OMG Certified UML Professional, voorbereiding op de certificering

Praktijkcursus van 3 dagen - 21u
Ref : OCU - Prijs 2024 : € 2 240 excl. BTW

Deze opleiding is gebaseerd op een casestudy en stelt u in staat de UML-taal op een efficiënte en pragmatische manier te leren en te gebruiken in uw softwareontwikkelingsprojecten. Ze bereidt u ook voor op de UML OCUP 2 Foundation certificering van OMG (Object Management Group).

PEDAGOGISCHE DOELSTELLINGEN

Na afloop van de opleiding kan de cursist:

Gebruik van UML in verschillende software engineering activiteiten

Structurele modellering beheersen

Gedragsmodellering beheersen

Beheersing van alle elementen die worden behandeld door de OCUP 2 Foundation certificering

PEDAGOGISCHE METHODEN

Er wordt gewerkt op papier, in de vorm van een workshop die uitwisselingen binnen de groep stimuleert.

CASE STUDY

Uitwerking van een volledige casestudy, waardoor een goed begrip mogelijk wordt van het nut van de diagrammen in een reële context.

CERTIFICERING

Deze opleiding omvat een voucher waarmee u het certificeringsexamen "OMG Certified UML Professional" kunt afleggen in een Pearson Vue centrum. De taal voor het behalen van de certificering is het Engels.

HET PROGRAMMA

laatste update: 11/2021

1) Waarom modelleren?

- Het belang van UML-modellering in het ontwikkelingsproject, van functiemodellering tot implementatie
- De organisatie van de modellering met behulp van structurering in pakketten.
- Aan de slag met de casestudy: begrijpen van de algemene behoeften van de cliënt.

2) Modellering van de functie

- Modelleren van de functieprocessen met het activiteitendiagram.
- Modelleren van het domein met het klassendiagram.
- Modelleren van de statussen van de functie-entiteiten met het toestandsdiagram.

3) Modellering van de eisen

- Modelleren van de externe omgeving van het systeem met de actoren.
- Het systeemgedrag definiëren met usecases. Usecasediagram. Sequentiediagram.
- Niet-functionele eisen met behulp van de FURPS-benadering en het begrip UML "requirement".

DEELNEMERS

Softwarearchitecten, ontwikkelaars, projectleiders, MOA, AMOA, MOE.

VOORAFGAANDE VEREISTEN

Ervaring vereist in softwareontwikkeling, bij voorkeur in een objectgeoriënteerde taal.

VAARDIGHEDEN VAN DE CURSUSLEIDER

De deskundigen die de cursus leiden zijn specialisten op het betreffende vakgebied. Zij werden geselecteerd door onze pedagogische teams zowel om hun vakkennis als hun pedagogische vaardigheden voor elke cursus die zij geven. Zij hebben minstens vijf tot tien jaar ervaring in hun vakgebied en oefenen of oefenden verantwoordelijke bedrijfsfuncties uit.

BEOORDELINGSMODALITEITEN

De cursusleider beoordeelt de pedagogische vooruitgang van de deelnemer gedurende de gehele cursus aan de hand van meerkeuzevragen, praktijksituaties, praktische opdrachten, …
De deelnemer legt ook van tevoren en naderhand een test af ter bevestiging van de verworven kennis.

PEDAGOGISCHE EN TECHNISCHE MIDDELEN

• De gebruikte pedagogische middelen en cursusmethoden zijn voornamelijk: audiovisuele hulpmiddelen, documentatie en cursusmateriaal, praktische oefeningen en correcties van de oefeningen voor praktijkstages, casestudies of reële voorbeelden voor de seminars.
• Na afloop van de stages of seminars verstrekt ORSYS de deelnemers een evaluatievragenlijst over de cursus die vervolgens door onze pedagogische teams wordt geanalyseerd.
• Na afloop van de cursus wordt een presentielijst per halve dag verstrekt, evenals een verklaring van de afronding van de cursus indien de stagiair alle sessies heeft bijgewoond.

TOEGANGSMODALITEITEN EN -TERMIJNEN

De inschrijving dient 24 uur voor aanvang van de cursus plaatsgevonden te hebben.

TOEGANKELIJKHEID VOOR MINDERVALIDEN

Is voor u speciale toegankelijkheid vereist? Neem contact op met mevr. FOSSE, contactpersoon voor mindervaliden, via het adres psh-accueil@ORSYS.fr om uw verzoek en de haalbaarheid daarvan zo goed mogelijk te bestuderen.

4) Modellering van het systeemontwerp. Structurele weergave

- Definiëren van het technisch platform - implementatiediagram.
- De architectuur van de code definiëren. Diagram van pakketten.
- Goede praktijken voor het scheiden van doelstellingen en het beperken van afhankelijkheden - gelaagd structureringspatroon.
- Klassen modelleren vanuit een structureel oogpunt? Klassendiagram.
- De attributen definiëren.
- Objecten verbinden met de koppelingen tussen klassen.
- Factoriseren van de code met veralgemening: correct gebruik van overerving.

5) Modellering van het systeemontwerp. Gedragsweergave

- De interne interacties van het systeem met het sequentiediagram beschrijven.
- De handelingen definiëren om te voldoen aan de functionele eisen.
- Het gedrag van een object samenvatten met het toestandsdiagram.

6) Modelleren van inzetbare componenten

- De componenten en hun interfaces definiëren - componentdiagram.
- De implementatie van de componenten definiëren - implementatiediagram.

7) Voorbereiding op het praktisch verloop van het OCUP 2-examen

- Algemene herhaling.
- Oefenen op voorbeeldvragen.

DATA

Neem contact met ons op