1. INTRODUCTION

1.1. Project Overview

The project focuses on the use of an AI-powered system, Civil Engineering Insight Studio, to assist construction supervisors in identifying and managing materials used at a building construction site. By uploading images of the construction area, the system analyzes visual data using image recognition and computer vision techniques to detect materials such as concrete, steel, and bricks. The tool then generates a detailed report that includes the type of materials present, their estimated quantity, and their location within the structure, helping supervisors gain a clear understanding of material usage.

This approach reduces the need for manual inspection and improves efficiency in construction monitoring and documentation. The system supports better planning and decision-making by providing accurate and organized material information, which helps in resource management and progress tracking. Overall, the project aims to enhance productivity, improve accuracy, and support modern digital practices in construction management through intelligent automation.

1.2. Objectives

The main objectives of the Civil Engineering Insight Studio project are:

- To develop an AI-based system for automatic identification of construction materials from site images.
- To accurately identify materials such as concrete, steel, and bricks used in construction.
- To reduce manual inspection time and effort for construction supervisors.
- To improve material tracking and documentation during construction activities
- To support better decision-making through clear and organized material reports.
- To minimize human errors in material identification and reporting.

2. Ideation Phase

2.1. Problem Statement

The problem is the lack of a quick and reliable method to analyze construction materials directly from site images and obtain accurate information about material type, quantity, and location. Therefore, the problem statement highlights the need for an intelligent tool, such as Civil Engineering Insight Studio, that can automate material identification and

provide clear insights to improve decision-making, save time, and reduce errors in construction management.

Problem Statement (PS)	I am	I'm trying to	But	Because	Which makes me feel
PS-1	A food blogger	Create detailed, and high-quality recipe blog content in less time	Writing long and recipe blogs manually is time-consuming	It requires research, creativity, formatting, and consistency	Frustrated and less productive
PS-2	A cooking enthusiast	Generate customized recipes	Existing recipes available online are generic and not customizable	Most platforms do not offer AI-driven recipe blog generation	Dissatisfied and overwhelmed while planning meals

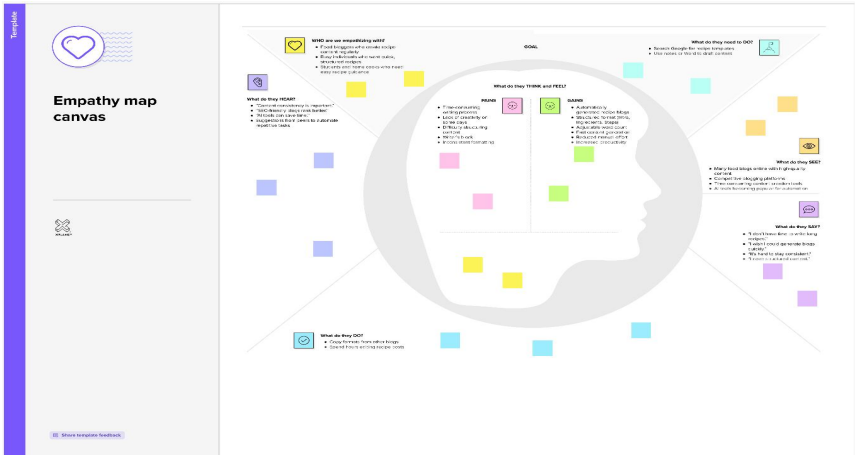
2.2. Empathy Map Canvas

Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

Example:



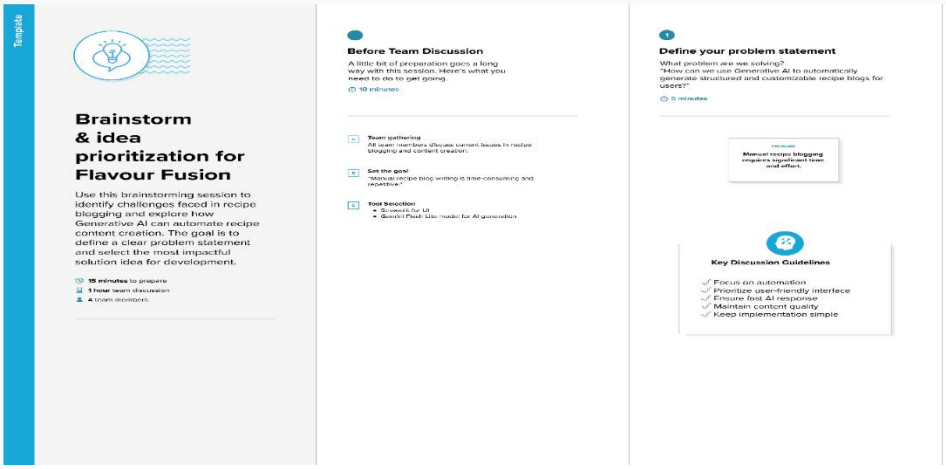
2.3 Brainstorming

Brainstorm & Idea Prioritization:

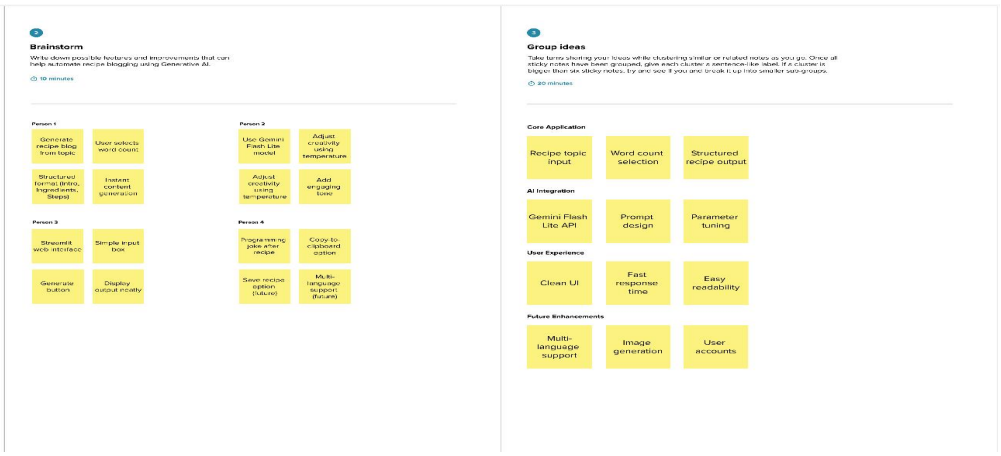
Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

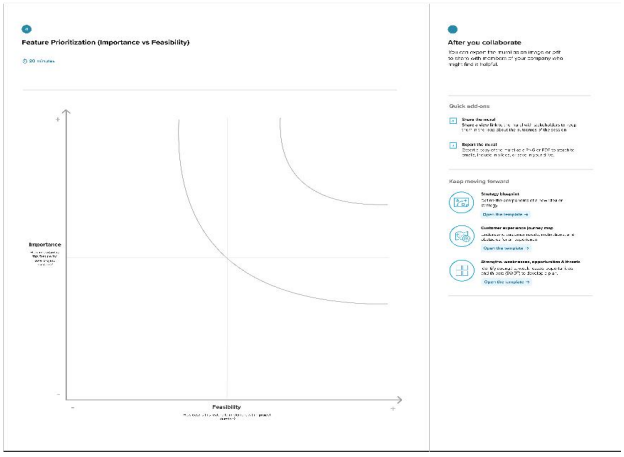
Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



3. Requirement Analysis

3.1. Solution Requirement

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Sturcture Input Module	Enter consturction structure name Select web browser
FR-2	Input Validation	Validate empty input Validate photo insertion
FR-3	AI Integration	Connect to Gemini Flash Lite API Send structured prompt to model
FR-4	Data Generation	Generate structured recipe blog Include introduction, ingredients, steps
FR-5	Output Display	Display generated data regarding to user inserted data
FR-6	Additional Feature	Generate programming joke (optional feature)

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application must have a simple and intuitive Streamlit interface that is easy to use.
NFR-2	**Security**	API keys must be securely stored and not exposed in the frontend.
NFR-3	**Reliability**	The system should generate consistent and structured outputs for valid inputs.
NFR-4	**Performance**	Data generation should complete within a few seconds.

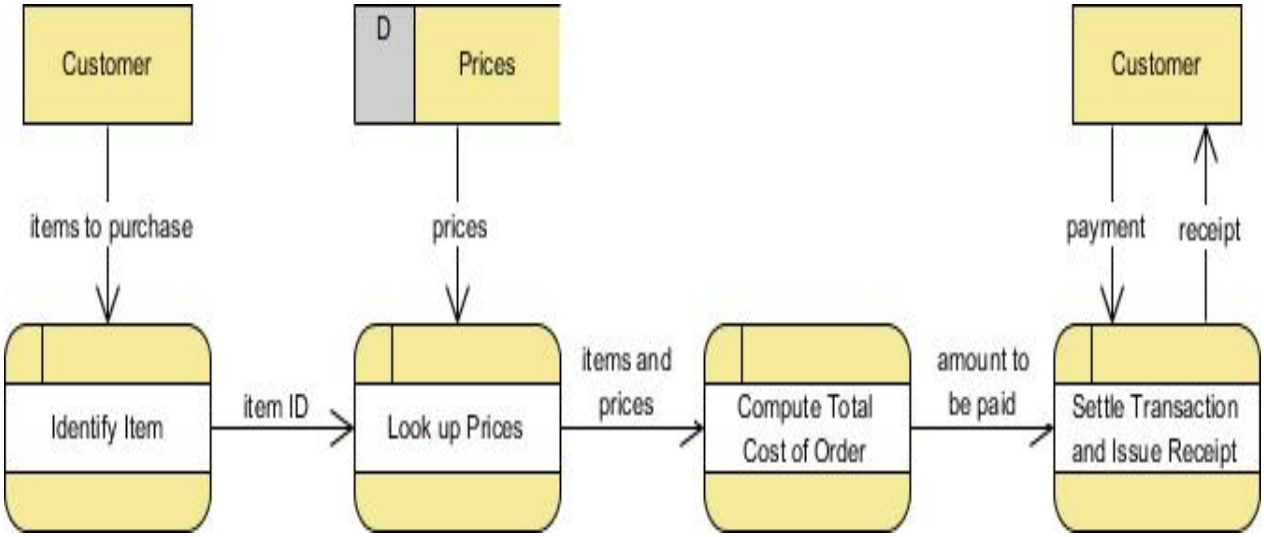
NFR-5	**Availability**	The application should be accessible online whenever deployed.
NFR-6	**Scalability**	The system should handle multiple users without significant performance degradation.

