

Flavour Fusion: AI-Driven Recipe Blogging

1. INTRODUCTION :

1.1 Project Overview

In today's digital era, content creation plays a vital role in blogging, marketing, and online communication. Recipe blogging, in particular, requires creativity, structured formatting, engaging storytelling, and detailed step-by-step explanations. However, manually writing high-quality recipe blogs can be time-consuming and repetitive for content creators.

Flavour Fusion: AI-Driven Recipe Blogging is a web-based application that leverages the power of Generative Artificial Intelligence to automate the creation of recipe blogs. The system integrates Google's Gemini 1.5 Flash large language model (LLM) with a Streamlit-based user interface to generate structured and customizable recipe content based on user input.

The application allows users to enter a recipe topic and specify the desired word count. Using advanced natural language processing and generative capabilities, the Gemini model produces a complete blog including an introduction, ingredients list, step-by-step instructions, serving suggestions, and cooking tips. Additionally, to enhance user engagement, the system displays a light-hearted programmer joke while the AI generates the content.

This project demonstrates the practical implementation of Large Language Models (LLMs) in real-world content automation scenarios and showcases how AI can significantly improve productivity and efficiency in digital content creation.

1.2 Purpose

The primary purpose of this project is to design and develop an AI-powered web application capable of generating structured and engaging recipe blogs automatically.

The specific objectives of this project include:

- To integrate Google's Gemini 1.5 Flash model with a web application.
- To allow users to generate customized recipe blogs based on topic and word count.
- To demonstrate the practical usage of Large Language Models (LLMs).
- To reduce the time and effort required for manual blog writing.
- To provide a user-friendly interface for AI-driven content creation.
- To explore prompt engineering techniques for structured output generation.

This project also aims to bridge the gap between theoretical understanding of Generative AI and practical implementation by developing a functional application that utilizes API-based model interaction.

2. IDEATION PHASE:

2.1 Problem Statement

Content creators, especially food bloggers, often spend considerable time drafting detailed and well-structured recipe blogs. Writing such blogs involves multiple components, including:

- Crafting an engaging introduction
- Listing accurate ingredients
- Explaining cooking steps clearly
- Adding serving suggestions
- Including useful tips

The process can be repetitive, time-consuming, and sometimes creatively exhausting. Additionally, maintaining consistency in tone, structure, and quality across multiple blog posts can be challenging.

With the rapid advancement of Artificial Intelligence and Generative AI technologies, there exists an opportunity to automate content creation tasks efficiently. However, many users lack access to easy-to-use tools that integrate powerful AI models into simple web interfaces.

Therefore, the problem addressed by this project is:

"How can we develop an AI-powered system that automatically generates structured, high-quality, and customizable recipe blogs while maintaining user engagement and reducing manual effort?"

Flavour Fusion aims to solve this problem by leveraging Gemini 1.5 Flash to generate well-formatted recipe blogs based on user-defined parameters.

3. REQUIREMENT ANALYSIS:

3.1 Customer Journey Map

1. User opens the web app
2. Enters recipe topic
3. Selects word count
4. Clicks “Generate Recipe”
5. Views joke while waiting
6. Receives complete recipe blog
7. Copies or exports content

3.2 Solution Requirement

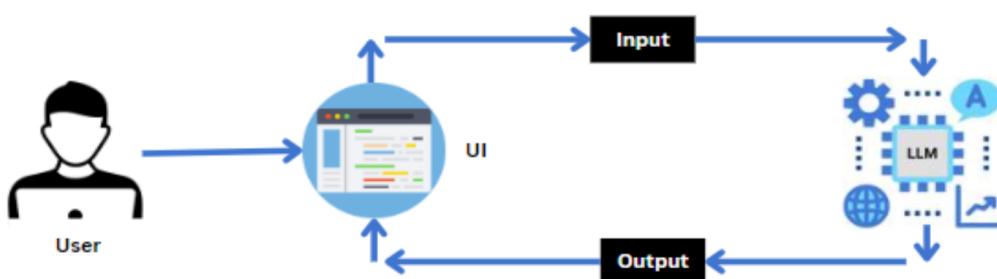
Functional Requirements:

- Accept topic input
- Accept word count input
- Generate recipe blog
- Display joke during generation
- Show structured output

Non-Functional Requirements:

- Fast response time
- Secure API integration
- User-friendly interface
- Scalable model usage

3.3 Data Flow Diagram :



3.4 Technology Stack

Frontend Streamlit

Backend Python

AI Model Gemini 1.5 Flash

API Integration Google Generative AI API

Deployment Streamlit Cloud

4. PROJECT DESIGN:

4.1 Problem Solution Fit

The proposed solution, *Flavour Fusion: AI-Driven Recipe Blogging*, effectively addresses the core problem of time-consuming and repetitive manual content creation in recipe blogging. Traditional blog writing requires significant effort in structuring content, maintaining engagement, organizing ingredients and instructions, and ensuring consistency in tone and formatting. This process can be exhausting, especially for content creators who need to publish blogs frequently.

