PESAHUB – SYSTEM OPERATION DOCUMENTATION

1. Overview of How the Application Works

BudgetingApp is designed to help users manage their personal finances by tracking income, expenses, budgets, and savings goals. The system follows the MVC (Model-View-Controller) architecture to ensure smooth data flow, proper organization, and maintainability. This section explains step-by-step how the application functions from user interaction to data storage and reporting.

2. User Registration and Authentication Process

When a user opens the app for the first time, they are required to create an account. The user enters their email and password, which are validated before being securely stored. Passwords are encrypted to ensure security.

During login, the system verifies the entered credentials against stored records. If authentication is successful, the user is redirected to the dashboard. If authentication fails, an error message is displayed.

Authentication is required before accessing any financial data to ensure privacy and security.

3. Dashboard Functionality

After login, the user is directed to the dashboard. The dashboard provides a financial summary, including:

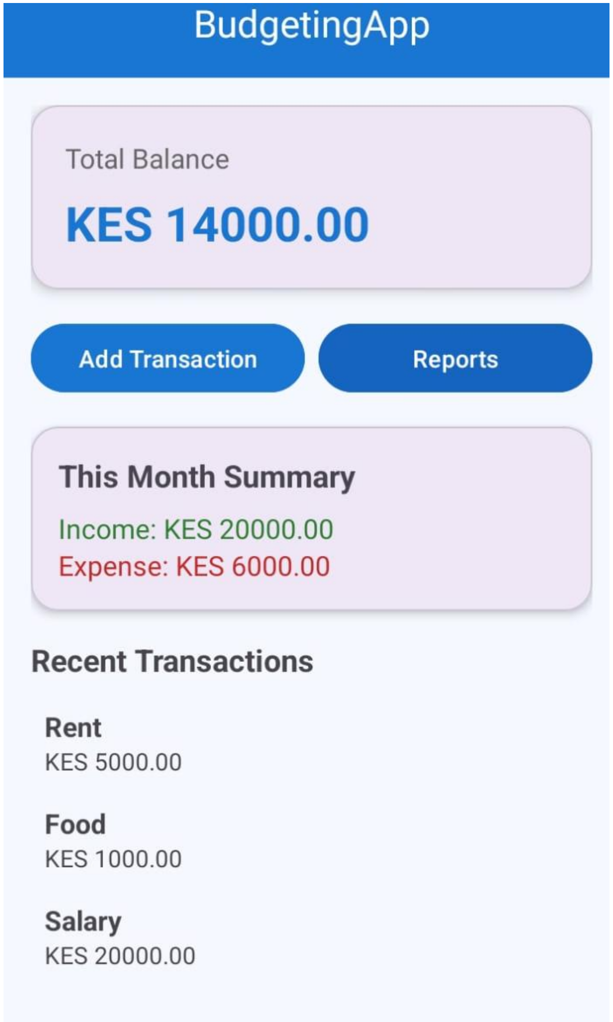
- Total income for the selected period.
- Total Budget for the selected period.
- Remaining balance (Income – Budget).
- Quick access buttons for adding income or expenses.

The dashboard automatically updates whenever new financial records are added, edited, or deleted.

4. Income Management Process

When the user selects 'Add Income', a form appears requesting the income amount, source, and date. After submission, the system validates the input and stores the data in the local database.

The ViewModel processes the data and updates the dashboard in real time. Users can also edit or delete income entries, which triggers automatic recalculation of totals.



Total Balance

KES 14000.00

This Month Summary

Income: KES 20000.00

Expense: KES 6000.00

Recent Transactions

Rent
KES 5000.00

Food
KES 1000.00

Salary
KES 20000.00

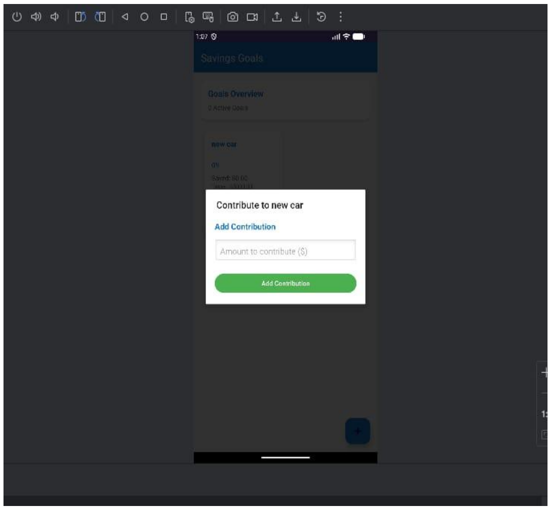
5. Expense Management Process

When the user selects 'Add Expense', a form appears where the user enters the amount, category, description, and date.

The system assigns the expense to a selected category. If the category has a budget limit, the system checks whether the new expense exceeds the allocated budget.

If the expense approaches or exceeds the budget limit, a notification alert is triggered.

All expense entries are saved in the database and reflected immediately in reports and summaries.



