



ComptaCom
162 bis Av. des Français Libres,
53000 Laval
+33 2 43 64 17 14



IUT Laval
Le Mans
Université

IUT de Laval
52 Rue des Docteurs Calmette et Guérin,
53000 Laval
+33 2 43 59 49 01

Réalisation de Tests Fonctionnels Automatisés

Rapport de stage de deuxième année

en vue de l'obtention du Bachelor Universitaire de Technologie (BUT)

Réalisé par **Mathys GESLIN**

Stage du 17 avril 2023 au 7 juillet 2023

Maître de stage

Fabrice TRIDON
fabrice.tridon@compta.com

Tuteur de stage

Bruno ERNET
bruno.ernet@univ-lemans.fr

1. Remerciements

Je tiens tout d'abord à remercier M. Guillaume CAPLAN et M. Gaëtan BRICHET qui ont accepté de me prendre en stage afin de me faire découvrir le monde de l'entreprise ainsi que M. Fabrice TRIDON, mon maître de stage, d'avoir pris le temps de répondre à mes questions et de m'avoir aidé à réaliser ce présent rapport.

Je remercie aussi Mme. Fédoua LAMTALSI et M. Giovanni HAQUIN qui ont pris de leur temps pour m'avoir expliqué le fonctionnement des différents outils et logiciels ainsi que d'avoir répondu à mes nombreuses questions.

Je veux aussi remercier M. Bruno ERNET, mon tuteur de stage, pour son accompagnement tout au long du stage et ses conseils apportés, ainsi que M. Ludovic HAMON pour nous avoir aidés à chercher un stage et à réaliser ce rapport.

Pour finir, je tiens à remercier l'ensemble du personnel pour leur accueil et leur bienveillance ainsi que la bonne humeur apportée chaque jour.

2. Résumé

ComptaCom est une société de comptabilité créée en 1988 à Laval. Pour simplifier le travail des entrepreneurs, le logiciel Compta.com a été mis en place en 1993 pour pouvoir effectuer la comptabilité à distance. Aujourd'hui, l'entreprise souhaite remettre aux goûts du jour le logiciel en le renommant et en faisant une refonte graphique. C'est dans ce contexte que j'ai rejoint ComptaCom pour y effectuer un stage afin de réaliser des tests fonctionnels automatisés (TFA) sur la nouvelle interface pour s'assurer que toutes les actions fonctionnent (pages non-manquantes, actions réalisables...). Ma mission a été de reprendre les tests existants afin de les migrer sur le nouveau logiciel tout en les optimisant. Il reste encore quelques tests qui ne sont pas encore réalisés.

3. Mots-clés

- Robot Framework
- Comptabilité
- Tests Fonctionnels Automatisés
- Web

4. Tables

4.1. Table des matières

1. Remerciements	2
2. Résumé	2
3. Mots-clés	2
4. Tables	3
4.1. Table des matières	3
4.2. Table des figures	4
4.3. Table des tableaux	5
5. Introduction	5
6. Présentation de l'entreprise	6
6.1. Historique	6
6.2. Cœur de métier	9
6.3. Organisation et structure	9
6.3.1. L'entreprise	9
6.3.2. Le service informatique	9
6.4. Services et produits	10
6.5. Présentation détaillée du service	11
7. Contexte et sujet de stage	11
7.1. Contexte	11
7.2. Sujet de stage	12
7.3. Méthodes de travail	12
7.4. Outils Utilisés	13
7.5. Existant à faire évoluer	13
8. Travaux et aspects techniques	13
8.1. Découverte de début de stage	13
8.2. Qu'est-ce que Robot Framework ?	14
8.3. Ma mission : concevoir des tests fonctionnels automatisés	16
8.3.1. Outils et méthodes d'organisation	17
8.3.2. Analyse et conception	19
8.3.3. Travaux réalisés	19
8.4. Difficultés rencontrées	24
9. Apports du stage	24
9.1. Connaissances et compétences acquises pendant le stage	24
9.2. Mise en relation des travaux avec les compétences du BUT	25
9.2.1. Compétence 1 : Réaliser un développement d'application	25
9.2.2. Compétence 2 : Optimiser des applications	25

9.2.3. Compétence 3 : Administrer des systèmes informatiques communicants complexes	26
9.2.4. Compétence 4 : Gérer des données de l'information	27
9.2.5. Compétence 5 : Conduire un projet	27
9.2.6. Compétence 6 : Collaborer au sein d'une équipe informatique	28
10. Conclusion	28
11. Annexes	29

4.2. Table des figures

Figure 1 - D. CAPLAN et C. CACHEUX dans les locaux d'Ouest France	6
Figure 2 - Carte répertoriant les croissances externes de ComptaCom en France	7
Figure 3 - Ancien et nouveau logo de l'entreprise	8
Figure 4 - Ancien siège social	8
Figure 5 - Nouveau siège social	8
Figure 6 - Service informatique de ComptaCom en forme de pyramide	9
Figure 7 - Organigramme du service informatique	10
Figure 8 - Partie Note de frais sur l'ancienne interface du logiciel	14
Figure 9 - Capture d'écran de la structure de Robot Framework	15
Figure 10 - Capture d'écran de la structure d'un fichier ".resource"	16
Figure 11 - Interface de TestRail avec le détail d'un scénario	17
Figure 12 - Historique du statut d'un test associé à un scénario	17
Figure 13 - Capture d'écran d'un ticket contenant des sous-tâches à réaliser	18
Figure 14 - Capture d'écran d'une erreur affichée en console	19
Figure 15 - Capture d'écran d'un fichier log après une exécution échouant	19
Figure 16 - Capture d'écran de la page d'accueil de Bleez	20
Figure 17 - Organisation des fichiers Tests	20
Figure 18 - Fichiers Ressources	21
Figure 19 - Fichiers de variables	21
Figure 20 - Capture d'écran d'une action "Click Element" avant changement	21
Figure 21 - Capture d'écran d'une action "Click Element" après changement	21
Figure 22 - Capture d'écran du mot-clé permettant de savoir si un texte est présent dans une colonne d'un tableau	22
Figure 23 - Capture d'écran du mot-clé permettant de cliquer sur un bouton dans un tableau	22
Figure 24 - Page d'emprunts du module Trésorerie	23
Figure 25 - Capture d'écran d'un temps de pause	23
Figure 26 - Capture d'écran d'un mot-clé permettant de répéter une action jusqu'à ce qu'elle fonctionne	23
Figure 27 - Capture d'écran d'un test passant	23

4.3. Table des tableaux

Tableau 1 - Relation compétences du BUT et travaux réalisés (Compétence 1)	25
Tableau 2 - Relation compétences du BUT et travaux réalisés (Compétence 2)	26
Tableau 3 - Relation compétences du BUT et travaux réalisés (Compétence 3)	26
Tableau 4 - Relation compétences du BUT et travaux réalisés (Compétence 4)	27
Tableau 5 - Relation compétences du BUT et travaux réalisés (Compétence 5)	27
Tableau 6 - Relation compétences du BUT et travaux réalisés (Compétence 6)	28

5. Introduction

Le travail décrit dans la suite de ce rapport a été réalisé dans le cadre d'un stage de deuxième année du Bachelor Universitaire de Technologie (BUT) Informatique à Laval.

Ce stage a été effectué au sein du service informatique de ComptaCom. Celui-ci avait pour but de mettre en pratique les connaissances et compétences accumulées au cours des deux années passées dans la formation, mais aussi d'apprendre de nouvelles méthodes de travail, voir le fonctionnement organisationnel d'une entreprise. Le stage avait aussi pour but de venir en aide à l'entreprise.

ComptaCom est une entreprise de comptabilité qui compte plus de 20 000 clients. Les clients, qui sont des entreprises, ont besoin d'outils simples afin de maîtriser leur comptabilité facilement. C'est pourquoi ComptaCom a créé le logiciel Compta.com qui, aujourd'hui, change de nom pour devenir Bleez profitant de cet événement pour faire une refonte du logiciel. Suite à cette refonte, il a été nécessaire de réaliser des tests fonctionnels automatisés (TFA) afin de s'assurer que les modifications apportées n'ont pas rendu certaines fonctionnalités inutilisables. C'est face à ce besoin que mon stage a eu lieu.

