MLOPS Assignment 1

Group 18

Title: Heart Disease Prediction

Submission: Documentation and Reporting (Part 9)

Course: MLOps Experimental Learning Assignment (Heart Disease UCI)

Name	ID	Contribution
Ashmita De	2024aa05248	100%
Ayush Goyal	2024aa05463	100%
Srinivasan V	2024aa05292	100%
Saurabh Vikas Kolhe	2024aa05350	100%

Project Summary

This report documents the full lifecycle of our heart disease risk prediction system, built using the UCI Heart Disease dataset and delivered as a production-ready API. We automated data preparation, feature engineering, and model training, tracked experiments with MLflow, packaged the final model for reproducible inference, containerized the API with Docker, deployed it on Kubernetes with auto-scaling, and added monitoring with Prometheus and Grafana. The chosen model (Random Forest) achieved a ROC-AUC around 0.918 on cross-validation.

1. Setup and installation

We prepared a predictable Python environment (3.9+) and installed all dependencies from a single requirements file. The project runs on Windows, macOS, or Linux. For local runs, we used a virtual environment and for production, we relied on an immutable Docker image. The same codebase supports both flows.

To set up from scratch:

- Clone the repository from the link provided at the end of this report.
- Create and activate a Python virtual environment.
- Install dependencies using the provided requirements.txt.
- Run the data acquisition and preprocessing scripts (Part 1) to produce cleaned and encoded datasets, along with EDA figures.
- Train models (Part 2 and Part 3) and optionally view MLflow runs locally.

- Package the best model (Part 4) to generate a joblib pipeline and an MLflow model directory.
- Execute the test suite (Part 5) to verify data processing, modeling, and inference logic.
- Start the API locally for quick checks or build the Docker image for containerized deployment (Part 6).
- Deploy to a Kubernetes cluster using provided manifests or Helm chart (Part 7).
- Enable monitoring with docker-compose or Helm (Part 8) and verify dashboards.

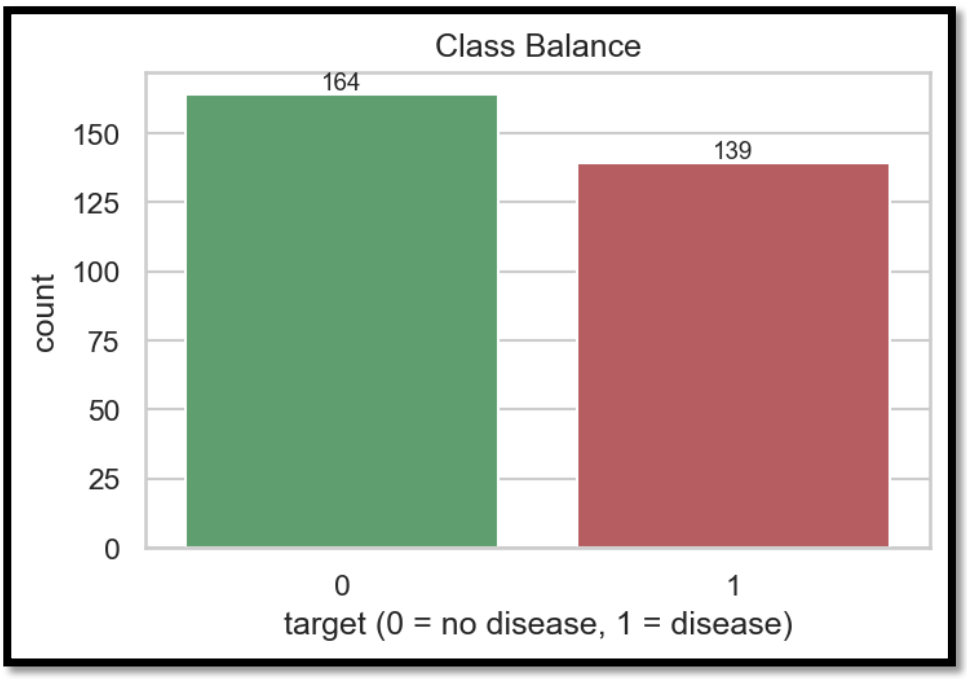
2. Data and EDA (exploratory analysis) methodology and decisions

We used the processed Cleveland subset from the UCI Heart Disease dataset (303 records and 14 attributes). We converted the original multi-class target (0–4) into a binary target: 0 for no disease and 1 for disease (num > 0). Data contained missing markers ("?") in a few columns, notably **ca** and **thal**. We replaced "?" with NaN and imputed missing values using median (numeric) and mode (categorical), then casted to appropriate dtypes.

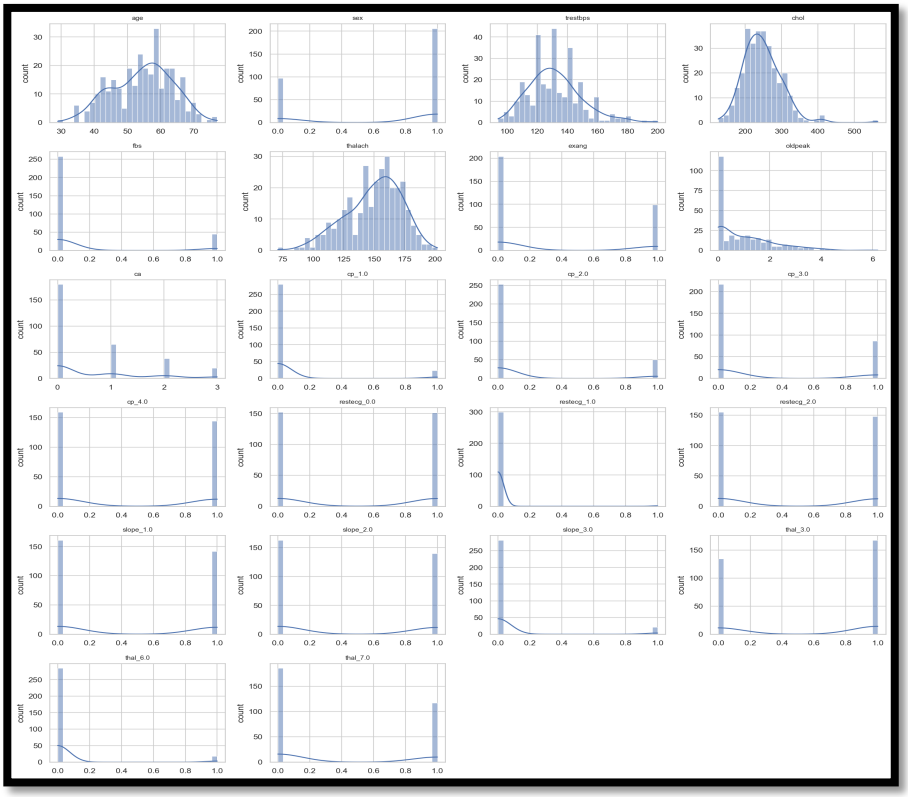
EDA confirmed that age, chest pain type, exercise-induced angina, ST depression (oldpeak), and the ca feature were influential for prediction. Numeric features showed typical ranges (e.g., age around mid-50s), with noticeable variance in cholesterol and maximum heart rate. The class distribution was relatively balanced (about 54% no disease vs 46% disease), so we did not need aggressive rebalancing.

We produced clear plots to summarize findings:

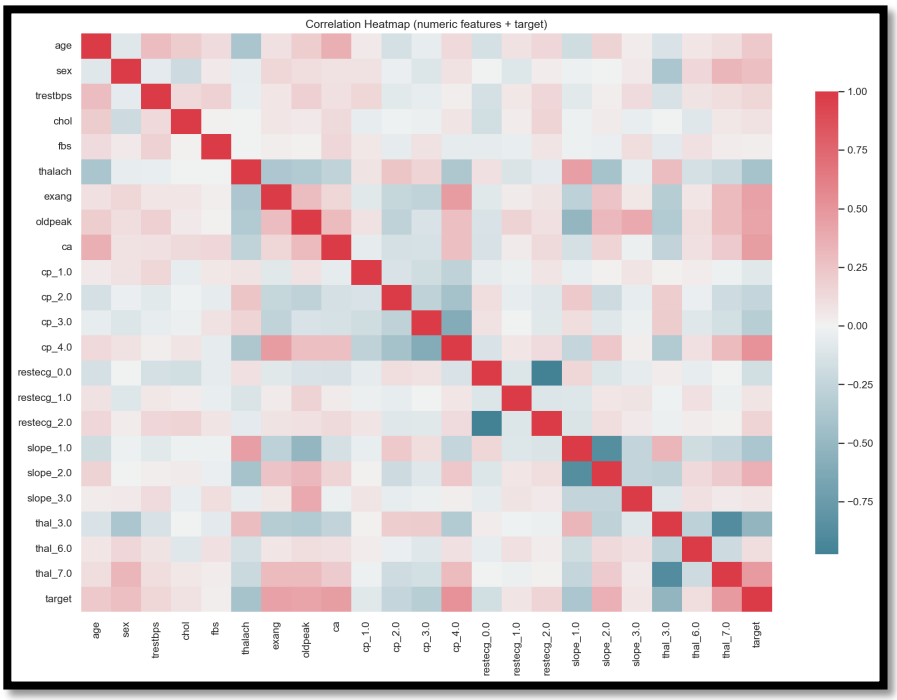
- A class balance chart



- Histograms for numeric features (distribution and outliers)



- A correlation heatmap showing relationships with the target



All generated figures are stored in Part1/reports/figures/ for reproducibility.

3. Feature engineering and model choices

We built a single, versioned preprocessing pipeline using scikit-learn's ColumnTransformer so that training and inference share identical transformations. The pipeline applied:

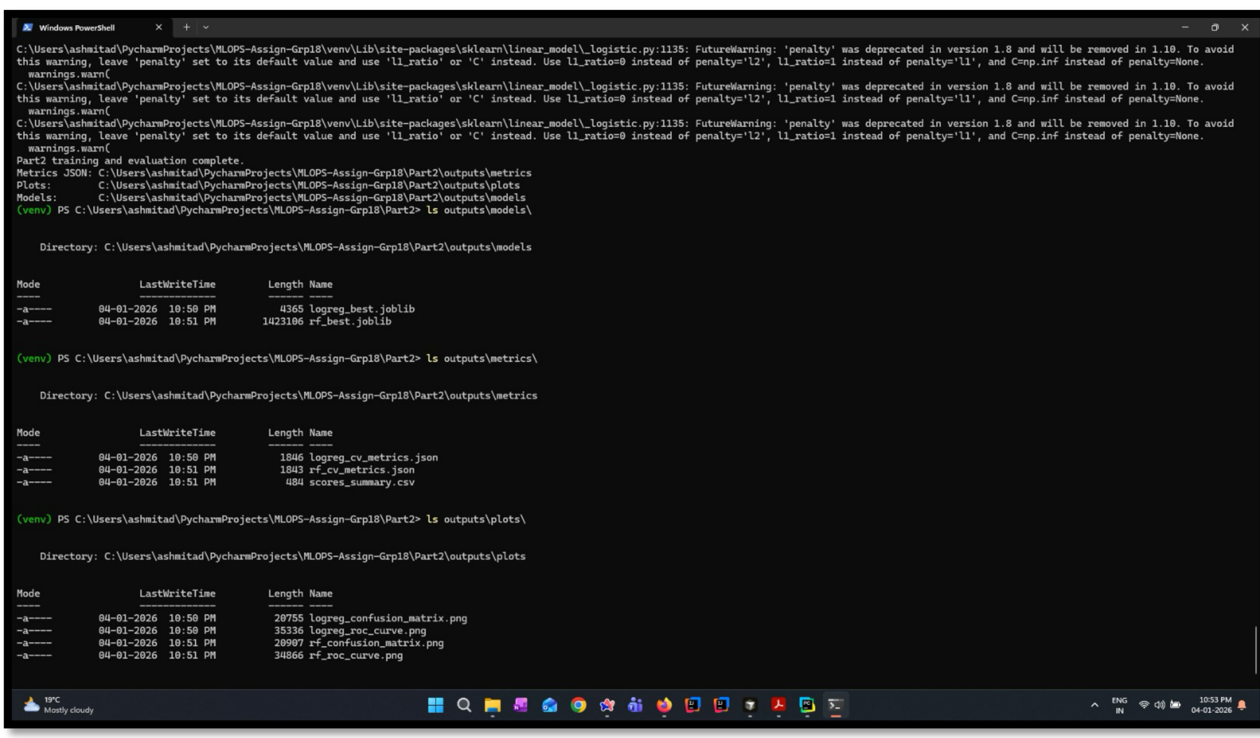
- Standardization for continuous features (age, trestbps, chol, thalach, oldpeak, ca)
- Passthrough for binary features (sex, fbs, exang)
- One-hot encoding for categorical features (cp, restecg, slope, thal)

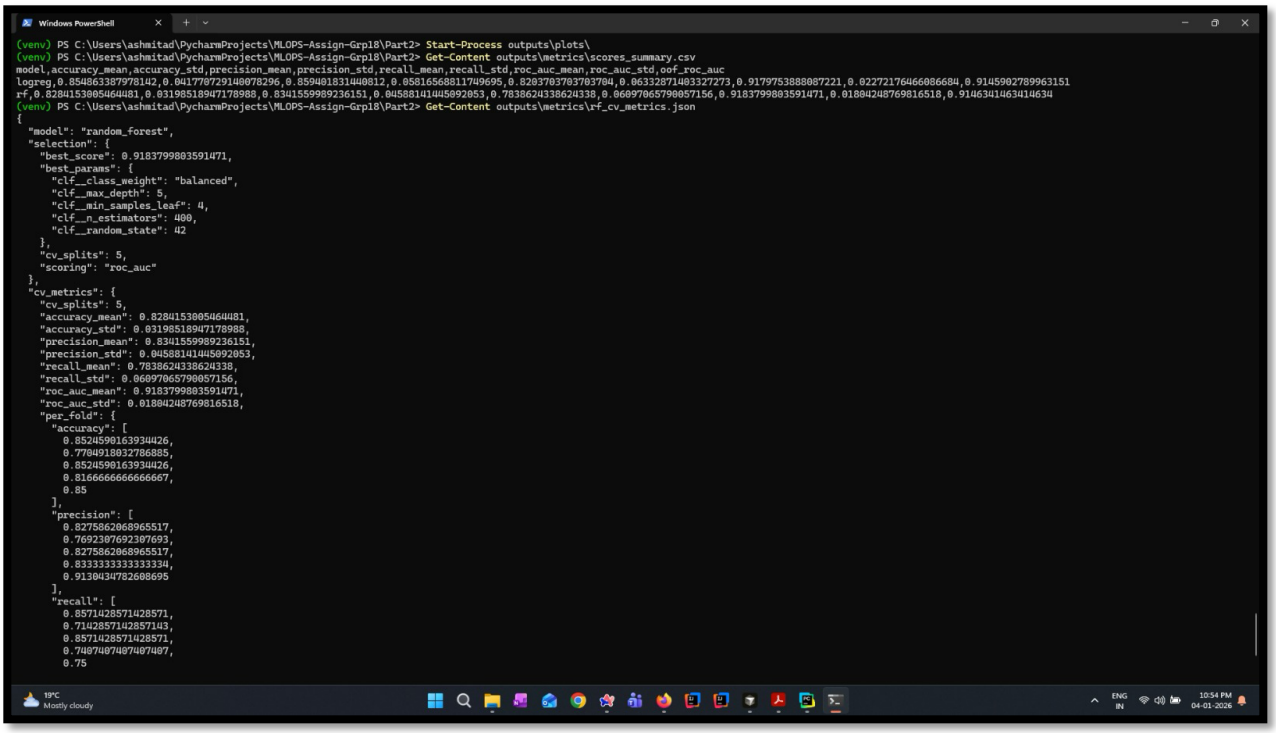
We compared two well-understood baselines:

