

**ZENITY**
Trust in quality

15/01/2019

Cours 2.3

Focus sur l'automatisation des tests fonctionnels

(14h à 16h)

Sommaire

▪ Objet du cours	...	3
▪ Automatisation des tests, grands principes	...	5
▪ Scripting	...	21
▪ Service Automatisation	...	29
▪ Retour d'expérience : ROI et démonstration	...	36
▪ Principaux Robots du marché	...	39
▪ Conclusion	...	42

**ZENITY**
Trust in qualityPage
2

Objet du cours

Objet du cours

- Ce cours a pour but de :
 - Appréhender les grands principes d'une automatisation raisonnée : stratégie et mode opératoire
 - Présenter un exemple de retour d'expérience et de ROI atteint

**ZENITY**
Trust in quality

15/01/2019

Automatisation des tests

Grands principes

Automatisation des tests

Pourquoi automatiser ?

- Exécution systématique de tests de non régression, parfois laissés à l'abandon lors de campagnes manuelles
 - Par faute de temps
 - Par manque de valeur ajoutée estimée
 - Par leur côté rébarbatif
 - Par la combinatoire importante engendrée (tests de portabilité Web et mobile par exemple)
- L'automatisation permet donc aux testeurs de se concentrer sur des actions à valeur ajoutée

**ZENITY**
Trust in qualityPage
6

Automatisation des tests

Coût de l'automatisation ?

- A iso-périmètre, la conception d'un test automatique coûte en moyenne 7 fois plus cher qu'un test manuel
- **Conception manuelle** = écrire un cas de test en français décrivant les actions à réaliser et les résultats à attendre
- **Conception automatique** = scripting = développer du code qui va réaliser les actions sur l'interface, vérifier des résultats, s'interfacer avec des jeux de données

Automatisation des tests

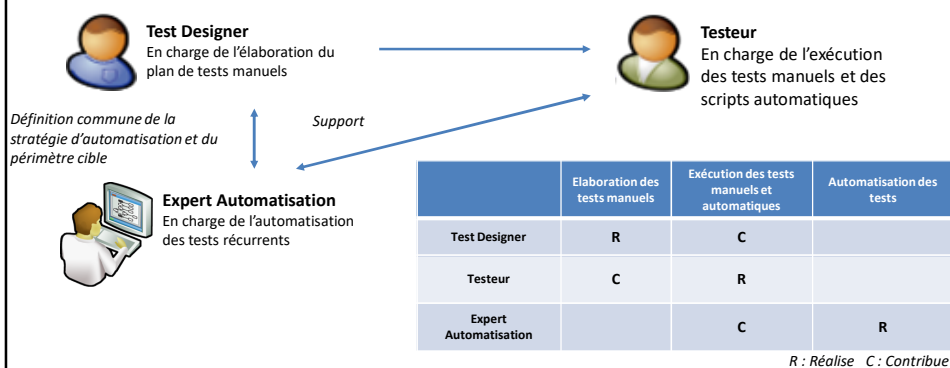
Quoi automatiser ?

- Très majoritairement les TNR pour une question de stabilité de l'environnement technique
- Rappel : TNR = Tests ne portant pas directement sur un périmètre de correction d'une anomalie, destiné à vérifier que ce qui fonctionnait avant correction continue de fonctionner après livraison
- Peut à la marge être certains tests critiques pour lesquels on veut s'affranchir des aléas humains

Automatisation des tests

Qui automatise ?

- Profil dédié ayant de fortes connaissances techniques et une maîtrise des robots d'automatisation



Automatisation des tests

Bonnes pratiques

- Déterminer un périmètre restreint pour réaliser un POC (**Proof Of Concept**) afin de :
 - Valider l'éligibilité technique (*exemple des Grilles dynamiques ALV de SAP non reconnues par QTP/UFT alors même que SAP et HP ont développé une intégration poussée entre leurs outils*)
 - Déterminer le niveau de détails nécessaire à l'automaticien pour comprendre la description des tests à automatiser
 - Faire une étude de rentabilité
- Etablir une stratégie d'automatisation

Automatisation des tests

Les bénéfices

- Gain de temps à l'exécution
- ROI certain en cas d'utilisations répétées des scripts
- Exécution des scripts planifiées durant la nuit ou le WE
- Diminution des équipes de test ou meilleure affectation des équipes
- Constitution rapide de jeux de données ou d'environnements (test, développement, pré-production, formation)
- Création d'anomalies automatiques

Automatisation des tests

Stratégie d'automatisation

Stratégie d'automatisation

Pré-requis

- **Pré-requis : organisation mature bénéficiant d'un référentiel de tests utilisé et dûment détaillé.**
- Il est préférable d'avoir recours à des outils de centralisation du référentiel de tests, pour une intégration totale et une automatisation pilotée
- Excel ou des solutions non interfaçables est possible à la condition qu'ils soient renseignés avec précision

