Unified Modeling Language

Level: intermediate

Length: 21 – 35 hours, depending on the practical part

Course objective: learn to use UML in software development

What You Will Learn

- Learn the role of UML, how UML modeling is used in Object Oriented analysis (OOA) and Object Oriented Design (OOD)
- Introduces the diagram types in order to express a software construct or behavior, or to document a software system
- Learn to translate the UML diagrams to code

Who Can Attend: programmers who want to apply UML during software development

Prerequisites: because UML is usually used in object oriented programming context programmers should know at least at a medium level a particular object oriented programming language like C++, Java, C#, Python, etc.

Required Facilities: VGA projector, white board, workstation, development tools for writing programs in an object oriented language (Java, C++, C#, etc.)

Related Courses: Design Patterns, Object Oriented Programming, object oriented programming languages (C++, Java, C#)

Description

The course offers a theoretical and practical approach of UML 2.x towards its usage by a programmer. The examples, case studies, hands on assignments offer a good understanding of UML diagrams. There are performed the following activities:

- Presentation of the main graphical elements, diagram types, their semantics and how they are used
- Understanding, „reading” the UML diagrams which were built by others
- Build of diagrams in order to express structures (static aspects) or behaviors (dynamic aspects)
- UML diagrams implementation („translation”) in one object oriented language (C++, Java, C#) in order to emphasize variants and particularities related to that language
- How the UML is used for modeling during software development, at requirements specifications, object oriented analysis (OOA) & object oriented design (OOD)

Even UML is not bound to a particular software development process, it is important the understanding of modeling particularities of every phase or activity type belonging to the software development. To illustrate this, the Rational Unified Process (RUP) is shortly presented; there are emphasized modeling perspectives, views on a software system.

The course is not based on any particular modeling tool (for example Rose) or any editing tool of the UML diagrams.

Contents: