Adama Science and Technology University

Course title: Mobile application and development

Project title : Expense tracker app (Documentation)

Department:software Engineering

Name	ID	section = 3
1)Hiba Ahmedhussen	UGR /30679/15	
2)Natati Birhanu	UGR/31029/15	
3) Abdulhafiz seid	UGR/30029/15	
4) yididiya admasu	UGR/31407/15	
5) Samuel Aklillu	UGR/31181/15	

Submitted to: Inst.Yared tekalign

Table of Contents

Mobile Application Design and Development Documentation

Project Title:Expense Tracker App: Simplified Finance Management for Everyone

1. Project Overview

The Expense Tracker App is a mobile application designed to help users efficiently manage their personal or small business finances. The core objective of the app is to provide a user-friendly platform where individuals can log their income and expenses, categorize spending, and visualize financial data through intuitive graphs and summaries. In a world where digital payments and spending are increasing, it is crucial for users to monitor their financial behavior in real time and make data-driven decisions.

The app targets users who seek a simplified yet effective financial management tool without complicated setup processes. Whether it is for daily expenses or monthly budgets, the Expense Tracker App empowers users to take control of their finances, develop better money habits, and achieve their savings goals.

Target Audience:

- Students
- Young professionals
- Freelancers
- Small business owners

2. User Requirements

Target Users:

- Individuals who want a lightweight and efficient way to track expenses
- People without financial or technical backgrounds
- Users who desire quick insights into their financial habits

User Needs & Features:

- Ability to add income and expense entries
- Categorize transactions (Food, Travel, Bills, etc.)
- View summaries via graphs (bar/pie chart)

- Filter transactions by date and category
- Responsive design for different mobile screen sizes
- Minimal steps to log a transaction

These requirements shaped the application's architecture, UI layout, and overall feature prioritization. User feedback (assumed through the lens of being stakeholders) directly influenced the focus on clarity, speed, and minimalism.

3. Design Concepts

Design Style:

The app embraces a clean and minimalist UI to ensure a smooth user experience. A light theme was chosen to enhance readability, with soft colors and gentle shadows to guide user focus without clutter.

Navigation Flow:

1. **Dashboard Screen**: Displays total balance, income, and expenses
2. **Add Transaction Screen**: Add income or expense with category and amount
3. **Transaction History Screen**: List of previous transactions, sortable and filterable
4. **Reports Screen**: Visual summaries in pie/bar charts

UI/UX Considerations:

- **Bottom Navigation Bar**: Easy access to all main screens
- **Minimal Taps**: Logging transactions can be done in 2–3 taps
- **Responsive Layouts**: Ensures proper appearance on various screen sizes
- **Instant Feedback**: Visuals and summaries update in real time

4. Project Scope

This section defines what the app **will** and **will not** cover. It ensures clarity for both developers and evaluators.

In Scope:

- Add, delete, and edit transactions
- Dashboard summary of financial data

- Report visualization (charts)
- Filtering transaction history

Out of Scope (Planned):

- Currency conversion
- Multi-user accounts
- Export to PDF
- Budget alerts

5. System Architecture Overview

Briefly describe the internal structure of the app and its integration with tools like Firebase. A simple architecture diagram may be included here.

- **Frontend:** Flutter widgets
- **State Management:** Provider
- **Backend:** Firebase Firestore for cloud data
- **Storage:** SharedPreferences (removed)
- **Future Integrations:** Firebase Authentication, PDF export libraries

6. Security and Privacy Considerations

- All financial data stored securely in Firestore with user-level access rules
- No sensitive data like passwords stored locally
- Will include Firebase Authentication for secure login (in future updates)
- Data encrypted in transit using HTTPS

7. Accessibility Features

- High-contrast, legible text
- Icons + color labels for quick interpretation

- Button and tap area size optimized for finger use
- Support for screen reader (planned)

8. Development Approach

Methodology:

We followed the **Agile development** approach, using weekly sprints to plan, design, code, test, and receive peer feedback. This methodology allowed flexible adjustments to UI and features based on internal testing.

