retrieval, which is crucial for a smooth user experience.
- **Multi-User Support and Syncing:**
The app supports multi-user functionality. Users can log in with their credentials and sync their expenses across multiple devices. This ensures that group expenses are updated in real time and everyone can track who owes what.
- **Login and User Authentication:**
Secure user authentication is implemented so that users can have personalized experiences. Data is stored separately for each user, ensuring privacy and security.

7. Conclusion and Future Work

The core functionality of the **Splitly** app has been successfully implemented, including expense tracking, group expense splitting, user authentication, spending analytics, and multi-user support. The user interface is clean, responsive, and offers a personalized experience with support for light and dark modes.

Future Work:

1. **Expense Notifications:**
Implement push notifications to remind users to settle up when they have outstanding payments.
2. **Group Chat Integration:**
Add a feature to allow users to communicate within a group to discuss and settle expenses.
3. **Multi-Device Syncing and Cloud Storage:**
While multi-user support is available, expanding to full multi-device syncing across platforms (iOS, Android, Web) via cloud integration could be a future enhancement.
4. **Advanced Analytics:**
Implement advanced spending analytics, such as spending trends over time, and allow users to filter by date, category, or payer.
5. **Cloud Integration:**
Integrate cloud storage to back up data and enable users to access their expense data across devices.
6. **Improved User Authentication:**
Enhance the login screen with options like social login (Facebook, Google) for quicker sign-in.