ECE PARIS · LYON
ÉCOLE D'INGÉNIEURS

Rapport de stage de fin d'études

à retourner au plus tard **1 semaine avant la date de la soutenance**

	Rapport de Stage **Ing5 - 3ème année Cycle Ingénieur** **Promotion 2021**
Nom : Hoffmann **Prénom : Alexander**	**Majeure :** ❒ ENE ☒ SI/BDA/CYD SE/VCA ❒ SAN ❒ FIN ❒ OCRES

Entreprise d'accueil

Nom : Adaltas

Adresse : 6 Rue Jules Simon, 92100 Boulogne-Billancourt

Adresse du lieu de stage si différent :

Engagement de Confidentialité : ❒ oui ☒ non

Reçu le ..... /...../.....

Rapport Confidentiel à remettre au tuteur de stage à l'issue de la soutenance : ❒ oui ☒ non

_____

Pénalités Observées :

Description de la mission

Dans le cadre du stage, l'étudiant participera à l'architecture et au déploiement des différents composants d'une plateforme PAAS en prenant en compte les impératifs de sécurité, de tolérance aux pannes et de performances. Pour assurer l'exploitation de la plateforme, il mettra en place des chaînes de traitement en streaming pour collecter, traiter, afficher et alerter à partir des évènements émis par le système.

Rapport de stage

Entreprise d'accueil :
Adaltas

MISE EN PLACE D'UNE SOLUTION D'AUTOMATISATION POUR LE DÉPLOIEMENT DE SYSÈMES BIG DATA

Auteur :
Alexander Hoffmann - alexander@hoffmann.ai

Maître de stage :
David Worms - david@adaltas.com

28 juin 2021

Remerciements

Les premiers pas dans le monde du travail se font rarement seul et sans aide ni soutien. Je souhaite remercier toutes les personnes ayant contribué à cette expérience professionnelle.

Je tiens tout d'abord à remercier toute l'équipe d'Adaltas pour son accueil, sa bienveillance et sa bonne humeur permanente. J'apprécie l'attention et la sollicitude qui m'ont été prodiguées par toute personne rencontrée.

Je voudrais ensuite exprimer ma sincère gratitude à David Worms, mon tuteur, pour la confiance qu'il a bien voulu m'accorder en acceptant de m'intégrer à Adaltas. Je le remercie pour sa grande disponibilité, sa patience, son soutien chaleureux et ses conseils avisés. Je tiens à lui exprimer ma profonde reconnaissance pour ses critiques constructives d'une rigueur absolue.

Mes remerciements s'adressent également à Prisca Borges pour son accueil, sa sympathie et ses conseils, ainsi qu'à Léo Schoukroun pour son encadrement et sa compréhension qu'il m'a accordé tout au long du stage.

En cette période inédite de crise sanitaire, j'ai eu la chance de pouvoir travailler avec une entreprise qui a su s'adapter aux difficultés posées par les mesures nécessaires à la protection de ses collaborateurs. Je suis reconnaissant d'avoir pu effectuer mon stage dans les meilleures conditions possibles.

Résumé

Ce rapport décrit les travaux effectués lors de mon stage de 6 mois chez Adaltas. J'ai travaillé sous la supervision de David Worms en tant que Infrastructure and Security Operations (InfraOps) Intern. En tant qu'ingénieur InfraOps, j'ai aidé à construire des systèmes automatisés, sécurisés, fiables et observables, ainsi que des processus permettant aux autres équipes d'Adaltas d'utiliser efficacement notre infrastructure pour lancer, exploiter et fournir des produits et services de grande qualité destinés aux clients et couvrant de multiples écosystèmes b2c, partenaires et vendeurs à travers le monde.

Mots clés : Big Data, DevOps, InfraOps.

Table des figures

Table des matières

Chapitre 1

Introduction

Mon intérêt pour la data science et plus généralement pour l'informatique et les nouvelles technologies m'a amené à chercher un stage dans une société qui en a fait son cœur de métier. Au sein de l'équipe InfraOps d'Adaltas, j'ai trouvé une opportunité unique pour mettre à profit mes compétences en matière de développement et de conception. J'ai eu la chance de participer à la création de systèmes et d'outils d'infrastructure permettant la mise en production de nouveaux produits qui s'étendent sur des millions d'utilisateurs. Pour assurer l'exploitation de la plateforme, j'ai mis en place des chaînes de traitement en streaming pour collecter, traiter, afficher et alerter à partir des évènements émis par le système.

La thématique de ce stage s'inscrit dans un contexte d'appréhension et d'approfondissement du monde de l'entreprise. Dans ce rapport, nous présenterons dans un premier temps, le contexte du stage, c'est-à-dire que nous décrirons l'entreprise d'accueil, nous étudierons son secteur d'activité et sa culture. Dans un second temps, nous traduirons les différents aspects de ma mission et les attentes de l'entreprise. Finalement, nous verrons les compétences et qualités acquises à la suite de ce travail ainsi que ma valeur ajoutée pour Adaltas.

Chapitre 2

Présentation de la structure d'accueil : Adaltas

Fondée en 2004, Adaltas est une société d'expertise en High Tech construite à partir de deux idées :

- l'innovation comme facteur de différenciation décisif pour les entreprises ;
- la capacité à mobiliser les meilleurs talents comme condition de succès.

FIGURE 2.1 – Logo d'Adaltas représentant un oiseau. Adaltas signifie "vers le haut".

Adaltas aide ses clients à s'orienter dans le monde en perpétuelle évolution de l'Open Source, leur donnant les clés pour développer les meilleures solutions, qu'il s'agisse simplement d'écrire une application ou d'élaborer une plateforme de traitement stratégique à plus long terme. L'entreprise se définit comme un acteur du Big Data autour des technologies Hadoop et NoSQL.

Les équipes apportent une expertise sur l'analyse et le traitement des données, leur gouvernance, le développement et la gestion opérationnelle. Les consultants adhèrent à la culture DevOps et ils sont formés à la méthodologie SRE [1]. Ils accompagnent leur client dans la mise en place d'infrastructures et d'applications résilientes, conscients des rapides innovations apportés par la communauté Open Source et de la nécessaire stabilité des systèmes.

L'expertise d'Adaltas dans le domaine Big Data a commencé dès 2009 par l'accompagnement de la société EDF et la collecte des données Linky dit compteurs intelligents.

1. What is Site Reliability Engineering (SRE)?

En 2012, la société a entrepris l'exploitation d'une plateforme commune à l'ensemble du groupe EDF avec la mise à disposition des composants de l'éco-système Hadoop. Le nombre de composants s'est élargi avec le temps ainsi que les services et les cas d'usage qui ont accosté sur la plateforme sécurisée et multi-tenante.

Depuis 2014, l'équipe s'est élargie en accueillant des talents majoritairement formés à l'ECE, école dans laquelle Adaltas est à l'initiative du programme Big Data. Adaltas donne également des cours au Data Science Tech Institute[2] et à l'Université Paris-Sorbonne.

2.1 Objectifs de l'entreprise

L'acquisition d'un cluster à forte capacité répond à la volonté d'Adaltas de construire une offre de type PAAS pour disposer et mettre à disposition des plateformes de Big Data et d'orchestration de conteneurs. Les plateformes sont utilisées pour l'acquisition de nouvelles compétences, l'évaluation de nouvelles technologies, l'utilisation d'outils DevOps et la mise à disposition d'environnements de développement, de PoCs et d'exploitation. Elles hébergent des Data Lakes, des DataLabs, des traitements et des modèles de Data Science, des outils orientés DevOps ou encore des services applicatifs. L'objectif est de porter cette offre à maturité.

Dans le cadre de ses cours et formations, Adaltas s'intéresse au domaine Big Data et Data Science. Les cours effectués au seins des différents établissements ont pour objectif de trouver des jeunes talents pour les faire monter en compétence. Ainsi, la société cherche à recruter et former ses futurs consultants le plus tôt possible afin qu'ils soient opérationnels dès la fin du stage de dernière année d'école.

2.2 Une entreprise "open"

Adaltas est une société "open". Leur engagement se construit sur les fondations d'un code source ouvert, d'une collaboration ouverte, de standards ouverts et d'une formation ouverte.

Les entreprises et les gouvernements utilisent les technologies Open Source pour remplacer les solutions propriétaires. Initialement, ces acteurs ont été attirés par les réductions de coût et la promesse de s'affranchir de la dépendance d'un éditeur. L'Open Source est désormais central à la transformation digitale avec des avantages avérés dans la sécurité, la qualité, la personnalisation, la flexibilité, l'intéropérabilité, l'auditabilité et le soutien.

