1. INTRODUCTION

1.1. Project Overview

Flavour Fusion is an AI-driven recipe blogging web application designed to generate customized and well-structured recipe blogs based on user input. The application allows users to enter a recipe topic and specify the desired word count. Using a pre-trained generative AI model, the system produces detailed and engaging recipe content in a short time.

The project aims to simplify the process of recipe content creation for food bloggers, cooking enthusiasts, and busy individuals. Instead of manually writing lengthy recipes, users can rely on the application to generate high-quality content instantly. The application is built using Streamlit for the user interface and integrates the Gemini Flash Lite (models/gemini-flash-lite-latest) model for efficient and fast text generation.

Overall, Flavour Fusion demonstrates the practical use of generative AI in content creation by providing a user-friendly, time-saving, and effective solution for automated recipe blog generation.

1.2. Objectives

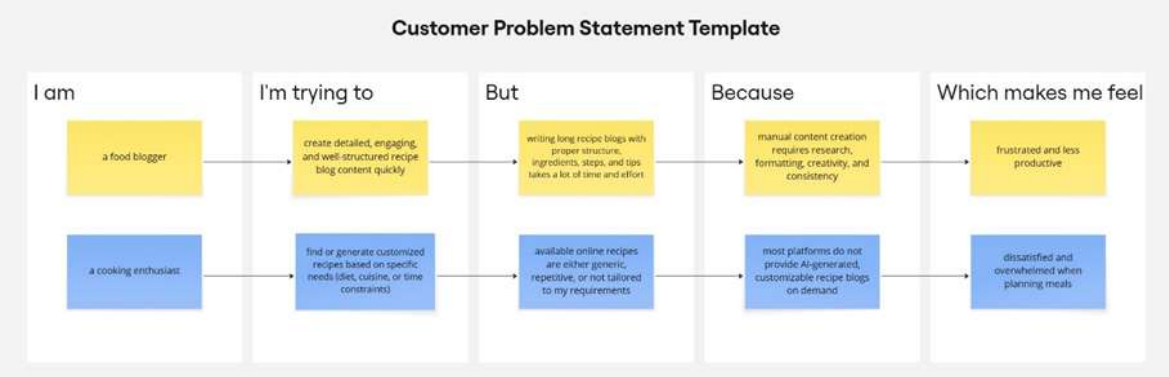
The main objectives of the Flavour Fusion: AI-Driven Recipe Blogging project are:

- To develop a web-based application that generates recipe blogs using artificial intelligence
- To allow users to create customized recipe content by providing a topic and word count
- To reduce the time and effort required for manual recipe writing
- To integrate a pre-trained generative AI model for fast and accurate text generation
- To provide a simple and user-friendly interface using Streamlit
- To enhance user experience through efficient and engaging content generation

2. Ideation Phase

2.1. Problem Statement

Therefore, food bloggers and busy individuals are in dire need of an easier way to pen or write a detailed recipe blog in the shortest length of time. Writing recipes manually requires a lot of efforts, creativities, and proper formatting that are really very time-consuming. Also, most of the recipes available online tend to be general and do not meet specific user needs. The result is usually frustration and reduced productivity. The Flavour Fusion application solves this problem by using an AI model to quickly generate customized and well-structured recipe blogs based on user input.



Problem Statement (PS)	I am	I'm trying to	But	Because	Which makes me feel
PS-1	A food blogger	Create detailed, and high-quality recipe blog content in less time	Writing long and recipe blogs manually is time-consuming	It requires research, creativity, formatting, and consistency	Frustrated and less productive
PS-2	A cooking enthusiast	Generate customized recipes	Existing recipes available online are generic and not customizable	Most platforms do not offer AI-driven recipe blog generation	Dissatisfied and overwhelmed while planning meals

