GLG101 - Test et Validation du Logiciel

Mis à jour le 12-02-2024

Code : GLG101

Unité d'enseignement de type cours

6 crédits

Volume horaire de référence (+/- 10%) : **50 heures**

Responsabilité nationale : EPN05 - Informatique / 1

Contact national :

EPN05 - Informatique

2 rue Conté
33.1.27A
75003 Paris
01 40 27 26 81
Safia Sider
safia.sider@lecnam.net

Présentation

Prérequis

- Avoir de préférence suivi les UE du cycle préparatoire I1 en Informatique. Complète l'UE de Informatique cycle préparatoire GLG 105 mais peut être suivi indépendamment. S'insère dans le cursus des diplômes concernés (RNCP, diplômes d'ingénieur, masters...).

Les auditeurs de ce cours sont supposés avoir une familiarité avec les méthodes et les langages de programmation pratiqués dans l'industrie (Java, PHP, C#, C++, C, Ada, etc.), ainsi qu'une connaissance générale, acquise dans d'autres cours ou dans la pratique, des systèmes d'exploitation et de l'architecture des applications.

Objectifs pédagogiques

La professionnalisation et la valorisation du test au sein de l'entreprise sont des enjeux majeurs, motivés par une réalité économique. Omniprésent tout au long du cycle de développement, le test logiciel est une activité qui mobilise de nombreuses connaissances et savoir-faire pour produire des tests pertinents.
L'unité d'enseignement GLG 101 vise à donner les fondamentaux dans ce domaine en forte croissance. Pour ceux qui le souhaitent, elle donne les notions pour préparer la certification ISTQB (International Software Testing Qualifications Board) du métier de testeur niveau fondation, certification délivrée en France par le CFTL (Comité Français des Tests Logiciels). Les cours se déroulent par plages de 2 fois 2h mêlant apports fondamentaux et mise en pratique.

Programme

Contenu

La formation a pour but de :

- Savoir placer les tests dans le cycle de développement
- Savoir ce que sont et comment utiliser les tests statiques
- Maîtriser les techniques de conception de tests dynamiques
- Connaître les principes du management des tests
- Savoir différencier les différents outils de tests du marché et avoir utilisé un outil

Les cours s'articulent autour des six modules suivants :

1. Fondamentaux des tests

- Origine des défauts logiciels
- Rôle des tests dans le développement et la maintenance
- Planification et contrôle des tests
- Exécutions et critères d'arrêt des tests
- Aspects sociologiques du métier de testeur

2. Tester pendant le cycle de vie logiciel

- Tests et modèles en V, ainsi que de développement itératif et agile
- Test en Agile
- Niveaux de tests : composants, intégration, système
- Cible des tests : fonctionnels / non fonctionnels, architecture logicielle, non régression
- Tests de maintenance

3. Techniques de test

- Techniques statiques et revues de code
- Techniques "boites noires"
- Techniques "boite blanches"
- Métriques logicielles - Qualité du code
- Impact du choix d'une technique de test

4. Gestion des tests

- Tâches du responsable des tests et des testeurs
- Planification des tests
- Estimation de l'effort de test en fonction des techniques utilisées
- Suivi et contrôle du déroulement des tests
- Gestion de configuration
- Gestion des rapports d'anomalie

5. Outils de support aux tests

- Classification des outils de tests
- Approche CI/CD (Intégration continue / déploiement continu)
- Bénéfices potentiels et risques liés aux outils de tests
- Outils d'aide aux tests statiques
- Outils d'aide à la spécification des tests
- Outils d'aide à l'exécution (test automatisés) et au suivi des tests
- Introduire un outil dans une organisation

6. Etude d'outils de tests :

- Présentation de la méthode Test Driven Development
- Famework JUnit (TP)
- Tests automatisés
- CI/CD

Modalités de validation

- Contrôle continu
- Examen final

Description des modalités de validation

Contrôle continu et examen final 3h.

Bibliographie

Titre	Auteur(s)
Le Génie logiciel (Que sais-je n°2956, PUF).	J. Printz
Software testing techniques (Van Nostrand Reinhold)	B. Beizer
Testing object-oriented systems, models, patterns, and tools (Addison Wesley) 2000	R. Binder
Software Engineering Standard Collection - Standards 829-2008, 982.1, 1044, 1061, 610.12	IEEE
Software engineering economics, 1981, et Software cost estimation with COCOMO II (Prentice Hall) 2002.	B. Boehm
Software testing a craftsmans approach 3rd edition - CRC press	Jorgensen

Introduction to software testing - Cambridge university press	Ammann
http://www.cftl.net/	CFTL
http://www.istqb.org/	ISTQB
Pratique des tests logiciels	J. Printz, JF. Peyre