

## Architectural Design Patterns

### Doelgroep Cursus Architectural Design Patterns

De cursus Architectural Design Patterns is bedoeld voor ervaren developers en software architecten die design patterns willen toepassen bij de architectuur van systemen.

### Voorkennis Cursus Architectural Design Patterns

Kennis van een object georiënteerde programmeertaal zoals C++, C# of Java en ervaring met object georiënteerde analyse en design met UML is gewenst.

### Uitvoering Training Architectural Design Patterns

De cursus Architectural Design Patterns heeft een hands-on karakter. De theorie wordt behandeld aan de hand van presentatie slides en wordt geïllustreerd met demo's van architectural patterns. Er zijn oefeningen met design problemen waar architectural patterns kunnen worden toegepast.

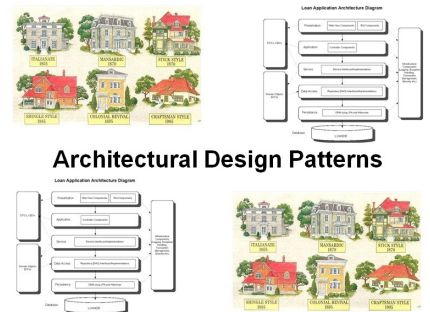
### Certificering Architectural Design Patterns

De deelnemers krijgen na het goed doorlopen van de cursus een certificaat van deelname aan de cursus Architectural Design Patterns.

Duur: 2 dagen

Prijs: € 1399

Open Rooster



## Inhoud Cursus Architectural Design Patterns

In de cursus Architectural Design Patterns wordt ingegaan op het belang en de principes van architectural modeling en de toepassing van Architectural Design Patterns. Architectural Design patterns hebben betrekking op de grootschalige organisatie van systemen en de samenwerking van hun componenten en lagen en niet op de interne architectuur van de afzonderlijke software componenten.

### Intro Architectural Design

De cursus gaat van start met een review van de basis principes van software architecture, software design patterns en de gebruikte terminologie.

### Architectural Patterns

Vervolgens wordt aandacht besteed aan de rol van design patterns in de architectuur van software systemen. De belangrijkste categorieën van architectural patterns worden daarbij besproken.

### Call and Return Patterns

Aandacht wordt bijvoorbeeld besteed aan Call en Return Patterns zoals het Layers pattern en het Client Server pattern. De voor en nadelen komen daarbij aan bod.

### Data Flow Patterns

Eveneens worden Data Flow patterns behandeld. Voorbeelden hiervan zijn het Pipes pattern zoals gebruikt wordt in het Linux operating system en het Filter pattern dat bekend is uit de Java Servlet implementatie. En ook Shared Memory patterns zoals het Blackboard pattern worden behandeld.

### Distributed Systems Patterns

Tevens komen distributed systems patterns zoals het Proxy en Broker pattern aan bod. Van ieder pattern passeren praktische implementaties de revue en worden de voor- en nadelen besproken.

### Concurrency Patterns

Tot slot worden geavanceerde patterns over concurrency en threads besproken en worden een aantal patterns die niet tot een bepaalde categorie horen belicht. De deelnemers zullen oefenen met de patterns en de toepassing van de patterns voor praktische problemen in de software architectuur bespreken.

## Modules Cursus Architectural Design Patterns

<b>Module 1 : Software Architecture</b>	<b>Module 2 : Architectural Patterns</b>	<b>Module 3 : Call and Return Patterns</b>
What is Architecture? Software Architecture Layered Architecture Software Characteristics Analogy to Building Role of Architect Software Architecture Elements Architecture Context Architectural Viewpoints Logical Architecture Non Functional Requirements Physical Architecture Early Load Testing	Architectural Modeling Model Characteristics Architectural Viewpoints Pattern Terminology Gang of Four Design Patterns Architectural Patterns Architectural Pattern Categories Batch Sequential Pattern Pipe and Filter Pattern Blackboard Pattern Publish and Subscribe Pattern Peer to Peer Pattern Model View Controller Pattern	Object Oriented Pattern OO Benefits and Drawbacks Object Oriented Architecture Layers Pattern Layers Problem Layers Solution Network Architecture Layers Benefits and Drawbacks Layers Pattern Variant Client Server Pattern Client Server Architecture Three Tier Pattern Three Tier Architecture
<b>Module 4 : Data Flow Patterns</b>	<b>Module 5 : Shared Memory Patterns</b>	<b>Module 6 : Distributed Systems</b>
Data Flow Architecture Batch Sequential Pattern Data Flow Pattern Problems Batch Sequential Pipes and Filter Pattern Pipes and Filter Forces Pipes and Filter Patterns Servlet Filters Web Service Handlers Call Chain of Handlers Benefits and Drawbacks Pipes and Filter Variants	Data Centered View Shared Repository Pattern Shared Repository Architecture Active Repository Pattern BlackBoard Pattern BlackBoard Architecture BlackBoard Context Speech Recognition BlackBoard Solution BlackBoard Variants BlackBoard Know Uses Benefits and Drawbacks	Proxy Pattern Types of Proxies Copy on Write Proxy Remote Proxy RMI Proxy Architecture Broker Pattern Broker Forces Broker Solution Bridge Component Broker Variations Benefits and Drawbacks Broker Class Diagram
<b>Module 7 : Interactive Systems</b>	<b>Module 8 : Implicit Invocation</b>	<b>Module 9 : Concurrency Patterns</b>
MVC Pattern MVC Architecture MVC Model MVC View MVC Controller Multiple Views Same Data Known Uses MVC Benefits and Drawbacks PAC Pattern PAC Structure PAC Solution PAC Benefits and Drawbacks	Communication Patterns RPC Pattern Publish Subscribe Pattern Queue versus Publish Subscribe Topics and Queues Data Distribution Pattern Request Reply Pattern Request Reply Correlation Multiple Replies Scalable Request Reply Guaranteed Delivery Scenario Guaranteed Delivery	Reactor Pattern Server Socket Loop Reactor Pattern Forces Event Driven Design Single Threaded Reactor Non Blocking IO Thread Pool Reactor Reactor Known Uses Reactor Benefits and Drawbacks Active Object Pattern Active Object Method Scheduling Active Object Method Dispatch
<b>Module 10 : Other Patterns</b>		
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		