

Updated on 11/08/2025

Sign up

Cadence training

3 days (21 hours)

Overview

Cadence is an open-source workflow orchestration engine designed to create stateful, resilient and scalable applications without the complexity of crashes or restarts.

Our Cadence training course teaches you how to model business processes in code (workflows & activities), interact in real time with Signals and Queries, and operate the platform (schemas, visibility, observability, multi-cluster).

You'll use the CLI, Go/Java SDKs and Cadence Web UI to debug, monitor and restart executions, while integrating Cadence with APIs and web services (asynchronous jobs, sagas, microservices orchestration).

At the end of the course, you'll be able to deploy a Cadence environment, build robust orchestrations and industrialize operations via Helm and CI/CD.

Like all our training courses, this one is based on the latest stable version [v1.3.3 of Cadence](#).

Objectives

- Design deterministic workflows and activities
- Master Signals, Queries, retries and timeouts
- Use Cadence Web for debugging and observability
- Operate the platform (schemas, visibility, multi-clusters)
- Securing, scaling and integrating into web APIs

Target audience

- Backend developers
- DevOps / SRE
- Tech Lead
- Distributed systems architects

Prerequisites

- Basic knowledge of Go or Java, Docker
- Knowledge of HTTP/API, Git, CI/CD
- Knowledge of microservices and messaging

Cadence training program

Cadence basics

- Understanding Cadence Workflow: code-driven orchestration and persistent state
- Typical use cases (sagas, batch processing, business processes)
- Architecture: frontend, history, matching, workers
- Domains/Namespaces, Task Lists and Shards
- Overview of SDKs (Go, Java) and CLI
- Workshop: starting Cadence locally with Docker and running a first workflow

Modeling workflows and activities

- The difference between workflow and activity, and writing rules
- Determinism and best practices
- Timers, sleep, child workflows
- Retry Policies, Backoff, timeouts
- Signals & Queries for runtime interaction
- Workshop: implementing a workflow + 2 activities with retries and timers

Data management and idempotency

- Activity idempotency and side-effect management
- Workflow versioning
- Controlled and mutable side effects
- Context management (cancellation, deadlines)
- Application vs. system retry strategies
- Workshop: seamless workflow evolution in production

Operating Cadence on the server side

- Server installation (Cassandra/MySQL/PostgreSQL, Kafka/Elastic)
- Schemas and upgrades; canary and bench
- Visibility and retention settings
- Multi-cluster / multi-region configuration
- Observability: metrics, logs, alerts
- Workshop: complete dev environment & visibility enabled

Cadence Web UI and troubleshooting

- Tour of Cadence Web v4: spaces, filters, multi-clusters
- Explore history and execution tree
- Relaunch, terminate, cancel, restart
- Read events and diagnose errors / timeouts
- Good debugging and troubleshooting practices
- Workshop: diagnose a faulty execution and correct it

Integrating Cadence with Web services

- Starting workflows from a web service/API
- Request/async?response patterns with Signals & Queries
- RBAC management / admin endpoints
- Integrations: queue, webhook, jobs, batch
- Error handling and HTTP mapping
- Workshop: exposing a Cadence-orchestrated web API

Resilience, performance and costs

- Worker scaling and task lists
- Rate limiting, heartbeats and sticky workers
- DLQ & recovery strategies
- Throughput/latency optimization
- Estimating infra costs & SLA
- Workshop: benchmarking and tuning a worker under load

Security and compliance

- AuthN/AuthZ, reverse proxy, mTLS
- Namespace partitioning, retention policies
- Encryption at rest / in transit
- Traceability: audits, visibility, logs
- Good governance practices
- Workshop: securing UI/CLI access and tracing operations

CI/CD, deployment and operation

- Helm Charts, Docker images, DEV? promotion PROD

- Unit testing & workflow integration
- CI/CD pipelines for servers, workers and schemas
- Incident response and rollback playbooks
- Release checklist and runbook
- Workshop: CI pipeline deploying a worker & schema migration

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced IT technology, or to acquire specific business knowledge or modern methods.

Positioning at training start

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical training: 60% hands-on, 40% theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Certification

A certificate will be awarded to each trainee who has completed the entire course.