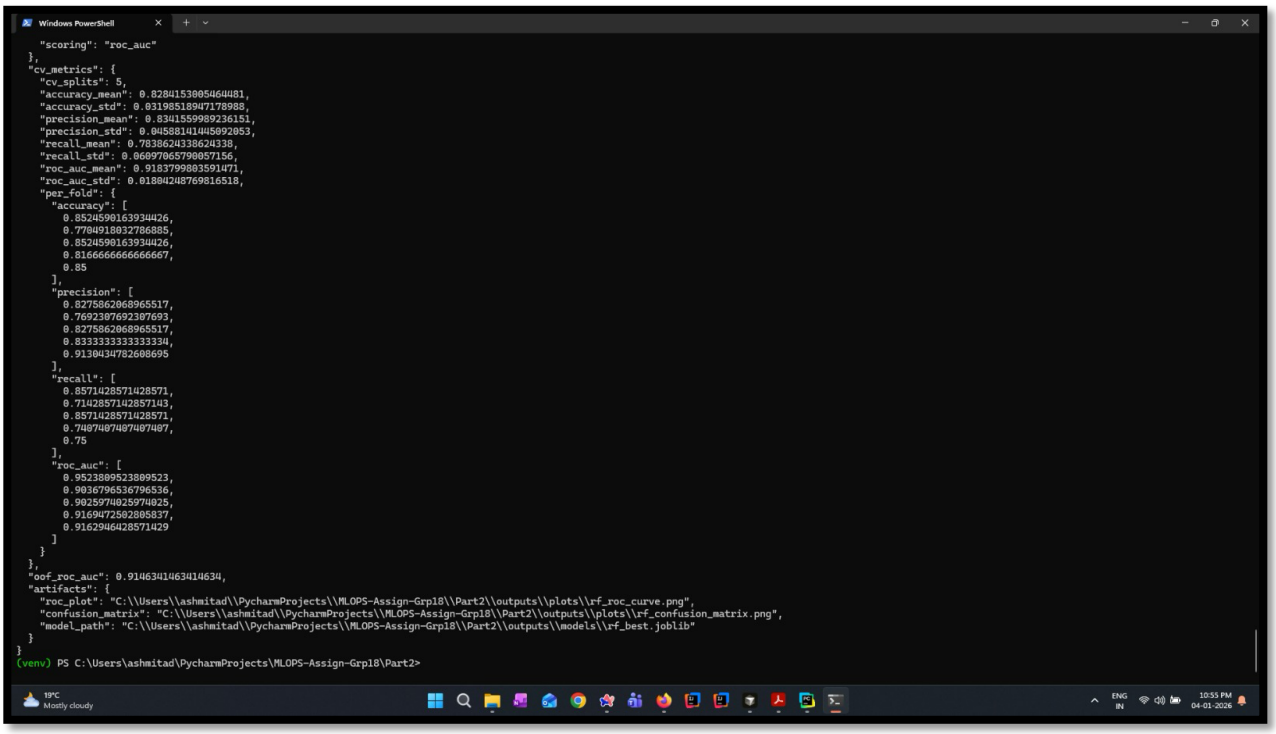
- Logistic Regression (interpretable, fast to train)
- Random Forest (non-linear, robust to interactions and outliers)

We tuned both with small, sensible grids and evaluated them using stratified 5-fold cross-validation. We optimized primarily for ROC-AUC since it is stable and informative for binary classification. Both models performed similarly (around 0.918 ROC-AUC), with the Random Forest having marginally higher mean ROC-AUC and acceptable variance. We therefore selected Random Forest as the final model, while retaining Logistic Regression as a strong baseline.

Post-Feature Engineering Model Evaluation Artifacts




Windows PowerShell
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part2> Start-Process outputs\plots\
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part2> Get-Content outputs\metrics\scores_summary.csv
model,accuracy_mean,accuracy_std,precision_mean,precision_std,recall_mean,recall_std,roc_auc_mean,roc_auc_std,oof_roc_auc
logreg,0.8548633879781142,0.04177072914007829,0.8594018314408124,0.05816568811749695,0.8203703703703704,0.06332871403327273,0.9179753888087221,0.02272176466086684,0.9145902789963151
rf,0.8284153005464481,0.03198518947178988,0.8341559989236151,0.04588141445092053,0.7838624338624338,0.06097065790057156,0.9183799803591471,0.01804248769816518,0.9146341463414634
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part2> Get-Content outputs\metrics\rf_cv_metrics.json
{
"model": "random_forest",
"selection": {
"best_score": 0.9183799803591471,
"best_params": {
"clf__class_weight": "balanced",
"clf__max_depth": 5,
"clf__min_samples_leaf": 4,
"clf__n_estimators": 400,
"clf__random_state": 42
},
"cv_splits": 5,
"scoring": "roc_auc"
},
"cv_metrics": {
"cv_splits": 5,
"accuracy_mean": 0.8284153005464481,
"accuracy_std": 0.03198518947178988,
"precision_mean": 0.8341559989236151,
"precision_std": 0.04588141445092053,
"recall_mean": 0.7838624338624338,
"recall_std": 0.06097065790057156,
"roc_auc_mean": 0.9183799803591471,
"roc_auc_std": 0.01804248769816518,
"per_fold": {
"accuracy": [
0.8524590163934426,
0.7704918032786885,
0.8524590163934426,
0.8166666666666667,
0.85
],
"precision": [
0.8275862068965517,
0.7692307692307693,
0.8275862068965517,
0.8333333333333334,
0.9130434782608695
],
"recall": [
0.8571428571428571,
0.7142857142857143,
0.8571428571428571,
0.7407407407407407,
0.75
19°C
Mostly cloudy
ENG
IN
10:54 PM
04-01-2026


Windows PowerShell
"scoring": "roc_auc"
},
"cv_metrics": {
"cv_splits": 5,
"accuracy_mean": 0.8284153005464481,
"accuracy_std": 0.03198518947178988,
"precision_mean": 0.8341559989236151,
"precision_std": 0.04588141445092053,
"recall_mean": 0.7838624338624338,
"recall_std": 0.06097065790057156,
"roc_auc_mean": 0.9183799803591471,
"roc_auc_std": 0.01804248769816518,
"per_fold": {
"accuracy": [
0.8524590163934426,
0.7704918032786885,
0.8524590163934426,
0.8166666666666667,
0.85
],
"precision": [
0.8275862068965517,
0.7692307692307693,
0.8275862068965517,
0.8333333333333334,
0.9130434782608695
],
"recall": [
0.8571428571428571,
0.7142857142857143,
0.8571428571428571,
0.7407407407407407,
0.75
],
"roc_auc": [
0.9523809523809523,
0.9036796536796536,
0.9025974025974025,
0.9169472502805837,
0.9162946428571429
]
}
},
"oof_roc_auc": 0.9146341463414634,
"artifacts": {
"roc_plot": "C:\\Users\\ashmitad\\PycharmProjects\\MLOPS-Assign-Grp18\\Part2\\outputs\\plots\\rf_roc_curve.png",
"confusion_matrix": "C:\\Users\\ashmitad\\PycharmProjects\\MLOPS-Assign-Grp18\\Part2\\outputs\\plots\\rf_confusion_matrix.png",
"model_path": "C:\\Users\\ashmitad\\PycharmProjects\\MLOPS-Assign-Grp18\\Part2\\outputs\\models\\rf_best.joblib"
}
}
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part2>
19°C
Mostly cloudy
ENG
IN
10:55 PM
04-01-2026

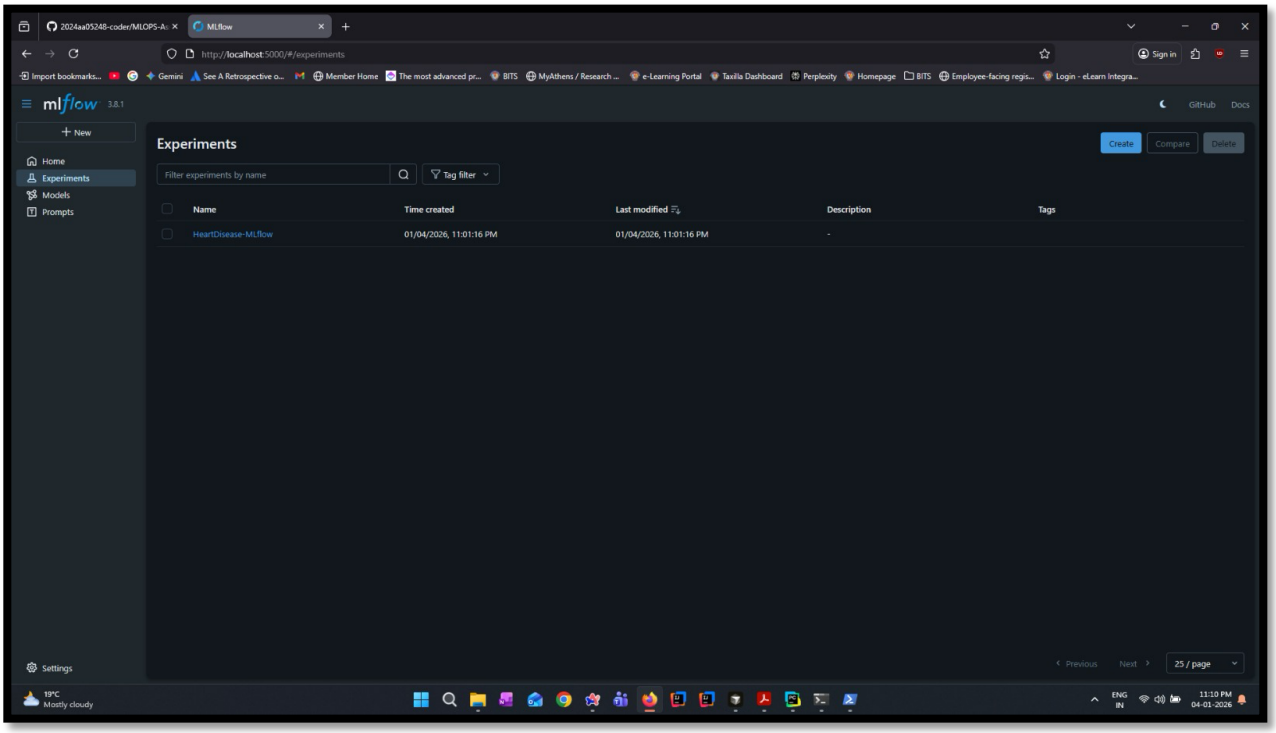
4. Experiment tracking summary

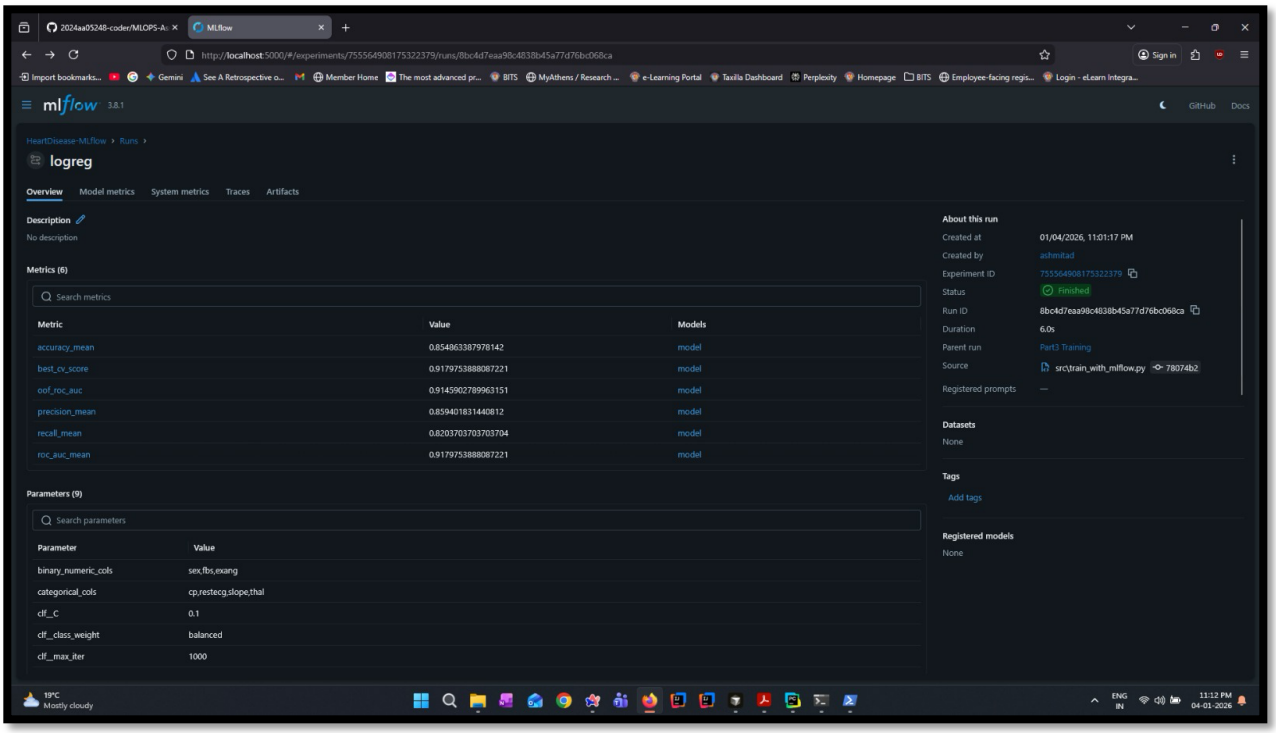
We integrated MLflow to capture all important aspects of our experiments:

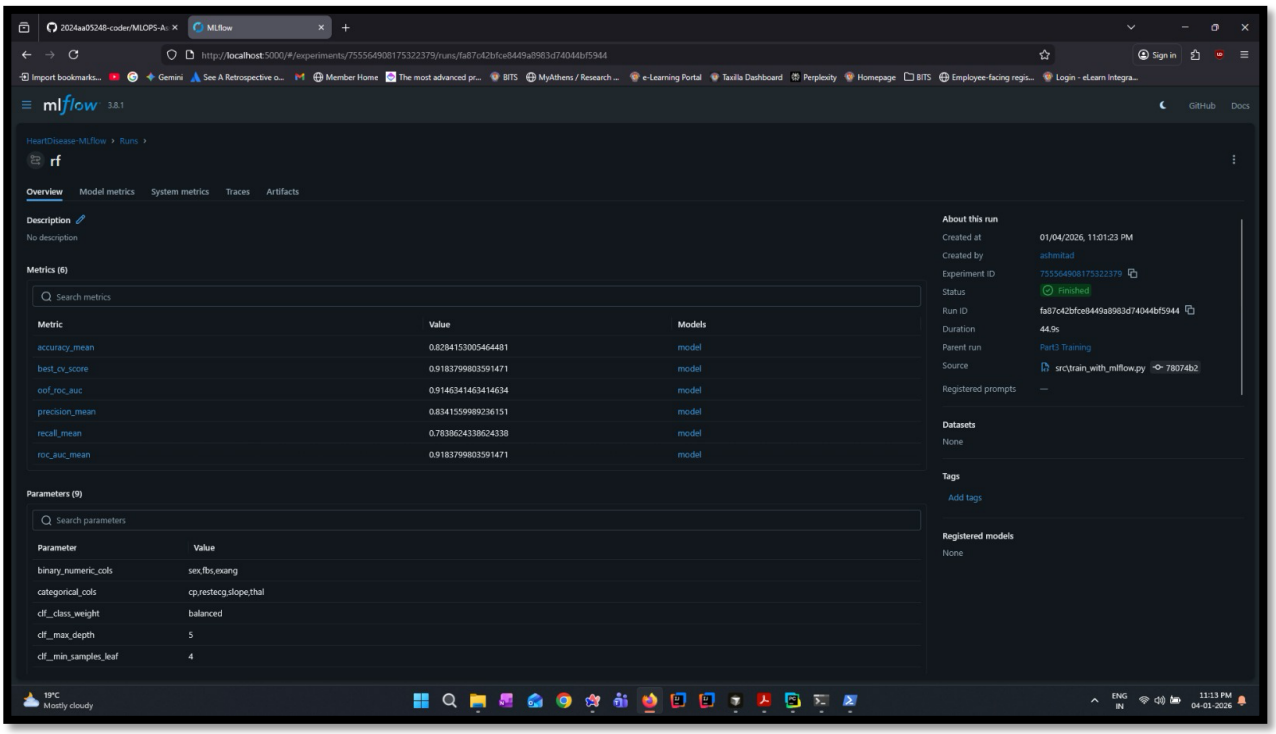
- Parameters: hyperparameters, preprocessing configuration, and CV settings
- Metrics: accuracy, precision, recall, ROC-AUC (mean and standard deviation)
- Artifacts: ROC curves, confusion matrices, and run reports
- Models: serialized pipelines (joblib) with input-output signatures

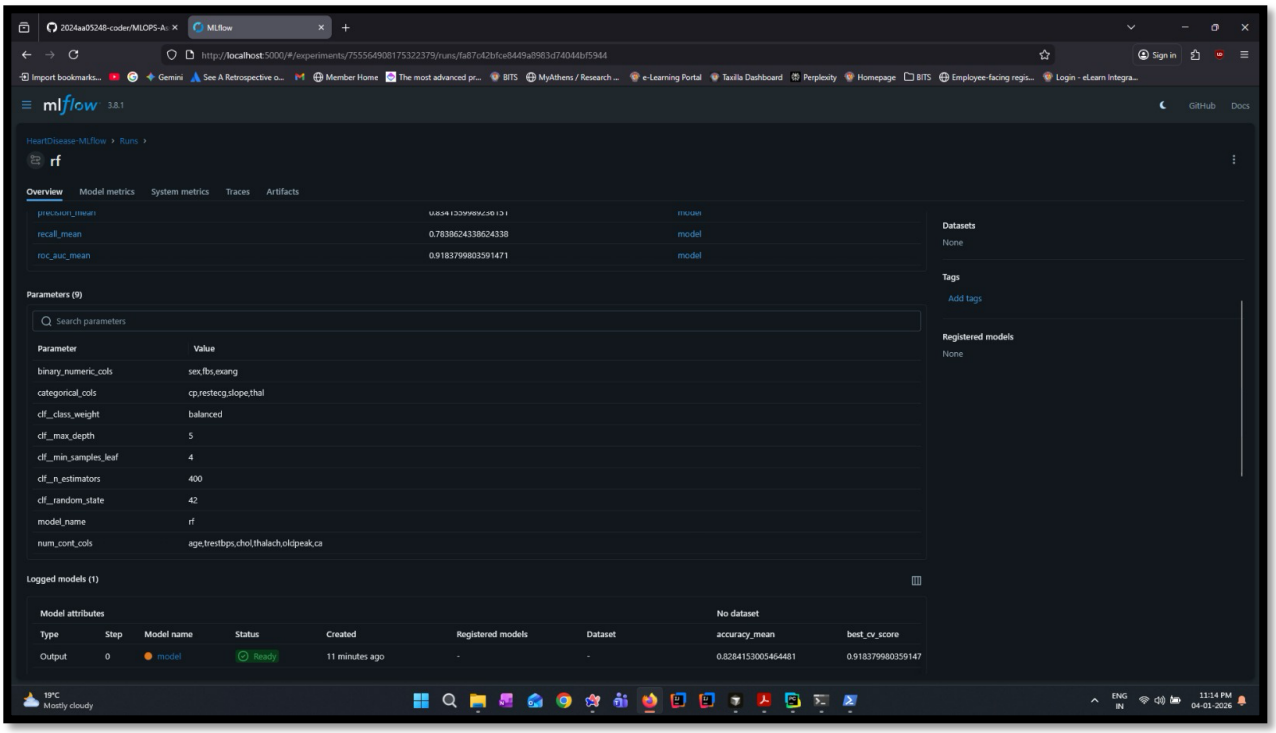
We kept a local file-based MLflow backend at Part3/mlruns/ so that experiments are reproducible and inspectable offline. We used logical parent and child runs to organize comparisons (e.g., one parent run grouping both model runs). The best run and its artifacts are clearly visible in the MLflow UI, which can be started locally for review. This tracking allowed us to make a defensible model selection and to preserve a complete audit trail of experiments.

MLflow UI




MLflow
http://localhost:5000/#/experiments/755564908175322379/runs/8bc4d7eaa98c4838b45a77d76bc068ca
mlflow 3.8.1
HeartDisease-MLflow › Runs ›
logreg
Overview
Model metrics
System metrics
Traces
Artifacts
Description
No description
Metrics (6)
Search metrics
Metric
Value
Models
accuracy_mean 0.85486338797814 2 model
best_cv_score 0.9179753888087221 model
oof_roc_auc 0.9145902789963151 model
precision_mean 0.859401831440812 model
recall_mean 0.8203703703703704 model
roc_auc_mean 0.9179753888087221 model
Parameters (9)
Search parameters
Parameter
Value
binary_numeric_cols sex,fbs,exang
categorical_cols cp,restecg,slope,thal
clf__C 0.1
clf__class_weight balanced
clf__max_iter 1000
About this run
Created at 01/04/2026, 11:01:17 PM
Created by ashmitad
Experiment ID 755564908175322379
Status Finished
Run ID 8bc4d7eaa98c4838b45a77d76bc068ca
Duration 6.0s
Parent run Part3 Training
Source src\train_with_mlflow.py 78074b2
Registered prompts —
Datasets
None
Tags
Add tags
Registered models
None


MLflow
http://localhost:5000/#/experiments/755564908175322379/runs/fa87c42bfce8449a8983d74044bf5944
mlflow 3.8.1
HeartDisease-MLflow › Runs ›
rf
Overview
Model metrics
System metrics
Traces
Artifacts
Description
No description
Metrics (6)
Search metrics
Metric
Value
Models
accuracy_mean 0.8284153005464481 model
best_cv_score 0.9183799803591471 model
oof_roc_auc 0.9146341463414634 model
precision_mean 0.8341559989236151 model
recall_mean 0.7838624338624338 model
roc_auc_mean 0.9183799803591471 model
Parameters (9)
Search parameters
Parameter
Value
binary_numeric_cols sex,fbs,exang
categorical_cols cp,restecg,slope,thal
clf__class_weight balanced
clf__max_depth 5
clf__min_samples_leaf 4
About this run
Created at 01/04/2026, 11:01:23 PM
Created by ashmitad
Experiment ID 755564908175322379
Status Finished
Run ID fa87c42bfce8449a8983d74044bf5944
Duration 44.9s
Parent run Part3 Training
Source src\train_with_mlflow.py 78074b2
Registered prompts —
Datasets
None
Tags
Add tags
Registered models
None



5. Model Reproducibility and packaging

We emphasized deterministic execution across environments. The final inference artifact is a joblib-serialized pipeline that includes all preprocessing and the estimator, ensuring that inputs are transformed identically at prediction time. We also exported an MLflow model directory to support pyfunc serving and portability.

We pinned versions in requirements.txt and fixed random states to reduce variance between runs. The packaging script writes auxiliary metadata (schema information, feature names after transformation, and final selection details) so that downstream services can validate inputs and understand the pipeline.

Reproducible training and selection (Part4/src/package_model.py)

- Fixed randomness: all stochastic components use fixed seeds (e.g., StratifiedKFold(shuffle=True, random_state=42), model random_state where applicable).
- Consistent preprocessing: a single scikit-learn Pipeline encapsulates both training and inference transformations via a ColumnTransformer:
    - StandardScaler on numeric: age, trestbps, chol, thalach, oldpeak, ca
    - Passthrough on binary flags: sex, fbs, exang
    - OneHotEncoder on categoricals: cp, restecg, slope, thal
- Model comparison and selection:
    - Evaluated Logistic Regression and Random Forest with stratified 5-fold CV.

    - Logged accuracy, precision, recall, ROC-AUC (mean and std), and Out-Of-Fold (OOF) ROC-AUC.
    - Selected the final model by highest mean ROC-AUC; ties broken by OOF ROC-AUC.
    - Refit the winning pipeline on all data prior to packaging.
- Experiment traceability: runs are tracked in MLflow (Part3/mlruns/), preserving parameters, metrics, and artifacts for auditability.

```
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part3> cd ..\Part4
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> python src\package_model.py
[INFO] Part4 packaging complete (reused Part2 trained model).
[INFO] Final joblib pipeline: C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\models\final_model.joblib
[INFO] MLflow model dir:      C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\models\mlflow_model
[INFO] Schema JSON:           C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\models\schema.json
[INFO] Final report JSON:     C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\metrics\final_report.json
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> ls models\


    Directory: C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\models


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        04-01-2026  11:16 PM                mlflow_model
-a----        04-01-2026  11:16 PM        1423298 final_model.joblib
-a----        04-01-2026  11:16 PM            939 schema.json


(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> ls models\mlflow_model\


    Directory: C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\models\mlflow_model


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        04-01-2026  11:16 PM            199 conda.yaml
-a----        04-01-2026  11:16 PM            505 input_example.json
-a----        04-01-2026  11:16 PM           1982 MLmodel
-a----        04-01-2026  11:16 PM        1407836 model.pkl
-a----        04-01-2026  11:16 PM            111 python_env.yaml
-a----        04-01-2026  11:16 PM             76 requirements.txt
-a----        04-01-2026  11:16 PM           1317 serving_input_example.json
```

```
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> python src\infer.py --input inputs\sample_X.csv
Wrote predictions to: C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\outputs\predictions.csv
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> Get-Content outputs\predictions.csv
age,sex,trestbps,chol,fbs,thalach,exang,oldpeak,ca,cp,restecg,slope,thal,prediction,proba
63.0,1.0,145.0,233.0,1.0,150.0,0.0,2.3,0.0,1.0,2.0,3.0,6.0,0,0.39344200497508375
67.0,1.0,160.0,286.0,0.0,108.0,1.0,1.5,3.0,4.0,2.0,2.0,3.0,1,0.8823699127625356
67.0,1.0,120.0,229.0,0.0,129.0,1.0,2.6,2.0,4.0,2.0,2.0,7.0,1,0.9604531757709505
37.0,1.0,130.0,250.0,0.0,187.0,0.0,3.5,0.0,3.0,0.0,3.0,3.0,0,0.217968869139218 95
41.0,0.0,130.0,204.0,0.0,172.0,0.0,1.4,0.0,2.0,2.0,1.0,3.0,0,0.05410236282790814
56.0,1.0,120.0,236.0,0.0,178.0,0.0,0.8,0.0,2.0,0.0,1.0,3.0,0,0.08034860029760844
62.0,0.0,140.0,268.0,0.0,160.0,0.0,3.6,2.0,4.0,2.0,3.0,3.0,1,0.6902673820590175
57.0,0.0,120.0,354.0,0.0,163.0,1.0,0.6,0.0,4.0,0.0,1.0,3.0,0,0.28834577721254817
63.0,1.0,130.0,254.0,0.0,147.0,0.0,1.4,1.0,4.0,2.0,2.0,7.0,1,0.9237971624432568
53.0,1.0,140.0,203.0,1.0,155.0,1.0,3.1,0.0,4.0,2.0,3.0,7.0,1,0.820101933177813
57.0,1.0,140.0,192.0,0.0,148.0,0.0,0.4,0.0,4.0,0.0,2.0,6.0,0,0.408937434487777253
56.0,0.0,140.0,294.0,0.0,153.0,0.0,1.3,0.0,2.0,2.0,2.0,3.0,0,0.21744575562243892
56.0,1.0,130.0,256.0,1.0,142.0,1.0,0.6,1.0,3.0,2.0,2.0,6.0,1,0.683530887232058
44.0,1.0,120.0,263.0,0.0,173.0,0.0,0.0,0.0,2.0,0.0,1.0,7.0,0,0.2633409753269987
52.0,1.0,172.0,199.0,1.0,162.0,0.0,0.5,0.0,3.0,0.0,1.0,7.0,0,0.241183915934875 06
57.0,1.0,150.0,168.0,0.0,174.0,0.0,1.6,0.0,3.0,0.0,1.0,3.0,0,0.12289655134352838
48.0,1.0,110.0,229.0,0.0,168.0,0.0,1.0,0.0,2.0,0.0,3.0,7.0,0,0.4405748055477054 5
54.0,1.0,140.0,239.0,0.0,160.0,0.0,1.2,0.0,4.0,0.0,1.0,3.0,0,0.23306143623794892
48.0,0.0,130.0,275.0,0.0,139.0,0.0,0.2,0.0,3.0,0.0,1.0,3.0,0,0.10287654520352436
49.0,1.0,130.0,266.0,0.0,171.0,0.0,0.6,0.0,2.0,0.0,1.0,3.0,0,0.05837263712983532
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> python src\infer.py --input test_patient.csv --output my_predictions.csv
Wrote predictions to: C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4\my_predictions.csv
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> ls .\my_predictions.csv


    Directory: C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        04-01-2026  11:25 PM            149 my_predictions.csv
```

