

Hackathon Project Phases Template for the Flavour Fusion: AI-Driven Recipe Blogging

App project.

Project Title:

Flavour Fusion: AI-Driven Recipe Blogging

Team Name:

Taste Techies

Team Members:

- P.Bhavana
 - P.Rashmitha
 - P.Navaneetha
 - S.Pranavi
-

Phase-1: Brainstorming & Ideation

Objective:

The **Flavour Fusion Project** aims to leverage **Generative AI (Gen AI)** to revolutionize culinary experiences by blending diverse cuisines, creating unique recipes, and enhancing food innovation. This initiative integrates AI-driven creativity with traditional culinary arts to develop novel taste profiles that appeal to global palates. The project seeks to utilize Gen AI to generate fusion recipes by combining elements from various culinary traditions, innovate flavor pairings based on AI-driven analysis of taste compatibility, enhance food personalization through AI-driven recommendations, explore sustainable ingredient alternatives using AI insights, and foster collaboration between chefs, food scientists, and AI experts.

Key Points:

1. Problem Statement:

• Fusion Concept Development

- Identify and analyze global cuisines for fusion (e.g., Japanese-Italian, Indian-Mexican).
- Use AI to suggest ingredient pairings and preparation techniques.

- **AI-Powered Recipe Generation**

- Train AI on culinary datasets to understand textures, flavours, and synergies.
- Generate and refine fusion recipes with chef collaboration.

- **Taste & Texture Optimization**

- Use AI to enhance flavour harmonization and plating aesthetics.
- Predict optimal ingredient combinations based on sensory data.

2. **Proposed Solution:**

- To address the challenges of fusion cuisine innovation, personalization, and sustainability, the **Flavour Fusion Project** leverages **Generative AI (Gen AI)** to enhance culinary creativity and efficiency.

3. **Target Users:**

- **Chefs & Culinary Professionals**

- Use AI-generated fusion recipes for menu innovation.
- Explore new ingredient pairings and molecular gastronomy techniques.

- **Food Enthusiasts & Home Cooks**

- Discover unique, AI-powered fusion recipes.
- Personalize recipes based on dietary preferences and available ingredients.

- **Restaurants & Food Businesses**

- Enhance menu offerings with AI-driven fusion dishes.
- Optimize ingredient sourcing and minimize food waste through AI insights.

4. **Expected Outcome:**

An AI-powered platform that generates innovative fusion recipes, optimizes flavors and textures, enhances culinary personalization, promotes sustainable ingredient usage, enhance word count and programming jokes.

Phase-2: Requirement Analysis

Objective:

Define the technical and functional requirements for the flavour fusion.

Key Points:

1. Technical Requirements:

- Programming Language: **Python**
- Backend: **Google Gemini Flash API**
- Frontend: **Streamlit Web Framework**
- Database: **Not required initially (API-based queries)**

2. Functional Requirements:

- AI-powered system to generate fusion recipes using global culinary datasets
- Personalized recommendations based on user preferences and dietary needs.
- AI-driven suggestions for sustainable ingredient alternatives and food waste reduction.
- Interactive platform enabling chefs and users to modify and experiment with AI-generated recipes.
-

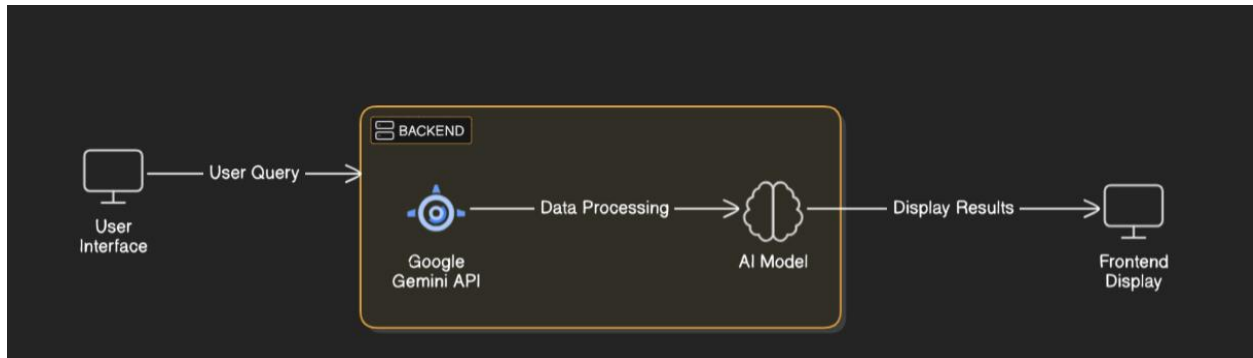
3. Constraints & Challenges:

- Requires large, high-quality culinary datasets for accurate AI training.
 - Balancing authenticity and creativity in fusion recipes.
 - Providing a **smooth UI experience** with Streamlit.
-

Phase-3: Project Design

Objective:

Develop the architecture and user flow of the application.



Key Points:

1. System Architecture:

- **Data Collection Module:** Gathers recipes, ingredient profiles, and taste pairings from diverse sources.
- **AI Model (Generative AI - NLP & Computer Vision):** Uses machine learning techniques to analyze, generate, and refine fusion recipes.
- **User Interface (Web/Mobile App):** Allows users to input preferences and receive AI-generated fusion recipes.
- **Feedback Mechanism:** Enables users to rate and refine AI-generated suggestions.

2. User Flow:

- Step 1: User enters a query.
- Step 2: The backend **calls the Gemini Flash API** to retrieve vehicle data.
- Step 3: The app processes the data and **displays results** in an easy-to-read format.

3. UI/UX Considerations:

- **User-Friendly Interface:** Ensure intuitive navigation with a clean, visually appealing design.
- **Personalization:** Adaptive UI elements based on user preferences, such as dietary restrictions and taste preferences.

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)	End of Day 1	P.Bhavana	Google API Key, Python, Streamlit setup	API connection established & working
Sprint 1	Frontend UI Development	☐ Medium	2 hours (Day 1)	End of Day 1	P.Rashmitha	API response format finalized	Basic UI with input fields
Sprint 2	Vehicle Search & Comparison	☐ High	3 hours (Day 2)	Mid-Day 2	P.Rashmitha	API response, UI elements ready	Search functionality with filters
Sprint 2	Error Handling & Debugging	☐ High	1.5 hours (Day 2)	Mid-Day 2	P.Navaneetha	API logs, UI inputs	Improved API stability
Sprint 3	Testing & UI Enhancements	☐ Medium	1.5 hours (Day 2)	Mid-Day 2	S.Pranavi	API response, UI layout completed	Responsive UI, better user experience
Sprint 3	Final Presentation & Deployment	☐ Low	1 hour (Day 2)	End of Day 2	Entire Team	Working prototype	Demo-ready project

Sprint Planning with Priorities

Sprint 1 – Setup & Integration (Day 1)

(High Priority) Set up the **environment** & install dependencies.

(High Priority) 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 Flavour Fusion: AI-Driven Recipe Blogging

App.

Key Points:

1. Technology Stack Used:

- **Frontend:** Streamlit
- **Backend:** Google Gemini Flash API
- **Programming Language:** Python

2. Development Process:

- Implement **API key authentication** and **Gemini API integration**.
- Develop **vehicle comparison and maintenance tips logic**.
- Optimize **search queries for performance and relevance**.

3. Challenges & Fixes:

- **Challenge:** Delayed API response times.
Fix: Implement **caching** to store frequently queried results.
 - **Challenge:** Limited API calls per minute.
Fix: Optimize queries to fetch **only necessary data**.
-

Phase-6: Functional & Performance Testing

CATEGORY	TEST SCENARIO	EXPECTED OUTCOME
Functional Testing	AI Recipe Generation	AI should generate relevant and unique fusion recipes.
Functional Testing	Customization & Preferences	Users should be able to modify recipes based on dietary needs.
Performance Testing	Scalability Testing	The system should scale effectively with user growth.