Justification:

Agile is ideal for mobile development projects with short deadlines and evolving ideas. The ability to make incremental improvements helped in early feature delivery.

Challenges Faced:

- **State Management**: We implemented the Provider package for clean and reactive state handling.
- **Chart Integration**: Dynamic chart updates required performance optimization.
- **Cross-Device UI Testing**: Ensuring consistency across Android phones with various screen resolutions.

9. Technological Stack

Category	Technology/Tool Used	Purpose
Framework	Flutter (Dart)	Cross-platform development
UI	Material Design (Flutter)	UI elements and layout
State Management	Provider	Efficient state handling
Database	Firebase Firestore	Cloud data storage
Local Storage	SharedPreferences	Offline data access
Charts	fl_chart	Bar and pie chart visualizations

Date Formatting	intl	Formatting date and currency
Notifications	flutter_local_notifications (planned)	Reminders and budget alerts

Rationale:

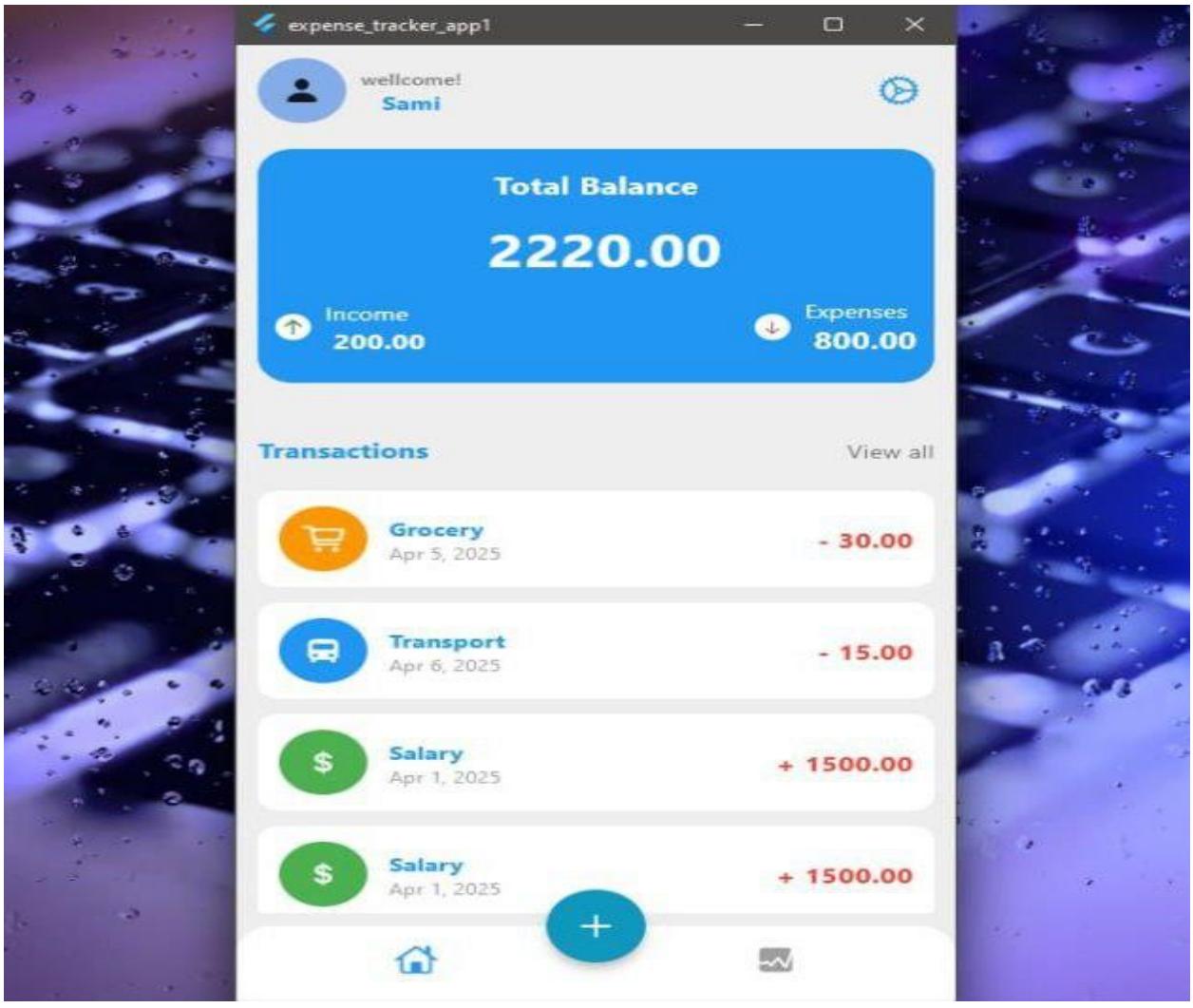
Flutter's cross-platform nature reduces development time. Firebase enables cloud sync and real-time updates, while SharedPreferences ensures basic functionality even offline.

