training code: AZ-400 / ENG DL 4d / EN

Design and Implement Microsoft DevOps solutions

Authorized Microsoft Design and Implement Microsoft DevOps solutions AZ-400 Distance Learning training.

Target audience:

- administrator
- developer
- DevOps engineer

Training recipients

The training is addressed to people interested in improving their knowledge and skills from DevOps processes, as well as people interested in taking Microsoft Azure DevOps Solutions certification exam.

The course is recommended to:

- developers
- administrators participating in projects related to developing new applications.

The course includes such issues as :

- making use of source code control, scaling Git for enterprise, as well as implementing and managin infrastructure.
- implementing continuous integration in Azure DevOps stream, managing code quality and security rules, as well as container building strategy.
- designing release strategy, configuring workflow of version management and implementing appropriate implementation pattern.

- designing strategy of dependency management, as well as security and compliance management.
- implementing infrastructure of code and infrastructure, providing Azure infrastructure using popular automation tools, implementing application infrastructure using various Azure platform services, as well as implementation methods, integrating external tools to implement Azure platform such as Chef and Puppet to involve compliance and security in release stream.
- designing system feedback mechanisms, implementing the process of conveying system opinons to development teams and optimising feedback mechanisms.
- planning transformations, choosing the project and creating team structure, developing quality and security strategy, planning migration and consolidation of artefacts and source control

Benefits

Gaining knowledge and practical skills from Office 365.
Including:

- Advantages of using source control, migration from TFVC to Git, scaling Git for Enterprise DevOps, implementing and managing compilation infrfastructure, application configuration management and sensitive data storage,DevOps mobile strategy
- Why continuous integration matters, implementing continuous integration using Azure DevOps, Configuring compilation and available options, creating automated workflows, integrating other tools for compilation with Azure DevOps platform, developing processes of hybrid code quality and methods of its measurment, detecting suspicious code, integrating automated tests of code quality, report related to code range during testing, tools to measure technical debt, detecting problems with Open Source licenses, as well as other licensing issues, implementing the strategy of building containers
- Differences between release and implementation, defining release stream components, about the things which you have to pay attention to when we design release strategies, classifying releases compared to the proces of releasing and controlling their quality, dealing with comments on release and documentation both in traditional and contemporary meaning, selecting release management tools, terminology used in Azure DevOps and other tools for release, understanding what Build and Release is, classifying the agent, queue and pool of agents, the necessity of several release tasks in one release stream, differentiating between release tasks for many agents and many configurations, the use of changeable versions and stage variables in release stream, remote implementation in environment using service connection, embedding testing in preparation, various ways of verifying the state of stream and release using alerts, service hooks and reports, creating release gateway, implementation patterns, implementing Blue Green deployment, Canary Release deployment, implementing progressive exposition
- Tools and practices of managing components, generalising packages which enable sharing and reuse, verifying base of codes to identify code dependencies which might be converted to packages, Identifying and recommending standard types and versions of packages in the whole solution,

existing streams to build refactor to implement version strategy which publishes packages, security and compliance management, verifying open source software packages in terms of security and compliance with licenses to adjust tchem to corporate standards, configuing structuring stream to safely access package channels

- Using infrastructure and configuration as a code rule, implementing and managing infrastructure using Microsoft automation technology, such as ARM schemes, PowerShell and Azure command line interafce, implementation models and services available in Azure, implementing and configuring Managed Kubernetes cluster, placing and configuring infrastructure using tools and services of other companies using Azure platforms, such as Chef, Puppet, Ansible, SaltStack and Terraform, infrastructure strategy and configuration, as well as appropriate tool sets for release stream and implementing compliance and security in application infrastructure
- Project practices to measure end-user satisfaction, designng processes to intercept and analyse users' opinions from external sources, routing design for data of Client's application failure report, recommended monitoring tools and technologies, suggested tools to track the use of system and functions, configuring integration failure reports for Client's applications, developing monitoring and state panels, routing implementation for data of Client's application failure, implementing tools to track the use of the system, the use of functions and flow, integration and configuration of ticketing systems with the development team work management system, analysing alerts to determine baseline, analysing telemetry to determine baseline, reviewing current websites nad intercepting opinions on system failures, performing constant tuning to diminish irrelevant or unfit for operation alerts
- Planning transformation with shared goals and timelines, choosing the project and specifying project indicators, as well as KPI indicators, creating the team and effective organisational structure, designing project quality strategy, planning secure development practice and compliance rule, migration and artefact consolidation, migrating and integrating source control means