Mon objectif était donc de réaliser des TFA sur la nouvelle interface du logiciel à partir de scénarios donnés.

La première semaine, j'ai réalisé l'installation de mon environnement de travail puis j'ai commencé à apprendre à utiliser le langage de test Robot Framework. Pour cela, on m'a donné un scénario à suivre afin de réaliser ce premier test sur l'ancienne interface. À partir de la deuxième semaine, j'ai terminé la phase d'apprentissage du langage et du fonctionnement du logiciel puis je suis

passé sur la nouvelle interface où on m'a donné tous les scénarios sur lesquels je devais me baser pour faire les différents tests.

Dans la suite de ce rapport, vous trouverez, en section 6, une présentation de l'entreprise puis en section 7, le sujet du stage avec son contexte et son objectif. En section 8, vous trouverez la mission réalisée suivie de la section 9 contenant ce que m'a apporté le stage que ce soit au niveau des compétences acquises qu'au niveau des nouvelles méthodes de travail. Enfin, en section 10, vous pourrez retrouver la conclusion de ce rapport.

6. Présentation de l'entreprise

6.1. Historique

ComptaCom, anciennement Compta Expert, a été fondée à Laval le 14 octobre 1988 par deux jeunes experts-comptables : Christian CACHEUX et Didier CAPLAN (Figure 1).



Figure 1 - D. CAPLAN et C. CACHEUX dans les locaux d'Ouest France

Les clients de Compta Expert sont des entrepreneurs et des personnes travaillant dans les professions libérales. L'entreprise acquiert aussi les parts de la société GECAGRI qui est une filiale de la SADECO, spécialisée dès l'origine dans le conseil et l'expertise comptable des exploitations agricoles. Cette société, animée depuis sa création en 1983 par Christian CACHEUX, a acquis très rapidement une excellente notoriété et un fort développement.

En 1993, l'entreprise eut l'idée de la comptabilité à distance. Elle a donc créé COMPTACOM, pour la comptabilité à distance et ARCHIMAINE, pour la numérisation et l'archivage électronique. À partir de ce moment, l'entreprise place l'innovation technologique au cœur de sa stratégie.

Au fil du temps, le logiciel évolue et devient le premier logiciel comptable en ayant intégré la numérisation des documents, la génération automatique d'écritures par reconnaissance optique (qui permet de convertir des documents numériques en fichiers numériques éditables) ou encore le classement dans une Gestion Électronique des Documents (GED).

L'année 2000 fut marquée par des récompenses. ComptaCom reçut le prix du meilleur logiciel comptable : la disquette d'OR (aujourd'hui trophée Eureka) ainsi que le trophée de l'innovation remis par la compagnie des Commissaires aux Comptes, qui est une instance représentative de la profession en France¹.

En 2003, l'entreprise voit sa première croissance externe s'ouvrir à Vitré (Ille-et-Vilaine). Les croissances externes se sont accumulées jusqu'à en avoir, aujourd'hui, plus d'une quarantaine (Figure 2).

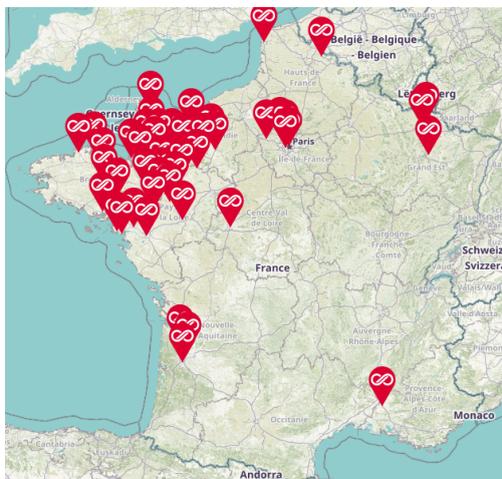


Figure 2 - Carte répertoriant les croissances externes de ComptaCom en France

C'est en 2004 que le logiciel entre dans le cloud et devient Compta.com. Ce changement sera suivi par un autre en 2007. En effet, Compta.com devient une filiale de Compta Expert. Afin d'apporter plus de cohésion et de simplicité, le nom de la filiale utilise le nom de domaine (adresse unique permettant d'identifier un site web autrement que par son IP) du logiciel. Ce nom est utilisé dans tous les cabinets d'expertise comptable.

¹ Définition provenant de : https://fr.wikipedia.org/wiki/Compagnie_nationale_des_commissaires_aux_comptes

En 2015, Compta Expert change de nom et devient ComptaCom (Figure 3) en créant la franchise du même nom. Il s'agira de la toute première franchise du monde de l'expertise comptable. Ce changement de nom s'accompagne d'un nouveau siège social (Figure 5) situé à 100 mètres du siège historique (Figure 4).



Figure 3 - Ancien et nouveau logo de l'entreprise



Figure 4 - Ancien siège social



Figure 5 - Nouveau siège social

En 2017, ComptaCom eut l'idée de créer une école basée sur un enseignement en alternance donnant sur le diplôme de comptabilité et de gestion après avoir rencontré des difficultés à recruter des jeunes. Suite à cette initiative, M. GENDRON s'est vu remettre le Trophée d'or pour la catégorie marketing RH.

L'année 2019 sera marquée par l'inauguration d'un nouveau bâtiment : la pyramide Compta.com (Figure 6). Elle est située juste à côté du siège historique et regroupe les services informatiques du siège.



Figure 6 - Service informatique de ComptaCom en forme de pyramide

6.2. Cœur de métier

Le cœur de métier de ComptaCom est principalement l'accompagnement de la gestion des entreprises et professions libérales. La mission des experts-comptables est d'aider les chefs d'entreprises dès la création de leur entreprise, en proposant une expertise comptable, fiscale, sociale et juridique. Ils apportent aussi des conseils et des outils pour faire avancer les projets professionnels des sociétés. La prise en charge des clients se fait quel que soit le domaine d'activité, que ce soient des entreprises industrielles, commerciales, artisanales et bien d'autres.

6.3. Organisation et structure

6.3.1. L'entreprise

ComptaCom est actuellement présidé par le créateur de l'entreprise : Didier CAPLAN.

Au niveau de sa structure, ComptaCom détient son siège social à Laval. L'entreprise possède un pôle investissement sur le lieu de l'ancien siège social et le service informatique juste à côté. Il existe aussi plus d'une quarantaine de cabinets d'expertise comptable franchisés à travers toute la France.

Au total, l'entreprise travaille avec plus de 30 associés et près de 500 collaborateurs.

6.3.2. Le service informatique

Le service informatique est dirigé par Guillaume CAPLAN en tant que Directeur Digital. Ce service est découpé entre plusieurs équipes. Vous trouverez

tous les membres du service dans chaque équipe ci-dessous (Figure 7). De mon côté, j'ai intégré l'équipe Business qui a pour but la fidélisation du client final et l'apport de services complémentaires.