10. Implementation Details

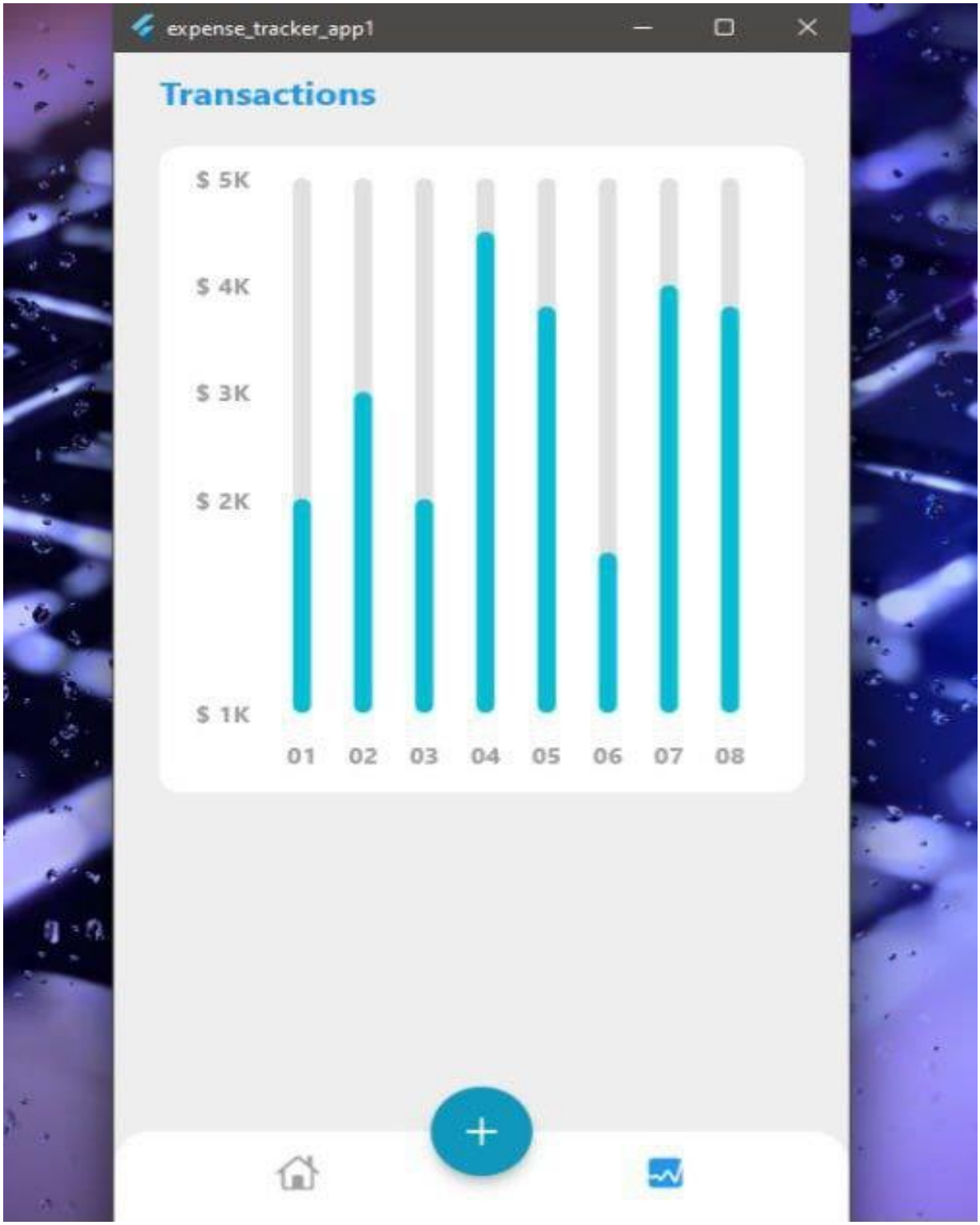
Key Features:

