Hackathon Project Phases Template

Project Title:

Advancing Nutrition Science through GeminiAI

Team Name:

TECK TWINKLES

Team Members:

B.PRAVALIKA(23WH1A6653
P.PRAVALIKA(23WH1A6609
M.VAISHNAVI PRASAD(23WH1A6614)
B.NANDINI(24WH5A6607)

Phase-1: Brainstorming & Ideation

Objective:

The objective of the web-based nutritional application is to provide users with instant, detailed nutritional information on food items using Google Generative AI, enabling informed dietary choices and personalized recommendations for healthier eating.

Key Points:

1. **Problem Statement:**

◯ Provide instant nutritional information on food items using Google Generative AI.

◯ Provide instant nutritional information on food items using Google Generative AI. Support informed dietary choices with personalized recommendations and educational resources.

2. **Proposed Solution:**

◯ Develop a user-friendly web app using Google Generative AI for instant nutritional data on food items.
Include personalized recommendations and tracking features to promote healthier eating habits.

3. **Target Users:**
   - ◯ Health-conscious individuals aiming to improve their
   - ◯ Fitness enthusiasts optimizing their nuitrition
   - ◯ People with specific dietary needs or restrictions.

4. **Expected Outcome:**

   ◯ Users will gain a better understanding of nutrition, leading to improved dietary choices and increased engagement with the application.

Phase-2: Requirement Analysis

Objective:

Define the technical and functional requirements for the AutoSage App.

Key Points:

1. **Technical Requirements:**
   ◯ Frontend and Backend: Utilize React.js for the frontend and Node.js with Express.js for the backend, with MongoDB or PostgreSQL for data storage.

2. **Technical Requirements:**

   ◯ AI and API Integration: Integrate Google Generative AI for real-time insights and connect to third-party nutritional data APIs for comprehensive food information.

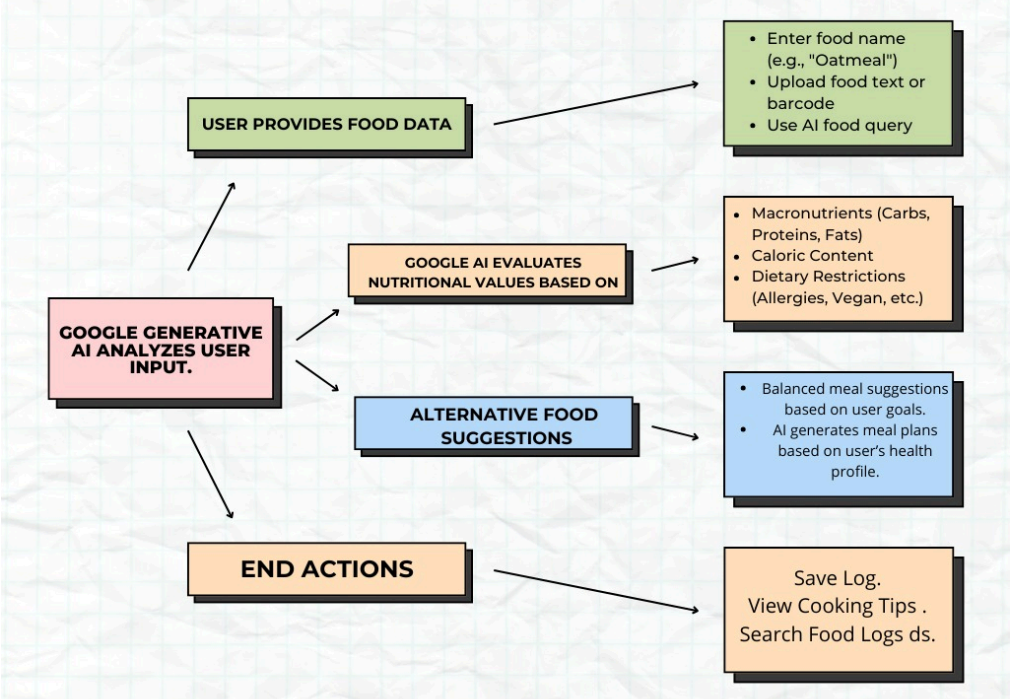
   ◯ Hosting and Security: Deploy on cloud platforms like AWS or Google Cloud, ensuring secure user authentication with OAuth 2.0 and HTTPS for data protection.

3. **Constraints & Challenges:**

   ◯Data Accuracy: Ensuring reliable nutritional information from APIs can affect user trust.
   ◯ User Engagement: Maintaining consistent user engagement requires effective UI/UX and feature u
   ◯ Scalability: Scaling the application to handle increased traffic without performance loss is a key challenge.

