# Application Lifecycle Management Using Azure DevOps Server 2022

ALM2022 | 3 Days

This three-day comprehensive course provides students with the knowledge and skills to effectively use the Application Lifecycle Management (ALM) tools found in Visual Studio and Azure DevOps Server (fka Team Foundation Server) to plan, track, design, develop, test, and deliver business value in the form of working software. The course demonstrates to developers, testers, product owners, project managers, architects, testers, and release managers the value of the various features and capabilities found throughout Visual Studio and Azure DevOps Server.
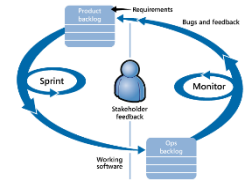
Course Objectives

At course completion, attendees will have had exposure to …

- ✓ Visual Studio and Azure DevOps ALM tools
- ✓ Editions, capabilities, and version compatibilities
- ✓ Azure DevOps Server vs. Azure DevOps Services
- ✓ Administrator types and related tools (Team Explorer)
- ✓ Planning/creating projects and collections
- ✓ Configuring services, security, teams, areas, iterations
- ✓ Introduction to *Azure Boards*
- ✓ Selecting an appropriate work item process
- ✓ Work item types and work item characteristics
- ✓ Using the agile tools (backlogs/boards) to visualize work
- ✓ Creating and refining a product backlog
- ✓ Tagging, finding, querying, and removing work items
- ✓ Using hierarchical (epics and features) backlogs
- ✓ Planning and tracking work in a sprint
- ✓ Introduction to *Azure Repos*
- ✓ Basic and advanced Git workflows
- ✓ Basic and advanced TFVC workflows
- ✓ Working with Azure Repos from Visual Studio
- ✓ Associating work items to commits for traceability
- ✓ Collaborating as a team and improving productivity
- ✓ Pairing, swarming, and mobbing patterns of work
- ✓ Creating and maintaining a wiki
- ✓ Performing code reviews using pull requests
- ✓ Requesting and capturing stakeholder feedback
- ✓ Using the Test and Feedback browser extension
- ✓ Using Visual Studio Live Share to collaborate in real time
- ✓ Writing and executing .NET unit tests
- ✓ Using IntelliTest to generate unit tests
- ✓ Using Live Unit Testing to run only impacted tests
- ✓ Using FxCop Analyzers and code metrics
- ✓ Using code clone analysis to find duplicate code
- ✓ Using IntelliTrace to troubleshoot and diagnose
- ✓ Using Performance Profiler to find problems in code
- ✓ Introduction to *Azure Test Plans*
- ✓ Test Case Management using test plans, suites, cases
- ✓ Testing web and desktop applications
- ✓ Capturing screenshots and video while testing
- ✓ Viewing and charting test run results
- ✓ Creating automated acceptance tests in Visual Studio
- ✓ Using Selenium and Appium for automated UI testing
- ✓ Using JMeter for load testing applications
- ✓ Practicing exploratory testing by taking testing tours
- ✓ Introduction to *Azure Pipelines*
- ✓ Creating and using build and release pipelines
- ✓ Using YAML to configure a build pipeline
- ✓ Using a multistage YAML pipeline for build and release
- ✓ Running automated tests in the pipeline
- ✓ Reviewing and understanding pipeline analytics
- ✓ Configuring on-premises agent for build/release
- ✓ Practicing Continuous Integration (CI) and Delivery (CD)
- ✓ Improving performance with Test Impact Analysis
- ✓ Agile metrics vs. traditional project metrics
- ✓ Configuring project alerts and notifications
- ✓ Using Excel for reporting and charting
- ✓ Using the Analytics Service and related widgets
- ✓ Using Power BI to query and visualize analytics data
- ✓ Using the REST API for reporting
- ✓ Relevant Visual Studio Marketplace extensions
- ✓ DevOps principles, challenges, and goals
- ✓ DevOps practices and related tools in Azure DevOps

Who Should Attend

This course is intended for current software development professionals who are involved in building applications with Visual Studio and Azure DevOps. Regardless of the student's role, he or she will be able to learn and get hands-on experience with all of the ALM features found in Visual Studio and Azure DevOps Server.

