Final Project Proposal

FinAntes: Your Finances, One Step Ahead

4/06/2025

Prepared for:
ComS / SE 319 - Construction of User Interfaces
Iowa State University of Science and Technology

Prepared by:
Jayson Acosta
Zaid Rachman

Introduction

We, Jayson Acosta(Jayson04@iastate.edu) and Zaid Rachman(ZaidR@iastate.edu) are from team IP_4. We are engineers with experience in many projects, such as Embedded Systems, Android Development, Web Development, and many more. Our Web Development background includes building websites using React, Typescript, and Traditional Web Technologies (HTML, CSS, JavaScript, PHP). We have developed web applications with frontend and backend technologies, including Vue.js, Vite, Express.js, and many more. Working together, we can create an excellent web application.

Purpose of the Proposal

Keeping track of finances in today's world can be tricky. People often need to use multiple applications to manage their finances. This can make organizing your finances inconvenient and time-consuming. Our solution to this problem is FinAntes. FinAntes aims to create a centralized platform where users can manage all aspects of their finances. This final project will expand our midterm implementation by moving from a client-side application to a full-stack solution with React, Node.js, and a database backend.

Goals & Objectives

Our primary goal remains to make financial management simpler while adding significant technological improvements:

- Reimplementing the entire application using React for frontend and Node.js with Express for backend
- Creating a database-driven architecture to replace local storage as the primary data source
- Implementing secure user authentication and authorization using JWT
- Developing a hybrid synchronization system that allows offline functionality with cloud backup
- Maintaining our original user-friendly interface with enhanced responsiveness
- Implementing all planned service tiers (Demo, Basic, Premium) with proper access controls
- Enabling cross-device access to financial data
- Delivering advanced analytics for financial insights (especially for Premium users)
- Implementing real-time market data integration (for Premium users)

Project Description

FinAntes will transform from our midterm's client-side application into a full-stack web application with enhanced features:

Core Features:

- **User Authentication System:** Secure login/signup with JWT authentication
- **Interactive Dashboard:** Budget, spending, investment, and goal widgets with enhanced functionality
- **Budget Management System:** Customizable categories with database persistence
- **Data Synchronization:** Ability to work offline with local storage and sync to the cloud when online
- **Theme Customization:** Enhanced with React component styling
- **Financial Analytics:** Advanced data visualization with server-side processing
- **Cross-device Access:** Access financial data from any device through account login

Service Tiers:

The main product of FinAntes is the service tiers. There are three tiers, each having its benefits. The three tiers are:

1. Free Demo
    - View-only access to the dashboard
    - Hardcoded data
    - Example reports and visualizations
    - Basic feature demonstrations
2. Basic User (Free)
    - Full budget planning functionality
    - Manual data entry
    - CSV import/export
    - JSON import/export
    - Basic analytics
    - Locally stored data
3. Premium User
    - Real-time market data
    - Custom theme options
    - Advanced analytics
    - API integration

The “Free Demo” tier is for users who have not accepted the terms and conditions of our application. These users will be able to see a Demo with hardcoded data. If they accept the conditions and create an account, they can access the “Basic User” tier. Premium users are basic users who have sent money to support the application.

Project Path Selection

We will be extending our midterm project. In our extended version of the project, we plan on implementing:

- Login Authentication
    - Users will be able to log in and out of their accounts regardless of what device they are using.
- Profile tiers
    - Users will be assigned roles (free, standard, premium) depending on which tier they signed up for.
    - Users can access different functionalities depending on the tier they signed up for.
- Storing/retrieving user data into a database
    - User information will be saved within a database and be retrieved and displayed upon logging in.
- Allow users to set and edit financial goals
    - Users will be able to set and edit their financial goals.
    - Users will be able to remove their financial goals if they are unsatisfied with them.
- Allow users to use APIs to add investments.

We plan on rebuilding our previous project from the ground up using React components. Information previously retrieved from a JSON file will now be pulled from a database.

