Wallee.
Project Plan

By: Emma Bahr, Matteo Caruso, Kyle Gibson, and Joshua Cajuste

Information

Faculty Advisor: Dr. Siddhartha Bhattacharyya sbhattacharyya@fit.edu

Client:

- Doug Gibson — Small Business Owner
- Dr. Siddhartha Bhattacharyya — SWE/CSE Professor & Advisor

Goals and Motivation

Our goal is to create an app to assist with financial goals to help save and manage money. Managing personal finances is a growing challenge for students, freelancers and young professionals who face variable income, recurring bills, and limited financial literacy. Existing budgeting tools often lack personalization, real-time updates, or intelligent guidance, leaving users without actionable insights.

Wallee aims to fill this gap by providing a smart, all-in-one finance app that integrates with user's bank accounts, automates expense tracking, delivers bi-weekly reports and includes an AI-powered financial assistant. Our motivation is to empower users to take control of their finances through simplicity, automation, and tailored insights.

Features

1. Track spending in real time: Securely connect your bank accounts to automatically track and categorize your income and expenses.
2. Receive personalized financial summaries: Easily compare your planned budget to your actual spending with visual graphs and detailed summaries.
3. Proactive financial assistance: Access educational content directly from the app to help you build your financial literacy as well as access to an AI chatbot that provides simple explanations of your financial data, offers personalized tips, and sends proactive reminders for bills or tax payments.

1. Tracking spending

- Users can connect their bank accounts security to track income and expenses in real time
- User will be able to view and adjust automatically categorized transactions or create custom budget categories
- Users can compare planned budgets with actual spending across categories.
- Users can track progress over time, with adjustments suggested if overspending occurs.
- Users can set short-term and long-term financial goals, such as saving for tuition, vacations
- Users can receive automated budget updates after each paycheck, showing exactly how much money is “safe to spend” after recurring bills are covered.

Use Case Scenario

A student sets a $12,000 savings goal for a car purchase with a 12-month deadline. If they overspend on dining by $100 during the first month, Wallee can recalculate the remaining budget by:

- Adjusting the goal deadline, increasing the time to reach the goal based on overspending.
- Adjusting the budget for the remaining time to account for the overspending (example: decreasing spending each month by 100$ / 11).

2. Personalized Financial Summaries

- Users can access bi-weekly reports (or use a custom pay-period) showing income, bills, discretionary funds, and savings progress.
- Users can view weekly and monthly summaries broken into tables with budgeted, spent, and remaining amounts.
- Users can compare budgeted versus actual spending in visual graphs that highlight trends and overspending.
- Users can view predictive insights, such as “safe-to-spend” numbers or upcoming bill reminders.

3. Proactive Financial Assistance

- Users can interact with an AI chatbot that explains financial data in simple terms.
- Users can receive personalized tips, such as saving more in high-income months or adjusting categories to meet goals.
- Users can access educational content designed to improve financial literacy.
- User can view graphs and tables to compare the set budget to their actual spending

Use Case Scenario

For example, a freelancer with irregular income could receive AI-driven alerts recommending when to save more after a high-earning month, or reminders about upcoming tax payments.

Novel Features

- Automated bi-weekly budgeting for recalculating finances with each paycheck
- AI-driven guidance that provides personalized financial insights
- Tailored user experiences for students, freelancers, retirees, and couples
- Intuitive visualization tools like weekly tables and graphs showing budget limits vs. actual spending
- Competing apps (e.g., Mint, PocketGuard) present raw data without context, while Wallee focuses on clarity and actionability
- Prioritizes proactive automation and user-focused design, making it more inclusive, supportive, and actionable than existing budgeting apps

Tools :

Backend runtime for APIs & webhooks:

- Node.js (TS) for public API/webhooks, Python (FastAPI) for analytics and budgeting workers. Best of both worlds.

Budgeting/analytics engine:

