

Testeur·euse Logiciels

Préparation au Titre Professionnel :
« Testeur Logiciels »

Reconnu par l'État de niveau 5 (ou niveau BAC+2)
Inscrit au RNCP

Code RNCP : RNCP39088 – Code NSF : 326 – Formacode : 31084, 46288, 31090

Objectifs de formation

Concevoir les tests d'une application
Exécuter et analyser les tests
Automatiser les scénarios de tests

Public et prérequis

- Tout public

Modalités et délais d'accès

- **Formation continue** : plusieurs rentrées par an. N'hésitez pas à nous consulter.
- **Alternance** (contrat de professionnalisation ou d'apprentissage) : rentrées tout au long de l'année.
- **Compte personnel de formation** (CPF).

Il vous faudra assister à une réunion d'information qui sera suivie d'un entretien avec le service relations École / Entreprises ainsi que des tests de logique et de positionnement.

Durée de la formation

En continu :

- Formation en centre : 630 heures
- Stage pratique en entreprise : 280 heures
- Durée totale de la formation : 910 heures

En alternance :

- Formation en centre : 630 heures
- Sur un contrat de 18 à 24 mois

Tarifs

Les tarifs dépendent de la modalité d'accès.
N'hésitez pas à nous consulter

Contact

ecole@eni-ecole.fr

Moyens pédagogiques - techniques - d'encadrement

Équipements pédagogiques :

- Un poste de travail par apprenant équipé (en quasi-totalité) :
 - d'un processeur i5
 - d'un disque NVMe (32 Go de RAM)
- Un tableau blanc interactif installé dans chaque salle de formation
- Une photocopieuse/imprimante en libre accès
- Un accès internet sur chaque poste de travail
- Un serveur et un commutateur Ethernet Gigabit dans chaque salle
- Dix routeurs Cisco

Moyens pédagogiques :

- Travaux dirigés après chaque phase de cours : explications et démonstrations par le formateur et exécutés ensuite par les stagiaires.
- Travaux pratiques pour que le stagiaire apprenne à appliquer seul ce qu'il a appris et cherche par lui-même.
Mises en situation professionnelle avec des cas spécifiques « entreprise ».
- Création de plateaux techniques qui recréent les conditions d'activités réelles des entreprises.
- Études de cas où le stagiaire doit résoudre une problématique technique ou managériale.
- Exposés oraux où le stagiaire doit préparer une présentation sur un thème technique donné.

Supports pédagogiques :

- Supports ENI Editions/internes pour chaque cours
- Un accès la Bibliothèque Numérique des Editions
- Un accès aux agréments techniques des éditeurs (Microsoft IT Academy et Microsoft Imagine Premium, Cisco Academy)
- Un accès aux revues informatiques
- Un accès au programme Microsoft Azure (licences logiciels Microsoft)
- Un accès à la plateforme [goFluent](#) (plateforme d'autoformation et programme individualisé en anglais)
- Systèmes d'exploitation Microsoft, GNU/Linux et Cisco IOS
- Logiciels bureautiques (Microsoft Office 365 : Word, Excel, Powerpoint, ...)
- Outils de développement (Visual Studio, Oracle, Eclipse, Struts, Hibernate, Tomcat, Java...)
- Systèmes de gestion de bases de données relationnelles (Oracle Database, MySQL, SQL Server, ...)

Suivi et évaluation :

Les blocs de compétences constituant le titre « Testeur Logiciels » sont les suivants :

- Concevoir les tests d'une application
- Exécuter et analyser les tests
- Automatiser les scénarios de tests

Évaluations en cours de formation mesurant compétence par compétence le degré de maîtrise de la situation professionnelle concernée : devoirs écrits, mises en situation professionnelle, QCM, études de cas, exposés oraux...

Épreuve finale : rédaction d'un rapport d'activité – soutenance orale devant un jury de professionnels extérieurs à l'École.

Possibilité d'accès au titre par capitalisation de blocs de compétences.

Débouchés et métiers

Le **Testeur Logiciels** s'assure au travers de ses actions que l'application est en tout point conforme aux demandes des utilisateurs.

Dans un cadre numérique responsable il s'assure également que les règles d'accessibilités.

Métiers :

- Testeur Logiciels
- Testeur d'Applications
- Testeur Fonctionnel
- ...

Équivalences et passerelles

La formation Testeur·euse Logiciels prépare au titre RNCP du même nom :

<https://www.francecompetences.fr/recherche/rncp/39088/>

Vous pouvez également poursuivre votre cursus de formation au sein de notre établissement (lien vers [tableau des formations](#)).

