

Multimodal Hospitality Creator

Course Name: Generative AI

Institution Name: Medicaps University – Datagami Skill Based Course

Student Name(s) & Enrolment Number(s):

Sr no	Student Name	Enrolment Number
01	Irina Khaishagi	EN22CS301435
02	Aakash Ghosh	EN22CS301082
03	Adarsh Dibey	EN22CS301037
04	Aman Gour	EN22CS301104
05	Aastha Gupta	EN22CS301019
06	Aeshna Preyasi	EN22CS301075

Group Name: Group 08D1

Project Number: GAI-08

Industry Mentor Name: Aashruti Shah

University Mentor Name: Prof. Ajaj Khan

Academic Year: 2025-2026

Problem Statement & Objectives

1.1 Problem Statement

In the hospitality industry, developing a new business concept (such as a hotel, café, or resort) requires significant creative effort, market research, branding design, and visualization. This process is time-consuming and often requires collaboration between designers, marketers, and strategists.

There is a need for an automated system that can:

- Generate detailed hospitality business concepts.
- Provide both textual descriptions and visual representations.
- Store previous ideas for reuse and improvement.

The problem addressed in this project is to design and implement an AI-based system that can automatically generate multimodal hospitality concepts (text + image) from a single user prompt.

1.2 Project Objectives

The main objective of the project are:

- To design a system that generates complete hospitality business ideas from a single user prompt. The application produces both textual descriptions and corresponding visual representations.
- To integrate the Groq API using the model *llama-3.1-8b-instant* for generating detailed narratives. The model produces structured and context-aware hospitality descriptions.
- To incorporate an image generation API that creates visuals based on user prompts. The generated image aligns with the theme described in the narrative.
- To convert prompts into vector embeddings and store them in ChromaDB. This enables similarity search and supports future enhancements like RAG.
- To structure the system into independent modules with defined responsibilities. This ensures maintainability, scalability, and ease of future expansion.
- To develop a functional prototype integrating text, image, and vector modules. The implementation demonstrates practical application of Generative AI concepts.

1.3 Scope of the Project

The scope of this project includes:

- The system allows users to provide hospitality concept prompts through a Command-Line Interface (CLI) or a Streamlit-based web interface. This ensures simple and flexible interaction with the application.
- The application generates comprehensive and structured business descriptions based on the user's input prompt. The narrative includes concept theme, ambiance, services, and overall business positioning.
- The system produces a visual representation of the hospitality concept using an image generation model. The generated image reflects the theme and design elements described in the narrative.
- User prompts are converted into vector embeddings and stored in ChromaDB. This enables semantic search and supports similarity-based retrieval of previous concepts.
- The system is structured into independent modules, each responsible for a specific function. This modular design improves maintainability and ensures clear separation of concerns.
- The architecture is designed to accommodate future improvements without major structural changes. It supports potential upgrades such as RAG integration, web deployment, and performance optimization.

Proposed Solution

2.1 Key features

- The system generates hospitality concepts using both textual and visual outputs. This multimodal approach enhances creativity and provides a more comprehensive concept presentation.
- The application uses a Large Language Model to generate detailed hospitality business descriptions. The generated narrative is context-aware and aligned with the user prompt.
- An image generation model creates visual representations of the hospitality concept. The image reflects the theme and design elements described in the text.
- User prompts are converted into vector embeddings and stored in ChromaDB. This enables efficient semantic storage and retrieval of concepts.

- The system supports similarity-based retrieval using stored embeddings. This allows related prompts to be identified for future recommendations or enhancements.

