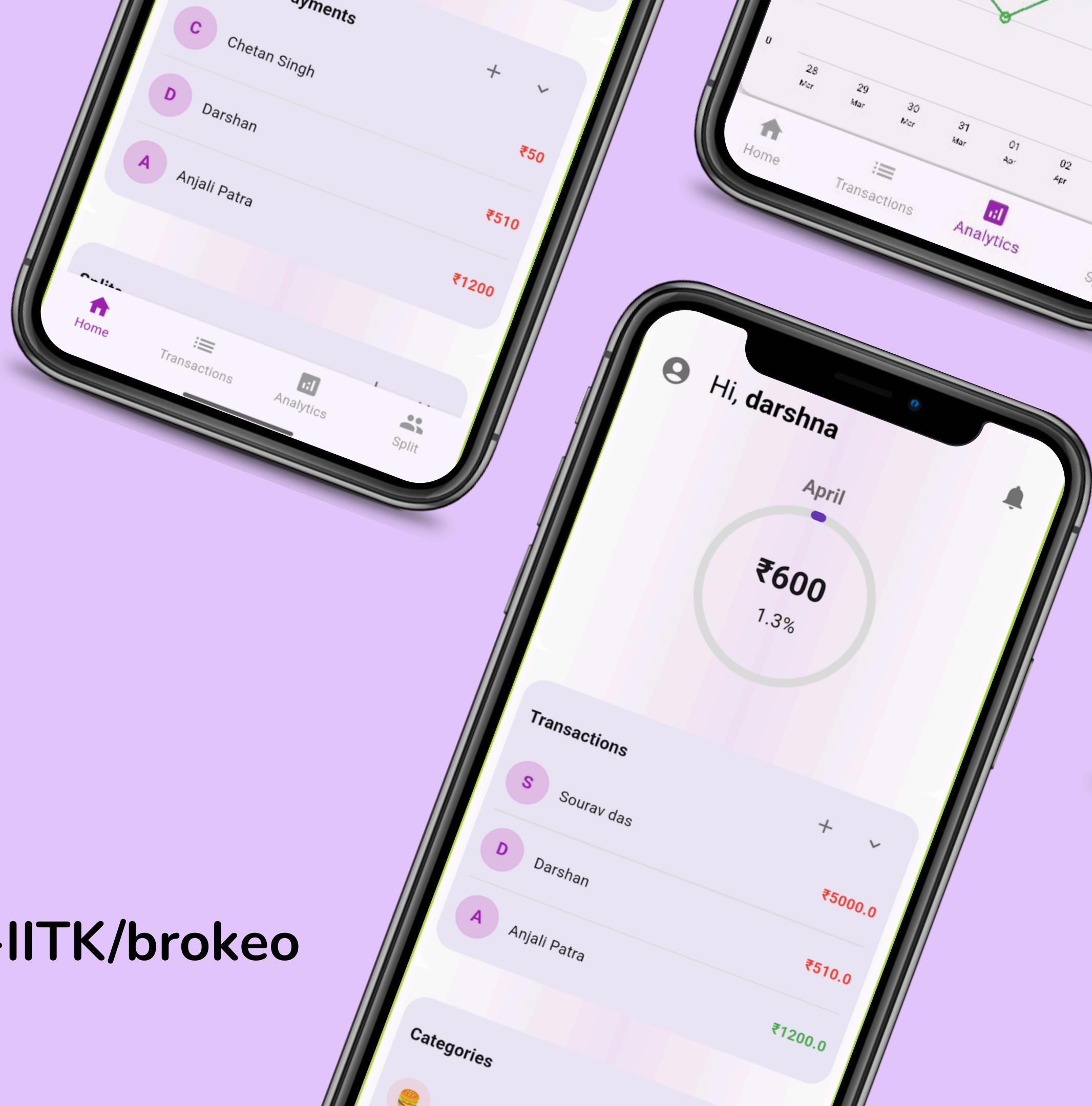


BROKEO

Track your finances on the go

Github Repository :