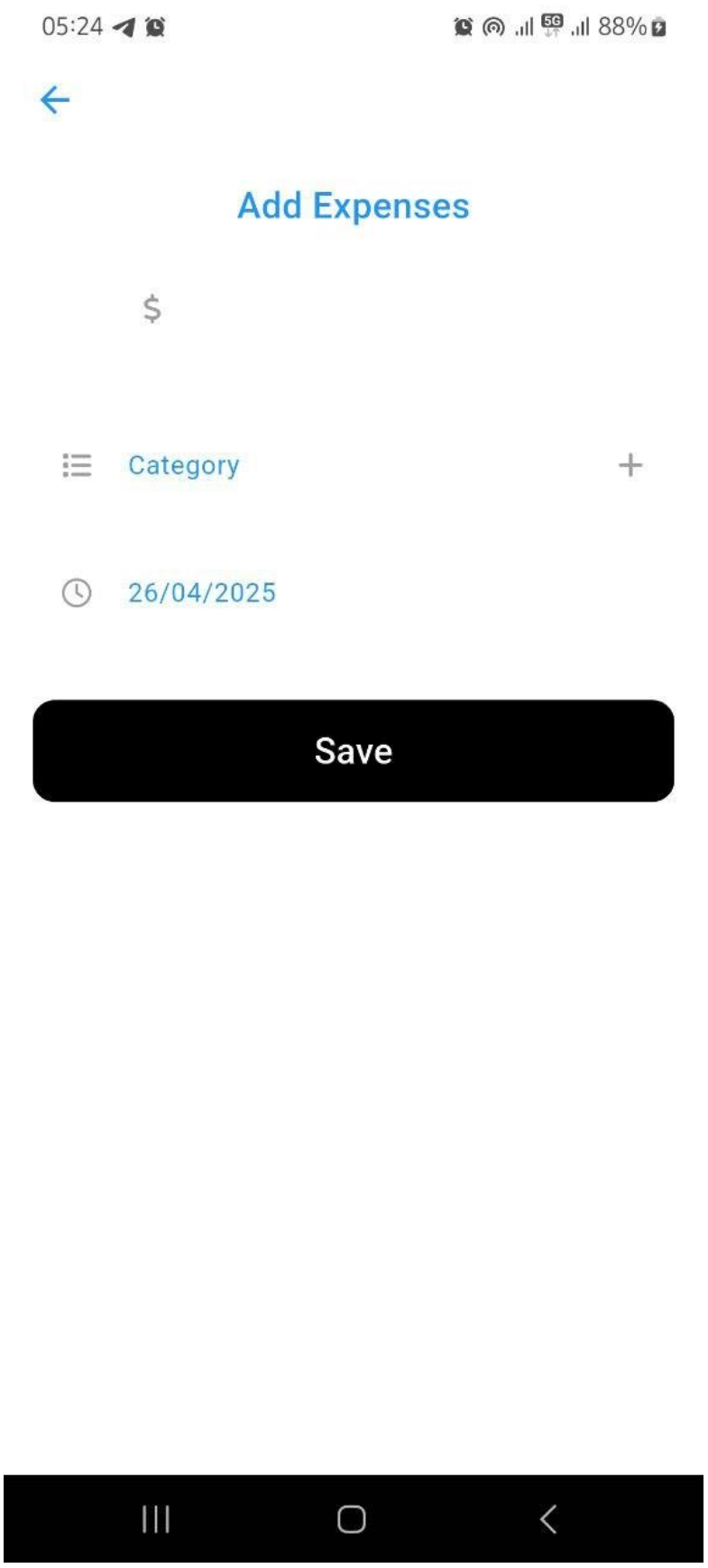
- Add and view income and expenses
- Categorize every transaction
- Dashboard showing total balance
- View past transactions with filters
- Visual reports

