

Aboutou OUATTARA

BTS SIO 1^{ère} année option SLAM

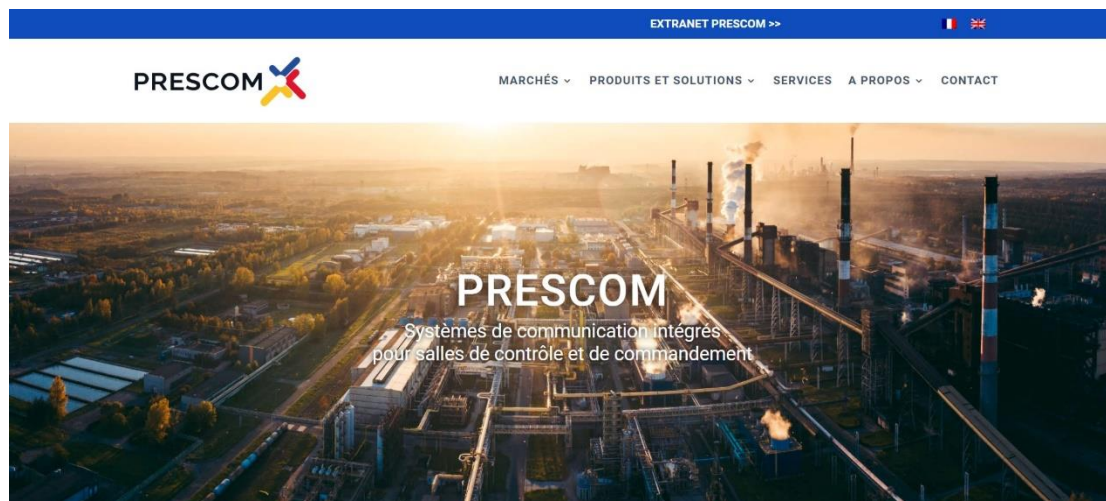
2023-2024



RAPPORT DE STAGE

3 JUIN 2024 – 06 JUILLET 2024

AUTOMATISATION DE TESTS SUR INTERFACES WEB



Encadrant lycée Rabelais 22 : M. Damien BOUCARD

Encadrant PRESCOM : M. Anthony HAMONOU

Résumé

Ce projet s'inscrivant dans le domaine des télécommunications a pour objectif l'automatisation d'installation d'OS (Operating Système) et de logiciels dans le but d'effectuer des tests automatiques.

Pour cela, nous sommes partis de zéro en installant l'environnement complet sur une machine physique qui a un OS Windows10 pour le bon fonctionnement des outils notamment (Python, Java, Robot Framework, NodeJS).

Le projet se poursuit par des tests fonctionnels automatiques sur 3 IHM web : le configurateur, l'Omnidispatch et la supervision (Interface Homme Machine) qui permet de faire des activités Radio (PTT : Push To Talk).

Table des matières

1 Prescom	6
1.1 Présentation générale	6
1.2 Historique	7
1.3 Qualité & Ethique	8
1.3.1 Qualité	8
1.3.2 Éthique & Développement durable.....	8
1.3.3 Assurer la viabilité et la rentabilité de l'entreprise	8
1.3.4 Être acteur du progrès social.....	8
1.3.5 Contribuer activement à un environnement préservé	8
1.3.6 Construire la gouvernance	8
2 Environnement technique	10
2.1 Distributions logicielles Prescom	10
2.1.1 CRITI-CALL	10
2.1.2 TACTICOM.....	11
3 Le projet	12
3.1 Description du développement réalisé	12
3.2 Scripts.....	13
3.3 Objectifs du projet	14
3.4 Phases du projet	15
Projet « 1 – installation complète »	15
Projet « 2 – installation des paquets »	15
Projet « 3 – vérification des paquets »	15
Projet « 4 – automatisation du configuration »	15
Projet « 5 – automatisation du WS »	16
Projet « 6 – automatisation de l'Omnidispatch ».....	17
.....	17
Projet « 7 – automatisation du superviseur ».....	18
4 Tests et résultats	19
4.1 Tests	19
4.2 Résultats	19
4.3 Perspectives	20

4.4 Problèmes rencontrés	20
4.5 Conclusion	21

Remerciements

Je tiens à remercier mes enseignants pour la formation que j'ai reçue au cours de ma 1^{ère} année de ce BTS et qui m'a beaucoup aidé dans la réalisation de ce stage et particulièrement à mon encadrant M. Damien BOUCARD qui a effectué le suivi de mon stage.

