

**Dilhan YILDIRIM**

BUT Réseaux et Télécommunications - 3ème année

IUT1 Grenoble

Du 26/02/2024 au 14/06/2024

# Rapport de stage

Mise en place d'une infrastructure d'hébergement sous Docker

Développement logiciel en Python

**WEBU**



Tuteur entreprise : CORDEL Dylann

Tuteur enseignant : DELNONDEDIEU Yves

<b>Rapport visé par le tuteur d'entreprise :</b>  M. CORDEL Dylann <i>Associé - Analyste développeur</i> WEBU	<b>Signature :</b>
---	--------------------

## Remerciements

Je tiens avant tout à remercier grandement mon tuteur de stage Dylann CORDEL pour m'avoir appris de nombreuses choses, m'avoir accompagné, et pour m'avoir fait confiance.

Je souhaiterais également remercier toute l'équipe de Webu pour m'avoir accordé leur confiance, pour m'avoir accueilli et pour leur bonne humeur permanente. Je suis vraiment heureux d'avoir pu effectuer mon stage à leurs côtés, car j'ai appris énormément de choses, toujours dans une ambiance des plus agréable.

Enfin, j'aimerais remercier Mme. Jacqueline MADRENNES, sans qui ce stage n'aurait jamais eu lieu, car elle m'a fait confiance et m'a mis en relation avec Webu.

## Table des matières

<b>I - Introduction.....</b>	<b>5</b>
<b>II - Présentation de Webu.....</b>	<b>6</b>
1 - Présentation générale.....	6
2 - La SCOP.....	7
3 - L'Équipe.....	7
<b>III - Mon projet.....</b>	<b>8</b>
1 - Introduction.....	8
a - Le contexte.....	8
b - Les limites actuelles.....	10
c - Mes missions.....	10
2 - Mise en place de la nouvelle infrastructure d'hébergement.....	11
a - Mon objectif.....	11
b - Les tâches effectuées.....	12
c - Résultats et conclusion.....	14
3 - Développement de fonctionnalités de WebuNG.....	15
a - Mon objectif.....	15
b - Les tâches effectuées.....	16
c - Résultats et conclusion.....	18
4 - Conclusion.....	19
<b>IV - Bilans.....</b>	<b>20</b>
1 - Bilan professionnel.....	20
2 - Bilan personnel.....	21
<b>V - Annexes.....</b>	<b>22</b>
<b>VI - Résumés.....</b>	<b>24</b>

## Index des Illustrations

Illustration 1: Schéma simplifié du cycle de vie d'un site web.....	6
Illustration 2: Organisation de l'équipe et postes des salariés.....	7
Illustration 3: Schéma de l'infrastructure d'hébergement actuelle.....	9
Illustration 4: Schéma de la constitution d'une instance de projet.....	11
Illustration 5: Héritage des images Docker.....	12
Illustration 6: Schéma simplifié de l'architecture d'une instance avec Docker.....	13
Illustration 7: Schéma simplifié des étapes de l'installation d'une instance en développement.....	16

## I - Introduction

Actuellement en 3ème et dernière année de BUT Réseaux et Télécommunications, j'effectue mon stage de fin de formation de 16 semaines, du 26 février 2024 au 14 juin 2024, au sein de l'entreprise de développement web<sup>1</sup> : Webu.

Lors de mon stage, j'ai travaillé sur le renouvellement de l'infrastructure d'hébergement<sup>2</sup>, notamment sur la mise en place de la nouvelle solution et sur le développement d'un logiciel en lien avec celle-ci. Cela m'as permis de mêler le domaine des infrastructures systèmes<sup>3</sup> et réseaux<sup>4</sup> au domaine du développement logiciel<sup>5</sup>. J'expliquerai donc en détails mes missions importantes dans la suite de mon rapport.

Pour ce faire, je vais commencer par présenter Webu, l'entreprise qui m'a accueilli, et dans quel cadre j'ai effectué mon stage. Puis je vais vous décrire les missions auxquelles j'ai participé durant mon stage, en parlant notamment des objectifs assignés, des tâches effectuées et des résultats obtenus. Puis je terminerais par un bilan, qui va permettre de montrer ce que j'ai appris et compris grâce à ce stage, autant sur le plan technique que sur le plan personnel.

---

1 **Développement web** : Création de sites web (ou sites internet) pour des clients.

2 **Hébergement** : Le fait de rendre un site web disponible sur Internet, en stockant les codes et les données sur un serveur.

3 **Systèmes** : Désigne les moyens matériels informatiques utilisés par l'entreprise.

4 **Réseaux** : Désigne des ensembles d'appareils informatiques connectés et communicants entre eux.

5 **Développement logiciel** : Création d'application informatique.

## II - Présentation de Webu

### 1 - Présentation générale

Webu est une agence de développement web en SCOP<sup>6</sup>, fondée en 2005, située rue Saint-Laurent à Grenoble. Elle se spécialise dans la création de sites web sur mesure et s'occupe du cycle de vie entier de ses projets. En effet, Webu s'occupe d'abord de la conception avec le client, nécessaire pour répondre le mieux possible à ses besoins, autant sur le design que sur les fonctionnalités attendues. Ensuite, vient la phase de développement qui peut être plus ou moins longue et complexe en fonction des besoins. Enfin, Webu héberge le site, et s'occupe de la maintenance de ce dernier.

