

Unified Modeling Language 2

Nivel: mediu

Durata: 21 – 35 ore (funcție de ponderea părții practice)

Obiectivul cursului: participanții vor învăța modalități de folosire a UML în dezvoltarea de software

Ce veți învăța

- Se va exersa folosirea UML în activitatea de modelare specifice fazelor de specificare a cerințelor, analiza orientată pe obiecte (OOA) și proiectarea orientată pe obiecte (OOD)
- Să folosească tipurile de diagrame pentru a exprima o structură sau o comportare în programare, pentru a documenta un sistem software
- Vor învăța să traducă diagramele în cod folosind un limbaj orientat obiect

Cine poate participa: programatori care doresc să se familiarizeze cu folosirea UML în activitatea de programare

Cerințe prealabile: chiar dacă UML nu se folosește exclusiv în OOP este necesară cunoașterea unui limbaj orientat pe obiecte (de preferință Java, C++, C# sau Python) cel puțin la nivel mediu precum și noțiunile de bază ale programării orientate pe obiecte.

Infrastructura folosită: proiector, tabla, calculator personal pe care este instalat un mediu de dezvoltare Java (Eclipse), C++ sau C# (Microsoft Visual C++/C# Express Edition, Microsoft Visual Studio, etc).

Cursuri conexe: Șabloane de proiectare, Programare orientată pe obiecte, limbaje de programare orientate pe obiecte (C++, Java, C#)

Descriere

Cursul oferă o abordare teoretică și practica a UML 2.x cu accent pe aplicarea lui în activitatea de programare. Exemplele, temele și problemele practice oferă o înțelegere bună a modului de folosire a diagramelor UML în principal prin următoarele activități:

- Parcurgerea și înțelegerea principalelor elemente grafice, tipurilor de diagrame și contextul în care acestea se folosesc
- Înțelegerea, interpretarea diagramelor UML construite de alții
- Construirea de diagrame pentru a exprima o structură (aspecte statice) sau un comportament (aspecte dinamice)

- Translatarea diagramelor în cod folosind un limbaj orientat pe obiecte (C++, Java, C#) pentru a vedea particularități sau variante legate de un limbaj de programare particular
- Folosirea UML pentru modelare în fazele de specificare a cerințelor, analiza și proiectarea orientată pe obiecte

Chiar dacă UML nu este legat de nici un proces de dezvoltare software, este importantă înțelegerea particularităților de modelare pentru fiecare fază sau tip de activitate ce țin de dezvoltarea de software. Pentru acest lucru se prezintă pe scurt RUP – Rational Unified Process, se pun în evidență perspectivele de modelare, vederile asupra unui sistem software.

Cursul nu este legat de nici o unealtă CASE (cum ar fi ROSE) sau unealtă particulară de editare a diagramelor UML.

Conținut

1. Introducere
2. Diagrame de clase
3. Diagrame de secvență
4. Diagrame de obiecte
5. Diagrame de pachete
6. Diagrame de instalare
7. Cazuri de folosire
8. Diagrame state machine
9. Diagrame de activitate
10. Diagrame de comunicare
11. Structuri compozite
12. Diagrame de componente
13. Colaborări
14. Diagrame sintetice de interacțiune
15. Diagrame de timp
16. Modelarea UML în timpul dezvoltării software-ului în fazele de specificare a cerințelor, analiza și proiectarea orientată pe obiecte
17. Bibliografie