Adaltas maintient près de 50 dépôts Open Source sur GitHub et encourage chaque développeur et client à contribuer à ces projets.

2. https://www.datasciencetech.institute/

2.3 La culture d'Adaltas

Adaltas préserve un esprit familial qui privilégie toujours une vision à long terme. L'entreprise a pour vocation d'assurer le développement de chacun de ses consultants dans le respect de leur identité et de leur autonomie en mettant à disposition toutes les ressources nécessaires. Chaque service ou fonctionnalité proposé est le fruit d'une collaboration où chacun contribue aux idées des autres. L'objectif principal est de créer les meilleures expériences possibles pour les clients.

Le respect de ces valeurs est l'une des clefs de la performance d'Adaltas, de son ancrage dans l'air du temps et dans la société qui l'entoure.

2.4 RSE : Responsabilité Sociétale de l'Entreprise

	Evaluation RSE	Oui/ Non/ N-A	Justifications, Commentaires
Gouvernance			
Code de conduite de l'entreprise s'appuyant sur ses valeurs pour la prise de décision et pour définir sa stratégie			
1	L'entreprise a défini des valeurs partagées par l'ensemble des salariés et en adéquation avec les principes du Dev Durable	Oui	Les valeurs d'Adaltas sont le partage des connaissances et de l'expérience entre employés pour nourrir chaque talent. Ces valeurs n'entrent pas en conflit avec les principes du développement durable.
2	La stratégie est communiquée au sein de l'entreprise et est en lien avec le développement durable	Non	Adaltas est une entreprise trop petite pour adopter une stratégie en lien avec le développement durable.
3	L'entreprise a mis en place des indicateurs (sociaux, environnementaux, économiques) pour piloter son activité	Non	Adaltas est une entreprise trop petite pour que la mise en place de tels indicateurs soit intéressante.
4	L'entreprise communique ses actions et décisions de façon transparente vers l'ensemble des parties prenantes	Oui	Étant donné la taille de l'entreprise, la communication entre la direction et les employés est directe. Le dirigeant d'Adaltas se présente d'ailleurs comme un employé de l'entreprise.
Droits de l'Homme			
L'entreprise applique l'ensemble des droits de l'Homme que ce soit les droits fondamentaux (santé, dignité, travail des enfants…) mais également elle doit s'efforcer d'éviter toute discrimination et exclusion envers les personnes les plus vulnérables			
1	L'entreprise a bien identifié les risques de non-respect des principes énoncés dans la charte des droits de l'homme	N-A	Adaltas ne fait appel à aucun sous-traitant de matériel et les solutions techniques utilisées par ses employées sont soit développées par des éditeurs reconnues, soit Open Source.
2	L'entreprise est organisée pour faire remonter toute information en cas d'atteinte constatée aux droits de l'homme	Oui	Adaltas met un place un suivi personnalisé et s'engage à défendre les droits des consultants dans le cadre de leurs missions chez ses clients. La remontée d'information est encouragée par la proximité entre les employés et le dirigeant.
3	L'entreprise a mis en place des actions en faveur de la diversité et notamment pour les personnes en situation d'handicap, et pour l'égalité professionnelle	N-A	Adaltas est une société trop petite pour qu'une politique favorisant la diversité soit mise en place.

4	L'entreprise veille au respect des droits fondamentaux (civils, politiques, économiques et culturels)	Oui	La facilité de communication entre les employés permet une remontée efficace des excès éventuels. Avant tout événement organisé, Adaltas prend en compte les restrictions, religieuses ou culturelles, de chaque employé et s'assure qu'aucun ne sera discriminé.

Relations et conditions de travail			
Ce principe regroupe les questions relatives au recrutement, au respect du code du travail, aux dispositions en place pour créer un climat social sain et respectueux, et à la sécurité au travail.			
1	Les relations Employeur-Employés sont basées sur le respect des principes d'égalité et le respect des droits et devoirs de chacun	Oui	Adaltas met un place un suivi personnalisé et s'engage à défendre les droits des consultants dans le cadre de leurs missions chez ses clients. La remontée d'information est encouragée par la proximité entre les employés et le dirigeant.
2	L'entreprise met en place les conditions de travail appropriées aux missions et une protection sociale convenable	Oui	Adaltas s'assure que ses clients, qui sont de grands groupes français, assurent les conditions de travail de ses consultants. Les salariés sont protégés par une mutuelle payée en partie par l'entreprise.
3	L'entreprise favorise le dialogue social avec son personnel et ses instances représentatives du personnel.	N-A	Adaltas est une trop petite entreprise pour avoir des instances représentatives du personnel.
4	L'entreprise prend toutes les mesures nécessaires pour garantir la santé et la sécurité de son personnel	Oui	Adaltas s'assure que les clients accueillent ses consultants qui garantissent la santé et la sécurité du personnel.
5	L'entreprise a mis en place une politique de gestion prévisionnelle des emplois et compétences	Oui	Adaltas s'assure de la pérennité des besoins clients avant d'engager de nouveaux consultants.
L'environnement			
L'entreprise doit avoir une démarche responsable quant à l'impact de son activité sur l'environnement : elle doit avoir une utilisation durable des ressources, et prévenir (qualifier et quantifier) toute pollution			
1	L'entreprise a qualifié toute type de pollution inhérent à son activité et prend les mesures nécessaires pour les prévenir et les réduire	N-A	L'activité d'Adaltas ne produit aucune pollution directe car elle se fait uniquement dans le cadre de l'informatique.
2	L'entreprise optimise sa consommation de ressources (matières premières, consommables, énergie)	N-A	Adaltas est actuellement en sous-location, elle ne gère donc pas la consommation d'énergie des locaux. De par son activité, Adaltas utilise très peu de consommables et pas de matières premières.

3	L'entreprise a identifié l'impact des ses activités sur le changement climatique et met en œuvre toutes les mesures pour le réduire	N-A	L'activité d'Adaltas ne produit aucune pollution directe car elle se fait uniquement dans le cadre de l'informatique.

Loyauté des pratiques			
L'entreprise doit agir de façon loyale envers ses concurrents, clients, partenaires, fournisseurs…C'est respecter les règles (concurrence, droits de propriété négociation) et prévenir la corruption,			
1	L'entreprise a mis en place un dispositif visant à lutter contre la corruption	N-A	Adaltas est une trop petite entreprise pour mettre en place ce genre de dispositif.
2	L'entreprise encourage la transparence en matière de politiques publiques. Elle veille à éviter les conflits d'intérêt, les abus d'influence…	N-A	Adaltas est une trop petite entreprise pour mettre en place ce genre de dispositif.
3	L'entreprise à mis en place des procédures garantissant le respect des droits de propriété	Oui	En tant que promoteur de l'Open Source, Adaltas attache une grande importance aux outils utilisés dans le cadre des missions de ses consultants.

Chapitre 3

Présentation de la mission

La mission principale de mon stage consistait à automatiser le déploiement d'un cluster Apache Hadoop.

3.1 Cahier des charges

Sur la base des différents objectifs présentés précédemment, un cahier des charges a été établi courant février 2021 par David Worms. Les activités précisées sont les suivantes :

1. Conception et mise en place d'architectures se basant sur des technologies open-source de l'écosystème Big Data ;
2. Déploiement manuel d'un cluster HDP ;
3. Automatisation de l'installation, la configuration et le déploiement d'un cluster HDP ;
4. Rédaction d'articles pour partager les résultats des recherches.

3.2 Planning

Le découpage temporel des missions proposées au début du stage est décrit sur la figure 3.1. Ce planning a été formulé en fonction de ma progression prévisionnelle de l'apprentissage des technologies Big Data. Lesdites technologies seront étudiées en détail ci-après.

3.3 Environnement de travail

La plus grande partie de mon stage s'est déroulée à distance. Quand j'étais sur les lieux de l'entreprise, j'ai eu l'opportunité de rencontrer et d'interagir de façon privilégiée avec les différentes collaborateurs. Les employés d'Adaltas ont des échanges quotidiens

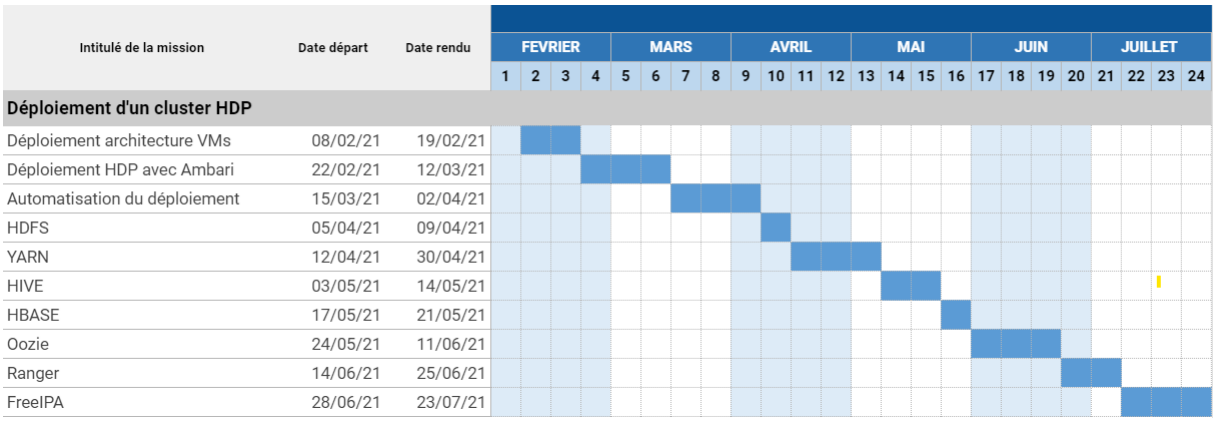

FIGURE 3.1 – Planning sous forme de diagramme de Gantt

via le chat interne de l'entreprise (Keybase[1]). Ainsi, j'ai pu solliciter l'expertise de chacun lorsque j'ai rencontré des difficultés.

Nous avions un meeting tous les deux jours pour faire un point sur l'avancée de nos missions. Cette réunion dure entre 15 et 20 minutes. Chaque membre de l'équipe prend la parole à tour de rôle et décrit au reste de l'équipe ce qu'il a fait la veille, les objectifs qui ont été atteints, ce qu'il prévoit de faire le reste de la journée avec les nouveaux objectifs, et les éventuels problèmes qu'il rencontre. De cette façon, il est facile de savoir qui peut lui venir en aide et comment, afin de résoudre ses problèmes et de lui permettre d'avancer de nouveau.

3.4 Environnement de développement

Pour remplir ma mission, Adaltas a mis à ma disposition un ordinateur à la pointe de la technologie. Il s'agit d'une machine de la marque Dell comportant les caratéristiques décrite dans la table 3.1.

Type de Hardware	Caractéristiques techniques
Processeur	Processeur Intel Core i7-9750H de 9e génération
Disque	Disque SSD hautes performances M.2 PCIE 40 de 1 To
Mémoire	32 Go de mémoire, 2 x 16 Go, DDR4 à 2 666 MHz

TABLE 3.1 – Caractéristiques techniques du matériel mis à disposition.

Ces spécifications techniques sont nécessaires car les consultants sont amenés à créer des clusters Big Data avec plusieurs noeuds nécessitant une plus grande puissance de calcul. Au début de mon stage, j'ai dû installer Arch Linux sur cet ordinateur. L'installation habituellement assez périlleuse car il faut mettre en place un grand nombre de

1. https://keybase.io/

services manuellement (notamment les services réseaux et l'interface graphique). Pour faire face à cela, Adaltas a développé une solution nommée Nikita Arch [2], un logiciel de déploiement pour le système d'exploitation Arch Linux. Arch Linux est plus simple que Debian ou Ubuntu car pacman ne touche pas à la configuration des paquets (ce que fait dpkg). Arch Linux est plus tolérant que Debian à propos des paquets "non-libres" tels que définis par GNU. Il est optimisé pour x86_64, et donc est plus rapide que Debian (i386). Les paquets d'Arch Linux sont plus récents que les paquets Debian. En revanche Debian est largement plus stable, c'est pour ça qu'il est généralement utilisé pour les serveurs.

2. https://github.com/adaltas/node-nikita-arch

Chapitre 4

Installation

Ce chapitre couvre la description et l'installation de tous les outils nécessaires à ce projet. Tous les logiciels et paquets utilisés sont des logiciels Open Source et sont disponibles gratuitement.

4.1 vim

FIGURE 4.1 – Logo de vim.

Vim est un éditeur de texte directement inspiré de vi (un éditeur très répandu sur les systèmes d'exploitation de type Unix). Son nom signifie d'ailleurs Vi IMproved, que l'on peut traduire par « VI aMélioré ». A priori, Vim n'est pas un IDE mais un simple éditeur de texte. Néanmoins, l'ajout d'extensions, ou la modification de son fichier de configuration en fait un environnement de développement optimal. L'avantage est qu'il n'est pas nécessaire de maîtriser plusieurs IDE, Vim suffit.

Etant donné que j'avais déjà quelques notions de Vim avant mon stage, j'ai été chargé de rédiger un tutoriel sur le site du cloud d'Adaltas. Cette introduction est destinée aux personnes n'ayant jamais utilisé Vim. Voici le lien vers mon tutoriel.

4.2 git

FIGURE 4.2 – Logo de git.

Git est un outil de contrôle de version similaire à CVS, Subversion et Mercurial. Cette famille d'outils s'appelle Système de Contrôle de Version (SCV) ou Gestion du Contrôle des Sources (GCS). Un contrôle de version permet de garder une trace des modifications apportées à un ou plusieurs fichiers au fil du temps afin de pouvoir accéder ultérieurement à une version spécifique. Voici quelques exemples : un développeur veut garder une trace de l'évolution de son code ; un ingénieur DevOps veut déclencher des tests sur les changements publiés et déployer de nouvelles versions à partir de points d'accès bien définis de l'historique du logiciel ; un développeur web a besoin de stocker chaque version d'une image ou d'une mise en page ; un ingénieur infrastructure veut stocker et garder une trace de ses procédures de déploiement et des changements de configuration ; un Data Scientist veut enregistrer toutes ses expériences et les évolutions des fonctionnalités. Chez Adaltas, la procédure d'installation des systèmes Arch Linux utilisée sur la majorité de nos ordinateurs portables est stockée et partagée sur un dépôt public.

4.3 VirtualBox

FIGURE 4.3 – Logo de VirtualBox.

Oracle VM VirtualBox est une application de virtualisation multiplateforme open-source. Le logiciel s'installe sur une machine physique basée sur Intel ou AMD, qu'elle fonctionne sous les systèmes d'exploitation (OS) Windows, Mac OS X, Linux ou Oracle Solaris. VirtualBox étend les capacités de la machine hôte afin d'y exécuter plusieurs OS, dans plusieurs machines virtuelles, en même temps. À titre d'exemple, il est possible

d'exécuter Windows et Linux sur Mac, d'exécuter Windows Server 2016 sur un serveur Linux, d'exécuter Linux sur un PC Windows, et ainsi de suite, le tout aux côtés des applications existantes. Il est possible d'installer et d'exécuter autant de machines virtuelles que souhaité. Les seules limites pratiques sont l'espace disque et la mémoire. La capture d'écran de la figure 4.4 montre comment VirtualBox, installé sur un ordinateur Microsoft Windows 10, exécute Ubuntu 20.04 dans une machine virtuelle.

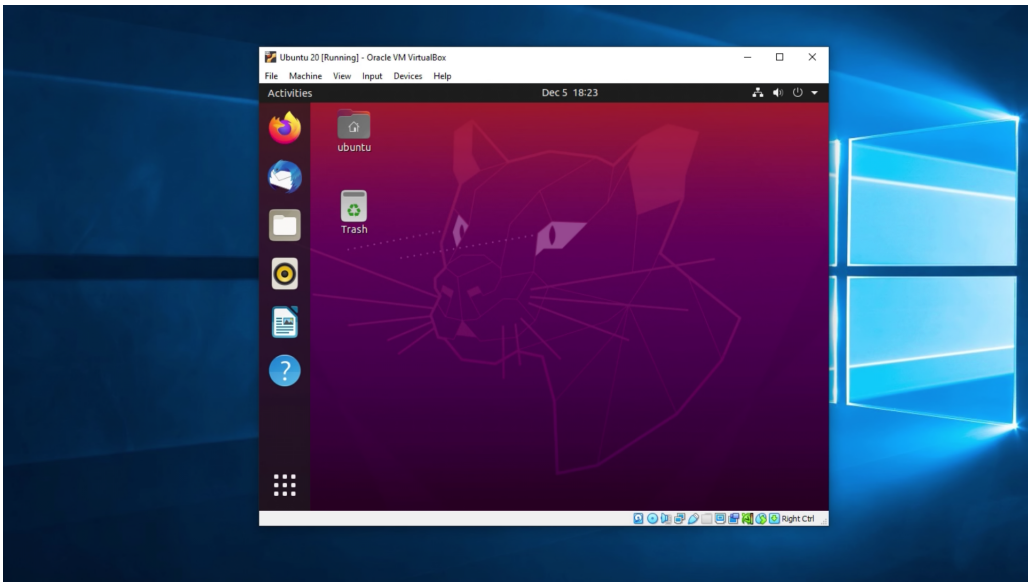

FIGURE 4.4 – Exemple d'utilisation de VirtualBox.

L'utilisation de la virtualisation est un avantage majeur dans le domaine du Big Data étant donné que cela permet de mettre en place des environnements de développement et de test rapidement et de les supprimer sans complexité.

4.4 Vagrant

FIGURE 4.5 – Logo de Vagrant.

Vagrant est un outil permettant de créer et de gérer des environnements de machines virtuelles. Vagrant réduit le temps de configuration de l'environnement de développement et automatise le déploiement de plusieurs machines virtuelles. Vagrant fournit des

environnements de travail faciles à configurer, reproductibles et portables afin d'optimiser la productivité et la flexibilité.

Pour créer et gérer des machines virtuelles, nous utilisons un `Vagrantfile`. La fonction principale du fichier `Vagrantfile` est de décrire le type de machine requis pour un projet, ainsi que la manière de configurer et de provisionner ces machines. Vagrant fonctionne avec un `Vagrantfile` par projet, et le fichier est sauvegardé dans le contrôle de version. Cela permet aux autres développeurs impliqués dans le projet de vérifier le code. Les fichiers `Vagrantfile` sont portables sur toutes les plateformes supportées par Vagrant. La syntaxe des `Vagrantfile` est Ruby, mais la connaissance du langage de programmation Ruby n'est pas nécessaire pour apporter des modifications au fichier, puisqu'il s'agit principalement de simples affectations de variables. La figure 4.6 montre un exemple de fichier `Vagrantfile` utilisé pour un cluster de développement.

```
box = "centos/7"

Vagrant.configure("2") do |config|
  config.vm.synced_folder ".", "/vagrant", disabled: true
  config.ssh.insert_key = false
  config.vm.box_check_update = false
  config.vm.define :master01 do |node|
    node.vm.box = box
    node.vm.network :private_network, ip: "10.10.10.11"
    node.vm.provider "virtualbox" do |d|
      d.memory = 8192
    end
    node.vm.hostname = "master01.nikita.local"
  end
  config.vm.define :worker01 do |node|
    node.vm.box = box
    node.vm.network :private_network, ip: "10.10.10.16"
    node.vm.provider "virtualbox" do |d|
      d.customize ["modifyvm", :id, "--memory", 2048]
      d.customize ["modifyvm", :id, "--cpus", 2]
      d.customize ["modifyvm", :id, "--ioapic", "on"]
    end
    node.vm.hostname = "worker01.nikita.local"
  end
end
```

FIGURE 4.6 – Exemple de fichier `Vagrantfile`.

Les Box sont le format de paquetage des environnements Vagrant. Une Box peut être utilisée par n'importe qui, sur n'importe quelle plate-forme prise en charge par

Vagrant, pour créer un environnement de travail. Le moyen le plus simple d'utiliser une Box est d'en ajouter une à partir du catalogue accessible au public. Pour la création du cluster, j'ai utilisé la Box Centos version 7 car c'est l'une des distributions Linux la plus stable et rapide à configurer.

4.5 Ansible

FIGURE 4.7 – Logo de Ansible.

Ansible est un logiciel d'automatisation informatique qui automatise le provisionnement des clouds, la gestion de la configuration, le déploiement des applications, l'orchestration intra-service et de nombreux autres besoins informatiques. Ansible fonctionne en se connectant aux nœuds et en leur envoyant des scripts, appelés "modules Ansible". Ces programmes sont écrits pour être des modèles de ressources de l'état souhaité du système. Ansible exécute ensuite ces modules (via SSH par défaut), et les supprime une fois terminés.

Par défaut, Ansible représente les machines gérées en utilisant un fichier INI qui place toutes les machines dans des groupes définis. Le fichier que j'ai utilisé pour le déploiement du cluster est décrit sur la figure 4.8.

4.6 Ambari

FIGURE 4.9 – Logo de Ambari.

Le projet Apache Ambari vise à simplifier la gestion d'Hadoop en développant un logiciel pour le provisionnement, la gestion et la surveillance de clusters Apache Hadoop.

Ambari fournit une interface web de gestion intuitive et facile à utiliser, soutenue par ses API RESTful. Ambari permet aux administrateurs système de provisionner un cluster Hadoop grâce à un assistant qui guide l'utilisateur étape par étape pour l'installation des services sur tous les hôtes. De plus, le logiciel fournit une gestion centrale pour le démarrage, l'arrêt et la reconfiguration des services sur l'ensemble du cluster. Enfin, Ambari fournit un tableau de bord pour surveiller la santé et l'état du cluster.

4.7 Hadoop

FIGURE 4.10 – Logo de Hadoop.

Apache Hadoop est un logiciel open-source pour le stockage et le traitement à grande échelle d'ensembles de données sur des clusters. Il est composé des modules suivants :

- Hadoop Common : contient des bibliothèques et des utilitaires nécessaires aux autres modules Hadoop.
- Hadoop YARN : une plateforme de gestion des ressources chargée de gérer les ressources de calcul dans les clusters et de les utiliser pour la programmation des applications des utilisateurs.
- Hadoop MapReduce : un modèle de programmation pour le traitement de données à grande échelle.
- Hadoop Distributed File System (HDFS) : un système de fichiers distribué qui stocke les données sur des machines de base, offrant une bande passante globale très élevée dans le cluster.

4.7.1 Stack Hadoop

Hadoop et ses différents composants s'assemblent pour garantir un modèle de stockage et de gestion des Big Data tolérant aux pannes, durable et hautement efficace. Un cluster Big Data peut être décomposé en plusieurs noeuds :

Namenode Namenode est le nœud qui stocke les métadonnées du système de fichiers, c'est-à-dire quel fichier correspond à quel emplacement de bloc et quels blocs sont stockés sur quel datanode. Le namenode maintient deux tables en mémoire, l'une qui mappe les blocs aux datanodes (un bloc mappe à 3 datanodes pour une valeur de réplication de 3) et une mappe de numéro de bloc à datanode. Chaque fois qu'un nœud de données signale une corruption de disque d'un

bloc particulier, la première table est mise à jour et chaque fois qu'un nœud de données est détecté comme étant mort (à cause d'une panne de nœud/réseau), les deux tables sont mises à jour.

Secondary Namenode Le noeud secondaire se connecte régulièrement au noeud primaire et récupère des métadonnées du système de fichiers dans le stockage local ou distant.

Datanode Le Datanode est l'endroit où se trouvent les données.

Associés à ces différents types de noeuds, il existe des gestionnaires

Node Manager Il s'agit d'un démon yarn qui fonctionne sur des nœuds individuels et reçoit des informations sur les conteneurs de ressources de leurs Datanodes individuels via des démons. Les différentes ressources telles que la mémoire, le temps processeur, la bande passante du réseau, etc. sont regroupées dans une unité appelée conteneur de ressources. Le Node Manager assure à son tour la tolérance aux pannes sur les Datanodes pour tous les travaux MapReduce.

Resource Manager Il s'agit d'un démon yarn qui gère l'allocation des ressources aux différents jobs et qui comprend un planificateur qui s'occupe de la programmation des jobs.

```
[cloudera_manager]
master01.nikita.local

[cluster_master_nodes]
master01.nikita.local host_template=Master1

[cluster_worker_nodes]
worker01.nikita.local

[cluster_worker_nodes:vars]
host_template=Workers

[cluster:children]
cluster_master_nodes
cluster_worker_nodes

[db_server]
master01.nikita.local

[deployment:children]
cluster
db_server

[deployment:vars]
# Ansible will defer to the running SSH Agent for relevant keys
# Set the following to hardcode the SSH private key for the instances
# ansible_ssh_private_key_file=~/.ssh/mykey.pem
ansible_user=vagrant
```

FIGURE 4.8 – Exemple de fichier `inventory.ini`.