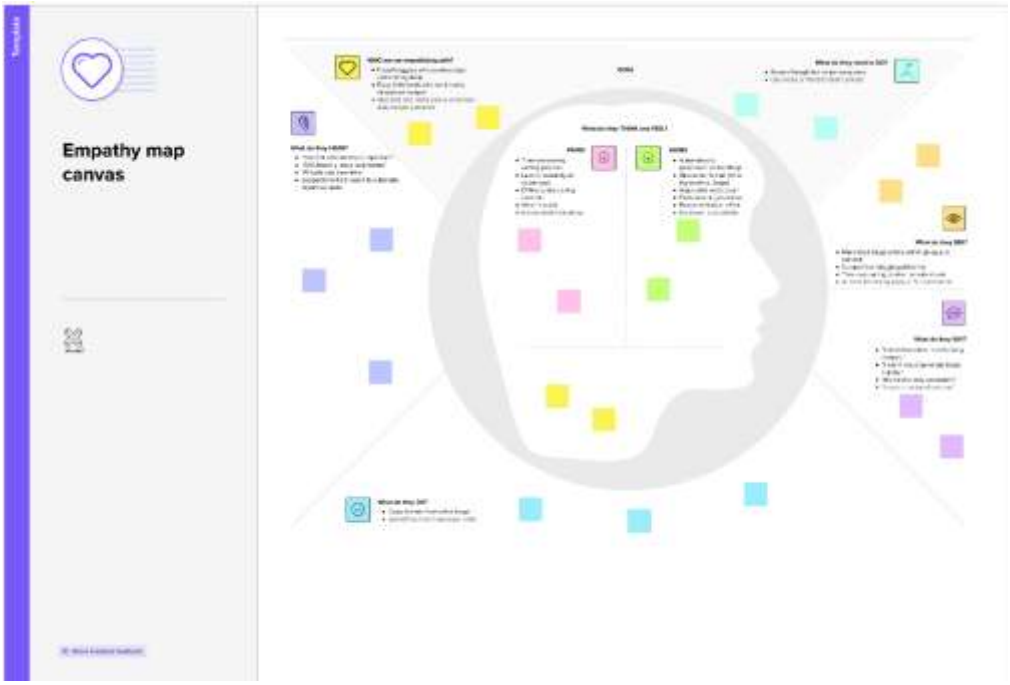
2.2. Empathy Map Canvas

Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

Example:



2.3 Brainstorming

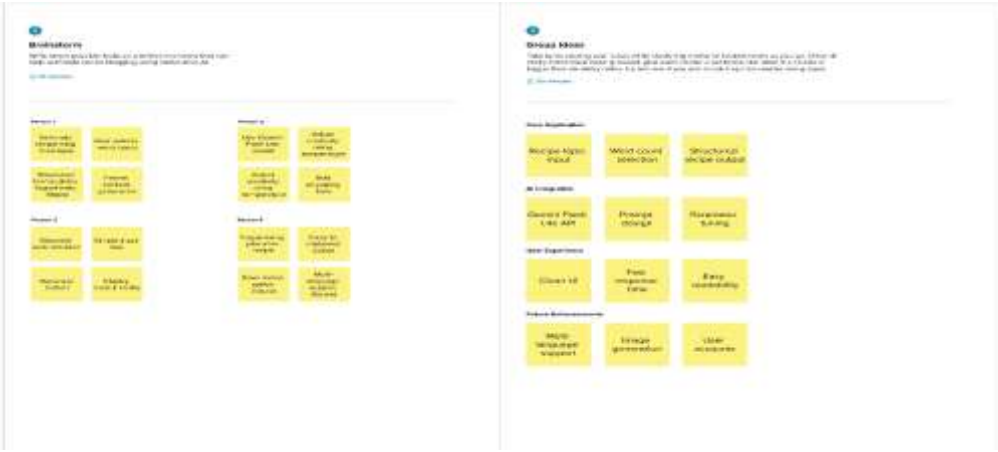
Brainstorm & Idea Prioritization:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

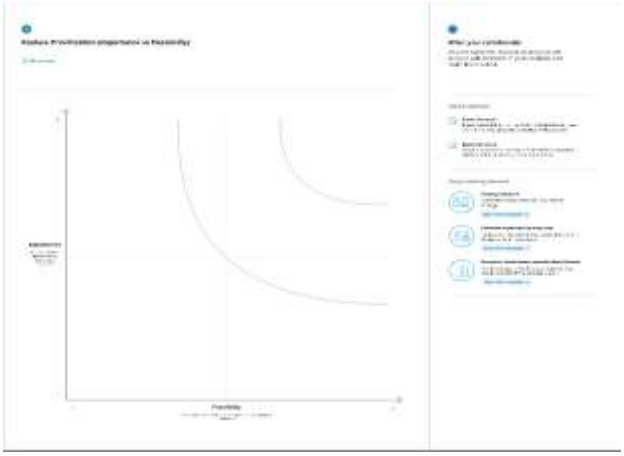
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



