

- Explore the DevOps background, approach, and best practices
- Integrate test automation with DevOps
- Implement continuous testing
- Learn how DevOps practices and principles improve software quality and efficiency
- Understand the differences between DevOps and traditional operational methodologies
- Discover the major steps required to successfully implement delivery pipelines

Organizations today are seeking ways to improve the efficiency of both their software development efforts and operations while still meeting quality objectives. Competitive pressures and customer demands continue to reduce software product release schedules, driving the pursuit of faster software releases, which in turn requires ever more efficient testing capabilities. DevOps is the combination of development, testing, and operations and includes continuous integration, automated testing, continuous delivery, and rapid deployment practices. Because DevOps practices require confidence that changes made to the code base will function as expected, automated testing is an essential ingredient that is integrated in the process in every step and relied upon for enforcement of quality gates and to ensure overall delivery quality. This course will teach you how to avoid the common mistakes of DevOps implementations and to leverage DevOps best practices:

- Test automation
- Automate everything
- Incremental build and delivery
- Continuous improvement
- Frequent code commits
- Infrastructure as code
- Fix the build(!) prioritization
- Repeatable, reliable processes
- Collaboration and communication
- Operations in DevOps

Upon completion of the course, students will be able to recognize positive and negative patterns of software build, test, and deployment in their organization that relate to DevOps. Key concepts that will be introduced and discussed include:

- Test strategy and implementation within a CI/CD context
- Automated quality gates
- Managing configuration
- Continuous integration and delivery
- Automated deployments
- Operations management of infrastructure and data
- Organizational impacts of DevOps implementation

Bring your specific issues and problems to the training course for discussion as well.

Who Should Attend

The audience includes software test professionals, operations engineers, software developers, project managers, and business owners.

Course Completion and Certification

Upon completion of this course the attendee will be certified by the International Consortium for Agile (ICAgile) and

awarded the ICAgile Professional (ICP-FDO) designation. The ICP-FDO is one of two Continuous Learning Certifications (CLCs) on the DevOps Track. *The ICAgile certification fee is included with your registration for your convenience.*

About the ICAgile

The International Consortium for Agile's goal is to foster thinking and learning around agile methods, skills, and tools. The ICAgile, working with experts and organizations across agile development specialties, has captured specific learning objectives for the different agile development paths and put them on the learning roadmap. For more information visit www.icagile.com [1].

No specific prerequisites are assumed; however, attendees are expected to have some experience with software builds, deployments, and automated testing.

Course Companions and Lab Extensions

Explore the four follow-on, companion courses gain hands-on experience with critical DevOps techniques:

- [Agile & DevOps Leadership Workshop](#) [2]
- [DevOps Test Integration Workshop: Automating your DevOps and Test Environment](#) [3]
- [Hands-on Docker and Kubernetes Workshop](#) [4]
- [Hands-on Chef Workshop](#) [5]
- [Implementing Pipeline as Code Using Jenkins](#) [6]

Course Outline

Introduction to DevOps

What is DevOps?

Business value and benefits of DevOps

DevOps vs traditional approaches

DevOps principles

Configuration Management

Source code control

Version management

Managing infrastructure and configuration

Managing data

Continuous Integration

CI culture

CI principles

CI best practices

Build automation

Build quality

Test Strategy, Integration, and Automation

Testing types

Testing integration

Automated testing

Continuous Delivery

CD introduction and definition

CD principles

CD best practices

Deployment pipelines

Deployment automation frameworks

Operations

Continuous monitoring

Managing infrastructure

Managing databases

Organizational DevOps

DevOps within an agile context

Cultural challenges

Addressing governance and policy requirements