Full Stack Development with MERN

Project Documentation

1. Introduction

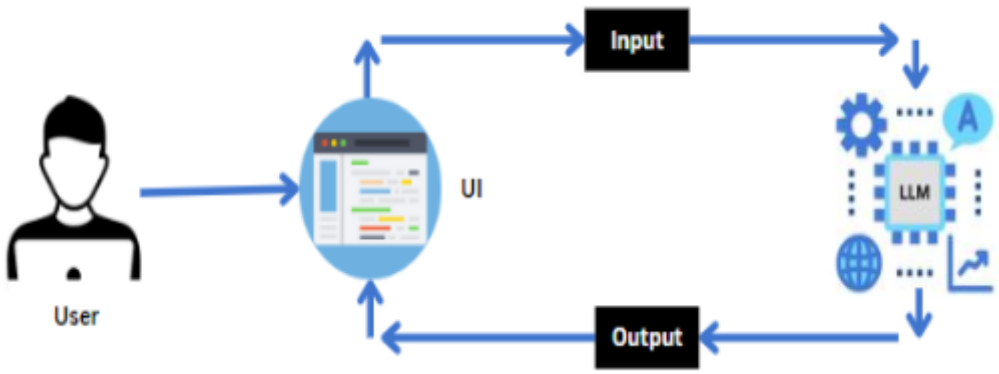
• **Project Title: Advancing Nutrition Science through GeminiAI**

• **Team Members:** Nagari Shaik Mohammad Fayak(22HM1A3326)
S Mokshagna
Shaik Rajak
Mannuru Sree Lakshmisai

2. Project Overview

• **Purpose:**. **Advancing Nutrition Science through Gemini AI** is a web-based intelligent nutrition assistant developed using **Streamlit and Google Gemini (Generative AI)**. The project aims to enhance personalized dietary planning, nutritional awareness, and wellness guidance through AI-powered solutions.

3. Architecture



4. Setup Instructions

Prerequisites

- **Python 3.9 or above**
- **pip (comes with Python)**
- **Internet connection**

- **Required libraries:**
    - **streamlit**
    - **pandas**
    - **google-generativeai**
- ---

- # Installation Steps

- 1️⃣ Clone the project
- `git clone https://github.com/your-username/Advancing-Nutrition-Science-GeminiAI.git`
- `cd Advancing-Nutrition-Science-GeminiAI`
- 2️⃣ Install dependencies
- `pip install -r requirements.txt`
- 3️⃣ Add Gemini API key in `.env` file
- `GEMINI_API_KEY=your_api_key_here`
- 4️⃣ Run the application
- `streamlit run app.py`
- The app will open in your browser automatically.

5. Folder Structure

• Client (Streamlit Frontend)

The frontend of the project is developed using Streamlit, which handles the user interface and user interactions.

• Server (Backend Logic – Python + Gemini API)

The backend logic is handled using Python and integrated with Google Gemini API for AI-based responses.

6. Running the Application

- **Steps to Run Locally**

1. **Open terminal inside the project folder**
2. **Install dependencies (if not already installed):**

3. **pip install -r requirements.txt**
4. **Run the application:**

streamlit run app.py

The application will automatically open in your browser at:

http://localhost:8501

7. API Documentation

response = model.generate_content("Create a healthy meal plan for 2000 calories")

8. Authentication

◈ API Security

- **Gemini API key is stored securely in a .env file.**
- **Environment variables prevent exposure of API credentials.**
- **No API key is hardcoded in the source code.**

◈ Session Handling

- **Streamlit session state is used to maintain chat history.**
- **No user login or token-based authentication implemented.**

9. User Interface

◈ Key UI Features:

- **Sidebar navigation**

- **Interactive forms for meal planning**
- **Nutritional data tables**
- **Macronutrient bar charts**
- **Virtual AI chat interface**
- **Progress tracking sliders**

10. Testing

🧩 Testing Strategy:

- **Manual functional testing**
- **API response validation**
- **Performance testing (response time measurement)**
- **UI interaction testing**

11. Screenshots or Demo

https://drive.google.com/file/d/1_61qg794b4-SuxJ2e12lReKa-SV3FwbZ/view?usp=drivesdk

12. Known Issues

□ Requires stable internet connection.

□ Depends on Gemini API availability.

13. Future Enhancements

□ User login and authentication system