3.2. Data Flow Diagram

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is

Sorted.



User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Release
Customer (Mobile user)	User Interface Setup	USN-1	As a user, I can access a Streamlit-based interface to enter a recipe topic and word count.	High	Sprint-1
Administrator	Input Validation	USN-2	As a user, I want the application to validate my inputs before generating the recipe.	High	Sprint-1
Customer Care Executive	AI Model Integration	USN-3	As a user, I want the system to generate a recipe blog using the Gemini Flash Lite model.	High	Sprint-2
Administrator	Joke Generation	USN-4	As a user, I want to see a programming joke while the recipe is being generated.	Medium	Sprint-2

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority	Release
Customer (Mobile user)	Output Display	USN-5	As a user, I want to view the generated recipe blog clearly on the screen.	High	Sprint-3
Customer Care Executive	Deployment	USN-6	As a user, I want the application to be deployed and accessible through the internet.	Medium	Sprint-3

3.3. Technology Stack

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web-based interface where users enter structure name and photo	Streamlit (Python Web Framework)
2.	Application Logic-1	Input validation and prompt construction logic	Python
3.	Application Logic-2	AI request handling and response processing	Google Generative AI API
4.	Application Logic-3	Data formatting and output structuring	Python
5.	File Storage	Local environment for source code and logs	Local File System
6.	External API-1	Generative AI service for information generation	Gemini Flash Lite (models/gemini-flash-lite-latest)
7.	Machine Learning Model	Pre-trained generative AI model for data generation	Gemini Flash Lite Model
8.	Infrastructure (Server / Cloud)	Deployment of application	Streamlit Cloud / Local Deployment

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Web framework and development tools used	Streamlit, Python
2.	Security Implementations	Secure storage of API keys and environment variables	Environment Variables (.env), Streamlit Secrets
3.	Scalable Architecture	Web-based architecture supporting multiple users	Cloud-based deployment (Streamlit Cloud)
4.	Availability	Application accessible online after deployment	Streamlit Cloud Hosting
5.	Performance	Fast response generation using lightweight AI model	Gemini Flash Lite (optimized for low latency)

4. PROJECT DESIGN

4.1. Problem Solution Fit

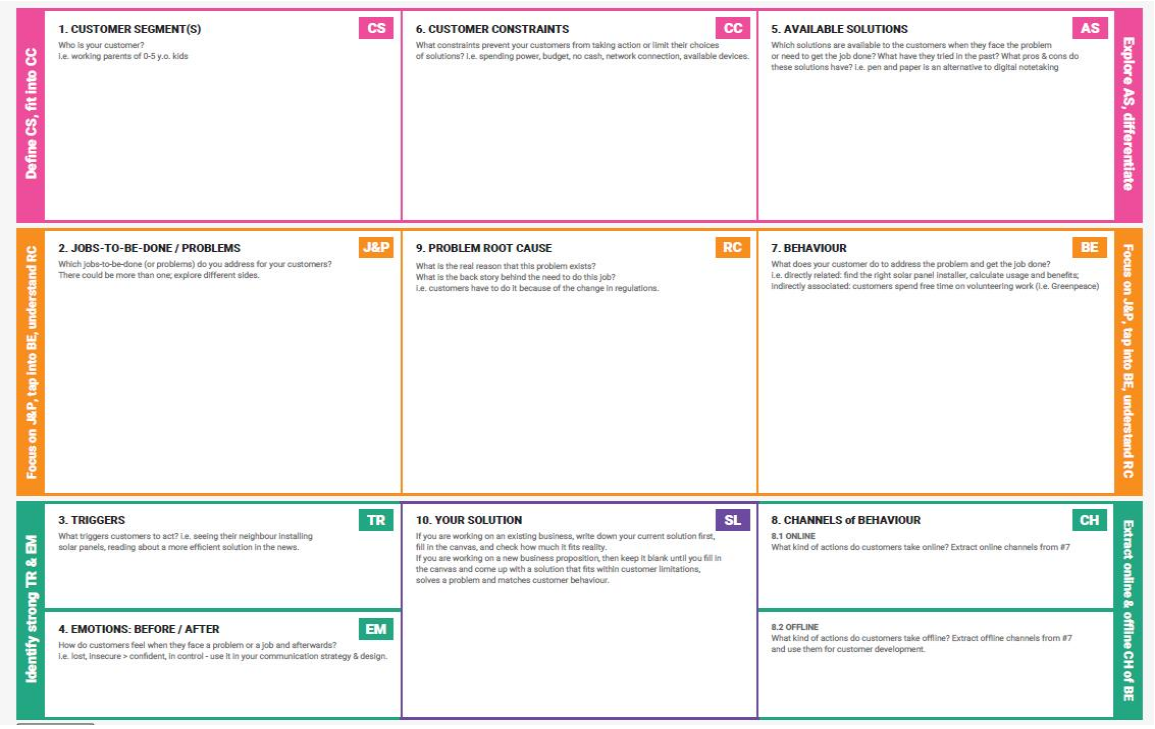
Problem – Solution Fit Template:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why