6. Automated testing

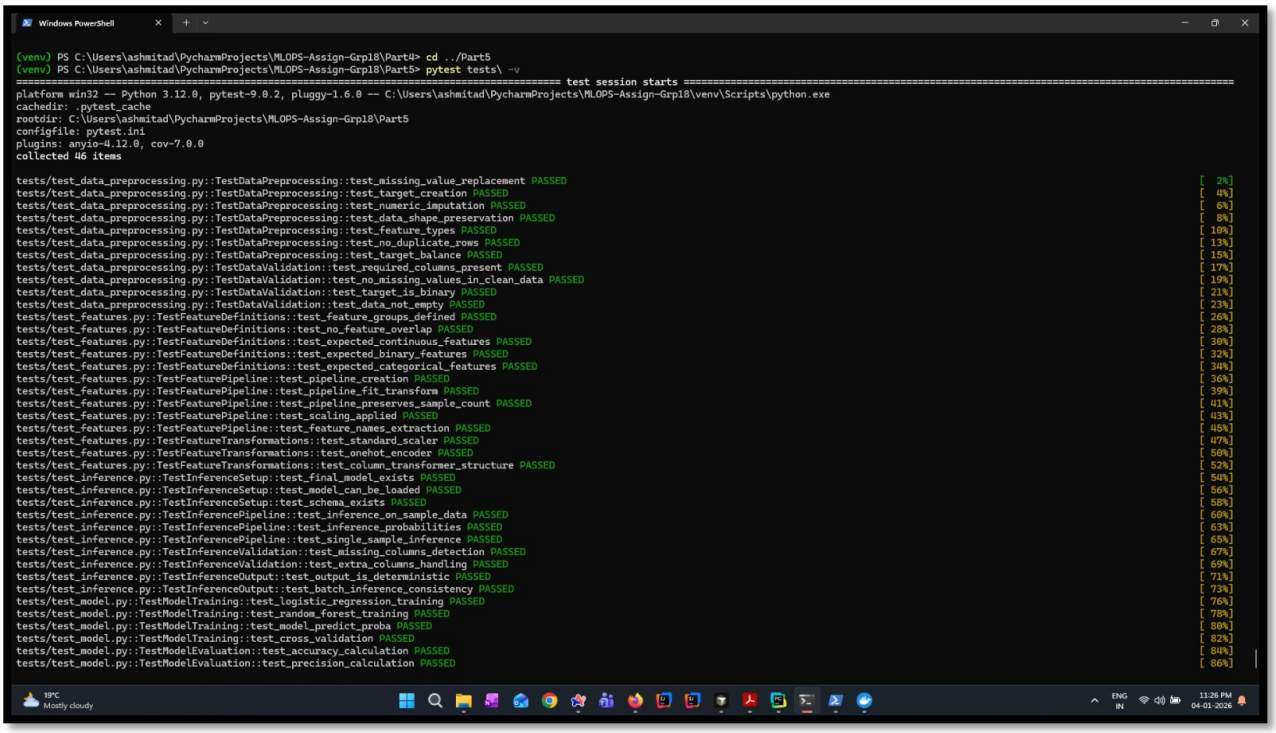
We implemented a comprehensive pytest suite that validates the entire ML pipeline which includes data cleaning, feature engineering (ColumnTransformer), model training, and inference. Tests enforce deterministic behavior (fixed random_state), schema/shape correctness, and probability bounds, with coverage reporting. The suite runs in CI on every push and gates the pipeline on failure.

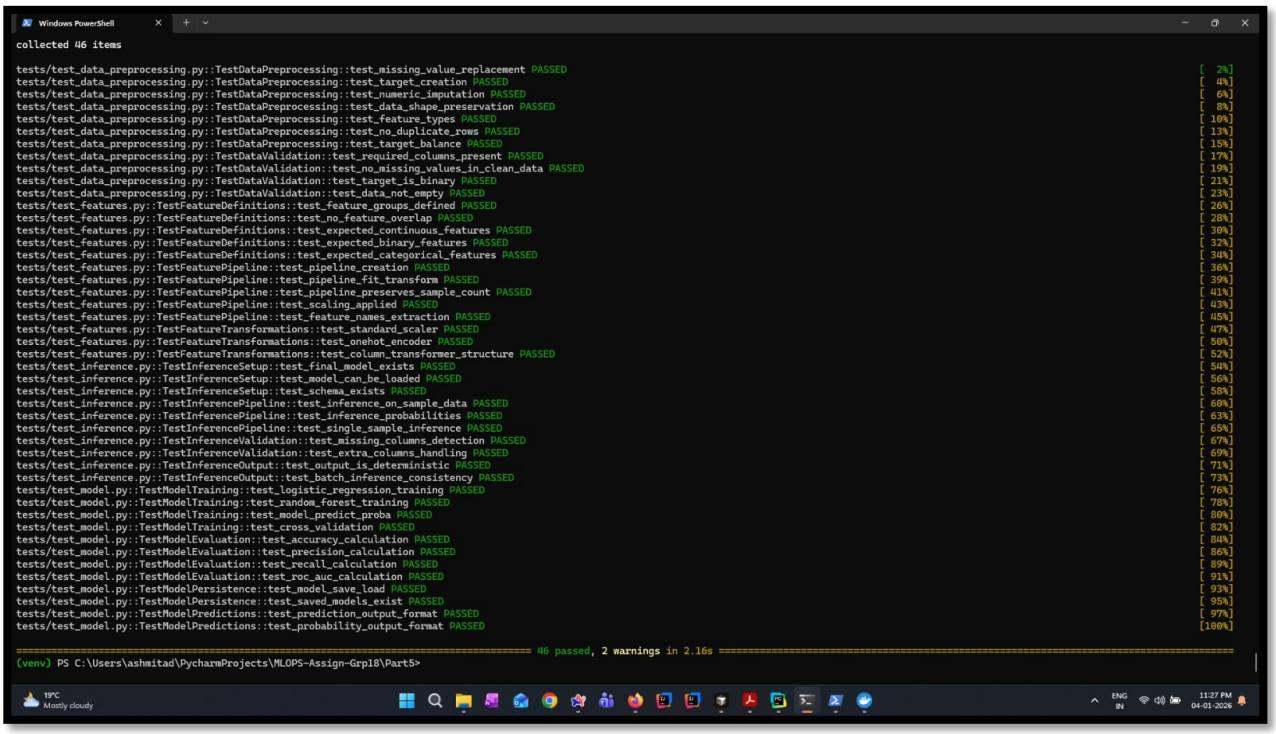
6.1 Scope covered by tests:

- Data preprocessing: handling "?" → NaN, imputation (median/mode), dtype casting, column/shape integrity.
- Feature engineering: ColumnTransformer behavior, one-hot categories present, standardized scaling applied, transformed feature names/order and dimensionality.
- Model development: Logistic Regression and Random Forest train deterministically (fixed random_state), predict/predict_proba shapes correct, basic sanity thresholds on CV metrics (accuracy/precision/recall/ROC-AUC).
- Inference: packaged pipeline loads successfully; single and batch predictions; schema/typing validation; probabilities within [0,1]; graceful errors for invalid inputs.

6.2 CI integration:

- Tests run in the CI workflow on each push/PR and failures block downstream stages.
- JUnit XML and (if enabled) coverage reports are uploaded as artifacts for each run.


Windows PowerShell
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part4> cd ../Part5
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part5> pytest tests\ -v
test session starts
platform win32 -- Python 3.12.0, pytest-9.0.2, pluggy-1.6.0 -- C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\venv\Scripts\python.exe
cachedir: .pytest_cache
rootdir: C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part5
configfile: pytest.ini
plugins: anyio-4.12.0, cov-7.0.0
collected 46 items
tests/test_data_preprocessing.py::TestDataPreprocessing::test_missing_value_replacement PASSED [2%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_target_creation PASSED [4%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_numeric_imputation PASSED [6%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_data_shape_preservation PASSED [8%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_feature_types PASSED [10%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_no_duplicate_rows PASSED [13%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_target_balance PASSED [15%]
tests/test_data_preprocessing.py::TestDataValidation::test_required_columns_present PASSED [17%]
tests/test_data_preprocessing.py::TestDataValidation::test_no_missing_values_in_clean_data PASSED [19%]
tests/test_data_preprocessing.py::TestDataValidation::test_target_is_binary PASSED [21%]
tests/test_data_preprocessing.py::TestDataValidation::test_data_not_empty PASSED [23%]
tests/test_features.py::TestFeatureDefinitions::test_feature_groups_defined PASSED [26%]
tests/test_features.py::TestFeatureDefinitions::test_no_feature_overlap PASSED [28%]
tests/test_features.py::TestFeatureDefinitions::test_expected_continuous_features PASSED [30%]
tests/test_features.py::TestFeatureDefinitions::test_expected_binary_features PASSED [32%]
tests/test_features.py::TestFeatureDefinitions::test_expected_categorical_features PASSED [34%]
tests/test_features.py::TestFeaturePipeline::test_pipeline_creation PASSED [36%]
tests/test_features.py::TestFeaturePipeline::test_pipeline_fit_transform PASSED [39%]
tests/test_features.py::TestFeaturePipeline::test_pipeline_preserves_sample_count PASSED [41%]
tests/test_features.py::TestFeaturePipeline::test_scaling_applied PASSED [43%]
tests/test_features.py::TestFeaturePipeline::test_feature_names_extraction PASSED [45%]
tests/test_features.py::TestFeatureTransformations::test_standard_scaler PASSED [47%]
tests/test_features.py::TestFeatureTransformations::test_onehot_encoder PASSED [50%]
tests/test_features.py::TestFeatureTransformations::test_column_transformer_structure PASSED [52%]
tests/test_inference.py::TestInferenceSetup::test_final_model_exists PASSED [54%]
tests/test_inference.py::TestInferenceSetup::test_model_can_be_loaded PASSED [56%]
tests/test_inference.py::TestInferenceSetup::test_schema_exists PASSED [58%]
tests/test_inference.py::TestInferencePipeline::test_inference_on_sample_data PASSED [60%]
tests/test_inference.py::TestInferencePipeline::test_inference_probabilities PASSED [63%]
tests/test_inference.py::TestInferencePipeline::test_single_sample_inference PASSED [65%]
tests/test_inference.py::TestInferenceValidation::test_missing_columns_detection PASSED [67%]
tests/test_inference.py::TestInferenceValidation::test_extra_columns_handling PASSED [69%]
tests/test_inference.py::TestInferenceOutput::test_output_is_deterministic PASSED [71%]
tests/test_inference.py::TestInferenceOutput::test_batch_inference_consistency PASSED [73%]
tests/test_model.py::TestModelTraining::test_logistic_regression_training PASSED [76%]
tests/test_model.py::TestModelTraining::test_random_forest_training PASSED [78%]
tests/test_model.py::TestModelTraining::test_model_predict_proba PASSED [80%]
tests/test_model.py::TestModelTraining::test_cross_validation PASSED [82%]
tests/test_model.py::TestModelEvaluation::test_accuracy_calculation PASSED [84%]
tests/test_model.py::TestModelEvaluation::test_precision_calculation PASSED [86%]
19°C
Mostly cloudy
ENG
IN
11:26 PM
04-01-2026