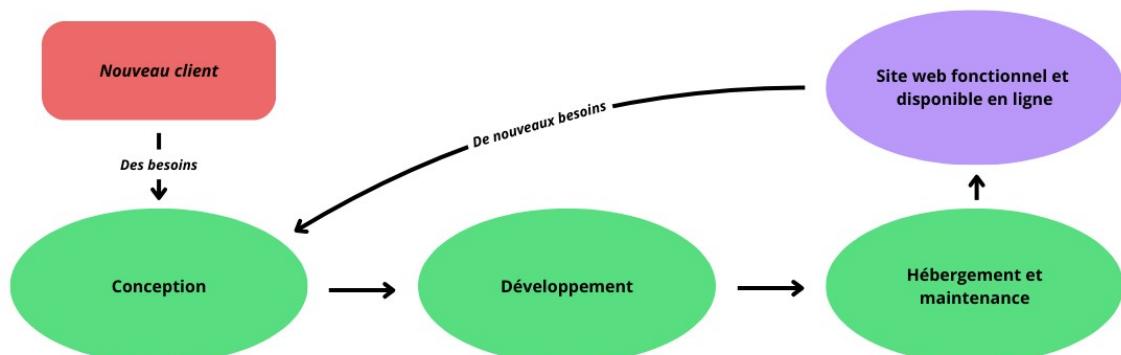


Illustration 1: Schéma simplifié du cycle de vie d'un site web

Les clients de Webu sont pour la plupart des moyennes entreprises ou des associations, qui cherchent généralement un site métier<sup>7</sup>, avec de nombreuses fonctionnalités pour répondre aux besoins et contraintes spécifiques. Pour répondre à ce type de demandes, Webu est surtout spécialisé dans le langage Python, notamment avec le framework<sup>8</sup> Django, mais parfois, ils utilisent aussi les langages PHP/Wordpress pour des sites avec peu de fonctionnalités et/ou des clients avec moins de budget.

<sup>6</sup> SCOP : « Société Coopérative Ouvrière de Production », une forme d'entreprise où les salariés peuvent également être associés à voix égales

<sup>7</sup> Site métier : Site web qui répond aux besoins spécifiques d'une entreprise/organisation.

<sup>8</sup> Framework: Un groupe de composants logiciels permettant de faciliter le développement.

## 2 - La SCOP

La particularité de Webu réside dans son statut de SCOP, elle permet notamment aux nouveaux salariés de devenir associés après un an. Ce statut permet à tout le monde de s'investir dans les décisions de l'entreprise. En effet, personne n'impose son autorité et tout le monde coopère pour prendre des décisions, ce qui favorise un sentiment d'appartenance et une meilleure implication dans la réussite de l'entreprise. La coopération est aussi présente entre Webu et les clients, permettant ainsi une expérience et un résultat bien meilleurs. Tout cet aspect coopératif constitue l'essence même de l'identité de Webu depuis sa fondation.

## 3 - L'Équipe

Webu se compose de 8 salariés dont 7 associés :

- Une équipe de 4 développeurs, qui s'occupent essentiellement du développement des sites web.
- 2 chefs de projets/commerciaux qui s'occupent principalement de la gestion de projet et des échanges avec les clients.
- Un administrateur systèmes et réseaux, qui est notamment responsable de la gestion des infrastructures informatiques de Webu.
- Un conseiller en gestion, qui s'occupe d'une partie des tâches administratives de l'entreprise.

De plus, certains salariés possèdent un double rôle/poste : Un des développeurs est aussi administrateur systèmes et réseaux, puis un développeur et un chef de projet s'occupent de la comptabilité. Ces doubles rôles permettent à Webu de s'investir plus en profondeur dans la structure et assurent une réelle coopération entre tous.

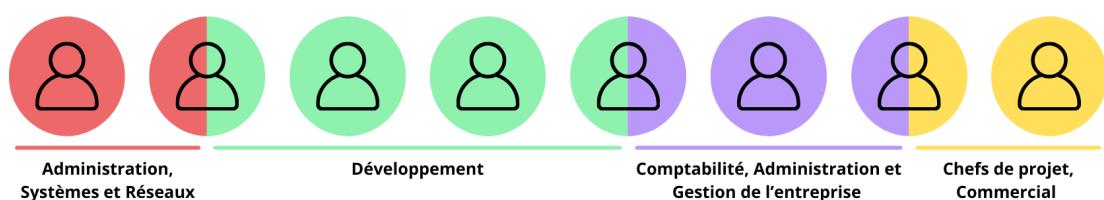


Illustration 2: Organisation de l'équipe et postes des salariés

## III - Mon projet

### 1 - Introduction

#### a - Le contexte

Il y a plus de 10 ans, Webu a fait le choix d'héberger soi-même les sites qu'il crée, pour ne plus dépendre d'entreprises externes (par exemple OVH), ce qui a beaucoup d'avantages. En effet, cela confère une maîtrise totale des serveurs<sup>9</sup>, ce qui permet par exemple une maintenance plus rapide et efficace sur le matériel et les logiciels. Également, Webu peut garder les données dans son infrastructure, ce qui apporte une meilleure confiance et une vraie transparence vis-à-vis des clients. Ainsi, pour héberger ses projets, il avait été choisi d'installer des serveurs physiques dans un datacenter<sup>10</sup>, et d'utiliser un hyperviseur, qui est un outil logiciel de gestion de machines virtuelles<sup>11</sup>, sur lesquelles les sites web sont installés.