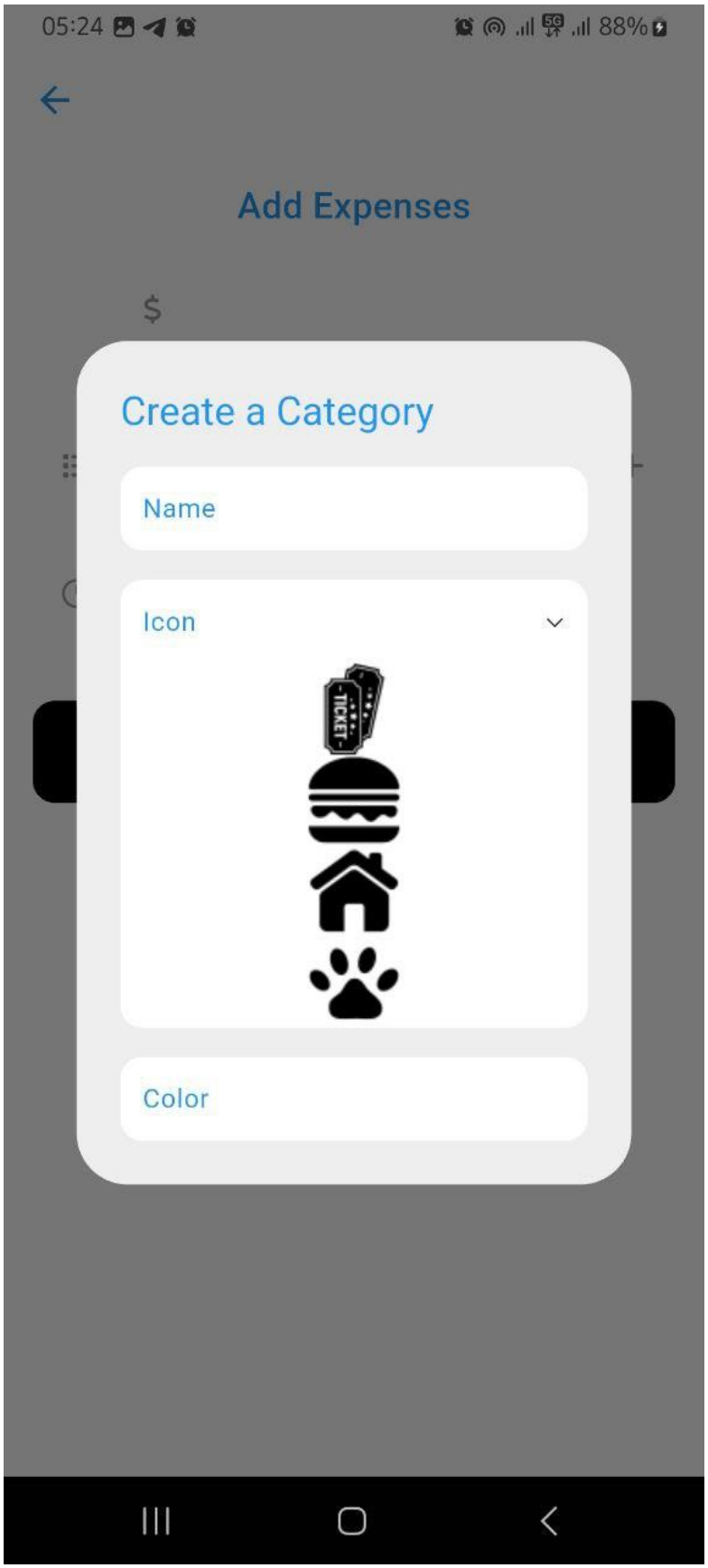
Home/Dashboard screen



Transactions




05:24
88%
Add Expenses
$
Category
26/04/2025
Save


05:24
88%
Add Expenses
$
Create a Category
Name
Icon
TICKET
Color

11. Testing and Quality Assurance

Testing Methods:

- **Manual Testing** on emulators and physical Android devices
- Tested different screen orientations and sizes
- Ensured chart and UI responsiveness
- Verified real-time updates after each transaction

Bug Fixing and Optimizations:

- Fixed layout breakages on smaller screens
- Improved chart rendering speed