Chapitre 5

Déploiement d'un cluster Hadoop

Cette section décrit la méthodologie permettant l'installation et la configuration de l'écosystème Hadoop dans un cluster multinode. J'ai mis en place une architecture composée de deux nœuds, un master et un worker. Pour ce projet, j'ai utilisé Ambari 2.7.3 et HDP-3.1.4. Les fichiers sources sont disponibles sur le cloud d'Adaltas à l'adresse suivante : https://repos.adaltas.cloud/.

5.1 Déploiement de l'architecture

Ce projet a été lancé avec un cluster multinode. Pour cela, il est nécessaire de déployer deux machines virtuelles Centos 7 et de les configurer de telle sorte à ce qu'elles puissent accueillir les différents services du cluster. Le déploiement de ces vms est effectué en quelques commandes grâce à Vagrant. Le `Vagrantfile` utilisé est décrit dans la figure 4.6. Le cluster est constitué des noeuds `master01` et `worker01` dont les caractéristiques sont décrites respectivement dans les figures 5.1 et 5.2.

Intitulé	Valeur
FQDN	master01.nikita.local
Adresse IP	10.10.10.11
Mémoire	8192 MB

TABLE 5.1 – Caractéristiques du noeud `master01`.

Intitulé	Valeur
FQDN	worker01.nikita.local
Adresse IP	10.10.10.16
Mémoire	2048 MB

TABLE 5.2 – Caractéristiques du noeud `worker01`.

Une fois ces machines définies dans le `Vagrantfile`, il suffit de taper la commande suivante pour démarrer les noeuds :

```
$ vagrant up
```

Nous avons maintenant un cluster avec deux machines virtuelles Centos 7 vierges qui seront utilisées pour installer les différents services Hadoop.

5.2 Installation manuelle

5.2.1 Prérequis

Avant de pouvoir installer les composants, il est nécessaire d'effectuerquelques prérequis afin d'assurer le bon fonctionnement du cluster.

Configuration de SSH sans mot de passe

Pour qu'Ambari Server installe automatiquement les Agents Ambari sur tous les hôtes du cluster, il faut configurer des connexions SSH sans mot de passe entre l'hôte Ambari Server et tous les autres hôtes du cluster. L'hôte Ambari Server utilise l'authentification par clé publique SSH pour accéder à distance et installer l'agent Ambari. Autrement, il est possible d'installer manuellement un Agent Ambari sur chaque hôte du cluster. Dans ce cas, il n'est pas nécessaire de générer et de distribuer des clés SSH. La première méthode est recommandée car elle permet de définir les connexions entre les noeuds avant la mise en place d'Ambari.

Création des comptes utilisateurs de service

Chaque service nécessite un compte utilisateur de service. L'assistant d'installation d'Ambari crée de nouveaux comptes d'utilisateur et utilise ces derniers lors de la configuration des services Hadoop. La création de comptes d'utilisateur de service s'applique aux comptes d'utilisateur de service sur le système d'exploitation local et aux comptes LDAP/AD. Par exemple, pour le service Hive, il faut créer un compte utilisateur `hive` sur tous les noeuds du cluster. Cette opération permet aux services Hive Metastore, HiveServer2 de fonctionner.

Configuration du DNS

Tous les hôtes du cluster doivent être configurés convenablement au niveau de la résolution DNS direct et inverse. Pour cela, il suffit de modifier le fichier `/etc/hosts` sur chaque hôte afin d'y ajouter l'adresse IP et le nom de domaine complet (FQDN) de chaque machine. Hadoop s'appuie fortement sur le DNS et effectue de nombreuses résolutions DNS en fonctionnement normal. Dans notre exemple, chaque VM contient les entrées suivantes dans le fichier `/etc/hosts` :

```
127.0.0.1   localhost
::1         localhost
10.10.10.11 master01.nikita.local
10.10.10.16 worker01.nikita.local
```

Configuration des connecteurs de base de données

Les services tels que Druid, Hive, Ranger et Oozie nécessitent une base de données opérationnelle. Pour qu'Ambari puisse se connecter à une base de données, il faut télécharger les pilotes de base de données et les connecteurs nécessaires avant d'installer le composant. Il est possible d'utiliser les bases de données MySQL, Oracle, PostgreSQL ou Amazon RDS. Dans notre cas, nous utilisons MySQL que nous installons sur un noeud qui sera utilisé comme serveur BDD.

5.2.2 Installation depuis un dossier dans le cloud

Les fichiers sources permettant l'installation de HDP et d'Ambari sont hébergés sur un dossier dans le cloud d'Adaltas accessible à l'adresse suivante : https://repos.adaltas.cloud/. Cela permet de bénéficier d'une plus grande gouvernance et de meilleures performances d'installation. Ensuite, il suffit de modifier le fichier `ambari.repo` et de remplacer l'URL de base Ambari `baseurl` obtenue lors de la configuration.

A présent, il s'agit de télécharger Ambari Server sur le noeud principal, c'est-à-dire `master01`. Avant de démarrer le serveur Ambari, il faut le configurer. L'installation configure la connexion à la base de données, installe le JDK et permet de personnaliser le compte utilisateur sous lequel le démon Ambari Server s'exécutera. La commande `ambari-server setup` gère le processus d'installation.

5.2.3 Configuration et déploiement d'un cluster

Pour installer, configurer et déployer un cluster HDP, il faut encore effectuer plusieurs opérations sur l'invité de commande du noeud principal ainsi que sur l'interface web d'Ambari.

Démarrage du serveur Ambari

Exécuter la commande suivante sur l'hôte du serveur Ambari :

```
ambari-server start
```

Connexion à l'interface Ambari

Pour se connecter à Ambari Web, il suffit d'ouvrir le navigateur web à l'adresse `http://master01.nikita.local/8080` et de se connecter au serveur en utilisant le nom d'utilisateur/mot de passe par défaut : admin/admin. Pour un nouveau cluster,

l'assistant d'installation de cluster affiche une page de bienvenue comme le montre la figure 5.1.

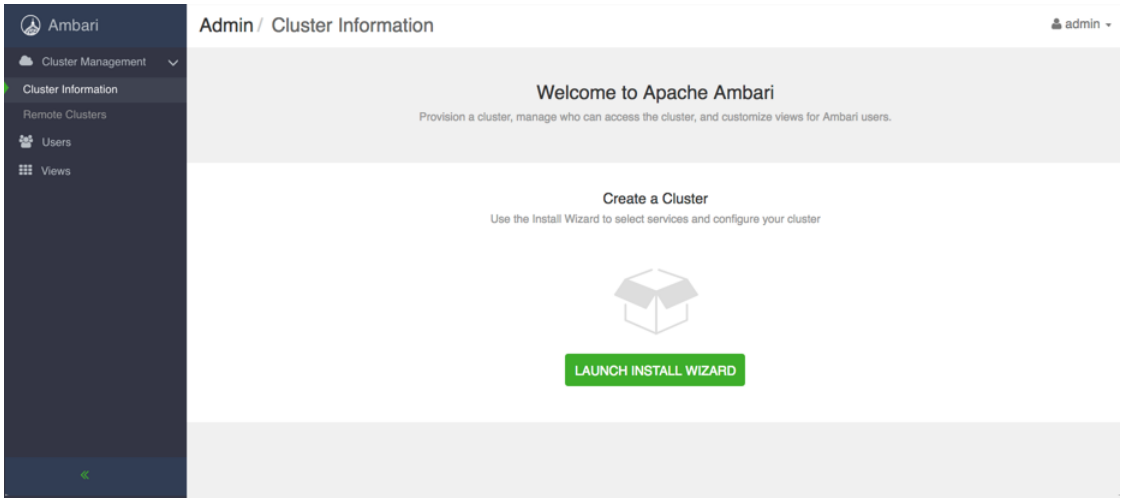

FIGURE 5.1 – Ecran de bienvenue de l'assistant d'installation Ambari.