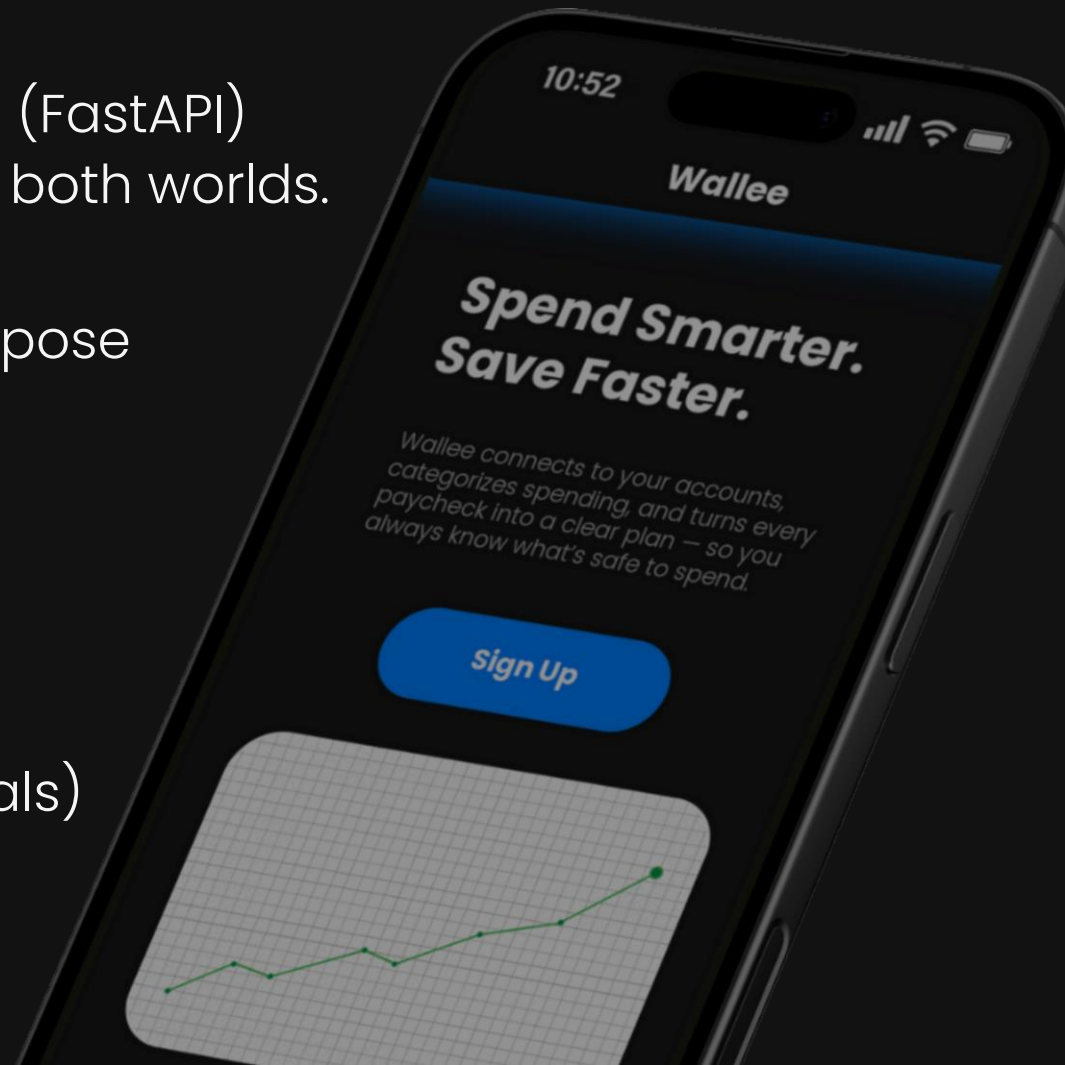
- Python worker for budgeting/forecasting; expose results via API.

Banking integration:

- Plaid for US launch; design adapters to add TrueLayer for EU later.

Database & data modeling:

- PostgreSQL (core ledger, users, budgets, goals)



Tools :

Job orchestration & scheduling:

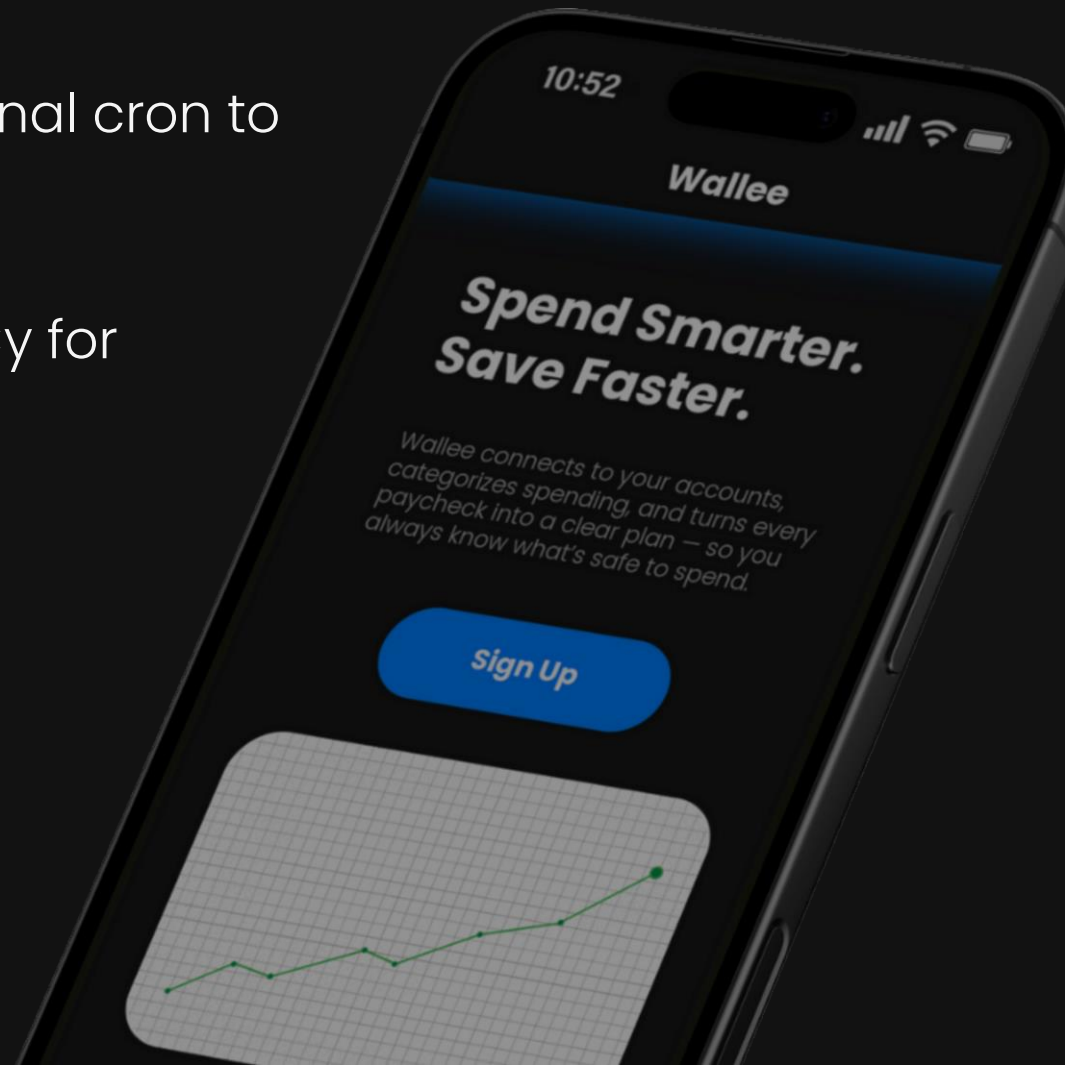
- Celery + Redis/SQS for analytics tasks; optional cron to trigger batch runs.

AI assistant & NLU:

- OpenAI for conversational guidance + spaCy for lightweight extraction

API framework & validation:

- NestJS for API; FastAPI for analytics service.



Challenges

- We plan to create an adaptive budgeting engine that will be the bulk of the application using Node.js and python but are not very familiarized with it
- We plan to develop an ethical and reliable AI Assistant using Google AI studio API but we are not very familiar with using API's
- We plan to develop a mobile app, but we are not overly familiar with mobile app development
- We plan to use Flutter for app development, but we are not overly familiar with flutter.
- We plan to add security measures to the application but are not familiar with implementing security precautions to applications

Task Matrix

Task	Emma	Matteo	Kyle	Joshua
Compare and select Technical Tools	20%	40%	20%	20%
Mobile Demos	20%	20%	20%	40%
Resolve Technical Challenges	20%	20%	40%	20%
Compare and select Collaboration Tools	40%	20%	20%	20%
Requirement Document	25%	25%	25%	25%
Design Document	25%	25%	25%	25%
Test Plan	25%	25%	25%	25%

Milestone 1 (Sept. 29th)

- Compare and select technical tools for creating the financial recommendation algorithm, designing the database, linking users bank information with the application, and security of user's information.
- Provide small mobile demos to evaluate the tools for the algorithm, the database, linking user account information, and the security of users information.
- Resolve technical challenges for the algorithm, the database, linking user account information, and the security of user's information.
- Compare and select collaboration tools for software development, documents/presentations, communication, task calendar.
- Create Requirement Document.
- Create Design Document.
- Create Test Plan.

Milestone 2 (Oct. 27th)

- Create data-base for each user's bank and financial information.
- Begin more evolved design for app UI and navigation flow.
- Implement, test, and demo initial account integration and transaction categorization.
- Implement, test, and demo weekly/bi-weekly budget summaries.
- Implement, test, and demo savings goal creation and tracking.
- Implement, test, and demo initial AI chatbot functionality with sample responses.
- Collect user feedback from early prototype testing.

Milestone 3 (Nov. 24th)

- Ensure integrity and security of the users' bank and financial information.
- Design and implement advanced menus (reports, settings, customization).
- Implement, test, and demo customizable budget categories with notifications.
- Implement, test, and demo AI-driven financial insights and proactive alerts.
- Implement, test, and demo visualizations (tables and graphs comparing budget vs. actual).
- Improve paycheck-based recalculations and overspending adjustments.
- Improve AI chatbot responses with tailored financial education modules.
- Refine security, encryption, and overall menu design for better usability.