GLOBAL MIT AI HACKATHON

Are you looking to experiment with AI and engage with AI talents around the world?

🗓️ May 2–3, 2025 24 h | 🕙 Starts around 10:00 AM ET | 🌐 Online

AI Copilot for Renewable Energy Data Rooms

---

1) Motivation / Goal to Achieve

Renewable energy developers face an overwhelming volume of repetitive documentation — land leases, permits, interconnection agreements, contracts — often embedded in lengthy PDFs or Excel files. These documents must be parsed manually to extract relevant information, which is then copied into firm-specific checklists or due diligence reports. This manual process is time-consuming, error-prone, and a bottleneck for project origination and financing.

This challenge aims to build an **AI copilot for renewable energy developers** that:

- Allows users to upload an entire data room.
- Enables natural language querying of documents (like ChatGPT).
- Returns **trusted, reference-backed answers** — with exact quotes, page numbers, and clickable links.
- In a second step, allows developers to upload company-specific checklists, which the AI pre-populates by extracting corresponding values from the documents, always referencing the source to build user trust.

---

2) Functionalities / Features

- Upload data room files (PDFs, Excels, etc.).
- Natural language querying across all uploaded content.
- Return answers with:
  - **Exact source reference** (quote, page number, clickable link if UI permits).
- Optional advanced feature:

Organizing team: Shreeansh Agawal, Linn Bieske, Andrea Jimenez, Kai Wiederhold, Lisa Sklyarova

    - Upload custom checklists and **auto-populate fields** using retrieved data and references.

---

3) Hints and Resources on How to Build It

- **Document parsing:** Use tools like GROBID or PDFPlumber for structured text extraction from PDFs.

- **Large-scale retrieval:** Use a vector database (e.g., **Weaviate**, **Pinecone**, or **FAISS**) to index embeddings from the data room content (chunked + metadata like file/page).

- **RAG architecture:** Combine OpenAI's GPT-4 or Claude with a **Retrieval-Augmented Generation (RAG)** setup to reduce hallucinations and ensure answers cite specific document excerpts.

- **Checklist mapping:** Use fuzzy matching or simple NLP techniques to align checklist labels with document content, or allow manual mapping if fields vary widely across companies.

- **Evaluation UI tools:** Consider using Streamlit or Gradio to build a fast, user-friendly prototype interface.

---

4) Evaluation Criteria for the Prototype

Criterion	Goal
Reliability & Accuracy	Answers must cite correct quotes with **low hallucination rate**.
Scalability	Must handle **large data rooms** (10,000–15,000 pages or 100+ files).
Speed	Query responses should return in **<10 seconds** ideally.
Checklist Pre-fill Accuracy	High match rate for checklist fields and reliable source reference.
Optional: Cost Efficiency	Lower inference + indexing costs are a plus for enterprise adoption.

Organizing team: Shreeansh Agawal, Linn Bieske, Andrea Jimenez, Kai Wiederhold, Lisa Sklyarova