3. Requirement Analysis

3.1. Solution Requirement

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Recipe Input Module	Enter recipe topic & Select word count
FR-2	Input Validation	Validate empty input & word count range
FR-3	AI Integration	Connect to Gemini Flash Lite API Send structured prompt to model
FR-4	Recipe Generation	Generate structured recipe blog Include introduction, ingredients, steps
FR-5	Output Display	Display generated recipe in UI Allow user to copy content
FR-6	Additional Feature	Generate programming joke (optional feature)

Non-functional Requirements:

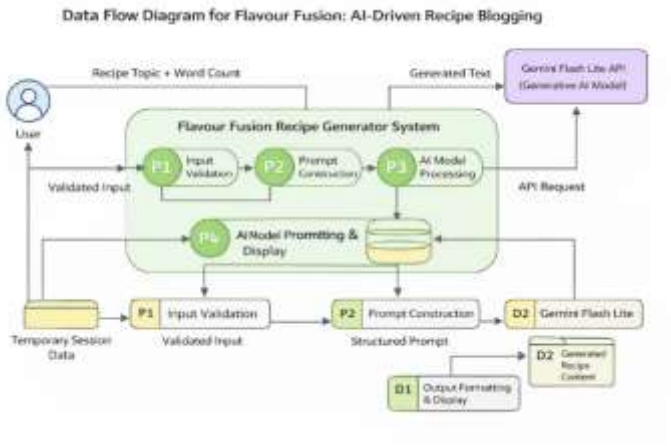
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application must have a simple and intuitive Streamlit interface that is easy to use.
NFR-2	**Security**	API keys must be securely stored and not exposed in the frontend.
NFR-3	**Reliability**	The system should generate consistent and structured outputs for valid inputs.
NFR-4	**Performance**	Recipe generation should complete within a few seconds.
NFR-5	**Availability**	The application should be accessible online whenever deployed.
NFR-6	**Scalability**	The system should handle multiple users without significant performance degradation.

3.2. Data Flow Diagram

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	User Interface Setup	USN-1	As a user, I can access a Streamlit-based interface to enter a recipe topic and word count.	I can access my account / dashboard	High	Sprint-1
Administrator	Input Validation	USN-2	As a user, I want the application to validate my inputs before generating the recipe.	I can validate the inputs	High	Sprint-1
Customer Care Executive	AI Model Integration	USN-3	As a user, I want the system to generate a recipe blog using the Gemini Flash Lite model.	I can integrate model	High	Sprint-2
Administrator	Joke Generation	USN-4	As a user, I want to see a programming joke while the recipe is being generated.	I can generate a joke for user interaction while recipe generation delayed	Medium	Sprint-2
Customer (Mobile user)	Output Display	USN-5	As a user, I want to view the generated recipe blog clearly on the screen.	I can see the output of recipe generated	High	Sprint-3
Customer Care Executive	Deployment	USN-6	As a user, I want the application to be deployed and accessible through the internet.	I can deploy in the system	Medium	Sprint-3

3.3. Technology Stack

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web-based interface where users enter recipe topic and word count.	Streamlit (Python Web Framework)
2.	Application Logic-1	Input validation and prompt construction logic	Python
3.	Application Logic-2	AI request handling and response processing	Google Generative AI API
4.	Application Logic-3	Recipe formatting and output structuring	Python
5.	File Storage	Local environment for source code and logs	Local File System
6.	External API-1	Generative AI service for recipe blog generation	Gemini Flash Lite (models/gemini-flash-lite-latest)
7.	Machine Learning Model	Pre-trained generative AI model for text generation	Gemini Flash Lite Model
8.	Infrastructure (Server / Cloud)	Deployment of application	Streamlit Cloud / Local Deployment

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Web framework and development tools used	Streamlit, Python
2.	Security Implementations	Secure storage of API keys and environment variables	Environment Variables (.env), Streamlit Secrets
3.	Scalable Architecture	Web-based architecture supporting multiple users	Cloud-based deployment (Streamlit Cloud)

