

Automatisation de Tests fonctionnels

Qilin ZHANG

Sorbonne Université - Master 2 Informatique STL Alternance

5 Septembre 2019



Tuteur d'entreprise : Camilla CHRISTIANSEN, Sébastien COUDRAY

Tuteur universitaire : Emmanuel CHAILLOUX, Binh-Minh BUI-XUAN

Plan

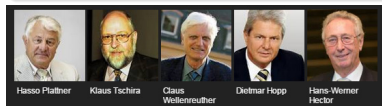
- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 Travaux réalisés
 - Architecture du projet Test-Auto
 - Développements réalisés
 - Outils et technologies utilisés
- 4 Conclusion

Plan

- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 Travaux réalisés
 - Architecture du projet Test-Auto
 - Développements réalisés
 - Outils et technologies utilisés
- 4 Conclusion

Historique de SAP

SAP SE (Systems, Applications & Products in Data Processing), fondé par 5 anciens employés d'IBM en 1972



SAP en 3 points

- 1^{er} éditeur de logiciels en Europe et 4^{ème} dans le monde
- Le plus grand fournisseur mondial de logiciels d'application d'entreprise
- Entreprise internationale : 98000 employés répartis dans 147 pays et 437000 clients de 180 pays.



Plan

- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 Travaux réalisés
 - Architecture du projet Test-Auto
 - Développements réalisés
 - Outils et technologies utilisés
- 4 Conclusion

Mon rôle au sein de l'équipe P&I LoB Finance

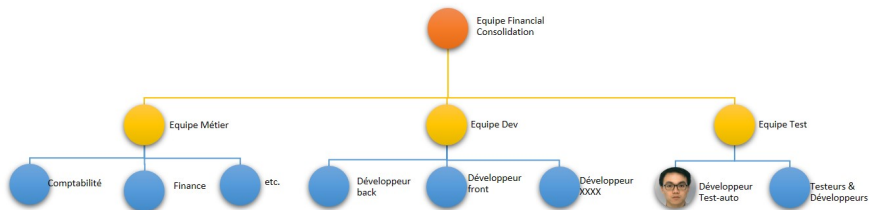
Equipe

- Apprenti Ingénieur Qualité intégré au sein de l'équipe P&I S/4HANA LoB Finance France. Elle est composée d'une quarantaine de personnes qui utilisent SCRUM comme méthode de travail.
- Développement et maintenance de SAP Financial Consolidation (FC), une application de consolidation financière et de reporting de gestion développée depuis 1999.

Mon rôle au sein de l'équipe P&I LoB Finance

Mon rôle

- Automatiser des scénarios de tests fonctionnels et tests de non-régression pour le client FC Web HTML5.



Plan

- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 Travaux réalisés
 - Architecture du projet Test-Auto
 - Développements réalisés
 - Outils et technologies utilisés
- 4 Conclusion

Deux types de test

Test des nouvelles fonctionnalités

Test fonctionnel sur les nouvelles fonctionnalités implémentés lors du dernier sprint.

Test de non-régression

Le but de la non-régression est de s'assurer qu'après une correction de bug, de nouveaux bug n'ont pas été introduits ou découverts dans l'application. Ces tests durent tout au long du développement du produit, d'où l'intérêt de les automatiser pour libérer du temps de test manuel sur les nouvelles fonctionnalités.

Deux types de test

Test des nouvelles fonctionnalités

Test fonctionnel sur les nouvelles fonctionnalités implémentés lors du dernier sprint.

Test de non-régression

Le but de la non-régression est de s'assurer qu'après une correction de bug, de nouveaux bug n'ont pas été introduits ou découverts dans l'application. Ces tests durent tout au long du développement du produit, d'où l'intérêt de les automatiser pour libérer du temps de test manuel sur les nouvelles fonctionnalités.

Ces deux types de test assurent la qualité de produit !!!

