PROJECT REPORT

Civil Engineering Insight Studio

Submitted by: Patnala Venu Madhavi

Institution: Sri Vasavi Degree College

Mentor: L. Lakshmi Narayana

Internship Program: SmartBridge Virtual Internship Program

---

1. INTRODUCTION

1.1 Project Overview

The "Civil Engineering Insight Studio" is an automated structural analysis system designed to describe civil engineering structures from images. By leveraging Artificial Intelligence and Large Language Models (LLMs), it generates detailed descriptions of structural types, materials, dimensions, and construction methods. This tool addresses the subjective and time-consuming nature of manual site documentation by providing instant, objective insights.

1.2 Purpose

- To automate the description of civil engineering structures using AI.
- To assist engineers in identifying construction materials such as concrete, steel, and bricks.
- To streamline project progress documentation for construction firms.
- To provide detailed structural analysis of components like beams, columns, and trusses.

---

2. IDEATION PHASE

2.1 Problem Statement

Civil engineers often manually describe structures based on images, which is time-consuming and prone to human error. There is a critical need for an automated tool that can analyze visual data to generate reliable, insightful documentation for effective project communication.

2.2 Empathy Map Canvas

- **User Thinks:** What materials are used here? Is this bridge structurally sound? How much progress has been made this week?
- **User Feels:** Overwhelmed by manual documentation tasks and worried about the subjectivity of site reports.
- **User Says:** I need an automated way to list these structural components. I want a report on material types and locations.
- **User Does:** Takes site photos, manually writes reports, and consults multiple experts for component identification.

2.3 Brainstorming

Selected Idea: **AI-Powered Structural Insight Studio**. We chose this approach because it uses multimodal LLMs to dynamically interpret complex visual engineering data rather than relying on static templates.

---

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

1. User uploads a construction site or bridge image.
2. User provides an optional text prompt (e.g., "Analyze material types").
3. The system processes the image using the Gemini API.
4. A detailed engineering report is generated.
5. The engineer reviews the components and documentation.

3.2 Solution Requirements

- **Functional:** Accept image uploads (JPG/PNG), process visual data via AI, and display structured reports including material quantity and location.
- **Non-Functional:** Professional engineering terminology, fast response times, and secure handling of project images.

3.3 Technology Stack

- **Python:** Core programming language.
- **Streamlit:** For the web interface.
- **Google Gemini API:** Large Language Model for vision and text generation.
- **Pillow (PIL):** For image processing.
- **Python-dotenv:** For secure API key management.

---

4. PROJECT DESIGN

4.1 Solution Architecture

- **User Interface:** Streamlit-based dashboard for image uploading.

- **Backend:** Python logic to handle environment variables and image formatting.
- **AI Engine:** Gemini model to interpret civil engineering features.
- **Flow:** User → Streamlit UI → Image Data Setup → Gemini API → Structural Report → UI Display.

---

5. FUNCTIONAL TESTING

- **Test 1 (Bridge):** Input image of a truss bridge. **Output:** Successfully identified beams, trusses, and structural steel material.
- **Test 2 (Site Progress):** Input image of a building foundation. **Output:** Correctly identified reinforced concrete and excavation phase.

---

6. RESULTS

The system generates a comprehensive output including:

1. **Structure Type:** (e.g., Suspension Bridge, High-rise)
2. **Materials:** (e.g., Reinforced Concrete, Masonry)
3. **Components:** (e.g., Columns, Beams, Abutments)
4. **Engineering Challenges:** (e.g., Observed corrosion or cracks)

---

7. CONCLUSION

The "Civil Engineering Insight Studio" successfully demonstrates the application of Generative AI in the civil engineering domain. By integrating the Gemini API, the tool reduces manual effort, standardizes documentation, and provides engineers with a reliable secondary analysis for structural assessments.

---

8. APPENDIX: SOURCE CODE

```
import streamlit as st
import google.generativeai as genai
import os
from PIL import Image
from dotenv import load_dotenv

load_dotenv()
genai.configure(api_key=os.getenv("GOOGLE_API_KEY"))

# Function to get response from Gemini model
def get_gemini_response(input_text, image, prompt):
    model = genai.GenerativeModel('gemini-3-flash-preview')
    response = model.generate_content([input_text, image[0], prompt])
    return response.text

# Image setup function
def input_image_setup(uploaded_file):
    if uploaded_file is not None:
        bytes_data = uploaded_file.getvalue()
        image_parts = [{"mime_type": uploaded_file.type, "data": bytes_data}]
        return image_parts
    else:
        raise FileNotFoundError("No file uploaded")

# Streamlit App UI
st.set_page_config(page_title="Civil Engineering Insight Studio")
st.header("🏗️ Civil Engineering Insight Studio")

input_text = st.text_input("Input Prompt (Focus Area):", key="input")
uploaded_file = st.file_uploader("Choose an image...", type=["jpg", "jpeg", "png"])

if uploaded_file is not None:
    image = Image.open(uploaded_file)
    st.image(image, caption="Uploaded Image", use_container_width=True)

submit = st.button("Describe Structure")

input_prompt = """
You are an expert Civil Engineer. Analyze the image and describe:
1. Structure Type and Materials.
2. Dimensions and structural components (beams, columns, etc.).
3. Construction methods and notable features or challenges.
"""

if submit:
```

```
image_data = input_image_setup(uploaded_file)
response = get_gemini_response(input_text, image_data, input_prompt)
st.subheader("Structural Report")
st.write(response)
```