Purpose:

- ❑ Solve complex problems in a way that fits the state of your customers.
- ❑ Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.
- ❑ Sharpen your communication and marketing strategy with the right triggers and messaging.
- ❑ Increase touch-points with your company by finding the right problem-behavior fit and building trust by solving frequent annoyances, or urgent or costly problems.
- ❑ **Understand the existing situation in order to improve it for your target group.**

Template:



4.2. Proposed Solution

Proposed Solution Template:

Project team shall fill the following information in the proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	In modern construction projects, supervisors face challenges in accurately identifying and documenting materials used at various stages of building development. Manual inspection of construction materials such as concrete, steel, and bricks is time-consuming, prone to human error, and often lacks detailed documentation regarding material quantity and placement within the structure.
2.	Idea / Solution description	The ideal solution is to develop an AI-powered image analysis system integrated within the Civil Engineering Insight Studio that enables construction supervisors to automatically identify and document materials used on-site. The system should allow users to upload construction site images and optionally provide contextual notes. Using advanced computer vision and generative AI models, the tool would analyze the image to detect construction materials such as concrete, steel, and bricks, estimate their quantities, and determine their locations within the structure.
3.	Novelty / Uniqueness	The novelty of the Civil Engineering Insight Studio lies in its integration of artificial intelligence and computer vision specifically tailored for civil engineering applications. Unlike traditional construction monitoring methods that rely heavily on manual inspection and documentation, this system introduces an automated, image-based material identification approach.
4.	Social Impact / Customer Satisfaction	The Civil Engineering Insight Studio project has a positive social impact by promoting safer, more efficient, and higher-quality construction practices. By enabling accurate identification and documentation of construction materials, the system helps ensure that appropriate materials are used in building projects, reducing the risk of structural failures and enhancing public safety. Improved monitoring and quality control contribute to stronger and more reliable infrastructure, which directly benefits communities.
5.	Business Model (Revenue Model)	The Civil Engineering Insight Studio can adopt business model, offering AI-powered material identification and structural analysis as a subscription-based web application. Construction companies, contractors, engineering consultants, and infrastructure firms can subscribe to the platform to access image-based analysis

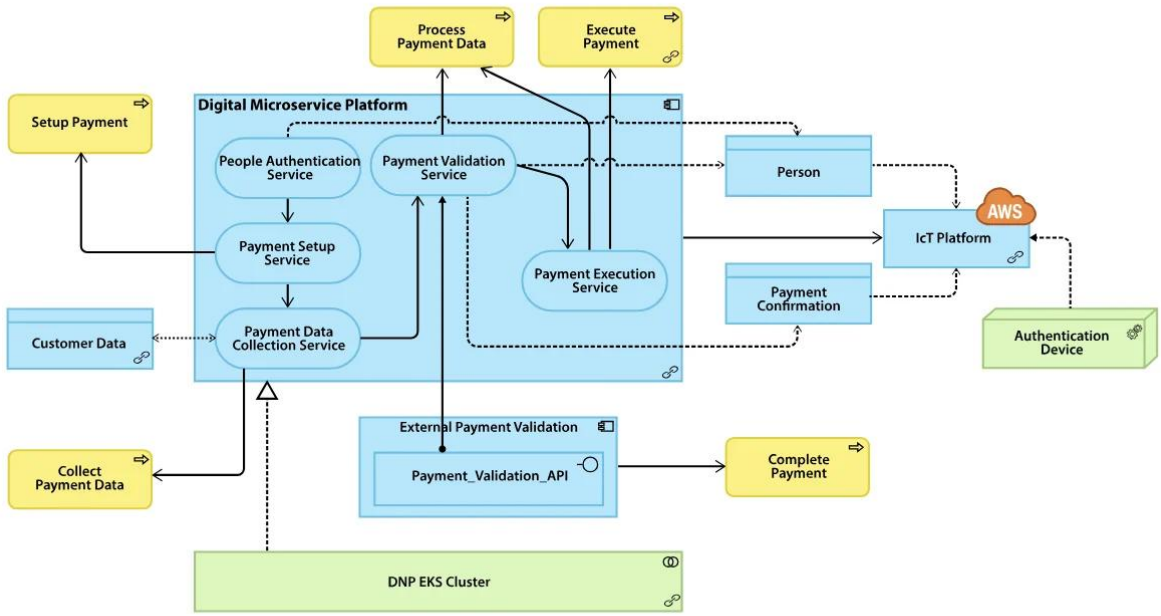
		tools that improve site monitoring and documentation efficiency.
6.	Scalability of the Solution	The Civil Engineering Insight Studio is designed to be highly scalable, both technically and operationally. Since the system is built as a web-based application using AI APIs, it can handle increasing numbers of users and image uploads by leveraging cloud infrastructure. As demand grows, the application can scale horizontally by adding more servers or cloud instances to manage higher traffic and processing loads without affecting performance.

