

Rapport de stage BTS 1 SIO option SLAM

Entreprise d'accueil : Weblib

Responsable de formation : Mr Jean-François REMAUD

Tuteur de stage: Mr. Souleymane TALL, Head of IT

Réalisé par : Melain Mahouton GBAGUIDI,

Etudiant en fin de 1^{ère} année de BTS SIO option SLAM au lycée Saint-Aspais et Stagiaire développeur web chez Weblib.

Année académique : 2024 / 2025

Sommaire

Introduction

Chapitre 1 : Etude de l'existant

1.1 Présentation de l'entreprise

1.2 Présentation du contexte informatique

Chapitre 2 : Cahier de charge

2.1 Objectif du stage

2.2 Schéma et explication du fonctionnement du test

Chapitre 3 : Travail accompli

3.1 Présentation approfondir des taches effectuées

3.2 Difficultés rencontrées et démarches de résolutions

Conclusion

Introduction

In an increasingly connected world, ensuring the reliability and performance of Internet access is a major challenge for companies like Weblib and their users. This is the context for my internship at Weblib, a company specializing in innovative technological solutions.

From May 26th to July 4th, I participated in a project within the Weblib development team called Smart WiFi. The main objective was to develop a solution for automating Internet access tests on Wi-Fi hotspots, particularly those using a captive portal. This project allowed me to leverage my skills in scripting, automated testing, and data processing, while discovering a demanding professional environment.

This report is organized into three main sections. First, I will present Weblib and the IT context in which the project is taking place. The second part will focus on the internship's objective, with graphical representations of the process and some illustrative code snippets.

Finally, I will detail the work accomplished, highlighting the challenges encountered and the proposed solutions, as well as the contributions of this experience.

Chapitre 1 : Etude de l'existant

Afin de mieux comprendre le cadre dans lequel s'est déroulé ce projet, il est essentiel de présenter un premier temps **l'entreprise Weblib**, ainsi que le contexte informatique qui a motivé la mise en place de cette solution d'automatisation des tests d'accès Internet.

1.1 Présentation de l'entreprise

Weblib est un éditeur logiciel qui accompagne les grands acteurs du retail, de la restauration, de l'hôtellerie, de la santé, du transport et de l'éducation dans leur transformation digitale. Les solutions développées par Weblib permettent d'exploiter le wifi et les tablettes pour connecter les consommateurs, les collaborateurs et les entreprises. Fondée en 2009, l'entreprise est présente en France, en Allemagne et au UK, et déploie ses solutions digitales dans plus de 70 pays. Après deux acquisitions réussies (Urban Koncept en 2017 et Ucopia en 2019), Weblib poursuit son développement rapide avec une ambition forte : devenir le leader de la transformation digitale des lieux accueillant du public.

Des solutions complémentaires et innovantes, qui réinventent l'expérience client

Cofondé par Arthur Philbé et Florian Galby en 2009, Weblib décline une gamme complète de solutions qui se mettent au service de la digitalisation des lieux accueillant du public : Smart Wifi, solution de marketing de proximité ; Tab in Store, logiciel dédié au déploiement et à la gestion de tablettes connectées ; Smart Kiosk, qui digitalise l'expérience client dans le secteur bancaire. Présent dans 80 pays et 10.000 lieux d'accueil, cet éditeur accompagne des leaders tels que McDonald's, Uniqlo, Total, AXA, Carrefour ou encore la SNCF. Depuis sa création, Weblib ne cesse de consolider son offre, en s'appuyant pour cela sur une innovation interne dynamique et des acquisitions stratégiques.

1.2 Présentation du contexte informatique

L'organisation de Weblib repose sur une structure articulée autour de quatre grandes équipes, chacune composée de plusieurs sous-groupes spécialisés, qui contribuent ensemble à l'efficacité opérationnelle, à l'innovation technologique et à la satisfaction client. La première grande équipe est celle des fonctions Finance & Administratives, qui regroupe les pôles de comptabilité, les administrateurs des ventes et une responsable administrative. La deuxième équipe est celle des Opérations composée de deux sous-équipes (...), qui assurent une prestation de service fiable, rapide et de haute qualité, indispensable dans un environnement en constante évolution. La troisième grande équipe est celle des Ventes & Marketing. Elle comprend une équipe de pré-vente, une équipe marketing, une équipe de vente dédiée au marché français, ainsi qu'une équipe de vente internationale. Ensemble, elles œuvrent à promouvoir les solutions de Weblib, à identifier de nouvelles opportunités 7 commerciales, à valoriser l'image de marque de l'entreprise, et à développer la présence de Weblib sur les marchés nationaux et mondiaux. Enfin, la quatrième grande équipe est celle de la Technologie, qui se divise en plusieurs sous-équipes

