SpendSense

Your AI-Powered Financial Coach

Dongjin Li

Summary of contents

01

Problem and Solution

02

Technical Architecture

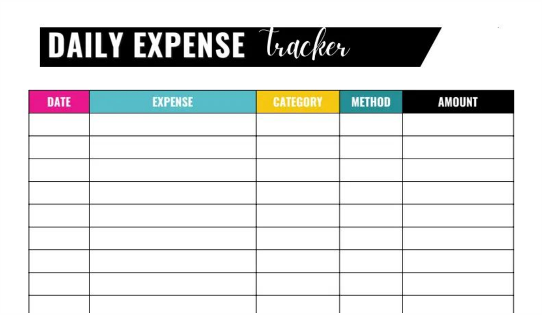
03

Live Demo

04

Learning and future

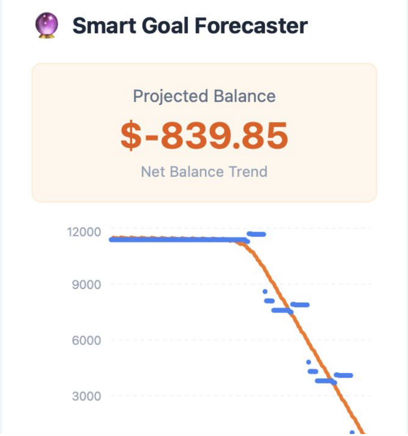
Problem and Solution



DAILY EXPENSE tracker

DATE	EXPENSE	CATEGORY	METHOD	AMOUNT

- Tracking every expense manually is tedious and unscalable.
- Only offer static charts that fail to inspire lasting behavioral change.
- Users lack visibility into their financial future, leaving them unaware of wasteful habits until it’s too late.



Smart Goal Forecaster

Projected Balance

$-839.85

Net Balance Trend

- Replaces manual entry with seamless aggregation through plaid API
- Shifts focus from "what happened" to "what will happen" using **Prophet ML model**.
- Transforms raw data into personalized behavioral nudges using **Generative AI**.

Architecture and Tech Stack

A RESTful Client-Server Architecture

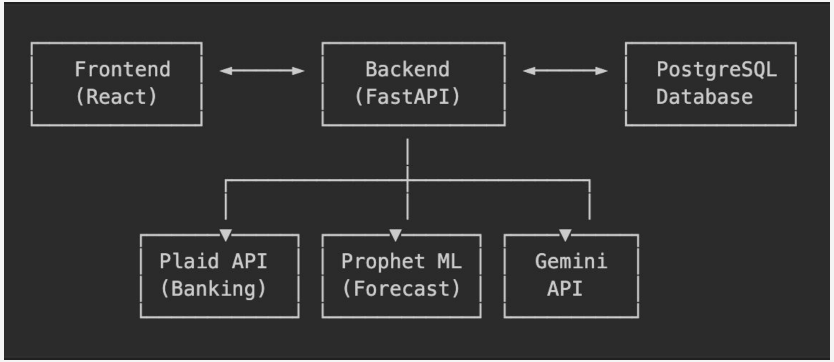
Frontend: React (Vite) + **Tailwind CSS** for responsive UI; **Recharts** for data visualization.

Backend: FastAPI (Python) for high-performance async orchestration.

Machine Learning: Facebook Prophet for time-series forecasting.

AI Agent: Google Gemini 2.5 Flash for generative insights.

Integrations: Plaid API for secure banking data aggregation.



Frontend (React)

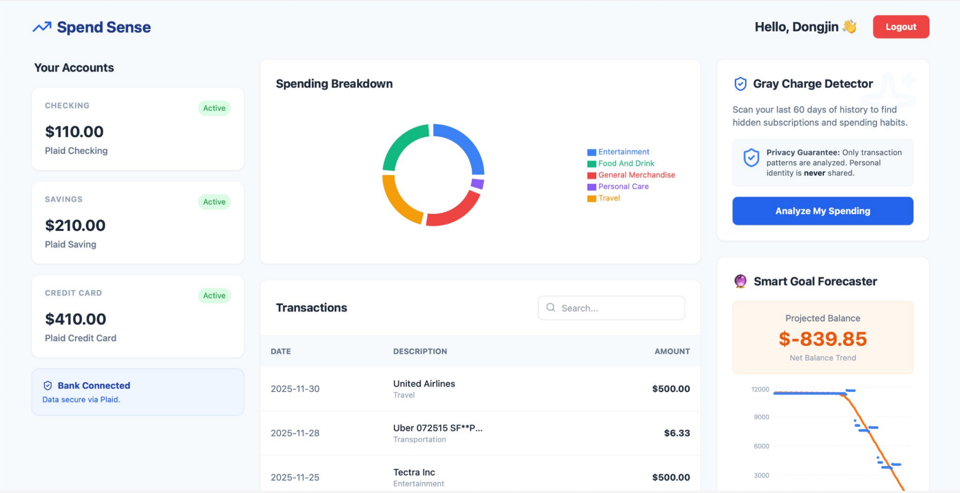
Backend (FastAPI)

PostgreSQL Database

Plaid API (Banking)

Prophet ML (Forecast)

Gemini API



Your Accounts

CHECKING Active
$110.00
Plaid Checking

SAVINGS Active
$210.00
Plaid Saving

CREDIT CARD Active
$410.00
Plaid Credit Card

Bank Connected
Data secure via Plaid.

Spending Breakdown

Transactions

DATE	DESCRIPTION	AMOUNT
2025-11-30	United Airlines Travel	$500.00
2025-11-28	Uber 072515 SF**P... Transportation	$6.33
2025-11-25	Tectra Inc Entertainment	$500.00

Gray Charge Detector

Scan your last 60 days of history to find hidden subscriptions and spending habits.

Privacy Guarantee: Only transaction patterns are analyzed. Personal identity is **never** shared.

Smart Goal Forecaster

Projected Balance
$-839.85
Net Balance Trend

Demo

Challenges and key learnings

Challenges:

- Plaid only provides current balances—reconstruct history balance by reversing transactions
- Linear Regression inadequate for account balance patterns—changed to Prophet
- Sequential API calls (Plaid → Prophet → Gemini) = high response time, identified need for async processing and caching for production

Key learning:

- Prioritizing scope, managing time constraints, and executing under pressure are critical
- Gained deeper understanding of trust, security, and privacy—sensitive data requires encryption, minimization, and thoughtful architecture

Thank you