The implemented solution leverages the power of Google's Gemini 1.5 Flash large language model to automate this entire process. By allowing users to input a recipe topic and specify a desired word count, the system dynamically generates a structured and comprehensive recipe blog in real time. This significantly reduces the time required to create high-quality content while maintaining readability and creativity.

The problem-solution alignment is evident in the following aspects:

1. Time Efficiency:

The application drastically reduces blog drafting time from hours to a few seconds. Users can instantly generate complete recipe blogs without manually researching, organizing, and writing content.

2. Customization Capability:

Users can define the desired word count, allowing flexibility in content length based on blogging requirements. This ensures adaptability for different platforms such as personal blogs, social media posts, or professional websites.

3. Structured Content Generation:

The system ensures that each recipe blog includes essential sections such as introduction, ingredients, preparation steps, serving suggestions, and tips. This structured output enhances content quality and usability.

4. Creativity and Engagement:

Through configurable parameters like temperature and top_p, the model produces diverse and creative outputs. Additionally, the inclusion of a lighthearted programmer joke during content generation enhances user engagement and overall experience.

5. Ease of Use:

The integration with Streamlit provides a simple and intuitive user interface, making advanced AI technology accessible even to non-technical users.

6. Scalability and Adaptability:

Since the system uses API-based AI integration, it can easily be extended to support other types of blog content beyond recipes, making it scalable for future enhancements.

Overall, the solution effectively bridges the gap between manual content creation challenges and modern AI-driven automation. It demonstrates a strong alignment between user needs and technological capabilities, ensuring both practical usability and innovative implementation.

4.2 Proposed Solution

The system integrates Streamlit UI with Gemini 1.5 Flash model.

The model processes user input and generates structured blog content including:

- Introduction
- Ingredients
- Instructions
- Serving Suggestions
- Tips

4.3 Solution Architecture

Architecture Flow:

User → Streamlit Interface → API Call → Gemini 1.5 Flash → Response → UI Display

The generation_config parameters control:

- Creativity (temperature)
- Content quality (top_p, top_k)
- Output length (max_output_tokens)

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Phase	Duration
Requirement Analysis	2 Days
API Setup	1 Day
Model Integration	2 Days
UI Development	2 Days
Testing	1 Day
Deployment	1 Day

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Functional Testing

Test Case 1:

Input: "Malai Kofta" – 800 words

Result: Successfully generated structured recipe

Test Case 2:

Input: Empty topic

Result: Error handled correctly

6.2 Performance Testing

- Response time tested under normal internet conditions
- Average generation time: 3–6 seconds
- No major latency issues observed

7.1 Output Screenshots

(Insert screenshots here)

- Homepage UI
- Input Section
- Joke Display
- Generated Recipe Output

The application successfully generates detailed and well-structured recipe blogs based on user specifications.

8. ADVANTAGES & DISADVANTAGES

Advantages

- ✓ Saves time for bloggers
- ✓ Generates creative content
- ✓ Customizable word count
- ✓ Easy to use interface
- ✓ Real-time AI generation

Disadvantages

- ✗ Requires internet connection
- ✗ Dependent on API usage limits
- ✗ Generated content may require minor editing

9. CONCLUSION

Flavour Fusion demonstrates the practical integration of Generative AI with web applications. The project successfully automates recipe blog generation using Gemini 1.5 Flash.

It highlights the potential of LLMs in content automation and productivity enhancement.

10. FUTURE SCOPE

- Add multi-language support
- Add downloadable PDF feature
- Add image generation for recipes
- Add user authentication system
- Save generated blogs in database
- Deploy as full SaaS platform