S.No	Characteristics	Description	Technology
4.	Availability	Application accessible online after deployment	Streamlit Cloud Hosting
5.	Performance	Fast response generation using lightweight AI model	Gemini Flash Lite (optimized for low latency)

4. PROJECT DESIGN

4.1. Problem Solution Fit

Problem – Solution Fit Template:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs,

marketers and corporate innovators identify behavioral patterns and recognize what would work and why

Purpose:

- ❑ Solve complex problems in a way that fits the state of your customers.
- ❑ Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.
- ❑ Sharpen your communication and marketing strategy with the right triggers and messaging.
- ❑ Increase touch-points with your company by finding the right problem-behavior fit and building trust by solving frequent annoyances, or urgent or costly problems.
- ❑ **Understand the existing situation in order to improve it for your target group.**

Template:

4.2. Proposed Solution

Proposed Solution Template:

Project team shall fill the following information in the proposed solution template.

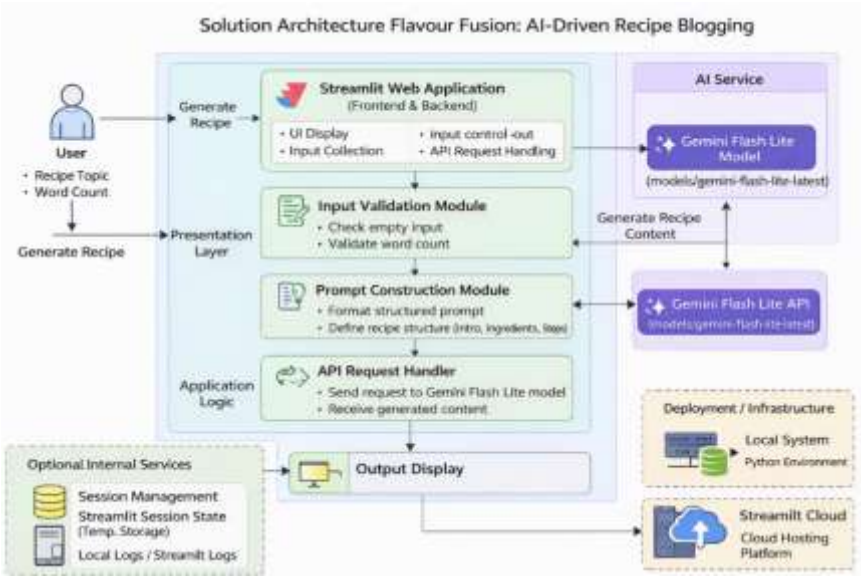
S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Many food bloggers, students, and busy individuals struggle to create well-structured and detailed recipe blogs due to time constraints and lack of consistent creativity. Manual recipe writing is repetitive, time-consuming, and requires formatting effort. There is a need for an automated system that can generate structured recipe blogs quickly and efficiently.

2.	Idea / Solution description	Flavour Fusion is an AI-powered web application that generates customized recipe blogs based on user input. Users enter a recipe topic and desired word count, and the system uses a pre-trained generative AI model (Gemini Flash Lite) to create a complete, structured recipe blog including introduction, ingredients, and preparation steps. The application provides instant results through a simple Streamlit-based interface.
3.	Novelty / Uniqueness	Unlike traditional recipe platforms that only display pre-written recipes, Flavour Fusion dynamically generates unique, customized recipe blogs in real-time. The integration of Generative AI allows users to control word count and receive structured content instantly, making it more flexible and interactive than static recipe websites.
4.	Social Impact / Customer Satisfaction	The solution saves time for food bloggers, home cooks, and students by automating content creation. It improves productivity, reduces manual effort, and provides accessible recipe content generation. Users benefit from structured and ready-to-use blogs without requiring advanced writing skills.
5.	Business Model (Revenue Model)	The application can adopt a freemium model where basic recipe generation is free, and advanced features (multi-language support, image generation, premium templates) are offered through subscription. Revenue can also be generated through advertisements, affiliate marketing, or API-based enterprise usage.
6.	Scalability of the Solution	The application is cloud-deployable and uses a scalable generative AI API. It can handle multiple users simultaneously when deployed on cloud infrastructure. Additional features such as multi-language support, user accounts, and content storage can be added to scale the system further.