- Verified data consistency after app restarts (using SharedPreferences)

12. Future Enhancements

- **PDF Export**: Monthly financial summaries exportable to PDF
- **User Authentication**: Login to sync data across devices
- **Dark Mode**: Optional theme for night-time usage
- **Budget Notifications**: Alert when nearing category budget
- **Currency Converter**: Real-time rate updates for foreign currency users

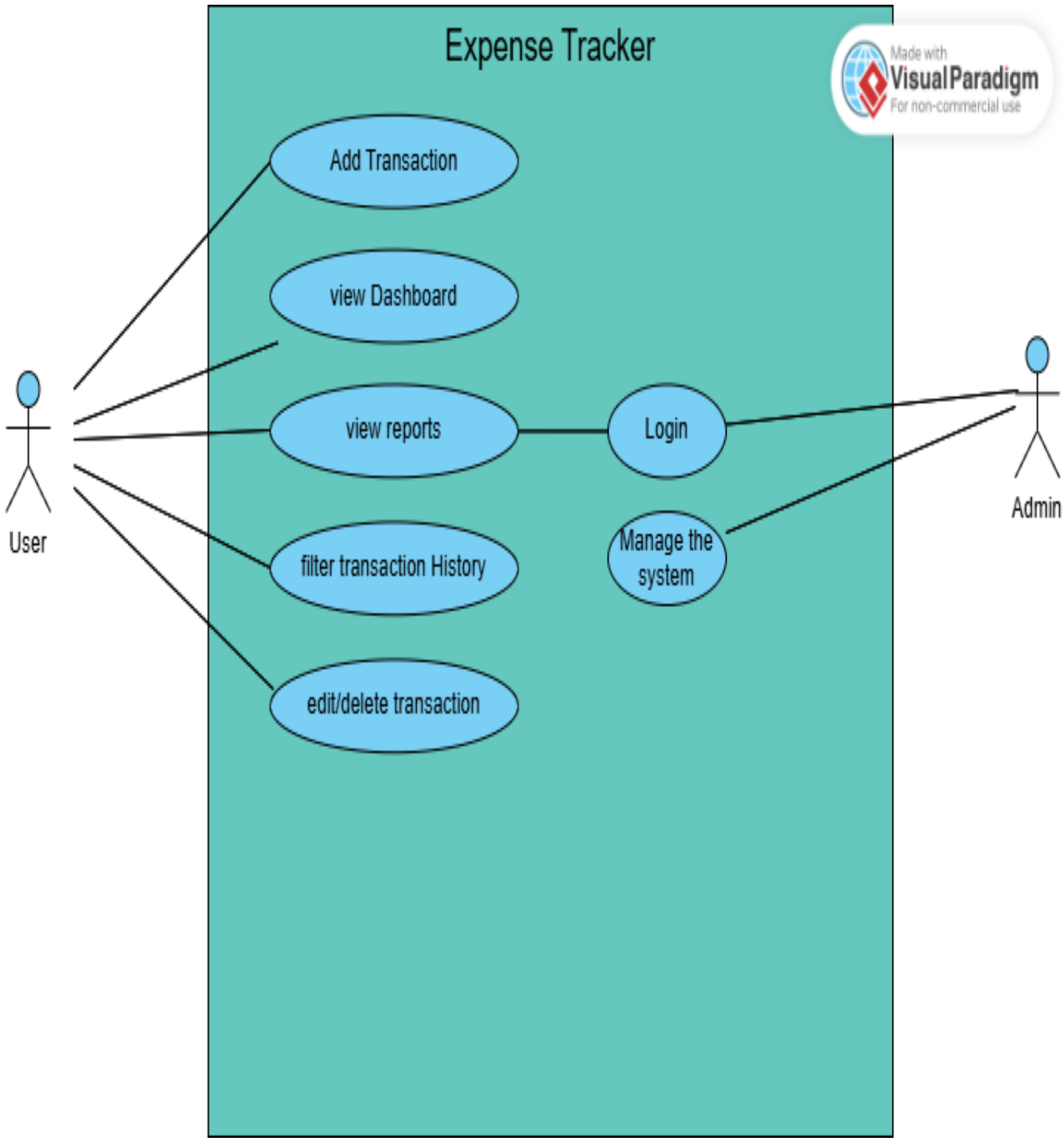
These features align with user requests for increased accessibility, control, and insight into their financial data.

