

# DevOps Training: DevOps Theory and Practice (DEVOPST, 4 jours)

---

## Description

The course DevOps Theory and Practice (DevOps Training) explores DevOps implementation through process and technology. The course begins with a discussion on the proper use of the agile scrum framework in developing software with a view to integrating software development with IT operations. Once the theory and practice of scrum has been addressed, we turn our focus to the design and implementation of value streams and the three ways of DevOps: fast left to right workflow, fast right to left feedback and a generative culture. The training includes a thorough discussion of business processes and practices in the creation of automated release and deployment pipelines, the containerization of applications and best practices to reduce lead time and application deployments. The course concludes with a discussion on telemetry and change management within the context of a DevOps transition.

## Tarifs

- Tarification: \$3,750/person
- Rabais de 10% lorsque vous inscrivez 3 personnes.

## Plan de cours

### DevOps: Laying the Foundation

---

The Nature of Projects

The Nature of Processes

Project Versus Process: Why the Difference Is Essential

How Projects and Processes Are Related

Agile Is About Values

The Scrum Framework

About Software Development and IT Operations

About Service Transition and Service Operations

And Then There Was DevOps

### Agile Scrum: Key Concepts

---

Why Traditional [Waterfall] Software Development Fails in the Modern World

Exploring Solutions to Common Software Development Pitfalls

And Then There Was Scrum: Agile Values and Principles

The Scrum Workflow: From Planning to Delivery

The Sprint: Planning, Execution, Review and Retrospective

How Requirements Are Gathered and Documented

The Key Roles: Product Owner, Developers and the Scrum Master

About the Product Backlog

### Agile Scrum: How It's Done

---

The Product Owner Is the Key: Organizing the Product Backlog

Managing Requirements in an Agile Organization

The Agile Nature of Requirements Gathering and Decomposition

About the Development Team, Sprint Planning and Sprint Execution

The Scrum Master as a Linchpin to Process Efficiency

About the Release Plan and Potentially Shippable Product Increments

Moving From Development to Operations: Service Transition

The Challenge of DevOps: Reducing Lead Time Through Process Efficiency

### DevOps Is About Value Streams

---

What Is a Value Stream?

What Is a Technology Value Stream?

DevOps: The Coexistence of Development and Operations

The Goal: Business Success and Customer Satisfaction

Understanding Lead Time and Process Time

The Three Ways of DevOps

The First: Fast Left to Right Workflow

The Second: Fast Right to Left Feedback

The Third: Creating a High Trust Culture

DevOps Considerations I – the Principle of Flow

---

What is the Principle of Flow?

How to Make Work Visible

Always Limit Work in Progress

Reducing Batch Size Is a Must

Reducing Handoffs Is an Even Bigger Must

How Bottlenecks Can Be Identified and Eliminated

The Fine Art of Eliminating Waste

DevOps Considerations II – the Principle of Feedback

---

The Core Principle: Seeing Problems as They Occur

Resolving Problems: Identify, Swarm and Solve

A Key Consideration: The Convergence of Quality and the Problem Source

What About Quality Controls?

The Importance designing feedback of Designing for IT Operations Downstream

DevOps Considerations III – the Culture

---

Continual Learning and Experimentation Is a Must

About Failure and the Safety Culture

Understanding Your Organization's Type

Improvement and Quality Are Everyone's Responsibility

Communicating Knowledge Throughout the Organization

Designing Feedback and Communication Systems for Resiliency

The DevOps Practice

---

How to Structure the Organization

How to Design the Value Stream

Coming to Agreement on a Shared Goal

Keeping the Planning Horizon Short Is a Must

Managing Technical Debt Is Also Must

The Deployment Pipeline

---

About On-Demand Creation

How to Achieve On-Demand Creation

Making Use of a Single Repository

Making Use of Continuous Build, Test and Integration

Automated Validation Test Suites

Ensuring Early Detection Through Testing

Continuous Integration and Deployment

---

Implementing a Branch Architecture

Implementing Small Batch Deployment Through Commit and Merge

Designing the Deployment Pipeline

Automating the Deployment Process and Pipeline

Making Use of Self-Service Deployment

Differentiating Between Deployment and Release

Release Patterns

---

Environment Versus Application Based Release Patterns

The Blue-Green Deployment Pattern

About Database Changes

The Canary and Cluster Immune Release Patterns

The Power of the Feature Toggles

Deconstructing Enterprise Applications: The Logical Tiers

How Logical Tiers Are Assigned to Physical Machines

Understanding Microservices Architectures and the Importance of Loose Coupling

Making Use of the Strangler Application Pattern

About Telemetry, Instrumentation and Change

---

About Event and Information Management

Creating a Centralized Log Management Infrastructure

Ensuring That Log Information Serves Both Development and Operations

Making Use of Peer-Reviewed and Pairs Programming

Making Use of Proactive Change Management

Putting It All Together