complémentaires. Elle inclut d'une part l'équipe Infrastructure, responsable de l'architecture technique, de la cybersécurité et de la performance des systèmes, et d'autre part l'équipe Produit, plus orientée vers le développement fonctionnel et l'innovation logicielle. Cette dernière se compose de plusieurs sous-groupes :

- L'équipe Web, au sein de laquelle j'ai eu l'opportunité d'être intégré. Cette équipe est spécialisée dans le développement web et travaille sur le projet Smart Wifi, qui travaille en lien étroit avec une équipe de développeurs Android dédiée à la déclinaison mobile des solutions. Elle est constituée de 9 membres actifs.
- L'équipe R&D Contrôleur, bénéficie d'une forte complémentarité de compétences, ce qui permet un développement agile et robuste des solutions.
- Enfin, les UX/UI designers apportent leur expertise en ergonomie et en design d'interface, afin de garantir une expérience utilisateur fluide, intuitive et esthétique sur l'ensemble des produits proposés.

Cette organisation cohérente et structurée, fondée sur des pôles bien définis, une collaboration transversale et une culture d'excellence, permet à Weblib de répondre efficacement aux enjeux technologiques, de s'adapter à un secteur en constante évolution, et de consolider sa position de leader tout en maintenant une dynamique continue d'innovation et de proximité avec ses clients.

Chapitre 2 : Cahier de charge

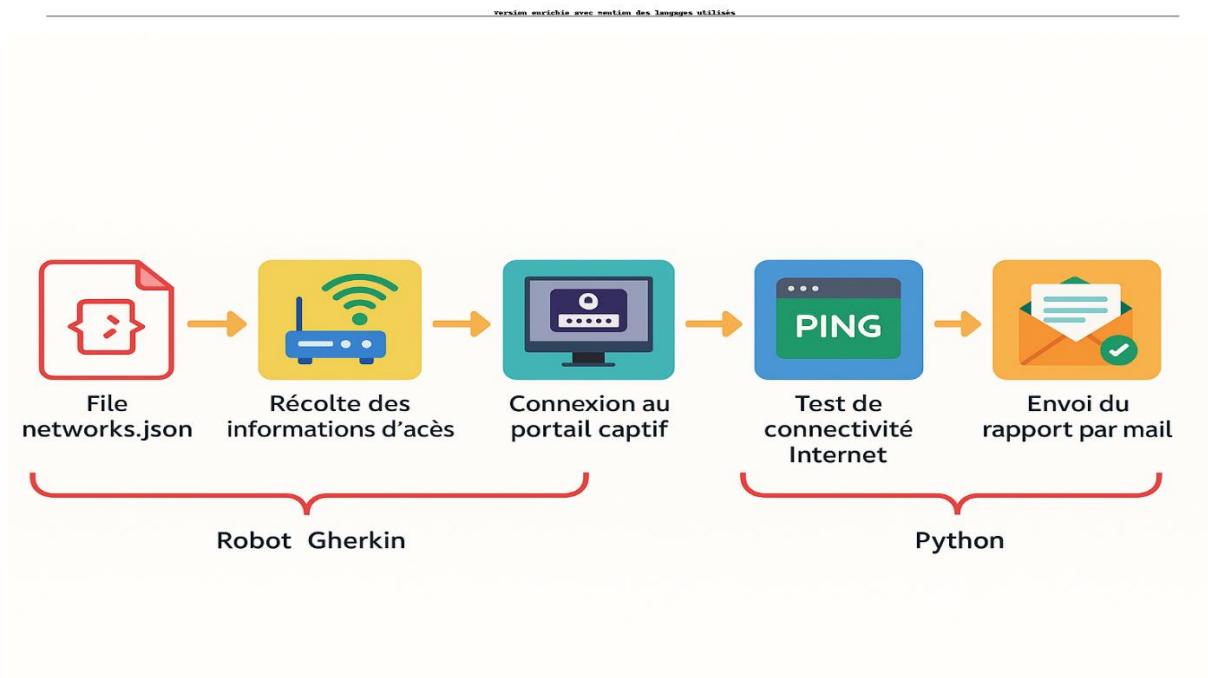
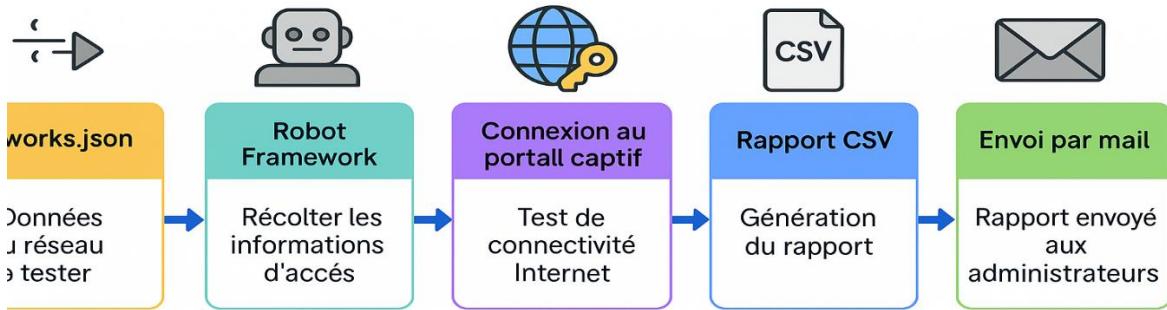
Dans le but de garantir l'efficacité du produit au service des clients, l'idée d'automatiser les tests afin d'assurer son bon fonctionnement en continu a émergé. J'ai été assigné à cette mission au sein de l'équipe web.