13. Use Case Diagram

Actors: User, Admin

Use Cases: Add Transaction, View Reports, Filter History, Set Budget, Export Summary

Figure 1: Use Case Diagram for Expense Tracker App



14. Competitive Analysis

Comparing our app to 2–3 existing apps (like Wallet, Monefy, etc.)

Feature	Expense Tracker	App Wallet	App Monefy
Add Transactions	☑ Yes	☑ Yes	☑ Yes
Category Filter	☑ Yes	☑ Yes	☑ Yes
PDF Export	❌ Planned	☑ Yes	☑ Yes
Login Required	❌ Planned	☑ Yes	☑ Yes
Local Storage	☑ Yes	☑ Yes	☑ Yes

15. Conclusion

The **Expense Tracker App** project demonstrates a complete mobile application development lifecycle—from ideation and design to implementation, testing, and deployment. It serves as a practical example of how mobile technology can solve real-world problems, specifically in the domain of personal finance.

Through this project, the development team successfully applied mobile development principles using Flutter, integrating cloud technologies like Firebase for real-time data storage and synchronization. The app emphasizes user experience by offering an intuitive interface, smooth navigation, and meaningful visualizations such as pie and bar charts that help users gain quick insights into their financial habits.

Furthermore, the project showcased the advantages of agile methodology. Weekly feedback and iterative updates helped fine-tune user experience and system performance while fostering team

collaboration. The real-time features and offline support also highlight the app’s robustness and scalability for broader usage.

Beyond the technical implementation, this project provided significant learning opportunities in UI/UX design, state management, asynchronous data handling, local vs cloud data synchronization, and testing strategies across multiple screen sizes and devices.

Looking forward, the app offers substantial potential for future enhancements such as budget limit warnings, dark mode, multi-user support, PDF exports, and currency conversion. These would enhance user personalization and expand its relevance to a global audience.

In conclusion, the Expense Tracker App is not just a student project but a scalable, user-centric, and efficient financial tool with real-world potential. It exemplifies what focused teamwork and practical application of mobile development technologies can achieve in a short time frame.