Framework

- **Airflow** for orchestration and automation of training workflows.
- **MLflow** for experiment tracking, model registry, and artifact storage.
- **FastAPI** for serving predictions and managing model versions.
- **Model Storage** for persisting trained models.
- **API Layer** for user interaction and inference.

Constraints

- **Resources**: Runs on local or cloud environment with minimal compute.
- **Data**: Uses Iris dataset (small, structured).

Assumptions

- MLflow tracking server is running and accessible at http://localhost:5000.
- Airflow DAGs are manually triggered (no automated scheduling required).
- FastAPI server runs locally or on a simple cloud VM (no container orchestration).
- Model input format matches training format (Iris features in correct order).
- No external authentication or authorization required for API endpoints.
- Single-user or small-scale usage; performance optimization is not critical.
- Use s3 to store data/models if needed

APIs

GET - /health
Shows current health

GET - /current-version
Shows current model version

POST - /set-version
Set model version

POST - /predict
Predicts based on input array

GET - /generate-and-predict
Generates a random dataset and predicts