

# UML Training: System Engineering & Application Architecture (SYSENG, 2 jours)

---

## Description

The course System Engineering & Application Architecture (UML Training) is an exploration of the challenges, tasks, roles & responsibilities of implementing a formal systems engineering process. The training begins with a comparison of waterfall & iterative methodologies. This is followed by a thorough discussion of the tasks and challenges of a formal system engineering process implementation. Every aspect of the System Engineering process is discussed including High Level Design, Requirements, Design Specifications, UML, topology design & more.

## Tarifs

- Tarification: \$1,500/person
- Rabais de 10% lorsque vous inscrivez 3 personnes.

## Plan de cours

### Essential Concepts

---

About Functional and Project Organizations

Introducing Waterfall Methods: Overview, Advantages and Disadvantages

Introducing the Iterative Method: The Need for Small Steps

Understanding the Rational Unified Process

Understanding Agile Methods

Clarifying Agile Methods: What is and is not Agile?

The Challenges of Applying an Iterative Method

An Overview of Change Management

### Iterative Methods

---

Exploring the Iterative Method from Inception to Deployment

The Rational Unified Process from Beginning to End

Exploring Inception

Exploring Elaboration

Exploring Construction

Exploring Transition

A Closer Look at Agile Methods

### Smooth Beginnings: Inception

---

How to Start? Comparing Waterfall and Iterative Methods

The Role of the Business Case

The Importance of Requirements: Business and System

Understanding Requirements in an Iterative Method: How Detailed are They?

Creating the Iteration Plan

Estimating Costs and Timelines in an Iterative Method

Creating a High Level Architecture

Common Mistakes when Migrating from a Waterfall Method

Roles and Responsibilities

### The Systems Engineering Process

---

Process Overview: Goals and Objectives

Inputs, Outputs and Expected Outcomes

Challenges and Risks

High Level Design

Detailed Design  
Build and Verify  
Implementation  
Life Cycle Management

#### High Level Design

---

Reviewing Inputs, Outputs and Activities  
Producing the System Engineering Plan  
Producing the System Requirements Specification (SRS)  
Producing the System Design Specification (SDS)

#### Writing the System Design Specification (SDS)

---

Introduction to the SDS  
About the SDS Structure  
System Wide Decisions  
Identity, Access Management and Directory Services Strategy  
Service Continuity Strategy  
System Management Strategy

#### Writing the System Requirements Specification (SRS)

---

The Structure of the SRS  
About Technical Requirements  
Managing Scope and Technical Details  
How much Details is too much Detail?  
Best Practices and Techniques  
Common Mistakes in Writing Technical Specifications  
Mapping Functional Requirements to Technical Requirements  
Using Appropriate Language: Editing and Rewriting Technical Requirements  
Managing Technical Requirements  
About Acceptance Criteria and Test Plans

#### Build and Verify and Requirements

---

Testing Fundamentals  
Tracing Requirements back to their Origins  
Finalizing Essential Documents  
Updating Related Documents  
Wrapping things Up

#### Finalizing the Design

---

Build and Verify  
Final Design Review (FDR  
Implementation  
In-Service Support and Life Cycle Management  
Review