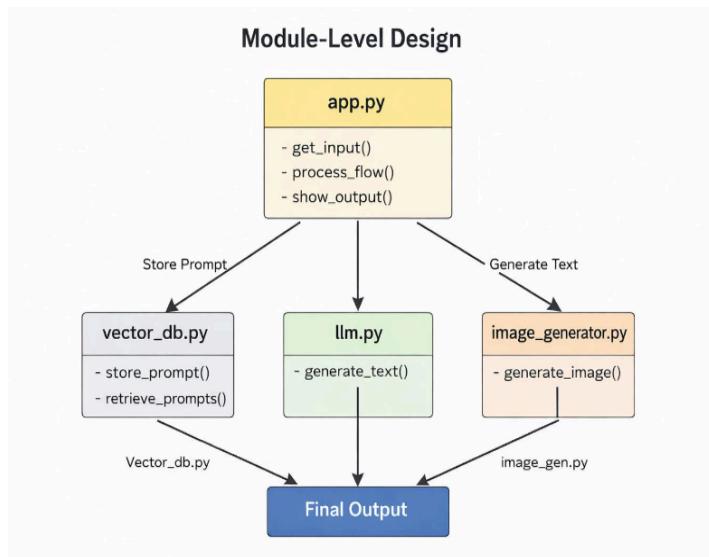
- The application is divided into independent modules with defined responsibilities. This ensures maintainability, scalability, and clean system organization.
- The system provides a Command-Line Interface for simple and direct user input. It allows users to generate concepts through structured terminal interaction.
- A Streamlit-based interface enables interactive web-based usage of the system. This improves accessibility and user experience compared to CLI-only execution.
- The system includes mechanisms to handle API failures and network issues. This ensures stable execution and prevents unexpected crashes during runtime.

2.2 Overall Architecture / Workflow

The system follows a modular architecture where each module has a clearly defined responsibility and interacts through a centralized controller. Each functional unit is implemented in a separate Python file to ensure modularity, maintainability, and scalability.

System Components with File Names

- Main Controller – app.py
- Text Generation Module – llm.py
- Image Generation Module – image_generator.py
- Vector Database Module – vector_db.py



The system operates through a well-defined sequential workflow consisting of the following stages:

Step 1: User Input (Handled by app.py)

- The user enters a hospitality concept prompt.
- The input is received by the Main Controller (`app.py`).
- The controller orchestrates the complete execution flow.

Step 2: Prompt Storage (Handled by vector_db.py)

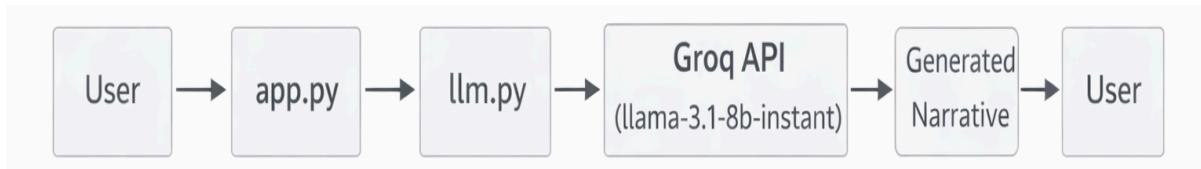
- `app.py` calls `store_prompt(prompt)` from `vector_db.py`.
- The prompt is converted into vector embeddings.
- Embeddings are stored in ChromaDB.
- A unique ID is assigned to each prompt. This enables:
 - Memory Storage
 - Similarity Search
 - Future Retrieval-Augmented Generation (RAG)

Step 3: Text Generation (Handled by llm.py)