Feature Ownership & Responsibility

Jayson Acosta will be in charge of:

- User Authentication
- Data Synchronization
- Rebuilding Dashboard
- Rebuilding Settings
- Rebuilding Editor

Zaid Rachman will be in charge of:

- Financial Analytics
- API integration
- Profile Tiers
- Rebuilding Landing Page
- Rebuilding Credit Card Page
- Rebuilding Authors Page

Resource & Tools

Technology Stack

- Frontend: React, React-Router, Chartjs, Bootstrap
- Backend: Node.js, Expressjs
- Database: MySQL
- Authentication: JWT (JSON Web Tokens)
- APIs: Alpha Vantage, Coin Gecko

Development Tools

- Visual Studio Code (Development Environment)
- Git (Version Control)
- GitLab (Repository Hosting)
- Excalidraw (Wireframing)
- Postman (API testing)

File Structure and Project Organization

Frontend Architecture

- frontend/
    - src/
        - assets/ Images and external data
        - components/ React components
        - App.jsx, main.jsx
- backend/ Express server, API routes, and database logic
- Documents/ Planning sketches, Software Architecture Document, final report, and video.

The front end will grab userdata from the backend on login. Information such as user tier, selected theme, and financial data will be grabbed and displayed to the user accordingly.

Backend Architecture

The backend will use a RESTful API architecture. The following are the current endpoints we will utilize:

- api/
    - auth/
        - register
        - login
        - user
        - logout
    - budgets/
        - budgetID
    - expenses/
        - expenseID
    - goals/
        - goalID
    - market/
        - stocks
        - crypto

Each budget, expense, goal, and market subcategory (IDs) will allow PUT and DELETE requests, while the main categories will allow GET and POST requests.

Data Sources and Management

Our data relies heavily on user input. Users can create financial goals, retrieve previous information, update their goals, and finally remove the goal when they reach it. Our application will utilize both local storage and MySQL to ensure data persistence.

Database Schema

User Table: userID, email, password, name. tier (basic, premium), createdAt, lastlogin

Budget Table: categoryID, userID, name, monthlyBudget, alertThreshold, createdAt, lastUpdated

Expense Table: expenseID, userID, name, description, date, createdAt

Goal Table: goalID, userID, name, target, current, deadline, createdAt

Data Synchronization

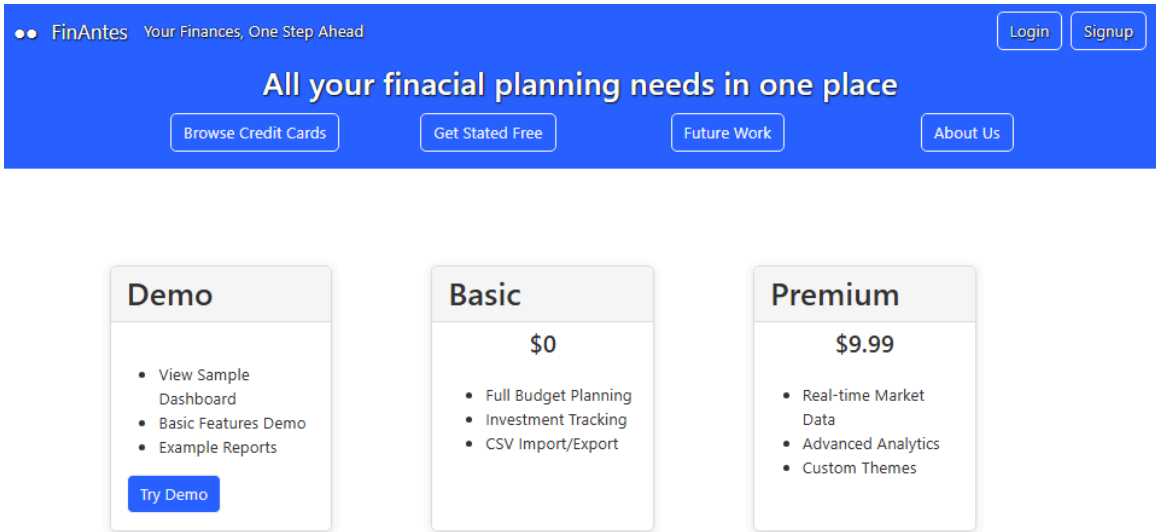
FinAntes will implement Data Synchronization by allowing users to make changes automatically saved to local storage and the MySQL database in real time. Data can be synchronized automatically or with a button on the settings page based on user preferences. In the case of conflicts, the application will choose between local and server data through timestamps, with the newest change taking precedence. During synchronization, there will be visual indicators, including progress status and a completion notification.