Alors, avec l'hébergement interne, on a distingué la gestion des sites dans 2 environnements différents :

- En développement, c'est-à-dire que, les projets sont installés sur les ordinateurs des développeurs pour leur permettre de développer les codes du site. Il y a alors plusieurs contraintes à respecter, notamment que les sites sont lancés dans le réseau interne à Webu, avec les codes accessibles et modifiables par les développeurs.
- En production, quand les sites sont fonctionnels et accessibles sur Internet. Cela apporte d'autres contraintes comme l'installation sur les serveurs dans le datacenter, et la sécurisation des codes, des fichiers et des données.

Aussi, en plus de ces 2 environnements bien différents, on va encore devoir les séparer en fonction du langage du site, et même de la version, car par exemple, un environnement permettant de lancer un site en Python ne doit pas pouvoir lancer un site dans un autre langage, pour éviter les conflits entre eux.

---

<sup>9</sup> **Serveur:** Un système informatique qui fournit des services à des clients.

<sup>10</sup> **Datacenter:** Lieu qui regroupe des équipements informatiques physiques, notamment des serveurs.

<sup>11</sup> **Machines virtuelles:** Une émulation d'un équipement informatique, sur une machine physique.

Alors, Webu s'est retrouvé avec beaucoup d'environnements différents à gérer, c'est pourquoi l'administrateur systèmes et réseaux de l'époque avait développé WebuEnv, pour Webu Environnement, qui est un outil logiciel permettant de faciliter la gestion des sites dans les différents environnements. Il permet de nombreuses fonctionnalités, comme par exemple, l'installation d'une instance d'un projet sur une machine. C'est-à-dire, l'installation d'un environnement propre à un projet (avec ses codes, ses données et ses fichiers), sur une machine choisie (un serveur ou l'ordinateur d'un développeur). Ce logiciel avait été développé en Shell POSIX<sup>12</sup>, notamment pour être compatible avec la plupart des systèmes Linux, qui sont ceux utilisés sur les ordinateurs et les serveurs de Webu.

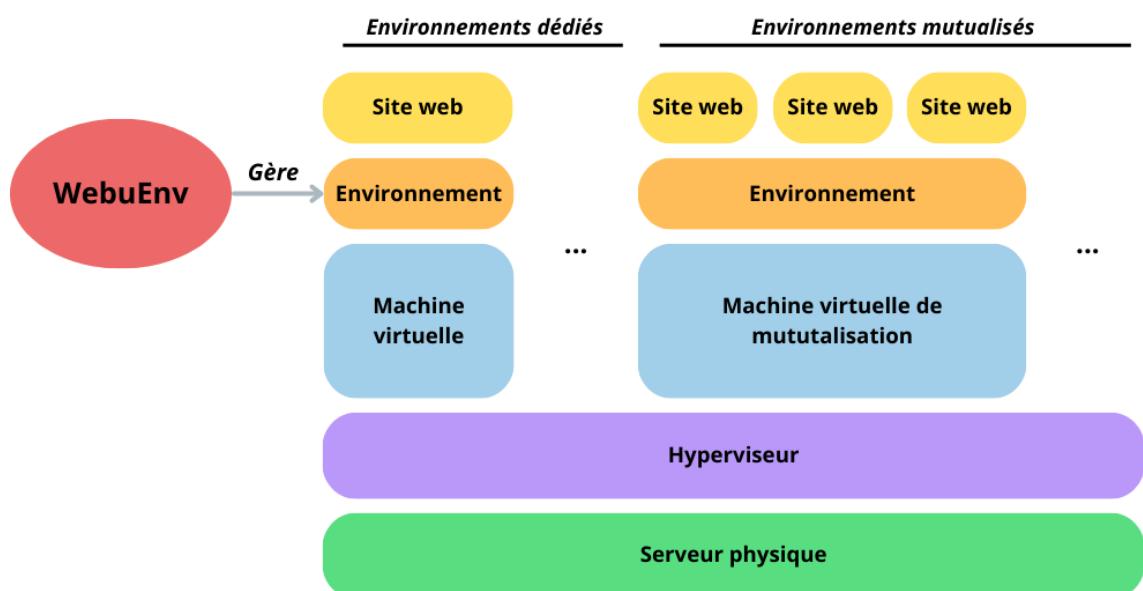


Illustration 3: Schéma de l'infrastructure d'hébergement actuelle

12 **Shell POSIX:** Langage informatique

## b - Les limites actuelles

Cette infrastructure avec de nombreux environnements différents apporte beaucoup de complexité, par exemple, les machines virtuelles étant assez imposantes en termes de ressources (Processeur, Mémoire vive, etc.), il n'est pas concevable d'en créer une par environnement. Il faut alors mutualiser les services, donc installer sur une même machine différents sites pour avoir un environnement commun, ce qui est plus complexe à gérer. Aussi, avec cette mutualisation d'environnements, chaque projet installe ses dépendances dans l'environnement, qui peuvent parfois empiéter sur les autres projets ou bien les ralentir. De plus, l'infrastructure en production et en développement est très différente, ce qui peut poser des soucis de cohérences entre ces environnements. Également, pour mettre à jour la version du langage d'un site, il faut transférer le site de l'ancien environnement au nouveau, ce qui peut être compliqué et délicat à effectuer.

## c - Mes missions

En réponse aux problèmes de l'infrastructure existante, Webu a décidé de la revoir en utilisant cette fois des conteneurs<sup>13</sup> Docker, qui contiendront un environnement par projet, qui sera le plus semblable possible entre la production et le développement. Le choix s'est porté sur Docker, car il permet de déployer facilement un service avec son propre environnement, qui aura été préparé en amont. Pour cela, il faut également modifier l'outil WebuEnv, mais cette tâche étant trop complexe, il a été décidé de développer un nouvel outil : WebuNG, pour Webu Next Generation. Ce logiciel serait alors codé en Python, car cela respecte toujours la compatibilité avec Linux, et car le langage est plus développé et complet que le Shell, ce qui permet d'implémenter plus de fonctionnalités, est plus facile à maintenir et à faire évoluer.

C'est donc dans ce contexte que j'effectue mon stage, où je dois répondre au mieux aux besoins de Webu concernant leur infrastructure d'hébergement. Je vais alors avoir plusieurs missions, dont 2 principales, ayant duré environ 7 semaines chacune :

- Commencer par mettre en place les nouveaux environnements sous Docker.
- Développer la fonctionnalité de WebuNG permettant d'installer une instance sur une machine.

J'ai aussi eu pour objectif de développer d'autres fonctionnalités de WebuNG comme la mise à jour et la désinstallation d'instance, la connexion au terminal de l'instance, etc. Cependant, je ne pourrais pas détailler ces missions dans ce rapport.

---

<sup>13</sup> **Conteneur:** Machines virtuelles légères, qui contient une application et son environnement d'exécution.

## 2 - Mise en place de la nouvelle infrastructure d'hébergement

### a - Mon objectif

Avec la nouvelle infrastructure d'hébergement, le but est, en une seule commande, de lancer/arrêter une instance, avec son environnement et le projet associé. Cela nécessite donc de préparer les environnements en amont. Webu avait déjà réfléchi à une structure de base pour les environnements sous Docker, avec beaucoup de spécificités, les plus importantes ci-dessous (les autres seront pour la plupart détaillées par la suite) :

- Avec un environnement pour chaque type de projet (Python 2.7, Python 3.11, PHP, etc.)
- Mis en place dans des images<sup>14</sup> Docker « de base ».
- Chaque instance sera donc représentée par un conteneur Docker, qui va hériter de l'image correspondante à l'environnement du projet.
- Les instances doivent, en fonction de la commande, pouvoir être lancé dans un contexte de développement ou de production et s'y adapter.

Mon objectif est alors de mettre en place un environnement sous Docker pour un type de projet : Un site web en Python Django, avec une base de données PostgreSQL. Les autres environnements seront assez semblables dans leur mise en place, c'est pourquoi je ne m'occupe que d'un seul type de projet.

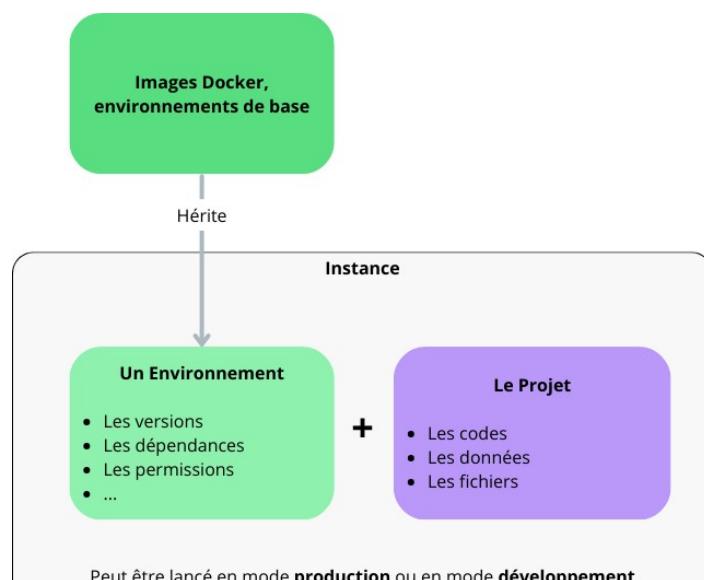


Illustration 4: Schéma de la constitution d'une instance de projet

<sup>14</sup> **Images Docker:** Une pré-configuration de conteneur, utilisée pour lancer un ou plusieurs conteneur

## b - Les tâches effectuées

Pour créer une image Docker, il faut écrire un « Dockerfile », qui est un fichier de script<sup>15</sup>, permettant de pré-configurer un conteneur en partant d'une image existante. J'ai donc préparé un Dockerfile pour créer un environnement Django de base, en partant d'une image officielle<sup>16</sup> avec Python préinstallé, puis en y ajoutant Django. Les images officielles sont assez fiables et les utiliser évite un lourd travail de maintenance qui est déjà effectué par des professionnels. C'est pourquoi j'ai choisi de partir de cette base plutôt que de recréer une image à zéro. J'ai ensuite effectué de nombreuses configurations, comme l'installation d'outils pour l'environnement ou la création d'utilisateurs<sup>17</sup> plus sécurisés. En plus de l'image Django, je vais créer 2 autres images, PostgreSQL et Apache (j'expliquerai leur utilité plus loin.), que je vais configurer de la même manière.

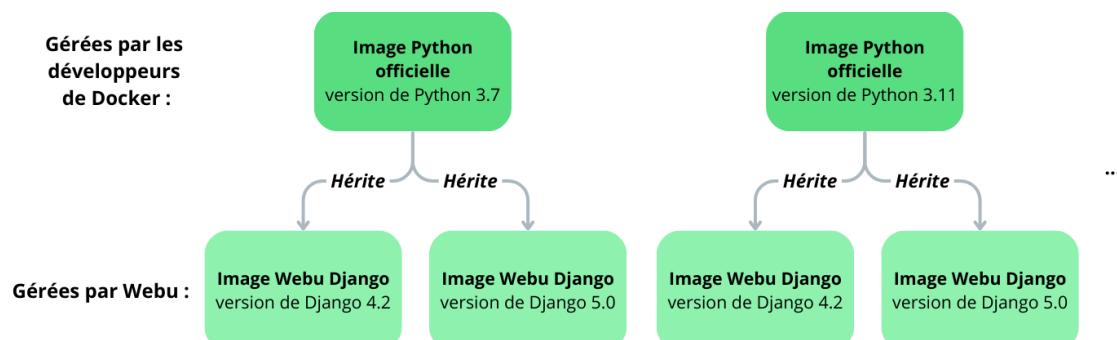


Illustration 5: Héritage des images Docker

En créant ces images, je me suis rendu compte qu'une grande partie de leurs configurations était communes. De plus, pour les futures images, il y aura également des sections semblables. Par exemple, j'ai besoin d'installer un même outil sur toutes les images et de créer des utilisateurs communs. C'est pourquoi j'ai mis en place un outil qui permet de générer des Dockerfiles à l'aide d'un script Ansible<sup>18</sup>. Je vais alors pouvoir écrire des bouts de configurations qui seront insérés dans les Dockerfiles. Cet outil me permet de centraliser les configurations, et ainsi de limiter les erreurs et les oubli.

15 **Script**: Fichier contenant dans l'ordre, les instructions à effectuer

16 **Image officielle**: Image Docker publique, vérifiée et approuvée par les développeurs de Docker

17 **Utilisateur**: Compte d'utilisateur avec des permissions définies, sur un appareil informatique

18 **Ansible**: Langage permettant d'écrire des scripts

Après avoir écrit les Dockerfiles, il faut construire<sup>19</sup> les nouvelles images et les pousser<sup>20</sup> sur leur dépôt. Ces tâches sont assez longues à effectuer manuellement, mais sont simples à automatiser. C'est pourquoi j'ai utilisé une CI/CD,<sup>21</sup> qui va, à chaque changement majeur dans les Dockerfiles, construire et pousser automatiquement les images ayant été modifiées. Cela m'a permis de gagner beaucoup de temps, surtout au début, lorsque je modifiais mes images un grand nombre de fois dans la même journée.

Enfin, quand les images ont été créées, il faut pouvoir lancer les différents conteneurs dont on a besoin dans l'environnement, et assurer la communication entre eux. Alors, j'ai utilisé l'outil « docker compose », qui va permettre de lancer plusieurs services en même temps, qui sera au final l'instance de projet que je dois lancer. Je vais donc configurer cette structure de conteneurs, qui contiendra dans mon cas :

- Un conteneur Django : qui contient le site web en lui-même.
- Un conteneur PostgreSQL : qui contient les données qu'utilise l'application.
- Un conteneur Apache : qui contient un serveur web, permettant de gérer les requêtes<sup>22</sup> en destination du site, soit les rediriger vers l'application web, soit directement servir les fichiers, sans faire un détour par le conteneur Django.

Ces conteneurs seront lancés dans un même réseau et communiqueront entre eux pour permettre de lancer l'application complète. Ci-dessous un schéma simplifié de l'architecture Docker :

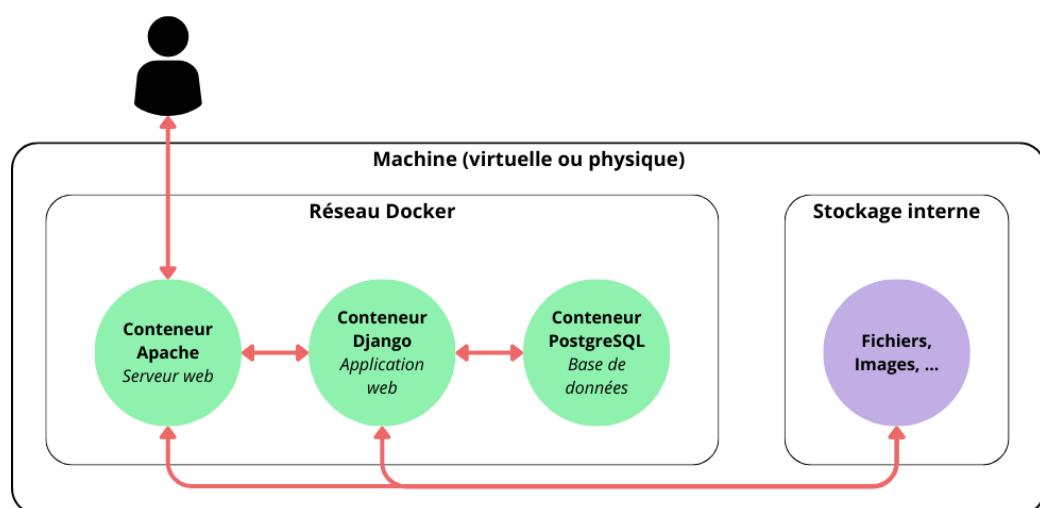


Illustration 6: Schéma simplifié de l'architecture d'une instance avec Docker

19 **Construire:** Pour une image Docker, c'est l'action de la créer à partir du Dockerfile.

