

Object Oriented Analysis & Object Oriented Design by using UML & Patterns

Level: intermediate / advanced

Length: 15 – 20 hours

Course Objective: learn and exercise how UML and patterns are used in object oriented software development during its earliest stages – object oriented analysis (OOA) & design (OOD)

Course Audience: product / system owners, business analysts, software architects, software designers, programmers

What you will learn

- Introduction to Unified Process, common base for all the software development processes used in industry
- Introduction to the early stages of software development before writing the code, i.e. requirements analysis and design, what activities are performed, what artefacts are elaborated
- How UML (Unified Modeling Language) is used in software development
- How to identify the elements, assign responsibilities, define links between elements by using patterns GRASP (General Responsibility Assignment Software Patterns) and GoF (Gang of Four design patterns)
- How all these tools are used to propose a design which will be implemented
- How to apply the theory in a project

Prerequisites: the attendees have to be implied in software development, have to have notions related to requirements analysis, design, object oriented programming, the knowledge of a particular object oriented language (C++, C#, Java, Python, etc.) is a plus

Required facilities: the course doesn't depend on any particular tool. The instructor will use a free tool for drawing UML diagrams named draw.io. The attendee can use any tool they are familiar with for UML and OO programming language

Support materials: the attendees will get the supporting materials in electronic form, they will have access to all the diagrams and code demonstrated by the instructor and all the artefacts constructed by them as exercises

Description

The course provides a top down view on how a software development project is handled until implementation, before writing the code. This view implies the object oriented methodology technology which implies the using of several tools and languages.

The base for the modern processes used in software development is Unified Process (UP) which is introduced through its main characteristics.

The stages & activities defined by UP are demonstrated and exercised on a demo project starting with requirements, with focus on OOA & OOD.

The focus will be on:

- How requirements are reflected during the stages performed during software development, how they are traced in all the created artefacts
- What patterns are applied in requirements analysis, design and architecture
- How UML is used

For a vertical, deeper study of the implied elements there are several specialized trainings – UML, Design Patterns, OO programming languages, etc.

Contents

1. Introduction – base elements: software development process, UP, standard stages & activities with focus on analysis & design, UML, patterns types. Their roles and how they integrates
2. Requirements analysis – how to identify the domain vocabulary based on requirements; methodology, artefacts used by OOA (Object Oriented Analysis). Use of UML and patterns.
3. Design – how is constructed based on the previous stage; methodology, use of UML and patterns GRASP & GoF. Artefacts produced by OOD (Object Oriented Design).
4. The project – is demonstrated stage by stage emphasizing the use of UML and patterns in order to fulfill the requirements in code.