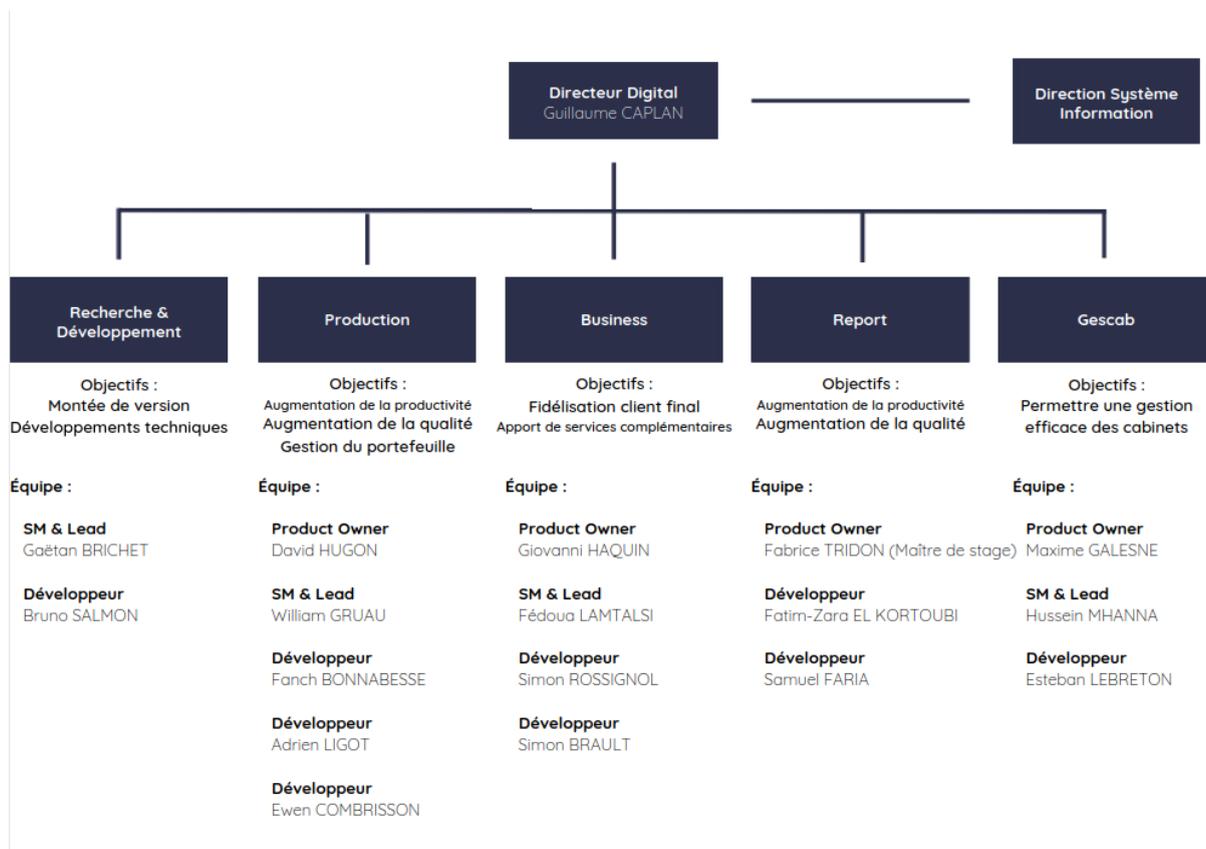


Figure 7 - Organigramme du service informatique

6.4. Services et produits

ComptaCom propose différents services et produits. On peut retrouver de l'expertise comptable, de l'expertise juridique, de l'expertise sociale et différents outils. Chacun de ces services se décompose en plusieurs sous-services, permettant aux entreprises d'avoir un accompagnement profond.

Expertise comptable :

- Saisie comptable : pour écrire manuellement les transactions effectuées sur les comptes de l'entreprise
- Bilan comptable
- Déclarations fiscales
- Prévisionnel : pour étudier le chiffre d'affaires, les charges et la trésorerie sur 3 ans

Expertise juridique :

- Constitution de société
- Secrétariat juridique annuel
- Modification statutaire
- Contrat et convention

Expertise sociale :

ComptaCom propose d'externaliser la gestion sociale des entreprises ou bien d'utiliser un logiciel permettant de simplifier le travail du quotidien.

Comme indiqué précédemment, ComptaCom propose aussi un logiciel de comptabilité et de facturation : Bleez (anciennement nommé Compta.com) ainsi qu'un logiciel de gestion de cabinet, Gescab, réservé aux cabinets comptables franchisés.

6.5. Présentation détaillée du service

J'ai réalisé mon stage dans le service informatique de l'entreprise. Le but de ce service est de réaliser et de maintenir les logiciels Bleez et Gescab. Le service possède aussi une équipe chargée de fournir aux franchises du matériel et logiciels informatiques. Cette équipe est aussi chargée de réparer le matériel.

Le service travaille aussi avec SII Service Maroc, un prestataire de service situé à Casablanca.

7. Contexte et sujet de stage

7.1. Contexte

ComptaCom a mis en place un logiciel de comptabilité et de facturation nommé Compta.com. Aujourd'hui, ce logiciel utilisé par les entreprises, est aussi utilisé par des cabinets comptables concurrents qui proposaient à leurs clients de l'utiliser. Étant donné que le nom du logiciel est très proche du nom de l'entreprise, le produit proposé affichait donc un nom de cabinet concurrent ce qui posait des problèmes de compréhension. Pour pallier cela, ComptaCom a donc décidé de renommer Compta.com en Bleez afin que ces cabinets ainsi que

leurs clients puissent utiliser ce logiciel sans se soucier du nom. Dans le même temps, il a été décidé de faire une refonte graphique pour plus de cohérence.

7.2. Sujet de stage

Le sujet de mon stage était donc de réaliser des tests fonctionnels automatisés sur la nouvelle interface. Mon but était de reprendre des tests déjà existants et fonctionnels sur l'ancienne interface, et de les adapter en fonction des nouveaux éléments présents sur les pages web ainsi que de les optimiser pour qu'ils prennent le moins de temps possible à s'exécuter.

7.3. Méthodes de travail

Du côté de l'organisation, le service informatique a mis en place un système de sprint qui dure trois semaines. Un sprint est une période de temps pendant laquelle sont définies, pour chacun, les différentes tâches à effectuer pendant cette période. Les tâches sont décrites à travers des tickets. Ils permettent de saisir des demandes (modifications, corrections de bugs, tâches à réaliser...), indiquer la priorité des tâches, mettre des commentaires sur ce qui va ou non pour apporter des suggestions de modifications. Ils permettent aussi de voir l'état des tâches, c'est-à-dire, si elles sont à faire, en cours, à tester, etc. Ils facilitent la communication entre les membres d'une équipe et entre les équipes.

À la fin de chaque sprint, chaque équipe réalise une revue de sprint afin de faire une démonstration de ce qui a été réalisé pendant la période. Cette revue de sprint est suivie par une rétrospective. Elle permet de voir ce qui a bien fonctionné ou encore les problèmes qu'ont rencontrés les membres de l'équipe pendant le sprint afin de trouver des solutions pour rendre le travail plus efficace. Aussi, chaque matin se tient une petite réunion de 10 minutes environ, se nommant Daily Meeting (DM), permettant de décrire ce que chaque membre de l'équipe a fait la veille, les problèmes rencontrés ainsi que d'indiquer les tâches qui seront réalisées dans la journée.

Pour l'organisation du code, tout le monde utilise GitLab. GitLab permet de regrouper le code de chaque membre, le code est stocké sur un serveur. Afin de ne pas écraser ce que chacun fait, chaque personne se voit attribuer une branche créée à partir d'un ticket. Une branche permet de faire une copie du projet sans risque d'une perte de donnée due à la modification du code par une

autre personne. La branche utilisée se retrouve, en local, sur l'ordinateur de la personne qui modifie celle-ci. Une fois que la personne a terminé sa tâche, elle pousse sa branche sur le serveur en mettant dans un message les modifications qui ont été apportées. Ensuite, pour que les modifications soient apportées au projet global, le développeur fait une "merge request" afin de proposer au Product Owner (PO) de fusionner la branche du développeur avec le projet.

7.4. Outils Utilisés

Dès mon arrivée, il m'a été confié un ordinateur portable sur lequel j'ai dû installer les bibliothèques liées à Robot Framework. On m'a aussi donné un compte pour que je puisse communiquer via les services de Microsoft tels que Teams et Outlook.

Comme indiqué précédemment, j'utilise GitLab pour organiser mon code et un système de tickets nommé Jira pour organiser les différentes tâches à effectuer. J'utilise aussi l'IDE IntelliJ pour coder les cas de tests ainsi que TestRail sur lequel je peux voir le scénario à suivre pour chaque test.

7.5. Existant à faire évoluer

Avant mon arrivée, il existait déjà des cas de tests codés avec Robot Framework qui fonctionnaient sur l'ancienne interface du logiciel Compta.com. Aujourd'hui, mon but est d'adapter et d'optimiser le code pour qu'il fonctionne avec la nouvelle interface de Bleez et éventuellement créer de nouveaux tests en suivant les scénarios indiqués dans TestRail.

8. Travaux et aspects techniques

8.1. Découverte de début de stage

Le stage a débuté par la mise en place de mon environnement de travail puis, pendant une semaine, j'ai travaillé sur la partie Note de frais de l'ancienne interface du logiciel (Figure 8) afin de prendre la main sur Robot Framework, sur le fonctionnement du logiciel et sur la manière dont se réalisent les tests.