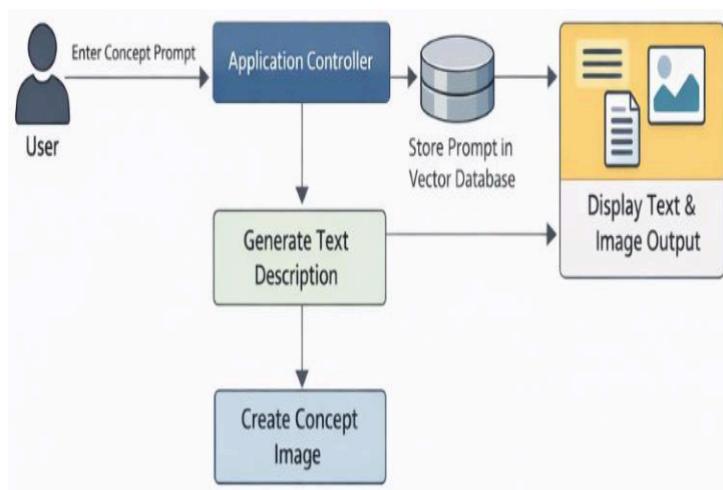
- `app.py` calls `generate_narrative(prompt)` from `llm.py`.
- The `llm.py` module:
 - Receives the user prompt.
 - Sends the prompt to the Groq API.
 - Uses the model: **"llama-3.1-8b-instant"**

- The Groq-hosted LLM processes the prompt.
- A detailed hospitality narrative is generated.
- The generated text is returned to app.py.

Text Generation Flow:

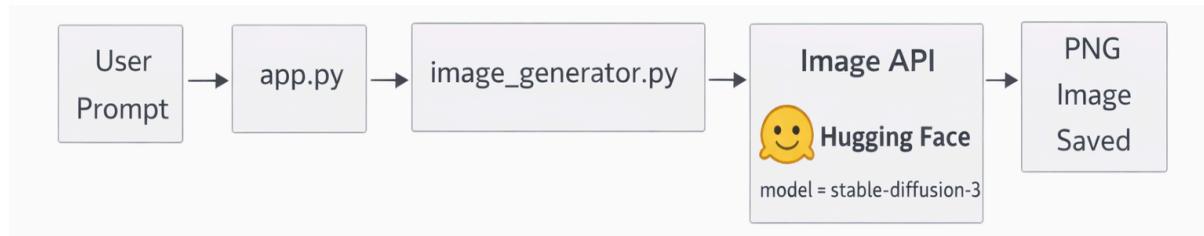


Process Flow:



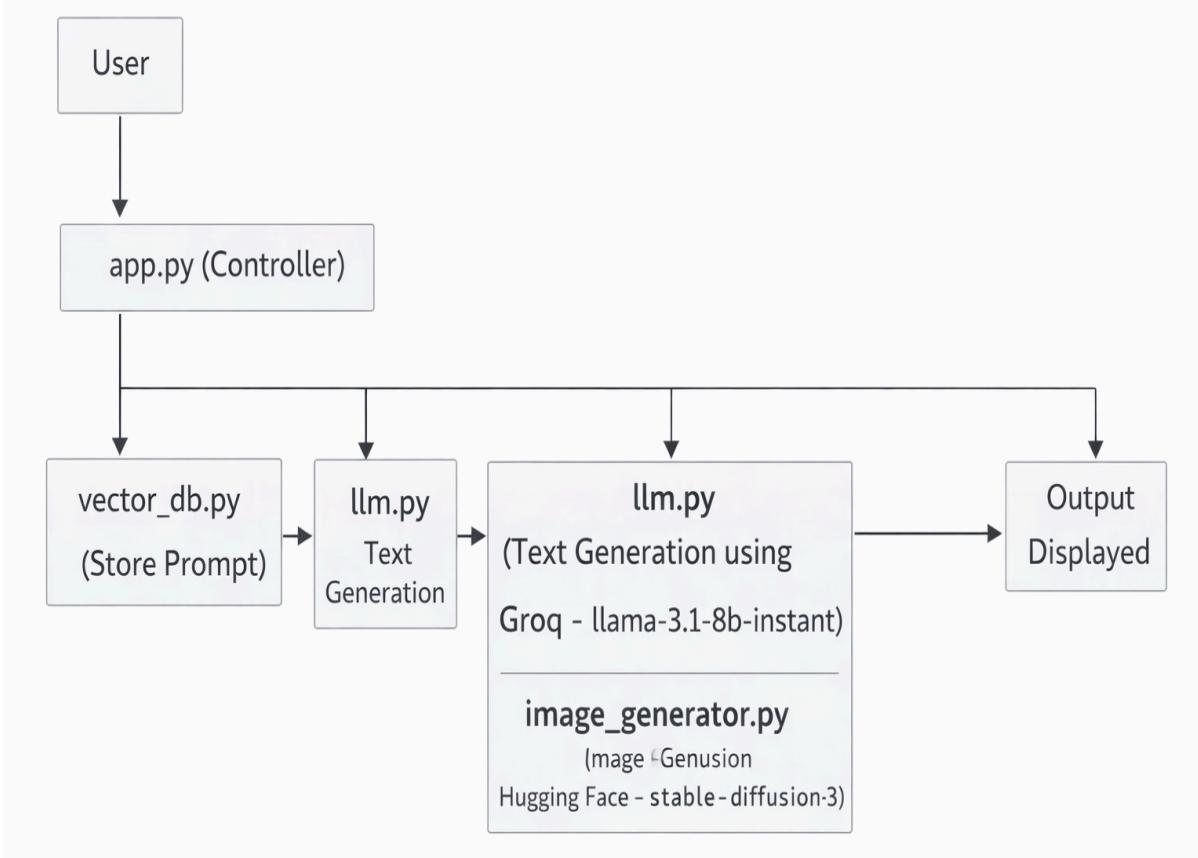
Step 4: Image Generation (Handled by image_generator.py)