Mes sincères remerciements à mon maitre de stage M. Anthony HAMONOU qui m'a guidé durant ce stage, m'a conseillé et m'a fait part de ses connaissances dans ce domaine que sont les télécommunications que je connaissais à peine. Je remercie également M. Yohan BOUDER et M. Adrien LEMINOR pour leurs explications bien détaillées dans les différentes phases du projet et qui m'ont permis d'avancer plus sereinement.

Je ne saurais terminer sans remercier l'ensemble du personnel de l'entreprise pour leur chaleureux accueil dans leur structure notamment le responsable du département M. Manuel BILLOT.

Introduction

Dans ce monde numérique en perpétuelle évolution et d'une complexité de plus en plus croissante, le besoin d'automatisation se fait ressentir. L'automatisation est un ensemble de techniques visant à réduire ou rendre inutile l'intervention humaine dans un processus où celle-ci est habituelle. Ceci se passe par la création de processus reproductibles par des systèmes informatiques. Cela peut aider à soulager les équipes de certaines tâches manuelles répétitives. Elle permet d'augmenter la productivité des équipes de développement, de fiabiliser les systèmes, de réduire le nombre d'erreurs, et de consacrer davantage de temps aux tâches stratégiques.

C'est dans ce cadre que s'inscrit ce projet. Dans un premier temps, ce projet vise à automatiser la création d'un environnement complet et fonctionnel. Une fois que cet environnement est fonctionnel, on va pouvoir y effectuer des tests fonctionnels automatiques des activités radio, interconnexions, réécoutes, ainsi que les résultats sur un site de supervision.

Dans cette optique, nous allons tout d'abord présenter l'entreprise Prescom ainsi que son secteur d'activité. Ensuite, nous allons présenter le projet, les différents aspects sur lesquels il intervient. Nous aborderons par la suite en détail les phases de mise en œuvre et nous présenterons enfin les résultats obtenus à l'issue de ce projet.

1 Prescom

Dans ce chapitre, nous allons décrire l'entreprise dans laquelle s'est déroulé ce projet et la politique de Responsabilité Sociale des Entreprises (RSE) adoptée par l'entreprise.

1.1 PRESENTATION GENERALE

Prescom est une PME (Petite et Moyenne Entreprise) Française spécialiste des Communications Critiques Unifiées. Elle est leader dans la conception de systèmes de communication pour les salles de contrôle et de commandement et de passerelles d'interconnexion entre réseaux hétérogènes. Elle équipe plus de 400 salles de contrôle dans le monde pour la Sécurité Civile et Publique, la Sécurité Maritime, les Aéroports, les Transports Publics et la Défense.

Son siège social se trouve à Montigny-le-Bretonneux mais est également présente à Lannion où est situé sa majeure partie de Recherche et Développement (R&D) : **conception et le développement des solutions.**

Cette entreprise a été créée en 1982 et compte aujourd'hui 65 salariés répartis entre ses deux sites. Elle présente un chiffre d'affaires s'élevant à 9.2 millions d'euro dont 20% est réinvesti dans l'export et 20% sert à la partie R&D.

Sa principale mission est d'unifier les communications critiques entre les utilisateurs présents sur les terrains et ceux présent dans les salles de commandement.

Les solutions, développées par les équipes de R&D, sont conçues autour de blocs matériels et logiciels alliant des technologies de pointe et des concepts éprouvés. L'entreprise fournit des produits à haute valeur ajoutée à de grands intégrateurs dans le domaine de la sécurité et des télécommunications. Son offre de service complète, permet de livrer à ses clients, en direct ou par l'intermédiaire d'intégrateurs, des systèmes clefs en main, évolutifs, et pérennes.

1.2 HISTORIQUE

1982 : Création de PRESCOM

1988 : Dépôt d'un brevet définissant une architecture à commutation distribuée

1992 : RINSIS : salles de commandement de la Sécurité Civile en Belgique

1998 : PRESCOM fait l'acquisition de la société SECTRAD, société leader sur le marché des centres d'appels d'urgence en France

2000 : BEMILSATCOM : réseau de communication par satellite (armée belge)

2002 : PRESCOM est choisi par le ministère de l'Intérieur français pour équiper les salles de commandement du réseau ACROPOL, réseau radio numérique de la Police Nationale Française

2003 : PRESCOM intègre la société SECMAT, leader dans le domaine de la sécurité maritime

2004 : PRESCOM intègre l'activité cartes et modules Télécom I.S.D.N. de la société Netbricks

2006 : PRESCOM intègre la société JD COM, spécialisée dans le domaine de la sécurité des personnes sur site

2009 : Réorganisation du groupe PRESCOM à partir de la Holding PRESCOM Gestion et de trois filiales PRESCOM SAD, JD-COM et NODCOM

2013 : Jacques d'Harcourt rachète la société PRESCOM SAD qui devient PRESCOM SAS

2017 : PRESCOM réorganise son actionnariat et renforce son équipe managériale. Erik LE ROY, nouveau Président de PRESCOM, entre au capital, accompagné de Paul CORBEL.