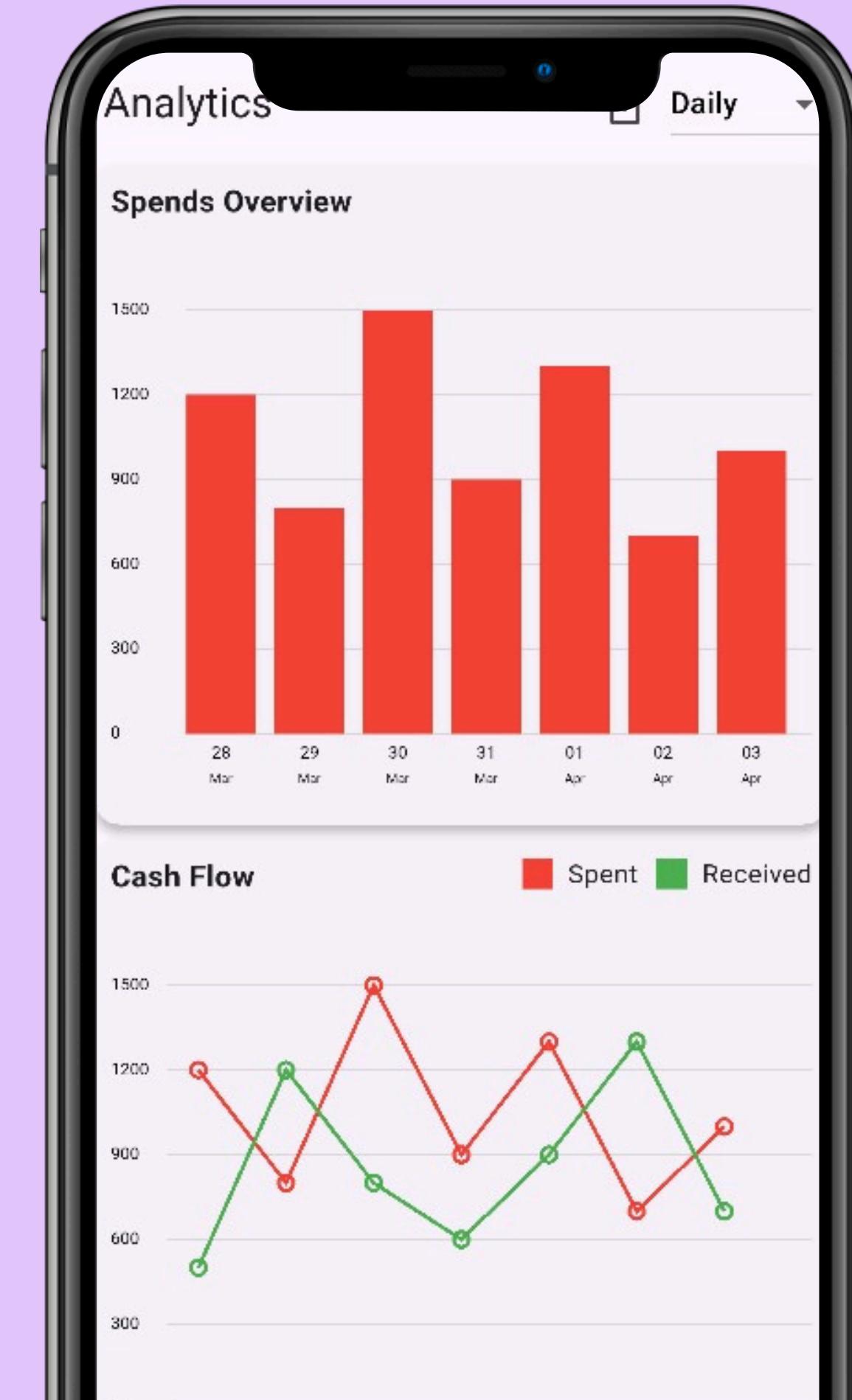
<https://github.com/Runtime-Error-IITK/brokeo>



Introduction

Brokeo is designed to help users develop better spending habits by providing a platform to manage, monitor, and analyze their expenses. The application offers detailed insights into spending patterns, empowering users to make more informed financial decisions. It automatically detects transactions from linked bank accounts via SMS detection and allows users to categorize these transactions or log additional expenses manually.

Shared expenses are monitored using the split feature which tracks contributions and amounts owed by others.