2.1 Objectif du stage

L'objectif principal de mon stage était de mettre en place un programme avec Robot Framework permettant d'effectuer des tests automatiques d'accès à Internet sur des points d'accès. Ces tests sont réalisés par un robot, qui va se connecter à un point d'accès en renseignant une adresse mail et un mot de passe prédefinis, puis vérifier le bon fonctionnement de la connexion Internet. Le résultat du test est ensuite envoyé par mail ou via Slack (application de messagerie utilisée chez Weblib), afin de tenir informé l'administrateur du bon fonctionnement continu des points d'accès. Ces derniers étant des produits destinés aux clients, il est essentiel d'anticiper tout problème potentiel.

Voici un schéma illustrant le processus de chaque test :

2.2 Schéma et explication du fonctionnement du test



Les étapes clés du test :

1. Récupération des informations du réseau à tester (SSID, email, mot de passe, URL, etc.) depuis un fichier .json, puis connexion au SSID ciblé.
2. Accès au portail captif, saisie des identifiants (email, mot de passe), vérification de la réussite de la connexion, puis lancement du test d'accès à Internet.

```

9 Access captive portal
10   [Arguments]  ${URL}
11   Open Browser  ${URL}    Chrome
12
13 # Connect to Captive Portal
14 Load in Interactive Console
15 Connect to Captive Portal
16   [Arguments]  ${email}  ${code}
17   # Wait for the page to load
18   Wait Until Page Contains Element  id=username
19
20   # Enter the username and password
21   Input Text  id=username  ${email}
22   Input Text  id=password  ${code}
23   Click Button  id=login-button
24
25   # Wait for the login to complete
26   Wait Until Page Contains Element  id=welcome-message
27
28 Load in Interactive Console
29 Check Internet Access
30   [Arguments]  ${ssid}
31   ${ping}=  Run Process  ping  -c  3  8.8.8.8  shell=True  stdout=PIPE
32   ${stdout}=  Set Variable  ${ping.stdout}
33
34   # check if the test pass
35   ${success}=  Run Keyword And Return Status  Should Contain  ${stdout}  bytes from

```

3. Génération d'un fichier de rapport de test via un script Python.
4. Envoi automatique du fichier par email ou via Slack, toujours à l'aide de Python.
5. Passage au réseau suivant : le processus de test se relance.