4.3 Solution Architecture

Solution Architecture:

The Civil Engineering Insight Studio follows a layered, modular architecture that integrates a web-based user interface with an AI-powered image analysis engine through secure cloud APIs. The architecture ensures scalability, security, and efficient data processing.At the presentation layer, the system uses a Streamlit-based web interface where construction supervisors upload images and optionally provide descriptive inputs. This layer handles user interaction, input validation, and result visualization.

Example - Solution Architecture Diagram:



5. PROJECT PLANNING & SCHEDULING

5.1. Project Planning

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Product Backlog, Sprint Schedule, and Estimation

The following table represents the product backlog and sprint-wise planning for the Civil Engineering Insight Studio project.

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint 1	User Interface Setup	USN-1	As a user, I can access a Streamlit-based interface to enter a recipe topic and word count.	2	High	All Team Members
Sprint 1	Input Validation	USN-2	As a user, I want the application to validate my inputs before generating the recipe.	1	High	All Team Members
Sprint 2	AI Model Integration	USN-3	As a user, I want the system to generate a recipe blog using the Gemini Flash Lite model.	3	High	All Team Members
Sprint 2	Joke Generation	USN-4	As a user, I want to see a programming joke while the recipe is being generated.	1	Medium	All Team Members
Sprint 3	Output Display	USN-5	As a user, I want to view the generated recipe blog clearly on the screen.	2	High	All Team Members
Sprint 3	Deployment	USN-6	As a user, I want the application to be deployed and accessible through the internet.	2	Medium	All Team Members

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint 1	20	4 Days	28 January 2026	31 January 2026	20	31 January 2026

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint 1	20	4 Days	28 January 2026	31 January 2026	20	31 January 2026
Sprint 2	20	8 Days	02 February 2026	09 February 2026	20	09 February 2026
Sprint 2	20	8 Days	02 February 2026	09 February 2026	20	09 February 2026
Sprint 3	20	7 Days	12 February 2026	18 February 2026	20	18 February 2026
Sprint 3	20	7 Days	12 February 2026	18 February 2026	20	18 February 2026

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

Test Scenarios & Results

Test Case ID	Scenario (What to test)	Test Steps (How to test)	Expected Result	Actual Result	Pass/ Fail
FT-01	Structure Topic Input Validation	Enter valid structue name and leave web browser fillingas empty	Valid input accepted, error shown for empty input	As Expected	Pass
FT-02	Data Input Validation	Enter invalid name	Accepts valid input, shows warning for invalid input	As Expected	Pass
FT-03	AI data Generation	Enter structure name and click “Describe structure”	Structured data with title, materials generated	As Expected	Pass
FT-04	Gemini API Connection Check	Trigger structure data generation with valid API key	API responds successfully and generates recipe	As Expected	Pass
PT-01	Response Time Test	Measure time after clicking generate	Structure details generated within 20 seconds	Within Limit	Pass

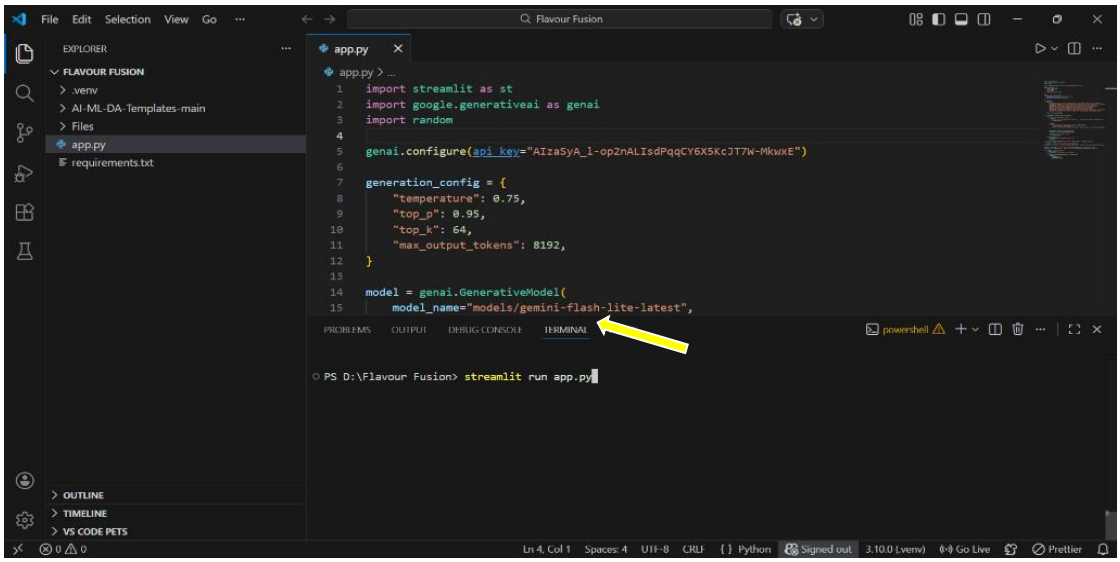
PT-02	Multiple Request Handling	Generate multiple structures sequentially	Application handles requests without crash	Stable	Pass
PT-03	Deployment Test	Access deployed app via browser	Application loads and works correctly online	Working	Pass

