



## Design Patterns

### Course Background:

The goal of Design Patterns is to create a common documentation approach to help software developers resolve recurring problems encountered in their daily work. Patterns help create a shared language for communicating knowledge and experience about these problems and their solutions. Formally codifying these solutions and their relationships let us successfully document for reuse. Software architects will better understand good architectures for robust software applications. Forming a common pattern language for complex concepts about architectures allows us to reason about them. This process will help software engineers to define the best possible strategy in solving complex architecture problems.

### Course Benefits:

After successful completion of the course, participants will:

- understand the architectural relevance of patterns
- have insight into the way patterns can be applied to comply to certain non-functional requirements.
- know of many of the available patterns;
- have some skills in recognizing situations where patterns can be applied;
- know decision criteria when to apply design patterns.

### Who will benefit from this course?

- Software Developers
- Software Engineers
- Software Architects

### Pre-requisites:

Software developers and software architects working in the area of design and implementation of object oriented systems, with working knowledge of C++, Java or C# and common UML class modelling techniques and some skills in creating object models.

### Course Contents:

Course duration: 4 days

Block 1 ( ¼ day):	The architectural relevance of design patterns
	Patterns and modularity
	Classification of patterns
Block 2 to 7 ( ½ day each):	All patterns from Gamma et al., "Design Patterns"
	These patterns are illustrated with examples and case studies.
	Examples
	Exercises
Block 8 (last ¾ day):	Patterns for real-time software (optional)
	Architecture patterns (Buschmann et al, "Pattern Oriented Software Architecture" )
	Layers, MVC, Broker
	Realizing architectures using design patterns
	Frameworks and patterns
	Conclusions

### Course Format:

Teaching method: Lectures, practical exercises and guided discussions  
 Teaching material: Copies of presentations, hand-outs of exercise solutions, text of cases and CD  
 Recommended book: "Design patterns: elements of reusable object oriented software" Gamma, et.al.  
 Language: English or Dutch  
 Instructor: This course is delivered by a certified "Advanced UML Professional"



During the course numerous exercises are given to the participants that are aimed at recognizing situations where patterns can be applied, identifying appropriate patterns and inserting patterns in the design. A CD is supplied with example programs related to the exercises and case studies.

### Additional Options:

Please indicate any desired options when signing up for this course, or contact your account manager.

### Terms and conditions:

The standard terms and conditions of Mithun Training & Consultancy are applicable. We will send you a copy on request.