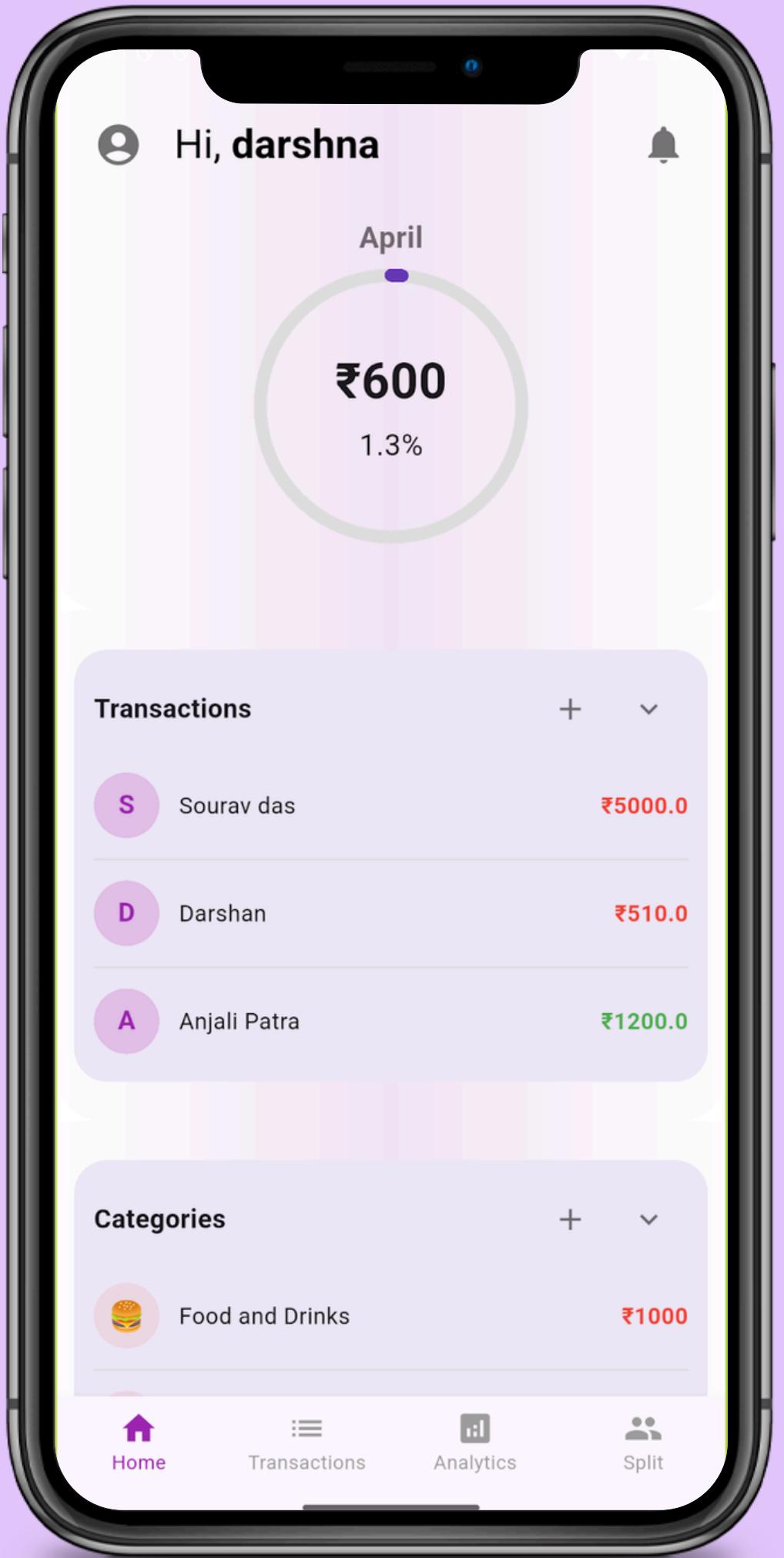




Need for Brokeo

Managing personal finances can often be challenging, leading to difficulties in tracking expenses, maintaining budgets, and saving effectively. Brokeo is a personal finance tracking application designed to address these issues by offering users a straightforward platform to manage their financial activities efficiently. It provides easy-to-use tools to help users make informed financial decisions and improve their money management habits.

Who are the target audience?



End-Users of Brokeo include individuals such as students, professionals, or anyone who is looking for a simple and effective tool to manage their finances.

Requirements

Smart Expense Tracking & Budgeting

Brokeo should feature a reliable system for recording and categorizing expenses across different categories. With SMS detection, the app can automatically log transactions, reducing the need for manual entry. Users should also be able to set budgets and view indicators when nearing their limits.

Insightful Visualizations

To help users better understand their financial behavior, the app should offer visual tools like charts and graphs that present spending patterns clearly and comprehensively.

