**Project Plan: AI-Powered Course Module Generator**

**1. Overview**

This project develops an AI chatbot that generates **structured course modules** based on **teacher-provided resources**(PDFs, articles, web links) and **example formats**. The AI will **automate lesson creation**, refine content based on teacher feedback, and dynamically **update a website** with new modules.

**2. Timeline (8 Weeks)**

| **Phase** | **Task** | **Duration** | **Deliverable** |
| --- | --- | --- | --- |
| **Week 1** | Model Selection & Setup | 1 week | LLM (Vicuna) running locally |
| **Week 2** | PDF & Web Extraction | 1 week | Text extraction pipeline (PyMuPDF, BeautifulSoup) |
| **Week 3-4** | AI-Powered Module Generation | 2 weeks | API for structured lesson generation |
| **Week 5** | Website Development | 1 week | Website prototype (React/Next.js) |
| **Week 6** | Integration & Auto-Updates | 1 week | AI-generated content deployment to website |
| **Week 7** | Feedback & Refinement | 1 week | Teacher feedback system |
| **Week 8** | Testing & Final Deployment | 1 week | Fully functional MVP |

**3. Value Over Commercial Models**

* **Multi Media Content Input**: Chat GPT free will not accept PDFs, Images, or Tables as Inputs
* **Multi Media Content Output**: Webpages, PDFs, Images, or Tables can be integrated in the LLM response
* **Webpage Integration**: Responses will be used to populate a webpage
* **Persistent Memory**: Context will be remembered by the LLM to reduce iteration time

**4. Existing Solutions**

* **Khanmigo by Khan Academy (Khan, 2024)**
  + **Benefit: Creates lesson plans from its own content**
  + **Limitation: Does not allow for teachers to provide input of specific sources**
  + **Link:** [**khanacademy.org/teacher/khanmigo-tools/lesson-plan?platform=KhanAcademy**](http://khanacademy.org/teacher/khanmigo-tools/lesson-plan?platform=KhanAcademy)

**5. Core Features**

**(a) Resource Processing**

* Extract text, image, and tables from **PDFs, articles, and web pages**.
* Use **PyMuPDF,** **PDFPulmber, Pillow, BeautifulSoup for extraction**

**(b) AI Module Generation**

* Generate structured lessons (Topic, Examples, Q&A).
* Use **LLaVA** for generation

**(c) Website Integration**

* Deploy modules automatically.
* Allow teacher feedback & updates.

**6. Tech Stack**

* **LLAVA:** Pretrained model accepting both Image, Table, and Text Inputs
* **BeatifulSoup:** Extract Images, Text, and Tables from Website
* **PyMuPDF:** Extract Text from PDFs
* **PDFPlumber & Pillow:** Extract Images and Tables from PDFs
* **Flask:** Publish LLM Response to Webpage

**5. Next Steps**

* Set up LLaVA model on a HCP. If necessary, adjust plans based on the HCPs configuration
* Build text extraction pipeline (PyMuPDF, BeautifulSoup).
* Develop AI-generated module API.
* Deploy website & feedback system.