	Etape	Titre	Emetteur	Bénéficiaire	Montant TTC	Depuis	
	Saisie		imane gharbi			05/05/23	<input type="checkbox"/>
	Saisie		imane gharbi			05/05/23	<input type="checkbox"/>

Figure 8 - Partie Note de frais sur l'ancienne interface du logiciel

Par exemple, le premier test consistait à créer une nouvelle note de frais et vérifier que les informations saisies correspondent aux informations inscrites dans le tableau ci-dessus.

8.2. Qu'est-ce que Robot Framework ?

Tout d'abord, pour reprendre ce qui a été dit plus tôt, Robot Framework n'est pas tout à fait un langage, mais plutôt un framework (une sorte de squelette) de test automatique en Python (langage de programmation). Il permet donc de créer des cas de tests qui sont séparés en plusieurs parties comme illustrées dans la figure 9. Les cas de tests sont décrits dans des fichiers ".robot".

```
AjoutPieceJointe.robot x CreerEmpruntSimple.robot x Emprunt.resource x StatusCompteBancaireBI.robot x MenuUI.resource x
1  *** Settings ***
2  Documentation    Test du ticket CCR-5428
3  Resource        ../../../../Ressources/Login.resource
4  Resource        ../../../../Ressources/Pages/Banque/Emprunt.resource
5  Resource        ../../../../Variables/ProjectVariables.resource
6  Test Teardown    Send results to TestRail    ${PROFILE}    ${CASES}
7  *** Variables ***
8  @{CASES}        371    372
9  ${DOSSIER}      Testrail Automate
10 ${CNX}          Revision
11 ${NUMERO_EMPRUNT}  8
12 ${COMPTE_EMPRUNT} 1650371 - TFA C371
13 *** Test Cases ***
14 Je réussis à créer puis à supprimer un emprunt
15     [Tags]      tag=beta
16     Given Log in by style    ${CNX}    ${URL}
17     And Open folder with argument    ${DOSSIER}
18     And Verify click Banque > Emprunts
19     And Delete emprunt with number    ${NUMERO_EMPRUNT}
20     ${NUMERO_EMPRUNT}=    Then Create Emprunt    no=${NUMERO_EMPRUNT}    compteEmprunt=${COMPTE_EMPRUNT}
21     And Delete emprunt with number    ${NUMERO_EMPRUNT}
```

Figure 9 - Capture d'écran de la structure de Robot Framework

La première partie, se nommant “***Settings***”, permet d’indiquer les fichiers nécessaires à la réalisation du cas de test comme, par exemple, les fichiers stockant les variables, les fichiers stockant les mots-clés ou encore de la documentation.

La deuxième partie, se nommant “***Variables***”, permet d’indiquer les variables ainsi que leur valeur nécessaire et uniquement utilisables dans ce cas de test.

Enfin, la troisième partie, “***Test Cases***”, permet de décrire le cas de test. Cette partie doit être claire pour qu’elle soit lisible par n’importe qui, même par quelqu’un qui n’a jamais codé. On y retrouve d’abord le nom ou la description de ce que fait le test, puis on a le scénario du test. Il doit commencer par “Given” afin d’indiquer les actions pré-requises. Les actions sont identifiables par leur couleur jaune et s’appellent des mots-clés. Une fois que les pré-requis sont donnés, on indique par le mot “When” la première action permettant d’accéder au résultat attendu. Lorsque toutes ces actions sont effectuées, on inscrit le terme “Then” afin d’indiquer l’action permettant de comparer le résultat obtenu avec le résultat attendu.

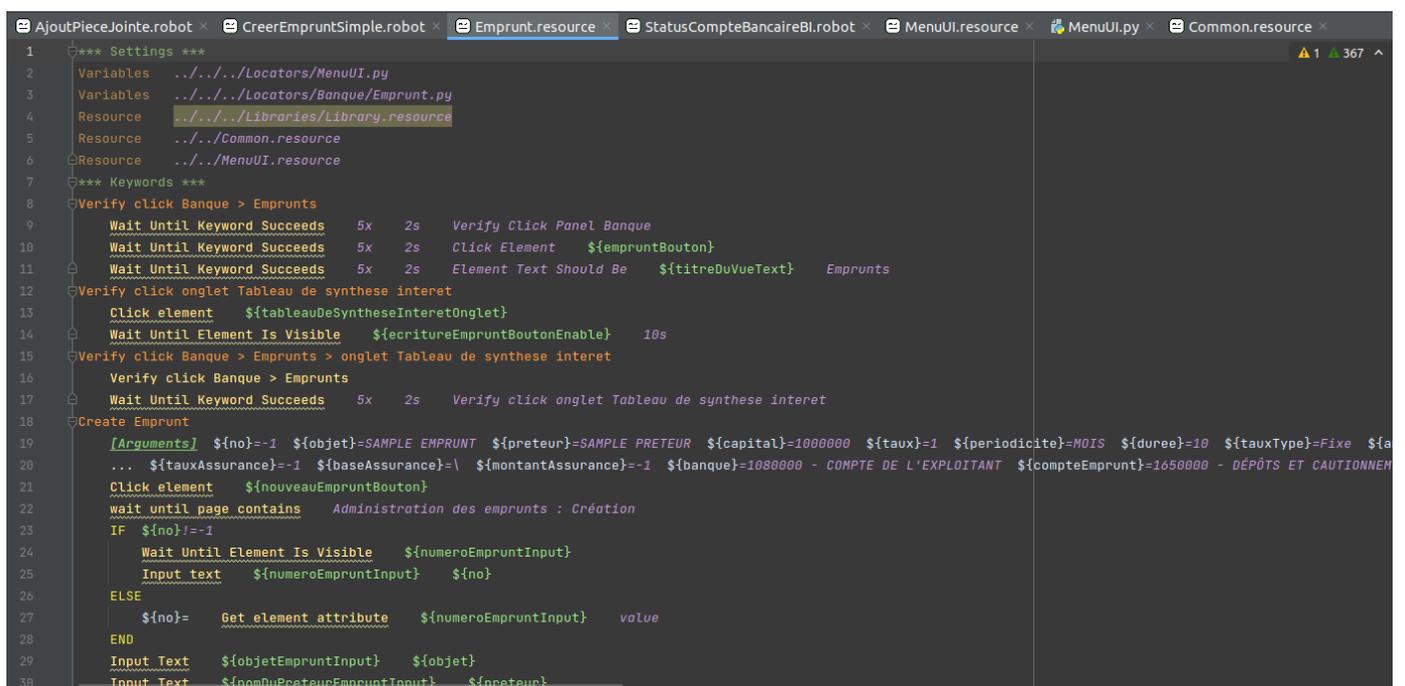
Comment sont définis les mots-clés ? Ils sont définis dans des fichiers “.resource”. Ces fichiers ont une structure semblable à celle des fichiers de cas de test. Ils possèdent une partie “***Settings***” et certaines fois une partie “***Variables***” mais aussi une partie “***Keywords***” dans laquelle sont définis

les mots-clés comme illustrés dans la figure 10 . Les mots-clés sont décrits de la manière suivante, ils possèdent :

- un nom
- des arguments (non-obligatoire)
- une documentation (non-obligatoire)
- un corps dans lequel sont décrits les instructions que doit faire le mot-clé
- et peut retourner une ou plusieurs valeurs (non-obligatoire)

Les mots-clés peuvent utiliser d'autres mots-clés et ainsi de suite.