Requirements

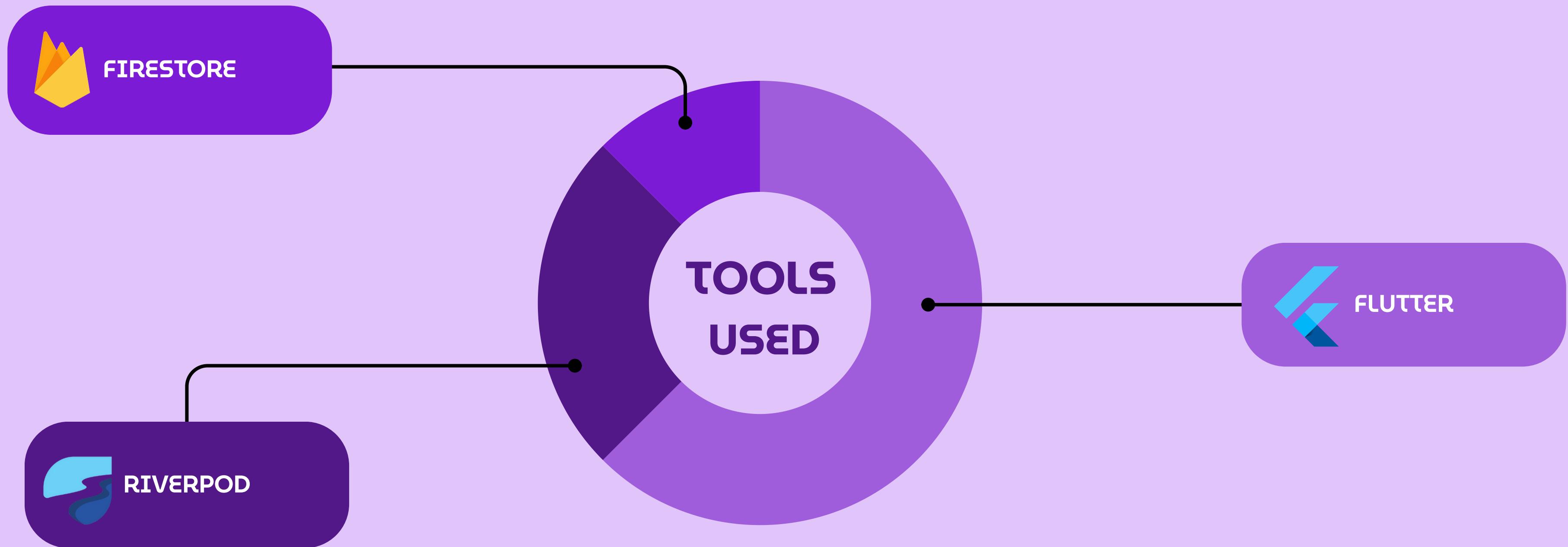
Shared Expenses

Brokeo includes a bill-splitting feature to simplify the management of shared expenses. Whether it's a trip, an event, or a group dinner, users can easily divide costs, keep track of who owes what, and settle up without confusion.

A Holistic Financial Management Solution

By combining expense tracking, budgeting tools, and shared expense features, Brokeo aims to deliver a well-rounded financial management solution. Our team is focused on making the app intuitive and user-friendly, encouraging healthy money habits and helping users meet their financial goals.

Key Design and Implementation Decisions



Why Flutter?

UI Consistency

Maintain uniform UI across devices without tweaks.



Hot Reload

Instantly view changes during development.

Native Compilation

Achieve high performance with direct ARM code compilation.

Single Codebase

Develop for both Android and iOS with one set of code.

Customizable Widgets

Create responsive UIs with pre-designed widgets.

Why Firestore?

Reduced Complexity

Eliminates the need
for separate
databases and
backends

Firebase Integration

Simplifies integration with
other Firebase services



Real-time Synchronization

Ensures data
consistency
across devices

Local Cache

Allows offline
functionality

Automatic Scaling

Handles performance
optimization without manual
setup

Why Riverpod?

Compile-Time Safety

Ensures early error detection through robust type checking.

Modular Code

Facilitates testable code by separating logic and UI.

Fine-Grained Listening

Optimizes performance by precisely monitoring data changes.