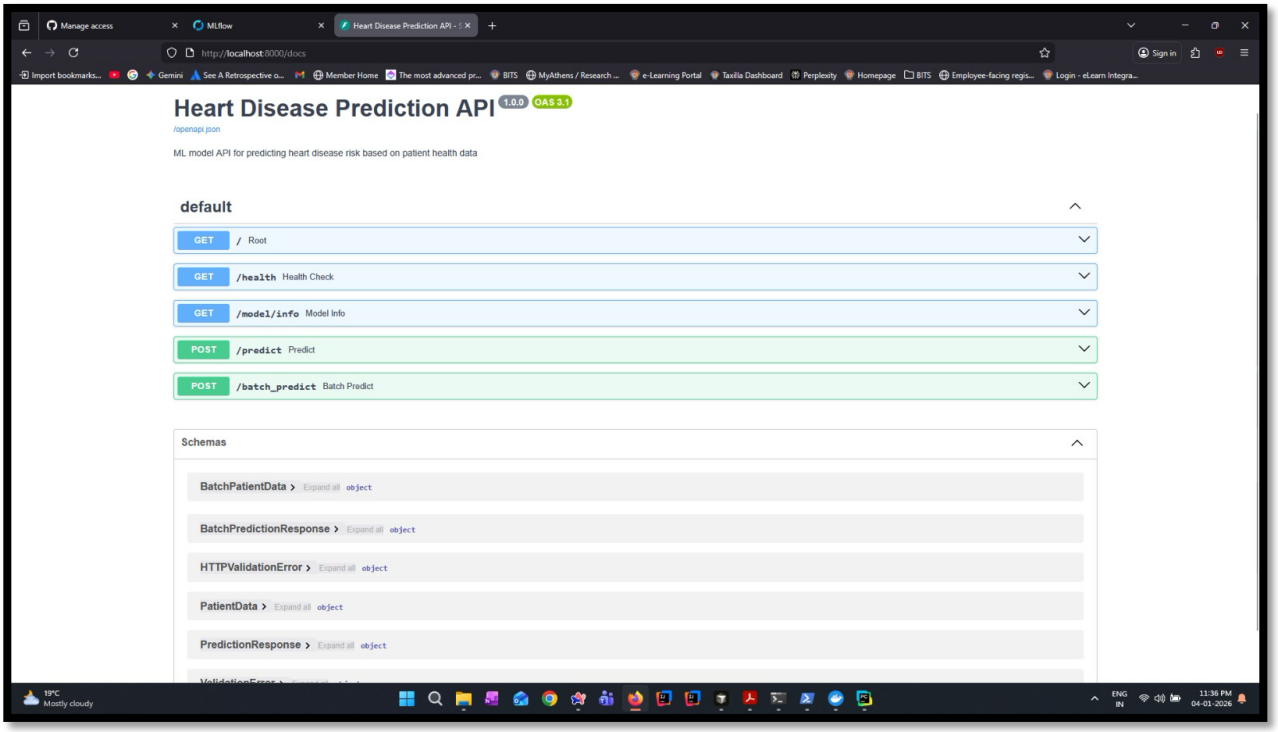

Windows PowerShell
collected 46 items
tests/test_data_preprocessing.py::TestDataPreprocessing::test_missing_value_replacement PASSED [2%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_target_creation PASSED [4%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_numeric_imputation PASSED [6%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_data_shape_preservation PASSED [8%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_feature_types PASSED [10%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_no_duplicate_rows PASSED [13%]
tests/test_data_preprocessing.py::TestDataPreprocessing::test_target_balance PASSED [15%]
tests/test_data_preprocessing.py::TestDataValidation::test_required_columns_present PASSED [17%]
tests/test_data_preprocessing.py::TestDataValidation::test_no_missing_values_in_clean_data PASSED [19%]
tests/test_data_preprocessing.py::TestDataValidation::test_target_is_binary PASSED [21%]
tests/test_data_preprocessing.py::TestDataValidation::test_data_not_empty PASSED [23%]
tests/test_features.py::TestFeatureDefinitions::test_feature_groups_defined PASSED [26%]
tests/test_features.py::TestFeatureDefinitions::test_no_feature_overlap PASSED [28%]
tests/test_features.py::TestFeatureDefinitions::test_expected_continuous_features PASSED [30%]
tests/test_features.py::TestFeatureDefinitions::test_expected_binary_features PASSED [32%]
tests/test_features.py::TestFeatureDefinitions::test_expected_categorical_features PASSED [34%]
tests/test_features.py::TestFeaturePipeline::test_pipeline_creation PASSED [36%]
tests/test_features.py::TestFeaturePipeline::test_pipeline_fit_transform PASSED [39%]
tests/test_features.py::TestFeaturePipeline::test_pipeline_preserves_sample_count PASSED [41%]
tests/test_features.py::TestFeaturePipeline::test_scaling_applied PASSED [43%]
tests/test_features.py::TestFeaturePipeline::test_feature_names_extraction PASSED [45%]
tests/test_features.py::TestFeatureTransformations::test_standard_scaler PASSED [47%]
tests/test_features.py::TestFeatureTransformations::test_onehot_encoder PASSED [50%]
tests/test_features.py::TestFeatureTransformations::test_column_transformer_structure PASSED [52%]
tests/test_inference.py::TestInferenceSetup::test_final_model_exists PASSED [54%]
tests/test_inference.py::TestInferenceSetup::test_model_can_be_loaded PASSED [56%]
tests/test_inference.py::TestInferenceSetup::test_schema_exists PASSED [58%]
tests/test_inference.py::TestInferencePipeline::test_inference_on_sample_data PASSED [60%]
tests/test_inference.py::TestInferencePipeline::test_inference_probabilities PASSED [63%]
tests/test_inference.py::TestInferencePipeline::test_single_sample_inference PASSED [65%]
tests/test_inference.py::TestInferenceValidation::test_missing_columns_detection PASSED [67%]
tests/test_inference.py::TestInferenceValidation::test_extra_columns_handling PASSED [69%]
tests/test_inference.py::TestInferenceOutput::test_output_is_deterministic PASSED [71%]
tests/test_inference.py::TestInferenceOutput::test_batch_inference_consistency PASSED [73%]
tests/test_model.py::TestModelTraining::test_logistic_regression_training PASSED [76%]
tests/test_model.py::TestModelTraining::test_random_forest_training PASSED [78%]
tests/test_model.py::TestModelTraining::test_model_predict_proba PASSED [80%]
tests/test_model.py::TestModelTraining::test_cross_validation PASSED [82%]
tests/test_model.py::TestModelEvaluation::test_accuracy_calculation PASSED [84%]
tests/test_model.py::TestModelEvaluation::test_precision_calculation PASSED [86%]
tests/test_model.py::TestModelEvaluation::test_recall_calculation PASSED [89%]
tests/test_model.py::TestModelEvaluation::test_roc_auc_calculation PASSED [91%]
tests/test_model.py::TestModelPersistence::test_model_save_load PASSED [93%]
tests/test_model.py::TestModelPersistence::test_saved_models_exist PASSED [95%]
tests/test_model.py::TestModelPredictions::test_prediction_output_format PASSED [97%]
tests/test_model.py::TestModelPredictions::test_probability_output_format PASSED [100%]
46 passed, 2 warnings in 2.16s
(venv) PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18\Part5>
19°C
Mostly cloudy
ENG
IN
11:27 PM
04-01-2026

7. API design, containerization and validation

The serving layer uses FastAPI/Uvicorn. A single scikit-learn pipeline is loaded at startup and exposed via typed, Pydantic-validated endpoints (/, /health, /model_info, /predict, /batch_predict). The service ships as a multi-stage Docker image with a healthcheck and runs locally, via Docker Compose, or on Kubernetes.

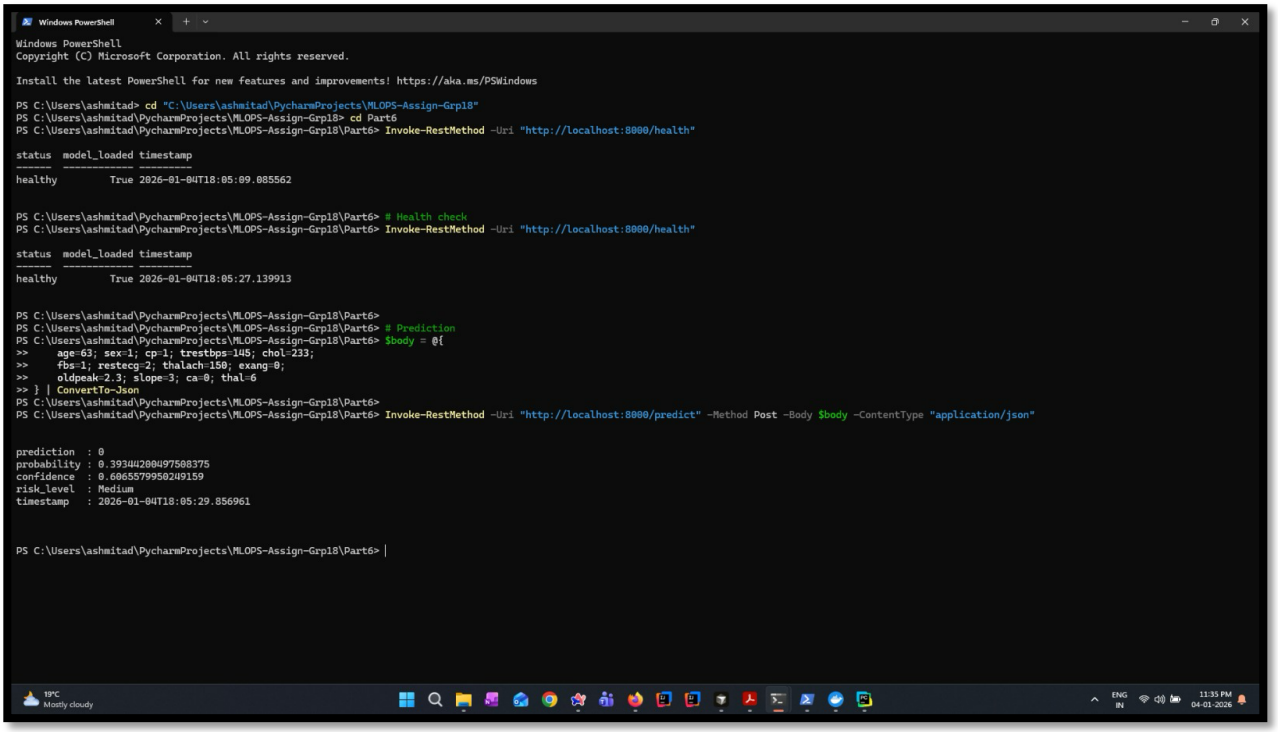
7.1 API Design & Endpoint Specification

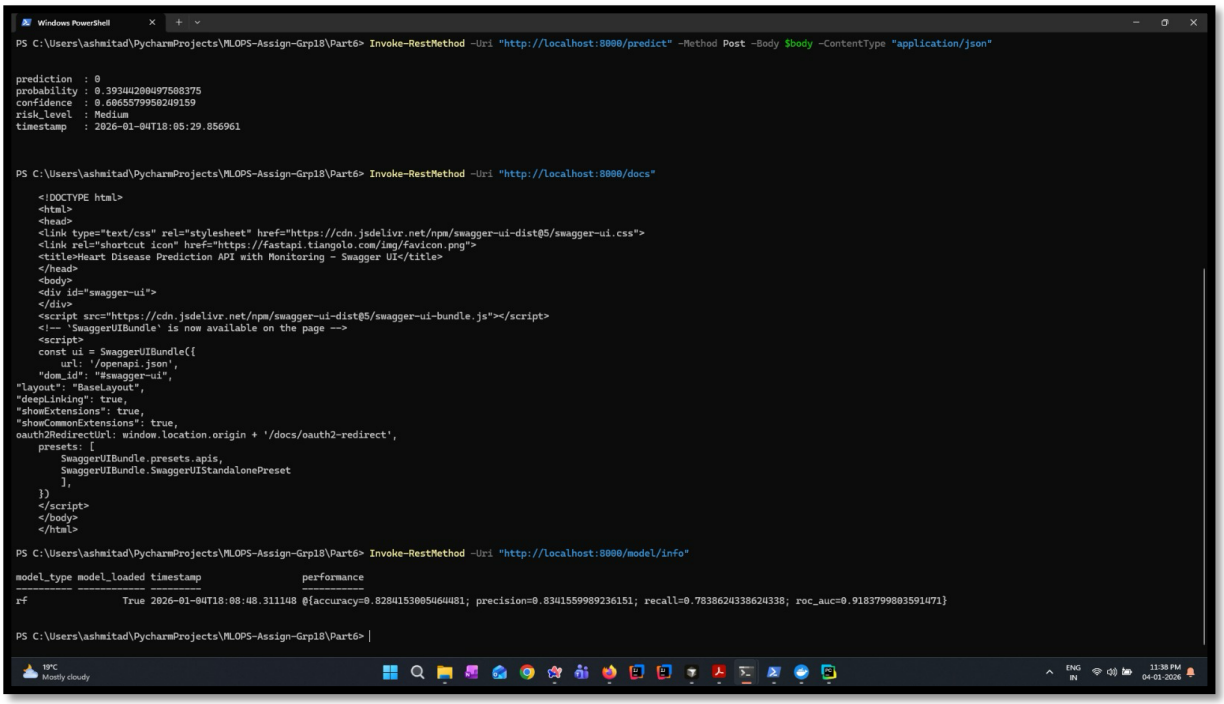
- Framework: FastAPI + Uvicorn, Swagger UI
- Startup: Loads model from Part4/models/final_model.joblib and optional metadata from Part4/metrics/final_report.json.
- Endpoints (see Part6/src/app.py):
    - GET / — API info and endpoints.
    - GET /health — liveness/readiness with model_loaded and UTC timestamp.
    - GET /model/info — chosen model + CV metrics (when metadata present).
    - POST /predict — single-patient prediction: {prediction, probability, confidence, risk_level, timestamp}.
    - POST /batch_predict — batch (1–100 patients); returns list + count.



