Production-Grade MLOps: Building Reliable Machine Learning Systems Using SRE Principles

19 – 23 April 2027
San Diego

AGILE LEADERS
Training Center

AGILE LEADERS
Training Center

Production-Grade MLOps: Building Reliable Machine Learning Systems Using SRE Principles

Ref: 61_86640 **Date:** 19 – 23 April 2027 **Location:** San Diego **Fees:** 14000 Euro

Overview

Operationalizing machine learning requires moving beyond experimental notebook workflows to establish high-availability, fault-tolerant architectures in live enterprise environments. This professional programme in production-grade MLOps bridges data science workflows with Site Reliability Engineering SRE paradigms to ensure statistical models function predictably under dynamic production workloads. Participants master the operationalization of Service Level Indicators SLIs, Service Level Objectives SLOs, automated deployment pipelines, and telemetry systems tailored for complex predictive pipelines. This course is delivered by Agile Leaders Training Center.

Who Should Attend

- Machine learning engineers seeking to integrate rigorous reliability frameworks into operational environments.
- Site reliability engineers tasked with maintaining infrastructure, availability, and error budgets for machine learning workloads.
- MLOps engineers building telemetry, packaging systems, and continuous delivery pipelines for intelligent applications.
- Data engineers and software developers responsible for building reliable feature stores, serving endpoints, and inference pipelines.
- DevOps specialists and AI product managers overseeing operational risk, uptime, and governance across production machine learning systems.

Departments and Industries

This course supports technical teams maintaining critical predictive systems across data-intensive sectors.

- Data Science and AI Units in Financial Services
- Engineering and DevOps Teams in Telecommunications
- IT Operations and Infrastructure Groups in Technology and Cloud Services
- Quality Assurance and Risk Divisions in Healthcare and Biotechnology
- ML Governance and Compliance Units in Government and the Public Sector
- Product Engineering Groups in E-Commerce and Digital Retail

Learning Objectives

By the end of this course, participants will be able to:

- Design resilient machine learning architectures grounded in core SRE operational principles.
- Construct robust model deployment pipelines using canary releases, shadow deployments, and automated rollbacks.
- Define, measure, and enforce ML-specific SLIs and SLOs across training and serving infrastructure.
- Implement telemetry and ML observability tools to identify data skew, feature drift, and latency bottlenecks.
- Formulate structured incident response playbooks to accelerate post-mortem root cause analysis for model failures.
- Enforce governance, reproducibility, and ethical safeguards across enterprise feature stores and data workflows.

Course Agenda

Day 1: Foundations of Reliable ML Systems

- Deconstructing the Machine Learning Lifecycle and Inherent Operational Vulnerabilities
- Adapting Site Reliability Engineering Tenets to Predictive System Architectures
- Data Intake Integrity: Managing Collection, Annotation, and Pipeline Ingestion Risks
- Architecting Resilient Pipeline Orchestration for Distributed Model Training
- Systematic Analysis of Failure Modes and Silent Faults in Operational Workflows
- Balancing Mathematical Complexity against System Maintainability Trade-Offs
- Operational Analysis of Workflow Feedback Loops and the YarnIt Case Study

Day 2: Data Management and Governance in ML

- Data Durability Architectures, Lineage Versioning, and Cryptographic Access Controls
- Feature Store Topology: Metadata Schema Design and Ingestion Latency Management
- Security Policies, Data Privacy Protection, and Model Fairness Considerations
- Audit Documentation Frameworks for Human Annotation Consistency and Quality Control
- Aligning Enterprise Regulatory Compliance Policies with Automated Pipeline Execution
- Systematic Triage of Data-Driven Production Failures and Pipeline Edge Cases
- Governance Retrospective: Mitigating Architectural Debt and Upstream Data Rot

Day 3: Model Validation, Observability, and Monitoring

- Formulating Validation Thresholds for Pre-Production Quality and Statistical Efficacy
- Offline Evaluation Protocols: Distribution Shift, Performance Baselines, and Slicing
- Online Validation Mechanics: Controlled A/B Experiments and Shadow Inference Traffic
- Telemetry Implementation: Telemetry Instrumentation and ML Observability Tools
- Defining SLIs and Enforcing Error Budgets for Inference Latency and Model Health
- Automated Drift Detection: Monitoring Covariate Shift, Label Skew, and Decay
- Observability Architecture: Dashboard Telemetry Configuration and Alert Thresholds

Day 4: Scalable Deployment and Incident Response

- Model Serving Topology: Designing High-Throughput Batch, Streaming, and Edge Systems
- Progressive Rollout Strategies: Blue/Green Swaps, Canary Pipelines, and Fast Rollbacks
- Autoscaling Mechanics, Inference Cache Invalidation, and High-Availability Failover
- Developing and Operationalizing Machine Learning Incident Response Playbooks
- Post-Mortem Frameworks: Blameless Root Cause Analysis for Algorithmic Failures
- Operational Governance: Mitigating Algorithmic Bias and Enforcing Ownership Boundaries
- Live Simulation: Outage Mitigation, Circuit Breaking, and Model Resilience Drills

Day 5: Organizational Integration and MLOps Best Practices

- Team Topologies: Defining Machine Learning Systems Engineering Roles and Interfaces
- Enterprise Operational Paradigms: Centralized MLOps Platforms vs. Embedded Squads
- Continuous Training Loops: Triggering Real-Time Pipeline Re-Execution and Validation
- Lifecycle Ownership: Auditing Frameworks, Compliance Guardrails, and Ethics Controls
- Applied Engineering Case Studies: NLP Inference Load Testing and Ad Click Latency
- Systematic Enterprise Auditing and Regulatory Compliance Verification Protocols
- Architecture Presentation: Capstone Technical Defense and Reliability Assessment

Practical Exercises

Participants apply technical concepts through hands-on reliability engineering exercises.

- Suggested activity: Establish SLIs, SLOs, and an error budget model for a high-volume inference service.
- Suggested activity: Configure Prometheus and Grafana dashboards to alert on data drift and distribution divergence.
- Suggested activity: Build an incident response playbook addressing sudden inference degradation and serving failure.
- Suggested activity: Formulate an automated rollback policy for a canary deployment experiencing feature skew.

FAQs

What specific qualifications or prerequisites are needed for participants before enrolling in the course?

Participants should have a foundational understanding of machine learning concepts, familiarity with software engineering or DevOps methodologies, and working knowledge of cloud environments or machine learning frameworks.

How long is each day's session, and is there a total number of hours required for the entire course?

Each day consists of 4 to 5 hours of instruction, structured discussions, and practical exercises, totalling 20 to 25 hours over the five-day duration.

What is the difference between monitoring ML models and traditional software systems?

Conventional software monitoring tracks infrastructure metrics such as CPU saturation, network throughput, and process uptime. Machine learning monitoring must additionally measure data distribution shifts, feature drift, prediction entropy, label latency, and degradation in mathematical performance that occurs without service crashes.

Conclusion

Scaling machine learning requires replacing manual handoffs with disciplined engineering practices. By embedding SRE methodologies, observability metrics, structured incident response, and continuous deployment workflows into production platforms, cross-functional teams ensure their machine learning systems deliver measurable business reliability and long-term operational resilience.