Talk to Tartan

Team Members: Christine Muthee, Daniel Owolabi, Tewodros Idris, Brian Ebiyau

Carnegie Mellon University Africa

Faculty Advisor: Charlie Wiecha

Date: 17th September 2024

Project purpose & goals:

We will simplify the way CMU-Africa students plan their courses and access information about campus events.

Methodology and Initial Approach:

Hill	Epics	User stories
CMU-Africa students can get information about CMU-Africa within one platform.	As a CMU-Africa student, I can access a user-friendly tool to plan my degree and academic schedule without needing to navigate complex university systems	As a student, I need to input my interests and receive a personalized degree plan. why?
		As a student, I can inquire about course prerequisites and receive accurate information. why?
		As a student, I want to ensure that my plan aligns with the school's requirements for graduation. why?
	As a student, I can receive personalized guidance on events happening on campus without looking through multiple emails.	As a student, I want to easily find current event details without searching through multiple sources.
		As a student, I want to be informed about events that fit my schedule.
		As a student, I want to

		discover events that align with my interests.

Proposed Solution.

1. Retrieval-Augmented Generation (RAG) System: We'll implement a RAG system to enhance the LLM's responses with accurate, up-to-date information from our database. This will involve:
    a. Vector database (e.g., Pinecone or Weaviate) for efficient similarity search
    b. Embedding model (e.g., BERT, Sentence-BERT or ColBERT) for converting text into vector representations
    c. We will utilize a state-of-the-art LLM (such as GPT-3 or GPT-4) or an open-source model like Llama 3.1 for natural language processing and generation.
2. Web Scraping: We will build a system that scrapes data from the course catalog, schedule of classes and the student's handbook. We'll use Python libraries such as BeautifulSoup or Scrapyto gather current information from CMU websites, ensuring our database stays up-to-date.
3. Database: A structured database PostgreSQL will be used for storing structured data (e.g., course information, event details), while MongoDB will be used for unstructured data storage.
4. Backend Framework: We'll use Fastapi (Python) for the backend, providing a robust foundation for our application in integrating with other systems through API calls. RESTful APIs will be created to facilitate communication between the frontend, backend, and the LLM service.
5. Frontend Framework: We'll develop the user interface using Streamlit or Dash, ensuring a responsive and interactive experience for users.
6. Containerization: Docker will be used for containerization, ensuring consistency across development and production environments.
7. Version Control: Git will be used for version control, with GitHub as our repository hosting platform. We will also incorporate github actions for CICD.
8. Testing Frameworks: We'll implement Pytest for backend testing to ensure code quality and reliability.

Possible Major Products

1. Intermediate Report: Mid-project progress update, including: Summary of work completed, preliminary findings, challenges faced and mitigation strategies and updated timeline for remaining tasks
2. AI-Powered application: A fully functional web application that provides CMU-Africa students and applicants with easy access to information about courses, events, and admissions.
3. Final Report: A comprehensive documentation of the project, including: executive summary, detailed methodology, technical architecture, challenges encountered and solutions implemented, recommendations for future enhancements.

4. User Guide: Detailed manual for end-users (students and applicants) on how to effectively use the information platform.
5. Technical documentation through a README on GitHub: Technical documentation on how to maintain, update, and troubleshoot the system.

Tasks, Approximate Timeline, and Resources

Sprint	Key Tasks	Main deliverables	Core Resources
1	Background research Design system and set up a mockup environment. Data collection planning - web scraping and questionnaires. Initial work on intermediate report	User research findings System mockup Populated initial report	School documentation Backend and frontend frameworks
2	Data scraping Refinement of environment setup Initial database design	Data catalog Development environment	Github/Docker accounts Web scraping tools
3	Develop data pipeline LLM selection & integration Continue intermediate report	Preliminary RAG pipeline Draft intermediate report	LLMs API tokens/Compute
4	Frontend-backend connection Build event retrieval pipeline Finalize intermediate report	Working prototype Final intermediate report	Frontend & backend framework Gmail API
5	LLM alignment for degree plan Begin user guide	Integrated RAG pipeline Draft user guide	Hardware compute
6	System testing & refinement Finalize documentation Prepare presentation & demo	Final system Final report User guide Technical documentation Project presentation	Testing tools Presentation software

Note: The AI-powered application development spans all sprints. The intermediate report is primarily worked on in sprints 3-4, while the final report, user guide, and technical documentation are focused on in sprints 5-6.

Risks and Mitigation

1. Limited access to Tokens and Compute to finetune and test the LLMs.
    a. Utilize Open Source LLMS.
    b. Make enquiries of tokens and compute on time with a rough estimate of the cost and time spent.
2. Integration challenge between LLM and RAG system.
    a. Use libraries that are well documented and have large community
    b. Set time aside enough time to figure any integration conflicts that may arise.
    c. Start with a simple solution before making it complex.
3. Insufficient data to integrate all functionalities.
    a. Provide incentives to students to encourage them to fill forms and questionnaires.
    b. Generate synthetic data using LLM.
4. Risk: Privacy concerns with student data
    a. Ensure compliance with data protection regulations; implement robust security measures