Milestone 2 : *MLOps Infrastructure & Advanced Training Workflows* -

Building Atomic Containers, Versioned Data Pipelines, and Scalable Computing Solutions

This milestone focuses on establishing the core infrastructure necessary for an MLOps pipeline. Teams are expected to create functional environments, containerized components, and a versioned data management strategy to ensure their work is reproducible and scalable.

For teams utilizing Large Language Models (LLMs), the emphasis is on setting up a RAG workflow, including data chunking and integration with a vector database. Teams focusing on computer vision or other modalities will develop fine-tune models, and conduct experiments to optimize performance.

By the end of this milestone, teams will have built foundational elements for their project, enabling integration of components and supporting the continued evolution of their models and applications. They will also be required to create a mock-up of their final application, either refining or extending previous submissions.

Key dates:

- Due date: 10/16

Template Repository

Objectives:

Virtual Environment Setup: Virtual machines and environments tailored to support containerized components must be fully implemented. This should include detailed documentation on the setup process.

Deliverables:

- A screenshot of the running instances in the cloud or local environment.

Containerized Components: All individual project components should be containerized using Docker, ensuring atomicity and isolation. Each container must perform a specific function (e.g., data scraping, preprocessing, data labeling) and be ready for integration into the project architecture.

Deliverables:

- Dockerfiles for each container and build instructions
- Pipfiles files for package management within each container
- Shell scripts or docker-compose.yml for orchestration, if multiple containers need to be run together
- Documentation explaining the purpose of each container and instructions for running them.

Versioned Data Strategy: Implement a data versioning strategy using tools like DVC or other suitable solutions. If feasible, this strategy should also be containerized to ensure portability and reproducibility of data processes.(Optional but recommended)

Please make sure to record your prompts as part of the data. For eg: If you are generating data using a LLM, please add the prompts and generated data as part of the dataset.

Deliverables:

- Documentation on the data versioning strategy chosen (e.g., DVC) and why
- A working containerized version of the data versioning pipeline (if applicable)

AC215

- Version control history showing tracked datasets, along with their respective versions, commits, and logs.

Teams Utilizing LLMs (Large Language Models): Teams working with LLMs should implement a RAG setup. This setup should include data collection, chunking into appropriate sizes for processing, and the integration of a vector database. Teams should also fine-tune models and document the experimentation process.

Deliverables:

- A containerized RAG pipeline, including scripts for data chunking, vectorization, and integration with a vector database
- Documentation of the fine-tuning process, including datasets used, hyperparameters, and models
- Experiment logs showing model performance across different fine-tuning and RAG configurations.

Teams Focusing on Computer Vision or Other Modalities: Teams working on computer vision or other modalities (e.g., audio, time series) should focus on creating a robust data pipeline, fine-tuning models for their respective task, and experimenting with different model architectures.

Deliverables:

- A containerized pipeline for data ingestion and preprocessing
- Model fine-tuning scripts with detailed documentation on hyperparameters, datasets, and model versions
- Experiment logs, including results of different models, architectures, or techniques used.

Mock-up of the Application: A working prototype or mock-up of the final application that integrates all project components. Teams that have already submitted this in Milestone 1 should refine or extend their prototype based on feedback or new progress.

Deliverables:

- An application mock-up or wireframe, including user interface elements and how the app will interact with back-end components