Programme détaillé

Les composantes d'un projet de développement informatique

Découverte des composantes de développement informatique (1 semaine)

- Définition et principes de base d'un projet
- Bien cadrer et démarrer un projet
- Choisir les bons outils de la gestion de projet
- Choisir un modèle de gestion de projet adapté
- Le cahier des charges
- Après le projet

Prise en main du métier de testeur

Découverte et prise en main du métier de testeur (2 semaines)

- Principes fondamentaux des tests
- Tester pendant le cycle de vie du développement logiciel
- Techniques statiques
- Techniques de conception des tests
- Gestion des tests
- Outils de support aux tests

Excel – des bases aux fonctions avancées

- Prise en main et découverte de l'outil
- Créer des tableaux avec des formules simples
- Mise en forme des tableaux
- Formules et fonctions
- Graphiques
- Gestion des tableaux longs
- Introduction aux tableaux croisés dynamiques

Savoir-Être - Posture du Consultant

Apprendre la posture professionnelle attendue

- Introduction à la Consultance
- Communication Efficace et Gestion des Relations
- Éthique et Professionnalisme
- Développement de la Pensée Critique
- Confiance en Soi et Présence Professionnelle
- Autogestion et Développement Continu

Utilisation d'un poste de travail Linux

Apprendre à utiliser un poste de travail sous Linux

- Introduction à Linux
- Découverte d'UNIX et Linux
- La ligne de commandes et le Shell
- Arborescence
- Les processus et mécanismes
- Archivage et compression

TestLink - Mise en œuvre d'un gestionnaire de tests manuels

- Outiller le référentiel de tests
- Mise en œuvre de TestLink
- Création et gestion des projets
- Gestion des exigences
- Conception et organisation des tests
- Gestion des campagnes de tests
- Exécution, suivi et reporting
- Métriques et éditions de rapports

Algorithmique – Initiation à la programmation

- Comprendre l’algorithmie
- Découvrir le Java avec Eclipse
- Connaitre les possibilités du développement
- Découvrir les variables et leurs types
- Découvrir les instructions répétitives
- Découvrir les instructions conditionnelles

Introduction au développement web front-end (HTML, CSS, JavaScript)

Découverte du développement Frontend

- Découverte du HTML5
- Fonctionnement des formulaires HTML
- Apprentissage du CSS3
- Prise en main des sélecteurs CSS
- Manipulation du DOM avec JavaScript
- Gestion d’événements avec JavaScript

Sensibilisation à l’accessibilité numérique

Appréhender le concept de l’accessibilité numérique :

- Introduction à l’accessibilité numérique
- Accessibilité et handicap, l’utilisateur dans sa diversité
- L’accessibilité au centre de votre projet web
- Études de cas

Tester l’accessibilité des sites web

Savoir auditer et tester l’accessibilité des sites web

- Les outils de l’auditeur/testeur
- Présentation et priorisation des critères de tests du RGAA
- Méthodologie d’audit de l’accessibilité d’un site web
- Études de cas

Programmation / Python

Apprentissage de la programmation avec le langage Python

- Introduction à Python
- Les fondamentaux du langage
- Les fonctions et paramètres
- Le débogage
- Les variables complexes
- La programmation orientée objet (POO)
- La gestion de données
- Introduction aux *Frameworks* Web

Langage SQL – Interrogation de bases de données

Connaître les bases du langage SQL

- Découverte des bases de données relationnelles (SGBDR)
- Prise en main de Microsoft SQL Server et Microsoft SQL Server Management Studio
- Apprentissage du SQL :
 - DDL : Structurer la base et les tables
 - DML : Manipuler les données
 - DQL : Réaliser des requêtes et exploiter les données

Comprendre l'automatisation des tests

Apprentissage d'une stratégie d'automatisation de test

- Les limites de l'exécution des tests par les *Frameworks* de test
- La nécessité de l'automatisation des tests
- L'intégration des tests dans la démarche globale
- Les tests en démarche agile : Peer Programming, Test-Driven Development
- La pratique du TDD (Test Driven Development)
- Les principes du ATDD (Acceptance Test-Driven Development)
- Définition d'une démarche d'automatisation
- Workflow de l'automatisation de test
- Architecture d'automatisation générique
- Conception de tests automatisés à partir d'un référentiel de test manuel
- Pilotage et reporting

Le Testeur Agile

Notions de test dans un environnement Agile

- Les bonnes pratiques agiles : Création collaborative de « User Story », rétrospectives, intégration continue, planification de release et d'itérations
- Les principes, les pratiques, et les processus fondamentaux Agile et les impacts sur l'organisation des Tests
- Méthodes, techniques, et outils pour les tests Agiles