Stratégie d'automatisation

Objectif

- **Objectif : identifier au sein du référentiel de cas de tests manuels ceux qui sont éligibles à l'automatisation**
- Réunir les différents acteurs du projet afin de déterminer des objectifs communs :
 - MOA
 - MOE
 - Testeurs

Stratégie d'automatisation

Périmètre éligible

- Afin de déterminer quels tests automatiser, différents critères sont valorisés :
 - **Fréquence d'exécution** : l'automatisation d'un cas de test n'atteindra son seuil de retour sur investissement qu'après un certain nombre d'exécutions (compris entre 4 et 7, mais pouvant selon les contextes atteindre 10 à 12 exécutions)
 - **Criticité fonctionnelle, technique** : l'automatisation peut alors être perçue comme un moyen robuste et infallible comparée à une exécution humaine.
 - **Éligibilité technique** : puis-je techniquement automatiser ce que j'ai identifié ?
 - **Maintenabilité** : quel processus de maintenance des scripts vais-je envisager ?
 - **Variabilisation** : à quelle alimentation en données vais-je avoir recours pour que mes scripts soient les plus modulaires possible ?

Automatisation des tests

Autonomie de l'automaticien

Autonomie de l'automaticien

Constat

- **Afin de rendre un cas de test compréhensible par un automaticien, il est nécessaire d'atteindre une maille de description extrêmement fine.**
- Le cas de test devient ainsi assimilable par un non-sachant qui n'a plus qu'à reproduire pas à pas les actions listées et les capturer au moyen du robot.
- **Limite** : une fois le script automatique enregistré, le cas de test manuel finement décrit devient inutile et la charge de description peu rentable...

Autonomie de l'automaticien

Constat

- **D'après vous** : comment rendre un automaticien suffisamment sachant d'une technologie et d'un métier pour qu'il soit autonome dans la création de scripts, sans pour autant exploser la charge de description du test à automatiser ?
- **Solution** : capture vidéo du test à automatiser


Autonomie de l'automaticien

Solution

- L'automaticien peut reproduire sur l'application ce qu'il visualise sur la vidéo, et capturer les différents événements
- Le tout autant de fois qu'il le souhaite, la vidéo étant un moyen statique de reproduire une suite d'actions (pas de créations intempestives de données dans les environnements de tests)

Autonomie de l'automaticien

Retour d'expérience

- Après expérimentation sur différents projets, notamment au sein du groupe  **SAFRAN** AEROSPACE - DEFENCE - SECURITY, Zenity est à même d'annoncer une réduction de 2 à 3 fois de la charge de description des tests.
- Il est toutefois indispensable de prévoir un support ponctuel permanent dans le cas où le comportement reproduit ne correspond pas exactement à la vidéo capturée préalablement, ou dans toute autre situation nécessitant un complément d'information.

Scripting

Scripting

Réflexion initiale

- Concevoir des scripts automatisés comme un assemblage de briques unitaires et non comme un monobloc de bout en bout
 - Méthodologie **B**usiness **P**rocess **T**esting de HP
 - Automatisation par mots clés (Cucumber, framework JBehave, ...)
 - Participation à la logique BDD (Behavior Driven Development)

Scripting

Réflexion initiale

- En cas de modification de l'interface, seules les briques impactées doivent être maintenues / rescriptées et non le script en entier
- Utilisation possible par des non-sachants techniques qui assemblent les briques afin de constituer leurs scénarios fonctionnels/métiers automatisés

Scripting

Mise en œuvre – Etape 1

- Enregistrement d'une version de base du script de la brique via un mode Record & Replay = capture des objets sur l'interface (même principe que l'enregistrement d'une macro sous Excel par exemple)
- Le script produit dispose d'une ossature à données figées et sans vérifications implémentées
- Selon les outils, le code produit peut être du VBS, du Python, du Java, etc.

Scripting

Mise en œuvre – Etape 2

- Instanciation du script afin de le rendre utile aux tests, évolutif et maintenable
 - **Utile aux tests** : Un script automatisé ne comportant pas de points de vérification permet juste de s'assurer de l'absence de crash inopiné, mais n'amène aucune valeur ajoutée d'un point de vue Testing. Par ailleurs il est impératif de déterminer des scénarios de reprise dans le cas où une brique tombe en erreur afin que les briques suivantes puissent s'exécuter
 - **Evolutif** : Un script à données figées (capturées lors du record) peut s'avérer obsolète après la première exécution car fonctionnellement l'interface ne peut pas boucler sur les mêmes valeurs
 - **Maintenable** : Conception en briques + documentation code + réalisation de documents externes décrivant le fonctionnement des scripts

Scripting

Définition des checkpoints

- Implémentation des vérifications définies dans la description des tests afin de s'assurer que le résultat affiché est conforme à celui attendu.
- Vérifications possibles en cours et en fin de script
- Développement de bouts de codes à intercaler dans le script produit via le Record & Replay, en utilisant les objets capturés, des expressions régulières, etc.