- app.py calls generate_image(prompt) from image_generator.py.
- The module:
 - Sends the prompt to the Image Generation API.
 - Receives generated image data.
 - Saves the image as a PNG file (generated_image.png).
 - Returns the image file path to app.py.

Image Generation Flow:**Step 5: Output Display (Handled by app.py)**

- app.py collects:
 - Generated narrative (from llm.py)
 - Image path (from image_generator.py)
- The final output is displayed to the user via CLI or Streamlit UI.

End-to-End Workflow Representation:



2.3 Tools & Technologies Used

Programming Language: Python

Vector Database: ChromaDB

LLM API/Model for Text Generation: Groq API/llama-3.1-8b-instant

API/Model for Image Generation: Hugging Face/stable-diffusion-xl-base-1.0

Web UI Framework: Streamlit

Results & Output

3.1 Results & Outputs

The developed Multimodal Hospitality Concept Generator was successfully executed and tested using multiple user prompts. The system demonstrated its ability to generate both textual and visual outputs in an integrated manner.

When the user provides a hospitality concept prompt through the CLI or Streamlit interface, the system performs the following operations

The outputs confirm that the system correctly implements the multimodal workflow and produces context-aware hospitality concepts.

- Stores the prompt in the vector database (ChromaDB)
- Generates a detailed narrative using the Groq LLM
- Produces a corresponding visual representation using the image generation model
- Displays the final output to the user

The outputs confirm that the system correctly implements the multimodal workflow and produces context-aware hospitality concepts.

The project was implemented in Python using a modular structure. Each module (app.py, llm.py, image_generator.py, vector_db.py) was developed and tested independently before integration.

Sample Execution

Input Prompt: "Luxury eco-friendly beach resort"

System Output:

- A structured hospitality concept description was generated.
- A themed visual image aligned with the concept was produced.
- The prompt was successfully stored in the vector database.

Screenshots of Outputs:

Multimodal Hospitality Concept Creator

Enter your hospitality concept:
Luxury beachfront eco-resort in Bali

Generate Concept

Concept Description

Concept: Aurora Eco Retreat

Location: Pererenan Beach, Bali, Indonesia

Mission Statement: Aurora Eco Retreat offers a serene, luxurious, and eco-friendly beachfront experience, blending seamlessly into Bali's natural beauty, while promoting environmental sustainability and cultural immersion.

