Personal Expense Tracker Documentation – Bibin Thomas

OVERVIEW

The Personal Expense Tracker application is designed to help users manage their expenses efficiently. The application allows users to record, categorize, and analyze their spending habits. It includes features for user management, expense tracking, category management, and provides insightful summaries.

SYSTEM ARCHITECTURE

1. Architecture Overview

The system is divided into three main components:

Front-End: Developed using **React.js**, responsible for the user interface and user interactions.

Back-End: Implemented with **Node.js and Express.js**, for interacting with the database.

Database: MongoDB is used to store expense information.

2. Data Flow

Expense Management: Users add, view, edit, or delete expenses via the front-end. The front-end sends requests to the back-end, which processes these requests and updates the database accordingly.

Category Management: Users manage expense categories through the front-end. The front-end interacts with the back-end to create, update, or delete categories in the database.

Summary and Insights: Users view summaries and insights through the front-end. The front-end requests summary data from the back-end, which processes the data and returns it for display.

3. Core Features

1. Expense Management

Add Expense: Users can add expenses with details such as date, amount, category, and description.

View Expenses: Users can view a list of their expenses.

Edit/Delete Expense: Users can modify or remove existing expenses.

2. Category Management

Default Categories: The application includes predefined categories.

Custom Categories: Users can create and manage their own categories.

3. Summary and Insights

Spending Summary: Provides total spending for a given time period (daily, weekly, monthly).

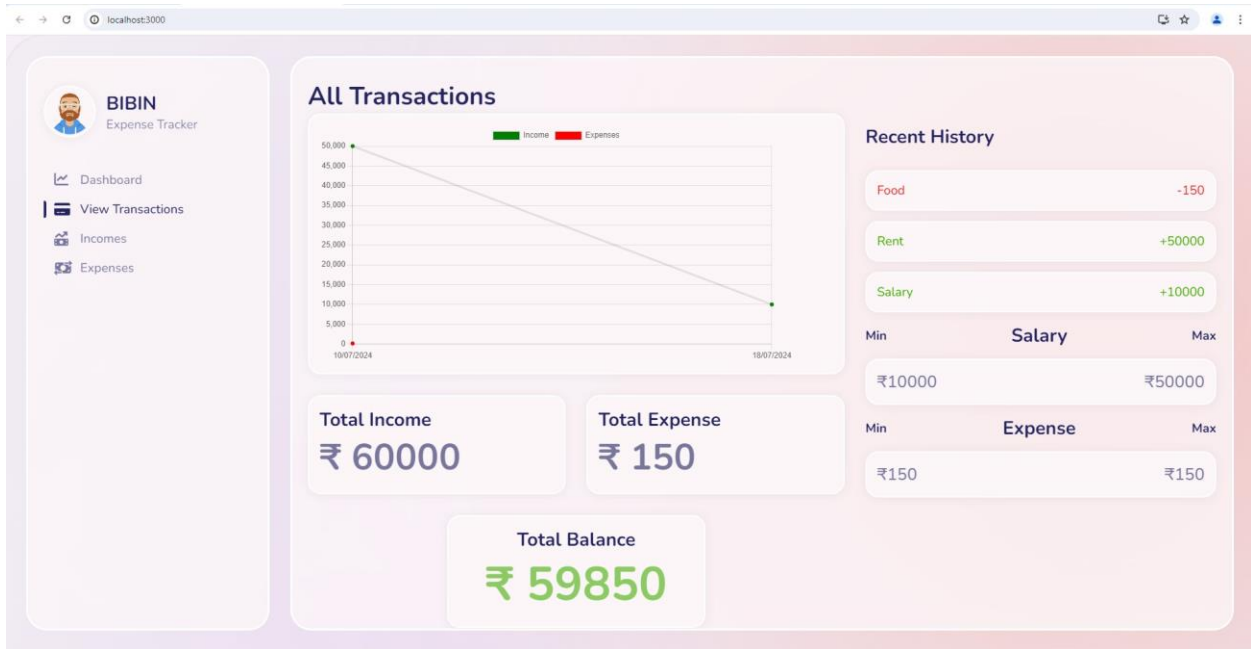Spending by Category: Displays spending breakdown by category.

Advanced Features

1. Graphs and Charts

Visualization: Spending data is visualized using pie charts for category spending and line charts for spending over time.

SCREENSHOTS

1.

2.