Plan

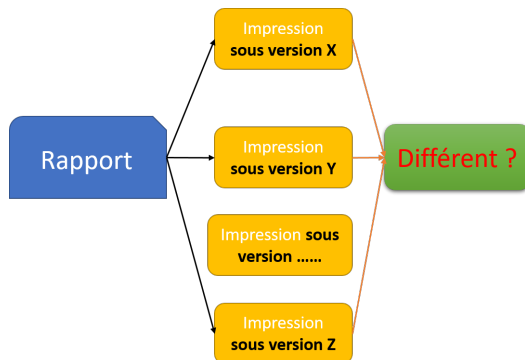
- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 Travaux réalisés
 - Architecture du projet Test-Auto
 - Développements réalisés
 - Outils et technologies utilisés
- 4 Conclusion

Problématique du test d'impression

- FC contient une fonctionnalité d'impression des rapports, ces rapports sont au format PDF, le *test de non-régression* doit assurer la qualité de cette fonctionnalité.

Problématique du test d'impression

- FC contient une fonctionnalité d'impression des rapports, ces rapports sont au format PDF, le **test de non-régression** doit assurer la qualité de cette fonctionnalité.
- Vérifier si l'impression de différentes versions de rapports donnent les mêmes résultats, et le test doit indiquer les différences s'il en trouve.



Etat de travail

Etat des tests d'impression avant mon arrivée

- Avant mon arrivée, l'équipe faisait ces tests manuellement. L'exécution répétée de ces scénarios entraînait un risque d'erreur humaine important dans la comparaison des fichiers PDF ainsi qu'une perte de temps, d'où le besoin d'automatiser ces tests.
- Le projet de Test-Auto utilisant le langage Java et plusieurs autres technologies a été développé depuis quelques années, et il continue à couvrir de plus en plus de fonctionnalités du produit.

Etat de travail

Statut des autres équipes de SAP France

- Une autre équipe a travaillé sur un projet de comparaison entre deux fichiers PDFs pixel par pixel en Java. Ils utilisent la bibliothèque *ice-pdf*, qui génère une image indiquant la différence entre les deux fichiers comparés.
- Cette solution ayant fait ces preuves, nous avons décidé de la récupérer et de l'adapter à nos besoin pour notre projet Test-Auto.

Plan

- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 Travaux réalisés
 - Architecture du projet Test-Auto
 - Développements réalisés
 - Outils et technologies utilisés
- 4 Conclusion

Projet Test-Auto : Comment ça marche ?

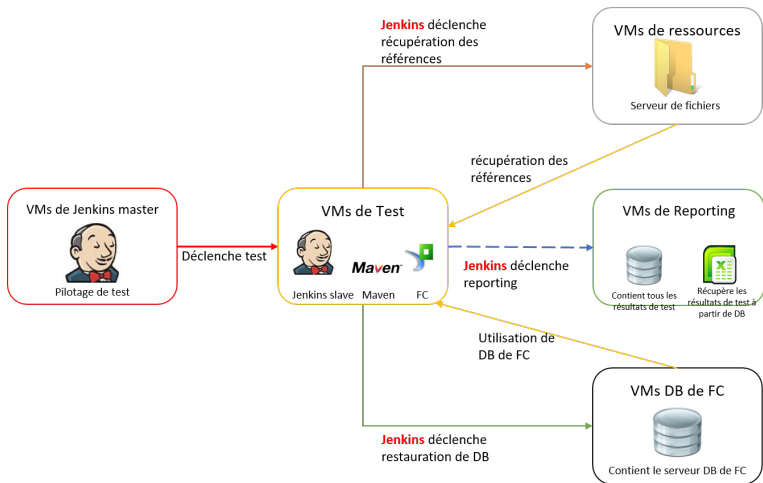
- **Projet Test-Auto** est un projet configuré par Jenkins et plusieurs types de machines virtuelles :
 - 1 **VM de Jenkins serveur** : Contient toutes les configurations de Jenkins pour le projet Test-Auto.
 - 2 **VM du DB de FC** : La DB de FC : SAP HANA, Oracle 12, Microsoft SQL Server.
 - 3 **VM de ressources** : Contient toutes les ressources que les test automatisé a besoin.
 - 4 **VM du DB de Reporting** : La DB sauvegarde les logs de tests, et visualiser ces logs au rapport Excel.
 - 5 **VM de Test** : L'environnement de test avec les logiciels et bibliothèques installés : Windows, Jenkins slave, Maven, FC client, IE, Chrom, Firefox.

