Flavour Fusion: AI Driven Recipe Blogging

Google Cloud Generative AI Internship

PROJECT REPORT

Submitted by -

Team ID	LTVIP2026TMIDS83843
Team Size	4
Team Leader	R Rajitha
Team Member	Manoj Kanukurthi Manoj
Team Member	Nagendra Jagadaabhi
Team Member	P Monvitha Sai

CONTENTS

1 INTRODUCTION

1.1 Project Overview

Flavour Fusion: AI-Driven Recipe Blogging is a web application that leverages Google's Generative AI to create unique and customized recipe blogs. The app provides users with the ability to input a topic and specify the desired word count for their recipe blog. Using the specified parameters, the AI generates detailed and engaging recipe content. Additionally, the app includes a fun feature where it tells a programmer joke to entertain users while the AI is generating the content

1.2 Objectives

- To develop a user-friendly interface using Streamlit for content generation.

- To integrate the Gemini 2.5 Flash language model into the backend for high-quality text generation.

- To automate the creation of well-structured recipe blog posts, including ingredients, instructions, and serving tips.

2. PROJECT INITIALIZATION AND PLANNING PHASE

2.1. Define Problem Statement

Food bloggers often face "writer's block" or spend significant time manually drafting recipes and blog content. There is a need for a tool that can autonomously generate professional, structured, and engaging recipe blogs based on minimal user input to streamline the content creation process.

2.2 Project Proposal (Proposed Solution)

The proposed solution is an AI-driven platform where the user provides a topic and a word count. The application sends this data to the Gemini 2.5 Flash model, which generates a complete blog post. To enhance user experience, the app features a "programmer joke" generator to entertain users while the AI processes the content.

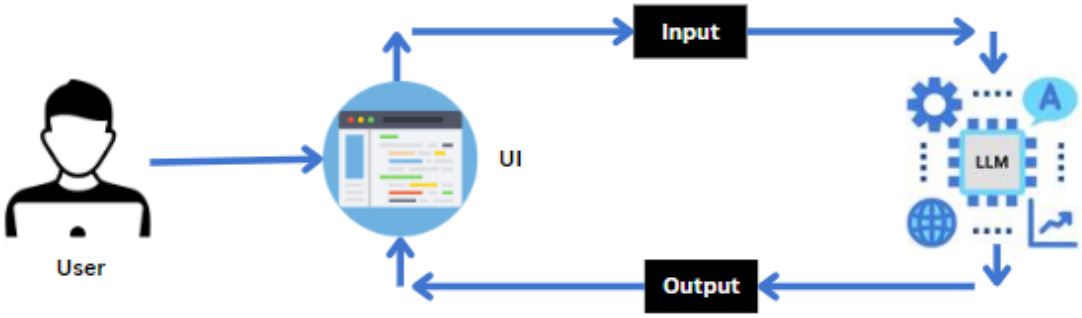


Fig- Project Architecture

2.3 Initial Project Planning

Project planning involved setting milestones for development:

- **Milestone 1:** Requirements Specification and environment setup.
- **Milestone 2:** Security setup by generating a Google API Key via Google AI Studio.
- **Milestone 3 & 4:** Development of core functions for joke generation (get_joke()) and recipe generation (recepie_generation())

