Personal Budget Tracker

- Personal Budget Tracker
- Team Member: Thong Dang
- Presentation Record: Meeting in _General_-20250504_222038-Meeting Recording.mp4
- Demo Record: Meeting in _General_-20250504_230127-Meeting Recording.mp4

Problem Statement

MANY PEOPLE STRUGGLE WITH PERSONAL FINANCE TRACKING.

MANUAL METHODS (SPREADSHEETS/PAPER) ARE INEFFICIENT.

NO CLEAR INSIGHTS INTO BUDGETS, RECURRING PAYMENTS, OR CATEGORY-LEVEL TRACKING.

Motivation

Improve personal financial control.

Build real-world full-stack development skills.

Create a customizable alternative to Mint or YNAB.

Solution Overview

Web app to manage income, expenses, and budgets.

Supports recurring transactions and auto-categorization.

Summarizes financial health through dashboards.

How It Helps

Reduces budgeting friction and manual entry.

Tracks all accounts and spending categories.

Offers insight through summaries and visualizations.

Biggest Challenges

Securely integrating JWT-based authentication.

Handling environment variables during Render deployment.

Designing a clean, functional UI across components.

Implementing file import/export features reliably.

Managing and toggling dark/light mode consistently.

Setting up notification and reporting via email or phone.

Technical Challenges

Backend and frontend CORS/config coordination.

Database schema for scheduled transactions.

Balancing minimalism with feature completeness.

Biggest Highlights

Dynamic dashboards with real-time category totals.

Support for recurring (scheduled) transactions.

Custom user preferences (currency, dark mode).

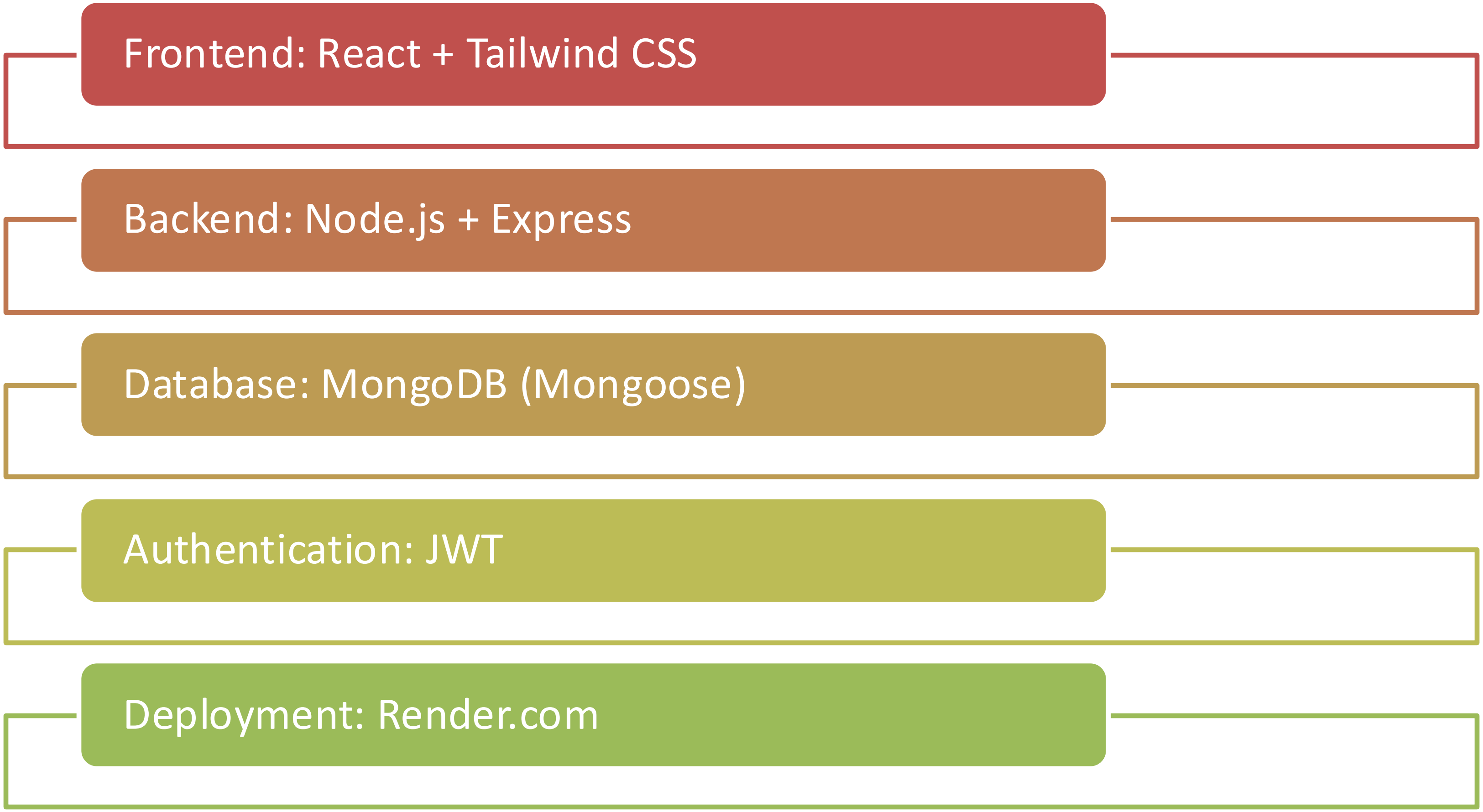
What We're Proud Of

Simple but robust design.

Deployment-ready full-stack system.

Smooth user experience with clear navigation.

Technology Stack



- Frontend: React + Tailwind CSS
- Backend: Node.js + Express
- Database: MongoDB (Mongoose)
- Authentication: JWT
- Deployment: Render.com

System Architecture

Client (React) communicates with REST API.

Express backend exposes endpoints.

MongoDB stores transactions, users, budgets.

JWT protects secure routes.

Application Structure

Models: Account, Budget, Transaction, Preferences

Routes: User auth, Transactions, Budgets

Frontend Pages: Dashboard, Settings, Accounts, Budgets

UI Screenshots (Placeholders)

→ Transactions Dashboard

→ Account Balance Overview

→ Settings with Preferences

→ Budget Progress Bars

Future Development

- PDF/CSV export for transactions.
- Spending predictions using ML.
- Mobile responsiveness or native app.
- Monthly spending analysis and alerts.

Project Repos & Contact

Repository: https://github.com/ThongDang251999/Final-project-.git

Live Demo: https://final-project-1-22hm.onrender.com

Presented by: Thong Dang