Phase-3: Project Design

Objective:



Key Points:

1. **System Architecture:**

   ◯ User Input: Enter food-related query via text, image, barcode, or voice.
   ◯AI Processing:Google Generative AI & Vision API analyze food and fetch nutritional data
   ◯ Educational Resources:Provides articles, videos, and expert tips based on user diet.

2. **User Flow:**

   ◯ Step 1: Login/Setup: Users register or log in and set up their dietary profile.

   ◯ Step 2:Search & Insights: Users search for food and receive nutritional information and personalized insights.

   ◯ Step 3: Tracking: Users log meals and get tailored dietary recommendations.

3. **UI/UX Considerations:**

   ◯ User -Friendly Design: Ensure intuitive navigation and a clean layout.
   ◯ Responsive & Accessible: Optimize for all devices and meet accessibility standards.

   ◯ Feedback & Customization: Provide action feedback and allow user personalization.

Phase-4: Project Planning (Agile Methodologies)

Objective:

Break down development tasks for efficient completion.

Sprint	Task	Priority	Duration	Deadline	Assigned To	Dependencies	Expected Outcome
Sprint 1	Environment Setup & API Integration	High	6 hours (Day 1) 2	End of Day 1	Member 2	None	working; Python e environment setup
Sprint 1	Frontend UI Development	Medium	hours (Day 1)	End of Day 1	Member 1 & 2	API response	Basic UI with input fields functional
Sprint 2	Food Search & Comparison	High	10 hours (Day 2)	Mid-Day 2	Member 1& 2	API response	Food search with filters
Sprint 2	Error Handling & Debugging	High	4 hours (Day 2)	Mid-Day 2	Member 1&2	API logs, UI bugs	Improved API stability
Sprint 3	Testing & Enhancements	Medium	1.5 hours (Day 2)	Mid-Day 2	Member 2& 1	APIresponse,UI layout	Better user experience
Sprint 3	Final Presentation & Deployment	Low	2 hour (Day 2)	End of Day 2	Entire Team	Working prototype	Deplayed

Sprint Planning with Priorities Sprint 1

– Setup & Integration (Day 1)

(High Priority) Set up **environment** & install dependencies.
(High Priority) the Integrate **Google Gemini API**
(Medium Priority) Build a **basic UI with input fields** .

Sprint 2 – Core Features & Debugging (Day 2)

High Priority) Implement **search & comparison functionalities**.
(**High Priority)** Debug API issues & handle **errors in queries**.

Sprint 3 – Testing, Enhancements & Submission (Day 2)

(**Medium Priority)** Test API responses, refine UI, & fix UI bugs.

(**Low Priority)** Final **demo preparation & deployment**.

Phase-5: Project Development

Objective:

Implement core features of the AutoSage App.

Key Points:

1. **Technology Stack Used:**
   - ◯ AI/NLP: Gemini API, Vertex AI.
     Cloud: GCP, Cloud SQL, Firestore, Cloud Functions/Run, Cloud Storage.
   - ◯ Frontend: React, Flutter, JavaScript/TypeScript.
     .Backend: Python, Flask/FastAPI.
   - ◯ Data: USDA FoodData Central, Open Food Facts API, Nutritionix API.
2. **Development Process:**
   - ◯Gemini Integration: (This is the core of your AI functionality)
   - ◯ User-Centric Iteration: (Prioritizing user feedback and continuous improvement)
   - ◯ Robust Testing/Deployment: (Ensuring reliability and scalability)
3. **Challenges & Fixes:**
   - ◯ **Challenge:** Accuracy/Consistency
     Fix: Clean, prioritize, disambiguate.
   - ◯ **Challenge:** Reliability
     Fix: Refine, tune, validate.

Phase-6: Functional & Performance Testing

Objective:

Ensure that the AutoSage App works as expected.

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional Testing	Query use case testing	Result should be display.	✅ Passed	Tester 1
TC-002	Functional Testing	Query “Maintenance tips for winter"	Detailed tips should be provided.	✅ Passed	Tester 2

TC-003	Performance Testing	API response time under 500ms	API should return results quickly.	Passed	Tester 3
TC-004	Bug Fixes & Improvements	Fixed correct API responses.	Data accuracy should be improved.	✅ Passed	Developer
TC-005	Final Validation	Ensure UI works across devices.	UI should work on mobile & desktop.	Passed	mobile Tester 2
TC-006	Deployment Testing	Seamless Shining	App should be accessible online.	Passed	DevOps

---

Final Submission

1. **Project Report Based on the templates**
2. **Demo Video (3-5 Minutes)**
3. **GitHub/Code Repository Link**
4. **Presentation**