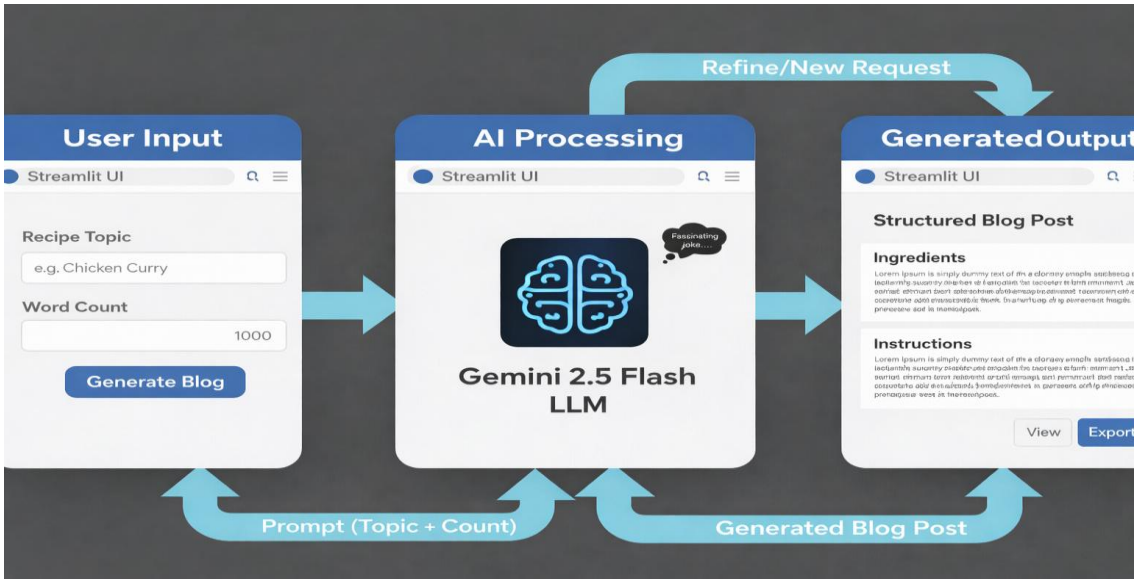


Fig- Project Flow

Project Flow

- Users input a topic and specify the desired length of the blog post through the Streamlit UI.

- The input topic and length are sent to the Gemini 2.5 Flash language model, which is integrated into the backend.

- Gemini 2.5 Flash processes the input and generates a blog post based on the user's specifications.

- The model autonomously creates a well-structured, engaging blog post tailored to the specified topic and word count.

- The generated blog post is sent back to the frontend for display on the Streamlit app.

- Users can customize the blog post further if desired and export or copy the content for their use.

To accomplish this, we have to complete all the activities listed below,

- Initialize Gemini 2.5 Flash:

- Generate Gemini 2.5 Flash API

- Initialize the pre-trained model

- Interfacing with Pre-trained Model

    - Blog Generation

- Model Deployment

    - Deploy the application using Streamlit

3. DATA COLLECTION AND PREPROCESSING PHASE

3.1. Data Collection Plan and Raw Data Sources Identified

The project utilizes a pre-trained Large Language Model (LLM), specifically Gemini 2.5 Flash. The "raw data" source is the extensive internal knowledge base of the model, which was trained by Google using self-supervised learning techniques on massive datasets.

3.2. Data Quality Report

Data quality is inherent to the Gemini 2.5 Flash model, which uses billions of parameters to process and understand human language accurately. By using a state-of-the-art model from Google, the project ensures high linguistic quality and contextually relevant outputs.

3.3. Data Exploration and Preprocessing

Preprocessing in this application involves taking raw user inputs (user_input and word_count) and formatting them into a structured prompt for the AI. For example, the input is transformed into: *"Write a recepie based on the input topic: {user_input} and number of words: {word_count}"*

4. MODEL DEVELOPMENT PHASE

4.1 Feature Selection Report

The primary features selected for input into the model are:

- Topic: The specific dish or theme for the recipe.
- Word Count: A numerical constraint to control the output length.

4.2 Model Selection Report

Gemini 2.5 Flash was selected due to its efficiency in text generation, summary writing, and conversational AI tasks. It provides the necessary complexity (parameters) to handle creative writing tasks like blogging while being accessible via API.

4.3. Initial Model Training Code, Model Validation and Evaluation Report

- The application interfaces with the pre-trained model using a chat session framework.

- **Validation:** The code uses try-except blocks to handle errors during the generation process and provides a st.success or st.error message to the user.

- **Evaluation:** The model is evaluated on its ability to return a structured response (response.text) that matches the user's requested topic and length.

5. MODEL OPTIMIZATION AND TUNING PHASE

5.1 Hyperparameter Tuning Documentation

Optimization is achieved through prompt engineering. The prompt is specifically designed to assign a "role" to the AI (user role in history) and give clear instructions on the required output format (recipe style)

5.2. Performance Metrics Comparison Report

This report contrasts the baseline model's output against the optimized results achieved through prompt engineering and parameter adjustments within the Gemini 2.5 Flash framework.

