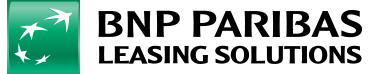


Rapport d’activité du semestre 10

**MENASRIA Salah**

Apprenti au sein de BNP Paribas Leasing Solution



**Tuteur pédagogique :**

M. GRILL Gary

**Maître d’apprentissage :**

M. CONTI Bruno

**Année scolaire : 2020/2021**

Table des matières

[I. Remerciements 3](#_Toc74660126)

[II. Contexte 3](#_Toc74660127)

[III. Présentation de l’entreprise 4](#_Toc74660128)

[1. Le groupe BNP PARIBAS 4](#_Toc74660129)

[a) Histoire 4](#_Toc74660130)

[b) Présentation des pôles et des métiers : 4](#_Toc74660131)

[c) Politique et engagement 5](#_Toc74660132)

[2. Présentation de BP leasing solutions 5](#_Toc74660133)

[a) Fonctions de BP leasing solutions CORPORATE 7](#_Toc74660134)

[IV. Présentation de l’équipe et des missions réalisées 7](#_Toc74660135)

[1. L’équipe Cloud 8](#_Toc74660136)

[a) Missions de l’équipe 8](#_Toc74660137)

[1. Le cloud chez BNP PARIBAS 8](#_Toc74660138)

[2. L’Importance du Cloud pour l’entreprise 9](#_Toc74660139)

[2. la mission réalisée 11](#_Toc74660140)

[a) Développement et déploiement des applications de démonstrations sur le Cloud 11](#_Toc74660141)

[V. Bilan des acquis 21](#_Toc74660142)

[1. Difficultés rencontrées et solutions apportées 21](#_Toc74660143)

[2. Résultats obtenus/compétences acquises 21](#_Toc74660144)

[3. Perspective d’avenir 21](#_Toc74660145)

# Remerciements

Tout d’abord, je tiens à remercie mon maitre d’apprentissage **M. CONTI Bruno** ainsi que chef de l’équipe Cloud **M. BERTAUX Cyril** pour le temps passé ensemble, le partage de leurs expertises au quotidien, leur aide précieuse et le suivi tout au long de ce semestre.

Ensuite, je remercie mon tuteur pédagogique **M. GRILL Gary** pour tous les bons conseils qu’il m’a transmis et pour sa disponibilité à mon égard tout au long de mon alternance. Je tiens aussi à remercier toute l'équipe pédagogique de l’école sup Galilée pour l’encadrement et l’accompagnement. Enfin, j'adresse mes remerciements aux membres de l’équipe Cloud pour leur partage de connaissances et aussi les missions que l’on me confie.

# Contexte

À travers ce rapport, je vais vous présenter les nouvelles expériences acquises durant le dernier semestre de mon d’alternance chez BNP PARIBAS LEASING SOLUTION située à 12 rue du Port, 92000 Nanterre. Je vais commencer par la présentation de l’entreprise, ensuite je vous détaillerai les missions qui m’ont été confiées, je ferai aussi une synthèse de de mes activités durant les deux années d’alternance et je finirai ce rapport par donner mon et ressenti face à cette expérience ainsi que mes perspectives d’évolution pour les deux mois restants.

# Présentation de l’entreprise

## Le groupe BNP PARIBAS

### Histoire

BNP Paribas est issue d’une fusion en 2000 entre deux banques, la Banque Nationale de Paris (BNP) et Paribas. L’établissement est aujourd’hui l'un des grands opérateurs mondiaux du secteur bancaire.

* La Banque Nationale de Paris (BNP) :

Dans un contexte de réformes et de concentration bancaire initié par le ministre des Finances Michel Debré, une fusion a eu lieu en 1966 entre deux des plus grandes banques de France : le Comptoir National d’Escompte de Paris (CNEP), et la Banque Nationale pour le Commerce et l’Industrie (BNCI). Cet évènement a donné naissance à la Banque Nationale de Paris (BNP), finalement privatisée en 1993.

* Paribas :

À Paris, en 1872, des banquiers français et allemands créent la Banque de Paris et des Pays Bas, qui devient Paribas en 1982. Dès sa création, la BPPB ne disposait pas d’un réseau bancaire proprement dit. Pour étendre son influence et développer ses activités dans le secteur de financement des entreprises et le placement des emprunts internationaux, elle établissait des relations étroites avec les grands représentants de la finance internationale (Baring, Deutsche Bank…). Pour échapper à sa nationalisation par la loi du 2 décembre 1945, la BPPB choisit le statut de banque d’affaire et elle oriente ses activités vers la restructuration de l’industrie française en s’appuyant sur l’émergence d’un espace économique européen. En 1982 la BPPB est nationalisée, et c’est là qu'apparaît officiellement le nom Paribas. Elle est privatisée à nouveau en 1987. En 2000, BNP gagne une longue bataille boursière l’opposant à la Société générale pour le contrôle de Paribas, et c’est comme ça que le groupe BNP Paribas est née.

### Présentation des pôles et des métiers :

Une image contenant capture d’écran

Description générée automatiquement

Les activités de BNP Paris sont organisées en deux domaines d’activités RETAIL BANKING & SERVICES (RBS) ET CORPORATE & INSTITUTIONAL BANKING.

* **RETAIL BANKING & SERVICES :**

Retail Banking & Services regroupe les réseaux de banque de détail et les services financiers spécialisés du Groupe en France et à l’international. Il se compose de Domestic Markets et International Financial Services. Présent dans plus de 60 pays et employant plus de 148 000 collaborateurs, Retail Banking & Services a réalisé en 2018 74 % des revenus des pôles opérationnels de BNP Paribas.

* **CORPORATE AND INTITUTIONAL BANKING:**

Avec plus de 30 000 personnes dans 56 pays, BNP Paribas CIB propose à deux types de clientèle –entreprises et institutionnels– des solutions sur-mesure dans les domaines des marchés de capitaux, des services de titres, des financements, de la gestion de trésorerie et du conseil financier. En 2018, 25 % des revenus des pôles opérationnels de BNP Paribas ont été réalisés par BNP Paribas CIB.

### Politique et engagement

La politique du Groupe BNP Paribas repose sur 4 principes fondamentaux :

* Responsabilité économique
* Financer l’économie de manière éthique
* Responsabilité environnementale
* Agir contre le changement climatique
* Responsabilité civique
* Etre un acteur engagé de la société
* Responsabilité sociale
* Favoriser le développement et l’engagement de ses collaborateurs

Depuis sa fondation, BNP Paribas accompagne et soutient de nombreux projets concernant la culture, la solidarité et la recherche environnementale. Le Groupe reste fidèle à ses partenariats historiques qui le lient aux univers du cinéma et du tennis (Roland Garros, Festival de Cannes …).

## Présentation de BP leasing solutions

BNP Paribas Leasing Solution est issue de la fusion et le rapprochement de nombreuses entreprises toute au long du dernier siècle. Le plus important événement dans son histoire est la fusion entre UFB Locabail et BNP Lease (filiale de BNP Paribas), ce qui a donné naissance à BNP Paribas Lease Groupe, qui change son nom en 2010 et devient BNP Paribas Leasing Solution.

BNP Paribas Leasing Solutions propose aux entreprises et aux professionnels des solutions locatives et de financement pour les équipements à usage professionnel.

Pour proposer ces solutions, elle utilise principalement trois canaux de communication :

* Partenaires industriels (constructeurs, distributeurs et éditeurs d’équipement professionnels) avec des solutions de financement pour leurs clients finaux.
* Entreprises avec des offres de location avec services et de gestion de parc.
* Clients professionnels des réseaux bancaires de BNP Paribas (BDDF, BNL...) en soutenant leurs projets d’investissements.

BNP Paribas Leasing Solutions est organisé en Business-Units développant des solutions spécifiquement adaptées selon les actifs :

* Equipment & Logistics Solutions (ELS): pour Les équipements logistiques roulants (machines agricoles, engins de BTP et de manutention, véhicules utilitaires et industriels).
* Technology Solutions (TS) : pour Les équipements technologiques (informatique, bureautique, télécoms, médical et technologies spécialisées).
* Bank Leasing Services (BLS) : l'activité Leasing pour les clients de la banque BNP Paribas.

Au cœur de l’économie de l’usage, BNP Paribas Leasing Solutions apporte aux entreprises la flexibilité dont elles ont besoin pour rester compétitives et se développer de manière durable et responsable, et pour cela elle propose quatre types de financements:

* Location financière avec ou sans services :

Mise à disposition d’un équipement contre le paiement d’un loyer mensuel. À la fin du contrat, le client peut restituer le bien, poursuivre la location ou souscrire un nouveau contrat pour un autre équipement.

Le contrat peut inclure des services additionnels tels que l’entretien et la maintenance.

* Location longue durée avec gestion de parc :

Service de location et de gestion de flottes d’équipements technologiques et de véhicules industriels et commerciaux. Service proposé par notre filiale BNP Paribas Rental Solutions.

* Crédit-bail mobilier :

A la fin du contrat, le client a la possibilité de devenir propriétaire de l’équipement en levant l’option d’achat, c’est à dire en réglant le montant prévu au moment de la signature du contrat.