User Experience Views

User Interface Design Mockups

This section showcases the basic design of FinAntes through detailed wireframes. The images are not precisely what will be implemented; however, they will serve as a basic understanding of what we will accomplish.

Landing Page



Your Finances, One Step Ahead

All your finacial planning needs in one place

Demo

- View Sample Dashboard
- Basic Features Demo
- Example Reports

Basic

$0

- Full Budget Planning
- Investment Tracking
- CSV Import/Export

Premium

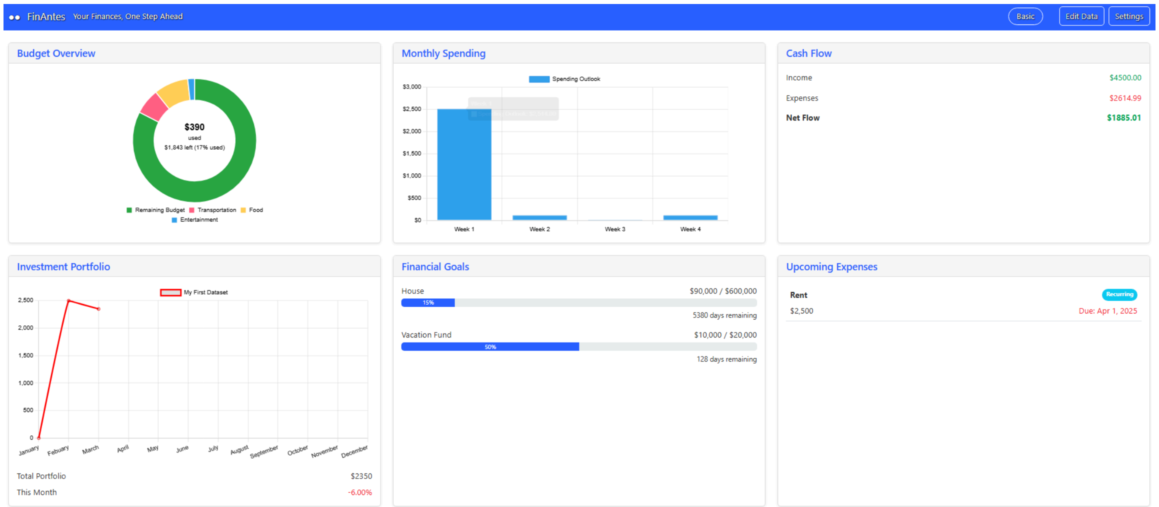
$9.99

- Real-time Market Data
- Advanced Analytics
- Custom Themes

Figure 1: Landing Page

The landing page is designed to immediately show off FinAntes' purpose with its slogan, “Your Finances, One Step Ahead,” displayed in the hero section. The three-tier pricing structure is presented in cards: Demo, Basic, and Premium. Teach tier-card lists the key features with easy-to-understand pricing. There are login and signup buttons on the top of the page to let user onboarding be straightforward.

