

MLOps Engineering

Course Duration ~110 Hours · 14 Weeks · Live Online · Infrastructure-First · Portfolio of Deployed Systems

MLflow

Docker & Kubernetes

CI/CD for ML

Monitoring & Drift

LLMOps

FinOps & IaC

In short: Our other AI courses teach you to *build* AI — this one teaches you to *operate* it. A live, infrastructure-first program that takes models from experiment to production: versioned, monitored, automatically retrained, and running at a cost the business can justify, for both classical ML and LLM workloads. Across 11 modules you ship reproducible MLflow pipelines, drift-triggered retraining in Airflow/Kubeflow, model-gating CI/CD with GitOps rollback, KServe deployments with canary rollouts, a full Prometheus + Grafana + Evidently monitoring stack, a vLLM-based LLMOps stack with cost attribution, and a Terraform-rebuilt platform — defended in a capstone review. Built for fresh IT graduates who want an infrastructure-side entry into AI and for DevOps, cloud, backend and data professionals whose experience transfers directly.

Duration	14 weeks · ~110 hours · live + hands-on	Format	Live online · infrastructure-first · portfolio of deployed systems
Prerequisite	Working Python, comfort with the command line and Git. No deep ML theory — you'll operate models, not derive them.	Curriculum	11 modules, from the production gap to cloud, cost and governance
Portfolio	7 deployed, reproducible systems plus a defended capstone	Ideal for	Fresh IT graduates and DevOps/cloud/backend/data professionals moving into ML platform work

Course Highlights

- Every module leads with what you build and ship, not what you "learn about"
- Reproducibility as a discipline: MLflow, DVC, model registry, exact rebuilds months later
- Automated retraining triggered by drift signal, not by calendar
- Kubernetes taught after Docker is solid — including when *not* to reach for it
- Monitoring for models that degrade quietly: drift, canary and shadow deployments
- Deep LLMOps: eval pipelines in CI, tracing, vLLM serving, token-level cost attribution
- FinOps for GPU spend, IaC with Terraform, governance and audit readiness

Hands-on Projects & Portfolio

- **End-to-End ML Pipeline** — MLflow tracking → model registry → containerized service → deployed endpoint, reproducible from a clean clone
- **Drift-Triggered Retraining** — automated retraining pipeline in Airflow or Kubeflow, triggered by an Evidently drift alert
- **A/B & Canary on Kubernetes** — KServe traffic splitting, compared on latency and accuracy
- **Full Monitoring Stack** — Prometheus + Grafana dashboards with alerting on drift and prediction distribution shift
- **Model-Gating CI/CD** — a pipeline that tests, validates and gates a model before it can reach production
- **LLM Serving Deployment** — vLLM with an eval regression suite and per-request cost tracking
- **Platform as Code** — infrastructure-as-code teardown and rebuild of the whole platform from Terraform

Tools & Stack

Python · Git · Docker · Kubernetes · MLflow · Weights & Biases · DVC · Apache Airflow · Kubeflow · KServe · BentoML · FastAPI · vLLM · Evidently · Prometheus · Grafana · LangSmith · Langfuse · GitHub Actions · Argo CD · Terraform · AWS SageMaker · GCP Vertex AI · Azure ML

Career Outcomes

MLOps is one of the more durable AI roles because it sits on DevOps and platform fundamentals that don't churn every six months — and enterprises now need classical ML operations and LLMOps practices side by side. The natural on-ramps are DevOps, cloud, backend and data engineering: existing infrastructure experience transfers directly. This course targets:

- MLOps Engineer · ML Platform Engineer · ML Engineer · AI Infrastructure Engineer
- LLMOps Engineer · Data Engineer (ML focus) · DevOps Engineer moving into ML · Site Reliability Engineer for AI systems

Who This Course Is For

- Final-year students and fresh IT graduates who want an infrastructure-side entry into AI
- DevOps, cloud and platform engineers adding ML workloads to what they already operate
- Data scientists tired of handing notebooks over the wall and wanting to own production
- Backend and data engineers moving into ML platform work
- SREs and architects responsible for AI systems' uptime, cost and governance

Module 1: The Production Gap

8 Hours

YOU SHIP

- A written post-mortem analysis of a real production ML incident — what failed, who should have owned it, and the maturity gaps that let it happen

TOPICS

- Why most models never ship; the ML lifecycle end to end
- MLOps maturity levels; who owns what across data science / engineering / ops
- Reading a real production incident post-mortem on day one

