

## Essentials of Software Architecture

An introductory course on Software Architecture

**Delivery:** On-site, Open Enrollment

**Duration:** 1 day

**Approach:** Lecture, discussions, workshops.

**Level:** Introductory

### Audience

This is an introductory course suitable for all team members including:

- Developers
  - Testers
- Project Leads
- Customer Representatives
  - Coaches

### Recommended Pre-requisites

None

### Recommended Experience

No specific experience is required, but the following is useful for getting full value from the discussions and exercises:

- Some experience of participating in software development projects
- Some involvement with software architectures

### Related Courses

For students interested in continued learning in related fields, we suggest:

- Use-Case Modeling
- Managing Iterative Software Development
  - Component-Based Development with the UML
  - Use-Case Driven Testing

## Workshop Description

This 1-day class is intended to provide software development project team members with the skills that they need to manage the technical risks facing a software development project by establishing and evolving an appropriate software architecture.

It focuses on introducing the practices and principles that underlie good software architectures and their successful communication to project stakeholders and team members. It provides practical advice on how organizations and projects can go about shaping, describing, evolving and testing an appropriate software architecture.

This is a highly interactive course that makes extensive use of discussions and exercises to simulate the decision making processes and activities that the team will need to engage in to shape and evolve the architecture on a real software development project.

## Objectives

Upon completion of the course, participants will understand:

- What we mean by software architecture
- The principles underlying good software architectures
- What resources, capabilities and skills we will need
- How to identify the requirements that shape the architecture
- What it means to shape the software architecture
- How to effectively manage technical risks
- How to document and communicate the architecture
- The role of architecture patterns and design patterns
- How to implement and verify the architecture
- How architecture evolves through the lifecycle

## Topics Covered

- Principles of Software Architecture
- Identifying Architectural Requirements
- Shaping the System
- Evolving the Architecture Implementation
- Testing the Architecture
- Architecture through the Lifecycle
- Review and Getting Started

### Europe

+44 (0)20 7025 8070

[info-eur@ivarjacobson.com](mailto:info-eur@ivarjacobson.com)

### Americas

978-649-2856

[info-usa@ivarjacobson.com](mailto:info-usa@ivarjacobson.com)

### Australia

1300 567 280

[Info-aus@ivarjacobson.com](mailto:Info-aus@ivarjacobson.com)

### Asia

+8610 82486030

[info-asia@ivarjacobson.com](mailto:info-asia@ivarjacobson.com)