6. RESULTS

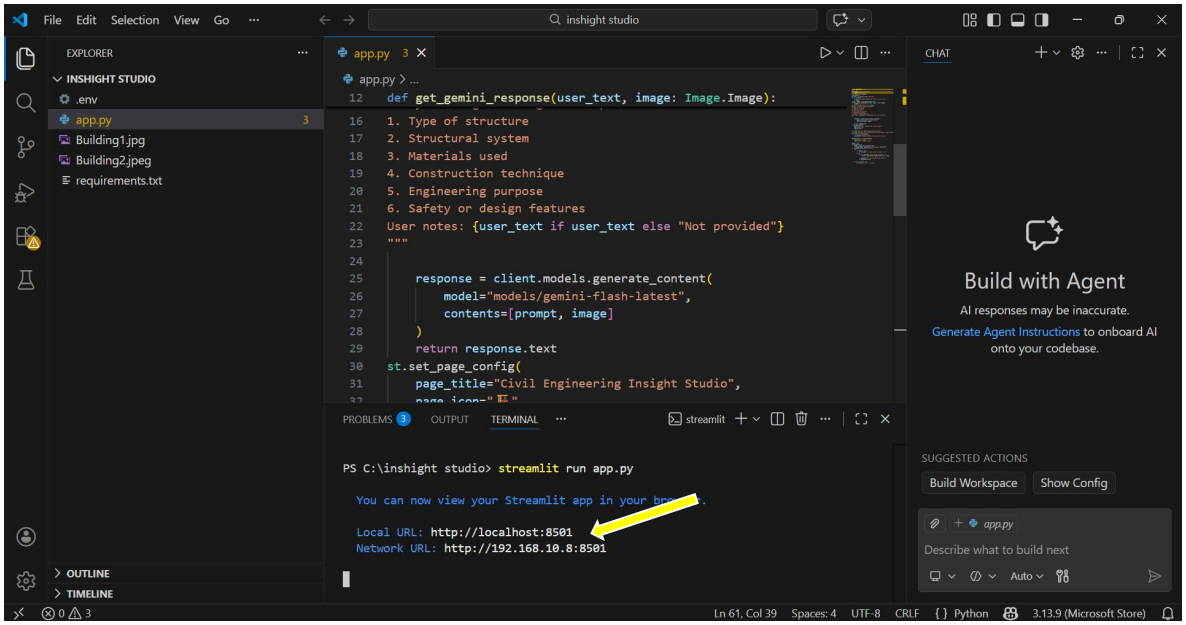
6.1. Output Screenshots

The complete execution of Civil Engineering Insight Studio application is represented step by step in the following screenshots.

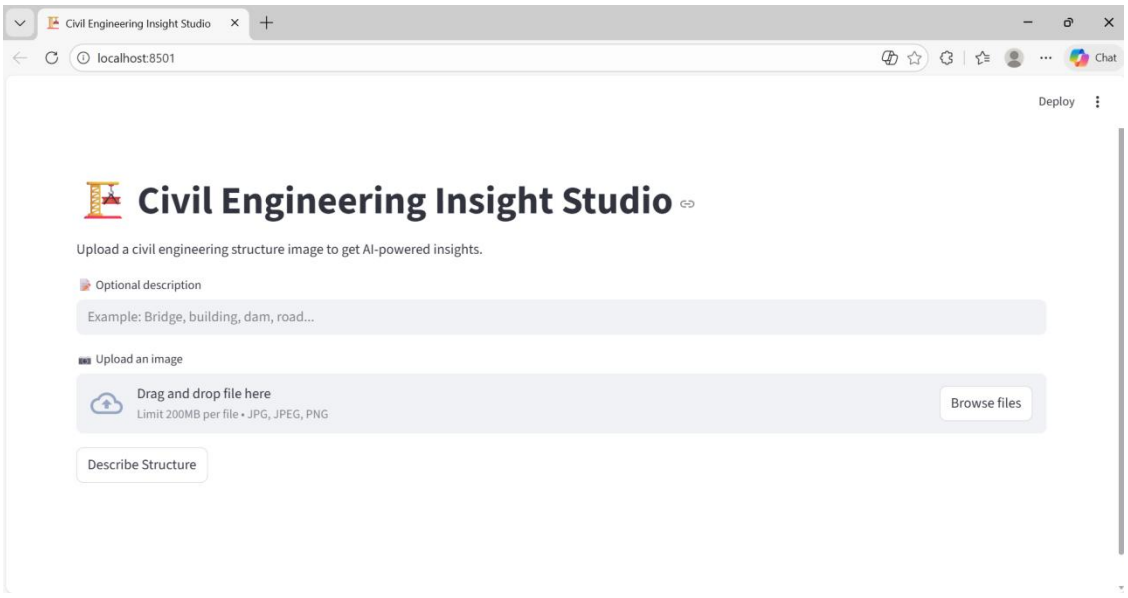
Step 1: To run the Streamlit Application we have to use the command streamlit run app.py in the terminal in path where the app.py file is located.



