



DMX FORMATION & TEST, OFFRE DE FORMATION :	2
AUTOMATISATION DES TESTS AVEC L'OUTIL SELENIUM (2 jours)	3
Objectif :	3
Public :	3
Prérequis :	3
Programme :	3
Rappel sur le test.....	3
Choix de l'automatisation.....	3
Outil Selenium	3
Selenium RC et Webdriver	3
Selenium Grid	4
Conclusion.....	4
Le Test Logiciel (3 jours).....	5
Objectif :	5
Public :	5
Prérequis :	5
Programme :	5
Rappel sur le test.....	5
Phase de préparation et suivi	5
Phase de conception	5
Phase d'exécution	6
La recette	6
Bilan.....	6
GESTION DES TESTS AVEC UN OUTIL OPENSOURCE (2 jours).....	7
Objectif :	7
Public :	7
Pré-requis :	7
Programme :	7
Rappel sur le test.....	7
Concevoir un test	7
Choisir un outil Open Source	7
Présentation de l'outil TestLink.	7
Organiser ses tests.....	7
Gérer l'exécution	8
Personnalisation.....	8
Conclusion.....	8
TESTER DES APPLICATIONS MOBILES (2 jours)	9
Objectif :	9
Public :	9
Prérequis :	9
Programme :	9
Les problématiques liés aux tests d'applications mobiles	9
Construire une stratégie de test	9
Tests génériques – initiative AQUA.....	9



Applications Android	9
Applications IOS	9
Outils-externalisation-Automatisation	10
FORMATION AUTOMATISATION DES TESTS LOGICIELS AVEC DES OUTILS OPEN-SOURCE (3 jours)	11
Objectif :	11
Durée : 3 jours	11
Audience :	11
Prérequis :	11
Programme :	11
Rappel sur les tests	11
Introduction à l'automatisation	11
Les tests unitaires	11
L'intégration continue	11
Les tests systèmes	12
Programmation par mot clef	12
Rapport de test	12
Conclusion	12
LES TESTS EN MODE AGILE (3 jours)	13
Objectif :	13
Public :	13
Prérequis :	13
Programme :	13
Cycle de vie et projets agiles	13
Le test dans le processus de développement Agile	13
Le modèle ATDD, les spécifications exécutable	14
Le TDD (Test Driven Development)	14
Intégration continue	14

DMX FORMATION & TEST, OFFRE DE FORMATION :

Automatiser ses tests d'interface avec un outil open source selenium (2 jours).

Test logiciel (3 jours).

Gestion des tests avec un outil open source testlink (2 jours).

Tester des applications Mobile (2 jours).

Automatisation des tests (3 jours).

Les tests en mode agile (3 jours).



AUTOMATISATION DES TESTS AVEC L'OUTIL SELENIUM (2 jours)

Objectif :

Acquérir des compétences pour mettre en œuvre une démarche d'automatisation et en particulier des tests d'interface. Cette formation permet de mettre en œuvre tous les concepts aux travers de TPs avec selenium.

Public :

Testeur ou développeur désirant se former aux techniques d'automatisation. Chef de projet test.

Prérequis :

Connaissance du test, niveau débutant en développement et connaissance HTML.

Programme :

Rappel sur le test

Choix de l'automatisation

- Qu'automatise-t-on ?
- Valeur ajoutée et ROI
- Bonnes pratiques d'automatisation
- Présentation des outils du marché

Outil Selenium

- Selenium IDE et autres composants
- Capture et rejeu
- Méthodes de localisation des éléments graphiques
- Présentation des commandes selenium
- Cas pratiques

Selenium RC et Webdriver

- Selenium RC
- Webdriver
- Génération de script de test (java, C#, python ...)
- Re travail des scripts de test selon les bonnes pratiques
 - Jeux d'essais
 - Page Object
- Génération de rapport (ant et JUNIT)
- Tester le web sur un mobile avec Webdriver
- Cas pratiques



Selenium Grid

- Présentation de Grid
- Cas pratiques

Conclusion



Le Test Logiciel (3 jours)

Objectif :

Savoir mener les activités de test de la planification au PV de recette. Cette formation inclue des travaux pratiques avec un outil de test open-source.

Public :

- Utilisateurs, chefs de projet, assistance à maîtrise d'ouvrage, MOE
- Responsables d'une cellule de tests
- Informaticiens participant aux tests

Prérequis :

Cette formation ne nécessite pas de prérequis.

Programme :

Rappel sur le test

- Enjeux des tests
- Les différents niveaux des tests au sein des différents cycles de développement
- Les différents types de tests : fonctionnels et non fonctionnels
- Processus de test
- Notion de couverture

Phase de préparation et suivi

- Qu'est-ce qu'une stratégie de test
- Zoom sur la stratégie basée risque
- Définir son effort de test, les moyens à mettre en œuvre, les critères d'arrêts
- Mise en application sur un cas concret.
- Estimation des tests
- Suivi qualité et planification
- Mise en application

Phase de conception

- Les revues
- Qu'est-ce qu'une exigence ?
- Répertoire et organiser ses exigences
- Gérer un référentiel d'exigences
- Mise en application avec un outil open source
- Répertoire et organiser ses tests
- Concevoir ses tests : techniques de test fonctionnel (partition par équivalence, valeurs aux limites, table de condition, cas d'utilisation etc.)
- Mise en application avec un outil open source
- Les tests non fonctionnels
- Cas particulier des tests d'intégration
- Cas particulier des tests unitaires : revue de code



Phase d'exécution

- Préparer et maîtriser son environnement de test et ses données de test
- Gérer les livraisons
- Exécuter les tests unitaires et mesurer la couverture de code
- Organiser ses tests d'intégrations : stratégie d'intégration et planification
- Organiser les campagnes de tests systèmes et d'acceptation
- Suivi des tests et des anomalies
- Quand arrêter les tests
- Mise en application avec un outil Open source

La recette

- Objectifs
- Responsabilité MOA -MOE
- Critères de démarrage et critères d'arrêt
- Recette opérationnelle (qualité de service, test de récupération, HA etc.)
- Recette utilisateur (cas d'utilisation, Beta test)

Bilan

- Générer un rapport de test
- Générer un bilan et améliorer son processus de test
- Mise en application avec un outil open source
- Comment utiliser un outil



GESTION DES TESTS AVEC UN OUTIL OPENSOURCE (2 jours)

Objectif :

Acquérir des compétences pour gérer un référentiel de test avec un outil de test. Tous les concepts seront illustrés aux travers de TP avec TestLink.

Public :

Testeur, développeur, Chef de projet test, chef de projet.

Pré-requis :

Connaissance du test.

Programme :

Rappel sur le test

Concevoir un test

- Présentation des techniques de test (partition et limites, table de condition, transition d'états) Exercices.
- Exigences, tests, condition de test, cas de test, procédure de test, pré-requis.
- Qu'est-ce qu'une couverture
- Nécessité des bonnes pratiques (couverture, valorisation, écriture des tests). Quelles informations sont nécessaires? Etc.

Choisir un outil Open Source

- Processus de choix d'un outil
- Nécessité d'un guide d'utilisation

Présentation de l'outil TestLink.

- Installation de l'outil
- Présentation des fonctionnalités

Organiser ses tests

- Gestion des exigences
- Gestion du référentiel de tests avec TestLink : écriture et organisation des tests
- Couverture



Gérer l'exécution

- Lien avec Mantis (gestion des anomalies)
- Organiser l'exécution de ses tests (plan de test, build, affectation)
- Gérer les environnements de tests
- Gérer les résultats de test : lien avec le gestionnaire d'anomalie, récupérer des résultats de tests automatiques
- Sortie des rapports de campagnes de tests
- Cas pratiques sur tous les points abordés

Personnalisation

- Gestion des droits, personnalisation : attribut et mot-clef,

Conclusion



TESTER DES APPLICATIONS MOBILES (2 jours)

Objectif :

Introduire le test des Applications Mobiles : stratégie de test, techniques de test et outillage. (2 jours).

Public :

Testeur ou développeur. Chef de projet test.

Prérequis :

Base en informatique.

Programme :

Les problématiques liés aux tests d'applications mobiles

- Les différents types de technologies
- Les différents types d'applications
- Les cycles de développement

Construire une stratégie de test

- ISO 25000 et test mobile
- Considérations pour sélectionner ses cibles de tests
- TP

Tests génériques – initiative AQUA

- Quelles techniques de test choisir.
- Initiative AQUA
- Construire des spécifications de test « génériques »
- TP

Applications Android

- Problématiques spécifiques (taille écran, cycle de vie, diversité ...)
- Les outils de test (SDK Android, contrôle de code, émulateur, robustesse ...)
- Tester sur des mobiles, tablette (logs, copie ...)
- TPs

Applications IOS

- Problématiques spécifiques (taille écran, cycle de vie, diversité ...)
- Les outils de test (contrôle, simulateur, ...)
- Tester sur des mobiles, tablettes (logs, copie ...)



