AI Lab Report Agent for CE331

- **Name:** Yash Pathak

- **University:** Indian Institute of Technology Kanpur (IIT Kanpur)

- **Department:** Civil Engineering

1. Project Objective

This project automates the manual, time-consuming task of writing university lab reports for the course **CE331: Principles of Geoinformatics**.

The goal was to build an AI agent that could reason, plan, and execute a multi-step process to generate a complete, formatted, and accurate lab report. The agent functions as an AI collaborator: the user provides the specific "human-in-the-loop" inputs (the lab title, key concepts, and raw experiment data), and the agent handles all the research, writing, data formatting, and file conversion, delivering a finished PDF.

2. AI Agent Architecture

This prototype is a **multi-agent system** that fulfills the assignment's "Multi-agent collaboration" bonus. It uses a "human-in-the-loop" pipeline where a central script coordinates four specialized agents to achieve the final goal.

Interaction Flow

The workflow is triggered by the user in a single script (`Cell 4` in the Colab notebook) and proceeds as follows:

1. **User Input:** The user provides three critical pieces of information:

   - **Lab Title:** (e.g., "Lab 4 Levelling")

   - **Key Concepts:** (e.g., "Rise and Fall Method, Height of Collimation, misclosure")

   - **Raw Experiment Data:** (The user pastes their raw data tables from the experiment).

2. **Agent 2 (Tool Use) - Image Search:**

   - The `Lab Title` is passed to the **Image Agent**.

   - It uses the Google Custom Search API to find a relevant image URL for the lab's equipment or methodology.

3. **Agent 1 (Knowledge) - RAG:**

   - The `Key Concepts` are passed to the **Knowledge Agent**.

   - This agent performs a semantic search on a `ChromaDB` vector store (which was built from the course textbooks and lecture notes).

   - It retrieves the 5 most relevant chunks of text (definitions, formulas, etc.).

4. **Agent 3 (The Writer) - Fine-Tuned LLM:**

   - A single, large prompt is constructed. This prompt contains the `Lab Title`, the `Raw Experiment Data`, the `image_url` from Agent 2, and the `book_context` from Agent 1.

- This combined prompt is fed to the **Writer Agent**.

- This agent uses its specialized, fine-tuned training to *synthesize* all these inputs into a single, cohesive, and perfectly formatted Markdown text.

5. **Agent 4 (The Publisher) - PDF Converter:**

   - The final Markdown text from Agent 3 is passed to the **Publisher Agent**.

   - This agent converts the Markdown into a PDF and saves it to disk.

6. **Final Output:** A complete, formatted `Lab_Report_...pdf` file.

3. Components & Models Used

Agent 1: The "Knowledge Agent" (RAG)

- **Component:** `ChromaDB` (persistent vector store).

- **Model:** `sentence-transformers/all-MiniLM-L6-v2`

- **Reason for Choice:** This is the industry-standard model for RAG. It is lightweight, fast, and highly effective at mapping the *semantic meaning* of the user's "Key Concepts" query to the content of the document chunks. As we discovered in our evaluation, providing specific `Key Concepts` (like "Rise and Fall Method") was critical to avoiding hallucinations and retrieving the correct context.

Agent 2: The "Image Agent" (Tool Use)

- **Component:** `googleapiclient` (Google Custom Search API).

- **Reason for Choice:** Lab reports require diagrams and images. This agent fulfills the "External integrations" bonus by calling a custom tool to find a relevant image URL, which is then passed to the Writer Agent.

Agent 3: The "Writer" (The Fine-Tuned Model)

- **Component:** `PeftModel` (from Hugging Face `peft` library).

- **Base Model:** `mistralai/Mistral-7B-Instruct-v0.2` (loaded in 4-bit with `bitsandbytes`).

- **Fine-Tuned Adapter:** `checkpoint-6` (the result of our 3-epoch LoRA training).

Justification for Fine-Tuning This Target:

This was the most critical decision in the project, as required by the assignment. A base, general-purpose LLM is not sufficient for this task.

1. **Adapted Style:** A base model does not know the specific academic structure required for a CE331 lab report (e.g., `## Objective`, `## Methodology`, `## Results and Discussion`, `## Conclusions`). Our fine-tuning on 5 example reports taught the model this rigid **stylistic structure**, making its output reliable.

2. **Task Specialization:** The most complex and valuable task for this agent is **data transformation**. The agent was specifically trained on examples where the prompt contained messy, raw, pasted-in data, and the expected output was that same data, perfectly re-formatted into clean Markdown tables. This is a highly specialized skill that our fine-tuned model now excels at.

3. **Improved Reliability:** The fine-tuning makes the agent's output predictable. It *always* generates the correct sections in the correct order, which is essential for a reliable automation tool.

Agent 4: The "Publisher" (PDF Converter)

- **Component:** `markdown-pdf` library.

- **Reason for Choice:** This agent completes the automation pipeline. It takes the final, structured Markdown text from the Writer Agent and converts it into the final deliverable: a portable, professional `.pdf` file.