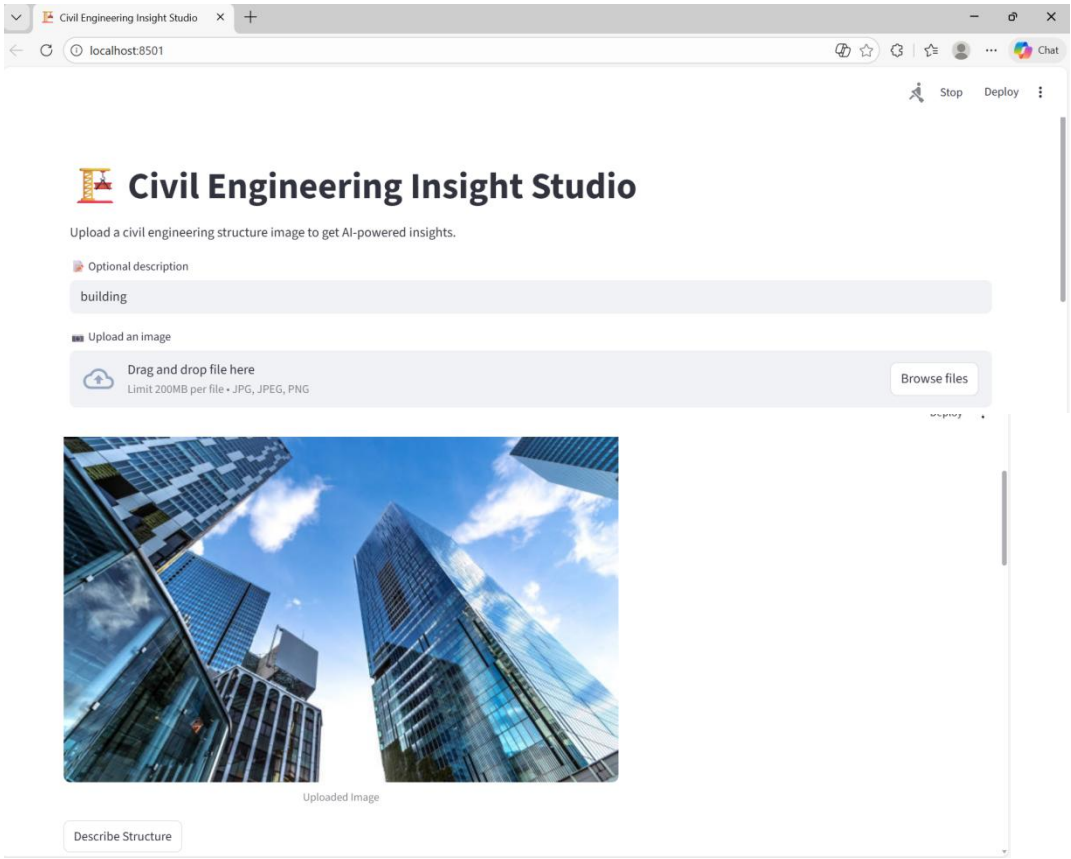
Step 2: After running the command in terminal, the code will get executed and the webpage will open directly. Another way to open webpage is that a localhost link will get generated in the terminal, we can access the webpage using that link.



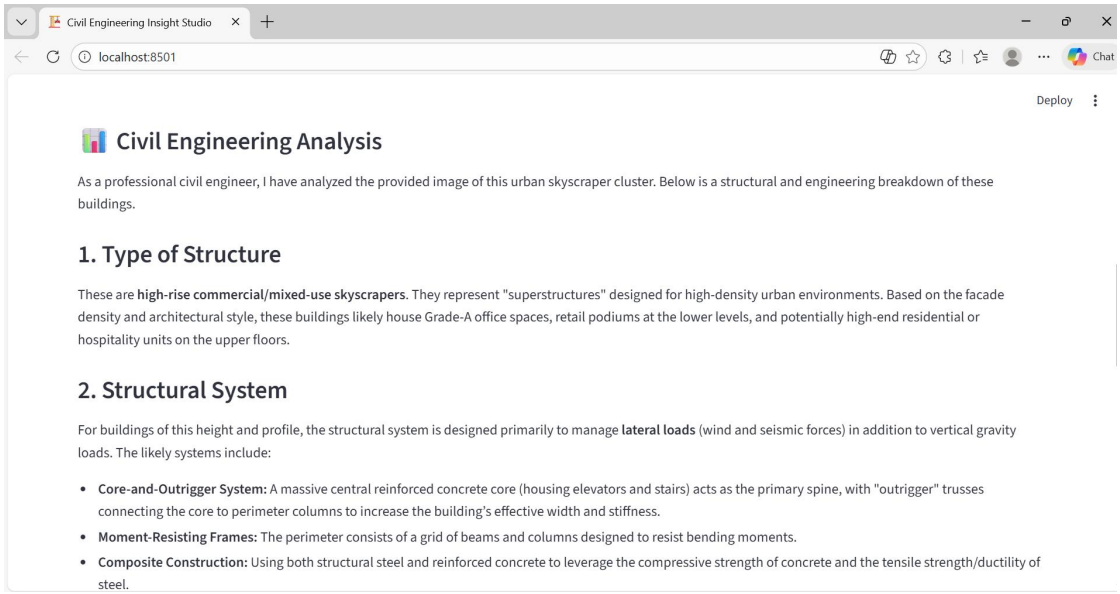
Step 3: The Streamlit webpage opens as shown in the figure given below. This is an automated webpage. No secondary HTML codes required to build this webpage. Python code itself consists the webpage building code.