Provider Flexibility

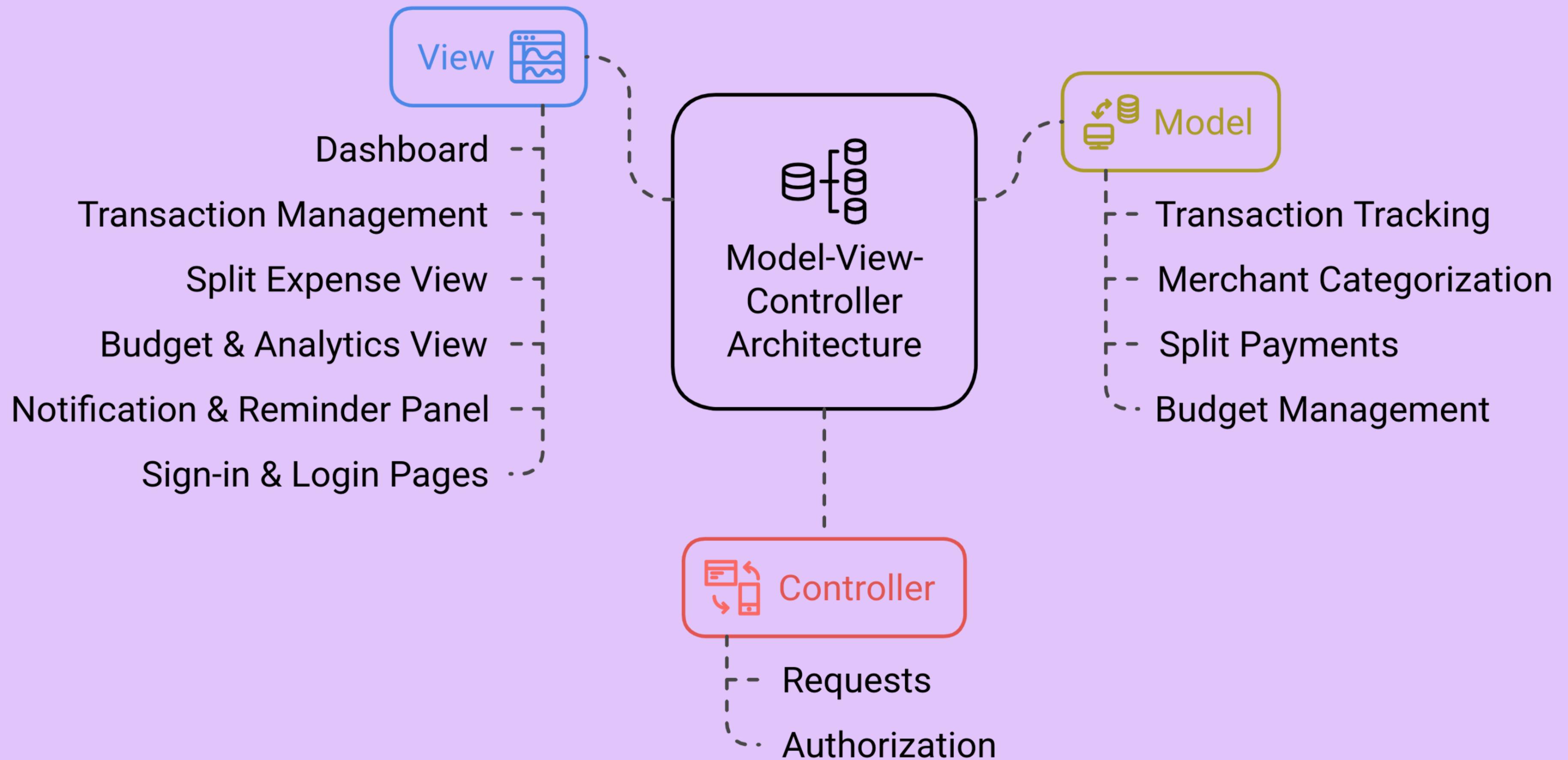
Supports both synchronous and asynchronous data providers.

Improved Developer Experience

Maintains familiarity while avoiding past solution pitfalls.



Architectural Pattern



DEMO



FUTURE PLANS

Future Expansion:

Looking ahead, we're excited to expand the app across platforms—including iOS, a web app, and a full-featured website—making it more accessible and useful for a wider audience.

Feature Enhancements:

Add AI-based spending insights, smarter SMS detection, OCR receipt scanning, and tracking for savings, investments, and debts. Add notifications for transactions, reminders, and financial tips.

UI & UX Improvements:

Introduce dark mode, custom themes, more interactive dashboard, and voice command support.

Performance & Backend:

Strengthen data security with encryption and authentication, and ensure smooth handling of transactions and analytics through a reliable and scalable backend.

Integration & Expansion:

Link bank accounts and integrate payment methods for real-time updates, support more data export, and add a community for sharing financial tips.

LESSONS LEARNED

Beyond Coding

We learned that good software isn't just about clean code—it's about understanding user needs, collaborating effectively, and embracing feedback.

Good Development Practices

We realized the importance of writing modular code, maintaining proper documentation, testing thoroughly, and using community support.

Practical Application

This project pushed us to apply theoretical concepts in a real-world setting, giving us hands-on experience with software development and finance

Financial Understanding

Developing features like transaction tracking, payment splitting, and spending analytics helped us understand digital finance workflows and user trust

Ethical Responsibility

Handling financial data made us more aware of the need for accuracy, security, and transparency in user-facing financial apps.

THANK YOU!