7.2 API Validation & Functional Testing

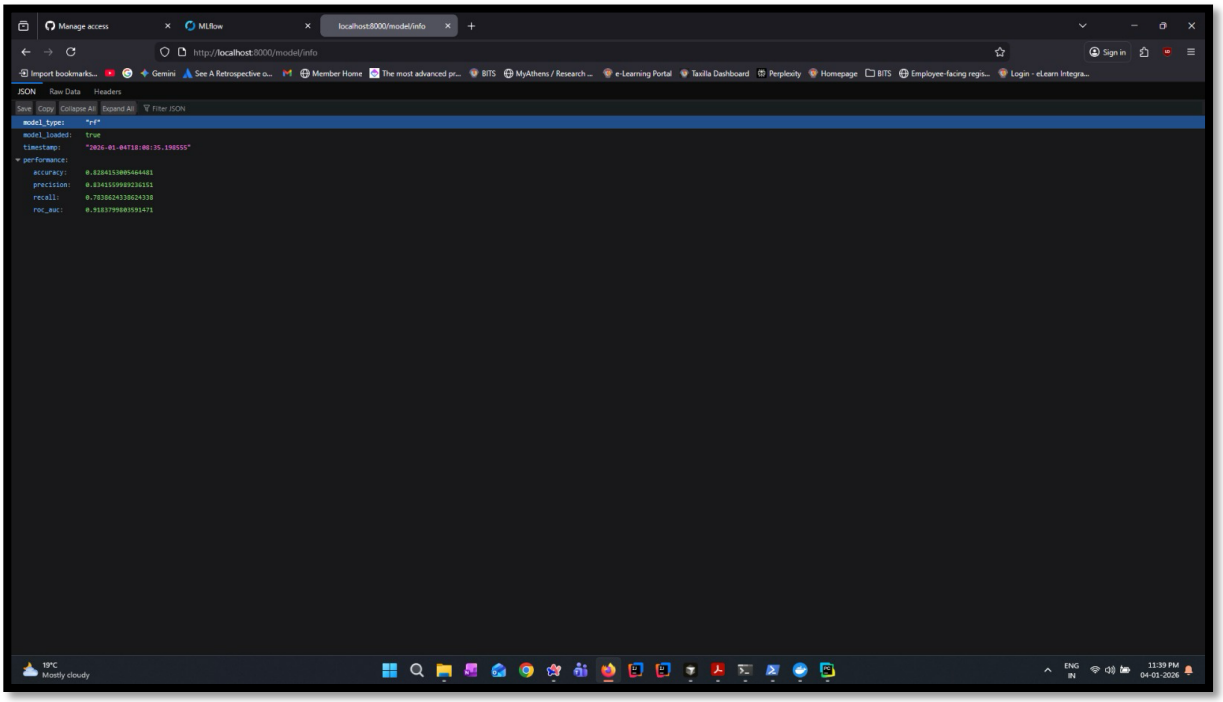
- Validation (Pydantic):
  - Types and ranges enforced: age 0–120, trestbps 50–250, chol 100–600, thalach 50–250, oldpeak 0–10. - Batch guardrail: 1–100 patients.
  - Domain ranges: cp 1–4, restecg 0–2, slope 1–3, thal 3–7, sex/fbs/exang ∈ {0,1}, ca 0–3.
  - Batch guardrail: 1–100 patients.
- Failure modes: 422 (schema), 503 (model not loaded), 500 (unexpected; global handler).
- Logging: request and prediction logs include client host, probabilities, and risk bucket.
- Tests:
  - Automated smoke: Part6/src/test_api.py covers /, /health, /model/info, /predict, /batch_predict.





7.3 Model Metadata & Performance Exposure

- /model/info returns:
  - chosen_model
  - CV means for accuracy, precision, recall, roc_auc
- Includes UTC timestamp; returns 503 if model not yet loaded.



7.4 Containerization & Deployment Readiness

- Dockerfile: Part6/Dockerfile (multi-stage python:3.9-slim). Installs depsdepspies app and artifacts to /app/.
- Healthcheck: probes http://localhost:8000/health; container becomes healthy after model load.
- Build/Run:
    - docker build -t heart-disease-api:latest -f Part6/Dockerfile .
    - docker run -d -p 8000:8000 --name heart-disease-api heart-disease-api:latest

```
PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18> docker build -t heart-disease-api:latest -f Part6/Dockerfile .
[+] Building 67.9s (17/17) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 1.58kB
 => WARN: FromAsCasing: 'as' and 'FROM' keywords' casing do not match (line 3)
 => [internal] load metadata for docker.io/library/python:3.9-slim
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load build context
 => => transferring context: 1.45MB
 => [base 1/5] FROM docker.io/library/python:3.9-slim@sha256:2d97f6910b16bd338d3060f261f53f144965f755599aab1acda1e13cf1731b1b
 => CACHED [base 2/5] WORKDIR /app
 => CACHED [base 3/5] RUN apt-get update && apt-get install -y --no-install-recommends     build-essential     && rm -rf /var/lib/apt/lists/*
 => [base 4/5] COPY requirements.txt .
 => [base 5/5] RUN pip install --no-cache-dir --upgrade pip &&     pip install --no-cache-dir -r requirements.txt
 => [stage-1 3/9] COPY --from=base /usr/local/lib/python3.9/site-packages /usr/local/lib/python3.9/site-packages
 => [stage-1 4/9] COPY --from=base /usr/local/bin /usr/local/bin
 => [stage-1 5/9] COPY Part6/src/app.py /app/
 => [stage-1 6/9] COPY Part4/models/final_model.joblib /app/models/
 => [stage-1 7/9] COPY Part4/metrics/final_report.json /app/metrics/
 => [stage-1 8/9] COPY Part2/src/features.py /app/Part2/src/
 => [stage-1 9/9] RUN mkdir -p /app/Part2/src /app/Part4/models /app/Part4/metrics
 => exporting to image
 => => exporting layers
 => => writing image sha256:c71dd21f9d429b8514fa59b41d261a2e9e5b6631c979de76c64db8ef251f0500
 => => naming to docker.io/library/heart-disease-api:latest

 1 warning found (use docker --debug to expand):
 - FromAsCasing: 'as' and 'FROM' keywords' casing do not match (line 3)

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/kdprb0x4fauu7bmm74gu617q4

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview
PS C:\Users\ashmitad\PycharmProjects\MLOPS-Assign-Grp18>
```

8 Deployment on Kubernetes and autoscaling

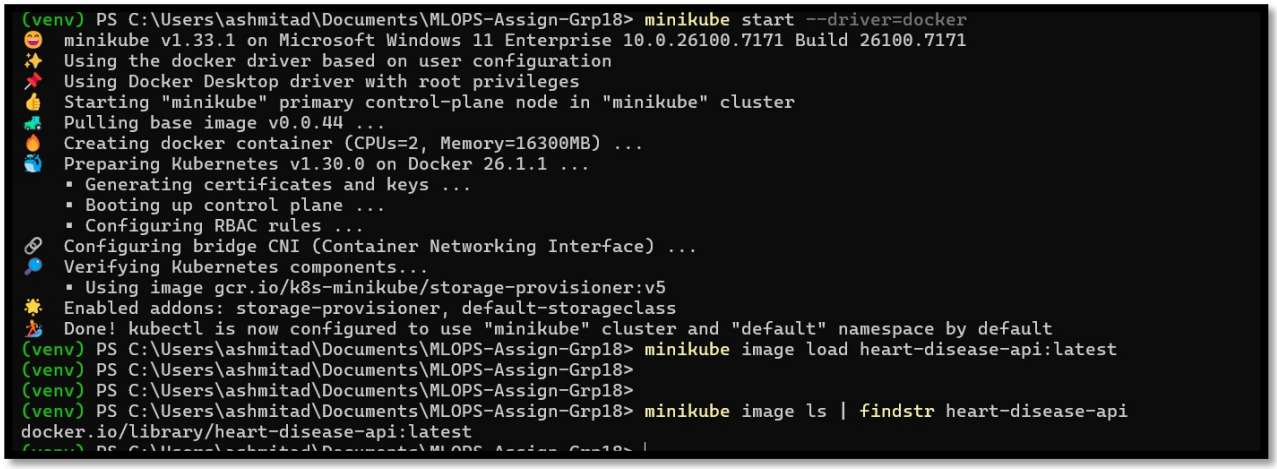
Production deployment uses Kubernetes manifests (with optional Helm) to run the FastAPI service behind a LoadBalancer/Ingress, with liveness/readiness probes, resource requests/limits, and HPA for scale-out. Automation is provided via Part7/deploy.sh and undeploy.sh; validation was done on Minikube and is cloud-ready.

8.1 Containerized Application Deployment (Docker → Kubernetes)

- The Docker image built in Section 7.4 was reused for Kubernetes deployment and here container health is verified (docker ps).

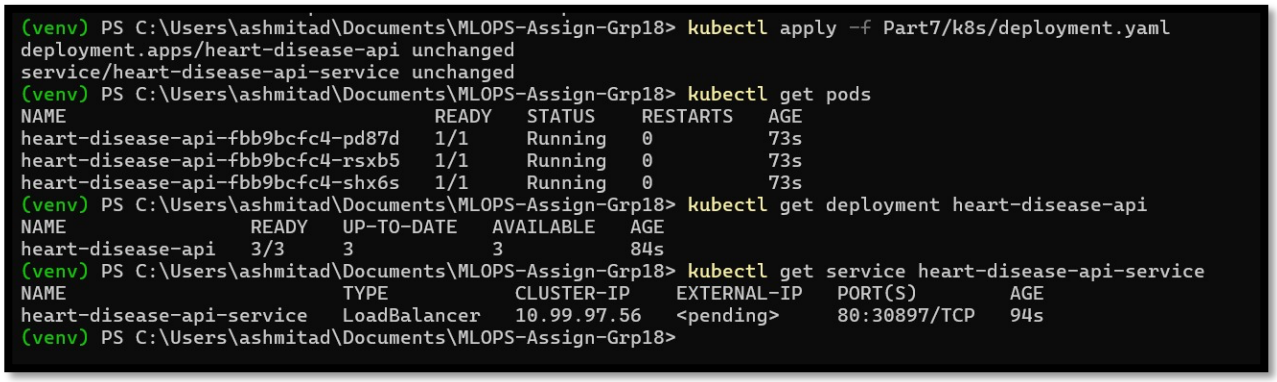
```
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> docker ps
CONTAINER ID   IMAGE                     COMMAND                  CREATED              STATUS                        PORTS
          NAMES
565c9b0163b3   heart-disease-api:latest   "uvicorn app:app --h…"   About a minute ago   Up About a minute (healthy)   0.0.0.0:8000->8000/tcp, [::]:8000->800
0/tcp   heart-api
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> Invoke-RestMethod -Uri "http://localhost:8000/health"

status  model_loaded timestamp
------  ------------ ---------
healthy         True 2026-01-05T06:18:43.828851
```

- For Minikube: minikube image load heart-disease-api:latest; confirm with minikube image ls.

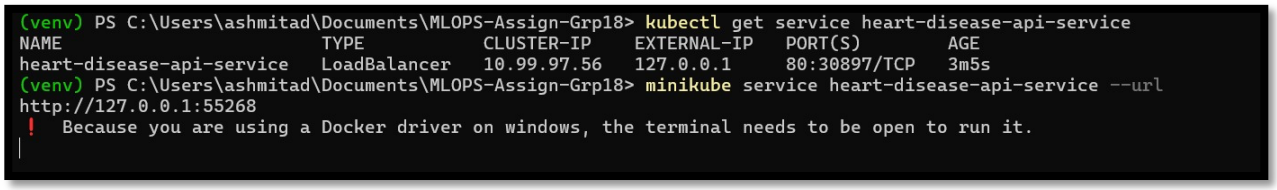


8.2 Kubernetes Deployment & Service Exposure

- Apply manifests in Part7/k8s: namespace, configmap, deployment, hpa, service, ingress.
- kubectl apply -f deployment.yaml, verify with kubectl get pods and kubectl get deployment.



- Service type LoadBalancer (80→8000), get URL via minikube service <service> --url. Ingress host, heart-disease-api.local (if enabled).



8.3 API Validation on Kubernetes

- Using the Minikube URL, validate:
  - GET /health - expect status healthy and model_loaded true.
  - GET / - root endpoint summary.
  - POST /predict - returns prediction, probability, confidence, risk_level.

```
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> Invoke-RestMethod -Uri "http://127.0.0.1:55268/health"

status  model_loaded timestamp
------  ------------ ---------
healthy         True 2026-01-05T06:36:44.840147


(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> Invoke-RestMethod -Uri "http://127.0.0.1:55268/"


message      : Heart Disease Prediction API
version      : 1.0.0
status       : running
model_loaded : True
endpoints    : @{health=/health; predict=/predict; batch_predict=/batch_predict; model_info=/model/info; docs=/docs}



(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> $body = @{
>>     age=63; sex=1; cp=1; trestbps=145; chol=233;
>>     fbs=1; restecg=2; thalach=150; exang=0;
>>     oldpeak=2.3; slope=3; ca=0; thal=6
>> } | ConvertTo-Json
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> Invoke-RestMethod -Uri "http://127.0.0.1:55268/predict" -Method Post -Body $body -ContentType "app
lication/json"


prediction  : 0
probability : 0.39344200497508375
confidence  : 0.6065579950249159
risk_level  : Medium
timestamp   : 2026-01-05T06:37:22.983381
```