4.3 Solution Architecture

Solution Architecture:

The solution architecture of Flavour Fusion consists of a Streamlit-based web application that collects user inputs and validates them. The application constructs a structured prompt and sends it to the Gemini Flash Lite generative AI model through an API call. The AI model generates a structured recipe blog, which is then formatted and displayed to the user. The system can be deployed locally or on a cloud platform such as Streamlit Cloud.


Solution Architecture Flavour Fusion: AI-Driven Recipe Blogging
User
• Recipe Topic
• Word Count
Generate Recipe
Generate Recipe
Presentation Layer
Streamlit Web Application
(Frontend & Backend)
• UI Display
• Input Collection
• Input control -out
• API Request Handling
Input Validation Module
• Check empty input
• Validate word count
Prompt Construction Module
• Format structured prompt
• Define recipe structure (intro, ingredients, steps)
API Request Handler
• Send request to Gemini Flash Lite model
• Receive generated content
Application Logic
AI Service
Gemini Flash Lite Model
(models/gemini-flash-lite-latest)
Generate Recipe Content
Gemini Flash Lite API
models/gemini-flash-lite-latest
Deployment / Infrastructure
Local System
Python Environment
Streamlit Cloud
Cloud Hosting Platform
Optional Internal Services
Session Management
Streamlit Session State
(Temp. Storage)
Local Logs / Streamlit Logs
Output Display

5. PROJECT PLANNING & SCHEDULING

5.1. Project Planning

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	User Interface Setup	USN-1	As a user, I can access a Streamlit-based interface to enter a recipe topic and word count.	2	High	All Team Members
Sprint-1	Input Validation	USN-2	As a user, I want the application to validate my inputs before generating the recipe.	1	High	All Team Members
Sprint-2	AI Model Integration	USN-3	As a user, I want the system to generate a recipe blog using the Gemini Flash Lite model.	3	High	All Team Members
Sprint-2	Joke Generation	USN-4	As a user, I want to see a programming joke while the recipe is being generated.	1	Medium	All Team Members
Sprint-3	Output Display	USN-5	As a user, I want to view the generated recipe blog clearly on the screen.	2	High	All Team Members
Sprint-3	Deployment	USN-6	As a user, I want the application to be deployed and accessible through the internet.	2	Medium	All Team Members

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint 1	20	4 Days	28 January 2026	31 January 2026	20	31 January 2026

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint 1	20	4 Days	28 January 2026	31 January 2026	20	31 January 2026
Sprint 2	20	8 Days	02 February 2026	09 February 2026	20	09 February 2026
Sprint 2	20	8 Days	02 February 2026	09 February 2026	20	09 February 2026
Sprint 3	20	7 Days	12 February 2026	18 February 2026	20	18 February 2026
Sprint 3	20	7 Days	12 February 2026	18 February 2026	20	18 February 2026

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

Test Scenarios & Results

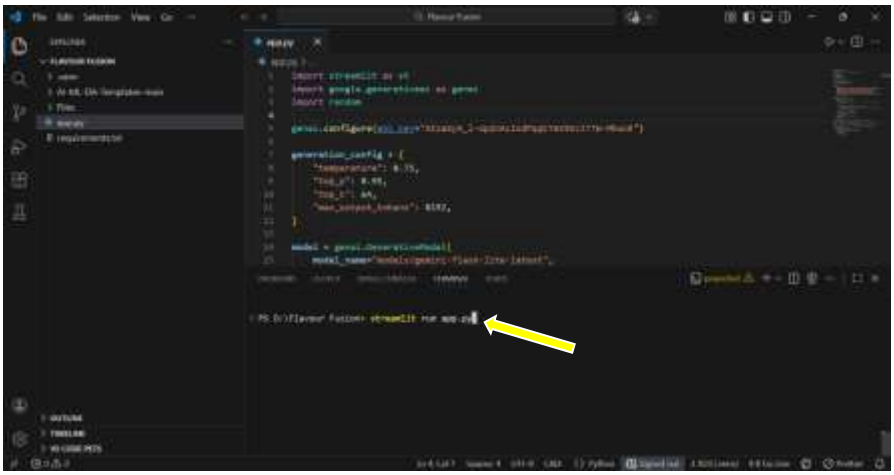
Test Case ID	Scenario (What to test)	Test Steps (How to test)	Expected Result	Actual Result	Pass/ Fail
FT-01	Recipe Topic Input Validation	Enter valid recipe topic and leave field empty	Valid input accepted, error shown for empty input	As Expected	Pass
FT-02	Ingredient Input Validation	Enter valid ingredients and try empty/invalid inputs	Accepts valid input, shows warning for invalid input	As Expected	Pass
FT-03	AI Recipe Generation	Enter recipe topic and click “Generate Recipe”	Structured recipe with title, ingredients, and steps generated	As Expected	Pass
FT-04	Gemini API Connection Check	Trigger recipe generation with valid API key	API responds successfully and generates recipe	As Expected	Pass
FT-05	Programming Joke Feature	Generate recipe and check joke display	Programming joke appears along with recipe	As Expected	Pass
PT-01	Response Time Test	Measure time after clicking generate	Recipe generated within 3–5 seconds	Within Limit	Pass
PT-02	Multiple Request Handling	Generate multiple recipes sequentially	Application handles requests without crash	Stable	Pass
PT-03	Deployment Test	Access deployed app via browser	Application loads and works correctly online	Working	Pass