Financial Dashboard



Your Finances, One Step Ahead

Budget Overview

Monthly Spending

Cash Flow

Income	$4500.00
Expenses	$2614.99
Net Flow	**$1885.01**

Investment Portfolio

Total Portfolio	$2350
This Month	-6.00%

Financial Goals

House $90,000 / $600,000

5380 days remaining

Vacation Fund $10,000 / $20,000

128 days remaining

Upcoming Expenses

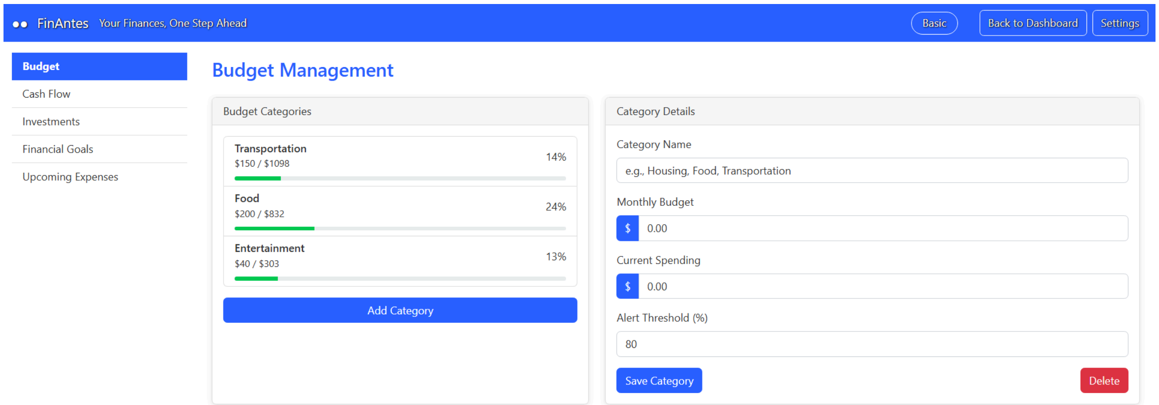
Rent

$2,500 Due: Apr 1, 2025

Figure 2: Financial Dashboard

The finance dashboard will serve as the central hub of FinAntes, featuring all the primary widgets. The budget overview will display a circular progress bar showing users’ budget utilization. Monthly spending includes a bar chart for expense tracking over time. The investment portfolio will be a line graph that updates after hours when the market closes to show how user investments performed. Savings Goals will show the progress towards each financial target. A user status indicator at the top of the screen will show either Demo, Basic, or Premium.

Budget Editor



Budget Management

Budget Categories

Transportation 14%
$150 / $1098

Food 24%
$200 / $832

Entertainment 13%
$40 / $303

Category Details

Category Name: e.g., Housing, Food, Transportation

Monthly Budget: $ 0.00

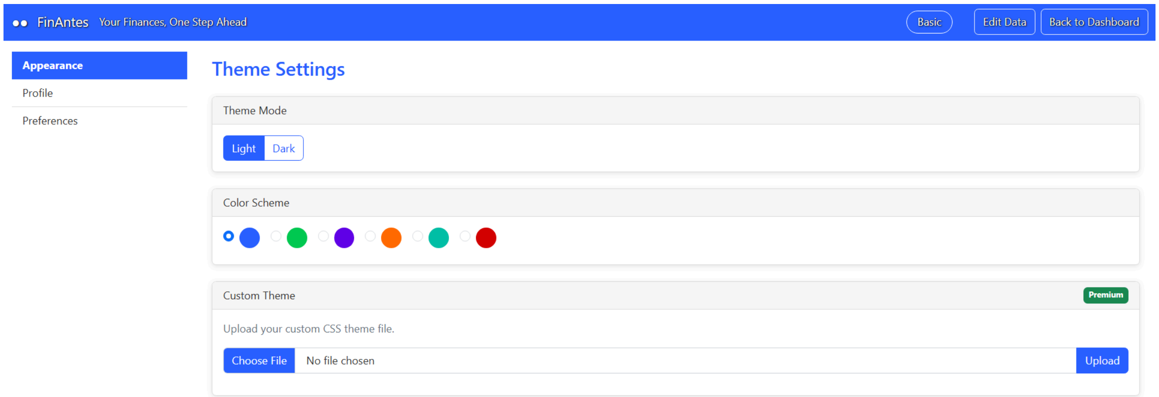
Current Spending: $ 0.00

Alert Threshold (%): 80

Figure 3: Budget Editor

The budget editor interface is split into two main sections. The lefthand panel shows a categorized view of all budget items, displaying allocated amounts and current spending for each category. Users can add new categories through a button that will submit the data. The righthand panel shows detailed controls that allow users to input the category name, the monthly budget, the alert threshold, and the total amount spent this month.

Settings Page



Theme Settings

Theme Mode: Light | Dark

Color Scheme: ● ○ ○ ○ ○ ○

Custom Theme — Premium

Upload your custom CSS theme file.

Choose File: No file chosen

Figure 4: Settings Page

The settings page provides customization options through a sidebar navigation system. The appearance section allows users to toggle between multiple colored themes, light and dark modes, custom-imported themes uploaded using a CSS file, and many more features. Premium features will be marked to encourage users to upgrade for enhanced functionality.

