



Continuous Delivery Ecosystem Foundation (CDEF)SM

Course Duration: 16 hours

OVERVIEW

This course is designed for participants who are engaged in the design, implementation, and management of DevOps deployment pipelines and toolchains that support Continuous Integration, Continuous Delivery, Continuous Testing and potentially Continuous Deployment. The course highlights underpinning processes, metrics, APIs and cultural considerations with Continuous Delivery.

Key benefits of Continuous Delivery will be covered including increased velocity to assist organizations to respond to market changes rapidly, thus being able to outmaneuver competition, reduce risk and lower costs while releasing higher quality solutions. Increased productivity and employee morale by having more activities performed by pipelines instead of humans so teams can focus on vision while pipelines do the execution.

Built upon the principles and practices highlighted in bestselling books such as “Continuous Delivery,” “Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations,” and more written by thought leaders in the DevOps movement. The Continuous Delivery Ecosystem Foundation course equips IT professionals with the broad-based competencies necessary in architecting and orchestrating effective and efficient automated deployment pipelines.

The course materials will include practical artifacts, templates, and lexicons collected by the author, Marc Hornbeek to assist learners post-class.

This certification positions learners to successfully complete the Continuous Delivery Ecosystem Foundation (CDEF) exam.

COURSE OBJECTIVES

The learning objectives for CDEF include a practical understanding of:

- Goals, history, terminology, and pipeline
- The importance, practices, and transformation of a DevOps collaborative culture
- Design practices, such as modular design and microservices
- Continuous Integration (CI), such as version control, builds, and remediation
- Tenets and best practices of Continuous Testing (CT)
- Continuous Delivery and Deployment (CD): packaging, containers, and release
- Continuous Monitoring (CM): monitoring and analysis infrastructure, process, and apps
- Infrastructure and tools: frameworks, tools, and infrastructure as code
- Security Assurance: DevSecOps
- The opportunity to hear and share real-life scenarios

AUDIENCE

The target audience for the Continuous Delivery Ecosystem Foundation course is anyone interested in learning about the principles of Continuous Integration and Continuous Delivery, such as:

- Build Engineers
- Enterprise Architects
- IT Managers
- Maintenance and Support Staff
- Operational and Infrastructure Teams
- Project Managers
- QA Managers
- Release Managers and Engineers
- Software Developers
- Security Professionals
- Testers

LEARNING MATERIALS

- Sixteen (16) hours of instructor-led training and exercise facilitation
- Digital Learner Manual (excellent post-class reference)
- Participation in exercises designed to apply concepts
- Sample documents, templates, tools and techniques
- Access to additional sources of information and communities

PREREQUISITES

An understanding and knowledge of common DevOps terminology and concepts and related work experience are recommended.

CERTIFICATION EXAM

Successfully passing (65%) the 60-minute examination, consisting of 40 multiple-choice questions, leads to the candidate's designation as a certified DevOps Continuous Delivery Ecosystem Foundation (CDEF). The certification is governed and maintained by the DevOps Institute.

COURSE OUTLINE

Course Introduction

- Course goals
- Course agenda

CDEF Concepts

- Continuous delivery (CD) definition
- Architecting for continuous delivery
- Continuous delivery and DevOps
- Relationships between CD, Waterfall, Agile, ITIL, and DevOps
- Benefits of continuous delivery

CDEF Culture

- Importance of culture to the CD Architect
- What a CD Architect can do about culture
- How to maintain culture
- Assignment: DevOps culture and practices to create flow

Design Practices for Continuous Delivery

- Why design is important to continuous delivery
- CD Architect's role in design
- Key design principles
- CD best practices
- Microservices and containers

Continuous Integration

- Continuous integration (CI) defined
- CD Architect's role in CI
- Importance of CI

- Benefits of CI
- CI best practices
- Assignment: Optimizing CI workflows

Continuous Testing

- Continuous testing (CT) defined
- Importance of CT
- Benefits of CT
- CD Architect's role in CT
- Five tenets of CT
- CT best practices
- Assignment: Handling environment inconsistencies

Continuous Delivery and Deployment

- Continuous delivery defined
- Continuous deployment defined
- Benefits of continuous delivery and deployment
- CD Architect's role in continuous delivery and deployment
- Continuous delivery and deployment best practices
- Assignment: Distinguishing continuous delivery and deployment

Continuous Monitoring

- Continuous monitoring defined
- Importance of continuous monitoring
- CD Architect's role in continuous monitoring
- Continuous monitoring best practices
- Assignment: Monitoring build progress

Infrastructure and Tools

- Importance of infrastructure and tools
- CD Architect's role in infrastructure and tools
- Building a DevOps toolchain
- Infrastructure/tools best practices
- Assignment: Identifying common infrastructure/tool components

Security Assurance

- Importance of security assurance
- DevSecOps and Rugged DevOps defined
- CD Architect's role in security
- Security best practices
- Assignment: Applying security practices

Capstone Exercise

- Identifying toolchain and workflow improvements

Summary

Additional Sources of Information

Exam Preparations

- Exam requirements
- Sample exam review