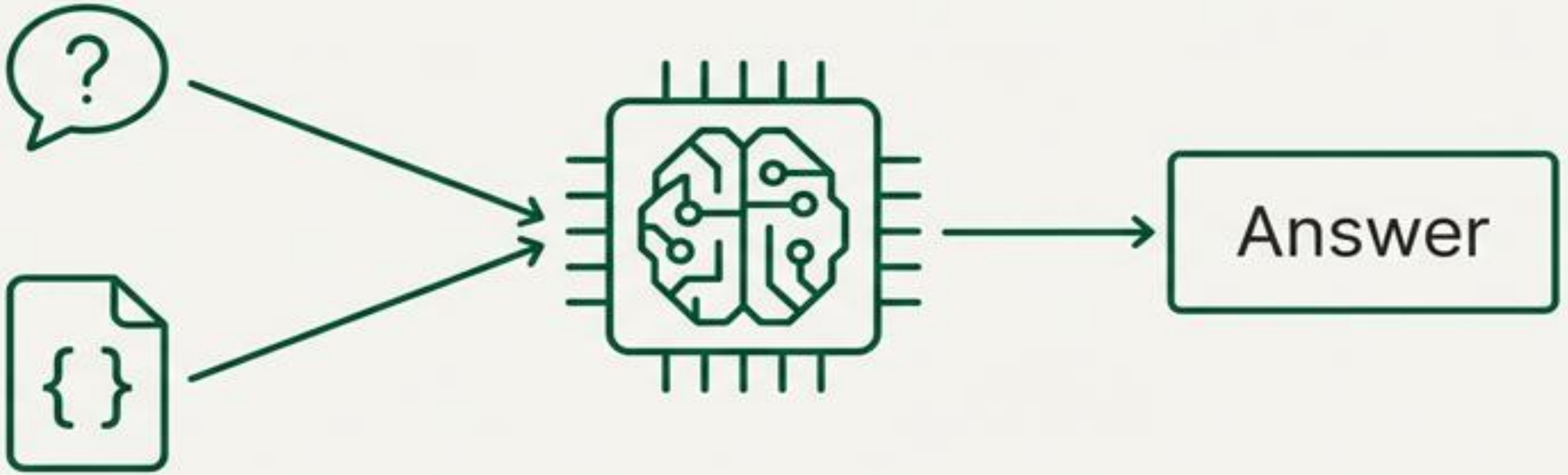
Course Resource Locator

Instant, Accurate Answers. Zero Database.



BUILT BY: LM. ILHAM
CSH 4112

Finding Course Information Can Be a Labyrinth

Time-Consuming Searches

Students and staff waste valuable time hunting through disparate portals, scattered PDFs, and outdated websites for simple information.