Project: Aurora Eco Retreat is a luxurious, 5-star luxury beachfront eco-resort nestled on the pristine Pererenan Beach in Bali, Indonesia. Our mission is to create a harmonious balance between luxurious amenities and environmentally responsible practices, allowing guests to unwind in a tranquil setting while supporting local communities and preserving Bali's natural heritage.

Key Features:

- Eco-Friendly Design: The resort will be built using locally sourced, sustainable materials, and incorporates green building techniques to minimize its carbon footprint.
- Luxurious Accommodation: 50 villas and suites, each with a private pool, outdoor bathroom, and breathtaking ocean views.
- Immersive Cultural Experiences: Guests can participate in traditional Balinese dance and cooking classes, learn about local arts, and engage with community-led initiatives.
- Wellness and Spa: A state-of-the-art spa center offering treatments inspired by Balinese traditions, yoga, and meditation classes, and access to a fitness center.
- Sustainable Dining: The resort's restaurant will serve locally sourced, organic cuisine, using compostable and reusable materials to minimize its environmental impact.
- Beach Activities: Guests can enjoy snorkeling, paddleboarding, kayaking, and surfing lessons, as well as participate in marine conservation efforts.

Sustainable Features:

- Renewable Energy: Solar panels and wind turbines will provide a significant portion of the resort's energy needs.
- Water Conservation: Greywater reuse systems, low-flow fixtures, and efficient irrigation systems will minimize water waste.
- Waste Management: Composting and recycling programs, as well as a zero-waste policy for single-use plastics.
- Local Sourcing: Sourcing materials and supplies from local communities to support the local economy.

Accommodation Types:

- Oceanfront Villas: Luxurious 1-3 bedroom villas with private pools, outdoor bathrooms, and breathtaking ocean views.
- Garden Villas: Cozy 1-2 bedroom villas surrounded by lush gardens and tropical greenery.
- Suites: Spacious 1-2 bedroom suites with private balconies and ocean views.

Room Amenities:

- Private Pool: Each villa and suite features a private pool, perfect for relaxation and enjoyment.
- Outdoor Bathrooms: Spacious outdoor bathrooms with natural stone features and rain shower heads.
- Wishbar: Stocked with local and organic refreshments, including coconut water, fresh fruits, and herbal teas.
- In-Villa Entertainment: Enjoy a selection of movies, music, and games on a state-of-the-art entertainment system.

Community Engagement:

- Local Artisans: Collaborate with local artisans to create unique, sustainable products and souvenirs.
- Community-Led Initiatives: Partner with local organizations to support marine conservation efforts, community development projects, and cultural preservation initiatives.

Revenue Model:

Revenue Model:

- Luxury Accommodations: Offer a range of accommodation options, including villas, suites, and room capsules.
- Food and Beverage: Provide fine dining experiences, room service, and in-villa catering services.
- Activities and Experiences: Offer surfing, paddleboarding, snorkeling, in-villa diving lessons, as well as cultural experiences, spa treatments, and wellness programs.

Marketing Strategy:

- Social Media: Utilize Instagram, Facebook, and Twitter to showcase the resort's lush landscapes, eco-friendly features, and cultural experiences.
- Influencer Partnerships: Collaborate with travel influencers, cultural ambassadors, and wellness thought leaders to promote the resort.
- Targeted Advertising: Advertise in luxury travel publications, online travel agencies, and social media platforms targeting high-end travelers.

Financial Projections:

- Revenue: \$5 million in the first year, increasing by 10% annually for the next three years.
- Expenses: \$3.5 million in the first year, decreasing by 5% annually for the next three years.
- Net Profit: \$1.5 million in the first year, increasing by 15% annually for the next three years.