Step 4: The user has to give inputs in the website such as a name of constructions. The material details of the construction should be generate. After entering the required details, the user should click on describe structure button to generate the regarding. Here I chose theglass building



Step 5: After clicking the describe structure button, in fraction of seconds the data related to construction will be generated as shown in the below figure. The construction data will be generated based on the user inputs as shown in the following Three images.



Deploy

3. Materials Used

- **High-Strength Reinforced Concrete:** Used for the central shear walls and foundation. Modern skyscrapers often use concrete with high compressive strengths (exceeding 10,000 psi).
- **Structural Steel:** Used for the perimeter framing, long-span beams, and floor decking.
- **Glass Curtain Wall:** The "skin" of the building is a non-structural curtain wall consisting of **Low-E (low-emissivity) double or triple-glazed units** held in place by aluminum or steel mullions.
- **Foundation Materials:** Massive reinforced concrete mat (raft) foundations supported by deep-seated bored piles (drilled shafts) reaching bedrock or stable strata.

4. Construction Technique

- **Self-Climbing Formwork:** The central concrete cores were likely built using hydraulic climbing forms that move upward independently of cranes.
- **Crane-Assisted Assembly:** High-capacity tower cranes (often internal or attached to the facade) are used to lift steel members and prefabricated curtain wall panels.
- **Modular Unitized Facade:** The glass panels are likely "unitized," meaning they are pre-assembled in a factory and hoisted into place, allowing for rapid enclosure of the building.
- **Just-In-Time (JIT) Logistics:** In such a dense urban environment, construction requires precise scheduling where materials arrive and are installed immediately due to the lack of on-site storage space.

5. Engineering Purpose

- **Optimization of Land Value:** The primary purpose is to maximize Floor Area Ratio (FAR) on a small footprint in high-value urban centers.
- **Urban Integration:** Providing high-density workspace and transit-oriented development to reduce urban sprawl.
- **Sustainability:** Modern facades are designed to maximize natural daylighting while minimizing solar heat gain, reducing the overall carbon footprint of the building's HVAC systems.

6. Safety and Design Features

- **Seismic and Wind Damping:** In high-wind or earthquake-prone zones, these buildings may contain **Tuned Mass Dampers (TMDs)**—large weights near the top that counteract sway—or base isolation systems at the foundation level.
- **Fire Life Safety:** These structures include pressurized stair enclosures to keep smoke out, high-flow sprinkler systems, and designated "refuge floors" where occupants can safely gather during an emergency if evacuation is not immediate.
- **Redundancy:** The structural design incorporates "load path redundancy," ensuring that if one localized element fails, the load is redistributed to prevent a progressive collapse.
- **Aerodynamics:** Note the varying shapes and setbacks of the buildings; these are often engineered and tested in wind tunnels to "confuse" the wind and reduce vortex shedding, which causes building sway.

7. ADVANTAGES AND DISADVANTAGES

Advantages

- Saves time by automatically generating detailed data
- Reduces manual effort for providing data
- Generates customized content based on user input
- User-friendly interface built using Streamlit
- Fast and efficient content generation using a pre-trained AI model
- No requirement for dataset collection or model training

Disadvantages

- Requires an active internet connection to access the AI model
- Depends on third-party AI APIs for content generation
- Limited to photo-based content only

- Output quality depends on the clarity of user input

8. CONCLUSION

The project demonstrates how AI-based tools like Civil Engineering Insight Studio can improve efficiency and accuracy in construction management by automatically identifying and analyzing building materials from site images. The system helps reduce manual effort, supports better material tracking, and provides clear insights into material type, quantity, and location within the structure. By combining image analysis with intelligent reporting, the project promotes smarter decision-making, improved resource management, and the adoption of modern digital technologies in the construction industry.

9. FUTURE SCOPE

The future scope of this project includes enhancing the system with advanced AI and computer vision techniques to improve the accuracy of material identification under different environmental conditions such as low lighting, complex backgrounds, and partially visible materials. The system can also be expanded to recognize a wider range of construction materials and structural components, enabling more detailed analysis and reporting for large-scale construction projects.

10. APPENDIX

10.1. Source Code

The source code for the Civil Engineering Insight Studio project includes the implementation of the Streamlit user interface, integration of the Gemini Flash Lite model using the Google Generative AI API, recipe blog generation logic, and the programmer joke feature. The code is written in Python and follows a modular and readable structure.

10.2. Github & Project Demo Link

Github Repository Link: https://github.com/NARASIMHA106525/Civil-Engineering--Insight-Studio

Demo Link:

https://drive.google.com/file/d/1qGJTwDCcrk_z-MSA6DZT14qvd8r7C3jH/view?usp=sharing