1.3 QUALITE & ETHIQUE

1.3.1 Qualité

PRESCOM s'est inscrit dans une démarche qualité de type ISO/CEI 25051 qui définit les exigences de qualité applicables aux progiciels.

1.3.2 Éthique & Développement durable

Le développement de la rentabilité économique est au centre des préoccupations de PRESCOM. Sa finalité doit être au service du progrès social. Toutefois celui-ci ne peut se faire au détriment des générations futures.

S'engager dans la voie du développement durable, et s'inscrire dans une démarche de progrès réaliste et concrète :

1.3.3 Assurer la viabilité et la rentabilité de l'entreprise

Investir dans le développement durable ; évaluer la rentabilité de cet investissement

Avoir une production et une politique de prix responsables

Apporter aux clients : conseil, transparence, assistance et service après-vente

Acheter en toute responsabilité : traçabilité des achats, garanties des fournisseurs

Suivre une démarche d'amélioration continue

1.3.4 Être acteur du progrès social

Assurer de bonnes conditions de travail et d'ambiance dans l'entreprise

Être bienveillant et respectueux dans les relations avec les clients, fournisseurs, salariés...

Garantir l'équité, promouvoir la diversité

Faire progresser les compétences, se former

1.3.5 Contribuer activement à un environnement préservé

L'énergie : surveiller et réduire les consommations

Les déchets : les diminuer à la source, les trier

Optimiser les transports et la logistique

1.3.6 Construire la gouvernance

S'engager dans une démarche valorisant les produits et services de proximité

Organiser les responsabilités et les processus de décision

Rendre compte, communiquer en interne et en externe

Devancer la réglementation : gestion et prévention des risques

Instaurer des relations durables, performantes et de proximité avec les fournisseurs

PRESCOM mobilise ses équipes pour être certifiée ISO 9001.

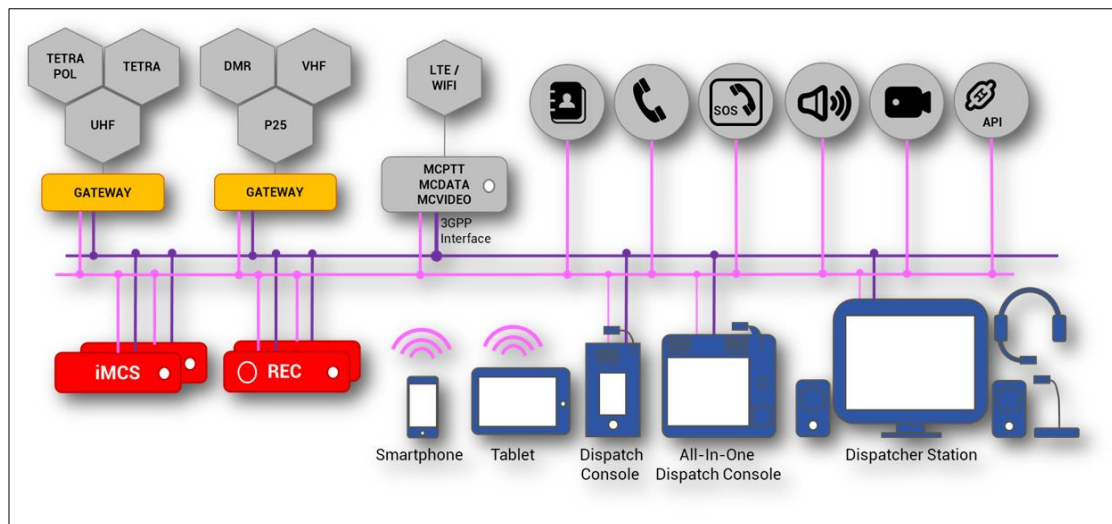
2 Environnement technique

Nous allons présenter les différents aspects de ce projet ainsi que les différentes technologies qui ont été utilisées.

2.1 DISTRIBUTIONS LOGICIELLES PRESCOM

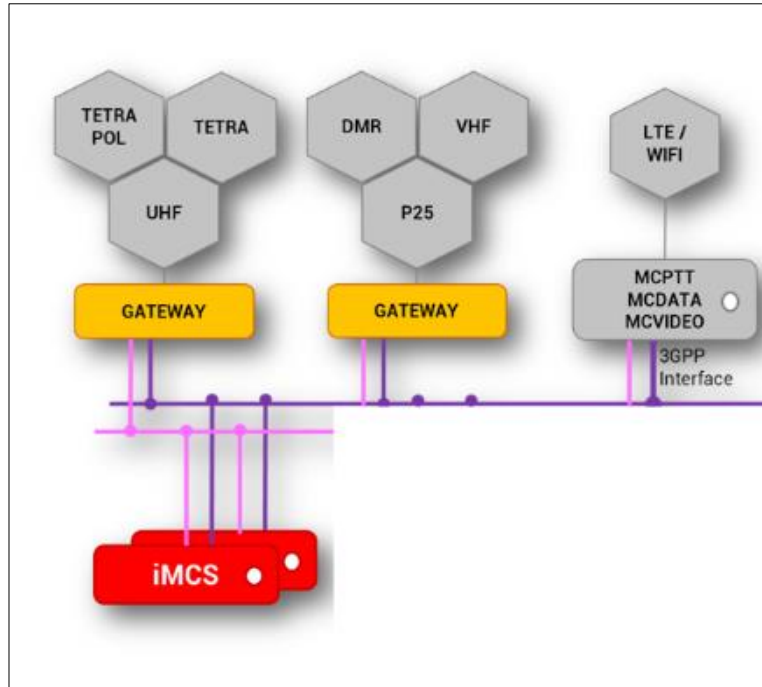
2.1.1 CRITI-CALL

CRITI-CALL est un système de communication critique unifiée permettant la conversion et l'adaptation à différents média, la distribution des communications, la mise en conférence et l'interphonie entre les opérateurs des centres de commandement, les mobiles radio et abonnés téléphoniques. Ce système assure également l'interopérabilité entre réseaux hétérogènes : réseaux radio analogiques et numériques, les réseaux de nouvelle génération LTE-MCPTT et des réseaux téléphoniques publics et privés analogiques et numériques.



2.1.2 TACTICOM

Un TACTICOM peut être défini comme étant un CRITI-CALL sans les postes opérateurs.



Les deux systèmes partagent le même coeur qui est l'iMCS (Integrated Mission Critical Server). C'est un commutateur qui permet de router des communications d'une entité (poste opérateurs, réseau radio, passerelle, ...) vers une autre.

3 Le projet

3.1 DESCRIPTION DU DEVELOPPEMENT REALISE

Settings (paramètres) : cette section concerne la configuration de notre environnement de test.

Variables : elles sont des éléments dynamiques que nous pouvons utiliser dans nos cas de tests. Elles permettent de rendre les tests plus flexibles et de les adapter à différentes situations. Par exemple, nous pouvons définir une variable pour stocker un nom d'utilisateur dans plusieurs cas de test.

Test Cases (cas de test) : Ce sont des scénarios détaillés qui décrivent comment tester une fonctionnalité spécifique. Chaque cas de test doit inclure une description, des étapes à suivre, des entrées, des sorties attendues et des critères de réussite. Ils servent de guide pour les testeurs lors de l'exécution des tests.

Keyword (mot-clé) : Ce sont des actions ou des instructions réutilisables dans les cas de test. Ils permettent d'éviter la duplication et de simplifier la maintenance. Par exemple, si nous avons une action « Se connecter », nous pouvons créer un mot-clé correspondant et l'utiliser dans les cas de test où la connexion est nécessaire.

3.2 SCRIPTS

```
*** Settings ***
Documentation      Simple example using SeleniumLibrary.
Library            SeleniumLibrary
```

```
*** Variables ***
${LOGIN URL}       https://www.saucedemo.com/v1/
${BROWSER}         Chrome
```

```
*** Test Cases ***
Valid Login
    Open Browser To Login Page
    Maximize Browser Window
    Input Username      standard_user
    Input Password      secret_sauce
    Submit Credentials
    Welcome Page Should Be Open
    [Teardown]         Close Browser
```

```
*** Keywords ***
Open Browser To Login Page
    Open Browser      ${LOGIN URL}      ${BROWSER}
    Title Should Be   Swag Labs
```

```
Input Username
    [Arguments]       ${username}
    Input Text         id= user-name     ${username}
```

```
Input Password
    [Arguments]       ${password}
    Input Text         id=password       ${password}
```

```
Submit Credentials
    Click Button       id=login-button
```

```
Welcome Page Should Be Open
    Title Should Be   Swag Labs
    Sleep             10s
```

```
*** Variables ***
${version_cible}    2.0.66.0
${version}          2.0.66.0

*** Test Cases ***
Vérifier la version
    ${element_cible}    Evaluate    [int(x) for x in "${version_cible}".split(".")]
    ${element_actuel}    Evaluate    [int(x) for x in "${version}".split(".")]
    Run Keyword If       "${element_actuel}" == "${element_cible}"    Log    Version
correcte
    ...    ELSE    Log    Erreur
```

