

Opleiding : Open Source, de Software Factory implementeren

Praktijkcursus - 5d - 35u00 - Ref. OSU

Prijs : 3010 € V.B.

Ter ondersteuning van ontwikkelteams brengt een softwarefactory de volgende diensten samen: IDE, versiebeheer, verschillende tests, implementatie en build. Deze cursus legt uit hoe je zo'n fabriek opzet met behulp van de bekendste Open Source software op dit gebied: GIT, JUnit, JMeter, Maven, Jenkins, enz.

Pedagogische doelstellingen

Aan het einde van de training is de deelnemer in staat om:

- ✓ De verschillende stadia in de softwarelevenscyclus begrijpen
- ✓ Een ontwikkelingsinfrastructuur bouwen en implementeren
- ✓ Bouwgereedschappen en integratietests opzetten
- ✓ Leer de beste praktijken in continue integratieprocessen
- ✓ Leren over samenwerkings- en gemeenschapswerkmethoden en de belangrijkste specifieke tools
- ✓ Weten hoe je risico's beoordeelt en contact onderhoudt met eindgebruikers

Doelgroep

Ontwikkelaars, projectmanagers.

Voorafgaande vereisten

Basiskennis van softwareontwikkeling. Kennis van het Java EE platform en de Java taal is een pré, maar niet essentieel.

Praktische modaliteiten

Praktisch werk

Alle programma's die tijdens de praktijksessies worden gemaakt, hebben de vorm van skeletten die de deelnemers zelf invullen.

Opleidingsprogramma

DEELNEMERS

Ontwikkelaars, projectmanagers.

VOORAFGAANDE VEREISTEN

Basiskennis van softwareontwikkeling. Kennis van het Java EE platform en de Java taal is een pré, maar niet essentieel.

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.

1 De grondbeginselen

- De stadia in de softwarelevenscyclus (ontwerp, ontwikkeling, acceptatie, productie).
- Het softwareontwikkelingsmodel: V-model, incrementeel en iteratief model.
- Softwarefabriek, continue integratie.
- De verschillende soorten omgevingen (ontwikkeling, acceptatie, productie).
- De dagelijkse werking van de fabriek. De belangrijkste gereedschappen die worden gebruikt.

2 Versiebeheer

- De functies.
- De verschillende bronbeheerders: gecentraliseerd of gedistribueerd.
- De problemen van het integreren van verandering.
- De rol van takken en tags.

Praktisch werk

Een GIT-repository beheren.

3 De build instellen en automatiseren

- Het opzetten van een buildserver, de verschillende tools en installatietypes.
- Configuratie: hoofdconfiguratiepagina, GIT/SVN-configuratie, mailserver.
- Jenkins en de build, best practices en aanbevolen methoden.
- Kennisgevingsstrategieën en -technieken.
- Repareer de afhankelijkheden tussen bouwtaak.
- Jenkins en Maven: herinnering over Maven, configureren van de Maven build, implementatie in een Maven repository.

Praktisch werk

Het maken en configureren van een Maven Jenkins project gebaseerd op een GIT repository.

4 Testautomatisering

- Welke tests en met welk doel?
- Testomgevingen.
- Eenheids- en integratietests automatiseren. PHPUnit, JUnit, ...
- Rapportconfiguratie.
- Testdekking meten.
- Automatisering van acceptatietests.
- Prestatietests automatiseren met JMeter.
- Testuitvoeringstijden optimaliseren.

Praktisch werk

Het Jenkins-project aanpassen met JUnit en prestatietests controleren met JMeter.

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.

5 Validatietests

- Vereisten voor validatietest.
- Testopslagplaatsen en hun implementatie.
- Gegevensgeneratoren testen.
- De functionaliteit van testrobots.
- HMI-testrobots.
- Testrobots voor webtoepassingen.

Praktisch werk

Implementatie van een testrepository, een testgegevensgenerator, GUI-testrobots en testrobots voor webapplicaties.

6 Geautomatiseerde inzet

- Het implementatiescript instellen.
- Databases bijwerken.
- Minimale tests. Achteruit.

Praktisch werk

Aanpassing van het Jenkins-project om de implementatie te automatiseren.

7 Metriek

- Rapport genereren.
- Analyse- en rapportagetools (PMD, Findbugs, Cobertura, Emma, Checkstyle, Jira Software, enz.).
- Publicatie van resultaten.

Praktisch werk

Metrics tools instellen.

8 Kwaliteit

- Risico's verminderen door continue integratie.
- Inzicht in samenwerken in een team. De bijdrage van gebruikers aan de tools.
- Werken met eindgebruikers.
- Interactie met bugbeheertools, Bugzilla, Mantis, Bug Tracking, enz.
- Het beheren van een Open Source migratieproject: onderzoek naar marktinstrumenten, risico's en voordelen.

Praktisch werk

Gebruik van tools voor het beheren van anomalieën.