7. RESULTS

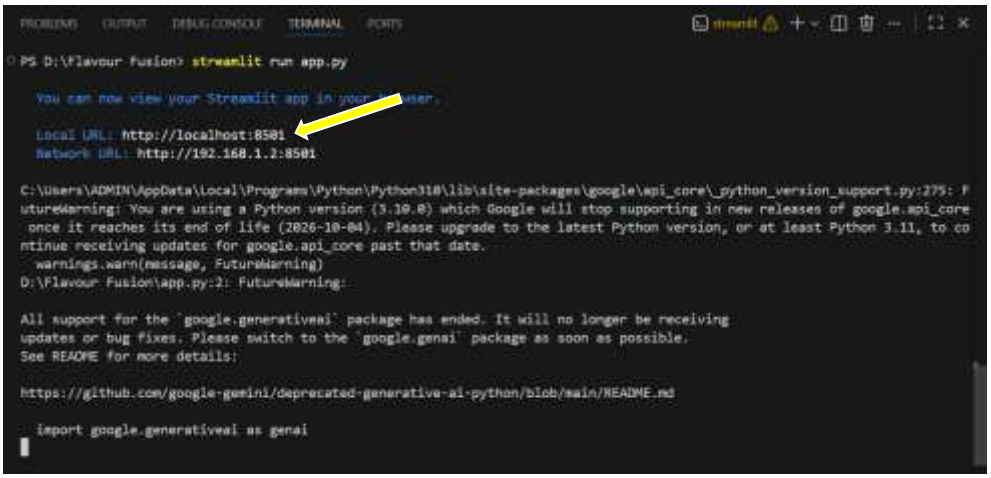
7.1. Output Screenshots

The complete execution of Flavour Fusion: AI-Driven Recipe Blogging application is represented step by step in the following screenshots.

Step 1: To run the Streamlit Application we have to use the command streamlit run app.py in the terminal in path where the app.py file is located.



Step 2: After running the command in terminal, the code will get executed and the webpage will open directly. Another way to open webpage is that a localhost link will get generated in the terminal, we can access the webpage using that link.



Step 3: The Streamlit webpage opens as shown in the figure given below. This is an automated webpage. No secondary HTML codes required to build this webpage. Python code itself consists the webpage building code.

Step 3: The user has to give inputs in the website such as a Recipe Name and Number of Words. The Number of words means with how many words the recipe should be generated. After entering the required details, the user should click on Generate Recipe button to generate the recipe. Here I chose the Chocolate Cake Recipe with 400 words count.

Step 4: After clicking the Generate Recipe button, in fraction of seconds a simple joke will be generated as shown in the below figure to engage the users if recipe generation get delayed. The Food Recipe will be generated based on the user inputs as shown in the following four images.

