Project Title:

Flavour Fusion: AI driven recipe blogging

Team Name:

The MohanGPT Syndicate

Team Members:

- Hrushikesh
- Prasad
- Vishwesh
- Laxman
- Chaitanya

Phase-1: Brainstorming & Ideation

Objective:

Develop an AI-powered recipe blogging web application that leverages Google's Generative AI to create unique and customized recipe blogs. The app allows users to input a topic and specify the desired word count for their blog, generating engaging content accordingly. Additionally, it enhances user experience with a fun feature—telling a programmer joke while the AI generates content.

Key Points:

1. **Problem Statement:**

Many food bloggers and content creators struggle with generating high-quality, engaging, and unique recipe content. Crafting detailed recipe blogs takes time and effort, and finding fresh ways to present recipes can be challenging.

2. **Proposed Solution:**

   An AI-powered application using Google's Generative AI to generate customized recipe blogs based on user input.
   The app allows users to enter a recipe topic and word count, generating detailed and engaging content. It also includes a fun feature that displays a programmer joke while processing. Additional options include downloading recipes as PDFs, providing a youtube link which has a detailed recipe, and displaying a nutrition breakdown block

3. **Target Users:**
   - ◯ Food bloggers looking for high-quality recipe content.
   - ◯ Content creators needing AI-generated blogs.
   - ◯ Cooking enthusiasts wanting structured recipes.
4. **Expected Outcome:**

   - ◯ A functional AI-powered recipe blogging app that provides high-quality, auto-generated recipe blogs with interactive features.

Phase-2: Requirement Analysis

Objective:

Define the technical and functional requirements for Flavour Fusion.

Key Points:

1. **Technical Requirements:**

   - ◯ Programming Language: **Python**
   - ◯ Backend: **Google Generative AI(Gemini API)**
   - ◯ Frontend: **Streamlit Web Framework**

2. **Functional Requirements:**

   - ◯ Generate recipe blogs based on user inputs.
   - ◯ Provide real-time AI-generated content with formatting.
   - ◯ Display programmer jokes while processing.
   - ◯ Allow users to download recipes as PDFs.
   - ◯ Integrate food ordering functionality via Swiggy API.
   - ◯ Embed YouTube video links for visual recipe guidance.
   - ◯ Displays the number of servings and preparation time for each recipe.

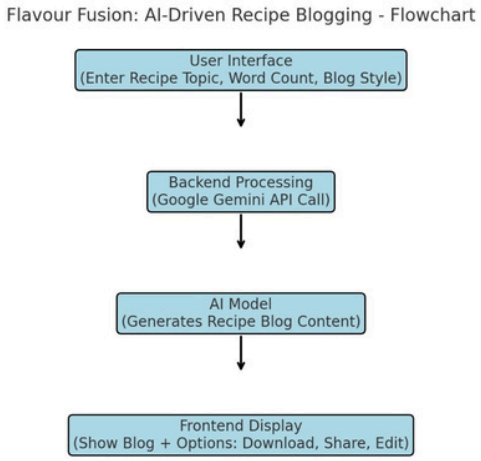
3. **Constraints & Challenges:**

   - ◯ Ensuring content accuracy and uniqueness.
   - ◯ Handling API rate limits effectively.
   - ◯ Optimizing UI/UX for better user engagement.

Phase-3: Project Design

Objective:

Develop the architecture and user flow of the application.



Key Points:

1. **System Architecture:**

   User enters a recipe topic and word count via the UI.

   Query is processed using Google Generative AI (Gemini API).

   AI model fetches and processes data to generate a structured recipe blog.

   The frontend displays the blog content with additional features like PDF downloads, YouTube link related to the recipe, Swiggy API

2. **User Flow:**

   - Step 1: User enters a recipe topic and selects word count.
   - Step 2: The backend calls the Gemini API to generate a blog post.
   - Step 3: The app processes the data and **displays results** in an easy-to-read format.

3. **UI/UX Considerations:**

   - **Minimalist, user-friendly interface .**
   - Options to customize blog style (casual, formal, fun, etc.).

