



Enterprise Computing Solutions - Education Services

TRAINING OFFERING

Sie erreichen uns hier

Arrow ECS Internet Security AG, Richtistrasse 11, CH-8304 Wallisellen

Email: trainings.ecs.ch@arrow.com
Phone: +41 43 222 80 00



Designing and Implementing Microsoft DevOps Solutions

CODE:

MCS_AZ-400T00

LENGTH:

40 Hours (5 days)

PRICE:

CHF2,800.00

Description

Dieser Kurs vermittelt die Kenntnisse und Fähigkeiten zur Implementierung, Umsetzung von DevOps-Prozessen und Praktiken. Entwerfen Sie dabei Release-Strategien und richten Sie Release-Management-Workflows ein. Ausserdem erwerben Sie Fähigkeiten zur Implementierung von Dependency Management Strategien und zur Bereitstellung von Anwendungsinfrastruktur in DevOps-Pipelines mit Rücksicht auf Compliance und Sicherheitsanforderungen.

Des Weiteren lernen Sie in diesem Seminar wie Sie kontinuierliches Feedback mithilfe von System-Feedback-Mechanismen implementieren können. Schliesslich können Sie nach diesem Training eine DevOps-Strategie entwerfen inkl. der Planung einer Transformation und des Projektes sowie die Erstellung von effizienten Teamstrukturen.

Nach Abschluss dieses Seminars haben die Teilnehmer Wissen zu folgenden Themen:

- Getting started with Source Control
- Scaling git for enterprise DevOps
- Implement & Manage Build Infrastructure
- Managing application config & secrets
- Implement a mobile DevOps strategy
- Implementing Continuous Integration in an Azure DevOps Pipeline
- Managing Code Quality and Security Policies
- Implementing a Container Build Strategy
- Design a Release Strategy
- Set up a Release Management Workflow
- Implement an appropriate deployment pattern
- Designing a Dependency Management Strategy
- Manage security and compliance
- Infrastructure and Configuration Azure Tools
- Azure Deployment Models and Services
- Create and Manage Kubernetes Service Infrastructure
- Third Party and Open Source Tools available with Azure
- Implement Compliance and Security in your Infrastructure
- Recommend and design system feedback mechanisms
- Implement process for routing system feedback to development teams
- Optimize feedback mechanisms
- Planning for DevOps
- Planning for Quality and Security
- Migrating and Consolidating Artifacts and Tools

Dieses Seminar basiert auf der originalen **Microsoft MOC Unterlage AZ-400** und bereitet auf die dementsprechende Prüfung vor.

Mögliche Zertifizierung: Microsoft Azure DevOps Engineer

Audience

Dieses Seminar richtet sich an:

- Teilnehmer, die DevOps Projekte und Prozesse planen und implementieren möchten
- Teilnehmer, die eine Anwendungsinfrastruktur implementieren und diese managen sowie konfigurieren möchten
- Teilnehmer, die an kontinuierlichen DevOps Integrationsprozessen interessiert sind
- Teilnehmer, die an der Zertifizierungsprüfung für Microsoft Azure DevOps Engineer interessiert sind

Prerequisites

Für dieses Seminar werden folgende Vorkenntnisse empfohlen:

- Fundierte Kenntnisse über Azure, Erfahrung mit virtuellen Maschinen und Containern und einige Erfahrungen mit Automatisierung und Skripting
- Grundlegende Kenntnisse über Versionskontrolle, Agile Softwareentwicklung und grundlegende Prinzipien der Softwareentwicklung
- Erfahrung in einer Softwareentwicklungs- oder Betriebsumgebung mit Windows oder Linux wäre hilfreich, ist aber nicht zwingend erforderlich
- Die Teilnehmer sollten auch Kenntnisse über allgemeine Prozesse der Anwendungsentwicklung und -bereitstellung haben
- Ausserdem wird empfohlen, dass Sie über Erfahrung in der Arbeit mit einer IDE sowie über einige Kenntnisse zum Azure-Portal verfügen. Teilnehmer, die keinen technischen Hintergrund in diesen Technologien haben, aber neugierig auf DevOps-Praktiken als Kulturwandel sind, sollten jedoch in der Lage sein, die prozeduralen und expositorischen Erklärungen der kontinuierlichen Integration unabhängig davon zu verfolgen.
- In diesem Seminar erlernen Sie notwendige Vorkenntnisse für das DevOps Seminar: Developing Solutions für Microsoft Azure

Programme

Getting started with Source Control

- What is Source Control?
- Benefits of Source Control
- Types of source control systems
- Introduction to Azure Repos
- Migrating from TFVC to Git
- Authenticating to your Git Repos

Scaling git for enterprise DevOps

- How to structure your git repo? Mono Repo or Multi-Repo?
- Git Branching workflows
- Collaborating with Pull Requests
- Why care about GitHooks?
- Fostering Internal Open Source
- Git Version
- public projects
- Storing Large files in Git

Implement & Manage Build Infrastructure

- The concept of pipelines in DevOps
- Azure Pipelines
- Evaluate use of Hosted vs Private Agents
- Agent pools
- Pipelines & Concurrency
- Azure DevOps loves Open Source projects
- Azure Pipelines YAML vs Visual Designer
- Setup private agents
- Integrate Jenkins with Azure Pipelines
- Integration external source control with Azure Pipelines
- Analyse & Integrate Docker multi stage builds

Managing application config & secrets

- Demo: SQL Injection attack
- Implement secure & compliant development process
- Rethinking application config data
- Manage secrets, tokens & certificates
- Implement tools for managing security and compliance in a pipeline

Implement a mobile DevOps strategy

- Introduction to Visual Studio App Center
- Manage mobile target device sets and distribution groups
- Manage target UI test device sets
- Provision tester devices for deployment
- Provision tester devices for deployment

Implementing Continuous Integration in an Azure DevOps Pipeline

- Continuous Integration Overview
- Implementing a Build Strategy
- Lab: Enabling Continuous Integration with Azure Pipelines
- Lab: Creating a Jenkins Build Job and Triggering CI

Managing Code Quality and Security Policies

- Managing Code Quality
- Managing Security Policies
- Lab: Managing Technical Debt with Azure DevOps and SonarCloud
- Lab: Checking Vulnerabilities using WhiteSource Bolt and Azure DevOps

Implementing a Container Build Strategy

- Lab: Existing .NET Applications with Azure and Docker Images

Design a Release Strategy

- Introduction to Continuous Delivery
- Introduction to Continuous Delivery
- Release strategy recommendations
- Building a High Quality Release pipeline
- Choosing a deployment pattern
- Choosing the right release management tool
- Building a release strategy
- Differentiate between a release and a deployment
- Define the components of a release pipeline
- Explain things to consider when designing your release strategy
- Classify a release versus a release process, and outline how to control the quality of both
- Describe the principle of release gates and how to deal with release notes and documentation
- Explain deployment patterns, both in the traditional sense and in the modern sense
- Choose a release management tool

Set up a Release Management Workflow

- Introduction
- Create a Release Pipeline
- Provision and Configure Environments
- Manage And Modularize Tasks and Templates
- Integrate Secrets with the release pipeline
- Configure Automated Integration and Functional Test Automation
- Automate Inspection of Health
- Building a release management workflow
- Explain the terminology used in Azure DevOps and other Release Management Tooling
- Describe what a Build and Release task is, what it can do, and some available deployment tasks
- Classify an Agent, Agent Queue and Agent Pool
- Explain why you sometimes need multiple release jobs in one release pipeline
- Differentiate between multi-agent and multi-configuration release job
- Use release variables and stage variables in your release pipeline
- Deploy to an environment securely, using a service connection
- Embed testing in the pipeline
- List the different ways to inspect the health of your pipeline and release by using, alerts, service hooks and reports
- Create a release gate

Implement an appropriate deployment pattern

- Introduction into Deployment Patterns
- Implement Blue Green Deployment
- Implement Canary Release
- Implement Progressive Exposure Deployment
- Describe deployment patterns
- Implement Blue Green Deployment
- Implement Canary Release
- Implement Progressive Exposure Deployment