Scripting

Variabilisation des données

- Variabiliser les données figées en les remplaçant par des variables informatiques interfacées avec un réceptacle de données (feuille Excel, instance de base de données, variables d'environnement, etc.)
- Plusieurs objectifs :
 - Faire boucler le script sur des données différentes
 - Permettre aux acteurs métiers/fonctionnels de renseigner des données pertinentes sans avoir à rentrer dans le code

Scripting

Scénarios de reprise

- L'automatisation ne peut être prédictive et exhaustive
- En cas de non implémentation d'un comportement particulier (par exemple ouverture d'une pop-up si un jeu de données spécifique est en entrée) le script a de fortes chances de tomber en erreur, annulant par la même occasion l'ensemble des tâches suivantes qu'il était censé exécuter
- **Implémentation de solutions de contournement autour des zones sensibles**

**ZENITY**
Trust in quality

15/01/2019

Service Automatisation

Exemple de découpage en unités d'œuvre

Service Automatisation

UO Simple

- **Périmètre** : réalisation d'un script simple composé d'une brique de connexion/déconnexion à un applicatif et d'une brique d'automatisation du formulaire d'une page sans vérification de valeurs.
- **Pré-requis** : données existantes
- **Charge prévisible** : 0.5 à 1 jour

**ZENITY**
Trust in qualityPage
30

Service Automatisation

UO Moyenne

- **Périmètre** : réalisation d'un script moyen enchainant plusieurs pages sur une même application
- **Actions à réaliser**
 - Scripting des actions + récupération des données entre les pages
 - Insertion de checkpoints
 - Variabilisation des données
- **Charge prévisible** : 4 à 8 jours (hypothèse d'un enchaînement moyen de 5 formulaires)

Service Automatisation

UO Complexe

- **Périmètre** : réalisation d'un script complexe de bout en bout traversant plusieurs applications, faisant état des vérifications de données en base et lié à un script de nettoyage des données de tests créées.
- **Actions à réaliser**
 - Scripting des actions : activité de scripting similaire Web ou hors-Web
 - Insertion de checkpoints : L'implémentation de Database Checkpoints est plus complexe car elle nécessite un travail complémentaire de requêtage (utilisation de l'assistant Microsoft Query + écriture manuelle des requêtes)

Service Automatisation

UO Complexe

■ Actions à réaliser

- Passage de paramètres d'une application à l'autre. La cinématique est :
 1. Récupération des données de chaque brique dans la DataTable
 2. Placement de ces données dans la feuille globale qui sera visible par toutes les briques du script de bout en bout, après avoir épuré la donnée (exemple message littéral à supprimer pour ne garder que le numéro de commande)
 3. Branchement sur la feuille globale des briques alimentées par ces données
- Nettoyage des données des jeux de test : constitution d'une brique de nettoyage en bout de chaque script complexe. Nécessité de travailler avec un DBA pour identifier les impacts potentiels et les canaux de suppression

Service Automatisation

UO Complexe

- **Charge prévisible** : 10 à 20 jours (hypothèse d'un enchaînement moyen de 10 formulaires, 5 sur une application Web, 5 sur une application hors-Web)

Service Automatisation

Conclusion

- Selon la nature du script automatique à réaliser, la charge peut être multipliée par 20 !
- Un POC et une étude de rentabilité sont donc **indispensables** avant de se lancer

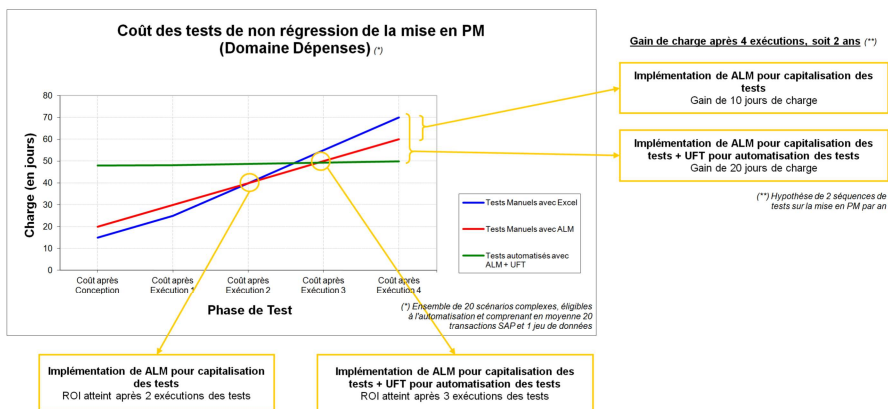
Retour d'expérience

ROI et Démonstration

Retour d'expérience

MAIRIE DE PARIS

ROI



Retour d'expérience

MAIRIE DE PARIS

Démonstration

■ Démarche :