Projet Test-Auto : Comment ça marche ?

La procédure d'un test automatisé

- ➊ Jenkins déclenche la restauration de DB de FC sur la VM de Test
- ➋ Jenkins déclenche la récupération des ressources de la VM de ressources
- ➌ Jenkins déclenche le test sur la VM de Test
- ➍ Jenkins déclenche la remontée des résultats de test dans la DB de Reporting

Projet Test-Auto : Comment ça marche ?



Plan

- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 **Travaux réalisés**
 - Architecture du projet Test-Auto
 - **Développements réalisés**
 - Outils et technologies utilisés
- 4 Conclusion

Automatisation des scénarios de test avec Silktest

Contenus de la tâche

- Lire et comprendre le code existant du projet Test-Auto.
- Implémentations de nouveaux scénarios de test en utilisant les technologies déjà mises en place pour le projet Test-Auto : Silktest (Silk4J), Junit, Maven, Jenkins.

Automatisation des scénarios de test avec Silktest

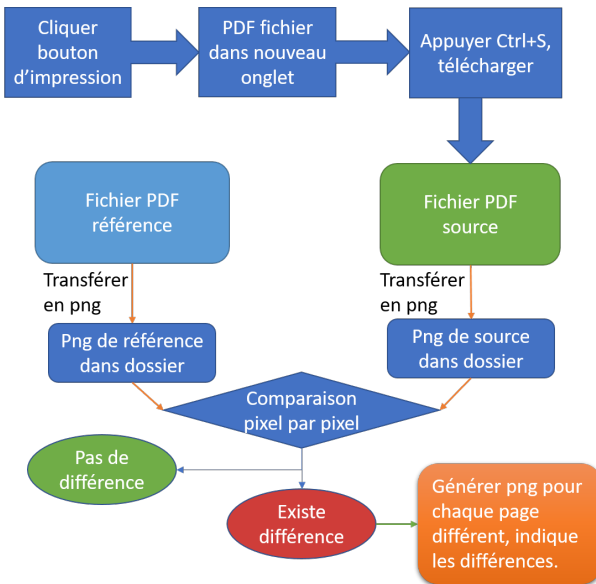
Contenus de la tâche

- Lire et comprendre le code existant du projet Test-Auto.
- Implémentations de nouveaux scénarios de test en utilisant les technologies déjà mises en place pour le projet Test-Auto : Silktest (Silk4J), Junit, Maven, Jenkins.

Résultats des premières tâches effectués

- 1 Bonne compréhension de l'architecture des Test-auto
- 2 Apprentissage du développement avec Silktest :
 - Comment Silktest interagit avec les contrôles de l'application en écrivant des requêtes XPath.

Comparaison entre deux PDF



Résultat :



Comparaison entre deux PDF

Résultat

- Les erreurs de test sont affichés dans les logs de test
 - Dans l'image de gauche, "OK" désigne un test qui passe bien et "ERROR" veut dire qu'il y a une différence dans le rapport.
 - Dans l'image de droite on voit les différences entre les rapports.

J	K	L	M
TEST_CLIENT	SCENARIO_NAME	SCENARIO_STATUS	WORKING_LANGUAGE
Web_UI5	WebUI5_PrintListReports05	OK	English
Web_UI5	WebUI5_PrintListReports04	OK	English
Web_UI5	WebUI5_PrintListReports02	OK	English
Web_UI5	WebUI5_PrintListReports01	OK	English
Web_UI5	WebUI5_PrintReport07	ERROR	English
Web_UI5	WebUI5_PrintReport06	OK	English
Web_UI5	WebUI5_PrintReport05	OK	English
Web_UI5	WebUI5_PrintReport04	OK	English
Web_UI5	WebUI5_PrintReport03	OK	English
Web_UI5	WebUI5_PrintReport02	ERROR	English
Web_UI5	WebUI5_PrintReport01	OK	English
Web_UI5	WebUI5_PrintPackage01	OK	English