Outillage-externalisation-Automatisation

- Sous-traiter les différentes possibilités
- Automatisation, état de l'art
- TP (automatisation)



FORMATION AUTOMATISATION DES TESTS LOGICIELS AVEC DES OUTILS OPEN-SOURCE (3 jours)

Objectif :

Acquérir les connaissances pour mettre en place une automatisation des tests logiciels avec des outils Open source. Ce cours couvre les différents types (statique, structurel et fonctionnel, IHM) et niveaux de tests (test unitaire et test système).

Durée : 3 jours

50% de travaux pratiques : JUNIT3, JUNIT4, cobertura, Selenium, Robot Framework etc.

Audience :

Chef de projet, développeur, testeur, responsable validation.

Prérequis :

Compétence minimum en développement logiciel, notion sur le test.

Programme :

Rappel sur les tests

Définition du test – Les processus de test – Les différents niveaux de test (unitaire, intégration, système, recette) – Les différents types de test (statique, structurel, fonctionnel et non fonctionnels) et leurs techniques associées. ISO-9126. Les différents cycles de développement (V, itératif, Agile)

Introduction à l'automatisation

Mythes et réalités-Que peut-on automatiser ? Pourquoi automatiser ? Dans quels types de cycle ? ROI.

Les tests unitaires

Environnement de test unitaire- Bouchon- Test statique et outillage – Tests structurel et outillage. Test fonctionnel et non fonctionnel.

Présentation JUNIT3, ant et Cobertura.

TP : JUNIT3, Cobertura, ant et eclipse.

L'intégration continue

Concepts, gains et objectifs. Présentation et mise en place d'une plateforme.



Les tests systèmes

Environnement de tests systèmes. Les différentes stratégies de développement d'un simulateur. Bonnes pratiques. Qu'est-ce qu'un framework de test? Test d'API, Web Service, Test d'IHM.

TP JUNIT JUNIT 4 , Web Services et DBUnit.

Test d'IHM les spécificités- Enregistrement Rejeu-Modularité- Reconnaissance graphique.

Presentation Selenium.

TP (enregistrement/Rejeu et Modularité) : Selenium + Junit 4. Mise en place avec la plateforme d'intégration continue.

Programmation par mot clef

Concepts, automatisation systématique.

Presentation Robot Framework.

TP (Programmation par mot clef) avec Robot Framework.

Rapport de test

Générer un rapport de test. Couplage avec un outil de gestion de test. Démo avec Testlink et selenium.

Conclusion

Quelle compétence, quel profil. Choix d'un outil- Mise en place- Formation- Manuel d'utilisation – Bonnes pratiques. Coût et ROI



LES TESTS EN MODE AGILE (3 jours)

Objectif :

Introduire le test en mode agile: stratégie de test, techniques de test, outillage et automatisation. (3 jours).

Public :

Testeur ou développeur. Chef de projet test.

Prérequis :

Base en informatique et programmation.

Programme :

Cycle de vie et projets agiles

- Pourquoi adopte-t-on les méthodes agiles
- Rappel Scrum et XP
- Les User Stories
- Expression du besoin

Le test dans le processus de développement Agile.

- Présentation du Quadrant Agile
 - Q1 techniques pour construire le produit
 - Q2 métiers pour construire le produit
 - Q3 métiers pour critiquer le produit
 - Q4 techniques pour critiquer le produit
- TP : session de test outillée en mode exploratoire
- Quels tests, à quels moments (tâches, Story, feature, Release)
- Construire une stratégie de test
- TP : Stratégie de test



Le modèle ATDD, les spécifications exécutable

- Les spécifications par les tests (USE CASE, Exemples, BDD)
- Principes de fonctionnement de ATDD
- Exemple et TP avec FitNess
- BDD (Behavior Driven Development)
- Exemple et TP avec RobotFramework

Le TDD (Test Driven Development)

- Origine du TDD
- Principe
- Les 3As
- Conception émergente
- Exercice pratique en java (ou autre)

Intégration continue

- Principe
- Les différents éléments d'une plateforme d'intégration continue
- Qualimétrie du code
- TP : Mise en place d'une plateforme d'intégration continue.