

Essential design patterns in every project

Duration: 1 day

Audience: Professionnel TI

Who is this training for?

Designers, architects and people developing applications

Training objectives

- Describe the different categories of patterns
- Apply architectural best practices through proven solutions in your applications
- Identify the patterns that are essential for any project

Summary

Design patterns are known and proven object-oriented solutions for recurring design problems. They have the advantage of being independent of the programming language, ensure modularity of the system and allow ease of evolution. They represent a common vocabulary for designers' expertise and promote the transfer of knowledge between expert and novice designers.

Course outline

- Introduction to the Gamma Catalog: Terms and Concepts, Essentials and Space of Design Patterns
- Construction Patterns: Factory Method, Abstract Factory, Builder and Singleton
- Structuring Patterns: Adapter, Composite, Facade and Proxy
- Behavior Patterns: Command, Iterator, Observer, Strategy and Template Method
- Practical technique for selecting a pattern
- Steps to make a pattern

Approach and methodology

Structured and pragmatic approach focused on understanding and applying design patterns. Each pattern is presented using a generic UML model, independent of the language, followed by a concrete

example (Java or C#) with detailed explanations. An approach that allows for a clear, structured and directly applicable understanding of design patterns in real projects.

Prerequisites

- Good experience in object-oriented programming
- Proficient in object design concepts (e.g. SOLID)
- Knowledge of UML diagrams (class diagrams)
- Experience in an object-oriented language (Java, C#, etc.)

Recommendations

- Hands-on experience in software development in a team
- Familiarity with application architecture and design issues
- Interest in good design practices and code maintainability
- Ability to analyze design issues