Szkolenie: Microsoft
AZ-400T01-T07 Azure DevOps Engineer Expert (exam AZ-400)

<table>
<thead>
<tr>
<th>FORMA SZKOLENIA</th>
<th>MATERIAŁY SZKOLENIOWE</th>
<th>CENA</th>
<th>CZAS TRWANIA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stacjonarne</td>
<td>Cyfrowe</td>
<td>3600 PLN NETTO*</td>
<td>5 dni</td>
</tr>
<tr>
<td>Stacjonarne</td>
<td>Tablet CTAB</td>
<td>4000 PLN NETTO*</td>
<td>5 dni</td>
</tr>
</tbody>
</table>

* (+VAT zgodnie z obowiązującą stawką w dniu wystawienia faktury)

LOKALIZACJE

Kraków - ul. Tatarska 5, II piętro, godz. 9:00 - 16:00
Warszawa - ul. Bielska 17, godz. 9:00 - 16:00

DOSTĘPNE TERMINY

2020-02-10  | 5 dni  | Warszawa  (Promocja, Termin gwarantowany)
2020-03-02  | 5 dni  | Kraków
2020-04-20  | 5 dni  | Warszawa

Cel szkolenia:

Pięciodniowe szkolenie **AZ-400T01-T07 Azure DevOps Engineer Expert (exam AZ-400)** zawiera w sobie następujące moduły: **AZ-400T01 Implementing DevOps Development Processes**, **AZ-400T02 Implementing Continuous Integration**, **AZ-400T03 Implementing Continuous Delivery**, **AZ-400T04 Implementing Dependency Management**, **AZ-400T05 Implementing Application Infrastructure**, **AZ-400T06 Implementing Continuous Feedback**, **AZ-400T07 Designing a DevOps Strategy**.

Plan szkolenia:

- **AZ-400T01 Implementing DevOps Development Processes**
  - 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
- Git Branching workflows
- Collaborating with Pull Requests
- Why care about GitHooks?
- Fostering Internal Open Source
- Git Version
- Public projects
- 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 and Open Source projects
  - Azure Pipelines YAML vs Visual Designer
  - Setup private agents
  - Integrate Jenkins with Azure Pipelines
  - Integration external source control with Azure Pipelines
  - Analyze & Integrate Docker multi-stage builds

- Managing application config & secrets
  - Introduction to Security
  - 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 Mobile DevOps
  - 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
  - Create public and private distribution groups

- AZ-400T02 Implementing Continuous Integration
  - 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 Whi
- Implementing a Container Build Strategy
  - Implementing a Container Build Strategy
  - Lab: Existing .NET Applications with Azure and Docker
- AZ-400T03 Implementing Continuous Delivery
  - Design a Release Strategy
    - Introduction to Continuous Delivery
    - Release strategy recommendations
    - Building a High Quality Release pipeline
    - Choosing a deployment pattern
    - Choosing the right release management tool
    - Lab: Building a release strategy
  - Set up a Release Management Workflow
    - 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
    - Lab: Automating your infrastructure deployments in the Cloud with Terraform and Azure Pipelines
    - Lab: Setting up secrets in the pipeline with Azure Key vault
    - Lab: Setting up and Running Load Tests
    - Lab: Setting up and Running Functional Tests
    - Lab: Using Azure Monitor as release gate
    - Lab: Creating a Release Dashboard
  - Implement an appropriate deployment pattern
    - Introduction into Deployment Patterns
    - Implement Blue Green Deployment
    - Feature Toggles
- Canary Releases
- Dark Launching
- AB Testing
- Progressive Exposure Deployment
- Lab: Blue-Green Deployments
- Lab: Traffic Manager

- AZ-400T04 Implementing Dependency Management
  - Designing a Dependency Management Strategy
    - Introduction
    - Packaging dependencies
    - Package management
    - Implement a versioning strategy
    - Lab: Updating packages

- Manage security and compliance
  - Introduction
  - Package security
  - Open source software
  - Integrating license and vulnerability scans

- AZ-400T05 Implementing Application Infrastructure
  - 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

○ 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

○ Course Completion

○ Final Exam

○ AZ-400T06 Implementing Continuous Feedback

○ Recommend and design system feedback mechanisms

○ The inner loop

○ Continuous Experimentation mindset

○ Design practices to measure end-user satisfaction

○ Design processes to capture and analyze user feedback

○ Design process to automate application analytics Lab : Integration between Azure DevOps and Teams Lab : Feature Flags

○ 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
Optimize feedback mechanisms
  - Site Reliability Engineering
  - Analyze 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

AZ-400T07 Designing a DevOps Strategy
  - 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

Wymagania:

Fundamental knowledge about Azure, version control, Agile software development, and core software development principles. It would be helpful to have experience in an organization that delivers software.

Poziom trudności

Certyfikaty:

Uczestnicy kursu AZ-400T01-T07 Azure DevOps Engineer Expert (exam AZ-400) otrzymują certyfikat ukończenia autoryzowanego szkolenia Microsoft.

Prowadzący:
Microsoft Certified Trainer.

Informacje dodatkowe:

Zajęcia prowadzone są w języku polskim, materiały źródłowe oraz oprogramowanie są w języku angielskim.