**HACKTHON PROJECT PHASES TEMPLATE FOR AI POWERED PDF ANALYZER FOR STUDENTS PROJECT**

**Project Title:**

**Study-Mate: An AI-Powered PDF-Based Q&A System for Students**

**Team Name:**

**CDN**

**Team members:**

* **B.Chandu**
* **K.S.S.Dinesh**
* **B.Nithin**

**Phase-1: Ideation**

**Objective:**

To build an AI-powered PDF analyser that helps students easily understand and interact with their study materials. Students can simply upload their academic PDFs, and the AI will read through the content, summarize key points, and be ready to answer any questions they have about the material or topic. The goal is to save time, reduce stress, and make studying smarter and more interactive.

**Main Insights:**

1. **Problem Statement:**

* Many Students face difficulties in comprehending the topics present in their academic pdfs
* They often struggle to grasp the content and may miss out on important topics or key concepts.

**2.** **Solution:**

To help such students, we have developed an **AI-powered** PDF analyser using **Streamlit** for the interface, **Hugging Face**  for language processing, and **PyMuPDF** for extracting and reading text from uploaded PDFs. We also integrated **FAISS** and **AI** to convert the content into vector embeddings and perform intelligent PDF analysis.

**3. Targeted Users:**

* **College and university students** who struggle with long, overwhelming academic PDFs**.**
* **Students looking for last-minute summaries** to aid in quick revision before exams or assignments
* Students who are struggling to **understand complex topics**.

**Phase-2 Requirements:**

**Objective:** Discuss the technical and functional aspects required.

1. **Technical Requirements:**

* Programming Languages: **Python**
* Backend: **FastAPI**
* Frontend: **Streamlit**
* Database: **MongoDB**
* LLM: **Hugging Face**

**2.Functional Requirements:**

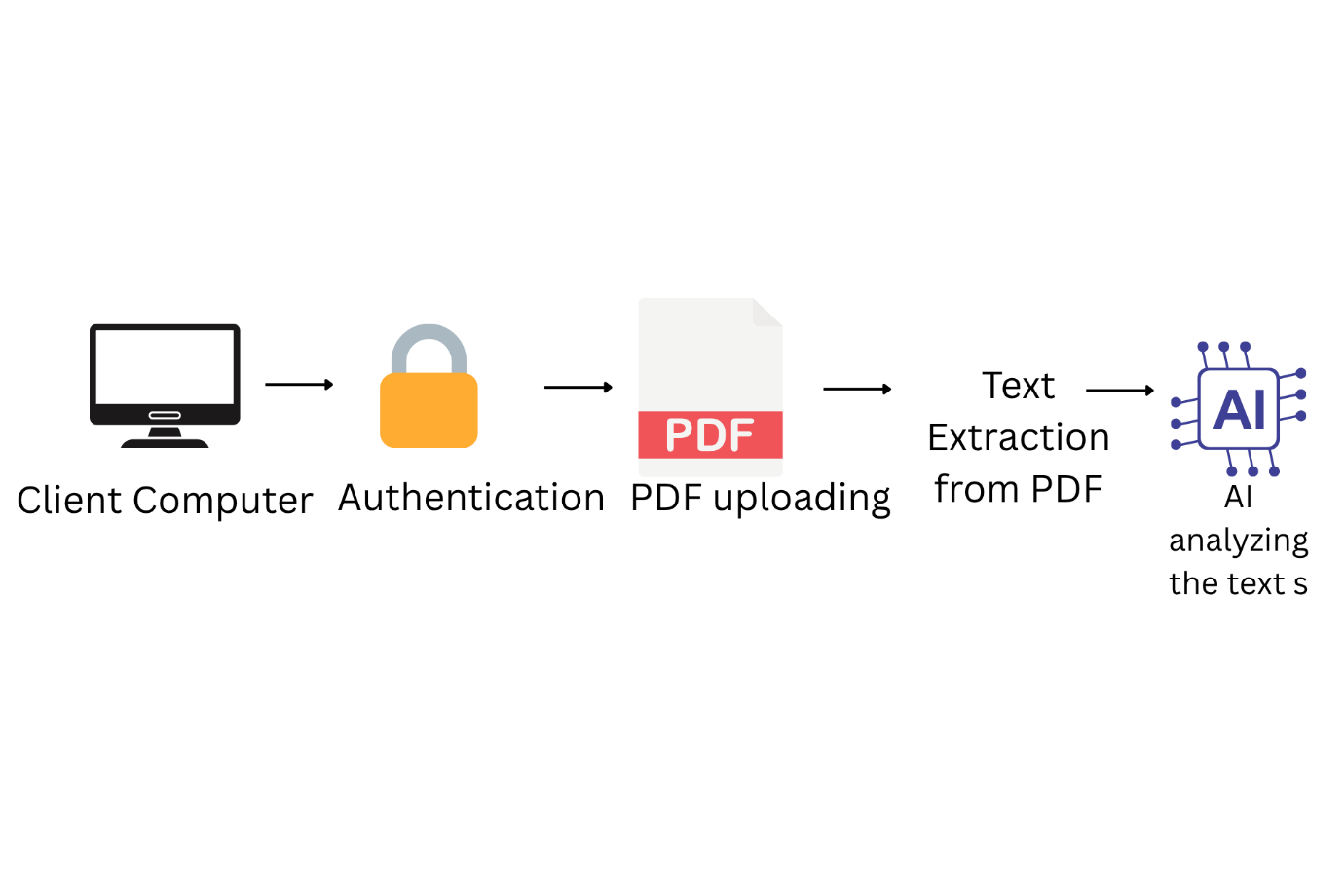
* Module to upload the pdf
* Module with the ability to extract the text from the pdf and analyze the text.
* AI with the ability to understand the extracted text and answer appropriate to the user.
* Database to collect the user data and authenticate with their details.

**3.Challenges:**

* Crashing of the backend server.
* Integrating the modules.
* Optimizing AI model performance

**Phase-3 Project Design**

**Objective:** Develop the user interface and computer architecture.

1. **Computer Architecture**

**Working:**

**1.**User first need to Login with their Account and Password.

**2.**After they logged in they can start uploading their PDFS

**3.**Then PyMuPDF extracts the text from the uploaded PDFS

1. The integrated FAISS converts the chunks of text into Vector’s
2. AI then analyses the text extracted from the pdf and understands it and answers appropriately to the user questions
3. **UI/UX Considerations:**

* **Minimalist, user-friendly Interface** for simple and easy access
* AI generative texts for simple understanding

**PHASE-4: Project Planning**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Task** | **Priority** | **Duration** | **Assigned**  **To** | **Dependencies** | **Expected**  **Outcome** |
| **Sprint-1** | Database, Backend  development | High | 6hrs | B.Chandu | MongoDB,  Python | Collects user data |
| **Sprint-1** | API integration | High | 3hrs | K.Dinesh | Hugging Face | API  integration  for AI |
| **Sprint-1** | UI/UX development | LOW | 2hrs | B.Nithin | Streamlit | User friendly fronted |
| **Sprint-2** | Error handling and debugging | High | 3hrs | B.Chandu  B.Nithin  K.Dinesh | VS Code | Finding the  errors  and  debug |
| **Sprint-2** | Testing | Mid | 2hrs | K.Dinesh | Postman | Testing  Website |

**FINAL SUBMISSION:**

**1.Project Report Based Template**

**2.Demo Video**

**3.GitHub/Code Repository Link**

**4.Presentation**