- Réalisation par les experts métier SAP d'une vidéo par processus permettant à l'automaticien non-sachant sur SAP d'automatiser pas à pas
- Mise à jour par les testeurs en place des cas de tests centralisés dans ALM afin d'y faire figurer toutes les configurations de données à éprouver

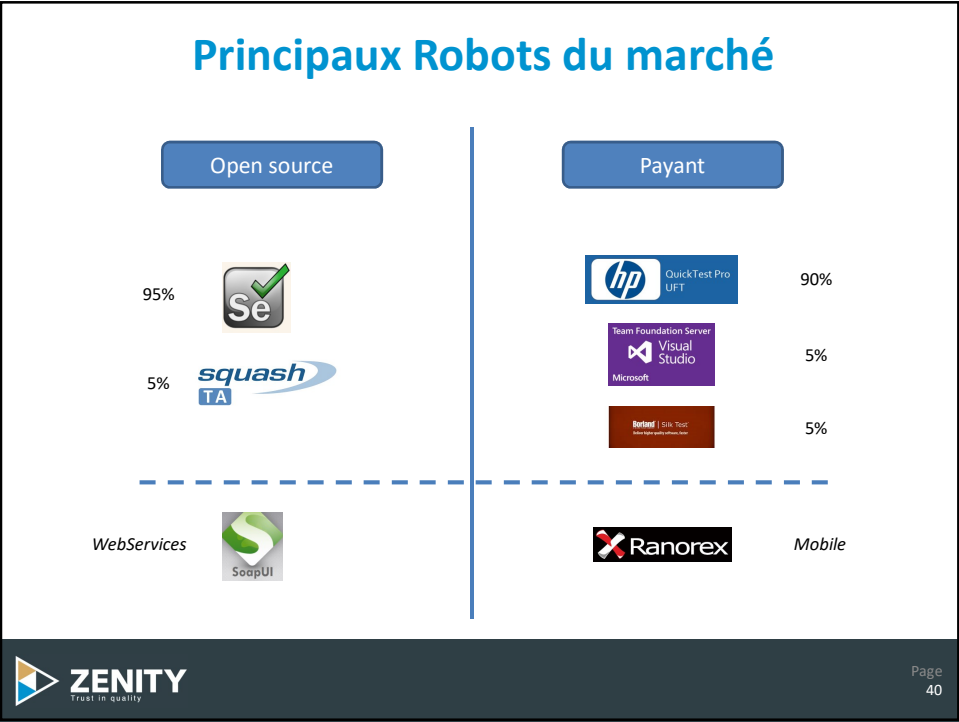
■ Vidéo démo



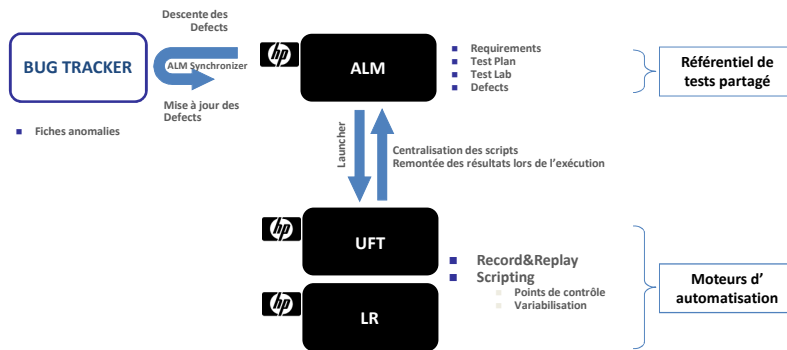
ZENITY
Trust in quality

15/01/2019

Principaux Robots du marché



Exemple d'intégration entre les outils



Conclusion

Conclusion

- L'automatisation des tests est à utiliser avec parcimonie
- Elle nécessite une organisation mature, la réalisation avant généralisation d'un POC et d'une étude de rentabilité, et la définition d'une stratégie d'automatisation
- Les profils d'automaticien sont très recherchés mais ils ont besoin de très fortes compétences techniques pouvant s'apparenter à celles d'un architecte logiciel

Merci de votre attention