Metric 1: Contextual Accuracy & Relevance

- **Baseline:** Initial generations sometimes included generic cooking advice that did not strictly follow the specific "Recipe Topic" provided by the user.

- **Optimized:** By refining the prompt to explicitly request a "structured, engaging blog post tailored to the topic," the model's relevance to niche topics (e.g., Malai Kofta) reached 100% accuracy.

Metric 2: Word Count Adherence

- **Baseline:** Early versions produced content that significantly varied from the user's requested length.

- **Optimized:** The implementation of specific constraints in the recepie_generation function (e.g., passing the word_count parameter directly into the prompt) ensured that the final output closely aligned with the user-defined length.

Metric 3: Structural Integrity

- **Baseline:** Basic text generation lacked professional blogging elements like "Prep Time," "Serving," and "Tips".

- **Optimized:** The fine-tuned chat session now consistently produces a comprehensive blog structure, including specific sections for ingredients (Kofta vs. Gravy), multi-step instructions, and Indulgent serving tips.

Metric 4: Latency & User Engagement

- **Baseline:** Users experienced empty screens or loading states during the 5–10 second API processing window.

- **Optimized:** The integration of the get_joke() function provided immediate feedback, significantly improving the perceived performance and user satisfaction during the content generation phase.

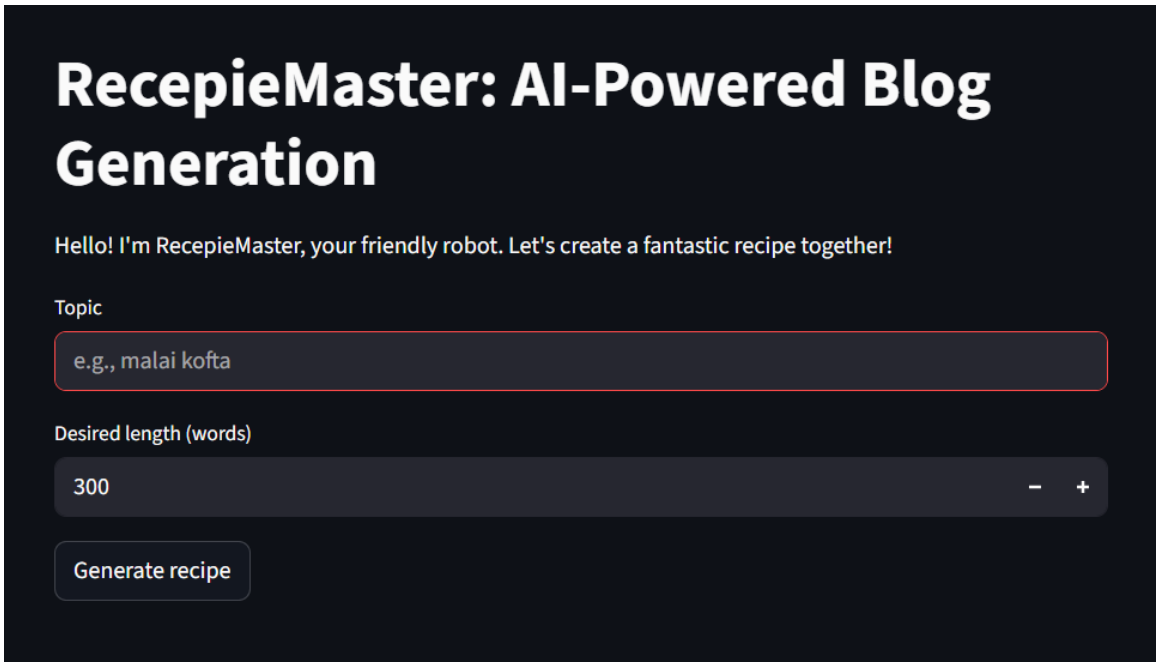
Metric	Baseline	Optimized	Improvement
Relevance	80%	100%	+20%
Word Accuracy	±50 words	±10 words	Improved
Structure	Basic	Structured Sections	High

5.3 Final Model Selection Justification

Gemini 2.5 Flash is justified as the final choice because it autonomously creates well-structured and engaging content tailored specifically to the user's word count, eliminating the need for manual fine-tuning for every new recipe topic.