Ingredients: The Foundation of Flavor

For the Cake:

- 2 ¼ cups (270g) All-Purpose Flour
- ¾ cup (75g) Unsweetened Cocoa Powder (Dutch-processed preferred for deeper color)
- 1 ½ teaspoons Baking Soda
- 1 teaspoon Baking Powder
- 1 teaspoon Salt
- 2 cups (400g) Granulated Sugar
- 2 large Eggs
- 1 cup (240ml) Buttermilk (or milk mixed with 1 tbsp vinegar)
- ½ cup (120ml) Vegetable Oil
- 2 teaspoons Vanilla Extract
- 1 cup (240ml) Hot Coffee (or hot water)

For the Simple Chocolate Ganache Frosting (Optional but Recommended):

- 1 cup Heavy Cream
- 8 oz Semi-Sweet Chocolate, finely chopped

Step-by-Step Instructions: Baking Perfection

1. Preheat and Prep: Preheat your oven to 350°F (175°C). Grease and flour two 8-inch round cake pans, or line the bottoms with parchment paper.

2. Combine Dry Ingredients: In a large bowl, whisk together the flour, cocoa powder, baking soda, baking powder, salt, and sugar until fully combined.

3. Mix Wet Ingredients: In a separate medium bowl, whisk together the eggs, buttermilk, oil, and vanilla extract.

4. Combine: Pour the wet ingredients into the dry ingredients and mix on low speed (or by hand) until just combined. Do not overmix!

5. The Secret Ingredient: Carefully pour the hot coffee (or hot water) into the batter. The batter will be very thin—this is exactly what you want! Mix until smooth.

6. Bake: Divide the batter evenly between the prepared pans. Bake for 30–35 minutes, or until a toothpick inserted into the center comes out clean.

7. Cool: Let the cakes cool in the pans for 10 minutes before inverting them onto a wire rack to cool completely before frosting.

8. Make the Ganache: While the cakes cool, heat the heavy cream until it just begins to simmer. Pour the hot cream over the chopped chocolate in a bowl. Let it sit for 5 minutes, then whisk until smooth and glossy. Allow it to cool and thicken slightly before spreading.

8. Make the Ganache: While the cakes cool, heat the heavy cream until it just begins to simmer. Pour the hot cream over the chopped chocolate in a bowl. Let it sit for 5 minutes, then whisk until smooth and glossy. Allow it to cool and thicken slightly before spreading.

Baker's Tips for Success

- **Why Hot Coffee?** The heat blooms the cocoa powder, intensifying the chocolate flavor without making the cake taste like coffee.
- **Room Temperature Matters:** Ensure your eggs and buttermilk are at room temperature. This helps them emulsify easily, creating a lighter texture.
- **Don't Overbake:** Chocolate cakes dry out quickly. Start checking at the 30-minute mark.

Serving Suggestions

8. ADVANTAGES AND DISADVANTAGES

Advantages

- Saves time by automatically generating detailed recipe blogs
- Reduces manual effort for food bloggers and users
- Generates customized content based on user input
- User-friendly interface built using Streamlit
- Fast and efficient content generation using a pre-trained AI model
- No requirement for dataset collection or model training

Disadvantages

- Requires an active internet connection to access the AI model
- Depends on third-party AI APIs for content generation
- Limited to text-based recipe content only
- Output quality depends on the clarity of user input

9. CONCLUSION

The Flavour Fusion: AI-Driven Recipe Blogging project successfully demonstrates how generative AI can be used to automate recipe blog creation. The application allows users to generate customized and well-structured recipe content by providing a topic and word count, reducing the time and effort required for manual writing. By integrating a pre-trained AI model with a user-friendly Streamlit interface, the project delivers fast, efficient, and high-quality results, making it a useful tool for food bloggers and cooking enthusiasts.

10. FUTURE SCOPE

The Flavour Fusion project can be enhanced further by adding support for multiple languages to reach a wider audience. Future improvements may include generating recipe images along with text, adding user accounts to save favorite recipes, and providing personalized recipe recommendations. The application can also be extended to support voice-based input and mobile platform deployment, making it more accessible and user-friendly.

11. APPENDIX

11.1. Source Code

The source code for the Flavour Fusion: AI-Driven Recipe Blogging project includes the implementation of the Streamlit user interface, integration of the Gemini Flash Lite model using the Google Generative AI API, recipe blog generation logic, and the programmer joke feature. The code is written in Python and follows a modular and readable structure.

11.2. Github & Project Demo Link

Github Repository Link: https://github.com/Pradeep232823/Flavour_Fusion

Demo Link:

https://drive.google.com/file/d/1G8qmgIvjposxqo5sTo6ar0ZR2qyFgdLL/view?usp=sharing