Timeline:

- Design and Planning: 6 months
- Construction: 12 months
- Soft Opening: 6 months
- Grand Opening: 18 months from project commencement

Budget: \$25 million

Management Team:

- General Manager: Experienced hospitality professional with a background in luxury resort management.
- Marketing Manager: Skilled marketing expert with expertise in luxury travel and eco-tourism.
- Operations Manager: Seasoned operations professional with a focus on sustainable practices and community engagement.
- Sustainability Manager: Certified environmental professional with expertise in green building, waste reduction, and energy efficiency.

Conclusion:

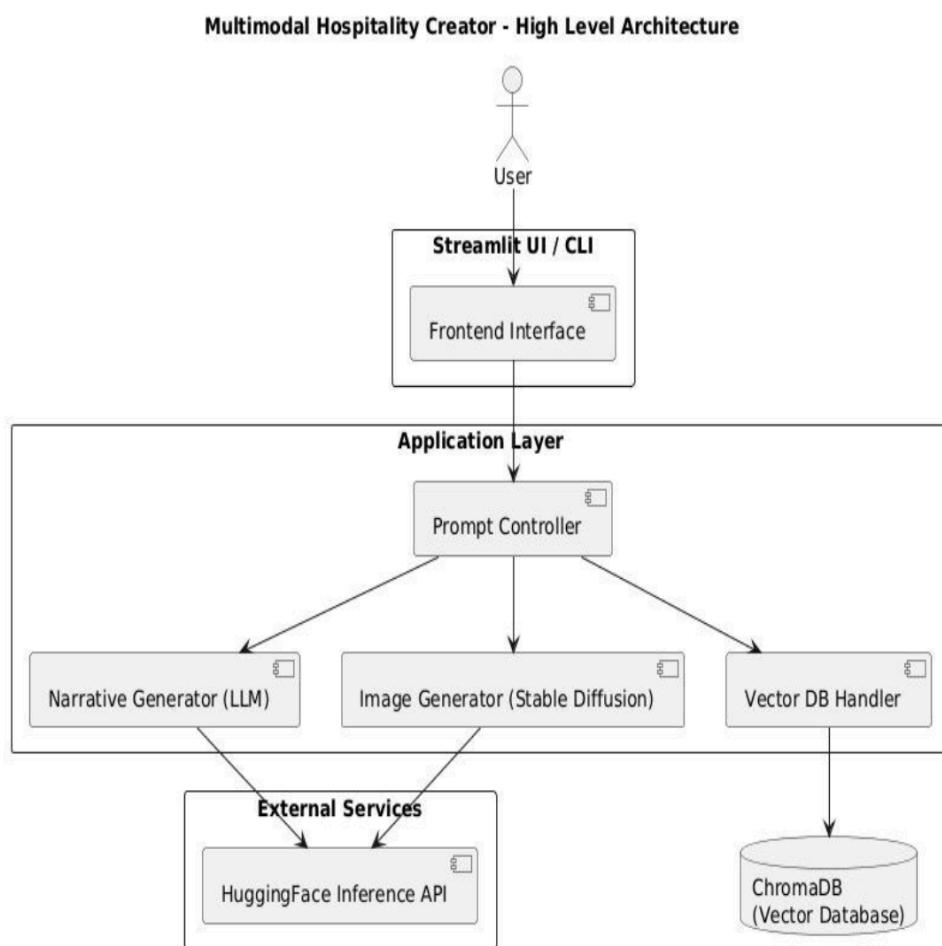
Aurora Eco Retreat is a pioneering luxury beachfront eco-resort in Bali, designed to set a new standard for sustainable hospitality. By blending luxurious amenities with eco-friendly features and cultural immersion, our resort will attract high-end travelers seeking a unique and responsible beach vacation. With a dedicated team of experts and a strong commitment to environmental stewardship, our management team, we are confident that Aurora Eco Retreat will become a benchmark for luxury eco-resorts worldwide.