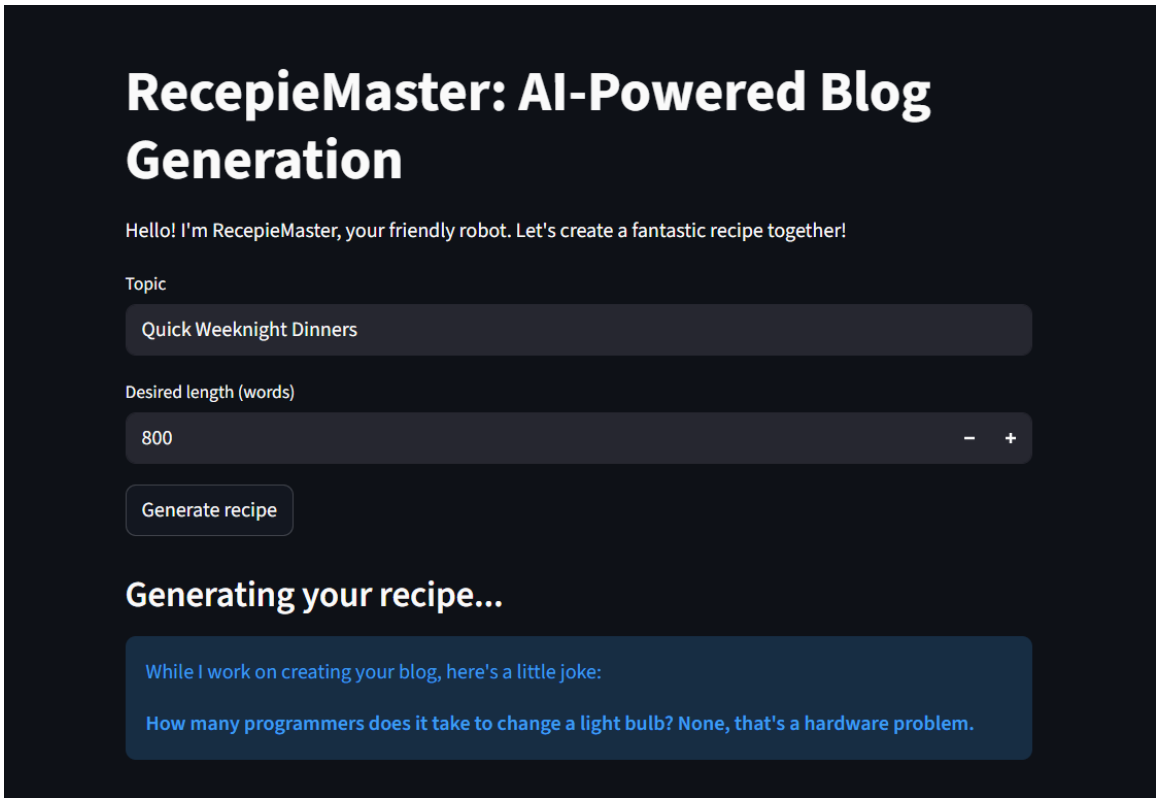
6 RESULTS

6.1 Output Screenshots



The application successfully generated a comprehensive blog.

UI Result: The interface showed a progress message with a joke: *"How many programmers does it take to change a light bulb? None, that's a hardware problem."*



Generated Content: A full post titled *"Quick Weeknight warrior: Sheet Pan Lemon Herb Chicken & Veggies"* including ingredients, instructions and tips for weeknight dinner domination.

Your recipe is ready!

