



**DIGILEAF INC.**

*Leading Excellence Among Fellows*

## Course Outline

### Enterprise Architecture Learning Track

## Service-Oriented Architecture with Java

Today's IT organizations invariably employ disparate systems and technologies. Creating applications that leverage different technologies has historically been a daunting task. Service-oriented Architecture (SOA) provides a clear solution to application integration issues by allowing systems to expose their functionality via standardized, interoperable interfaces. SOA offers several key advantages such as enabling applications to adapt to changing technologies, easily integrate applications with other systems, leverage existing investments in legacy applications, Quickly and easily create a business process from existing services. Service-oriented architecture (SOA) is popular because it lets you reuse applications and it promises interoperability between heterogeneous applications and technologies.

This course explains and demonstrates how to build and implement service-oriented solutions by combining established Service-Oriented Architecture (SOA) principles, patterns, and practices with modern Java services technology. This course focuses on implementing SOA using Web Services with the adoption of new, simpler protocols such as REST.

### Training Objectives

1. Implement web services with various degrees of complexity and flexibility using Java.
2. Describe the major web service platforms available in the Java world, namely JAX WS 2.0, Apache Axis, Spring, and XFire.
3. Apply the concepts of SOA Core, the business layer, in creating SOA applications, with the help of case studies and real-world examples.
4. Use Java Web Services to build effective SOA applications.

### Topics

#### Part 1 – The Mantra of SOA

- I. Architecture
- II. Application Architecture
- III. Enterprise Computing or Architecture
- IV. Loose Coupling
- V. Reusability
- VI. Seamless Integration
- VII. Return on Investment (ROI)

#### Part 2 – Web Services and SOA

- I. The SOA Approach
- II. XML-Advantages and Disadvantages
- III. Introduction to Web Services, RESTful Services, and Other Transport with XML
- IV. RPC and Document Based-WS: How to Communicate, Pros and Cons of the Two Approach
- V. Why We Should Use Doc-WS?
- VI. Implementations: JAX-WS 2, Axis2, Spring-WS, and XFire/CXF 2.0

#### Part 3 – Web Services Implementations

- I. Web Service Using JAX-WS 2.0
- II. Web Service Using Apache Axis
- III. Web Service Using Spring
- IV. Web Service Using Xfire

#### Part 4 – Data and Services

- I. JDO
- II. Data Services
- III. Service Data Objects
- IV. Service Components Architecture
- V. Message-Oriented Middleware
- VI. Enterprise Service Bus

**Duration** 4 day(s)