Training program

Learning Path 1: Implement development for enterprise DevOps
Module 1: Introduction to DevOps
Module 2: Plan Agile with GitHub Projects and Azure Boards
Module 3: Design and implement branch strategies and workflows
Module 4: Collaborate with pull requests in Azure Repos
Module 5: Explore Git hooks
Module 6: Plan foster inner source
Module 7: Manage and configure repositories
Module 8: Identify technical debt
Learning Path 2: Implement CI with Azure Pipelines and GitHub Actions

Module 1: Explore Azure Pipelines
Module 2: Manage Azure Pipeline agents and pools
Module 3: Describe pipelines and concurrency
Module 4: Design and implement a pipeline strategy
Module 5: Integrate with Azure Pipelines
Module 6: Introduction to GitHub Actions
Module 8: Learn continuous integration with GitHub Actions
Module 9: Design a container build strategy
Learning Path 3: Design and implement a release strategy
Module 1: Create a release pipeline
Module 2: Explore release recommendations
Module 3: Provision and test environments
Module 4: Manage and modularize tasks and templates
Module 5: Automate inspection of health
Learning Path 4: Implement a secure continuous deployment using Azure Pipelines
Module 1: Introduction to deployment patterns
Module 2: Implement blue-green deployment and feature toggles
Module 3: Implement canary releases and dark launching
Module 4: Implement A/B testing and progressive exposure deployment
Module 5: Integrate with identity management systems
Module 6: Manage application configuration data
Learning Path 5: Manage infrastructure as code using Azure and DSC
Module 1: Explore infrastructure as code and configuration management
Module 2: Create Azure resources using Azure Resource Manager templates
Module 3: Create Azure resources by using Azure CLI
Module 4: Explore Azure Automation with DevOps
Module 5: Implement Desired State Configuration (DSC)
Module 6: Implement Bicep
Learning Path 6: Implement security and validate code bases for compliance
Module 1: Introduction to Secure DevOps
Module 2: Implement open-source software
Module 3: Software Composition Analysis
Module 4: Security Monitoring and Governance
Learning Path 7: Design and implement a dependency management strategy
Module 1: Explore package dependencies
Module 2: Understand package management
Module 3: Migrate consolidate and secure artifacts
Module 4: Implement a versioning strategy
Module 5: Introduction to GitHub Packages
Learning Path 8: Implement continuous feedback
Module 1: Implement tools to track usage and flow

Module 2: Develop monitor and status dashboards
Module 3: Share knowledge within teams
Module 4: Design processes to automate application analytics
Module 5: Manage alerts, blameless retr

Expected preparation of the participant

- Fundamental knowledge about Azure platform, version control, rozwoju oprogramowania Agile software development and basic software development rules. Experience in organisations providing software is strongly recommended
- Experience in IDE environment as well as certain knowlege on Azure portal are advised
- An ability to use materials in English
- previous courses: AZ-104, AA_10961

To increase the comfort of work and training's effectiveness we suggest using an additional monitor. The lack of additional monitor does not exclude participation in the training, however, it significantly influences the comfort of work during classes.
You can find information and requirements of participation in Distance Learning trainings at:
https://www.altkomakademia.pl/en/how-it-works/about-courses/

Training Includes

* electronic handbook available at:
https://learn.microsoft.com/pl-pl/training/
* access to Altkom Akademia student portal
To increase the comfort of work and training's effectiveness we suggest using an additional monitor. The lack of additional monitor does not exclude participation in the training, however, it significantly influences the comfort of work during classes.
You can find information and requirements of participation in Distance Learning trainings at:
https://www.altkomakademia.pl/en/how-it-works/about-courses/

Language

- Training: English
- Materials: English

Examination method

The exam is on-line. You can enroll at: https://home.pearsonvue.com/Clients/Microsoft.aspx

Duration

4 days / 29 hours

Examination description

After the **AZ-400** course, you can take **Microsoft** certification **exams**:an Authorized Test Center,online being monitored by an offsite proctor. Details on the website:
https://docs.microsoft.com/en-us/learn/certifications/exams/az-400