* Crédit-bail immobilier :

Financement de la construction ou l’achat de bâtiments à usage professionnel.

### Fonctions de BP leasing solutions CORPORATE

Une image contenant vert, assis, rue, table

Description générée automatiquementLes Fonctions de BNP PARIBAS sont bien définies et chacune cible une activité bien précise:

Chaque Fonction est dirigée par un Responsable qui rapporte, soit au Directeur Général, soit au Directeur Général délégué, soit au Directeur Général Adjoint transversal. Ce Responsable exerce une autorité hiérarchique sur les collaborateurs qui lui sont directement rattachés et dispose d’un lien fonctionnel avec les principaux collaborateurs de la Fonction au sein des Pôles et/ou Métiers et autres Fonctions.

# Présentation de l’équipe et des missions réalisées

Dans cette partie, je vais vous présenter l’équipe, les tâches et les formations effectuées lors de ce dernier semestre ainsi que le contexte de mes missions et les objectifs visés par mon maitre d’apprentissage et moi-même.

## L’équipe Cloud

L’équipe Cloud de BPLS fait partie des équipes IT, elle devient de plus en plus indispensable pour toutes les autres équipes qui travaillent sur des applications déployées sur le Cloud ou qui préparent la migration de leurs applications vers ce dernier.

Grâce à l’acquisition d’un nouveau type de cloud (IBM cloud dédié DMZR) par le groupe BNP PARIBAS, l’activité de l’équipe prend une tournure importante en ce moment, elle va avoir la possibilité de réaliser plus de projets de migration sur cette nouvelle plateforme, et c’est pour cela des nouveaux recrutés ont rejoint l’équipe récemment.

### Missions de l’équipe

Étant donné que le cloud est nouveau pour l’entreprise, notre équipe a pour objectif d’instaurer une culture autour de cette technologie chez les collaborateurs, commençant par les équipes techniques en leurs expliquant l’intérêt de passer au cloud et de l’utiliser à la place de nos serveurs interne pour l’hébergement des applications tout au long de leurs cycles de vie.

En parallèle de ça, notre équipe accompagne l’équipe DEVOPS lors des déploiements des nouvelles applications sur le cloud et la migration des anciennes vers ce dernier.

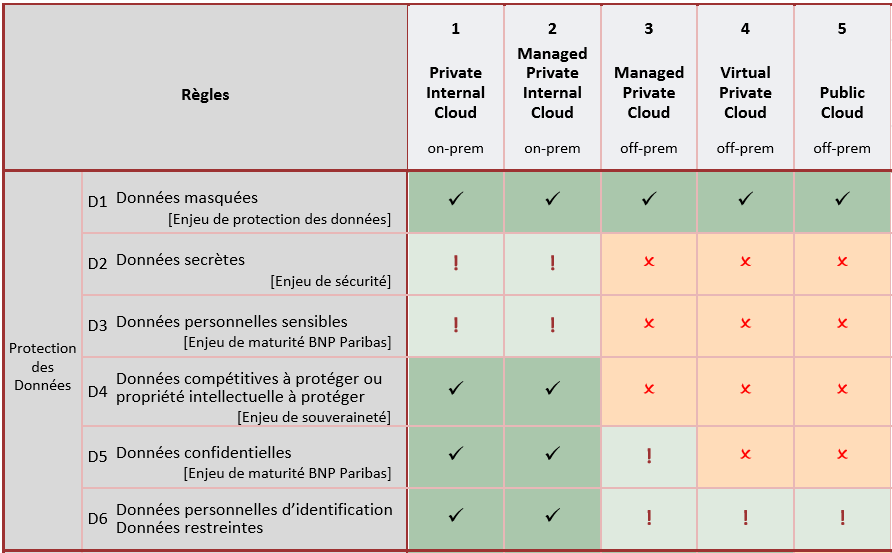
### Le cloud chez BNP PARIBAS

Le cloud computing est la capacité de l’utilisation des infrastructures informatiques et des logiciels sans avoir besoin de les installer sur les machines locales. En réservant des capacités de calcule en fonction du besoin et aussi l’utilisation des services gérés qui permettent de faciliter les déploiements et le monitoring des applications.