Figure – Résultat de test

	Transaction Name	Begin Time	End Time	Duration (s)	status
43	clickPrint	29/08/2019 11:19:14 AM	29/08/2019 11:19:20 AM	6405	OK
44	browserSaveAsSrc	29/08/2019 11:19:20 AM	29/08/2019 11:19:34 AM	13825	OK
45	compareRefAndSrc	29/08/2019 11:19:34 AM	29/08/2019 11:19:37 AM	3358	ERROR

Figure – Détaille d'une erreur

Téléchargement de zip, unzip, comparaison de PDF

Problématique

- Pour certains scénario de tests nous avons besoin de télécharger un fichier zip.
- Le suffix du nom de ce zip contient un GUID (*globally unique identifier*).
- Ce zip contient un nombre connu de fichiers au format pdf qui ont tous le même GUID comme préfixe.

Téléchargement de zip, unzip, comparaison de PDF

Problématique

- Pour certains scénario de tests nous avons besoin de télécharger un fichier zip.
- Le suffix du nom de ce zip contient un GUID (*globally unique identifier*).
- Ce zip contient un nombre connu de fichiers au format pdf qui ont tous le même GUID comme préfixe.

Solution

- Utilisation de la bibliothèque `java.util.zip.ZipEntry` et `java.util.zip.ZipInputStream` afin de décompresser le zip.
- Triage des fichiers par date de créations puis renommage des noms par ceux défini dans les scénarios. Pour trier, j'utilise les λ -Expression.
- Comparaison des fichiers avec les références.

Comparaison de deux répertoire avec λ -Expression

Problématique

- Téléchargement d'un fichier zip
- Nom des fichiers PDF inconnus, ainsi que leur nombre dans ce zip.
- Autrement dit, nous devons comparer le contenu des deux zip (ils ont le même nombre des PDFs, les PDFs ont les mêmes noms).

Comparaison de deux répertoire avec λ -Expression

Solution λ -Expression

- 1 Décompresser le zip téléchargé, et le garder comme répertoire source.
- 2 Parcourir le répertoire de référence et mettre tous les fichiers PDF dans une liste.
- 3 En prenant chaque élément de cette liste, parcourir le répertoire source, trouver le PDF avec le même nom, puis faire la comparaison en appelant la fonction de comparaison PDF déjà développé.
- 4 Si le nombre de fichier est différent, ou les noms ne sont pas les mêmes, il y aura une Exception Java.

Comparaison entre deux répertoire avec λ -Expression

```
Path dirRef = Paths.get(refFolder);
Path dirSrc = Paths.get(srcFolder);
// filter the .pdf file
List<Path> listRefPath = Files.list(dirRef)
    .filter(f -> f.toString().endsWith(".pdf"))
    .collect(Collectors.toList());
for (int ii = 0; ii < listRefPath.size(); ii++){
    Path refPath = listRefPath.get(ii);
    Path srcPath = Files.list(dirSrc).filter(path ->
        path.getFileName().equals(refPath.getFileName()))
        .findFirst().get();
    comparePDF(refPath.toString(), srcPath.toString());
}
```

Plan

- 1 Présentation de l'entreprise et de l'équipe
 - Présentation de SAP
 - Mon rôle au sein de l'équipe
- 2 Contexte et problématique
 - Test de nouvelles fonctionnalités et test de non-régression
 - Test d'impression
- 3 Travaux réalisés
 - Architecture du projet Test-Auto
 - Développements réalisés
 - Outils et technologies utilisés
- 4 Conclusion

Outils et technologies utilisés

IDE	 eclipse			
Communication	 Skype for Business	 Outlook		
Partager	PERFORCE	 OneDrive	 OneNote	
Outils & Bibliothèques	 JUnit	 Maven™	 Jenkins	

Résultats de travail

- Les scénarios de test de la première tâche sont bien intégrés dans le projet
- Le projet de test d'impression est bien intégré dans le projet et il fonctionne bien sur le projet Test-auto

Merci pour votre attention ^ _ ^