3.3 OBJECTIFS DU PROJET

L'objectif était de découvrir le logiciel « Robot Framework », et de mettre l'environnement en place, de faire des tests de non-régression sur les interfaces web. Robot Framework est un environnement open source dédié à l'automatisation des tests. Il nous permet d'écrire et d'exécuter des cas de test sous formes de mots clés à l'aide de librairies intégrés disponibles sous <https://robotframework.org/>

L'installation d'un environnement complet nécessite :

- Python
- PIP
- RIDE
- Robot Framework
- wxPython

SeleniumLibrary	Web testing library that uses popular Selenium tool internally.	1304	WEB, SELENIUM
Browser Library	A modern web testing library powered by Playwright . Aiming for speed, reliability and visibility.	777	WEB
HTTP RequestLibrary (Python)	HTTP level testing using Python Requests internally.	458	HTTP
AppiumLibrary	Android and iOS testing. Uses Appium internally.	371	MOBILE
RESTtestrunner	Test library for HTTP JSON APIs.	195	HTTP
SSHLibrary	Enables executing commands on remote machines over an SSH connection. Also supports transferring files using SFTP.	143	FTP, SSH
Database Library (Python)	Python based library for database testing. Works with any Python interpreter, including Jython.	141	DB

Robot Framework permet la réalisation d'un rapport de test à la fin de chaque session indiquant le statut de chacune des étapes.

Log level: INFO

Generated 20240702 15:41:57 UTC+02:00
2 minutes 44 seconds ago

Robot Files Log

Test Statistics

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	6	5	1	0	00:00:00	<div></div>

Statistics by Tag

Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip	
No Tags						

Statistics by Suite

Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip	
Robot Files	6	5	1	0	00:00:01	<div></div>
Robot Files Test	6	5	1	0	00:00:01	<div></div>

Test Execution Errors

20240702 15:41:56.968

WARN

Error in file '/robot_files/keywords.resource' on line 66: The '[Return]' setting is deprecated. Use the 'RETURN' statement instead.

20240702 15:41:57.006

WARN

Error in file '/robot_files/keywords.resource' on line 66: The '[Return]' setting is deprecated. Use the 'RETURN' statement instead.

Test Execution Log

SUITE Robot Files

00:00:00.822

Full Name: Robot Files

Start / End / Elapsed: 20240702 15:41:56.392 / 20240702 15:41:57.214 / 00:00:00.822

Status: 6 tests total, 5 passed, 1 failed, 0 skipped

SUITE Test

00:00:00.511

PAGE 14

3.4 PHASES DU PROJET

Le projet s'est déroulé en plusieurs étapes.

Projet « 1 – installation complète »

Nous avons commencé par installer les pilotes nécessaires pour notre projet telles que :

- robot framework v7.0.1
- python3.12
- RIDE v2

Projet « 2 – installation des paquets »

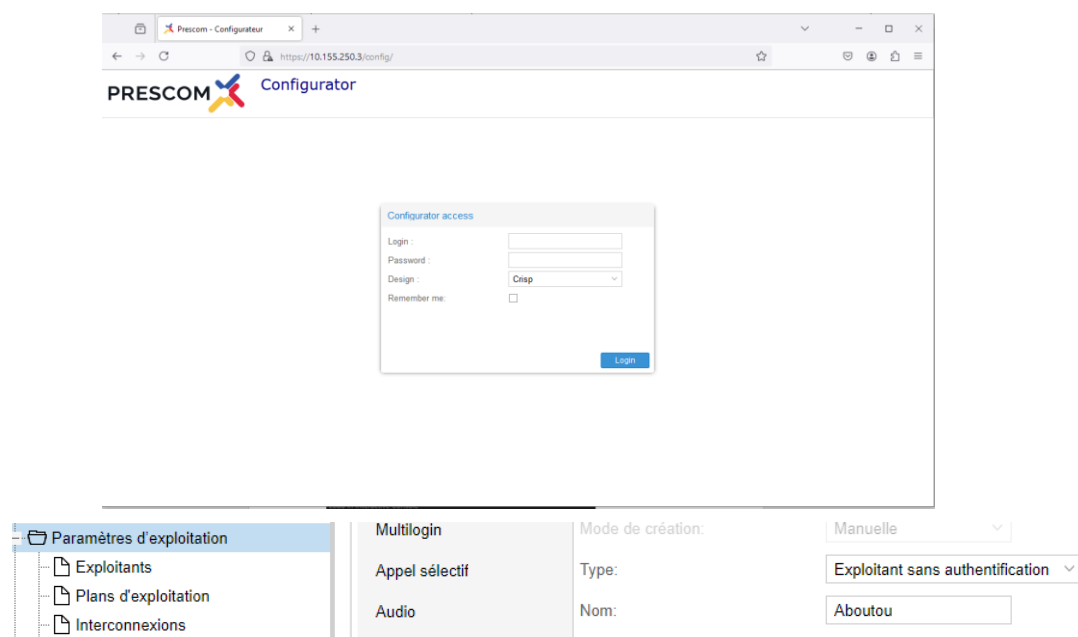
- pip
- seleniumLibrary
- chrome driver

Projet « 3 – vérification des paquets »