La plupart des variables sont définies dans des fichiers “.py”.



```
1 *** Settings ***
2 Variables    ../../Locators/MenuUI.py
3 Variables    ../../Locators/Banque/Emprunt.py
4 Resource     ../../Libraries/Library.resource
5 Resource     ../../Common.resource
6 Resource     ../../MenuUI.resource
7 *** Keywords ***
8 Verify click Banque > Emprunts
9     Wait Until Keyword Succeeds    5x    2s    Verify Click Panel Banque
10    Wait Until Keyword Succeeds    5x    2s    Click Element    ${empruntBouton}
11    Wait Until Keyword Succeeds    5x    2s    Element Text Should Be    ${titreDuVueText}    Emprunts
12 Verify click onglet Tableau de synthese interet
13 Click element    ${tableauDeSyntheseInteretOnglet}
14 Wait Until Element Is Visible    ${ecritureEmpruntBoutonEnable}    10s
15 Verify click Banque > Emprunts > onglet Tableau de synthese interet
16 Verify click Banque > Emprunts
17 Wait Until Keyword Succeeds    5x    2s    Verify click onglet Tableau de synthese interet
18 Create Emprunt
19     [Arguments]    ${no}=-1    ${objet}=SAMPLE EMPRUNT    ${preteur}=SAMPLE PRETEUR    ${capital}=1000000    ${taux}=1    ${periodecrite}=MOIS    ${duree}=10    ${tauxType}=Fixe    ${a
20     ...    ${tauxAssurance}=-1    ${baseAssurance}=\    ${montantAssurance}=-1    ${banque}=1000000 - COMPTE DE L'EXPLOITANT    ${compteEmprunt}=1650000 - DÉPÔTS ET CAUTIONNEM
21 Click element    ${nouveauEmpruntBouton}
22 wait until page contains    Administration des emprunts : Création
23 IF    ${no}!=1
24     Wait Until Element Is Visible    ${numeroEmpruntInput}
25     Input text    ${numeroEmpruntInput}    ${no}
26 ELSE
27     ${no}=    Get element attribute    ${numeroEmpruntInput}    value
28 END
29 Input Text    ${objetEmpruntInput}    ${objet}
30 Input Text    ${nomDuPreteurEmpruntInput}    ${preteur}
```

Figure 10 - Capture d'écran de la structure d'un fichier “.resource”

8.3. Ma mission : concevoir des tests fonctionnels automatisés

Directement après la découverte, ma mission était de concevoir des tests fonctionnels automatisés afin de faire la migration vers le logiciel Bleez.

8.3.1. Outils et méthodes d'organisation

Pour réaliser ces tests, je me suis basé sur ceux qui existaient déjà et qui fonctionnaient sur le logiciel Compta.com ainsi que sur un logiciel nommé TestRail sur lequel je pouvais retrouver tous les scénarios à suivre (Figure 11).

The screenshot displays the TestRail interface. On the left, a table lists various test scenarios under the category 'Banque'. The table columns include ID, Title, Tested On, Priority, and Status. The status column shows 'Failed' (red) or 'Passed' (green) with a dropdown arrow. The main part of the interface shows the details for scenario T141, 'Saisie comptable avec des actions banque automatique'. This detail view includes fields for Type, Priority, Estimate, and References. Below these are sections for 'Preconditions' and 'Steps'. The 'Steps' section contains three numbered steps, each with a 'Step description' and an 'Expected Result' field. A sidebar on the right shows a navigation menu with categories like 'Accueil', 'Facturation', 'Banque', 'Emprunt', 'Compta', 'Documents', 'Paramétrage', 'Gestion', 'API', 'Social', and 'Note de frais'.

ID	Title	Tested On	Priority	Status
T141	Saisie comptable avec des actions banque automatique	08/06/2023	High	Failed
T142	Générer les relevés	08/06/2023	Critical	Failed
T144	Vérifier la qualité et prénom du responsable du dossier sur le mandat EBICS	08/06/2023	Critical	Passed
T145	Relance par mail sur les factures impayées détectées depuis l'échéancier	08/06/2023	Critical	Passed
T146	Créer un compte bancaire avec un connecteur BI	08/06/2023	High	Retest
T147	Activation, Désactivation et Réactivation "Via EBICS"	08/06/2023	High	Failed
T148	Suppression de relevé API	09/06/2023	High	Failed
T149	Saisie comptable avec le mode de comptabilisation : Associé à une écriture	08/06/2023	Critical	Failed
T150	Saisie comptable avec le mode de comptabilisation : Rapproché avec facturé(s)	08/06/2023	Critical	Failed
T151	Dédoublonner les libellés lors de l'import BI	09/06/2023	Medium	Passed
T831	Enregistrer les règlements clients	08/06/2023	Critical	Passed
T879	Créer une règle de banque automatique	08/06/2023	Medium	Passed
T1095	Encaissement des factures par prélèvement SEPA	08/06/2023	Critical	Failed
T1443	Saisie comptable avec le mode de comptabilisation : Mise en compte d'attente	08/06/2023	Critical	Failed
T1463	Réaliser la remise de chèques d'un règlement client	08/06/2023	High	Failed
T2746	Relance par courrier sur les factures impayées détectées depuis l'échéancier	08/06/2023	High	Passed
T3835	Réaliser une remise de chèques à travers l'interface	08/06/2023	Medium	Passed

Figure 11 - Interface de TestRail avec le détail d'un scénario

<p>Passed</p> <p>07/06/2023 23:20 Guillaume C. Edit</p>	<i>This test was marked as 'Passed'.</i>
<p>Passed</p> <p>07/06/2023 16:07 Guillaume C.</p>	<i>This test was marked as 'Passed'.</i>
<p>Passed</p> <p>06/06/2023 23:37 Guillaume C.</p>	<i>This test was marked as 'Passed'.</i>
<p>Passed</p> <p>06/06/2023 01:20 Guillaume C.</p>	<i>This test was marked as 'Passed'.</i>
<p>Failed</p> <p>06/06/2023 00:52</p>	WebDriverException: Message: Failed to decode response from marionette

Figure 12 - Historique du statut d'un test associé à un scénario

On peut y voir, sur la gauche, le titre de chaque scénario rangé dans des catégories. Ces catégories correspondent à des parties du logiciel. Au milieu, on trouve un scénario avec les préconditions puis, pour chaque étape numérotée, la description de l'action à effectuer suivie du résultat attendu. En dessous, se trouve un historique du statut du test. C'est-à-dire s'il est passant ou non (Figure 12). Enfin, à droite, on peut trouver une arborescence dans laquelle se trouve toutes les catégories des scénarios.

On retrouve le nom des scénarios sous la forme de sous-tâches dans Jira décrites via un ticket (Figure 13). Il est ainsi possible de créer une branche pour chaque sous-tâche afin de travailler dessus et de suivre le scénario associé. Une fois le travail terminé, on peut voir qui a réalisé la sous-tâche et son état, c'est-à-dire si elle est à faire, en cours, terminée ou en code review (à faire valider par le PO).

The screenshot shows a Jira ticket interface for 'Compta.com - Business / CCMS-7816' with the title 'TFA : Banque'. Below the title are buttons for 'Modifier', 'Ajouter un commentaire', 'Saisir du temps', 'Plus', and 'Admin - Resolution'. The ticket is organized into sections: 'Informations', 'Description', 'Pièces jointes', and 'Sous-tâches'. The 'Sous-tâches' section contains a list of 8 tasks with their respective status, assignee, and progress bar.

Numéro	Description	Statut	Assigné à	Progression
1.	Saisie comptable avec des actions banque automatique	A FAIRE	Non attribuée	
2.	Générer les relevés	FINI	Soufiane IHHOUF	96%
3.	Vérifier la qualité et prénom du responsable du dossier sur le mandat EBICS	FINI	Soufiane IHHOUF	95%
4.	Relance par mail sur les factures impayées détectées depuis l'échéancier	CODE REVIEW	Mathys GESLIN	100%
5.	Créer un compte bancaire avec un connecteur BI	A FAIRE	Non attribuée	
6.	Activation, Désactivation et Réactivation "Via EBICS"	A FAIRE	Non attribuée	
7.	Suppression de relevé API	A FAIRE	Non attribuée	
8.	Saisie comptable avec le mode de comptabilisation : Associé à une écriture	A FAIRE	Non attribuée	100%

Figure 13 - Capture d'écran d'un ticket contenant des sous-tâches à réaliser

J'ai aussi à ma disposition plusieurs outils permettant de savoir ce qui a fait échouer un test. Je peux directement voir l'erreur en console (Figure 14), ou bien avoir une vue plus détaillée du déroulement du scénario à travers les mots-clés et ainsi voir l'endroit exact de l'erreur grâce à un fichier de log (Figure 15).

```

=====
testsAncienneInterface | FAIL |
1 test, 0 passed, 1 failed
=====