3.2 Reports / dashboards / models

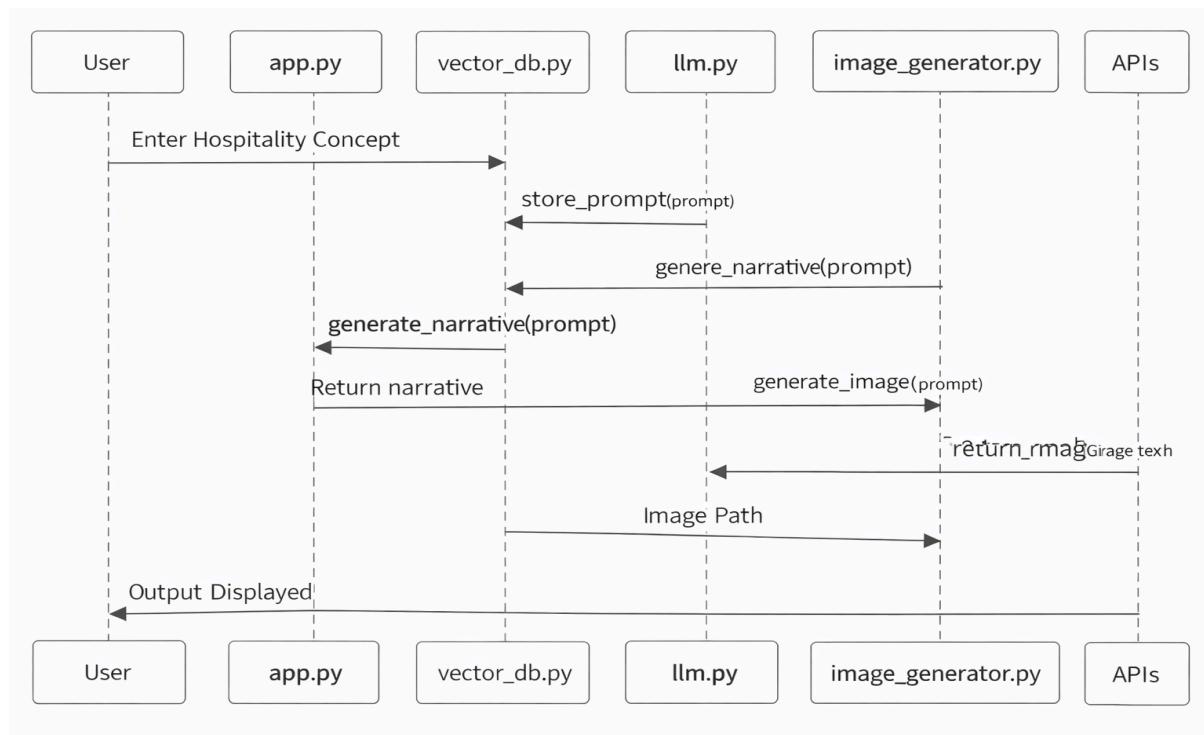
This section presents the technical artifacts, architectural validation, and model details that support the implementation of the Multimodal Hospitality Concept Generator. These reports demonstrate the internal working of the system beyond the visible outputs.

System Architecture Validation

The system follows a modular architecture where each component performs a single responsibility. The Main Controller (app.py) orchestrates communication between the text generation module, image generation module, and vector database.



Runtime Sequence Diagram



Model Details:

The system integrates external AI services to enable multimodal generation:

Text Generation Model

- Provider: Groq API
- Model: llama-3.1-8b-instant
- Purpose: Generate structured hospitality narratives