- pip --version
- robot --version
- ride.py
- pip show robotframework
- pip list

Projet « 4 – automatisation du configuration »

Le premier test de non-régression avait pour but de se connecter sur le configurateur automatiquement. Ensuite aller vérifier le profil de notre utilisateur, le renommer, l'enregistrer et appliquer la configuration. L'accès au configurateur nécessite un login/mot de passe que nous avons automatisés.




```
Load in Interactive Console
*** Variables ***
${nav}      chrome
${url_config}  https://10.155.250.3/config
#https://10.155.250.3/config
${login config}  xpath://*[@id="ext-53-inputEl"]
${mdp config}    xpath://*[@id="ext-54-inputEl"]
```

TEST Test_config
Full Name: Test1 Config Complet Test_config
Start / End / Elapsed: 20240702 12:06:23.842 / 20240702 12:06:47.651 / 00:00:23.809
Status: **PASS**

KEYWORD Test_config
Start / End / Elapsed: 20240702 12:06:23.848 / 20240702 12:06:47.649 / 00:00:23.801

- KEYWORD** SeleniumLibrary.Open Browser \${url_config} \${nav} alias=config
- KEYWORD** SeleniumLibrary.Wait Until Page Contains text=Login timeout=15s
- KEYWORD** SeleniumLibrary.Maximize Browser Window
- KEYWORD** BuiltIn.Sleep 3s
- KEYWORD** SeleniumLibrary.Click Element \${Avancer}
- KEYWORD** BuiltIn.Sleep 3s
- KEYWORD** SeleniumLibrary.Click Element \${Continuer}
- KEYWORD** BuiltIn.Sleep 3s

Projet « 5 – automatisation du WS »

Le web service permet à des applications d'échanger des informatiques sans avoir été construites dans le même langage ce qui est le cas avec nos IHM :

- le configurateur
- le superviseur
- l'Omnidpistach

```
${ws}      E:\\Aboutou Ouattara\\websevice\\WsPoImc.exe
${url_omni}  http://10.155.250.3/omnidispatch/#/plan
```

TEST Test_Ws
Full Name: Test1 Config Complet Test_Ws
Start / End / Elapsed: 20240702 12:07:22.018 / 20240702 12:07:27.044 / 00:00:05.026
Status: **PASS**

KEYWORD Test_Ws
Start / End / Elapsed: 20240702 12:07:22.020 / 20240702 12:07:27.042 / 00:00:05.022

- KEYWORD** Process.Start Process \${ws} alias=ws
- KEYWORD** BuiltIn.Sleep 5s

Projet « 6 – automatisation de l'Omnidispach »

L'application Omnidispach est une application permettant l'exploitation centralisée de divers réseaux de communications.

Cette application se présente sous la forme d'un exécutable faisant le lien entre l'IHM et le serveur. L'IHM se présente sous la forme d'un client léger, accessible via les navigateurs WEB Mozilla Firefox et Google Chrome.

L'automatisation de l'Omnidispach consistait à faire un appel et à la réécouter pour ensuite faire une interconnexion. Il fallait faire une diffusion, en choisissant une mélodie pour cela, j'ai produit un code qui devait automatiser ce processus. En premier, nous ouvrons un navigateur et en l'indiquant l'URL à accéder. Dans une deuxième une fois sur la page, je clique sur les paramètres → diffusion→je donne un nom à ma diffusion→ je sélectionne les différents canaux et une mélodie→je valide ma diffusion.

```
`${ws}` E:\\Aboutou Ouattara\\webservice\\WsPoImc.exe  
`${url_omni}` http://10.155.250.3/omnidispach/#/plan
```

```
- TEST Test_affichage_Omni  
Full Name: Test1 Config Complet Test_affichage_Omni  
Start / End / Elapsed: 20240702 12:07:27.049 / 20240702 12:08:10.432 / 00:00:43.383  
Status: PASS  
- KEYWORD Test_Open_Omni  
Start / End / Elapsed: 20240702 12:07:27.055 / 20240702 12:08:10.429 / 00:00:43.374  
+ KEYWORD SeleniumLibrary.Open Browser `${url_omni}` `${nav}` alias=omni  
+ KEYWORD SeleniumLibrary.Maximize Browser Window  
+ KEYWORD BuiltIn.Sleep 40s
```