20 **Pousser:** Pour une image Docker, c'est l'action de la transférer sur un dépôt pour la stocker.

21 **CI/CD:** Outil permettant d'automatiser les tâches du développement.

22 **Requête:** Une demande ou une action à destination d'un service.

### **c - Résultats et conclusion**

Pour cette mission, j'ai réussi à mettre en place l'infrastructure Docker pour déployer d'un coup un projet avec son environnement, en répondant aux besoins primaires énoncés en introduction de partie, et en respectant la plupart des contraintes qui m'étaient fixées. J'ai également apporté mes idées au projet, notamment en trouvant des solutions à certains problèmes imprévus. De plus, j'ai fait de nombreuses démonstrations à mes référents, pour leur faire suivre mon avancée et poser mes questions lorsque les problèmes étaient trop complexes ou me prenaient trop de temps.

Cependant, cette mission étant assez longue et complexe, je me suis concentré sur l'objectif principal, mais je ne pense pas avoir totalement terminé, notamment, car il reste de nombreux points à améliorer. Par exemple, il faudrait à mon sens revoir quelques aspects de la sécurité au niveau des conteneurs ou encore rédiger une documentation complète et détaillée, ce que j'ai fait de manière assez simple et brève pour pouvoir me concentrer sur mes tâches principales.

## 3 - Développement de fonctionnalités de WebuNG

### a - Mon objectif

Comme énoncé plus tôt, Webu avait besoin de revoir son outil WebuEnv, et donc de développer le nouveau WebuNG pour gérer la nouvelle infrastructure d'hébergement. Cet outil, une application TUI<sup>23</sup> développée en Python, doit permettre d'automatiser les tâches des salariés de Webu concernant l'hébergement des projets, et doit donc être utilisable et compréhensible par tout le monde, les développeurs comme l'administrateur systèmes et réseaux, et même les chefs de projets.

L'équipe de Webu avait déjà réalisé un POC<sup>24</sup>, où ils avaient développé les bases de l'application, ainsi que quelques fonctionnalités simples comme par exemple lister tous les projets. Pour cela, ils avaient notamment utilisé « Cement », un framework Python, permettant de nombreuses fonctionnalités pour aider au développement d'une application TUI. Il permet par exemple de créer des commandes, en configurant par exemple son nom, ses paramètres et sa description, puis on pourra coder le script Python qui sera exécuté au lancement de la commande.

Alors, en reprenant leur travail, mon objectif est de développer la fonctionnalité de WebuNG permettant d'installer sur une machine, une instance d'un projet avec le nouvel hébergement Docker.

---

<sup>23</sup> **Application TUI:** Application avec une interface utilisateur directement sur le terminal de commande.

<sup>24</sup> **POC:** « Proof Of Concept », est une expérimentation permettant de tester les fonctionnalités d'une application.

## b - Les tâches effectuées

Pour commencer, j'ai passé un moment à analyser ce qui a été fait par l'équipe de Webu sur WebuNG. Cela m'a surtout permis de mieux comprendre de quelle manière je dois développer, car le code est assez complexe. J'ai eu du mal à comprendre de nombreuses parties du code, mais j'ai pris le temps de me documenter et poser des questions à mon tuteur, ce qui m'a grandement aidé. Alors, en ayant mieux compris le fonctionnement, j'ai pu m'intégrer plus facilement au développement de l'application.

Ensuite, sur les conseils de mon tuteur, j'ai également pris le temps de réfléchir à la commande, notamment à l'objectif concret, avec quelles étapes, et de quoi elle a besoin. Cela était nécessaire pour que je sache clairement quoi faire, sans me perdre et aller trop loin. De plus, j'ai choisi de commencer par développer le fonctionnement d'installation en développement, en laissant de côté l'installation en production. J'ai fait ce choix pour séparer le travail et avancer plus simplement, avec des étapes moins complexes. J'ai alors décrit le fonctionnement de la commande d'installation d'instance en développement sous la forme d'un schéma. Ci-dessous un schéma très simplifié, avec seulement les grandes étapes, un schéma plus complet est disponible en annexes.