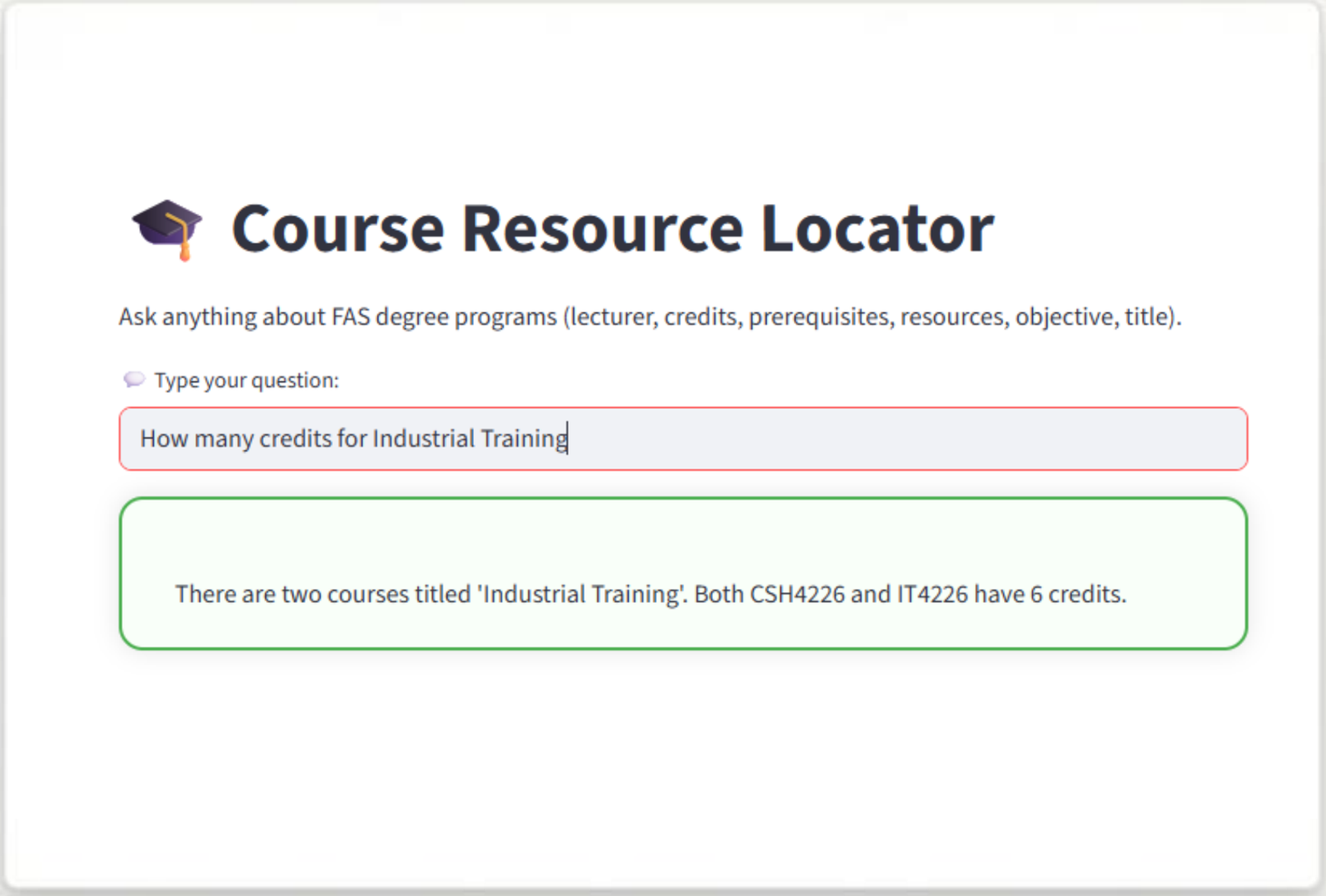
Inconsistent Data

Information is often fragmented across multiple sources, leading to confusion and the risk of relying on out-of-date details.

High-Friction Experience

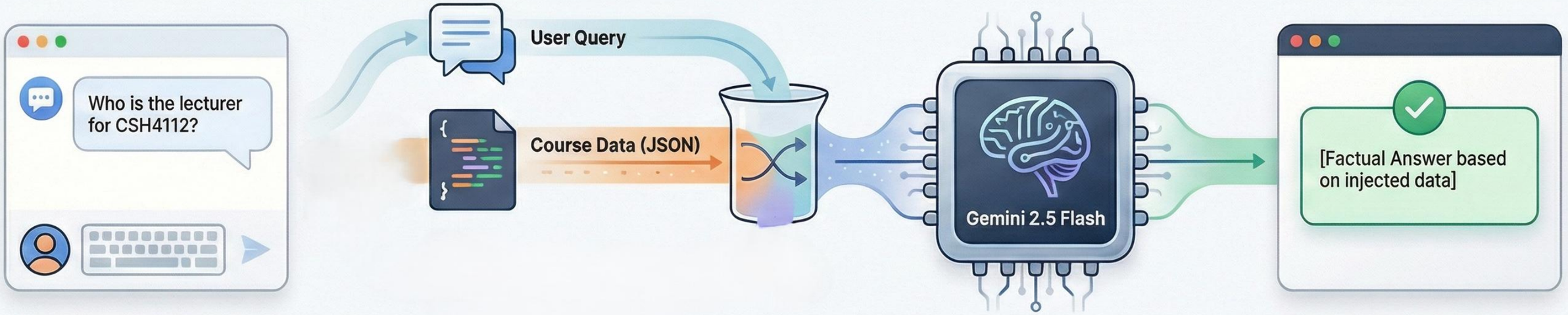
Answering even the simplest questions often requires navigating complex, unintuitive systems, creating unnecessary frustration.

Ask a Question. Get an Instant, Sourced Answer.



The Course Resource Locator is a lightweight web app that uses Generative AI to provide immediate, accurate answers to queries about course details, directly from the source data.

FROM USER QUERY TO RENDERED ANSWER



1. User Asks a Question

The user opens the app and types a query like, "Who is the lecturer for CSH4112?"

2. Data is Injected

The system combines the question with course data (JSON) and strict "guardrail" prompts.

