

Designing and Implementing Microsoft DevOps solutions

Code:	AZ-400T00
Length:	4 days
URL:	View Online

This course provides the knowledge and skills to design and implement DevOps processes and practices. Students will learn how to plan for DevOps, use source control, scale Git for an enterprise, consolidate artifacts, design a dependency management strategy, manage secrets, implement continuous integration, implement a container build strategy, design a release strategy, set up a release management workflow, implement a deployment pattern, and optimize feedback mechanisms

Audience Profile

Students in this course are interested in designing and implementing DevOps processes or in passing the Microsoft Azure DevOps Solutions certification exam.

Skills Gained

- Plan for the transformation with shared goals and timelines
- Select a project and identify project metrics and Key Performance Indicators (KPI's)
- Create a team and agile organizational structure
- Design a tool integration strategy
- Design a license management strategy (e.g., Azure DevOps and GitHub users)
- Design a strategy for end-to-end traceability from work items to working software
- Design an authentication and access strategy
- Design a strategy for integrating on-premises and cloud resources
- Describe the benefits of using Source Control
- Describe Azure Repos and GitHub
- Migrate from TFVC to Git
- Manage code quality including technical debt SonarCloud, and other tooling solutions
- Build organizational knowledge on code quality
- Explain how to structure Git repos
- Describe Git branching workflows
- Leverage pull requests for collaboration and code reviews
- Leverage Git hooks for automation
- Use Git to foster inner source across the organization
- Explain the role of Azure Pipelines and its components
- Configure Agents for use in Azure Pipelines

- Explain why continuous integration matters
- Implement continuous integration using Azure Pipelines
- Define Site Reliability Engineering
- Design processes to measure end-user satisfaction and analyze user feedback
- Design processes to automate application analytics
- Manage alerts and reduce meaningless and non-actionable alerts
- Carry out blameless retrospectives and create a just culture
- Define an infrastructure and configuration strategy and appropriate toolset for a release pipeline and application infrastructure
- Implement compliance and security in your application infrastructure
- Describe the potential challenges with integrating open-source software
- Inspect open-source software packages for security and license compliance
- Manage organizational security and compliance policies
- Integrate license and vulnerability scans into build and deployment pipelines
- Configure build pipelines to access package security and license ratings

Prerequisites

Successful learners will have prior knowledge and understanding of:

- Cloud computing concepts, including an understanding of PaaS, SaaS, and IaaS implementations.
- Both Azure administration and Azure development with proven expertise in at least one of these areas.
- Version control, Agile software development, and core software development principles. It would be helpful to have experience in an organization that delivers software.
- If you are new to Azure and cloud computing, consider one of the following resources:
- Free online: Azure Fundamentals (<https://docs.microsoft.com/en-us/learn/paths/azure-fundamentals/>)
- Instructor-led course: AZ-900: Azure Fundamentals (<https://docs.microsoft.com/en-us/learn/certifications/courses/az-900t00>)
- If you are new to Azure Administration, consider taking:
- Free online: Prerequisites for Azure Administrators (<https://docs.microsoft.com/en-us/learn/paths/az-104-administrator-prerequisites/>)
- Instructor-led course: AZ-104: Microsoft Azure Administrator (<https://docs.microsoft.com/en-us/learn/certifications/courses/az-104t00>)
- If you are new to Azure Developer, consider taking:
- Free online: Create serverless applications (<https://docs.microsoft.com/en-us/learn/paths/create-serverless-applications/>)
- Instructor-led course: AZ-204: Developing Solutions for Microsoft Azure (<https://docs.microsoft.com/en-us/learn/certifications/courses/az-204t00>)

Course Details

Outline

- Introduction to DevOps
- Choose the right project
- Describe team structures
- Choose the DevOps tools
- Plan Agile with GitHub Projects and Azure Boards
- Introduction to source control
- Describe types of source control systems
- Work with Azure Repos and GitHub
- Structure your Git Repo
- Manage Git branches and workflows
- Collaborate with pull requests in Azure Repos
- Identify technical debt
- Explore Git hooks
- Plan foster inner source
- Manage Git repositories
- Explore Azure Pipelines
- Manage Azure Pipeline agents and pools
- Describe pipelines and concurrency
- Explore continuous integration
- Implement a pipeline strategy
- Integrate with Azure Pipelines
- Introduction to GitHub Actions
- Learn continuous integration with GitHub Actions
- Design a container build strategy
- Introduction to continuous delivery
- Create a release pipeline
- Explore release recommendations
- Provision and test environments
- Manage and modularize tasks and templates
- Automate inspection of health
- Introduction to deployment patterns
- Implement blue-green deployment and feature toggles
- Implement canary releases and dark launching
- Implement A/B testing and progressive exposure deployment
- Integrate with identity management systems
- Manage application configuration data
- Explore infrastructure as code and configuration management
- Create Azure resources using Azure Resource Manager templates
- Create Azure resources by using Azure CLI
- Explore Azure Automation with DevOps

- Implement Desired State Configuration (DSC)
 - Implement Bicep
 - Introduction to Secure DevOps
 - Implement open-source software
 - Software Composition Analysis
 - Static analyzers
 - OWASP and Dynamic Analyzers
 - Security Monitoring and Governance
 - Explore package dependencies
 - Understand package management
 - Migrate consolidating and secure artifacts
 - Implement a versioning strategy
 - Introduction to GitHub Packages
 - Implement tools to track usage and flow
 - Develop monitor and status dashboards
 - Share knowledge within teams
 - Design processes to automate application analytics
 - Manage alerts, blameless retrospectives and a just culture
-

Download Whitepaper: Accelerate Your Modernization Efforts with a Cloud-Native Strategy

Get Your Free Copy Now