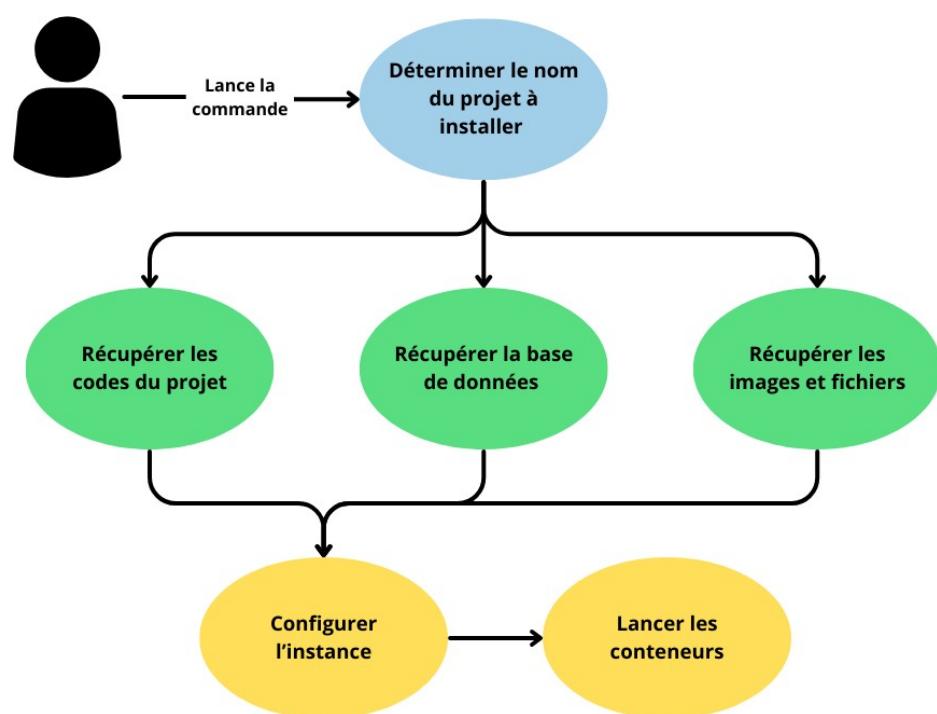


Illustration 7: Schéma simplifié des étapes de l'installation d'une instance en développement

Ensuite, j'ai suivi mon schéma pour développer le script. J'ai d'abord fait en sorte de le faire fonctionner, qu'il réussisse à installer le projet sur mon ordinateur. J'ai fait cela pour poser les bases du script et me concentrer sur le fonctionnement principal, pour passer aux améliorations après. Alors, j'ai ensuite ajouté plusieurs fonctionnalités dont Webu avait besoin sur cette commande, par exemple pouvoir choisir la version des codes ou la base de données à installer, proposer une liste de projets à l'utilisateur pour éviter de taper le nom à chaque commande. Pour trouver les bonnes fonctionnalités à ajouter, j'ai surtout essayé l'outil WebuEnv et discuté avec l'équipe pour savoir ce dont ils avaient besoin.

Également, j'ai passé un moment à réfléchir à tous les cas d'utilisation possibles de la commande, pour la préparer dans le but de gérer ces différents cas. Par exemple, si une instance est déjà installée, si le serveur de fichier n'est pas joignable, si le projet recherché n'existe pas, etc. Je suis alors repassé sur tout le code pour trouver les éventuels cas à gérer et erreurs à traiter. Ensuite, j'ai cherché les solutions les mieux adaptées à chaque cas, par exemple, lorsque le projet est déjà installé, je propose à l'utilisateur de mettre à jour l'instance, de la supprimer au préalable, ou de simplement annuler l'installation. Dans certains cas, la meilleure solution est simplement d'arrêter le script et de décrire l'erreur à l'utilisateur, pour qu'il en comprenne la cause. Par exemple, lorsque le serveur de fichier est injoignable, j'arrête le script et j'explique via un message à l'utilisateur que l'erreur vient certainement d'un problème au niveau du réseau informatique, car je ne peux pas le résoudre dans mon script. Penser à toutes ces éventualités est très important pour que ma commande puisse être exécutée proprement dans la plus de contexte possible. Par ailleurs, j'ai très certainement oublié quelques cas, mais pour limiter cela, j'ai pris beaucoup de temps pour tester ma commande dans de nombreux contextes différents.

### **c - Résultats et conclusion**

Lors de cette mission, j'ai pu développer une commande fonctionnelle et assez complète de WebuNG, en suivant les nombreuses contraintes et demandes de mon tuteur. Notamment, j'ai développé mes connaissances dans le langage Python en l'utilisant dans un contexte bien plus complexe que ce qu'il y a eu l'habitude de voir auparavant, mais surtout, j'ai acquis énormément de bonnes pratiques en développement logiciel.

Cependant, je n'ai pas pu répondre à toutes les contraintes qui m'étaient confiées, car j'ai passé bien plus de temps que prévu à comprendre le code et m'intégrer au développement, mais aussi à tenter de perfectionner au mieux ce que je faisais, car cela me paraissait plus important. Par exemple, je n'ai pas développés de tests<sup>25</sup>, pour ma fonction, car il me restait peu de temps et je n'ai pas voulu bâcler ce travail qui est assez important.

---

<sup>25</sup> **Tests:** Des programmes permettant de tester des parties du code de l'application.

## 4 - Conclusion

Pour conclure, lors de mon projet, j'ai réussi comme prévu à poser les bases de la nouvelle infrastructure d'hébergement avec Docker, et j'ai pu développer des fonctionnalités du futur outil WebuNG, notamment l'installation d'instance, mais également la mise à jour et la désinstallation. Il reste cependant encore de nombreuses choses à améliorer, sur les 2 parties, comme j'ai pu le décrire plus tôt. Il reste aussi énormément de choses à ajouter, comme de nouveaux environnements à héberger ou de nouvelles commandes de WebuNG. C'est pourquoi j'ai pris le temps de bien expliquer à l'équipe tout ce que j'ai fait, pourquoi et comment, pour qu'ils puissent reprendre mon travail pour terminer le projet.