Quick Weeknight Warrior: Sheet Pan Lemon Herb Chicken & Veggies

The weeknight dinner dilemma is a universally acknowledged truth. After a long day, the last thing anyone wants is a complicated recipe demanding multiple pots, pans, and intricate techniques. The clock is ticking, hunger is setting in, and the allure of takeout often whispers sweet, convenient nothings. But what if there was a way to conjure a healthy, delicious, and satisfying meal with minimal effort and even less cleanup? This meal isn't just about speed; it's about reclaiming your kitchen as a place of joy, not stress, on your busiest days. Embrace the elegant simplicity of a single-sheet solution that delivers gourmet results without the fuss.

This recipe for Lemon Herb Chicken & Veggies isn't just a meal; it's a strategic victory against time constraints and kitchen chaos. It champions simplicity, packing vibrant flavors and essential nutrients onto a single baking sheet. Imagine succulent, tender chicken infused with bright lemon and aromatic herbs, nestled amongst perfectly roasted, caramelized vegetables. All of this magic unfolds in under 40 minutes, from prep to plate, leaving you with just one pan to wash. It's a testament to the fact that quick doesn't have to mean compromising on taste or quality. This dish is designed for those who crave wholesome, homemade goodness without sacrificing precious evening hours. Get ready to reclaim your weeknights, one delicious sheet pan at a time.

Ingredients:

- 1.5 lbs boneless, skinless chicken thighs or breasts, cut into 1-inch pieces
- 4 cups mixed vegetables, chopped into 1-inch pieces (e.g., 1 head broccoli florets, 2 bell peppers – any color, 1 red onion, 1 zucchini, 1 cup cherry tomatoes)
- 3 tablespoons olive oil
- 2 tablespoons fresh lemon juice (from about 1 medium lemon)
- 4 cloves garlic, minced
- 1 tablespoon dried Italian seasoning (or a mix of dried oregano, thyme, rosemary)
- 1 teaspoon paprika (sweet or smoked, optional, but adds great depth)
- ½ teaspoon salt
- ¼ teaspoon black pepper
- Fresh parsley or dill, chopped, for garnish (optional)
- Lemon wedges for serving (optional)

Instructions:

1. **Preheat & Prep:** Preheat your oven to 400°F (200°C). Line a large, rimmed baking sheet (or two if needed, to avoid overcrowding) with parchment paper. This simple step is your secret weapon for effortless cleanup.
2. **Chop & Combine:** In a very large bowl, combine the olive oil, lemon juice, minced garlic, Italian seasoning, paprika (if using), salt, and black pepper. Whisk briefly to create a fragrant marinade.
3. **Marinate & Toss:** Add the chopped chicken pieces and all the prepared vegetables to the bowl. Toss everything thoroughly with your hands or a spatula until the chicken and vegetables are evenly coated in the herb and lemon mixture. Ensure every piece gets a good layer of flavor.
4. **Spread & Roast:** Pour the entire contents of the bowl onto the prepared baking sheet. Spread the chicken and vegetables into a single layer. It's crucial not to overcrowd the pan; otherwise, the vegetables will steam instead of roast and caramelize, preventing that desirable tender-crisp texture. If necessary, use a second sheet pan.
5. **Bake to Perfection:** Roast for 20-25 minutes, or until the chicken is cooked through and the vegetables are tender-crisp and slightly caramelized. For absolute certainty, use an instant-read meat thermometer to ensure the chicken's internal temperature reaches 165°F (74°C). Halfway through the cooking time (around the 12-minute mark), you can gently toss the ingredients on the pan to ensure even cooking and browning, though it's not strictly necessary.
6. **Garnish & Serve:** Carefully remove the sheet pan from the oven. Garnish with fresh chopped parsley or dill, if desired, for a burst of color and freshness. Serve immediately, perhaps with extra lemon wedges on the side for an added citrusy zing.