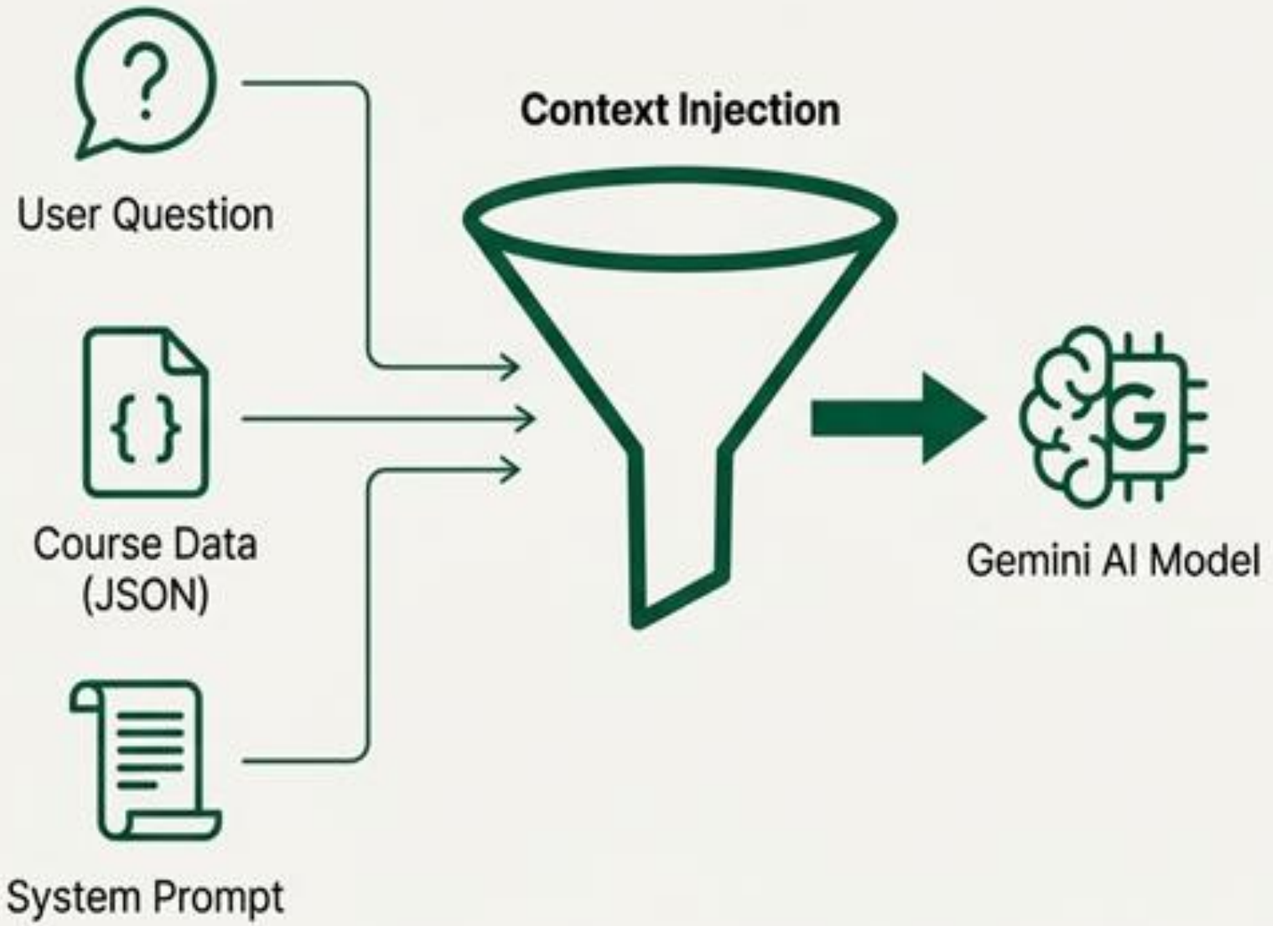
3. AI Processes the Request

Gemini 2.5 Flash reads the injected data to generate an accurate, hallucination-free answer.

4. Answer is Displayed

The result appears in a custom-styled green box, indicating the process is complete.

The Secret Sauce: No Database, No Problem



What It Is

Instead of querying a database, we dynamically inject the entire course dataset (as a JSON dictionary) directly into the prompt along with the user's question. The AI has all possible information available for every query.

Why It Matters

- **Simplicity:** Eliminates database setup, management, schemas, and complex query logic.
- **Speed:** Eradicates network latency associated with traditional database calls.
- **Accuracy:** Guarantees the AI model reasons over the complete and most current dataset for every single request.

