

GTL-202: GitLab CI and CD Essentials

Course Description: This training introduces participants to the basic concepts of CI/CD and provides a deep dive introduction of the architecture of GitLab CI and CD as well as helping them to gain practical experiences at creating GitLab CI/CD pipelines.

Structure: 50% theory 50% hands on lab exercises.

Target audience: developers, system administrators and devops professionals who want to learn the concepts and usage of GitLab CI/CD in cloud-native environments and want to be able to create pipelines with it.

Prerequisites: Linux sys.admin and basic Git knowledge, experience in software development, basic knowledge of Kubernetes

Duration: 2 days

Detailed Course Outline:

Day 1: CI part

Module 1: The big picture

- Motivation behind CI/CD/CD
- Structure of a typical pipeline
- Introducing GitOps (focusing on Kubernetes)

Module 2: CI

- CI system generations
- GitLab CI:
 - Types of runners
 - Exercise: install your own runner
 - The structure of gitlab-ci.yml
 - Exercise: build a basic pipeline
 - Environment variables
 - Caching
 - Triggers
 - Exercise: trigger a job after merge request
 - Dependencies
 - Using GitLab registry
 - Exercise: build and push a Docker image from CI
 - Preview environments
 - Exercise: create a preview environment from the CI

Module 3: Basics of testing the test pyramid

- Unit tests
- Module/service tests
- Integration tests
- UI tests
- Performance tests

Day 2: CD part

Module 4: An overview on Continuous Delivery and Continuous Deployment

- Tools and what part they cover
- Delivery: Image and package creation
- Deployment:
 - Trigger
 - Install
 - Rollback
- FluxCD, Argo CD etc.
- GitOps for Kubernetes

Module 5: Helm (as a packaging format for K8s)

- Motivation
- Architecture
- State management
- Charts
- Templating Engine
- Flow control
- Testing
- Packaging
- Handling dependencies

Module 6: Flux CD

- Architecture
- Installing Flux CD on Kubernetes
- Creating a new repository
- Using an existing repository
- Configuring automated deployments
- Using Helm based deployments

Module 7: Outlook: Release management

- Strategies:
 - Replace
 - Rolling update
 - Blue/green
 - Canary
- Complex release management tools:
 - Flagger
 - Argo Rollouts