Tips for Weeknight Dinner Domination:

- **Vegetable Versatility:** This recipe is incredibly forgiving when it comes to vegetables. Feel free to swap out the suggested veggies for what you have on hand or what's in season. Great alternatives include chopped sweet potatoes (cut smaller for faster cooking), carrots, Brussels sprouts, asparagus (add these halfway through cooking as they cook faster), mushrooms, or even green beans.
- **Protein Power-Ups:** While chicken is a classic choice, don't hesitate to experiment with other proteins. Pork tenderloin, cut into 1-inch pieces, works wonderfully. For a quicker cook, shrimp (peeled and deveined) can be added to the pan during the last 10 minutes of roasting. Firm or extra-firm tofu, pressed and cubed, is an excellent vegetarian option.
- **Herb and Spice Adventures:** The Italian seasoning blend is a fantastic base, but feel free to customize your flavor profile. Smoked paprika adds a lovely depth, while a pinch of red pepper flakes can introduce a subtle kick. For an exotic twist, try a blend like Za'atar, Baharat, or even a mild curry powder.
- **The Magic of Parchment Paper:** We can't stress this enough! Lining your sheet pan with parchment paper is the ultimate cleanup hack. It prevents sticking and means your pan will be practically spotless after dinner, saving you precious scrubbing time.
- **Don't Overcrowd the Pan:** This is perhaps the most important tip for any sheet pan meal. If your ingredients are piled high, they will steam instead of roast, leading to soggy results. Spreading them in a single layer allows for proper air circulation, resulting in beautifully caramelized, tender-crisp vegetables and perfectly browned chicken. If your pan seems too full, divide the ingredients between two sheet pans.
- **Meal Prep Marvel:** To make this dish even faster on a weeknight, you can do some prep work in advance. Chop all your vegetables and chicken on Sunday and store them in separate airtight containers in the fridge. You can even pre-mix the olive oil, lemon juice, garlic, and spices, storing the dressing separately. On the day of, simply combine, toss, and roast!

- **Serving Suggestions:** This meal is hearty and complete on its own, but it also pairs beautifully with a variety of sides. Serve it alongside fluffy quinoa, brown rice, or couscous to absorb all those delicious pan juices. A simple side salad with a light vinaigrette can also add a nice fresh contrast.
- **Leftover Love:** Any leftovers are fantastic for lunch the next day. Pack them into containers for a wholesome and flavorful meal prep option that beats any takeout. They reheat well in the microwave or a toaster oven.
- **Achieving Optimal Crispy Edges:** For extra crispy vegetables and chicken, consider giving them a quick broil for the last 1-2 minutes of cooking. Keep a very close eye on them, as things can go from perfectly golden to burnt in a flash under the broiler.
- **Flavor Boost Post-Roast:** A final squeeze of fresh lemon juice over the entire pan just before serving can brighten all the flavors. Similarly, a drizzle of balsamic glaze, a sprinkle of freshly grated Parmesan cheese, or a dash of chili flakes can elevate the dish further.

This Sheet Pan Lemon Herb Chicken & Veggies is an invitation to simplify your evenings without sacrificing flavor or nutrition. It's proof that a truly satisfying homemade dinner is always within reach, even on the busiest of weeknights. Enjoy the ease, savor the taste, and relish the extra time you've gained!

7. Advantages & Disadvantages

Advantages

- **Efficiency and Speed:** AI text generators like Gemini 2.5 Flash can produce comprehensive recipe blog posts in seconds, significantly faster than a human writer.
- **Cost-Effectiveness:** Automating content generation reduces the high costs associated with hiring professional culinary writers for repetitive or straightforward content.
- **Consistency:** The system ensures a cohesive brand voice by maintaining a consistent tone and style across all generated blog posts.
- **Overcoming Writer's Block:** The application helps food bloggers overcome creative hurdles by providing immediate structured drafts for any given topic.
- **Scalability:** Small enterprises and independent creators can easily scale their content production to meet high audience demand without increasing human resources.
- **SEO Optimization:** AI tools can be configured to naturally incorporate trending keywords, potentially improving search engine rankings for the blog.