Projet « 7 – automatisation du superviseur »

Pour la partie du superviseur l'objectif était d'aller sur l'IHM et vérifier les activités qui ont été faits depuis les autres IHM notamment le configurateur, l'Omnidispach et les versions des IHM si elles sont à jour.

```
${url_superviseur} https://10.155.250.3/supervision/
```

```
Click Element    xpath://*[contains(text(),"A propos")]
Sleep           2s
Run Keyword If    '${element_actuel}' != '${element_cible}'
...              Log To Console    Erreur
...              ELSE
...              Log To Console    Version à jour
```

```
- KEYWORD TC2_Switch_Browser
Start / End / Elapsed: 20240702 17:12:10.011 / 20240702 17:13:07.758 / 00:00:57.747
+ KEYWORD SeleniumLibrary.Open Browser ${url_superviseur} ${nav} alias=superviseur
+ KEYWORD SeleniumLibrary.Maximize Browser Window
+ KEYWORD BuiltIn.Sleep 5s
+ KEYWORD SeleniumLibrary.Wait Until Page Contains text=Login timeout=10s
+ KEYWORD BuiltIn.Sleep 3s
```

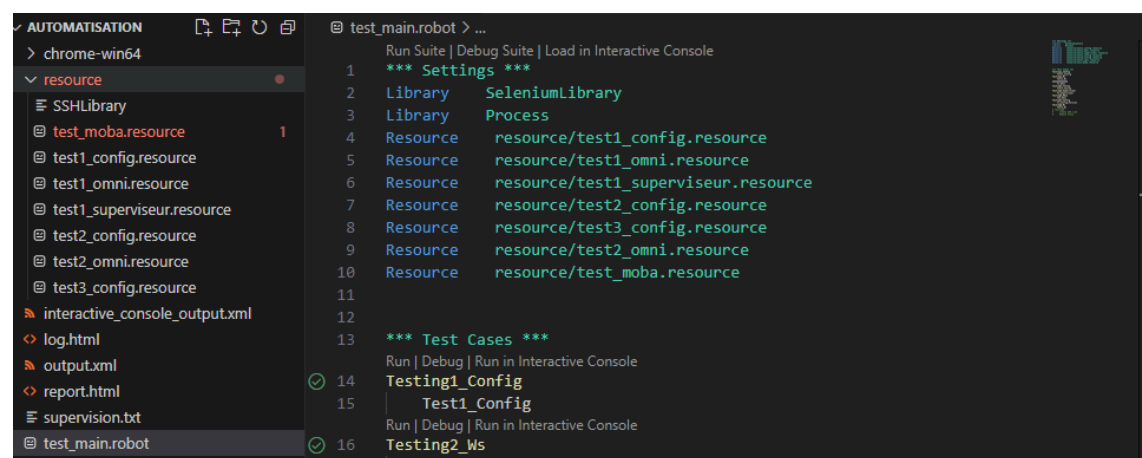
4 Tests et résultats

4.1 TESTS

Quelques tests ont été fait pour nous assurer du bon fonctionnement de notre environnement Robot Framework.

4.2 RESULTATS

Compte tenu des résultats, les test fonctionnement pour les rendre plus lisibles, nous avons optés pour des fichiers « .resources » des fonctions que nous appelons dans le programme principal.



The screenshot shows the Robot Framework IDE interface. On the left, the 'AUTOMATISATION' pane displays the project structure under 'chrome-win64', including a 'resource' folder with files like 'SSHLibrary', 'test_moba.resource', 'test1_config.resource', 'test1_omni.resource', 'test1_superviseur.resource', 'test2_config.resource', 'test2_omni.resource', 'test3_config.resource', 'interactive_console_output.xml', 'log.html', 'output.xml', 'report.html', 'supervision.txt', and 'test_main.robot'. The main editor shows the 'test_main.robot' file with the following content:

```
1 Run Suite | Debug Suite | Load in Interactive Console
2 *** Settings ***
3 Library SeleniumLibrary
4 Library Process
5 Resource resource/test1_config.resource
6 Resource resource/test1_omni.resource
7 Resource resource/test1_superviseur.resource
8 Resource resource/test2_config.resource
9 Resource resource/test3_config.resource
10 Resource resource/test2_omni.resource
11
12
13 *** Test Cases ***
14 Run | Debug | Run in Interactive Console
15 Testing1_Config
16 | Test1_Config
17 Run | Debug | Run in Interactive Console
18 Testing2_Ws
```