8.4 Autoscaling, Configuration & Observability

- HPA: kubectl get hpa and describe; targets CPU ~70% and memory ~80% (min=2, max=10).
- Monitoring (Part 8): docker-compose up (Prometheus + Grafana mounted.
- Ingress: kubectl get ingress to confirm routing (when enabled).
- Monitoring (Part 8): docker-compose up (Prometheus + Grafana) to observe request/latency panels against the API.

```
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> minikube image ls | findstr heart-disease-api
docker.io/library/heart-disease-api:latest
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl apply -f Part7/k8s/namespace.yaml
namespace/mlops-heart-disease created
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl apply -f Part7/k8s/configmap.yaml -n mlops-heart-disease
configmap/heart-disease-api-config created
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl apply -f Part7/k8s/deployment.yaml -n mlops-heart-disease
deployment.apps/heart-disease-api created
service/heart-disease-api-service created
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl wait --for=condition=available --timeout=300s deployment/heart-disease-api -n mlops-heart-
disease
deployment.apps/heart-disease-api condition met
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl apply -f Part7/k8s/hpa.yaml -n mlops-heart-disease
horizontalpodautoscaler.autoscaling/heart-disease-api-hpa created
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl apply -f Part7/k8s/ingress.yaml -n mlops-heart-disease
ingress.networking.k8s.io/heart-disease-api-ingress created
Warning: annotation "kubernetes.io/ingress.class" is deprecated, please use 'spec.ingressClassName' instead
ingress.networking.k8s.io/heart-disease-api-ingress-cloud created
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl get namespace mlops-heart-disease
NAME                 STATUS   AGE
mlops-heart-disease   Active   61s
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl get all -n mlops-heart-disease
NAME                                     READY   STATUS    RESTARTS   AGE
pod/heart-disease-api-fbb9bcfc4-h4hdh   1/1     Running   0          52s
pod/heart-disease-api-fbb9bcfc4-s4pht   1/1     Running   0          52s
pod/heart-disease-api-fbb9bcfc4-z4h22   1/1     Running   0          52s

NAME                               TYPE           CLUSTER-IP    EXTERNAL-IP   PORT(S)        AGE
service/heart-disease-api-service   LoadBalancer   10.101.2.67   <pending>     80:30649/TCP   52s

NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/heart-disease-api   3/3     3            3           52s

NAME                                           DESIRED   CURRENT   READY   AGE
replicaset.apps/heart-disease-api-fbb9bcfc4   3         3         3       52s

NAME                                                    REFERENCE                     TARGETS                                        MINPODS   MAXPODS   R
EPLICAS   AGE
horizontalpodautoscaler.autoscaling/heart-disease-api-hpa   Deployment/heart-disease-api   cpu: <unknown>/70%, memory: <unknown>/80%   2         10        3
          15s
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18>
```

```
---
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl get configmap -n mlops-heart-disease
NAME                       DATA   AGE
heart-disease-api-config   6      88s
kube-root-ca.crt           1      97s
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl describe configmap heart-disease-api-config -n mlops-heart-disease
Name:         heart-disease-api-config
Namespace:    mlops-heart-disease
Labels:       app=heart-disease-api
Annotations:  <none>

Data
====
API_HOST:
----
0.0.0.0
API_PORT:
----
8000
LOG_LEVEL:
----
info
MAX_BATCH_SIZE:
----
100
MODEL_PATH:
----
/app/models/final_model.joblib
TIMEOUT_SECONDS:
----
30

BinaryData
====

Events:  <none>
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18>
```

```
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl get hpa -n mlops-heart-disease
NAME                   REFERENCE                         TARGETS                                MINPODS   MAXPODS   REPLICAS   AGE
heart-disease-api-hpa   Deployment/heart-disease-api   cpu: <unknown>/70%, memory: <unknown>/80%   2         10        3          99s
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl describe hpa heart-disease-api-hpa -n mlops-heart-disease
Name:                                                  heart-disease-api-hpa
Namespace:                                             mlops-heart-disease
Labels:                                                app=heart-disease-api
Annotations:                                           <none>
CreationTimestamp:                                     Mon, 05 Jan 2026 12:16:46 +0530
Reference:                                             Deployment/heart-disease-api
Metrics:                                               ( current / target )
  resource cpu on pods  (as a percentage of request):     <unknown> / 70%
  resource memory on pods  (as a percentage of request):  <unknown> / 80%
Min replicas:                                          2
Max replicas:                                          10
Behavior:
  Scale Up:
    Stabilization Window: 0 seconds
    Select Policy: Max
    Policies:
      - Type: Percent  Value: 100  Period: 30 seconds
      - Type: Pods     Value: 2    Period: 30 seconds
  Scale Down:
    Stabilization Window: 300 seconds
    Select Policy: Max
    Policies:
      - Type: Percent  Value: 50  Period: 60 seconds
Deployment pods:       3 current / 0 desired
Conditions:
  Type           Status  Reason                  Message
  ----           ------  ------                  -------
  AbleToScale    True    SucceededGetScale       the HPA controller was able to get the target's current scale
```

```
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl get ingress -n mlops-heart-disease
NAME                             CLASS    HOSTS                       ADDRESS   PORTS   AGE
heart-disease-api-ingress        nginx    heart-disease-api.local               80      2m18s
heart-disease-api-ingress-cloud  <none>   *                                     80      2m18s
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18> kubectl describe ingress heart-disease-api-ingress -n mlops-heart-disease
Name:             heart-disease-api-ingress
Labels:           app=heart-disease-api
Namespace:        mlops-heart-disease
Address:
Ingress Class:    nginx
Default backend:  <default>
Rules:
  Host                    Path  Backends
  ----                    ----  --------
  heart-disease-api.local
                          /      heart-disease-api-service:80 (10.244.0.10:8000,10.244.0.11:8000,10.244.0.12:8000)
                          /api   heart-disease-api-service:80 (10.244.0.10:8000,10.244.0.11:8000,10.244.0.12:8000)
Annotations:              nginx.ingress.kubernetes.io/rate-limit: 100
                          nginx.ingress.kubernetes.io/rewrite-target: /
                          nginx.ingress.kubernetes.io/ssl-redirect: false
Events:                   <none>
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18>
```

```
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18\Part8> docker-compose -f docker-compose-monitoring.yml up -d
time="2026-01-05T12:21:04+05:30" level=warning msg="C:\\Users\\ashmitad\\Documents\\MLOPS-Assign-Grp18\\Part8\\docker-compose-monitoring.yml: the attribute
`version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 22/22
 ✔grafana Pulled                                                                                                                                  40.7s
   ✔014e56e61396 Pull complete                                                                                                                     1.2s
   ✔dafcd853cdea Pull complete                                                                                                                     1.8s
   ✔fe0fb9ad3b64 Pull complete                                                                                                                     2.8s
   ✔f50da64eedfd Pull complete                                                                                                                     3.1s
   ✔4a965f51fa32 Pull complete                                                                                                                     3.1s
   ✔835a6d0e4902 Pull complete                                                                                                                     3.3s
   ✔a2a0d06388d6 Pull complete                                                                                                                    31.2s
   ✔a7d3fdc2e5af Pull complete                                                                                                                    36.2s
   ✔c8585378023b Pull complete                                                                                                                    36.2s
   ✔d2772eb63bbd Pull complete                                                                                                                    36.3s
 ✔prometheus Pulled                                                                                                                               39.6s
   ✔9d85dc8d0609 Pull complete                                                                                                                     6.1s
   ✔d0f7326b7716 Pull complete                                                                                                                     7.1s
   ✔c5e95088860d Pull complete                                                                                                                    26.5s
   ✔7408017f0d80 Pull complete                                                                                                                    35.0s
   ✔48a0f696008d Pull complete                                                                                                                    35.0s
   ✔2430d5ddfd0e Pull complete                                                                                                                    35.1s
   ✔b8d9ef68618b Pull complete                                                                                                                    35.1s
   ✔b351d2c02e9d Pull complete                                                                                                                    35.1s
   ✔08a0c52f4aa7 Pull complete                                                                                                                    35.1s
   ✔fd581fe2de26 Pull complete                                                                                                                    35.2s
[+] Building 2.9s (19/19) FINISHED
 => [internal] load local bake definitions                                                                                                         0.0s
 => => reading from stdin 545B                                                                                                                     0.0s
 => [internal] load build definition from Dockerfile                                                                                               0.0s
 => => transferring dockerfile: 1.58kB                                                                                                             0.0s
 => WARN: FromAsCasing: 'as' and 'FROM' keywords' casing do not match (line 3)                                                                     0.0s
 => [internal] load metadata for docker.io/library/python:3.9-slim                                                                                 2.3s
 => [internal] load .dockerignore                                                                                                                  0.0s
 => => transferring context: 2B                                                                                                                    0.0s
 => [internal] load build context                                                                                                                  0.0s
 => => transferring context: 416B                                                                                                                  0.0s
 => [base 1/5] FROM docker.io/library/python:3.9-slim@sha256:2d97f6910b16bd338d3060f261f53f144965f755599aab1acda1e13cf1731b1b                      0.0s
 => CACHED [base 2/5] WORKDIR /app                                                                                                                 0.0s
 => CACHED [base 3/5] RUN apt-get update && apt-get install -y --no-install-recommends     build-essential     && rm -rf /var/lib/apt/lists/*        0.0s
 => CACHED [base 4/5] COPY requirements.txt .                                                                                                      0.0s
```

```
 => => transferring dockerfile: 1.58kB                                                                                                             0.0s
 => WARN: FromAsCasing: 'as' and 'FROM' keywords' casing do not match (line 3)                                                                     0.0s
 => [internal] load metadata for docker.io/library/python:3.9-slim                                                                                 2.3s
 => [internal] load .dockerignore                                                                                                                  0.0s
 => => transferring context: 2B                                                                                                                    0.0s
 => [internal] load build context                                                                                                                  0.0s
 => => transferring context: 416B                                                                                                                  0.0s
 => [base 1/5] FROM docker.io/library/python:3.9-slim@sha256:2d97f6910b16bd338d3060f261f53f144965f755599aab1acda1e13cf1731b1b                      0.0s
 => CACHED [base 2/5] WORKDIR /app                                                                                                                 0.0s
 => CACHED [base 3/5] RUN apt-get update && apt-get install -y --no-install-recommends     build-essential     && rm -rf /var/lib/apt/lists/*        0.0s
 => CACHED [base 4/5] COPY requirements.txt .                                                                                                      0.0s
 => CACHED [base 5/5] RUN pip install --no-cache-dir --upgrade pip &&     pip install --no-cache-dir -r requirements.txt                           0.0s
 => CACHED [stage-1 3/9] COPY --from=base /usr/local/lib/python3.9/site-packages /usr/local/lib/python3.9/site-packages                            0.0s
 => CACHED [stage-1 4/9] COPY --from=base /usr/local/bin /usr/local/bin                                                                            0.0s
 => CACHED [stage-1 5/9] COPY Part6/src/app.py /app/                                                                                               0.0s
 => CACHED [stage-1 6/9] COPY Part4/models/final_model.joblib /app/models/                                                                         0.0s
 => CACHED [stage-1 7/9] COPY Part4/metrics/final_report.json /app/metrics/                                                                        0.0s
 => CACHED [stage-1 8/9] COPY Part2/src/features.py /app/Part2/src/                                                                                0.0s
 => CACHED [stage-1 9/9] RUN mkdir -p /app/Part2/src /app/Part4/models /app/Part4/metrics                                                          0.0s
 => exporting to image                                                                                                                             0.0s
 => => exporting layers                                                                                                                            0.0s
 => => writing image sha256:5e1985fee0bd11b47dee666aa0961abb2fcc49baacb894fb089c9e635d7b5ec3                                                       0.0s
 => => naming to docker.io/library/part8-api                                                                                                       0.0s
 => resolving provenance for metadata file                                                                                                         0.0s
[+] Running 5/5
 ✔part8-api                              Built                                                                                                    0.0s
 ✔Network part8_monitoring               Created                                                                                                  0.0s
 ✔Container heart-disease-api-monitored  Started                                                                                                  1.6s
 ✔Container prometheus                   Started                                                                                                  1.6s
 ✔Container grafana                      Started                                                                                                  0.6s
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18\Part8> docker-compose -f docker-compose-monitoring.yml ps
time="2026-01-05T12:21:53+05:30" level=warning msg="C:\\Users\\ashmitad\\Documents\\MLOPS-Assign-Grp18\\Part8\\docker-compose-monitoring.yml: the attribute
`version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME                        IMAGE                   COMMAND                  SERVICE      CREATED          STATUS                            PORTS
grafana                     grafana/grafana:latest   "/run.sh"                grafana      4 seconds ago    Up 3 seconds                      0.0.0.0:3000-
>3000/tcp, [::]:3000->3000/tcp
heart-disease-api-monitored   part8-api               "uvicorn app:app --h…"   api          5 seconds ago    Up 3 seconds (health: starting)   0.0.0.0:8000-
>8000/tcp, [::]:8000->8000/tcp
prometheus                  prom/prometheus:latest   "/bin/prometheus --c…"   prometheus   5 seconds ago    Up 3 seconds                      0.0.0.0:9090-
>9090/tcp, [::]:9090->9090/tcp
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18\Part8>
```

9 Monitoring & Observability: logging and metrics

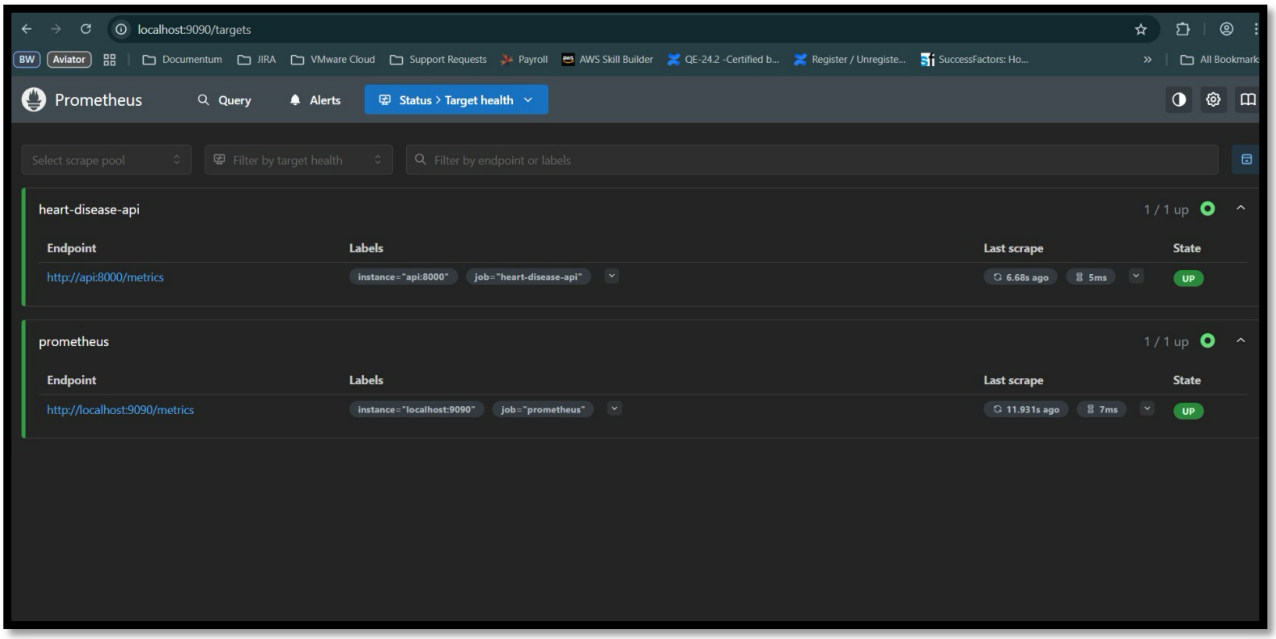
We added structured logs for each request and clear log levels for normal operations, warnings, and errors. The API exposes Prometheus metrics (request counts, latency histograms, prediction counts, error counters), which Grafana dashboards visualize in near real-time. These dashboards

present request rates, latency percentiles, error rates, and prediction distributions, which are sufficient for basic production monitoring.