Hands-On Lab

- Lab: Microsoft 365 Tenant and Service Management
- Exercise 1: Set up a Microsoft 365 trial tenant
- Exercise 2: Managing Microsoft 365 users, groups, and administration
- Exercise 3: Configuring Rights Management and compliance
- Exercise 4: Monitor and troubleshoot Microsoft 365

Designing a Dependency Management Strategy

- Introduction
- Packaging dependencies
- Package management
- Implement versioning strategy
- Recommend artifact management tools and practices
- Abstract common packages to enable sharing and reuse
- Inspect codebase to identify code dependencies that can be converted to packages
- Identify and recommend standardized package types and versions across the solution
- Refactor existing build pipelines to implement version strategy that publishes packages
- Manage security and compliance

Manage security and compliance

- Introduction
- Package security
- Open source software
- Integrating license and vulnerability scans
- Inspect open source software packages for security and license compliance to align with corporate standards
- Configure build pipeline to access package security and license rating
- Configure secure access to package feeds

Infrastructure and Configuration Azure Tools

- Learning Objectives
- Infrastructure as Code and Configuration Management
- Create Azure Resources using ARM Templates
- Create Azure Resources using Azure CLI
- Create Azure Resources by using Azure PowerShell
- Additional Automation Tools
- Version Control
- Lab Deploy to Azure using ARM templates
- Module Review Questions

Azure Deployment Models and Services

- Learning Objectives
- Deployment Models and Options
- Azure Infrastructure-as-a-Service (IaaS) Services
- Azure Automation with DevOps
- Desired State Configuration (DSC)
- Azure Platform-as-a-Service (PaaS) services
- Azure Service Fabric
- Lab Azure Automation - IaaS or PaaS deployment
- Module Review Questions

Create and Manage Kubernetes Service Infrastructure

- Learning Objectives
- Azure Kubernetes Service
- Lab Deploy and Scale AKS Cluster
- Module Review Questions
- After completing this module, students will be able to:

Third Party and Open Source Tools available with Azure

- Learning Objectives
- Chef
- Puppet
- Ansible
- Cloud-Init
- Terraform
- Lab Provision and configure an App in Azure Using X
- Module Review Questions

Implement Compliance and Security in your Infrastructure

- Security and Compliance Principles with DevOps
- Azure Security Center
- Lab Integrate a scanning extension or tool in an AZ DevOps pipeline/security center
- Module Review Questions

Module 1: Recommend and design system feedback mechanisms

- The inner loop
- Continuous Experimentation midset
- Design practices to measure end-user satisfaction
- Design processes to capture and analyze user feedback from external sources
- Design process to automate application analytics

Implement process for routing system feedback to development teams

- Implement tools to track system usage, feature usage, and flow
- Implement routing for mobile application crash report data
- Develop monitoring and status dashboards
- Integrate and configure ticketing systems with development team's work management system

Optimize feedback mechanisms

- Site Reliability Engineering
- Analyse telemetry to establish a baseline
- Perform ongoing tuning to reduce meaningless or non-actionable alerts
- Analyze alerts to establish a baseline
- Blameless PostMortems and a Just Culture

Planning for DevOps

- Transformation Planning
- Project Selection
- Team Structures
- **Lab:** Agile Planning and Portfolio Management with Azure Boards

Planning for Quality and Security

- Planning a Quality Strategy
- Planning Secure Development
- **Lab:** Feature Flag Management with LaunchDarkly and AzureDevOps

Migrating and Consolidating Artifacts and Tools

- Migrating and Consolidating Artifacts
- Migrating and Integrating Source Control
- Lab: Integrating Azure Repos and Azure Pipelines with Eclipse

Further Information

- Durchführung in Kooperation mit Arrow ECS Österreich und ETC
- Für Produkte, die mit Fremdwährung berechnet werden, behalten wir uns das Recht vor, die Preise bei Währungsschwankungen anzupassen.

Session Dates

Auf Anfrage. Bitte [kontaktieren Sie uns](#)

Zusätzliche Information

[Diese Schulung ist auch als Vor-Ort-Schulung verfügbar. Bitte kontaktieren Sie uns, um mehr zu erfahren.](#)