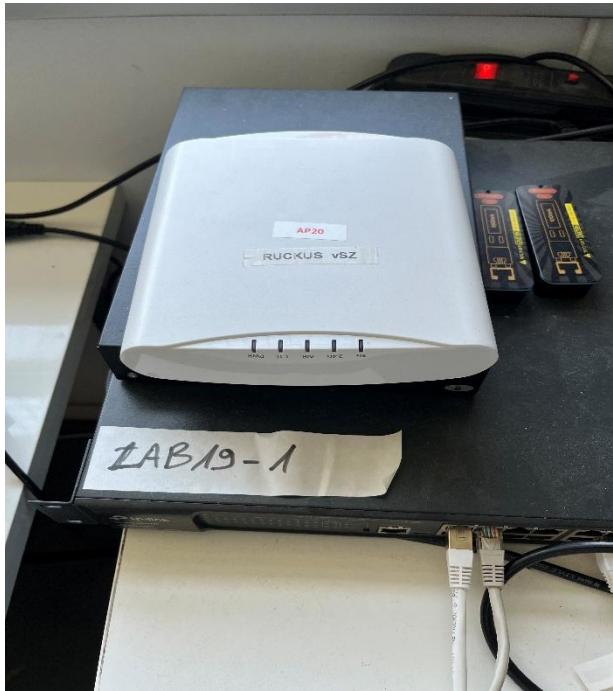
Architecture

```

├── src
│   ├── config           # json test environnement configuration
│   │   └── networks.json
│   │
│   ├── features          # suite written in BDD (Gherkin)
│   │   ├── auth
│   │   │   ├── login.robot
│   │   │   └── test.feature
│   │   └── tests           # functionality test
│   │       └── test.robot
│   │
│   ├── misc
│   │   ├── report.py      # generate a csv file for the test results
│   │   ├── report_by_mail.py
│   │   └── slack.py
│   │
│   └── resources
│       ├── keywords        # keywords test
│       │   └── keywords.resource
│       └── library          # Library test
│           └── library.resource
└── variable.resouce      # retrieve the values from the network
├── .gitignore
└── README.md
└── dependances.txt

```

Equipement du test



Chapitre 3 : Travail accompli

3.1 Présentation approfondir des tâches effectuées

a. Tests manuels sur les points d'accès

J'ai commencé par effectuer des tests manuels sur des points d'accès. Cela consistait à configurer un SSID via l'interface d'administration, puis à vérifier la connectivité Internet. Ces tests m'ont permis de mieux comprendre le fonctionnement réseau des équipements et d'identifier les éventuels problèmes d'accès rencontrés par les utilisateurs.

b. Mini-projet en Node.js

Pour me familiariser avec Node.js, un mini-projet m'a été confié dès le début du stage. Cela m'a permis de découvrir ses bases : création de serveur local, manipulation des modules, compréhension de l'asynchrone et gestion des dépendances via npm. Grâce à ce projet d'introduction, j'ai pu acquérir des compétences essentielles en développement backend avec Node.js.

c. Utilisation de Git et GitLab

L'environnement de travail reposait fortement sur GitLab pour le versionnage de code. J'ai donc appris à :

- Configurer la connexion SSH entre mon dépôt local et le repository distant
- Cloner des projets existants
- Travailler sur des branches dédiées à des fonctionnalités
- Fusionner les branches via des *merge requests* après validation

- Gérer les conflits liés à des modifications simultanées sur le même code
Cette maîtrise progressive de GitLab a été fondamentale pour collaborer efficacement en équipe.

d. Familiarisation avec l'environnement Linux (WSL)

Afin de développer dans un environnement Linux, j'ai installé et utilisé WSL (Windows Subsystem for Linux) sur ma machine. Cela m'a permis d'exécuter des commandes Unix, d'installer des outils nécessaires au projet, et de me familiariser avec la ligne de commande. Cette approche m'a ouvert à une nouvelle manière de travailler, que je trouve aujourd'hui incontournable.

e. Rédaction de documentation technique

Une partie importante du travail consistait à rédiger une documentation claire et complète pour chaque projet. J'ai utilisé Confluence pour les documents internes et un fichier README.md dans chaque projet pour décrire :

- Le but du projet
- Son architecture
- Son installation et sa configuration
- Son fonctionnement et ses cas d'usage

Cette documentation est essentielle pour garantir la maintenabilité du code et faciliter le travail des collaborateurs.

f. Conception de l'architecture d'un projet

En m'appuyant sur l'analyse d'autres projets existants, j'ai conçu l'architecture de mon propre projet. J'ai structuré les dossiers selon les bonnes pratiques : dossier `src` pour les sources, dossier `test` pour les scripts de test, et un fichier `README.md` pour documenter le tout. Cette tâche m'a permis de réfléchir à l'organisation du code, sa modularité et sa facilité de compréhension.