Assessment: Quiz + Lab

Module 2: Engineering Foundations for ML

8 Hours

YOU SHIP

- A production-structured ML repository — tested, environment-pinned, configuration-managed — that becomes the base for every later module

TOPICS

- Git workflows for ML repos; project structure; dependency and environment management
- Testing ML code; configuration management
- The Linux/networking basics the rest of the stack assumes

Assessment: Quiz + Lab

Module 3: Experiment Tracking & Reproducibility

10 Hours

YOU SHIP

- A tracked, versioned training workflow — MLflow runs, registered models, DVC-versioned data — where a result from three months ago can be rebuilt exactly

TOPICS

- MLflow and Weights & Biases; run tracking, parameters and metrics; artifact storage
- The model registry; model lineage
- DVC for data and pipeline versioning

Assessment: Quiz + Lab

Module 4: Data & Feature Engineering at Scale

8 Hours

YOU SHIP

- A validated data pipeline with schema-enforced quality gates and a skew check that catches training-serving drift before it reaches customers

TOPICS

- Data validation and quality gates; schema enforcement
- Feature stores and online/offline consistency
- Training-serving skew and how to detect it

Assessment: Quiz + Lab**Module 5: Containerization & Model Serving**

12 Hours

YOU SHIP

- A containerized model service — FastAPI or BentoML — with warm-up, request batching, and a measured latency budget, plus a defended GPU-vs-CPU serving decision

TOPICS

- Docker done properly for ML images
- FastAPI and BentoML services; real-time vs batch vs streaming inference
- Model loading and warm-up; request batching; latency budgets; GPU vs CPU serving decisions

Assessment: Quiz + Lab**Module 6: Pipeline Orchestration**

10 Hours

YOU SHIP

- An automated retraining pipeline in Airflow or Kubeflow — idempotent, retry-safe, parameterized — triggered by signal rather than by calendar

TOPICS

- Apache Airflow and Kubeflow Pipelines; DAG design
- Scheduling and backfills; retries and idempotency; parameterized training runs
- Automated retraining triggered by signal

Assessment: Quiz + Lab**Module 7: CI/CD for Machine Learning**

10 Hours

YOU SHIP

- A CI/CD pipeline that tests, validates and gates a model before production — with GitOps rollback so a bad model reverts with a git revert

TOPICS

- GitHub Actions / GitLab CI; why ML pipelines need continuous training on top of CI/CD
- Automated model validation gates; model approval workflows
- GitOps with Argo CD

Assessment: Quiz + Lab**Module 8: Kubernetes for ML Workloads**

12 Hours

YOU SHIP

- A KServe deployment serving a predictive model and an LLM — autoscaled, quota-bound, scale-to-zero — plus a one-page "do we even need Kubernetes?" decision brief

TOPICS

- Introduced only after Docker is solid: deployments, services, autoscaling
- KServe for serving predictive models and LLMs; scale-to-zero; resource quotas and GPU scheduling
- An honest module on when *not* to reach for Kubernetes

Assessment: Quiz + Lab

Module 9: Monitoring, Drift & Observability

12 Hours

YOU SHIP

- A full monitoring stack — Prometheus + Grafana + Evidently — with drift alerts, a canary rollout with traffic splitting, and an incident runbook for models that degrade quietly

TOPICS

- Prometheus and Grafana for system health; Evidently for data drift and concept drift
- Prediction distribution monitoring; alerting thresholds; feedback loops
- Shadow deployments; canary and A/B rollouts with traffic splitting; incident response for models that degrade quietly instead of failing loudly

Assessment: Quiz + Lab

Module 10: LLMops

10 Hours

YOU SHIP

- A production LLM serving stack — vLLM, caching, model routing — with an eval regression suite in CI, tracing, token-level cost attribution and guardrail monitoring

TOPICS

- Where operating LLMs differs from classical ML: prompt and context versioning; eval pipelines in CI
- Tracing with LangSmith / Langfuse; vLLM and inference optimization; caching; model routing
- Token-level cost attribution; guardrail monitoring

Assessment: Quiz + Lab

Module 11: Cloud, Cost & Governance + Capstone

10 Hours

YOU SHIP

- One end-to-end production ML platform — built as code with Terraform, torn down and rebuilt on demand — and defended in a capstone review

TOPICS

- SageMaker / Vertex AI / Azure ML managed paths vs self-hosted; infrastructure as code with Terraform
- FinOps for GPU spend; security, access control, audit trails
- Model documentation and regulatory readiness; capstone design, build and defense

Assessment: Capstone Project + Review