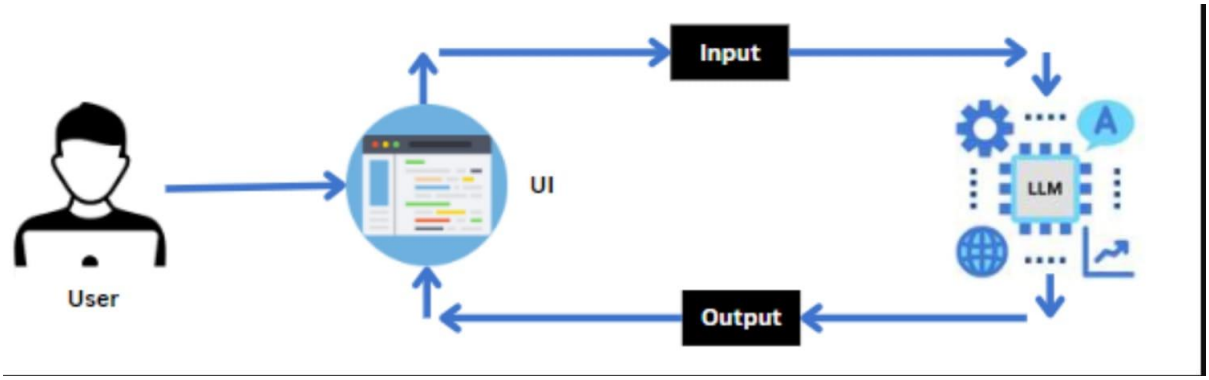
Project Design Phase
Solution Architecture

Date	10-02-2026
Team ID	LTVIP2026TMIDS65437
Project Name	Flavour Fusion: -Ai-Driven Recipe Blogging
Maximum Marks	5 Marks

Solution Architecture:

The solution architecture for Solution Architect for an AI-driven "Flavours Fusion" recipe blogging project bridges the gap between culinary innovation and technical execution. The primary goal is to build a scalable, AI-powered platform that generates, tests, and publishes creative, fusion-style recipes and images, while optimizing for SEO and user engagement.



Here is the proposed architectural solution and strategic roadmap for this project.

1. Project Vision & Core Functionalities

The Platform will automatically generate, visualise, and publish unique fusion recipes (e.g., "Miso Tacos") by blending ingredients, techniques, and culinary cultures.

AI Recipe Generator: Creates unique recipes based on ingredient inputs, dietary constraints, and theme prompts.

AI Visualizer: Generates photorealistic food images from text descriptions or rough sketches.

Automated Blogging Platform: Integrates with CMS (e.g., WordPress) to publish, tag, and optimize recipes for SEO.

"Flavours Fusion" Engine: Leverages Large Language Models (LLMs) to identify complementary tastes from different cuisines.

2. Proposed Technical Architecture

A serverless, API-driven architecture is recommended for high scalability and low maintenance.

- **Frontend/CMS**: WordPress (with SEO plugins like RankMath) or a headless frontend (React/Next.js) for high performance.
- **Automation Engine:** Pabbly Connect or Make.com to connect triggers (e.g., weekly topic) with OpenAI and CMS.

- **AI Brain (LLM):** GPT-4o or Claude 3.5 Sonnet for detailed, creative recipe generation and instruction coherence check.
- **Image Generation:** Midjourney, Stable Diffusion (via API), or SideChef Studio for high-quality, realistic food imagery.
- **Vector Database (RAG):** Pinecone or pgvector to store existing recipes, ensuring the AI generates new ideas rather than repeating known ones.
- **Data Storage**: PostgreSQL for structured data (ingredients, nutritional data, user profiles).

3. Solution Architect Responsibilities

- **Component Interoperability**: Ensuring the AI, Database, and WordPress CMS work seamlessly together.
- **Scaling the Platform:** Planning for increased user traffic and high-volume image generation.
- **Data & AI Security**: Implementing RAG (Retrieval-Augmented Generation) to ground AI in reliable culinary data, reducing hallucinations.
- **Cost Optimization**: Selecting the right APIs (e.g., using GPT-3.5/4o-mini for draft creation, 4o for final polishing) to keep per-recipe costs low.

4. Implementation Roadmap

- **Phase 1: Prototyping (Weeks 1-2):** Set up OpenAI API connection with a "Fusion-Focused" prompt template. Test with manual image generation (Midjourney).
- **Phase 2: Automation (Weeks 3-4):** Use Pabbly/Make to connect the AI output directly to a WordPress draft.
- **Phase 3: RAG Implementation (Weeks 5-6):** Build a vector store of 1000+ top recipes to allow the AI to "understand" flavour combinations and prevent repetitive content.
- **Phase 4: Launch & Optimize (Weeks 7+):** Integrate analytics (Google Analytics) to track which fusion styles gain the most traffic.

5. Key Non-Functional Requirements

- **Performance:** Fast loading times for image-heavy pages.
- **SEO Reliability**: Ensuring generated recipes include SCHEMA markup (Recipe JSON-LD) for rich search results.
- **Maintainability**: Using modular, containerized components (Docker) if bespoke development is used.

Example for Solution Architecture: -

Solution architecture for “Flavour fusion:- Ai-Driven Recipe Blogging”