A présent, il s'agit de lancer l'assistant d'installation de cluster d'Ambari durant lequel il faudra effectuer les étapes suivantes :

1. Attribuer un nom au cluster
2. Sélectionner la version de HDP
3. Fournir le FQDN de chacun des noeuds
4. Confirmer qu'Ambari a localisé tous les hôtes pour s'assurer qu'ils ont les les paquets et les processus requis
5. Sélectionner les services souhaités au sein du cluster
6. Assigner les noeuds "master"
7. Assigner les noeuds "esclaves"
8. Configurer les services
9. Confirmer la configuration

Fort de toutes les étapes que nous venons d'effectuer, nous avons déployé un cluster HDP localement. Cela étant, cette méthodologie n'est pas viable si l'on venait à déployer une centaine de clusters sur plusieurs infrastructures indépendantes. Ainsi, il nous faut trouver une solution qui permettrait d'automatiser toutes les étapes redondantes telles que l'installation de la base de données ou encore la configuration du DNS.

5.3 Automatisation de l'installation et du déploiement

La méthode d'installation décrite dans la section précédente n'est pas viable car elle est chronophage et répétitive. Afin de valoriser le temps passé à mettre en place le cluster, il s'agira d'automatiser l'installation et le déploiement du cluster.

5.3.1 Automatisation avec Ansible

Ansible permet la configuration, le déploiement, le provisionnement, et l'orchestration de clusters multi-nœuds. Pour cela, il existe un concept nommé **playbook**. Les playbooks enregistrent et exécutent les fonctions de configuration, de déploiement et d'orchestration d'Ansible. Ils peuvent décrire une configuration à appliquer pour les systèmes, ou un ensemble d'étapes d'un processus informatique général. Les playbooks sont conçus pour être lisibles par l'homme et sont développés dans un langage texte de base. Pour l'automatisation du déploiement, j'ai créé un playbook Ansible, inspiré de celui mis à disposition par Hortonworks [1], capable de construire un cluster HDP en utilisant Ambari Blueprints. Les Blueprints d'Ambari sont une définition déclarative d'un cluster. Avec un Blueprint, il est possible de définir le Stack Hadoop, la disposition des composants et les configurations pour matérialiser une instance de cluster Hadoop (via une API) sans avoir à utiliser l'assistant d'installation de cluster Ambari. Ainsi, le playbook se charge des étapes suivantes :

1. Préparation des noeuds (prérequis)
2. Installation d'Ambari
3. Configuration d'Ambari
4. Déploiement du Blueprint
5. Post-installation (tests fonctionnels)

Avec cette méthodologie, nous arrivons au même résultat qu'avec l'installation manuelle. Il est possible de modifier quelques fichiers de configuration et ainsi de déployer n'importe quel cluster sur n'importe quelle infrastructure.

5.4 Services Hadoop

Dans cette section, nous décrivons les services Hadoop installés sur le cluster, indépendamment de la méthodologie utilisée.

5.4.1 HDFS

Hadoop Distributed File System (HDFS) est un système de fichiers distribués. HDFS est hautement tolérant aux pannes. Il fournit un accès haut débit aux données et convient à des applications qui ont de grands ensembles de données. HDFS assouplit quelques exigences POSIX pour permettre un accès en continu aux données du système de fichiers.

1. https://github.com/hortonworks/ansible-hortonworks

5.4.2 YARN

YARN est un système d'exploitation distribué à grande échelle pour les applications big data. Cette technologie est conçue pour la gestion de clusters. YARN est capable de découpler les capacités de gestion et d'ordonnancement des ressources de MapReduce du composant de traitement des données.

5.4.3 HBase

HBase est un système de gestion de base de données non relationnelle, open-source, distribuée, versionnée, qui fonctionne au-dessus du système de fichiers distribués Hadoop (HDFS). HBase offre un moyen tolérant aux pannes de stocker des ensembles de données, ce qui est courant dans de nombreux cas d'utilisation du Big Data. Il est bien adapté au traitement des données en temps réel ou à l'accès aléatoire en lecture/écriture à de grands volumes de données.

5.4.4 Hive

Hive est utilisé comme substitut de SQL pour le système de fichiers Hadoop, ce qui permet aux utilisateurs connaissant SQL d'interroger facilement des données à partir de ce système au lieu de devoir apprendre Map-Reduce. Le logiciel facilite la lecture, l'écriture et la gestion de grands ensembles de données au sein d'un stockage distribué.

5.4.5 Spark

Apache Spark est un moteur d'analyse unifié pour le traitement des données à grande échelle. Il fournit des API de haut niveau en Java, Scala, Python et R, ainsi qu'un moteur optimisé qui prend en charge les graphes d'exécution généraux. Il prend également en charge un riche ensemble d'outils de plus haut niveau, notamment Spark SQL pour le traitement des données SQL et structurées, MLlib pour l'apprentissage automatique, GraphX pour le traitement des graphes et Structured Streaming pour le calcul incrémental et le traitement en continu.

5.4.6 Oozie

Apache Oozie est un logiciel utilisé pour planifier les travaux Apache Hadoop. Oozie combine plusieurs travaux de manière séquentielle en une seule unité logique de travail. Il est intégré au Stack Hadoop, avec YARN comme centre architectural, et prend en charge les travaux Hadoop pour Apache MapReduce, Apache Pig, Apache Hive et Apache Sqoop. Oozie peut également planifier des tâches spécifiques à un système, comme des programmes Java ou des scripts shell.

5.4.7 Ranger

Apache Ranger est un logiciel permettant de mettre en place la sécurité d'un cluster Hadoop. Il fournit une plateforme centralisée pour définir, administrer et gérer les politiques de sécurité de manière cohérente à travers les services Hadoop.

Chapitre 6

Conclusion et perspectives

Ce chapitre clôture ce rapport de stage qui traite de l'automatisation de l'installation, de la configuration et du déploiement d'un cluster HDP. Au fil de ce rapport, nous avons présenté une méthode manuelle de déploiement de cluster, c'est-à-dire en accédant aux différents noeuds du cluster pour installer les prérequis puis en suivante les étapes de configuration grâce à l'assistant de configuration Ambari. Puis nous avons amélioré ce déploiement en l'automatisant grâce à Ansible. Enfin, l'ensemble des publications réalisées à partir des travaux décrits dans ce rapport sont disponibles sur le site d'Adaltas.

6.1 Résultats obtenus

Lors de l'installation, la configuration et le déploiement manuel du cluster, une étude de la durée et de l'extensibilité a été effectuée. Cette méthodologie n'a pas la capacité de s'adapter à un changement d'ordre de grandeur de la demande. En d'autres termes, il semble chronophage et redondant d'appliquer cette méthode d'installation sur un grand nombre de clusters avec des configurations différentes et des environnements de déploiement variables. Ainsi, l'automatisation du processus présente des avantages majeurs, notamment d'un point de vue temporel.

6.2 Perspectives

Le projet visant à développer une plateforme ou un logiciel permettant le déploiement facile et rapide d'un cluster Hadoop est loin d'être abouti. La solution proposée par Cloudera constitue une fondation solide mais ne présente pas les avantages d'un logiciel open-source. L'objectif, à long terme, consiste en le développement d'un logiciel open-source permettant l'analyse de données grâce à des outils analytiques en libre-service au sein d'environnements hybrides et multicloud tout en proposant une expérience des données partagée qui garantit sécurité, gouvernance et métadonnées.

6.3 Mot de la fin

Ce stage technique effectué au sein de l'entreprise Adaltas constitue une expérience des plus enrichissantes étant donné la complexité technique des missions auxquelles j'ai pu prétendre participer. Outre l'aventure humaine enrichissante que j'eus la chance et le privilège de vivre, ce stage m'apprit le sens de la rigueur, du professionnalisme, ainsi que l'importance du temps et de son agencement. Grâce à cette expérience, j'ai acquis des nouvelles compétences qui me seront utiles pour mes futurs projets. De plus, les différentes missions effectuées m'ont permis d'accroître ma volonté de savoir et de connaissance, notamment dans le domaine du Big Data.

La réalisation des travaux décrits dans ce rapport m'a permis d'acquérir de nouvelles compétences en matière de développement, de gestion et de déploiement d'un cluster Big Data. J'ai appris à construire des systèmes faciles à utiliser, automatisés, sécurisés et fiables.