- SSHLibrary
- test_moba.resource
- test1_config.resource
- test1_omni.resource
- test1_superviseur.resource
- test2_config.resource
- test2_omni.resource
- test3_config.resource

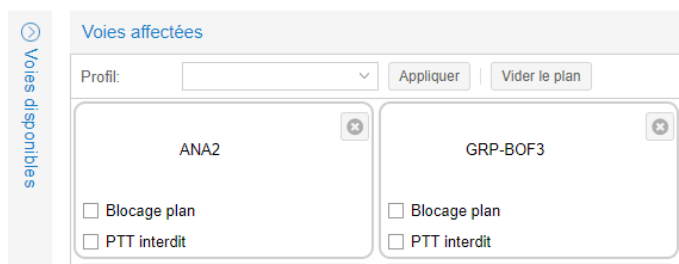
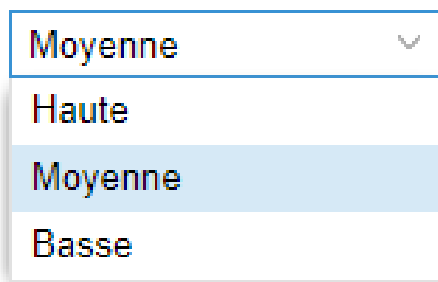
4.3 PERSPECTIVES

Quelques améliorations possibles :

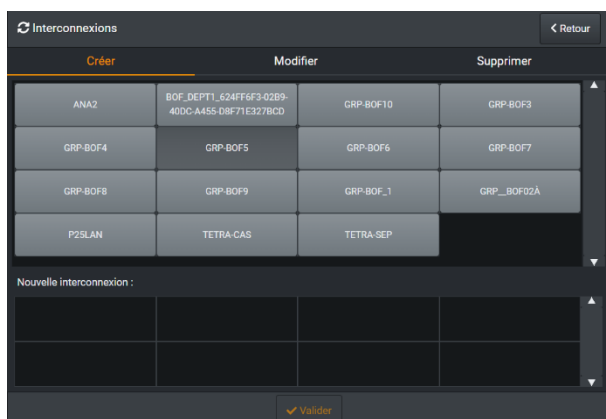
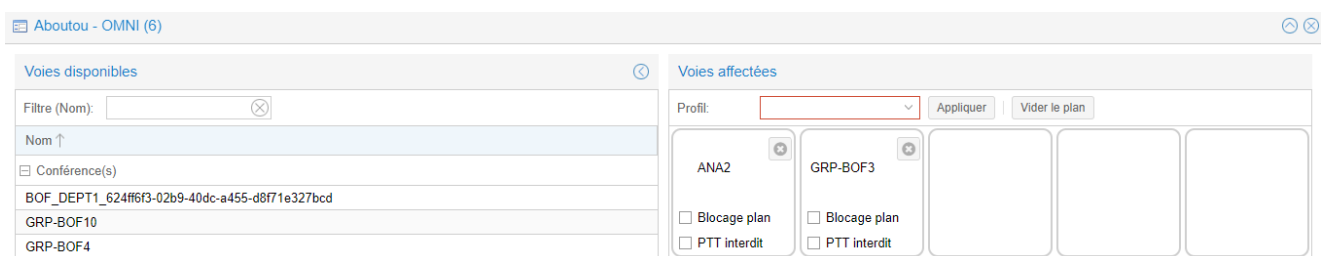
- possibilité de faire une condition sur le login et mot de passe lorsqu'on a dépassé le nombre de tentative.

4.4 PROBLEMES RENCONTRES

Lors des tests, la plupart des problèmes rencontrés étaient sur les dropdown « Menu-déroulant », car ils sont dynamiques pour les automatiser cela n'étaient pas chose facile mais avec acharnement, j'ai fini par y arriver.



```
Click Element    ${param_exploi}
Sleep            2s
Click Element    ${plan_exploi}
Sleep            2s
Double Click Element    ${omni_profil}
Sleep            2s
Click Element    ${voies_dispo}
Sleep            2s
Drag And Drop    xpath:/*[contains(text(),"GRP-BOF9")]    id=panel-1127-innerCt
Sleep            2s
```



```
Mouse Down    id=listen-5
Sleep         1s
Mouse Out     id=listen-5
Sleep         1s
Mouse Over    id=available-listen-4
Sleep         1s
Mouse Up      id=available-listen-4
Sleep         3s
```

4.5 CONCLUSION

Pour terminer, nous pouvons dire que les tests des scénarios définis pour l'instant répondent au besoin.

Dans l'ensemble, le besoin d'automatiser l'installation de logiciels et l'exécution des tests dans le but de fiabiliser les systèmes est plus ou moins comblé.