Business Requirement Document (BRD)

Project Title: SpendSmart

Prepared by: Aashin K S & Avula Venumadhav

1 Project Overview

Managing personal finances can be difficult, especially for individuals who lack financial literacy or have impulsive spending habits. Traditional budgeting tools often require manual entry and offer limited guidance, leaving users overwhelmed.

SpendSmart aims to automate personal finance management by integrating with global financial APIs to fetch real-time bank transaction data, help users track expenses, manage budgets, receive financial advice via an AI-powered chatbot, and stay informed with personalized alerts—all from a single intuitive platform.

2 Business Goals

- Enable users to link their bank accounts securely and automatically import transactions.
- Automate expense tracking and categorize spending intelligently.
- Provide personalized financial insights and budget alerts.
- Help users set, monitor, and achieve their savings goals.
- Ensure a scalable, secure, and user-friendly platform with microservices architecture.

3 User Stories

- As a user, I want to securely link my bank account to automatically fetch my transactions.
- As a user, I want my expenses to be automatically categorized to understand my spending.
- As a user, I want to receive smart alerts when I'm close to exceeding my budget.
- As a user, I want to create savings goals and track progress.
- As a user, I want to view monthly and yearly spending summaries and financial health.
- As a user, I want to interact with a chatbot for personalized financial advice.
- As an admin, I want to manage user accounts, monitor platform usage, and ensure security.

4 Solutions Provided by SpendSmart

- Bank Integration: Securely link and fetch bank transactions via APIs like Plaid.
- Automatic Expense Categorization: Classify transactions by merchant, category, and pattern.
- Smart Budget Planner: Define monthly/annual budgets, with real-time monitoring.
- Goal Tracker: Set, edit, and monitor savings goals and track achievements.
- AI-Driven Chatbot: Financial advice chatbot powered by Ollama LLM models.
- Email Notifications: Important updates and alerts via Nodemailer.
- Interactive Dashboard: Visual charts and summaries for income, expenses, goals, and trends.

5 Tech Stack

- Backend: Node.js, Express.js
- Frontend: Angular
- Database: MongoDB (via Mongoose)
- API Gateway: Express Gateway (for routing and API management)
- Third-Party APIs:
  - Plaid API (Bank linking and transactions)
  - Ollama LLM (Financial advice chatbot)
- Email Service:
  - Nodemailer (for sending email notifications)

6 Key Milestones

1. Backend Setup: Node.js + Express.js services for Authentication, Bank Account Linking, Transaction Fetching and Categorization, Budget and Goal Management, Notification Services, AI Chatbot Microservice.
2. Frontend Setup: Angular development, UI component creation, User-friendly routing.
3. API Gateway Setup: Setup Express Gateway, Configure services, routes, and policies for security and load management.
4. Third-Party API Integration: Plaid, Ollama, Nodemailer integration.
5. UI/UX Designing: Responsive designs for Dashboard, Budget Tracker, Goal Management, Transaction Insights, and Chatbot Interface.
6. Frontend-Backend Integration: Connect Angular frontend to Node.js backend via Express Gateway.
7. Testing & Deployment: Perform full-stack testing, optimize performance, and deploy to cloud (e.g., AWS).

7 Deliverables

- Full-stack responsive web application (MEAN stack)
- Secure bank linking and transaction integration
- Categorized transaction dashboard
- Budget management and goal tracking modules
- AI-powered financial chatbot
- REST APIs with full documentation
- Admin control panel (for user and system management)
- Deployed application with user instructions
- Data privacy and security compliance documentation