Également, j'ai pu participer à des missions qui n'étaient pas prévues initialement, notamment l'installation des nouveaux serveurs dans le datacenter, ce qui m'a appris énormément de choses, toujours en lien avec la mise en place d'une solution d'hébergement, mais cette fois au niveau des équipements physiques.

## IV - Bilans

### 1 - Bilan professionnel

Lors de mon stage, j'ai acquis beaucoup de compétences techniques, dans plusieurs domaines :

- J'ai d'abord appris à mettre en place une solution d'hébergement avec Docker, notamment en comprenant les contraintes à respecter pour avoir une infrastructure fiable et efficace.
- Également, je me suis grandement amélioré avec le langage Python, et plus généralement, j'ai appris à mener à bien un projet de développement logiciel, surtout grâce à mon intégration au développement de WebuNG.
- De plus, j'ai pu voir comment fonctionne un datacenter, et comment le matériel physique y est installé, et les bonnes pratiques à avoir lors d'une intervention directement sur les serveurs.
- Aussi, j'ai pu appliquer les connaissances que j'ai acquises en BUT en créant des outils d'automatisation avec Ansible et les CI.
- Ensuite, grâce aux nombreuses réunions et discussions, et grâce à l'outil GitLab, je me suis plongé dans un vrai environnement de gestion de projet de développement, ce qui m'a surtout fait réaliser à quel point cela est important.
- Enfin, en écoutant et en voyant mes collègues, j'ai pu comprendre quelles sont les étapes de création d'un site web, et comment les mener à bien, de la conception du projet à l'hébergement.

Toutes ces compétences me permettent aujourd'hui de mettre en place une réelle infrastructure informatique et de m'intégrer dans une équipe développement, en connaissant l'environnement et les enjeux.

## 2 - Bilan personnel

En plus des nombreuses compétences techniques que j'ai acquises grâce à mon stage, j'ai développé mes compétences personnelles en entreprise. J'ai appris à mieux m'organiser pour mener des projets, notamment en développant de nombreuses bonnes pratiques qui rendent mon travail plus efficace. Je suis également plus autonome, tout en osant poser les bonnes questions au bon moment, ce que j'ai toujours eu beaucoup de mal à faire, par peur de mal faire et de déranger. De plus, j'ai énormément apprécié le projet qui m'a été confié, ce qui m'a montré que pour des projets vraiment intéressants, je suis réellement motivé, et prêt à surmonter les difficultés.

Enfin, j'ai découvert une nouvelle façon de gérer une entreprise. En effet, comme je l'ai expliqué au début, Webu est une SCOP qui est dirigée de manière égale par tous les associés. Lors de mon stage, 7 des 8 salariés étaient associés, car le dernier a été embauché récemment. Cette gestion permet à tout le monde de s'exprimer et de donner son opinion, donc à Webu de faire de meilleurs choix. Pratiquement toutes les décisions de l'entreprise doivent être discutées, ce qui crée des réunions sur beaucoup de sujets différents, auxquelles j'ai souvent été convié. Cela m'a permis de m'intégrer au mieux dans l'entreprise. En comparant cette expérience aux précédentes, dans des entreprises plus classiques, j'ai remarqué tout de suite que ce type de gestion possède de nombreux avantages, comme par exemple des meilleures conditions de travail pour les salariés, une meilleure ambiance générale ou encore une meilleure entente. Je pense également que cela favorise grandement la qualité des sites que Webu créé. La gestion en coopération possède aussi quelques défauts, comme par exemple la prise de décision qui est plus longue. Mais cela est, à mon sens, moins important que tout les avantages que ce type d'entreprise possède.

## V - Annexes

### Table des Annexes

Figure 1: Schéma complet des étapes de la commande d'installation d'instance.....24



Figure 1: Schéma complet des étapes de la commande d'installation d'instance

## VI - Résumés

Actuellement en 3ème année de BUT réseaux et Télécommunications, j'ai effectué mon stage de fin de formation chez WEBU, du 26 février 2024 au 14 juin 2024. Webu est une entreprise de développement web, qui crée des sites internet sur-mesures pour ses clients. Lors de mon stage, je me suis occupé de mettre en place les bases d'une nouvelle infrastructure d'hébergement, avec des conteneurs Docker. Je me suis également intégré au développement d'un outil permettant d'automatiser de nombreuses tâches, en lien avec la nouvelle infrastructure que j'ai mise en place. Ce stage m'a permis de mêler l'aspect administration systèmes et réseaux au développement logiciel, en m'apprenant énormément de choses, techniques ou plus générales. J'ai également découvert un nouveau type d'entreprise, qui favorise la coopération entre tous les salariés, et qui m'a montré que décider à plusieurs, même pour une structure assez imposante comme Webu, possède de nombreux avantages pour tout le monde, les salariés comme les clients.

Currently in my 3rd year of BUT networks and telecommunications, I did my final internship at WEBU, from February 26, 2024 to June 14, 2024. Webu is a web development agency that creates customized websites. For my internship, I was in charge of setting up the foundations of a new hosting infrastructure, using Docker containers. I was also invited to take part in the development of a tool to automate tasks related to the new infrastructure. This internship has enabled me to combine systems and network administration with software development, teaching me a lot of things, both technical and more general. I also discovered a new type of company, which favors cooperation between all the members, and which showed me that making decisions in group, has many advantages for everyone, employees and customers.