Prerequisites

Before attending this course, a student should have experience working on a software development team and be familiar with that team's development processes, practices, and tools. Additionally, students should:

- ✓ Have familiarity with agile practices and Scrum
- ✓ Be able to read and understand C# .NET code (all source code will be provided)
- ✓ Have used Visual Studio 2019, 2020, or 2022
- ✓ Be able to read and understand requirements
- ✓ Understand Microsoft Windows basics

Accentient™

Application Lifecycle Management Using Azure DevOps Server 2022

ALM2022 | 3 Days

Modules

Module 1: Introduction to ALM
- ✓ Application Lifecycle Management overview
- ✓ Visual Studio and Azure DevOps tools and features
- ✓ Azure DevOps Server vs. Azure DevOps Services
- ✓ Features and capabilities by edition and role

Module 2: Azure DevOps Projects
- ✓ The various administrator roles
- ✓ Project collections and projects
- ✓ Creating a project collection and project
- ✓ Configuring a project (areas, iterations, etc.)
- ✓ Configuring teams and team membership
- ✓ Securing a project

Module 3: Azure Boards
- ✓ Introduction to Azure Boards
- ✓ Selecting a work item process (e.g. Scrum)
- ✓ Creating a custom, inherited process
- ✓ Work item types, categories, and hierarchies
- ✓ Creating, tagging, finding, and managing work items
- ✓ Querying and charting work items
- ✓ Using the agile backlogs, boards, and task boards
- ✓ Using Excel to query and update work items
- ✓ Hierarchical backlogs (e.g. epics and features)

Module 4: Azure Repos
- ✓ Introduction to Azure Repos
- ✓ Git version control system overview
- ✓ Basic and advanced Git workflows
- ✓ TFVC version control system overview
- ✓ Basic and advanced TFVC workflows
- ✓ Working with Azure Repos from Visual Studio
- ✓ Associating work items to commits for traceability

Module 5: Collaborating as a Team
- ✓ Collaborating effectively as a team
- ✓ Improving team productivity
- ✓ Pairing, swarming, and mobbing patterns of work
- ✓ Creating and maintaining a wiki
- ✓ Using pull requests to perform code reviews
- ✓ Requesting and capturing stakeholder feedback
- ✓ Collaborating in real time with Visual Studio Live Share

Module 6: Writing Quality Code
- ✓ Writing and running unit tests
- ✓ Using Visual Studio Test Explorer
- ✓ Leveraging parameterized unit tests
- ✓ Measuring code coverage while testing
- ✓ Using IntelliTest to generate unit tests
- ✓ Using Live Unit Testing to run impacted tests
- ✓ Test-Driven Development (TDD) overview
- ✓ Code analysis, code metrics, and code clone analysis
- ✓ Using application profiling and IntelliTrace

Module 7: Azure Test Plans
- ✓ Introduction to Azure Test Plans
- ✓ Test case management (test plans, suites, cases)
- ✓ Manually testing web and desktop applications
- ✓ Automated acceptance testing in Visual Studio
- ✓ Testing through the UI using Selenium and Appium
- ✓ Load testing using JMeter
- ✓ Exploratory testing using Test & Feedback extension

Module 8: Azure Pipelines
- ✓ Introduction to Azure Pipelines
- ✓ Configuring and using build pipelines
- ✓ Running tests in the pipeline
- ✓ Practicing Continuous Integration (CI)
- ✓ Configuring and using release pipelines
- ✓ Practicing Continuous Delivery (CD)
- ✓ YAML pipelines

Module 9: Reporting
- ✓ Agile metrics vs. traditional metrics
- ✓ Configuring alerts and notifications
- ✓ Ad-hoc reporting/charting using Excel
- ✓ Using the Analytics Service
- ✓ Using Power BI to query analytics
- ✓ Querying data using the REST API

Module 10: Improving DevOps
- ✓ What is DevOps?
- ✓ Principles, challenges, and goals
- ✓ The Three Ways (flow, feedback, continual learning)
- ✓ Achieving Continuous Delivery (CD)

Course Designer

This course was designed by Richard Hundhausen, Microsoft's first Visual Studio ALM/DevOps MVP, Professional Scrum Trainer, co-creator of the Nexus Scaled Professional Scrum framework, and an experienced software developer. To see other developer courses, visit www.accentient.com.

Accentient™