



ClearML ML Engineer Certification

- 3 Days
- Lecture and Hands-on Labs

Course Overview

This class prepares students for the ClearML Engineer certification. ClearML is an open-source MLOps platform that enables teams to seamlessly track, orchestrate, and scale machine learning workloads across Kubernetes, cloud, and hybrid environments. By the conclusion of this hands-on training, you will return to work with the skills to deploy, secure, and operate a full ClearML environment — from experiment tracking to GPU-powered model serving.

Throughout the course, you will learn to use Helm, Kubernetes, and cloud-native tools to manage ClearML at scale. You'll configure external data stores, automate agent scaling, integrate with Hugging Face and vLLM, and practice troubleshooting real-world ClearML incidents. The curriculum combines scenario-based labs with production-focused simulations.

Review this course online at <https://www.alta3.com/courses/clearml-me>

Who Should Attend

- ML Engineers
- Data Engineers
- DevOps Engineers
- AI Platform Specialists

What You'll Learn

- Build automated ML pipelines with ClearML orchestration and CI/CD in ≤ 30 minutes.
- Scale training and inference using queues and GPU agents.
- Monitor models for drift, performance, and operational health.
- Integrate Data Scientist outputs (e.g., Sarah's models) into production pipelines.
- Collaborate with Data Scientists (Sarah) and Developers (Joe) using ClearML projects.

Outline

Introduction to ClearML for ML Engineers

- 🗣️ Lecture: What is MLOps? Role in Production Workflows
- 🗣️ Lecture: ClearML Overview: The Server, The SDK, and The Agent
- 🗣️ Lecture: Introduction to ML Pipelines and Automation
- 🖥️ Lecture + Lab: Set Up Python Environment and Install ClearML Agent
- 🖥️ Lecture + Lab: Configure and Run a Local ClearML Agent (Worker)
- 🖥️ Lecture + Lab: Run a Baseline Pipeline Script

Dataset and Model Versioning

- 🗣️ Lecture: Ensuring Reproducibility with Data and Model Versioning

-  Lecture: Detecting Data Drift with Integrations (Evidently/Deepchecks)
-  Lecture: Model Management and Formats
-  Lecture + Lab: Version a Dataset
-  Lecture + Lab: Query and Validate Model Artifacts
-  Lecture + Lab: Simulate Drift and Trigger Alerts

Collaboration and Governance

-  Lecture: Collaborating with Data Scientists and Developers
-  Lecture: Meet Sarah: The Data Scientist Persona (Handoff Context)
-  Lecture: ClearML Projects and Team Visibility
-  Lecture: Governance with Model Cards for Compliance
-  Lecture + Lab: Access a Shared Project from Sarah
-  Lecture + Lab: Apply Model Card Metadata for Production
-  Lecture + Lab: Troubleshoot Pipeline Integration Issues

Automated Pipelines & Orchestration

-  Lecture: Building End-to-End ML Pipelines with ClearML
-  Lecture: Orchestration: Managing Queues and Agents
-  Lecture: Handling Pipeline Failures and Retries
-  Lecture + Lab: Define and Run a Pipeline (Data → Train → Eval)
-  Lecture + Lab: Modify Pipeline to Retry on Failure
-  Lecture + Lab: Trigger Pipeline via ClearML API

Deployment and Operations

-  Lecture: ClearML Serving: Deployment and Canary Strategies
-  Lecture: Monitoring Models for Drift and Performance
-  Lecture: Integrating Monitoring Tools (e.g., Prometheus)
-  Lecture + Lab: Deploy Model to ClearML Serving
-  Lecture + Lab: Set Up Drift Alerts for a Deployed Model
-  Lecture + Lab: Manage Compute Queues (Docker/CPU optimization)
-  Lecture + Lab: Execute a Canary Rollback (Traffic Update)

Capstone: Deploy a Real-World AI Pipeline

-  Lecture: Capstone Overview: Inventory Forecasting Pipeline
-  Lecture: Simulating Handoff from Data Scientist (Sarah)
-  Lecture: Certification Prep: Scenarios and Best Practices
-  Lecture + Lab: Build and Schedule Retraining Pipeline
-  Lecture + Lab: Automate Response to Drift (Trigger Retraining)
-  Lecture + Lab: Full System Test: Ingest → Train → Deploy
-  Lecture + Lab: Practice Certification Exam Tasks

Prerequisites

- Python - PCEP Certification or Equivalent Experience
- Familiarity with ML frameworks (e.g., PyTorch, TensorFlow) and basic DevOps (e.g., Docker, CI/CD)
- Basic Linux command-line skills