Decoding Data Science(DDS) Academy

AI Application Building Challenge

Okezino Akpobome

Day 1 Idea Submission Template

Project Title:
SmartSaver: Your Personal Finance Coach

Concept Summary:
(Briefly describe your idea, including its purpose and the problem it aims to solve.)

SmartSaver is a smartphone software that links directly to your bank account to assess your spending habits and help you save money more effectively. The software categorizes spending, detects unnecessary ones, and provides individualized budgeting plans based on AI-powered insights. It sends real-time alerts for overspending and suggests micro-savings options for simple financial security.

Target Audience:
(Who will benefit from your project? Describe their needs.)

The app is designed for individuals who struggle with managing their finances, particularly:

- Young professionals learning financial independence
- Students looking to save on limited budgets
- Families seeking better money management for household expenses

Their needs include:

- Tracking spending effortlessly.
- Identifying areas where they can cut costs.
- Creating and sticking to personalized savings goals.

Key Features:
(List the main features or functionalities of your application.)
1. **Bank Integration:** Securely link your bank accounts to import and categorize transactions automatically.
2. **Micro-savings Automation:** Automatically round up transactions to the nearest dollar and save the difference.
3. **Expense Insights:** Provide detailed reports highlighting areas for improvement in financial habits.

Technical Approach:
(How do you plan to implement your idea? Mention technologies or methods.)

- Use banking APIs to integrate with user bank accounts securely.

- Implement AI algorithms for expense categorization, pattern recognition, and personalized budgeting.
- Develop a mobile application using Flutter for cross-platform compatibility (iOS & Android).
- Employ a cloud-based database like Firebase for secure data storage and real-time updates.

Expected Challenges:

(What potential obstacles do you foresee, and how will you address them?)

1. **Data Privacy & Security:** Ensuring the secure handling of sensitive user data.
   a. Solution: Implement encryption and authentication protocols.
2. **Bank Integration:** Supporting diverse banking systems with varying API structures.
   a. Solution: Partner with a robust financial data aggregator like Plaid.
3. **User Engagement:** Keeping users actively involved in the app's features.
   a. Solution: Introduce gamification and personalized notifications to maintain engagement.

Submission Format:

(Outline how your submission should be presented: documents, slides, etc.)

- **Slides:** Visual presentation summarizing features, benefits, and implementation approach.
- **Demo Video:** Prototype showcasing app functionality.

Expected Outcome:

(What do you aim to achieve by the end of the challenge?)

By the end of the project, the goal is to deliver a functional prototype of SmartSaver that securely connects to bank accounts, analyzes expenses, and offers actionable saving strategies. This app will empower users to achieve their financial goals effortlessly.

Additional Notes (Optional):

Achievements on Day 3

1. API Enhancements

- **Functionality Expansion**:

- Integrated OpenAI's GPT-4 API to provide multi-turn conversations and context-aware interactions.

- Implemented functionality for user spending analysis by categorizing transactions and generating savings advice.

- **Error Handling**:

  - Added robust error-handling mechanisms:

    - Retry logic for API calls in case of timeouts or failures.

    - Fallback responses to ensure a seamless user experience during API unavailability.

  - Graceful handling of rate limits and invalid user inputs with appropriate feedback messages.

- **Performance Optimization**:

  - Reduced API latency by:

    - Implementing caching for repeated queries.

    - Batching multiple requests for efficiency.

  - Enhanced context management to ensure concise, relevant responses from the API.

2. UI Integration

- Developed a Flask-based interface to connect the backend API functionality with a basic user-friendly front-end.

- Displayed personalized savings suggestions dynamically based on user data and GPT-4 interactions.

- Integrated initial feedback options to capture user opinions on provided suggestions (e.g., thumbs up/down).

Challenges Encountered and Solutions

- **Challenge 1**:

  - **Issue**: Managing API rate limits.

- **Solution**: Implemented retry mechanisms with exponential backoff and added local caching for frequently accessed data.

- **Challenge 2**:

    - **Issue**: Dynamically handling diverse user inputs for personalized suggestions.

    - **Solution**: Used natural language parsing to detect patterns and match user intents, improving interaction accuracy.

Current Project Status

- **Core Functionalities**:

    - Successful integration of GPT-4 API for transaction analysis and savings advice generation.

    - Basic Flask-based UI to demonstrate backend functionality.

- **Planned Next Steps**:

    - Enhance UI/UX for better interactivity and user engagement.

    - Expand functionalities to include additional APIs for transaction data retrieval and deeper financial insights

    - Add logging for user feedback to improve suggestions iteratively.

https://colab.research.google.com/drive/14EmbKhcpsOS-QMHcbwXEZve8z8eH4Nu2