```

Figure 14 - Capture d'écran d'une erreur affichée en console

```

- [TEST] Je réussis à vérifier le formulaires par défaut et ajouter des formulaires à une déclaration de TVA puis les supprimer 00:00:19.400
Full Name: testsAncienneInterface.Tests.Compta.DeclarationTVA.AjouterSupprimerFormulaire.Je réussis à vérifier le formulaires par défaut et ajouter des formulaires à une déclaration de TVA puis les supprimer
Tags: tag=beta
Start / End / Elapsed: 20230612 16:09:10.316 / 20230612 16:09:29.716 / 00:00:19.400
Status: FAIL
Message: != "Dumb note for testing"

+ [KEYWORD] $(CNX) = Builtin.Run Keyword If '$(PROFILE)'=='BETA' or '$(PROFILE)'=='MAINTENANCE'; Set Variable, ComptaComBeta, ELSE, Set Variable, Revision 00:00:00.000
+ [KEYWORD] Login.Given Log in by style $(CNX), $(URL) 00:00:07.094
+ [KEYWORD] MenuUtil.And Open folder with argument $(dossier) 00:00:02.883
+ [KEYWORD] DeclarationTVA.And Verify click Compta > Saisies > Declaration TVA 00:00:01.810
+ [KEYWORD] Builtin.And run keyword and return status Delete editable 00:00:00.121
- [KEYWORD] DeclarationTVA.Then Create dumb declaration 00:00:04.776
Start / End / Elapsed: 20230612 16:09:22.227 / 20230612 16:09:27.003 / 00:00:04.776
+ [KEYWORD] Common.Verify And Click On $(nouvelleDeclarationBouton) 00:00:04.579
+ [KEYWORD] Selenium2Library.Wait Until Page Contains Etape 1 - Identification 00:00:00.047
+ [KEYWORD] $(checkAccept) = Builtin.Run Keyword And Return Status page should contain element, $(accepteDeclarationBouton) 00:00:00.016
+ [IF] $(checkAccept)==False 00:00:00.000
- [KEYWORD] Common.Verify Clear And Write In $(notePourLaCorrespondanceDeclarationInput), "Dumb note for testing" 00:00:00.134
Documentation: Vérifie qu'un champ de texte donné en paramètre existe, le vide pour le remplir avec une donnée entrée en paramètre
Start / End / Elapsed: 20230612 16:09:26.869 / 20230612 16:09:27.003 / 00:00:00.134
+ [KEYWORD] Selenium2Library.Wait Until Element Is Visible $(elem) 00:00:00.059
+ [KEYWORD] Selenium2Library.Clear Element Text $(elem) 00:00:00.010
+ [KEYWORD] Selenium2Library.Input Text $(elem), $(data) 00:00:00.049
- [KEYWORD] Selenium2Library.Get Value $(elem) 00:00:00.014
Documentation: Returns the value attribute of the element identified by locator.
Start / End / Elapsed: 20230612 16:09:26.988 / 20230612 16:09:27.002 / 00:00:00.014
16:09:27.002 [INFO] $(getData) =
- [KEYWORD] Builtin.Should Be Equal As Strings $(getData), $(data) 00:00:00.000
Documentation: Fails if objects are unequal after converting them to strings.
Start / End / Elapsed: 20230612 16:09:27.002 / 20230612 16:09:27.002 / 00:00:00.000
16:09:27.002 [FAIL] != "Dumb note for testing"

```

Figure 15 - Capture d'écran d'un fichier log après une exécution échouant

8.3.2. Analyse et conception

Mon travail d'analyse a été d'aller voir le PO de mon équipe afin de lui demander des précisions sur les différents scénarios lorsque je ne les comprenais pas. J'ai aussi eu l'occasion de modifier certains scénarios qui ne correspondaient pas ou plus forcément par rapport à ce que le logiciel propose de faire.

8.3.3. Travaux réalisés

J'ai principalement réalisé mes tests sur les parties Administration et Trésorerie de Bleez. Le but de ces tests est de voir si les interactions entre les différents modules (Facturation, Trésorerie, Administration...) fonctionnent toujours même lorsque des modifications sont apportées par les développeurs étant donné que chacun de ces modules est dépendant les uns des autres.

Page d'accueil de Bleez (Figure 16) :

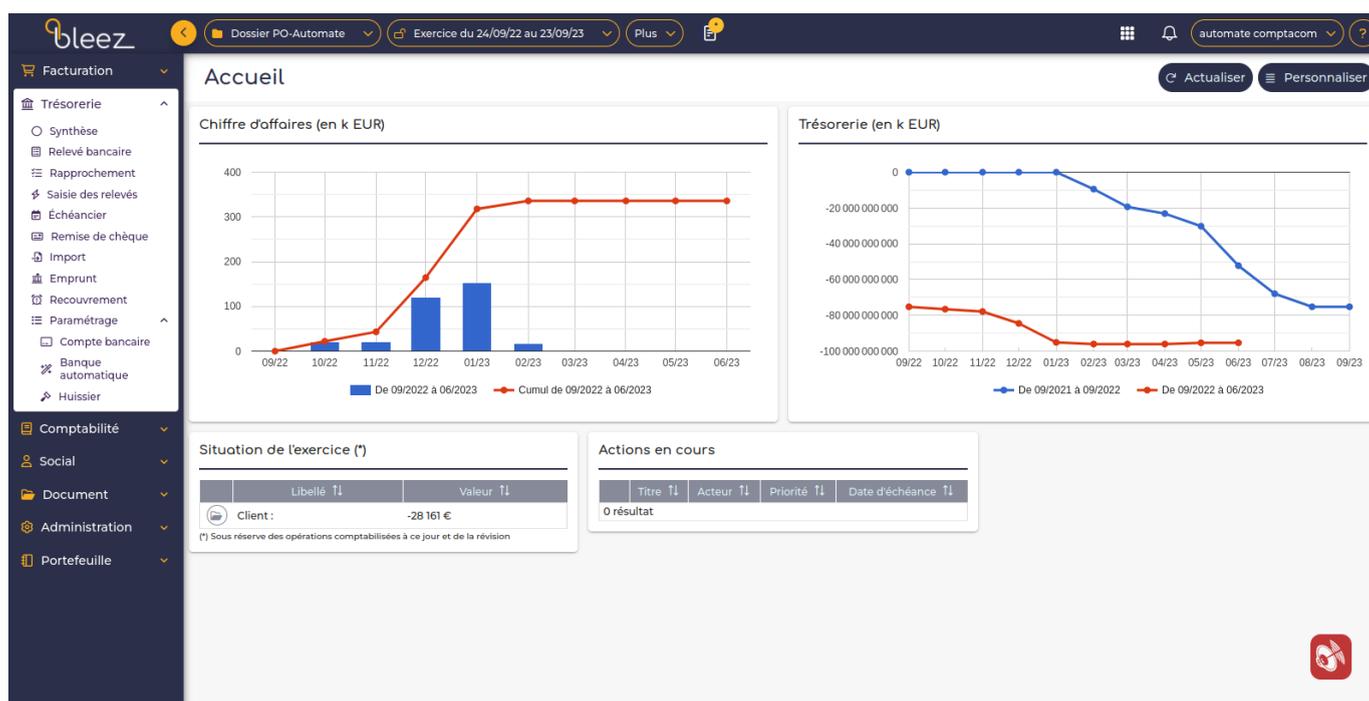


Figure 16 - Capture d'écran de la page d'accueil de Bleez

Dans mon projet, les tests (les fichiers ".robot") sont organisés en fonction de ce que chacun d'entre eux réalise. C'est-à-dire que les tests sont rangés dans le répertoire correspondant à la catégorie dans lequel le test tourne (Figure 17).

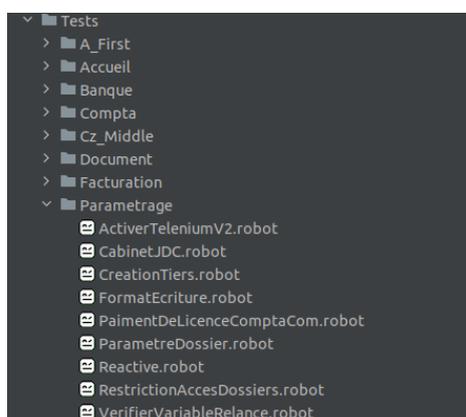


Figure 17 - Organisation des fichiers Tests

Les fichiers contenant les mots-clés (les fichiers ".resource") sont structurés de la même manière (Figure 18) tout comme les fichiers contenant les variables (Figure 19).