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	Laxman, Hrushikesh	Google API Key, Python, Streamlit setup	API successfully configured and tested
Sprint 1	Frontend UI Development	🟡 Medium	2 hours (Day 1)	End of Day 1	Prasad	UI/UX design finalized	Functional and visually appealing UI
Sprint 2	Recipe Generation Feature	🔴 High	3 hours (Day 2)	Mid-Day 2	Vishwesh	API integration, UI input fields	AI generates recipes with user-defined parameters
Sprint 2	Joke generator Feature	🔴 High	1.5 hours (Day 2)	Mid-Day 2	Chaitanya	API response handling, UI updates	Display random cooking jokes with each recipe
Sprint 3	Testing & UI Enhancements	🟡 Medium	1.5 hours (Day 2)	Mid-Day 2	Chaitanya	Recipe Text Output	Users can download the recipes as PDFs
Sprint 3	Final Presentation & Deployment	🟢 Low	1 hour (Day 2)	End of Day 2	Prasad	Fully functional web app	Demo-Ready project for the hackathon

Sprint Planning with Priorities

Sprint 1 – Setup & Integration (Day 1)

(🔴 **High Priority)** Set up Google Gemini API and configure Python & Flask environment.
(🔴 **High Priority)** Ensure API connectivity and successful response handling.
(🟡 **Medium Priority)** Build a **basic UI with input fields**.

Sprint 2 – Core Features & Debugging (Day 2)

(🔴 **High Priority)** Implement Recipe Generation using Gemini AI to generate customized recipes.
(🔴 **High Priority)** Add a Joke Generator feature that displays random cooking jokes along with recipes.

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

(🟡 **Medium Priority)** Implement Testing & UI Enhancements, fixing any detected bugs.

(🟡 **Medium Priority)** Ensure a smooth user experience across devices.

(🟢 **Low Priority)** Conduct the Final Presentation & Deployment to make the project demo-ready.

Phase-5: Project Development

Objective:

Implement core features of the Flavour Fusion Application.

Key Points:

1. **Technology Stack Used:**
    - ◯ **Frontend:** Streamlit
    - ◯ **Backend:** Google Gemini API
    - ◯ **Programming Language:** Python
2. **Development Process:**
    - ◯ Implement **API key authentication** and **Gemini API integration**.
    - ◯ Develop recipe generation logic with custom user inputs.
    - ◯ Add a joke generator feature for user engagement.
    - ◯ Enable recipe download in PDF format.
    - ◯ Embed YouTube video links for visual recipe guidance.
    - ◯ Include a nutrition breakdown block for calorie and macronutrient insights.
    - ◯ Display the number of servings and estimated preparation time for each recipe.
3. **Challenges & Fixes:**
    - ◯ **Challenge:** Delayed API response times.

      Fix: Optimize API calls and handle retries for failures.
    - ◯ **Challenge:** PDF encoding issues for special characters.

      Fix: Convert text encoding to avoid Unicode errors.

Phase-6: Functional & Performance Testing

Objective:

Ensure that the Flavour Fusion AI Recipe Generator works as expected and provides a smooth user experience.

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional Testing	Query “Vegan Chocolate Cake”	AI should return a well-structured recipe blog	✅ Passed	Hrushikesh
TC-002	Functional Testing	Query "Muddapappu”	It should return the recipe blog, eventhough if its a regional recipe	✅ Passed	Prasad

TC-003	Performance Testing	API response time under 5s	API should return results quickly.	⚠ Needs Optimization	Vishwesh
TC-004	Bug Fixes & Improvements	Ensure correct AI response handling	AI should generate consistent and relevant recipes	✅ Fixed	Chaitanya
TC-005	Final Validation	Ensure PDF generation works correctly	User should be able to download the PDFs	✅ Fixed	Laxman
TC-006	Deployment Testing	Host app using streamlit sharing	App should be accessible online	🚀 Deployed	Entire team

---

Final Submission

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