En terme de compétences techniques nouvellement acquises, nous pouvons énoncer :
- Configuration et utilisation de systèmes et outils d'infrastructure
- Déploiement de systèmes évolutifs avec une infrastructure de production
- Automatisation du déploiement et provisionnement de clusters Big Data

Glossaire

Ambari logiciel pour le provisionnement, la gestion et la surveillance des clusters Apache Hadoop. 6, 23, 24, 28–31

Ansible plate-forme logicielle open-source pour la configuration, automatisation et la gestion de machines. 6, 23, 31

API Ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. 31

Arch Linux Système d'exploitation Linux simple et sans outils de configuration destiné aux utilisateurs avancés. 17, 18, 20, 36

Centos distribution GNU/Linux destinée aux serveurs. 23, 27, 28

Debian Système d'exploitation libre basé sur Linux. 36

DevOps Pratique technique visant à l'unification du développement logiciel (dev) et de l'administration des infrastructures informatiques (ops). 10, 11

dpkg Gestionnaire de paquets de Debian. 18

Git Logiciel libre de gestion de versions décentralisé. 36

GitHub Service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git. 11

GNU Système d'exploitation constitué de logiciel libre. 18

Hadoop Framework libre et open source écrit en Java destiné à faciliter la création d'applications distribuées. 6, 10, 16, 23, 24, 28

HDP Hortonworks Data Platform (HDP) est un logiciel open source pour le stockage et le traitement distribués de grands ensembles de données multi-sources. 16, 29–31

NoSQL Famille de systèmes de gestion de base de données. 10

PAAS Platform as a service (Plate-forme en tant que service) - Types de cloud computing où le fournisseur cloud maintient la plate-forme d'exécution des applications. 11

pacman Gestionnaire de paquets d'Arch Linux. 18

SRE Discipline qui intègre des aspects de l'ingénierie logicielle et les applique aux problèmes d'infrastructure et d'exploitation. 10

Vagrant Outil permettant de créer et de gérer des environnements de machines virtuelles. 6, 21–23, 27

VirtualBox Logiciel de virtualisation Open Source et multiplateforme. 6, 20, 21

Alexander Hoffmann

Looking for a 6 months software engineering internship during Summer 2021

Email : alhffn@gmail.com
Mobile : +33 6 76 02 20 22

Education

- **ECE Paris** — France
  Master of Science in Computer Science — *Sept 2016 – Sept 2021 (Expected)*
- **University of California San Diego** — CA, USA
  Extension program in Computer Science and Engineering — *Sept 2018 – Dec 2018*

Experience

- **Teaching Assistant** — Paris, France
  ECE Paris — *Sept 2019 - Present*
  - Assisting the instructor by providing tutoring to individual students or small groups of students in C/C++.
- **Junior Software Developer** — Duesseldorf, Germany
  Sadnacaya GmbH & Co. KG — *Aug 2018 - Present*
  - Creation of a Python script to upload a file onto an OCR web server and retrieve it as a PDF.
  - Configuration of a Manjaro VM onto a QNAP NAS to automatically back up mission-critical data on Write-Once-Read-Many archive storage.
  - Implementation of a OPNSense Firewall onto a virtual machine.
  - Development of an image analysis tool aiming to scan invoices and export its content to a database.
- **Junior Software Qualification Consultant** — Duesseldorf, Germany
  Ayacandas GmbH — *May 2017 - Present*
  - Development of an exchange platform for hospitals and other medical institutions.
  - Quality control of the system to ensure that the software solution developed meets the requirements.
  - Analysis of the anomalies, documentation of the tests carried out and progress report on the test campaign.
- **Software Developer Intern** — Paris, France
  Préfecture de police de Paris — *Jul 2019 - Aug 2019*
  - Development of a Java application to control backups and automatically notify system administrators.
  - Implementing new features to local web applications to meet user demands and improve overall performance.

Open Source Projects

- **Coin counter (Java)** `[github]` : Implementation of OpenCV Coin Detection in an image.
- **Node metrics (Typescript)** `[github]` : Simple web API to work on metrics.
- **School Management Tool (Java Spring MVC)** `[github]` : A web-based, extensible platform for managing schools. Used Hibernate to implement a relational database. Created a dynamic search engine using Elasticsearch.
- **Ecezon (HTML5, CSS, PHP Laravel)** `[github]` : A free, open-source e-commerce platform written in PHP based on Laravel. Implemented online payments with Stripe. Created dynamic search engine using Algolia.
- **Mini Router (C/C++)** `[github]` : Simple router given a static network topology and routing table.

Technical Skills

- **Languages (alphabetically)** : C, C++, CSS, HTML, Java, LaTeX, Markdown, PHP, Python, SQL, UML.
- **OS** : Linux, Microsoft Windows.
- **Software & Technology** : AWS, Computer Vision, Elasticsearch, Github, Gitlab, Hibernate, Maven, Networking, NodeJS, OPNsense, REST API, Travis CI, vim, Virtualization, Wireshark.

Volunteer work

- **Treasurer** — Paris, France
  Genius Campus Eiffel — *Apr 2019 - Present*
  - Holding a volunteer position as a treasurer at the entrepreneurship club at ECE Paris.
  - Overseeing and presenting budgets, accounts and financial statements to the management committee.

Skills & Interests

- **Languages** : French (mother tongue), German (mother tongue), English (fluent), Spanish (notions).
- **Actuarial science** : applying mathematical and statistical methods to assess risk in finance.
- **Hobbies** : cycling, athletics, chess.

Alexander Hoffmann

Looking for a software engineering position or freelance contract

Email : alexander@hoffmann.ai
Mobile : +33 6 76 02 20 22

Education

- **ECE Paris** — France
  Master of Science in Computer Science — *2016 – 2021*
- **University of California San Diego** — CA, USA
  Extension program in Computer Science and Engineering — *2018*

Experience

- **CEO** — Paris, France
  HOFFMANN.AI — *2020 – Present*
  - **Spotmydive** Write client-side code (GatsbyJS) for web-based high-volume production application (120,000 unique visitors/mo). Improve conversion rate and overall SEO by 20%.
  - **Machine Break** Produce front-end code to create a polished and highly functional user interface with a focus on usability and an accurate representation of the approved design mocks.
  - **Orange (contractor)** Design, create and maintain a scalable web-based dashboard application using TypeScript and MongoDB for back-end and ReactJS for front-end. Reduce customer acquisition cost by 5%.
  - **GEFCO** Design, develop, test, deploy, maintain and improve front-end (ReactJS) and back-end (C#) for highly available web application. Improve work performance and productivity by 10%.
- **InfraOps Engineer** — Paris, France
  Adaltas — *2021*
  - Build infrastructure frameworks, systems and tools to deploy new global Adaltas products that scale over millions of users.
  - Ensure that on-premise and cloud-based systems and infrastructure is designed appropriately for security to ensure protection against current and future threats.
  - Carry out deployment, maintenance, monitoring, and management tasks within a cloud structure.
- **Digital Competence Instructor** — Bonn, Germany
  Deutsche Telekom — *2021*
  - Plan for and implement digital devices and resources in the teaching process.
  - Use digital technologies and services to enhance the interaction with learners, individually and collectively, within and outside the learning session.
  - Enable learners to use digital technologies as part of collaborative assignments, as a means of enhancing communication, collaboration and collaborative knowledge creation.
- **Junior Software Developer** — Düsseldorf, Germany
  Sadnacaya GmbH & Co. KG — *2018 – 2021*
  - Contribute to a wide variety of projects using natural language processing, artificial intelligence, data compression, machine learning and search technologies.
  - Configuration of a Manjaro VM onto a QNAP NAS to automatically back up mission-critical data on Write-Once-Read-Many archive storage.
  - Provide advanced corporate network infrastructure and server support.
- **Assistant professor of Computer Science** — Paris, France
  ECE Paris — *2020*
  - Teach and supervise two classes of 35 undergraduate students in C and C++.
- **Software Engineering Intern** — Paris, France
  Adaltas — *2020*
  - Research, conceive and develop software applications to extend and improve on Adaltas' product offering.
  - Contribute to DevOps processes and the requirements of producing and operating a machine learning model.
  - Collaborate on scalability issues involving access to data and information.
- **Software Developer Intern** — Paris, France
  Préfecture de police de Paris — *2019*
  - Development of a Java application to control backups and automatically notify system administrators.
  - Implement new features to local web applications to meet user demands and improve overall performance.
- **Software Tester** — Düsseldorf, Germany
  Ayacandas GmbH — *2017 – 2018*
  - Development of an exchange platform for hospitals and other medical institutions.
  - Quality control of the system to ensure that the software solution developed meets the requirements.
  - Analysis of the anomalies, documentation of the tests carried out and progress report on the test campaign.

Volunteering

- **Treasurer** — Paris, France
  Genius Campus Eiffel - entrepreneurship club — *2019 – 2021*
  - Overseeing and presenting budgets, accounts and financial statements to the management committee.

