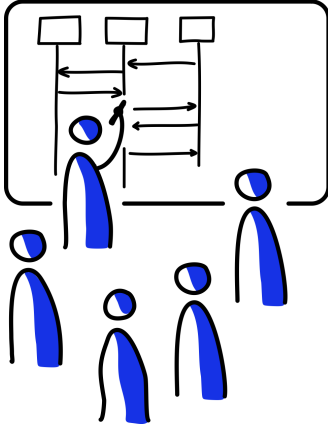


Technical Practices: Agile Architecture



What makes an architecture "agile"? As agility is the capability to react to change, when talking about architecture this means giving a structure to the software that is easy to evolve to future challenges. An agile architecture accepts the reality that structural changes in software are both unavoidable and unpredictable and, as most software products are maintained over a time span that is longer than the original development, reducing the cost of change of a software is therefore at the very core of the development of a good system.

In this two-day course we will discuss the ideas that help creating and sustaining a clean and easy to maintain architecture, starting from the basis of a good object orientation and continuing to the most recent development of these ideas.

The course is for software developers interested in learning or improving their knowledge of effective technical practices.

The training material is in English.

Your Take-Aways

- understanding the definition and fundamental concepts of Agile Software Architecture
- awareness of the importance to create and sustain a clean and easy to maintain code architecture
- knowledge of the most modern ideas that help reduce the cost of change
- having experienced live and practical exercises for the application and integration of the skills acquired in the class

Course Organisation

The course with a total contact time of 16 hours is delivered in presence or in interactive online mode. The course is split in various modules, none of which exceeds two hours, with short breaks as needed and sufficiently long breaks between the sessions. The actual times for breaks are agreed upon in the group at the beginning of the course.

Pre-course and post-course activities are part of the training and are presented via our interactive online learning platform in various formats (video, text, quizzes, worksheets, further reading, ...).

Course Agenda

- What constitutes a good architecture?
- An architecture screams Use Cases
- Centralised and De-centralised architectural decisions
- Architectural practices supporting agility
- Selected SOLID and GRASP principles
- Cohesion and Coupling
- Components
- Introduction to Hexagonal architecture
- Architecture and Design as a team modelling activity