

Continuous Delivery

Doelgroep Cursus Continuous Delivery

De cursus Continuous Delivery is bestemd voor developers, testers en administrators die betrokken zijn bij software ontwikkeling en die continuous delivery willen implementeren.

Voorkennis Cursus Continuous Delivery

Algemene kennis en vertrouwdheid met software ontwikkeling, programmeren, testen en installeren is vereist om aan de cursus Continuous Delivery te kunnen deelnemen.

Uitvoering Cursus Continuous Delivery

De theorie wordt behandeld aan de hand van presentaties. De concepten worden toegelicht met demo's. De theorie wordt afgewisseld met oefeningen. De cursustijden zijn van 9.30 tot 16.30.

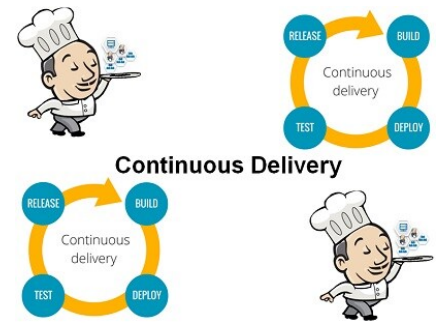
Certificaat Cursus Continuous Delivery

De deelnemers krijgen na het goed doorlopen van de cursus een officieel certificaat Continuous Delivery.

Duur: 2 dagen

Prijs: € 1499

[Open Rooster](#)



Inhoud Cursus Continuous Delivery

In de cursus Continuous Delivery leren de deelnemers hoe een continuous delivery proces voor het automatisch testen en deployen van software applicaties kan worden ingericht met [Docker](#) en [Jenkins](#). Door middel van Continuous Delivery kan de kwaliteit van software worden verbeterd en de doorlooptijd worden verkort.

Intro Continuous Delivery

In de cursus Continuous Delivery worden in de eerste plaats de principes van Continuous Delivery uitgelegd. In een Continuous Delivery proces wordt software frequent in korte cycles gereleased, getest en via een geautomatiseerde deployment pipeline in gebruik gesteld.

Docker Containers

Vervolgens wordt ingegaan op het gebruik van Docker Containers voor het snel in de lucht brengen van een ingerichte omgeving. De werking, architectuur en configuratie van Docker Containers komt uitgebreid aan de orde.

Jenkins Essentials

Ook is er aandacht voor de tool Jenkins voor het automatiseren van het bouwen van software. Hierbij komt het opzetten van een Jenkins Continuous Integration Pipeline aan de orde en wordt ingegaan op de inhoud en structuur van het Jenkinsfile.

Acceptance Testing

Ook het automatiseren van Acceptance Tests met Docker en Cucumber is onderdeel van het cursus programma. En is er aandacht voor de verschillende omgevingen in een Continuous Delivery proces en de tests die daarin worden gedaan.

Configuration Management

Vervolgens komen Applicatie en Infrastructuur Configuratie en het gebruik van de configuratie taal Ansible met Playbooks, Handlers en Variables aan bod.

Advanced Topics

Tenslotte wordt nog ingegaan op een aantal geavanceerde Continuous Delivery aspecten zoals het omgaan met veranderingen in de Database, het paralleliseren van pipelines en het gebruik van shared libraries.

Modules Cursus Continuous Delivery

Module 1 : Intro Continuous Delivery	Module 2 : Docker Essentials	Module 3 : Docker Applications
What is Continuous Delivery? Traditional Delivery Process Shortcomings Traditional Delivery Benefits of Continuous Delivery Fast Delivery and Feedback Cycle Low Risk Releases Automated Deployment Pipeline Continuous Integration Automated Acceptance Testing Configuration Management DevOps Culture	Virtualization and Containerization Disadvantages of Virtualization Benefits of Containers Isolation and Portability Installing Docker Docker Architecture Docker Components Docker Client Docker Server Docker Daemon Docker REST API	Building Docker Images Docker Commit Dockerfile Environment Variables Running Docker Containers Docker Container States Docker Networking Container Networks Exposing Container Ports Automatic Port Assignment Using Docker Volumes
Module 4 : Jenkins Essentials	Module 5 : CI Pipeline	Module 6 : Acceptance Testing
What is Jenkins? Extensibility by Plugins Jenkins Installation Installing on Docker Jenkins Pipeline Master and Slaves Vertical and Horizontal Scaling Test and Production Instances Configuring Agents Jenkins Swarm Agents Custom Jenkins Images	What is a Pipeline? Multi Stage Application Sections, Directives and Steps Commit Pipeline Pushing to GitHub Compile Stage Unit Test Stage Jenkinsfile Code Coverage and CheckStyle Scheduled Builds Development Workflows	Acceptance Testing Intro? Docker Registry Artifact Repository Docker Hub Private Docker Registry Domain Certificates Building Images Pushing and Pulling Images Acceptance Test in Pipeline Acceptance Testing Stage Running Acceptance Tests
Module 7 : Configuration Management	Module 8 : CI Pipeline	Module 9 : Docker Swarm
Application Configuration Infrastructure Configuration Automation and Version Control Configuration Languages Chef, Puppet and Ansible Agent Based Using Ansible Creating Inventory Playbooks Handlers and Variables Deployment with Ansible Working with Redis Ansible and Docker	Types of Environment Production Environment Staging Environment Test Environment Development Environment Non Functional Testing Performance Testing Load and Stress Testing Scalability Testing Security Testing Non Functional Challenges Application Versioning Complete Jenkins File	Server Clustering Docker Swarm Intro Setting up a Swarm Adding Worker Nodes Deploying a Service Publishing Ports Rolling Updates Draining Nodes Multiple Manager Nodes Scheduling Strategy Docker Stack Specifying docker-compose.yml Kubernetes
Module 10 : Advanced Continuous Delivery		
Managing Database Changes Understanding Schema Updates Database Migrations Using Flyway Configuring Flyway SQL Migration Script Backwards Compatibility Changes Non-Backwards Compatibility Changes Adding and Dropping Columns Changing Code Merging Data Avoiding Shared Database Parallelizing Pipelines Shared Libraries		