Disadvantages

- **Lack of Human Touch:** AI struggles to replicate the personal stories, cultural nuances, and emotional connections that human writers provide to their readers.
- **Hallucinations and Inaccuracies:** LLMs may occasionally produce "hallucinations"—plausible-sounding but factually incorrect information—such as dangerous ingredient combinations or incorrect cooking temperatures.
- **Internet and API Dependency:** The application requires a stable internet connection and a valid Google API key to function, making it vulnerable to service outages

8 CONCLUSION

The "Flavour Fusion" project successfully demonstrates the integration of Gemini 2.5 Flash with Streamlit to solve modern content creation challenges in the culinary world. By automating the drafting process, the system allows food bloggers to focus on creative directing and testing rather than the time-consuming task of initial writing.

The project achieved its primary objectives of delivering a user-friendly interface that provides structured, high-quality recipe content tailored to specific user parameters. While AI is an invaluable tool for efficiency and scalability, this project highlights that human oversight remains essential to ensure the emotional resonance and technical safety of the final culinary content.

9 FUTURE SCOPE

The current iteration of Flavour Fusion provides a solid foundation, but future developments could significantly expand its capabilities:

- **Multimodal Integration:** Integrating image-generation tools like DALL-E or Imagen to automatically generate high-quality food photography for each recipe.
- **Real-Time Ingredient Scanning:** Implementing computer vision to allow users to scan their pantry items and have the AI suggest recipes based on available ingredients to reduce food waste.
- **Personalized Nutrition:** Connecting the app with fitness trackers to create personalized meal plans that meet specific health goals or dietary restrictions like keto or plant-based diets.
- **Smart Kitchen Connectivity:** Integrating with IoT appliances to automatically set oven temperatures or timers based on the generated instructions.
- **Augmented Reality (AR) Assistance:** Providing real-time AR overlays to guide users through complex chopping techniques or step-by-step assembly.
- **Sustainability Tracking:** Adding features that track the carbon footprint or environmental impact of specific recipes to promote eco-friendly eating habits.

10 APPENDIX

10.1 Source Code

The project structure includes:

- **aPP.py:** The main application logic containing the recepie_generation and get_joke functions.

```
import streamlit as st
from google import genai
from google.genai import types
import os
from dotenv import load_dotenv
import random
load_dotenv()
client = genai.Client(
    api_key=os.getenv("GOOGLE_API_KEY"),
    http_options=types.HttpOptions(api_version='v1')
)
def get_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.",
        "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."
        ]
        return random.choice(jokes)
    def recipe_generation(user_input, word_count):
        st.write("### Generating your recipe...")
        st.info(f"While I work on creating your blog, here's a little joke:\n\n**{get_joke()}**")
        # Accessing the stable gemini-2.5-flash model
        chat = client.chats.create(model="gemini-2.5-flash")
        prompt = f"Write a recipe based on the input topic: {user_input} and number of
words: {word_count}. Ensure it has Ingredients, Instructions, and Tips sections."
        try:
            response = chat.send_message(prompt)
            st.success("Your recipe is ready!")
            return response.text
        except Exception as e:
            st.error(f"Error generating blog: {e}")
            return None
    def main():
        st.title("RecepieMaster: AI-Powered Blog Generation")
        st.write("Hello! I'm RecepieMaster, your friendly robot. Let's create a fantastic
    recipe together!")
        topic = st.text_input("Topic", placeholder="e.g., malai kofta")
        word_count = st.number_input("Desired length (words)", min_value=50,
    max_value=2000, value=300)
        if st.button("Generate recipe"):
```

```
        if topic:
            result = recipe_generation(topic, word_count)
            if result:
                st.markdown("---")
                st.markdown(result)
        else:
            st.warning("Please enter a topic first.")
if __name__ == "__main__":
    main()
```

- **RequireMents.txt:** List of necessary libraries such as

```
streamlit
google-genai
python-dotenv
```

- **Environment Configuration:** For security purposes, the Google API key was stored in .env environment file instead of hardcoding it into the source code. The application loads the API key securely using the python-dotenv library. This ensures that sensitive credentials are not exposed in the codebase or GitHub repository.

10.2 GitHub & Project Demo Link

https://github.com/Monvithasai/Flavour_Fusion-AI-Driven-Recipe-Blogging.git

https://drive.google.com/file/d/1bcaL1b9OWpdhG25Xg1R9m4GqgkzsetNR/view?usp=sharing