Login / Signup

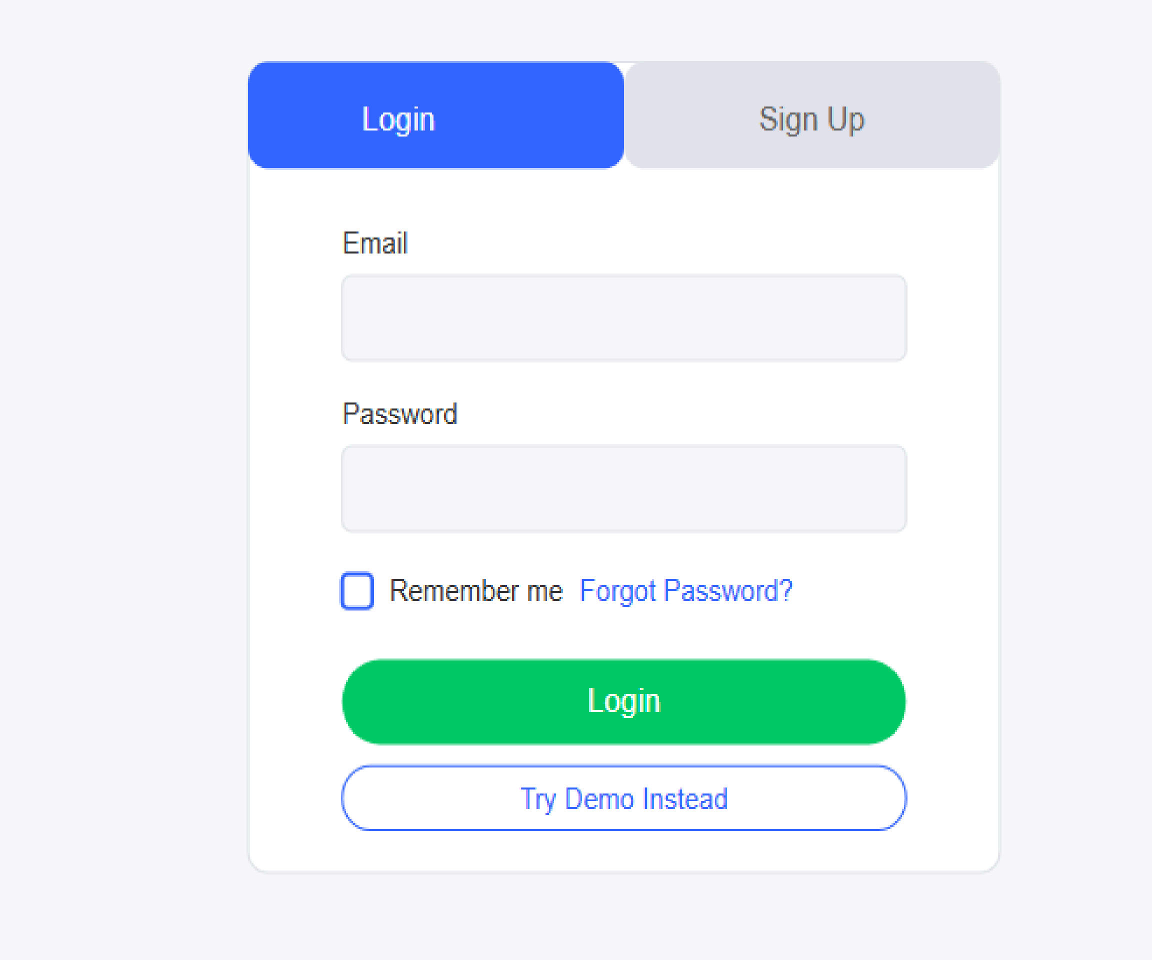
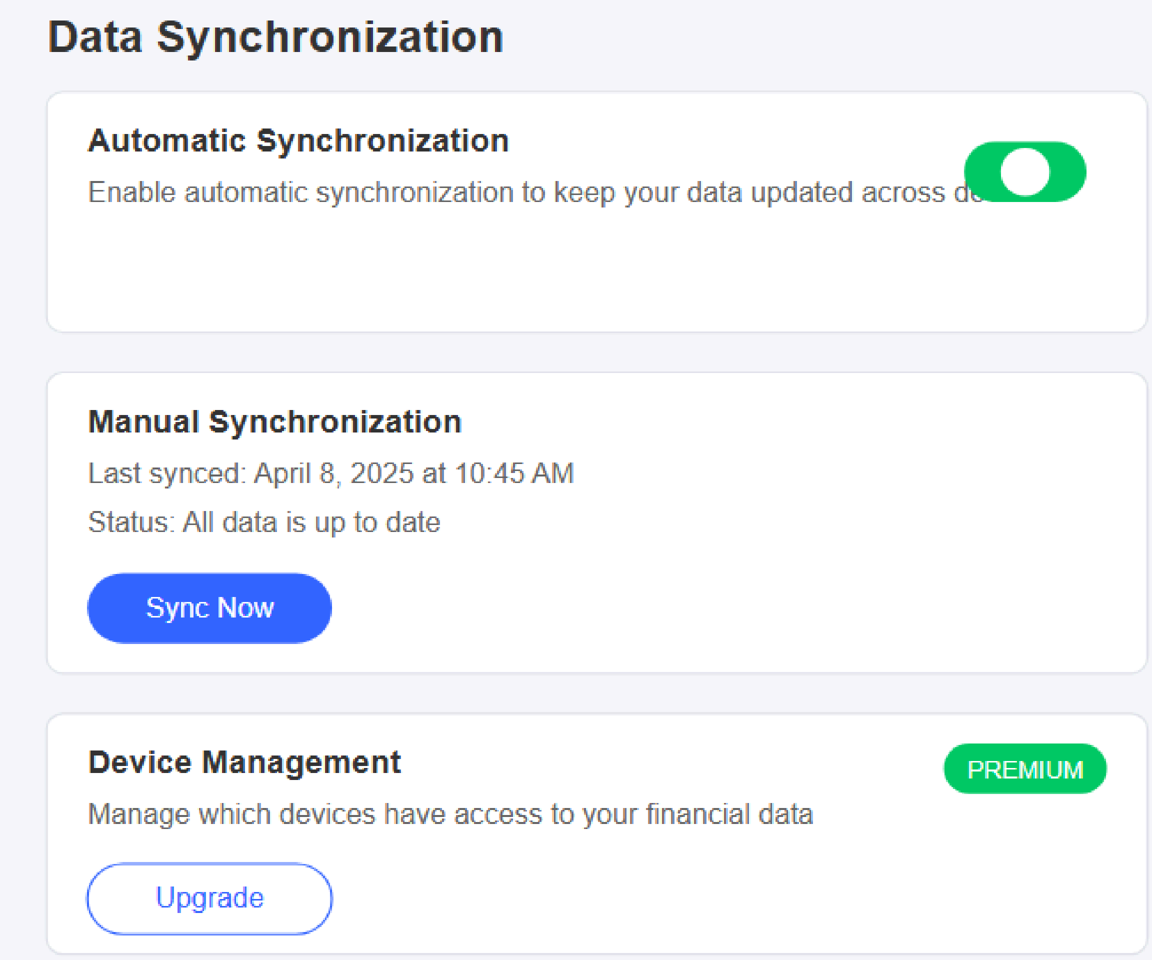


Figure 5: Login / Signup page

The login/signup interface features a clean, tabbed design, allowing users to easily switch between logging in and creating a new account. The login form includes fields for email and password, with a "Remember me" checkbox for convenience and a "Forgot Password" link for account recovery. For users unsure about creating an account, there is a static demo page allowing users to preview the page before deciding to sign up.

Data Synchronization in Settings Page



Data Synchronization

Automatic Synchronization

Enable automatic synchronization to keep your data updated across de

Manual Synchronization

Last synced: April 8, 2025 at 10:45 AM

Status: All data is up to date

Device Management

Manage which devices have access to your financial data

Figure 4: Settings Page

The data synchronization tab in the settings page provides users with control over how their financial data synchronizes across devices. It features an automatic synchronization toggle in the settings menu, allowing users to turn real-time updates on or off. Under the Manual Synchronization section, a button will enable users to synchronize their data, along with the last synched time and date and the synchronization status.

Final Comments

This project will be effective in helping people organize and manage their finances. By the end of this project, we hope to have gained valuable experience in developing a full-stack application. If you need to contact us, please email us at Jayson04@iastate.edu or ZaidR@iastate.edu.