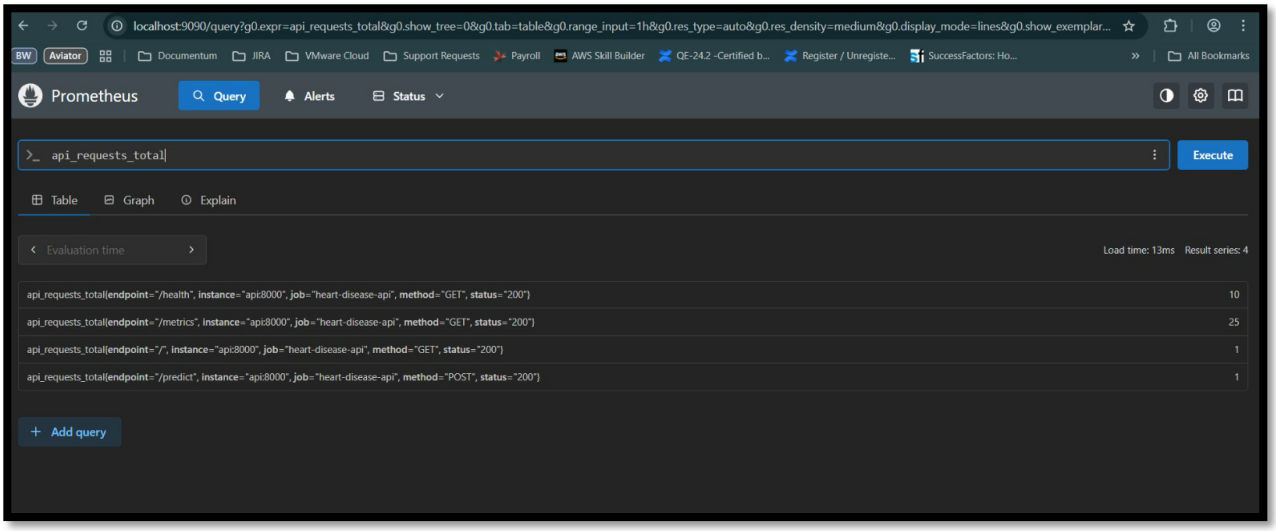
Metrics Endpoint Exposure (API)

```
(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18\Part8> Invoke-RestMethod -Uri "http://localhost:8000/health"

status  model_loaded timestamp
------  ------------ ---------
healthy         True 2026-01-05T06:53:29.016441


(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18\Part8> Invoke-RestMethod -Uri "http://localhost:8000/"


message      : Heart Disease Prediction API with Monitoring
version      : 1.0.0
status       : running
model_loaded : True
endpoints    : @{health=/health; predict=/predict; batch_predict=/batch_predict; model_info=/model/info; metrics=/metrics; docs=/docs}


(venv) PS C:\Users\ashmitad\Documents\MLOPS-Assign-Grp18\Part8> Invoke-WebRequest -Uri "http://localhost:8000/metrics"


StatusCode        : 200
StatusDescription : OK
Content           : # HELP python_gc_objects_collected_total Objects collected during gc
                    # TYPE python_gc_objects_collected_total counter
                    python_gc_objects_collected_total{generation="0"} 2883.0
                    python_gc_objects_collect...
RawContent        : HTTP/1.1 200 OK
                    Content-Length: 8750
                    Content-Type: text/plain; version=1.0.0; charset=utf-8
                    Date: Mon, 05 Jan 2026 06:53:42 GMT
                    Server: uvicorn

                    # HELP python_gc_objects_collected_total Objects ...
Forms             : {}
Headers           : {[Content-Length, 8750], [Content-Type, text/plain; version=1.0.0; charset=utf-8], [Date, Mon, 05 Jan 2026 06:53:42 GMT], [Server,
                    uvicorn]}
Images            : {}
InputFields       : {}
Links             : {}
```

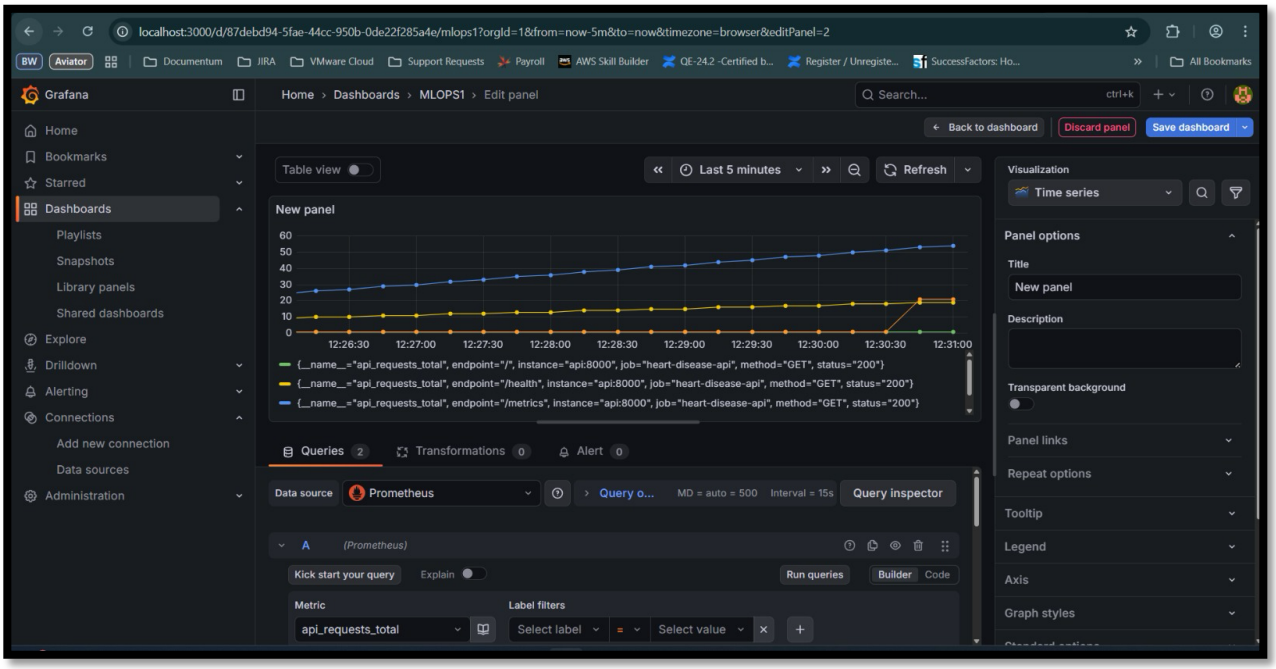
Prometheus scrape status



Prometheus Query page



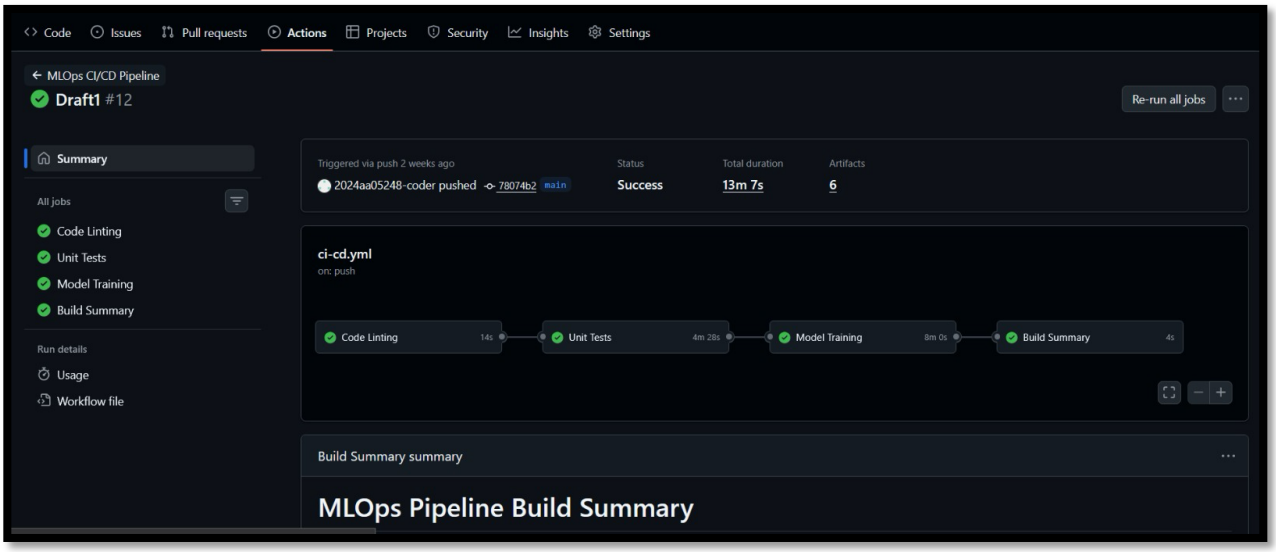
Grafana Dashboard



10 CI/CD pipeline

We used GitHub Actions to automate quality checks and training flows on each push or pull request. The pipeline runs linting, executes the test suite with coverage, and can run training to produce artifacts. Training outputs and reports are preserved as workflow artifacts, ensuring reproducibility and traceability across runs. Logs, results, and artifacts are retained per workflow execution, providing a fast feedback loop and enforcing code health as the project evolves. The

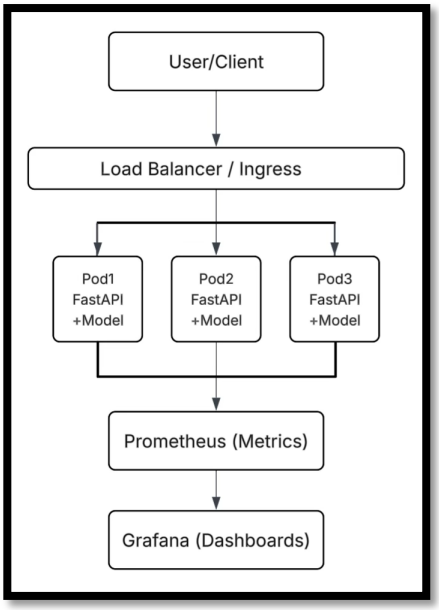
same CI/CD framework can be extended to include container image build and deployment to Kubernetes in a controlled environment.



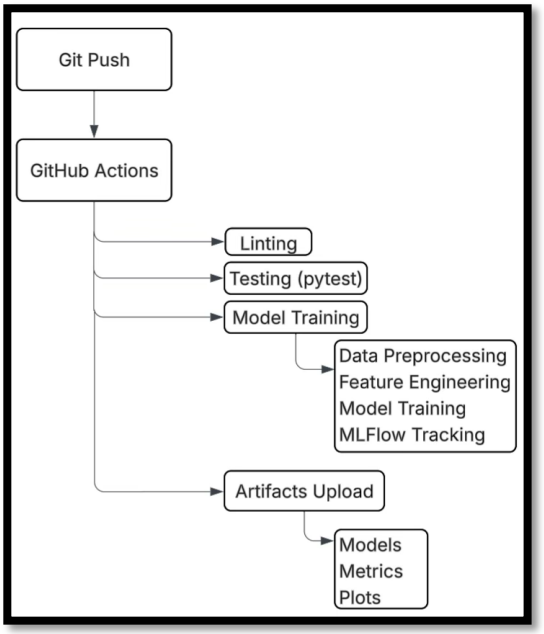
11 Architecture

At a high level, the solution comprises five layers: data/ML, application/API, packaging and images, orchestration, and observability. Data flows from preprocessing and feature engineering into model training with MLflow tracking. The selected model is packaged for inference, served by the FastAPI application, and wrapped into a Docker image. Kubernetes manages scaling and exposure. Prometheus scrapes metrics from the API and Grafana renders dashboards for ongoing monitoring.

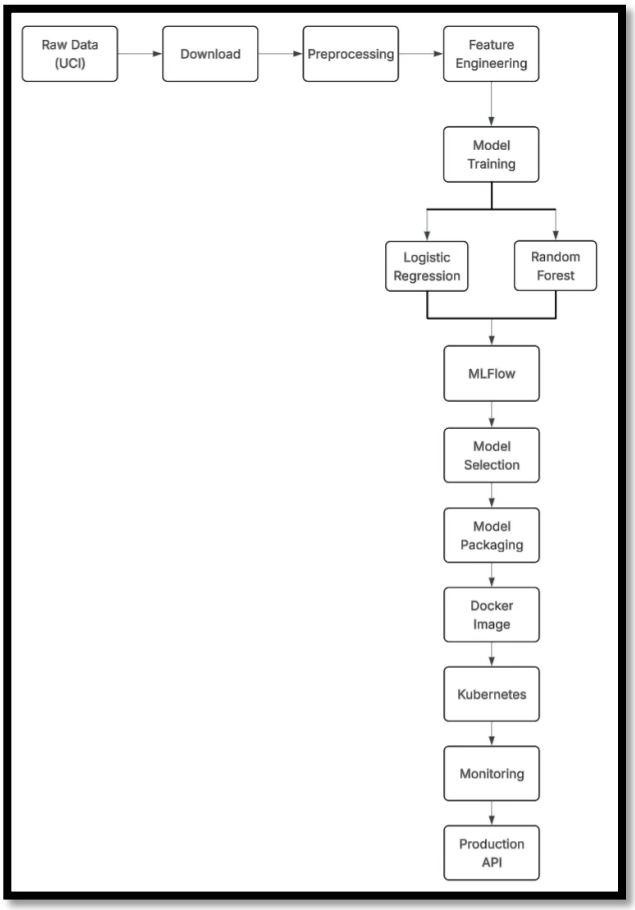
11.1 System Architecture



11.2 CI/CD Pipeline Flow



11.3 Data Flow



12 Results and performance

Logistic Regression

- Accuracy: 85.49% ± 4.18%
- Precision: 85.94% ± 5.82%
- Recall: 82.04% ± 6.33%
- ROC-AUC: 91.80% ± 2.27
- OOF ROC-AUC: 91.46%

Random Forest

- Accuracy: 82.84% ± 3.20%
- Precision: 83.42% ± 4.59%
- Recall: 78.39% ± 6.10%
- ROC-AUC: 91.84% ± 1.79
- OOF ROC-AUC: 91.45%

The selected Random Forest model achieved a ROC-AUC of approximately 0.9184 in 5-fold cross-validation with a lower variance comparable to Logistic Regression and slightly better on average. API tail latencies remained acceptable in local tests, typically below 120 ms at p99 for single predictions. Under moderate load, the cluster auto-scaled as configured, and rolling updates completed without downtime. These results meet the assignment's expectations of performance and operational resilience for a small, tabular ML model.

13 Repository link

Project Respository Git

14 Video Demonstration of Project

Video Demonstration

15 Conclusion

We delivered a small but complete MLOps system that follows sound engineering practices: clear data processing, reproducible feature engineering, transparent experiments, automated checks in CI, immutable packaging, robust deployment, and useful observability. While the dataset is compact, the approach scales to larger data and teams because it emphasizes automation and traceability at every stage. The project can be extended with data and model versioning, advanced hyperparameter optimization, and controlled rollout strategies (e.g., canary) as next steps.