11. APPENDIX

Source code

App.py

```
import streamlit as st
from google import genai
import random

# ● REPLACE WITH YOUR NEW API KEY
API_KEY = "AlzaSyBnQULMueUYSofEEKsl77o-m4OW9-ymBfA"
try:
    client = genai.Client(api_key=API_KEY)
except Exception as e:
    st.error(f"Setup Error: {e}")
generation_config = {
    "temperature": 0.75,
    "top_p": 0.95,
    "top_k": 64,
```

```
"max_output_tokens": 8192,  
"response_mime_type": "text/plain",  
}  
  
def get_joke():  
    """Selects and returns a random programming joke."""  
  
    jokes = [  
        "Why don't programmers like nature? It has too many bugs.",  
        "Why do Java developers wear glasses? Because they don't see sharp.",  
        "Why was the JavaScript developer sad? Because he didn't know how to 'null' his feelings.",  
        "Why don't programmers like nature? It has too many bugs.",  
        "Why do programmers prefer dark mode? Because light attracts bugs!",  
        "Why do Java developers wear glasses? Because they don't see sharp.",  
        "Why was the JavaScript developer sad? Because he didn't know how to 'null' his feelings.",  
        "Why do Python programmers prefer using snake_case? Because it's easier to read!",  
        "How many programmers does it take to change a light bulb? None, that's a hardware problem.",  
        "Why did the developer go broke? Because he used up all his cache.",  
        "Why do programmers always mix up Christmas and Halloween? Because Oct 31 == Dec 25.",  
        "Why did the programmer get kicked out of the beach? Because he kept using the 'C' language!",  
        "Why was the computer cold? It left its Windows open."  
    ]  
  
    return random.choice(jokes)  
  
def recipe_generation(user_input, word_count):  
    """  
    Function to generate a blog based on user input and word count.  
    """
```

```
# Display a message while the blog is being generated
st.info("### 🍽️ Generating your recipe...")
st.write(f"While I work on creating your blog, here's a little joke to keep you entertained:\n\n**{get_joke()}**")

try:
    # Prompt logic from Source 84
    prompt = f"Write a recipe based on the input topic: {user_input} and number of words: {word_count}\n"
    # Using Gemini 1.5 Flash as requested in Source 23
    response = client.models.generate_content(
        model="gemini-2.5-flash",
        contents=prompt,
        config=generation_config
    )
    st.success("🎉 Your recipe is ready!")
    return response.text

except Exception as e:
    st.error(f"Error generating blog: {e}")
    return None

st.title("Flavour Fusion: AI-Driven Recipe Blogging")
topic = st.text_input("Topic")
words = st.number_input("Number of words", min_value=100, max_value=2000, step=100, value=300)
if st.button("Generate recipe"):
    if not API_KEY or API_KEY == "YOUR_API_KEY_HERE":
        st.error("Please provide a valid API Key in the code.")
    elif topic:
```

```
result = recipe_generation(topic, words)

if result:

    st.markdown("### 🍔 Recipe Blog:")

    st.write(result)

else:

    st.warning("Please enter a topic")
```

OUTPUT

Flavour Fusion: AI-Driven Recipe Blogging

Topic

mafia kotla

Number of words

555

- +

Generate recipe

 Generating your recipe...

While I work on creating your blog, here's a little joke to keep you entertained:

Why did the developer go broke? Because he used up all his cache.

 Your recipe is ready!

Recipe Blog:

The Kotla Don's Midnight Masala Chicken

In the shadowy lanes where history whispers tales of old Delhi, and modern ambitions intertwine, lies the legend of 'Mafia Kotla.' Not a place on any map, but a culinary concept – a dish that embodies the clandestine richness, the deep-seated secrets, and the bold, unforgettable flavors of a world unseen. This is not just a chicken curry; it's a testament to indulgence, a masterpiece of slow-cooked complexity, designed for those who appreciate the finer, darker side of flavor. Prepare to be initiated into the world of The Kotla Don's Midnight Masala Chicken.

This recipe delivers a deeply aromatic and darkly rich chicken curry, where traditional Indian spices meet unexpected notes for an unforgettable experience.

Yields: 4-6 servings **Prep time:** 30 minutes (plus 1-4 hours marination) **Cook time:** 60-75 minutes

Ingredients:

For the Chicken & Marinade:

- 1 kg bone-in chicken pieces (thighs and drumsticks preferred)
- ½ cup thick plain yogurt
- 2 tablespoons ginger-garlic paste

- 1 teaspoon turmeric powder
- 1 teaspoon Kashmiri red chili powder (for color, less heat)
- 1 teaspoon salt
- 1 tablespoon fresh lemon juice

For the Midnight Masala Gravy:

- 4 tablespoons ghee or neutral oil
- 3 large onions, thinly sliced
- 2 black cardamoms
- 3 green cardamoms
- 1-inch cinnamon stick
- 5-6 cloves
- 2 bay leaves
- 1 teaspoon cumin seeds
- 2 tablespoons ginger-garlic paste
- 2 large tomatoes, pureed (or 1/2 cup canned crushed tomatoes)
- 1 tablespoon coriander powder
- 1 teaspoon cumin powder
- 1 teaspoon regular red chili powder (adjust to your spice preference)
- ½ teaspoon turmeric powder
- 1 teaspoon garam masala
- ¼ cup cashews, soaked in warm water for 15 minutes (or 2 tbsp cashew paste)
- ½ teaspoon instant coffee powder (a subtle secret for depth)
- 1 small square (10g) dark chocolate (70% cocoa or higher, for richness and color)
- ½ cup hot water or chicken stock

- Salt to taste
- Optional: $\frac{1}{4}$ cup full-fat cream, for extra richness at the end

For Garnish:

- Fresh coriander leaves, chopped
- Slivered almonds (optional)
- A drizzle of fresh cream (optional)

Instructions:

1. **Marinate the Chicken:** In a large bowl, combine the chicken pieces with yogurt, ginger-garlic paste, turmeric, Kashmiri red chili powder, salt, and lemon juice. Mix thoroughly, ensuring every piece is coated. Cover the bowl and refrigerate for at least 1 hour, preferably 4 hours or even overnight, allowing the flavors to deeply penetrate.
2. **Caramelize the Onions (The Foundation of Flavor):** Heat the ghee or oil in a heavy-bottomed pot or Dutch oven over medium heat. Add the thinly sliced onions and cook slowly, stirring frequently. This step is crucial: allow them to turn a deep golden brown and almost caramelized, which will take 20-25 minutes. This patient process develops the profound sweetness and dark color essential for the gravy. Once done, remove half of the caramelized onions and set aside for later.
3. **Infuse with Aromatics:** To the remaining onions in the pot, add the whole spices: black cardamom, green cardamoms, cinnamon stick, cloves, bay leaves, and cumin seeds. Sauté for about 30 seconds until they become fragrant.

4. **Build the Masala Base:** Add the second batch of ginger-garlic paste to the pot and sauté for 1-2 minutes until its raw aroma disappears. Stir in the tomato puree and cook, stirring occasionally, until the oil separates from the mixture (approximately 5-7 minutes).
5. **Introduce Dry Spices & Cashews:** Lower the heat. Add coriander powder, cumin powder, regular red chili powder, turmeric powder, and salt. Cook for 2 minutes, adding a splash of water if the masala begins to stick. Meanwhile, blend the soaked cashews with a little water to form a smooth, thick paste. Stir this cashew paste into the pot and cook for another 3-4 minutes, stirring constantly to prevent it from sticking to the bottom.
6. **Unveil the Don's Secrets:** Now, stir in the instant coffee powder and the dark chocolate. These unconventional additions are the 'secret syndicate' behind this dish's unique depth, enhancing the color and adding a subtle, complex bitterness that elevates the richness without overpowering the traditional Indian spices.
7. **Add Chicken & Slow Cook:** Add the marinated chicken pieces to the pot. Mix well, ensuring the chicken is thoroughly coated with the rich masala. Pour in the hot water or chicken stock. Bring the mixture to a gentle simmer, then cover the pot, reduce the heat to low, and cook for 30-40 minutes, or until the chicken is tender and cooked through, stirring occasionally to prevent sticking.
8. **Finishing Touches:** Stir in the garam masala and the reserved caramelized onions. Cook for another 5 minutes, allowing the flavors to meld beautifully. If desired, stir in the full-fat cream at this stage for an extra layer of indulgence. Taste and adjust salt if needed.
9. **Serve:** Garnish generously with fresh chopped coriander leaves and slivered almonds (if using).

Serving Suggestion:

Serve this regal 'Kotla Don's Midnight Masala Chicken' hot with fluffy basmati rice, warm garlic naan, or flaky parathas. Pair it with a simple cucumber raita to balance the richness, and let your guests uncover the layers of flavor that make this dish truly legendary.

Chef's Note:

The slow caramelization of onions and the strategic additions of coffee and dark chocolate are the 'secret syndicate' behind this dish's unique depth and enigmatic character. Don't rush these steps; they are key to achieving the dark, powerful essence of the Kotla Don's signature meal. For an even smokier note, you can briefly smoke the finished curry using the traditional dhungar method (placing a small bowl with a hot, smoking charcoal piece in the pot, drizzling with ghee, and covering for a few minutes) before serving.

Github Link:

<https://github.com/RupaDeviBeella/Flavour-Fusion-AI-Driven-Recipe-Blogging>

Video Demo Link:

https://drive.google.com/file/d/1xJBi2rC3favFBnp2akEcrNxjHapdgLyQ/view?usp=drive_link