The Intelligence: Controlled and Factual AI Processing

The Code

```python
# System Prompt
SYSTEM_PROMPT = """
You are a Course Information Assistant.

You must:
1. Read the user's question.
2. Identify the course (by code or title).
3. Identify what field they want (lecturer, credits, objective, prerequisites, resources, title).
4. Answer using ONLY the course_data dictionary provided.
5. If the course or field does not exist, say so.
6. Never hallucinate; do not invent details.
"""

# Gemini LLM Function
def gemini_answer(query):
    prompt = f"{SYSTEM_PROMPT}\n\nHere is the course_data dictionary:\n{course_data}\n\nUser Question: {query}"
    response = client.models.generate_content(
        model="gemini-2.5-flash",
        contents=prompt
    )
    return response.text
```

The Explanation

Model

Powered by Google's Gemini 2.5 Flash

Strict Guardrails

The application is governed by a `SYSTEM_PROMPT` that enforces two non-negotiable rules for the AI:

> 1. `Answer using ONLY the course_data provided.`
> 2. `Never hallucinate or invent information.`

Result

This transforms the AI into a highly reliable **data parser**, not an unpredictable conversationalist.

Built with a Modern, Efficient Toolset

Streamlit

- Language: Python 3.13.2
- Web Framework: Streamlit (for rapid UI development and deployment)
- AI Service: Google Gemini API
- Environment Management: python-dotenv (for secure API key handling)
- IDE: Visual Studio Code / Google Colab

Elegant, Efficient, and Effective

The Problem: Access to course information was scattered, time-consuming, and frustrating for users.

The Solution: A clean, intuitive, AI-powered Q&A interface that provides instant answers.

The Core Logic: A novel, database-less architecture built on "Context Injection" for simplicity and speed.

The Result: A fast, accurate, and entirely free-to-operate application built with a modern tech stack.

Anatomy of the Course Resource Locator App

The App's Logic Flow: From Question to Answer

1. Start & Security Check

Checks for GEMINI_API_KEY. If missing, stops & shows error.

2. User Input

Who is the lecturer for CSH4112?

3. Data Fusion

- User Question
- Strict Rules (System Prompt)
- Course Data (JSON)

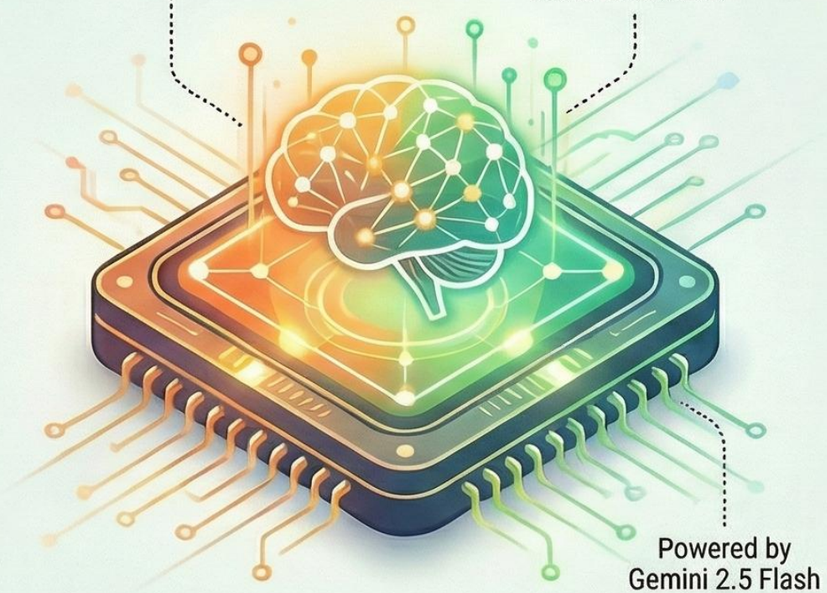
The Core Intelligence: How it Works

Context Injection Method

Feeds course data dictionary directly to AI with each query, eliminating database.

Strict AI Guardrails

SYSTEM_PROMPT enforces rules, Answer using ONLY provided course, data, Never hellucinate.



Powered by Gemini 2.5 Flash

Intelligence Core How it Works

5. Result Rendering

Wrap the raw AI output with Styled Green Box

Custom HTML and CSS

6. Display Answer

Lecturer:
Course: CSH4112 - Advanced Algorithms
Location: Room 305.

The User Experience

Easy UI with Streamlit

The entire user interface is built using the Python library Streamlit.

Real-time Feedback

Displays st.spinner("Thinking...") during processing

Custom University Look

Stylee output with green borders and shadow for professional appearance

Crash-Proof Error Handling

Usee try...except block to manage API errors grecefully, preventing crashes

Tech Stack & API Details

Development Tools & Libraries

VS Code | Python 3.13.2 | Streamlit | python-dotenv | google-geminl API

Cost:

Free
Gemini 2.5 Flash (no credit card)

15

Requests Per Minute

1,500

Requests Per Day

1 Million

Token Context Window
Model processes very large amount of information in a single request.