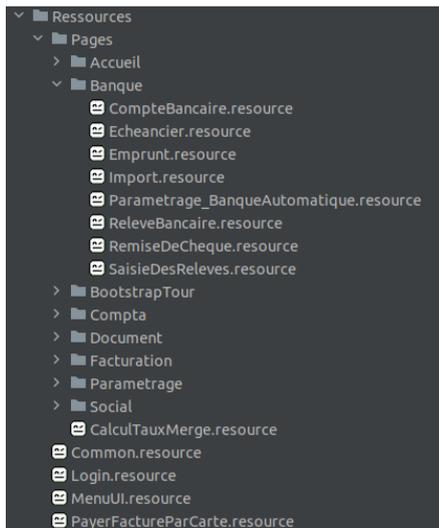


Figure 18 - Fichiers Ressources

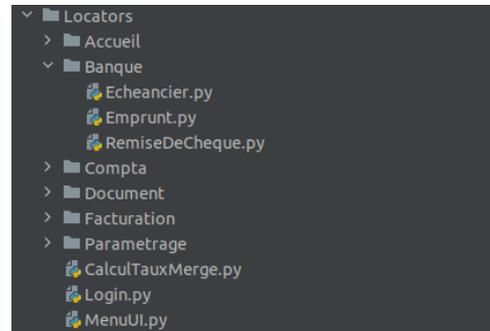


Figure 19 - Fichiers de variables

Le principal travail réalisé a été de vérifier que le scénario de chaque test existant correspondait bien à ceux indiqués dans TestRail ainsi que d'apporter des modifications sur la manière dont les éléments sont utilisés pour interagir avec. En effet, avec le changement d'interface, les éléments ne possédaient plus les mêmes propriétés ce qui faisait échouer le test. Par exemple, si je veux cliquer sur un bouton, je récupère son id et j'indique le mot-clé "Click Element" suivi de l'id du bouton en question (Figure 20 et 21). Très souvent, les id n'étaient pas dans des variables, je me suis donc occupé de le faire pour qu'ils puissent être réutilisés plus facilement et pour que le code soit plus facilement lisible.

```
Click element //*[@aria-label="Générer l'échéancier"]
```

Figure 20 - Capture d'écran d'une action "Click Element" avant changement

```
Click element ${genererEcheancierEmpruntBouton}
```

Figure 21 - Capture d'écran d'une action "Click Element" après changement

Il arrivait que certaines actions se répètent, j'ai alors créé des mots-clés regroupant ses actions. On peut alors les réutiliser autant de fois qu'on veut sans les dupliquer.

Par exemple, j'ai pu créer un mot-clé permettant de savoir si un texte est présent dans une colonne d'un tableau (Figure 22). Ce mot-clé est souvent utilisé pour vérifier que des données ont bien été insérées ou modifiées. J'ai aussi développé un mot-clé permettant de cliquer sur un bouton présent dans une certaine ligne d'un tableau à partir d'une case d'une colonne contenant un texte particulier (Figure 23). Pour illustrer ceci (Figure 24), imaginons que je souhaite modifier les informations de l'emprunt du compte comptable ayant pour numéro

1650014. Je vais donc chercher dans le tableau le numéro en regardant uniquement dans la colonne "Compte Comptable". Une fois le compte trouvé, je cherche le bouton présent sur la ligne en fonction de l'élément passé en paramètre, puis je clique dessus.

```
Find A Text In A Column In A Table
[Documentation] Cherche un texte dans une colonne d'un tableau passé en paramètre. Renvoie True ou False.
[Arguments] ${texteRecherche} ${texteColonne} ${table}
Wait Until Element Is Visible ${table}
${result} Set Variable ${False}
${columnNumber} Get Column Number Of ${texteColonne} ${table}/thead
${nbLignesTableau} Get Element Count ${table}/tbody/tr
FOR ${ligne} IN RANGE 1 ${nbLignesTableau} + 1
  ${texteCase} Get Text ${table}/tbody/tr\[${ligne}\]/td\[${columnNumber}\]
  IF '${texteRecherche}'=='${texteCase}'
    ${result} Set Variable ${True}
    BREAK
  END
END
END

[Return] ${result}
```

Figure 22 - Capture d'écran du mot-clé permettant de savoir si un texte est présent dans une colonne d'un tableau

```
Click On Button In A Line In a Table Searching a Case
[Arguments] ${texteCase} ${table} ${numCase}=-1 ${finXpassBtn}=${empty} ${typeCase}=td
[Documentation] Clique sur le bouton correspondant à une ligne spécifiée dans un tableau, qui contient une case particulière
#Recherche de la case contenant le texte en fonction du numéro de la colonne
${status} ${case} Find A Case In A Line In A Table ${texteCase} ${table} ${numCase}
Run Keyword If not ${status} Fail La case contenant le texte "${texteCase}" n'a pas été trouvée
${ligne}= Get The Line Of A Case ${case} ${typeCase}
Verify And Click On ${ligne}${finXpassBtn}
```

Figure 23 - Capture d'écran du mot-clé permettant de cliquer sur un bouton dans un tableau

Emprunts													
Ecriture Impression Import + Nouveau													
Liste Tableau de synthèse capital Tableau de synthèse intérêt Tableau prévisionnel des échéances													
#	Compte comptable	Prêteur	Objet	Capital emprunté	1er déblocage	Durée (mois)	Comptabilisé sur l'exercice	Capital théorique restant dû	Solde comptable	Ecart	Etat du solde comptable	OD éclatement	
829	1640000		ACQUISITION IMMEUBLE	400 000,00	25/05/20	180	(0/12)	400 000,00	0,00	400 000,00	✗	⊙	
4	1650004	SAMPLE PRETEUR	Emprunt non-modifiée	1 000 000,00	01/06/22	10	(0/6)	1 000 000,00	0,00	1 000 000,00	✗	⊙	
14	1650014	SAMPLE PRETEUR	Test valeurs Variables	900 000,00	01/06/22	25	(0/6)	900 000,00	0,00	900 000,00	✗	⊙	
15	1650015	SAMPLE PRETEUR	Test valeurs Variables	900 000,00	01/06/22	25	(0/6)	900 000,00	0,00	900 000,00	✗	⊙	
56	1650056	Michel Drucker	Test ajout déblocages	1 000 000,00	01/06/22	12	(0/6)	1 000 000,00	0,00	1 000 000,00	✗	⊙	
377	1650377	SAMPLE PRETEUR	Emprunt non-modifiée	1 000 000,00	01/06/22	10	(0/6)	1 000 000,00	0,00	1 000 000,00	✗	⊙	
827	1650379	Sombre Vador	Test Pointage	5 000,00	30/06/22	12	✓(6/6)	2 512,52	0,00	2 512,52	✗	⊙	
Total				5 205 000,00									

Emprunts : 7

Figure 24 - Page d'emprunts du module Trésorerie

Les tests comportaient aussi parfois de longs moments de pause pour laisser le temps au logiciel d'effectuer les actions demandées ou de charger une page (Figure 25). Il a été nécessaire d'enlever ces pauses et de les remplacer par des mots-clé permettant d'attendre qu'un certain texte ou élément apparaisse à l'écran, ou bien de faire en sorte de répéter plusieurs fois un mot-clé jusqu'à ce qu'il fonctionne (Figure 26). Cela peut paraître plus long à première vue, mais il est rare que les mots-clés nécessitent de se répéter. Il est préférable d'en mettre pour anticiper un potentiel échec d'une action.

Sleep 2s

Figure 25 - Capture d'écran d'un temps de pause

Wait Until Keyword Succeeds 4x 3 sec Click Element \${DROPDOWN_PATH}

Figure 26 - Capture d'écran d'un mot-clé permettant de répéter une action jusqu'à ce qu'elle fonctionne

Un test est terminé lorsqu'il n'échoue pas. La console affiche "PASS" (Figure 27) et aucune erreur n'est trouvée lorsqu'il est lancé sur la nouvelle interface du logiciel. Il est considéré réellement terminé lorsque le scénario est respecté et qu'on estime qu'on ne peut pas plus l'optimiser.

```
=====  
Tests | PASS |  
1 test, 1 passed, 0 failed  
=====
```

Figure 27 - Capture d'écran d'un test passant

8.4. Difficultés rencontrées

La principale difficulté a été de comprendre comment fonctionne le logiciel, la comptabilité étant un domaine complexe. Il a aussi été difficile de déterminer à quel moment un test est suffisamment optimisé.