Image Generation Model

- Provider: Hugging Face

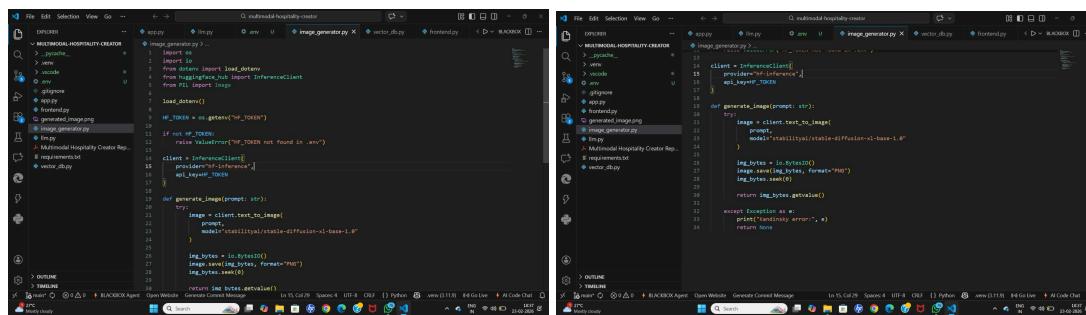
- Model:
stable-diffusion-xl-base-1.0
- Purpose: Generate concept visuals

Vector Database

- Technology: ChromaDB
- Purpose:
 - Store prompt embeddings
 - Enable similarity search
 - Support future RAG implementation

Code Implementation Evidence

The project was implemented in Python using a modular structure. Each module (app.py, llm.py, image_generator.py, vector_db.py) was developed and tested independently before integration.



```

MULTIMODAL-HOSPITALITY-CREATOR
+- app.py
+- llm.py
+- image_generator.py
+- vector_db.py
+- frontend.py
+- .env
+- .gitignore
+- .gitmodules
+- .pre-commit-config.yaml
+- .vscode
+- .wheels
+- requirements.txt
+- requirements.in
+- setup.py
+- vector_db.py

client = InferenceClient()
provider="hf-inference"
api_key=_HFTOKEN

def generate_image(prompt: str):
    try:
        image = client.text_to_image(
            prompt,
            model="stability/stable-diffusion-xl-base-1.0"
        )
        img_bytes = io.BytesIO()
        image.save(img_bytes, format="JPEG")
        img_bytes.seek(0)
        return img_bytes.getvalue()
    except Exception as e:
        print(f"Handling error: {e}")
        return None

```

3.3 Key outcomes

- A fully functional prototype was successfully developed integrating text and image generation. The system demonstrates practical multimodal AI implementation.
- The project follows a well-structured modular design with clearly separated components. This improves maintainability and ensures clean system organization.

- User prompts are stored as vector embeddings in ChromaDB for semantic processing. This enables efficient similarity search and future retrieval enhancements.
- The system successfully integrates external AI services for text and image generation. API communication and response handling were implemented effectively.
- The architecture supports future expansion without major redesign. It provides a strong foundation for enhancements such as RAG and web deployment.

Conclusion

The Multimodal Hospitality Concept Generator successfully demonstrates the integration of Generative AI technologies into a unified multimodal system. By combining text generation, image generation, and vector storage, the system automates hospitality concept creation.

The project enhanced understanding of:

- Gained practical understanding of integrating text and image generation into a unified AI system.
- Learned how to generate, store, and use vector embeddings for semantic search and retrieval.
- Developed hands-on experience in integrating and managing external AI APIs within an application.
- Understood the importance of separation of concerns and structured module design for maintainability.
- Built knowledge of designing AI systems that support future scalability and architectural expansion.

The system fulfills all objectives and provides a robust foundation for future innovation.

Future Scope & Enhancements

- Web deployment on cloud can be implemented to make the system accessible remotely and suitable for real-world usage.
- User authentication and role-based access control can be added to support secure multi-user environments.
- Multiple image generation support can be introduced to create different design variations for a single prompt.

- Retrieval-Augmented Generation (RAG) can be implemented to improve text generation using stored prompts and embeddings.
- A recommendation engine can be developed to suggest similar or enhanced hospitality concepts based on semantic similarity.
- Performance optimization techniques such as parallel API calls and caching can be introduced to reduce response time and API costs.
- A dashboard and analytics module can be integrated to track usage patterns and prompt trends.
- Multi-language support can be added to generate hospitality concepts in different languages.