Architectural Design Patterns

Audience Course Architectural Design Patterns

The course Architectural Design Patterns is aimed at experienced developers and software architects who want to apply design patterns for the architecture of systems.

Prerequisites Course Architectural Design Patterns

Knowledge of an object-oriented programming language like C++, C# or Java and experience with object oriented analysis and design with UML is desired.

Realization Training Architectural Design Patterns

The course Architectural Design Patterns has a hands-on character. The theory is covered on the basis of presentation slides and is illustrated with demos of architectural patterns. There are exercises in design problems where architectural patterns can be applied.
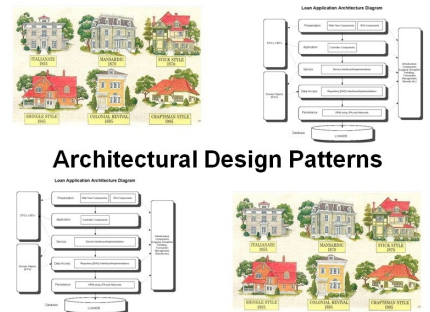
Certificate Architectural Design Patterns

Attendants receive a certificate of participation in the course Architectural Design Patterns after successful completion of the course.

Duration: 2 days   **Price: € 1399**

Open Schedule

Architectural Design Patterns

Content Course Architectural Design Patterns

The course Architectural Design Patterns discusses the importance and principles of architectural modeling and the application of Architectural Design Patterns. Architectural Design patterns are concerned with the large-scale organization of systems and the cooperation of their components and layers and not to the internal architecture of the individual software components.

Intro Architectural Design

The course starts with a review of the basic principles of software architecture, software design patterns and the terminology used.

Architectural Patterns

Subsequently attention is paid to the role of design patterns in the architecture of software systems. The main categories of architectural patterns are discussed.

Call and Return Patterns

For example Call and Return Patterns such as the Layers pattern and the Client Server pattern are covered. The pros and cons are discussed.

Data Flow Patterns

Data Flow patterns are treated as well. Examples are the Pipes pattern as used in the Linux operating system and the Filter pattern that is known from the Java Servlet implementation. Shared Memory patterns such as the Blackboard pattern are also on the program of the course.

Distributed Systems Patterns

Distributed systems patterns such as the Proxy and Broker pattern are also discussed. Practical implementations of each pattern are reviewed and the pros and cons are explained.

Concurrency Patterns

Finally advanced patterns about concurrency and threads are covered and a number of patterns that do not belong to a particular category are treated. Participants will practice with the patterns and discuss the application of the patterns to practical problems in software architecture.

Modules Course Architectural Design Patterns

Module 1 : Software Architecture	Module 2 : Architectural Patterns	Module 3 : Call and Return Patterns
What is Architecture?	Architectural Modeling	Object Oriented Pattern
Software Architecture	Model Characteristics	OO Benefits and Drawbacks
Layered Architecture	Architectural Viewpoints	Object Oriented Architecture
Software Characteristics	Pattern Terminology	Layers Pattern
Analogy to Building	Gang of Four Design Patterns	Layers Problem
Role of Architect	Architectural Patterns	Layers Solution
Software Architecture Elements	Architectural Pattern Categories	Network Architecture
Architecture Context	Batch Sequential Pattern	Layers Benefits and Drawbacks
Architectural Viewpoints	Pipe and Filter Pattern	Layers Pattern Variant
Logical Architecture	Blackboard Pattern	Client Server Pattern
Non Functional Requirements	Publish and Subscribe Pattern	Client Server Architecture
Physical Architecture	Peer to Peer Pattern	Three Tier Pattern
Early Load Testing	Model View Controller Pattern	Three Tier Architecture
Module 4 : Data Flow Patterns	**Module 5 : Shared Memory Patterns**	**Module 6 : Distributed Systems**
Data Flow Architecture	Data Centered View	Proxy Pattern
Batch Sequential Pattern	Shared Repository Pattern	Types of Proxies
Data Flow Pattern Problems	Shared Repository Architecture	Copy on Write Proxy
Batch Sequential	Active Repository Pattern	Remote Proxy
Pipes and Filter Pattern	BlackBoard Pattern	RMI Proxy Architecture
Pipes and Filter Forces	BlackBoard Architecture	Broker Pattern
Pipes and Filter Patterns	BlackBoard Context	Broker Forces
Servlet Filters	Speech Recognition	Broker Solution
Web Service Handlers	BlackBoard Solution	Bridge Component
Call Chain of Handlers	BlackBoard Variants	Broker Variations
Benefits and Drawbacks	BlackBoard Know Uses	Benefits and Drawbacks
Pipes and Filter Variants	Benefits and Drawbacks	Broker Class Diagram
Module 7 : Interactive Systems	**Module 8 : Implicit Invocation**	**Module 9 : Concurrency Patterns**
MVC Pattern	Communication Patterns	Reactor Pattern
MVC Architecture	RPC Pattern	Server Socket Loop
MVC Model	Publish Subscribe Pattern	Reactor Pattern Forces
MVC View	Queue versus Publish Subscribe	Event Driven Design
MVC Controller	Topics and Queues	Single Threaded Reactor
Multiple Views Same Data	Data Distribution Pattern	Non Blocking IO
Known Uses MVC	Request Reply Pattern	Thread Pool Reactor
Benefits and Drawbacks	Request Reply Correlation	Reactor Known Uses
PAC Pattern	Multiple Replies	Reactor Benefits and Drawbacks
PAC Structure	Scalable Request Reply	Active Object Pattern
PAC Solution	Guaranteed Delivery Scenario	Active Object Method Scheduling
PAC Benefits and Drawbacks	Guaranteed Delivery	Active Object Method Dispatch

Module 10 : Other Patterns
MicroKernel Pattern
Microkernel Structure
State Machine Pattern
State Machine Structure
Reflection Pattern
Reflection Structure
Process Control Pattern
Process Control Structure
Master and Slave Pattern
Master and Slave Structure