Project Planning Road Map Evaluation

MS EAI Capstone, CMU-Africa, Fall 2024

Project Title: Assistant for Effortless Cloud Resource Management and Experimentation

Team: {Carmel Prosper Sagbo, Ellon Berhanu, Birhanu Shimelis Girma, Ngoga Alexis}Carnegie Mellon University Africa

Advisor: Charles Wiecha, Carnegie Mellon University, cwiecha@andrew.cmu.edu

Date of submission: 17th September, 2024

The Planning Roadmap

1) Project Purpose and Goals:

With the rise of AI and cloud technologies, social goods services providers and startups, as well as big companies, have the requirements to deliver high-performance and efficient solutions to their clients. However, with the increasing demand and requirements, it is challenging for these service providers or individuals to run their experiments on-premise or using local resources due to the high financial cost, lack of time, and quality of specialists in infrastructure management. It is therefore important to find an approach for them to cost-effectively and efficiently analyze different architecture propositions to effortlessly run their business within budget limits and systems requirements. The current solution leverages the state-of-the-art solution through the Large Language Models to create a user-friendly web platform that assists startups and company developers in generating effortless architecture fitting a limited budget with high performance using the recommended solution based on three different cloud service providers: AWS, GCP, and Azure.

Goals and Intended Outcomes:

1. Automate cloud resource allocation and management.
2. Provide real-time cloud architecture recommendations.
3. Simplify experimentation workflows for cloud services.
4. Reduce operational costs and time spent on manual cloud management.

2) For the project roadmap to effectively design and build the above-proposed solution, we narrowed each epic of our work into different user stories coupled with some prior requirements and tasks to fulfill that we will highlight later. A big picture of the detailed plan, including Hill, Epics, and user stories, is described as follows:

Key Milestones:

Hills	Epics	**User Stories**
A startup can buy or utilize cloud resources and run experiments on the cloud effortlessly and	As a startup developer, I need to be able to find and purchase the recommended cloud services plan suitable	- As a user, I want to generate recommended and efficient architecture

cost-effectively	for my project and budget limit.	for my project requirements and budget. - As a user, I need to compare the cost for different architectures including resources from different cloud providers (AWS, GCP, Azure) and the pricing models for the most cost-efficient option. - As a user, I want to download a detailed report on the cost projections for a chosen cloud architecture and services.
	As a startup developer, I need easy step-by-step guidance and assistance on how to efficiently run experiments using my selected cloud service	- As a user, I want to choose a given architecture and receive a getting-started guide with the proposition of language to run and configure the environment for deployment. - As a user, I want a code generator that provides sample scripts to automate deployment on AWS, GCP, or Azure based on a selected Language. - As a user, I want to receive recommendations on how to optimize my provisioned cloud resources based on ongoing performance metrics (using RAG and LLM).

Prior Task on the Roadmap:

- Environment setup for the team.
- Collection, Preprocessing, and Validation of needed Data for the system.

Conjectured Solution: Web interface with the above functionalities integrated

Technologies: AWS and Azure for cloud management; Terraform for orchestration.

Software Frameworks: React (UI), Flask/Node.js (backend), TensorFlow/PyTorch (optimization models).

3) The major product delivery will include the following:

I. **Final Web Platform:** The Final Product is a user-friendly web platform that enables startups and developers to effortlessly access, compare, and purchase cloud services.
II. **Project GitHub Repository**: A well-documented repository containing all the code, scripts, and set up instructions. This will serve as the primary deliverable for all the product developed during the project.
III. **User Manual:** A detailed guide to help users navigate and utilize the system. It will include instructions on how to find and purchase cloud services, set up experiments, and interpret cost estimates.
IV. **Final Report**: A document summarizing the project's objectives, development process, outcomes, and how it meets the requirements set out in the Hill, Epics and user stories.
V. **Intermediate Report**: An update provided at a midpoint in the project timeline, detailing the progress and any preliminary findings or results.
VI. **Demo Video**: Visual demonstration that showcases the product's capabilities and how to use it by taking an example of user interaction.

4) Tasks, Approximate Timeline, and Resources

Sprint	Tasks/Deliverables	Timeframe	Resources
Sprint 1	Data collection, preprocessing, and system architecture design	Sept 16 - Sept 22	Team meetings, cloud service APIs (AWS, Azure), architecture planning tools
Sprint 2	Develop the cloud resource management MVP and backend integration	Sept 23 - Oct 6	React (frontend), Flask/Node.js (backend), Cloud SDKs
Sprint 3	Implement resource optimization engine (initial version)	Oct 7 - Oct 20	TensorFlow/PyTorch (optimization models), cloud infrastructure
Sprint 4	Build experiment management dashboard UI and backend	Oct 21 - Nov 3	React, MongoDB/PostgreSQL, API development tools
Sprint 5	Finalize cost-saving analytics and real-time feedback features	Nov 4 - Nov 17	Data visualization libraries, Python, Cloud APIs
Sprint 6	Testing, bug fixes, optimization, and documentation	Nov 18 - Dec 1	Testing frameworks (JUnit, PyTest), cloud sandbox,

			documentation tools
Sprint 7	Final report, User manual, Github Repository, client demo, and project handover	Dec 2 - Dec 5	Word processing tools, presentation software, client meetings

5) Risks and Mitigation

The major risk in this product development is related to the ability of the team to collect the data, extract, treat, and load for model fine-tuning. They are a lot of data available about each of the selected cloud providers, but at least some of the team members have to be skilled enough at manipulating cloud resources to mitigate the process and provide guidance on the solution development. A major advantage here is that two of the team members have basic or intermediate interactions with cloud resources and are familiar with the concepts being discussed. Additionally, the code generation part of this web platform solution requires, in terms of technology, finding a model or LLM tools to generate valid YAML or JSON code or any other particular language mainly used in cloud resource management such as Java, Python, or Harchicorp Language. Moreover, variability in the schedule of team members and the client (Prof Charlie and Tim) might delay some delivery, so we need a regularly planned meeting and task management tools for the agile requirements to track the processes. We also need to implement clear communication channels like Slack for communication and notification and Jira for Sprint and task tracking for each feature being developed and tested.