1:07

Savings Goals

Goals Overview

0 Active Goals

new car

0%

Saved: $0.00

Contribute to new car

Add Contribution

Amount to contribute ($)

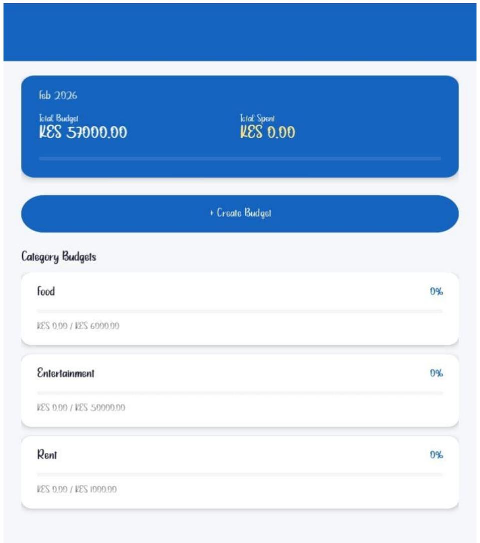
Add Contribution

6. Budget Calculation and Monitoring

Users can create category-based budgets (weekly or monthly). The system continuously monitors total expenses within each category.

Remaining Budget = Set Budget – Total Category Expenses

If the remaining budget reaches a defined threshold (for example 80%), the system sends a warning notification. If the limit is exceeded, a critical alert is displayed.



feb 2026

Total Budget
KES 57000.00

Total Spent
KES 0.00

+ Create Budget

Category Budgets

Food 0%
KES 0.00 / KES 6000.00

Entertainment 0%
KES 0.00 / KES 50000.00

Rent 0%
KES 0.00 / KES 1000.00

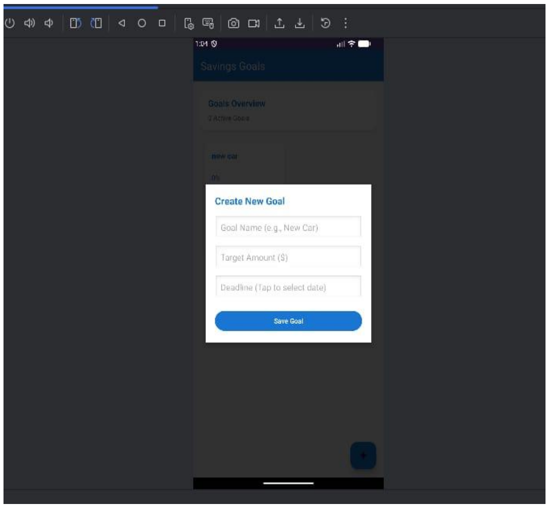
7. Savings Goals Tracking

Users can create savings goals by entering a goal name, target amount, and deadline.

The system calculates progress using the formula:

Goal Progress (%) = (Saved Amount / Target Amount) × 100

Progress is displayed visually using a progress bar. Users can update or delete goals at any time.

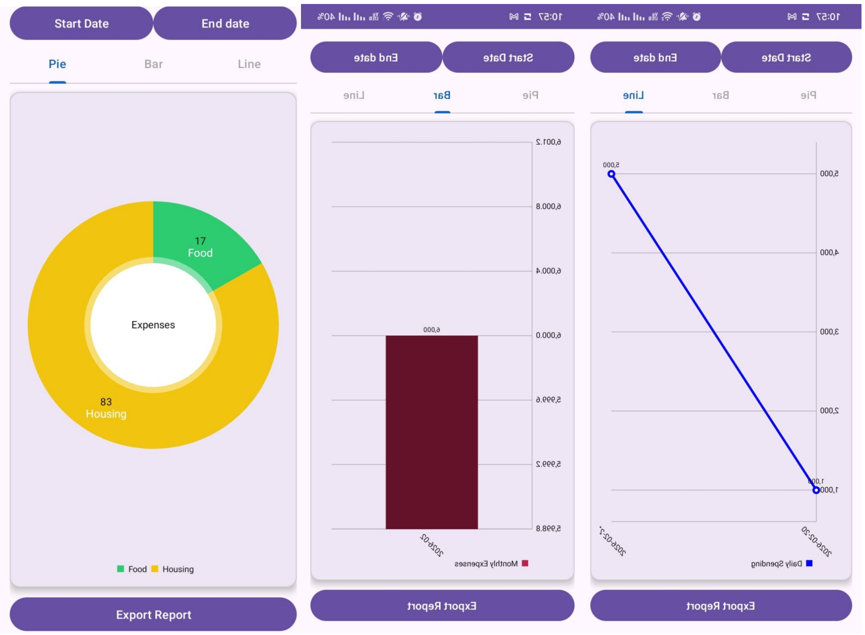


8. Reports and Data Visualization

The application generates financial reports automatically based on stored income and expense data.

- Pie charts for category distribution.
- Bar charts for income vs expenses comparison
- Line graphs for financial trends over time.

Users can filter reports by selecting a date range (daily, weekly, monthly, or custom range).



11. Summary of System Workflow

In summary, BudgetingApp operates through structured data input, secure storage, real-time calculations, and automatic updates to the dashboard and reports. The

MVCarchitecture ensures separation of concerns between the user interface, business logic, and database operations, allowing the application to function efficiently and reliably.