

# AZ-400 Microsoft Azure DevOps Solutions Training

## *COURSE CONTENT*

### GET IN TOUCH

---



Multisoft Systems  
B - 125, Sector - 2, Noida



(+91) 9810-306-956



info@multisoftsystems.com



www.multisoftsystems.com

## About Multisoft

Train yourself with the best and develop valuable in-demand skills with Multisoft Systems. A leading certification training provider, Multisoft collaborates with top technologies to bring world-class one-on-one and certification trainings. With the goal to empower professionals and business across the globe, we offer more than 1500 training courses, which are delivered by Multisoft's global subject matter experts. We offer tailored corporate training; project Based Training, comprehensive learning solution with lifetime e-learning access, after training support and globally recognized training certificates.

## About Course

The AZ-400 Microsoft Azure DevOps Solutions Training offered by Multisoft Systems is a comprehensive program designed to equip professionals with the skills and knowledge required to excel in the field of DevOps within the Microsoft Azure ecosystem. This training program focuses on the key concepts, tools, and practices necessary for effective software development, delivery, and deployment using Azure DevOps Services.

## Module 1: 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 Lessons**

- ✓ 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

## **Module 2: Implementing Continuous Integration**

### **Implementing Continuous Integration in an Azure DevOps Pipeline**

- ✓ Continuous Integration Overview
- ✓ Implementing a Build Strategy

### **Managing Code Quality and Security Policies**

- ✓ Managing Code Quality
- ✓ Managing Security Policies

### **Implementing a Container Build Strategy**

- ✓ Implementing a Container Build Strategy

## Module 3: 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
- ✓ 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**

- ✓ 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

### **Implement an appropriate deployment pattern**

- ✓ Introduction into Deployment Patterns
- ✓ Implement Blue Green Deployment
- ✓ Feature Toggles
- ✓ Canary Releases
- ✓ Dark Launching
- ✓ AB Testing

- ✓ Progressive Exposure Deployment

## **Module 4: Implementing Dependency Management**

### **Designing a Dependency Management Strategy**

- ✓ Introduction
- ✓ Packaging dependencies
- ✓ Package management
- ✓ Implement a versioning strategy

### **Manage security and compliance**

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

## **Module 5: 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

## Module 6: 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

### **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

## Module 7: Designing a DevOps Strategy

### **Planning for DevOps**

- ✓ Transformation Planning
- ✓ Project Selection
- ✓ Team Structures

### **Planning for Quality and Security**

- ✓ Planning a Quality Strategy
- ✓ Planning Secure Development



## Migrating and Consolidating Artifacts and Tools

- ✓ Migrating and Consolidating Artifacts
- ✓ Migrating and Integrating Source Control