Open Source Personal Projects

- **Wava (Java)** `[github]` : Collection of general-purpose utility classes with wide applicability.
  - Created static methods that operate on or return hash-maps.
  - Implemented unit tests using JUnit.
  - Continuous integration using Travis CI.
- **Docsight (Python)** `[github]` : A tool for analyzing scanned files and export their content.
  - Used OpenCV to remove noise and clean up image.
  - Implemented Tesseract OCR to recognize text from image.
  - Exported content as JSON, could easily be exported to a NoSQL database.
- **School Management Tool (Java Spring)** `[github]` : A web-based, extensible platform for managing schools.
  - Used Hibernate to implement a relational database.
  - Created a dynamic search engine using Elasticsearch.
  - Deployed web application to AWS.
- **Ecezon (PHP Laravel)** `[github]` : A free, open-source e-commerce platform written in PHP based on Laravel.
  - Implemented online payments with Stripe.
  - Created dynamic search engine using Algolia.
  - Deployed web application to Heroku.
- **Mini Router (C/C++)** `[github]` : Simple router given a static network topology and routing table.
  - Created a simple router given a static network topology and routing table.
  - Implemented ping and traceroute to and through the router.
  - Handled various Ethernet packets including ARP, TCP/UDP, HTTP and ICMP.

Open Source Contributions

- **Node.js Nikita (CoffeeScript)** `[github]` : Automation and deployment solution with Node.js.
  - Migrated and refactored low level functions of the core engine.
  - Implemented unit tests to ensure functioning of actions both locally and remotely (via ssh).
  - Created Docker containers to ensure tests are executed in a proper Linux-based environment.
- **gatsby-theme-catalyst (ReactJS)** `[github]` : Themes and starters to accelerate development with GatsbyJS.
  - Improved post footer styles and format in blog themes.

Technical Skills

- **Languages (alphabetically)** : C, C++, HTML/CSS, Java, JavaScript, LaTeX, Markdown, ReactJS.
- **Big Data & Analytics** : Ambari, Elasticsearch, Hadoop Stack.
- **Databases** : MongoDB, MySQL, PostgreSQL.
- **DevOps & InfraOps** : Ansible, Docker, Git, Kubernetes, Maven, Unix, Vagrant.
- **Web technologies** : AWS, DevOps, Git, Java Spring, NodeJS, Shopify, Wordpress.
- **Other Software & Technologies** : Computer Vision (OpenCV), Networking, vim, Virtualization.

Skills & Interests

- **Languages** : French (mother tongue), German (mother tongue), English (fluent), Spanish (notions).
- **Actuarial science** : applying mathematical and statistical methods to assess risk in finance.
- **Chess** : Ranked approx. 1600 elo.
- **Athletics** : 2018 vice-champ in 100-metre dash, third place in triple jump at the regional level (Île-de-France).
- **Sailing** : seaman with a track record of successfully operating and supervising boating excursions.

ECE PARIS · LYON ÉCOLE D'INGÉNIEURS

EVALUATION DE L'ENTREPRISE

à compléter par le Maître de stage

à insérer dans le rapport de stage **Obligatoirement**

Fiche d'évaluation du stage
3ème année Cycle Ingénieur
Promotion 2021

Etudiant :.....Alexander Hoffman **Majeure** : SI Big Data
Entreprise :.....Adaltas.....
Maître de stage :...David Worms...
Téléphone:.....0676887213.....**@** : david@adaltas.com

Maîtrise des domaines scientifiques et techniques : • Capacité d'analyse/compréhension des problèmes • Mise en œuvre de ses connaissances • Aptitude à acquérir de nouvelles connaissances (formation ou autoformation) **Maîtrise des méthodes et des outils de l'ingénieur :** • Méthodologie/organisation du travail, gestion de projet : développement d'un outil ou d'une méthodologie • Synthèse et communication des résultats, maîtrise des outils de communication **Conduite de l'action et prise de décision :** • Réalisation des objectifs, qualité du travail réalisé • Autonomie/initiative/créativité/ouverture • Respects des procédures (qualité, sécurité, santé…)	10/10 9 /10 10/10
Intégration dans une organisation et capacité d'animation : • Capacité à s'intégrer dans une équipe : exprimer ses attentes, donner son point de vue, sens de l'écoute, accepter la critique et se remettre en cause • Communication sur ses activités et aptitude à rendre compte (réunion, relation client…) • Prise en compte des enjeux métiers et économiques **Respect des valeurs sociétales, sociales et environnementales :** • Appropriation des valeurs, des codes, et de la culture de l'équipe et de l'organisation • Comportement éthique	5 /5 4 /5
TOTAL	38**/40**
Appréciation globale sur le stage et observations Alexander possède d'excellentes fondations techniques. Quoique déjà mature pour son age, il va continué a gagner en maturité ce qui le conduira rapidement à être en capacité de mener des missions de type expert ou architecte nécessitant la capacité d'écoute et de dialogue entre différents acteurs.	**TOTAL** 19**/20**

A Boulogne **le** 23 juin **2021**
Signature du Maître de stage (obligatoire)

Cachet de l'entreprise (Obligatoire)

ADALTAS
SAS au capital de 8000 €
6 rue Jules Simon
92100 Boulogne-Billancourt - France
RCS Nanterre B 452 561 913
Siret: 452 561 913 00032
TVA Intracommunautaire: FR31452561913

ECE PARIS • LYON
ÉCOLE D'INGÉNIEURS

Fiche d'évaluation
Rapport de stage de Fin d'Etudes
Cycle Ingénieur • 3ème année
2020/2021

Etudiant : ……..

Entreprise : …….. **Majeure : ……..**

Correcteur : ……

Date de correction :……

	Note
Présentation du contexte du stage - Présentation et histoire de l'entreprise, description des activités, organigramme - Politique de Responsabilité Sociétale de l'Entreprise (RSE) – - Définition et enjeux de la mission - Détails de la spécification des besoins, du CDC et du planning prévisionnel (Gestion de projet) – Document Obligatoire	**/5**
Présentation et valorisation du travail réalisé - Méthodologie et outils - Connaissances préalables utilisées, et compétences acquises. - Qualité pédagogique du rapport - Qualité rédactionnelle	**/20**
Bilan et perspectives - Analyse critique de la réalisation des objectifs et du travail personnel réalisé (prise d'initiative, créativité, recul, valeur ajoutée) - Difficultés rencontrées : réflexion sur les erreurs commises, les pertes de temps à postériori - Conséquences de ce stage sur votre avenir professionnel (secteur d'activité, type d'entreprise, fonctions, relation client, collaborations…) - Connaissance du métier de l'ingénieur : réflexion sur le métier d'ingénieur. La mission correspond-elle à ce que vous attendez du métier d'ingénieur ? - Perspectives	**/15**
Total	**/40**
NOTE FINALE	**/20**

Observations :

A ……, le ……2021
Signature du Correcteur (obligatoire)

ECE PARIS · LYON
ÉCOLE D'INGÉNIEURS

Fiche d'évaluation
Soutenance de Stage de fin d'études
Cycle Ingénieur 3ème année
2020/2021

Etudiant : Majeure :

Entreprise :

Composition du Jury
Représentant l'ECE : ..... Président du Jury : .....
Représentant l'entreprise d'accueil :
Maître de Stage et invité(s) :.....

Soutenance effectuée le : ..... /...../2021

	Note
Présentation du contexte du stage - Présentation de l'entreprise et de **sa politique RSE** et description des activités - Définition et enjeux de la mission	**/5**
Présentation et valorisation du travail réalisé - Méthodologie et outils - Identification des connaissances utilisées et compétences acquises - Analyse critique de la réalisation des objectifs et du travail personnel réalisé	**/15**
Perspectives et bilan personnel - Perspectives de l'expérience en entreprise - Conséquences de ce stage sur votre avenir professionnel (secteur d'activité, type d'entreprise, fonctions, relation client, collaborations…) - Connaissance du métier de l'ingénieur : réflexion sur le métier d'ingénieur. La mission correspond-elle à ce que vous attendez du métier d'ingénieur ?	**/10**
Qualité générale de la soutenance - Gestion du temps - Qualité de l'expression orale (prestance, fluidité verbale, clarté, …) - Efficacité des supports (visuels, graphiques, messages clés, …)	**/5**
Sous total	**/35**
Qualité des réponses fournies au jury	**/5**
TOTAL	**/40**
Soit	**/20**

Observations :

Signatures des membres du Jury (Signature obligatoire du Président) :