3.2 Difficultés rencontrées et démarches de résolutions

❖ Difficultés rencontrées

- **Pas de connaissance en Node.js**

En début de stage, je ne maîtrisais pas Node.js, qui est pourtant central dans l'environnement technique de l'entreprise.

- **Pas de connaissance de base sur la gestion de conflits Git et la configuration d'un pipeline CI/CD**

La collaboration au sein d'une équipe technique nécessite une bonne maîtrise de Git, notamment pour résoudre les conflits liés au travail simultané sur le même code. De plus, je ne connaissais pas les principes fondamentaux de l'intégration et du déploiement continu (CI/CD), indispensables pour automatiser le cycle de vie des applications.

- **Débutant sur Linux**

L'environnement de développement utilisé reposait majoritairement sur des serveurs Linux. En tant que débutant, j'avais des difficultés à naviguer dans le terminal, gérer les fichiers et exécuter des scripts, ce qui ralentissait mes tâches au quotidien.

- **Pas de connaissance en Robot Framework**

Robot Framework étant l'outil choisi pour automatiser les tests, il m'était essentiel de comprendre son fonctionnement. N'ayant jamais travaillé avec cet outil auparavant, cela représentait une difficulté supplémentaire, notamment dans la création de cas de tests pertinents et leur exécution sur les points d'accès.

❖ Démarches de résolutions

Pour surmonter ces obstacles et mener à bien mon projet, j'ai adopté plusieurs stratégies :

- **Consultation régulière de la documentation**

J'ai investi du temps dans la lecture des ressources officielles, de tutoriels et de guides en ligne pour chacun des outils. Cela m'a permis de mieux comprendre les concepts de base, le fonctionnement des pipelines, les commandes Linux essentielles et la syntaxe propre à Robot Framework.

- **Utilisation de l'intelligence artificielle comme assistant technique**

En complément, j'ai utilisé l'IA comme soutien dans mes apprentissages. Elle m'a aidé à clarifier des notions complexes, à résoudre des erreurs techniques et à me proposer des exercices adaptés à mon niveau, notamment sur Robot Framework .

- **Échanges et accompagnement par les membres de l'équipe**

J'ai sollicité régulièrement l'aide de mes collègues, qui m'ont guidé avec bienveillance, partagé leurs bonnes pratiques et m'ont donné des explications concrètes sur le fonctionnement de l'architecture technique en place. Ces échanges ont grandement accéléré ma montée en compétence.

Conclusion

Ce stage au sein de l'entreprise **Weblib** m'a offert une expérience enrichissante, à la fois sur le plan technique et professionnel. En participant au projet **Smart WiFi**, j'ai pu contribuer à la mise en place d'une solution d'**automatisation de tests d'accès à Internet sur des bornes WiFi**, un enjeu concret et actuel dans le domaine des réseaux.

Ce travail m'a permis de consolider mes compétences en **Robot Framework**, en **scripting Python**, et en **gestion de fichiers de données**, tout en découvrant les méthodes de travail en environnement professionnel. J'ai appris à structurer des scénarios de test, à interagir avec des portails captifs, à manipuler des fichiers JSON et CSV, et à automatiser l'envoi de rapports via email. J'ai également été confronté à plusieurs difficultés techniques que j'ai pu résoudre grâce à des recherches, des tests et l'aide de l'équipe encadrante.

Au-delà des aspects techniques, ce stage m'a permis de développer mon autonomie, mon sens de l'analyse et mes capacités de communication, en m'intégrant dans une équipe projet et en contribuant activement à un livrable concret.

Cette première expérience professionnelle constitue ainsi un **véritable tremplin** pour la suite de mon parcours, et renforce ma motivation à évoluer dans le domaine des **réseaux, de l'automatisation et de l'ingénierie logicielle**.