

GitLab CI/CD

Doelgroep Cursus GitLab CI/CD

De cursus GitLab CI/CD is bestemd voor DevOps-engineers, Software Developers en QA engineers die pipelining met GitLab willen leren.

Voorkennis Cursus GitLab CI/CD

Voor deelname aan cursus is basiskennis Git, versiebeheer en software workflows vereist. Bekendheid met containers is nuttig.

Uitvoering Training GitLab CI/CD

Theorie en praktijk worden in de cursus met elkaar afgewisseld. Uitleg aan de hand van case studies uit de praktijk.

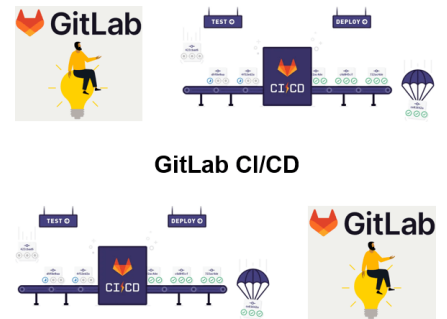
Certificaat GitLab CI/CD

De deelnemers krijgen na het goed doorlopen van de cursus een certificaat van deelname aan GitLab CI/CD.

Duur: 2 dagen

Prijs: € 1499

Open Rooster



Inhoud Cursus GitLab CI/CD

In de cursus GitLab CI/CD leer je hoe je GitLab pipelines kunt ontwerpen, implementeren en trouble shooting om builds, tests en deployments te automatiseren. De cursus behandelt het schrijven van pipelines, automatiseren van builds, uitvoeren van deployments, integratie van Docker en het toepassen van best practices voor performance en security.

Introductie tot GitLab CI/CD

Deze module biedt een introductie tot de basisprincipes van CI/CD en de rol van GitLab in het automatiseren van softwarelevering. Deelnemers leren over GitLab Runners, belangrijke componenten zoals jobs, stages, artifacts en caching. Ook worden gedeelde versus specifieke runners en best practices voor een betrouwbare CI/CD-opzet behandeld.

GitLab Pipelines Schrijven

Deelnemers leren hoe ze efficiënte GitLab pipelines kunnen schrijven en structureren met behulp van het .gitlab-ci.yml bestand. Onderwerpen zijn het definiëren van jobs en stages, instellen van afhankelijkheden, werken met variabelen en het uitvoeren van jobs parallel of sequentieel. Er wordt ook aandacht besteed aan security scans en het onderhoud van pipelines.

Automatiseren van Builds

In deze module staat het automatiseren van build processen binnen GitLab CI/CD centraal. Deelnemers leren hoe ze unit tests, integratietests en linting kunnen uitvoeren. Er is aandacht voor het afhandelen van fouten, instellen van notificaties, integreren van codekwaliteitstools en statische analyse, en automatisering van builds en tests in de pipeline.

Pipelines Troubleshooten

Deelnemers leren technieken voor het diagnosticeren en oplossen van pipeline problemen. Onderwerpen zijn het interpreteren van GitLab logs en artifacts, veelvoorkomende CI/CD-fouten, retry-strategieën, voorwaardelijke uitvoering, gebruik van job tokens en het debuggen van kapotte pipelines.

Deployments met GitLab

Deze module behandelt het uitvoeren van deployments via GitLab omgevingen en deployment jobs. Er wordt aandacht besteed aan strategieën zoals blue-green en canary deployments, toevoegen van security checks en het veilig beheren van secrets en omgevingsvariabelen voor een efficiënte deployment.

Docker in GitLab CI/CD

De cursus sluit af met het inzetten van Docker in GitLab CI/CD pipelines. Deelnemers leren hoe ze Docker-images kunnen bouwen, pushen en deployen naar Kubernetes en cloud omgevingen, prestaties optimaliseren via caching, werken met distributed runners, en Docker-containers naadloos integreren in GitLab pipelines.

Modules Cursus GitLab CI/CD

Module 1: Intro to GitLab CI/CD	Module 2: Writing GitLab Pipelines	Module 3: Automating Builds
What is CI/CD? Overview of GitLab Continuous Integration Continuous Deployment GitLab Runners Shared vs. Specific GitLab Jobs Key GitLab Components Stages Artifacts Caching	.gitlab-ci.yml Structure Defining Jobs Defining Stages Dependencies Parallel Job Execution Sequential Execution Using Variables Maintainable pipelines Multi-stage CI/CD Pipeline Implementing Security Scans Dependency Scanning	Automating Builds Running Unit Tests Integration tests Linting in CI/CD Handling Failures Notifications Tools Integration Code Quality Tools Static Analysis Tools Build Automation in Pipeline Test Automation in Pipeline
Module 4: Troubleshooting Pipelines	Module 5: Deployments with GitLab	Module 6: Docker in GitLab CI/CD
Understanding GitLab Logs Understanding Artifacts Common CI/CD errors Fixing Errors CI/CD Job Tokens Retry Strategies Conditional Execution Debugging Broken Pipeline	Deployment Strategies Using environments Deployment jobs Blue-Green Deployments Canary Deployments Adding Security Checks Managing Secrets Managing Environment Variables	Deploying to Docker Deploying to Kubernetes Cloud Providers Pushing GitLab Container Registry Docker Images in Pipelines Jobs inside Docker Containers Optimizing Performance with Caching Distributed Runners