Git – Gestion du contrôle de version

Connaître le fonctionnement de Git

- Introduction à Git
- Installation et configuration
- Gestion un dépôt local
- Synchronisation avec un dépôt distant
- Découvertes des branches

DevOps : Démarche et processus

Découverte de la méthodologie et des processus DevOps

- Origines et objectifs du DevOps
- Les apports des méthodes agiles sur la démarche DevOps
- Présentation de la démarche d'automatisation induite par le DevOps
- Les différents types d'automatisation : industrialiser les développements et les livraisons
- Les outils d'automatisation au sein d'une organisation DevOps
- Système de gestion des versions
- Outils de construction et gestionnaire d'artefacts
- Cycle de vie logiciel et outils pour l'intégration continue (CI) et le déploiement continu (CD)

Selenium

Automatiser les tests fonctionnels des applications web avec Selenium Web Driver

- La problématique du test des IHM web et de leur automatisation
- Présentation de la suite d'outils Selenium
- Mise en œuvre de Selenium IDE pour la capture et l'enregistrement et l'ajustement de scénarios de tests
- Piloter et exécuter les tests avec Selenium Web Driver et analyser les résultats
- Intégration de Cucumber à Selenium Web Driver pour exécuter les tests BDD
- Paralléliser l'exécution des tests avec Selenium Grid

Appium

Automatiser les tests des applications mobiles natives et web avec Appium

- Introduction à l'automatisation des tests mobiles
- Présentation d'Appium et de son principe de fonctionnement
- Mise en œuvre des tests mobiles avec Appium
- Fonctionnement d'Appium avec Selenium Web Driver pour tester les applications web mobiles
- Création et amélioration des scripts de test
- Création de rapports de test

NeoLoad

Analyser les tests de performances des services web avec NeoLoad

- Introduction aux critères de Performance
- Effectuer des Tests de Performance
- Enregistrer un parcours utilisateur
- Modifier un parcours utilisateur
- Construire une Population
- Mettre en place un Monitoring d'Infrastructure
- Exécuter un Test
- Analyser les Résultats d'un Test

Behavior Driven Development (BDD) / Cucumber

Concevoir des tests d'acceptation avec Cucumber

- Présentation des concepts liés au développement piloté par les comportements
- Le langage naturel Gherkin pour décrire des scénarios de test
- Présentation et mise en œuvre de l'outil Cucumber
- Conception et exécution de cas de tests BDD écrits en langage Gherkin et exploitation des résultats

Katalon Studio

Automatiser les tests des applications web, mobile et des API REST

- Présentation de Katalon Studio et de ses principales fonctionnalités
- Mise en œuvre des tests : capturer et rejouer un test, création et modification de scripts
- Reporting et exploitation des résultats de tests
- Mettre en œuvre le pilotage des tests par les données (DDT)
- Mettre en œuvre le pilotage des tests par les comportements (BDD)
- Intégration d'outils avec Katalon : Cucumber, Git, Jenkins

Postman

Automatiser les tests des API REST

- Principes fondamentaux des tests d'API web
- Installation et création d'un environnement pour réaliser les tests
- Création des collections de test
- Tester les requêtes des API web
- Transmissions de données dans les demandes
- Création de requêtes dynamiques avec des variables
- Inspecter les réponses
- *Automatiser l'exécution des tests d'API web avec Newman*

Jira/Xray

Créer et gérer les scénarios de test dans les projets

- Présentation de Jira
- Gestion des projets agiles dans Jira
- Gestion des utilisateurs et des groupes
- Gestion des demandes
- Gestion des workflows
- Gestion des bugs et améliorations
- Intégrer Xray à Jira pour gérer le référentiel et les campagnes de tests des projets : conception et exécution des tests
- Création de rapports et tableaux de bords pour suivre la couverture des exigences

Robot Framework

Automatiser les tests d'acceptation

- Présentation et concepts de Robot Framework
- Mise en place et installation
- Syntaxe d'écriture des tests
- Lancement des tests
- Les bibliothèques de Robot Framework
- Gérer les tests des applications Web : intégration avec Selenium
- Extension de Robot Framework avec des nouveaux keywords
- Création de bibliothèques de test
- Gestion de la documentation

Projet fin d'étude

Mise en pratique des compétences de testeur.euse logiciels

- Mise en pratique d'un scénario complet de test : gestion des exigences,
- Conception des cas de tests manuels et automatisés,
- Exécution de campagnes de tests,
- Gestion des anomalies et rapport de tests

Certification ISTQB CFTL Foundation

Préparation et passage de la certification ISTQB niveau fondation