9. Apports du stage

9.1. Connaissances et compétences acquises pendant le stage

Durant mon stage, j'ai pu acquérir de nombreuses compétences ainsi qu'appliquer celles que j'ai apprises durant ma formation du BUT Informatique.

En effet, ce stage m'a permis de découvrir Robot Framework bien que j'avais déjà eu l'occasion d'utiliser un outil similaire lors d'un projet pendant ma formation. J'ai appris à utiliser des outils de gestion de projet tels que Jira ou même TestRail. J'ai pu approfondir mes connaissances sur l'utilisation de Git que je n'osais pas utiliser avant le stage par peur de perdre le travail déjà réalisé et aussi parce que je trouvais cet outil difficile à prendre en main. Ce stage m'a fait prendre conscience de l'importance de mettre en place une convention de codage pour plus de compréhension pour moi-même ou bien les personnes qui reprendront mon travail.

De manière générale, le stage m'a permis de mieux comprendre le fonctionnement d'une équipe dans une entreprise, ou même tout simplement le fonctionnement d'une entreprise en elle-même. De plus, il m'a permis de m'ouvrir davantage et d'oser poser des questions plus souvent lorsque je faisais face à des difficultés ou que j'avais des doutes.

9.2. Mise en relation des travaux avec les compétences du BUT

9.2.1. Compétence 1 : Réaliser un développement d'application

Compétence 1 : Réaliser un développement d'application	
AC 1 : Élaborer et implémenter les spécifications fonctionnelles et non fonctionnelles	Les travaux réalisés ont été effectués selon les demandes du service informatique avec les outils donnés (Jira, TestRail...) sans utilisation de cahier des charges.
AC 2 : Appliquer des principes d'accessibilités et d'ergonomie	n/a
AC 3 : Adopter de bonne pratiques de conception et de programmation	Le code adopte la convention de nommage Gherkin et il n'existe pas de duplication de données. Le nom des mots-clés est clair et explicite par rapport à leur fonction. Le code adopte le design pattern Page Object Model soit un test par fichier.
AC 4 : Vérifier et valider la qualité de l'application par les tests	n/a

Tableau 1 - Relation compétences du BUT et travaux réalisés (Compétence 1)

9.2.2. Compétence 2 : Optimiser des applications

Compétence 2 : Optimiser des applications	
AC 1 : Choisir des structures de données complexes adaptées au problème	n/a

AC 2 : Utiliser des techniques algorithmiques adaptées pour des problèmes complexes	Dans le cadre de TFA, les techniques utilisées sont le design pattern Page Object Model ainsi l'utilisation de données non-statiques comme par exemple la recherche d'une donnée dans un tableau faite en fonction du nom de la colonne plutôt que de sa position qui peut changer.
AC 3 : Comprendre les enjeux et moyens de sécurisation des données et du code	n/a
AC 4 : Évaluer l'impact environnemental et sociétal des solutions proposées	Les tests sont réalisés dans le but de prendre le moins de temps possible à s'exécuter en optimisant la complexité du code, impliquant une moins grande consommation d'énergie.

Tableau 2 - Relation compétences du BUT et travaux réalisés (Compétence 2)

9.2.3. Compétence 3 : Administrer des systèmes informatiques communicants complexes

Compétence 3 : Administrer des systèmes informatiques communicants complexes	
AC 1 : Concevoir et développer des applications communicantes	n/a
AC 2 : Utiliser des serveurs et des services réseaux virtualisés	Le lancement des tests s'effectuait sur un serveur dédié pour éviter tout problème lié à une mauvaise manipulation dans le code. Les données utilisées telles que les clients, les fournisseurs, les utilisateurs, etc. sont hébergées sur un serveur réservé aux tests.
AC 3 : Sécuriser les services et données d'un système	n/a

Tableau 3 - Relation compétences du BUT et travaux réalisés (Compétence 3)

9.2.4. Compétence 4 : Gérer des données de l'information

Compétence 4 : Gérer des données de l'information	
AC 1 : Optimiser les modèles de données de l'entreprise	n/a
AC 2 : Assurer la confidentialité des données (intégrité et sécurité)	n/a
AC 3 : Organiser la restitution de données à travers la programmation et la visualisation	Afin d'organiser au mieux mes données, j'ai pu utiliser des listes.
AC 4 : Manipuler des données hétérogènes	Certaines données sont récupérées à partir du logiciel pour, ensuite, être utilisées plus tard dans le test. Les données choisies peuvent influencer le résultat du test.

Tableau 4 - Relation compétences du BUT et travaux réalisés (Compétence 4)

9.2.5. Compétence 5 : Conduire un projet

Compétence 5 : Conduire un projet	
AC 1 : Identifier les processus présents dans une organisation en vue d'améliorer les systèmes d'information	n/a
AC 2 : Formaliser les besoins du client et de l'utilisateur	n/a
AC 3 : Identifier les critères de faisabilité d'un projet informatique	n/a
AC 4 : Définir et mettre en œuvre une démarche de suivi de projet	Grâce aux outils mis en place (Jira, Git, réunions...) j'ai pu indiquer le travail réalisé, à faire, etc. TestRail permet aussi de voir l'avancée des tests fonctionnant.

Tableau 5 - Relation compétences du BUT et travaux réalisés (Compétence 5)

9.2.6. Compétence 6 : Collaborer au sein d'une équipe informatique

Compétence 6 : Collaborer au sein d'une équipe informatique	
AC 1 : Comprendre la diversité, la structure et la dimension de l'informatique dans une organisation	Le but du service informatique chez ComptaCom est de fournir un logiciel de comptabilité ainsi que du matériel informatique. Le service possède plusieurs équipes remplissant chacun des objectifs différents.
AC 2 : Appliquer une démarche pour intégrer une équipe informatique au sein d'une organisation	Je participais aux différentes réunions et discutais avec les membres de mon équipe ainsi que les autres personnes du service.
AC 3 : Mobiliser les compétences interpersonnelles pour intégrer une équipe informatique	Il a fallu faire preuve d'ouverture d'esprit et d'adaptabilité pour prendre en compte les conventions mises en place par l'entreprise. Il a aussi fallu vulgariser afin d'expliquer simplement les problèmes rencontrés.
AC 4 : Rendre compte de son activité professionnelle	Les daily meetings, revue de sprint et les autres échanges m'ont permis de rendre compte des travaux que j'ai produit ainsi que des potentielles difficultés rencontrées.

Tableau 6 - Relation compétences du BUT et travaux réalisés (Compétence 6)

10. Conclusion

Suite au changement de nom du logiciel Compta.com en Bleez, il a été décidé de changer la charte graphique de celui-ci impliquant une grande quantité de changements. Ces changements ont eu un impact sur les TFA existants, c'est pourquoi il a été nécessaire pour l'entreprise de faire appel à des stagiaires.

C'est dans ce contexte que mon stage a eu lieu, afin d'adapter les TFA à la nouvelle interface.

J'ai donc repris les tests existants et les ai améliorés en les optimisant et en les adaptant avec les nouveaux éléments web. Ces tests ont été effectués à l'aide du framework Robot Framework du langage Python et de TestRail pour réaliser les cas de test à partir des scénarios.

Cette production permet aux développeurs de tester, de manière automatique, les différents changements qu'ils effectuent lors des maintenances ou des migrations.

Pour la fin de mon stage, il reste encore une bonne quantité de tests à effectuer. Certains tests pourraient peut-être être améliorés un peu plus. Aussi, il pourrait être intéressant de créer des tests de force afin de mesurer l'impact d'une grande quantité de modifications à un moment donné.

11. Annexes

Les informations sur la présentation de l'entreprise ainsi que les images proviennent du site vitrine de ComptaCom :

Historique : <https://www.compta.com/expert-comptable/histoire/>

Coeur de métier : <https://www.compta.com/domaines-expertise/>

Organisation et structure : <https://www.compta.com/expert-comptable/> et Documents internes

Services et produits : <https://www.compta.com/domaines-expertise/>

Les informations sur Robot Framework ont été inspirées de sa page wikipédia :

Qu'est-ce que Robot Framework ? :

https://fr.wikipedia.org/wiki/Robot_Framework

Les images des outils utilisés et les codes produits ont été réalisés par mes soins et avec l'autorisation du personnel.