**Document Generated: 09/12/2025**
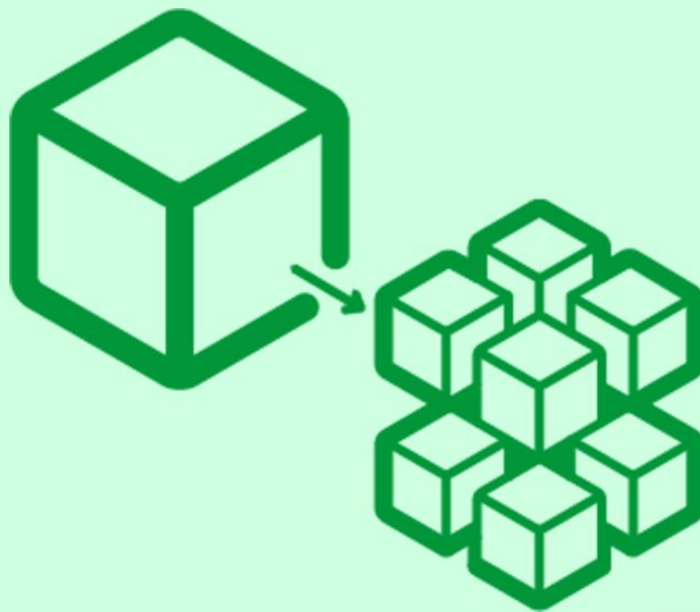
**Learning Style: Virtual Classroom**

**Technology:**

**Difficulty: Beginner**

**Course Duration: 2 Days**

# Understanding MicroServices: A Technical Overview (TT7050)

## About this course:

Understanding Microservices is a two-day, technical overview services training course geared for managers and project stakeholders who need to understand the impact of microservices, what is different about a microservice orientation and the overall process of adoption and implementation. This course focuses on what separates an ad hoc set of services from a managed, vibrant, reusable catalog of microservices. It provides an overview of the entire spectrum from the promise of shattering monolithic computing to the grit of exchanging data. The course presents a clear portrait of how a microservice orientation can fundamentally change the dynamics of how software is developed and "lives" within an enterprise.

Organizations are struggling with how to translate the microservice/DevOps/Continuous Deployment vision into practical terms and concepts. Understanding Microservices: A Technical Overview is designed for enterprise managers and stakeholders who wish to understand what microservices and a microservice-orientation are and how implementing microservices will impact their projects.

While our training is "technology centric", it is vendor independent, meaning that the content is not clouded by sales and marketing messages. Although a specific product can be discussed, the comprehensive lessons are geared towards teaching the practical application of microservices, rather than focusing on the finer points of the tools in use. Of course, microservices is not a technology but a set of practices and concepts that hold the promise of more closely aligning business with IT applications.

The average salary of a Java Developer is **$90,992** per year.

## Course Objective:

Microservices continues the evolution of systems and applications being decomposed into smaller components that work together to accomplish larger business tasks and services. Microservices are single purpose, autonomous, and independently deployable.

This course provides a solid understanding of what microservices bring to the table and how to facilitate the implementation, deployment, and management of microservices.

Students are taken on an in-depth tour of the basic microservices concepts as well as how they relate to SOA and other architectural styles.  They will examine the challenges associated with microservices as well as how to identify and design them.   The course then moves into designing and building REST-based services and working those services using various components.  The course addresses additional challenges such as scaling, monitoring, securing, and containerizing microservices.

Working in a dynamic, interactive discussion and demo environment, led by our experienced SOA professional, students will be able to:

- Explain the issues associated large, monolithic applications and how microservices present an opportunity to address many of those issues.
- Understand the relationships between microservices and SOA, SOAP services, and other recent initiatives.
- Implement RESTful microservices that take advantage of containers and the declarative nature of assembling simple components into executable entities.
- Work with implemented microservices to package, deploy, and manage them.
- Understand and work with the Cloud to scale microservices
- Containerize microservices using Docker
- Recognize and use best practices relative to designing and working with microservices.

## Audience:

This an overview level services training course, designed for people who need to understand and manage existing or upcoming microservice projects. Experience with managing and working with enterprise applications will be helpful. We will explore the terminology, the specification, the processes and technologies specific to microservices.

## Prerequisite:

Attendees should have a minimum of 2 years working knowledge in the IT industry. A basic understanding of software development and web-based applications is necessary. Actual development working knowledge is helpful but not necessary.

## Course Outline:

### Module 1: Microservices Overview

### Lesson: Monolithic Versus Micro

- Monolithic Applications
- Deploying and Scaling Monolithic Applications
- Microservices Described
- Deploying Microservices
- Scaling Microservices
- Benefits of Microservices
- Implementing Microservices

### Lesson: Microservices: The Business Proposition

- Dealing with Change

- Leveraging business processes
- Challenges to adoption
- The Service Analysis and Design Process
- Relating Microservices to Agile Practices
- DevOps Defined
- Relating Microservices to DevOps
- Continuous Deployment Defined
- Relating Microservices to Continuous Deployment

## Lesson: Designing Microservices

- Microservice Boundaries
- Size of Deployable Unit
- Communication Patterns
- Microservice endpoints
- Data Stores and Transaction Boundaries
- Challenges with Microservices

## Module 2: RESTful Microservices in Spring

## Lesson:  Overview of REST

- REpresentational State Transfer
- REST Characteristics
- REST Elements
- REST Architectural Principles
- REST and HTTP
- REST/HTTP: Representation-Oriented
- REST Design Principles

## Lesson:  Designing RESTful Microservices

- Effectively Designing RESTful Services
- Best Practices for Endpoint Definition
- Using Query Parameters
- Working with HTTP GET and DELETE
- Working with HTTP PUT
- Working with HTTP POST
- Best Practices for HTTP Methods
- Handling Additional Operations

## Module 3: The Microservices Ecosystem

## Lesson: Working with Microservices

- Typical Microservices Stack
- Monitoring Microservices
- Logging

- Containerizing with Docker
- Deploying into Docker
- Orchestration of Microservices

## Lesson: Microservice Best Practices

- Motivation and Mindset
- Minimum Viable Product
- Challenges of Data and Data Islands
- Spring Data and Microservices
- PrePersist, PreUpdate, and Repository Interface
- A DevOps-Style Microservice Life Cycle
- Continuous Delivery Pipeline
- Governance
- Tracking APIs and API Consumers

## Lesson: Microservice Patterns

- Aggregator Pattern
- Branch Pattern
- Proxy Pattern
- Chained Pattern
- Circuit Breaker/Bulkhead Isolation Pattern
- Continuous Integration/Delivery Pattern
- Shared Resources as an Anti-Pattern
- Shared Resources as a Pattern
- Async Messaging Pattern

## Lesson: Microservice Anti-Patterns and Challenges

- Microservice Costs
- When to Apply and NOT Apply
- Data Islands
- Dependency management
- Cohesion Creep
- Avoiding Versioning

## Credly Badge:

### Display your Completion Badge And Get The Recognition You Deserve.

Add a completion and readiness badge to your Linkedin profile, Facebook page, or Twitter account to validate your professional and technical expertise. With badges issued and validated by Credly, you can:

- Let anyone verify your completion and

achievement by clicking on the badge

- Display your hard work and validate your expertise
- Display each badge's details about specific skills you developed.

Badges are issued by QuickStart and verified through Credly.

Find Out More or See List Of Badges