Deployment Report

Report Period: March 26, 2024 – April 03,2024

Project Title: ARA (AI-powered Research Assistant)

Guide: Dr. D. J. Chaudhari

Report Prepared By: Kaustubh Warade
Aditya Deshmukh
Devansh Parapalli
Yashasvi Thool

Executive Summary: The deployment phase for the project has been successfully completed, with the system now live and operational on AWS using a CI/CD pipeline, Docker containers with custom build scripts, and Kubernetes orchestration to address scalability. Challenges like ensuring environment consistency, handling large data volumes with a scalable PostgreSQL database, and mitigating container failures were addressed. With a total expenditure of Rs. 2900 against a budgeted Rs. 1000 for hosting, risk mitigation strategies are in place for potential issues like container failures and high traffic. Key recommendations include continuous monitoring, robust logging, regular risk assessment reviews, considering auto-scaling, and periodic financial reviews to ensure optimal performance and cost-effectiveness.

Current Phase Progress:

Task Description	Deployment phase
Scheduled Completion	April 03,2024
Actual Completion	April 03,2024
Status	Completed
Remarks	

Accomplishments:

- Implementation of Continuous Integration / Continuous Deployment pipeline was successfully done.
- Custom building script for the "Runners" was created and used to deploy ML models along with web-scraper to a cloud platform. *docker-compose* is used to create containers with the requisite software stack.
- Scalability issues were mitigated by ensuring multiple running containers orchestrated using Kubernetes. The containers are running on AWS under region ap-south-1.

Challenges & Mitigation:

- The ML models require a specific version of the software stack as developed to ensure correct working. docker-compose allows to create identical containers minimizing the difference between development and deployment environments.
- Handling large volumes of data required use of scalable databases. A version of PostgreSQL was hosted along with requisite APIs and plugins to ensure that the data is stored properly.
- Containers are prone to failure, and do not have auto-restart functionality built into it, out of the box. A container orchestrator (Kubernetes) was used, along with a heartbeat endpoint to ensure all containers are functioning as expected and to restart slow or failed containers.

Financial Summary:

Description	Hosting Charges
Budgeted Amount	Rs. 1000
Amount Spent This Week	Rs. 850
Total Expenditure	Rs. 2900

Risk Assessment:

- The containers may fail when deployed. To mitigate this, a heartbeat endpoint is used and container orchestration is asked to keep at least 2 working containers in the pool.
- The volume of requests may become too large during peak times. To mitigate this, the orchestrator is required to increase the number of containers when the current pool usage capacity reaches 80% of total available capacity.
- The connection to model may fail anytime. To mitigate this, all requests to the containers are stateless, and can be continued from another container.

Conclusions and Recommendations:

In conclusion, the deployment phase has been successfully completed, with the system now live and operational on the cloud platform. The implementation of CI/CD, containerization, and container orchestration has addressed scalability and consistency issues, while the use of a scalable database and heartbeat monitoring mitigates potential risks.

Furthermore, it is recommended to implement robust logging and monitoring mechanisms to quickly identify and resolve any issues that may arise. It is also recommended to regularly review and update the risk assessment and mitigation strategies to address any new or emerging risks.

Approval:

Dr. D. J. Chaudhari

Project Guide

Assistant Professor, CSE Department

Government College of Engineering Nagpur

Sector-27, Mihan Rehabilitation Colony

Khapri, Nagpur

441108

Date: April 05, 2024