Pour répondre aux exigences réglementaires extrêmes de l'industrie financière (plus de 200 exigences de conformité́ européennes et internationales), BNP PARIBAS utilise plusieurs types de cloud en fonction le type des données traités:

1. **Private Cloud on-prem** : Cloud privé défini et construit par BNP Paribas, hébergé et administré en interne dans les centres de calcul du Groupe et mis à disposition des Métiers.
2. **Managed Private Internal Cloud on-prem** : Cloud dédié, défini et construit par un grand fournisseur de services Cloud, hébergés et administrés dans les centres de calcul du Groupe et mis à disposition des Métiers. L’objectif est de tirer parti des standards et des innovations d’un Cloud Public tout en profitant des propriétés d’un Cloud privé (sécurité, souveraineté, …)
3. **Managed Private Cloud off-prem** : Cloud dédié, défini et construit par un grand fournisseur de services Cloud, hébergés et administrés dans les centres informatiques du fournisseur et mis à disposition des Métiers. L’enjeu ici est économique : tirer profit des économies d’échelle dans la gestion d’un centre informatique via un fournisseur majeur de services Cloud, tout en conservant un certain cloisonnement physique vis-à-vis d’autres clients du fournisseur.
4. **Virtuel Private Cloud off-prem** : adaptation des services du Cloud Public, proposant à travers des techniques de virtualisation, un certain degré de cloisonnement entre les différentes ressources pour mieux adresser les besoins de ségrégation des traitements comprenant des données à protéger.
5. **Cloud Public off-prem** : ensemble de ressources informatiques (serveurs, stockage, applications) mutualisées et partagées simultanément par les clients - la protection des données et des traitements étant assurée par un système logiciel de gestion des droits d’accès.

Le tableau suivant résume les cas d’utilisation des différents types de cloud :



✓ usage autorisé sous décision locale de l’Entité.

! usage autorisé sous décision locale de l’Entité avec une attention particulière concernant la protection de nos actifs.

? usage interdit - dérogation nécessitant une remontée au niveau du Groupe.

### L’Importance du Cloud pour l’entreprise

Le cloud est considéré comme une source d’opportunités pour l’entreprise pour les raisons suivantes :

1. Créer un avantage compétitif : Accéder à des technologies et à des partenaires permettant d’accompagner le développement de produits ou de services innovants et différenciants.
2. Améliorer le Time To Market : Accéder à des plates-formes et outils permettant d’accélérer les développements et les cycles de déploiement.
3. Gagner en flexibilité : Utiliser les services Cloud pour assurer un passage rapide à l’échelle (élasticité). Passer d’un modèle financier d’investissement et d’immobilisation à un modèle de paiement à l’usage pour mieux gérer l'incertitude du succès commercial d’un service.
4. Rester compétitif : Construire des initiatives externes via des collaborations basées sur le Cloud (par exemple Blockchain, Apprentissage et Intelligence Artificielle) et maintenir notre présence sur le marché.
5. Améliorer l’efficience opérationnelle : S’appuyer sur des plates-formes et des services permettant de décharger et basculer certains traitements sur des solutions partagées plus efficientes
6. Améliorer l’expérience client : Fournir des services orientés utilisateurs (y compris des APIs) pour créer des parcours utilisateurs innovants, en intégrant des solutions alternatives de partenaires externes.
7. Améliorer la disponibilité : S’appuyer sur des plates-formes Cloud mondialement distribuées afin d’améliorer la disponibilité des services métiers et leur résilience face à de potentielles catastrophes (climatiques, sociales, géopolitiques, …).
8. Attirer et retenir nos talents : Permettre au collaborateurs d’accéder aux dernières technologies et outils.

## la mission réalisée

### Développement et déploiement des applications de démonstrations sur le Cloud

#### Contexte

Le Cloud Dédié aussi appelé dMZR (Dedicated Multi Zone Region) est un ensemble de composants d’infrastructure et de services, copiés du monde IBM, qui seront installés dans les datacenters de BNP Paribas pour permettre la mise à disposition de services qu’IBM offre sur son Cloud Public au niveau mondial. le même écosystème sera créer dans les datacenters de l’entreprise. Donc, les données resteront dans le réseaux BNP, protégées par le système de sécurité interne.

L’arrivée prochaine ce nouveau Cloud à BNP Paribas implique de nombreux changements pour L’entreprise, c’est toute une partie d’outils et méthodes de travail utilisé maintenant qu’il faut revoir pour s’adapter à cette nouvelle technologie et à ces nouveaux usages. Dans ce cadre, et afin de garantir la sécurité des applications, le groupe BNP Paribas a défini un ensemble d’architectures de référence qui permettent de guider les ingénieurs DEVOPS lors des déploiements de leurs applications sur le DMZR.

Pour objectif d’avoir un premier retour d’expérience, un projet a été mis en place pour tester la faisabilité des déploiements des applications tout en respectant les architectures de références. Ce projet qui m’a été confié consiste à développer et déployer des applications de démonstration, tout en restant conforme aux exigences du groupe BNP Paribas.

#### Tache effectuées

En autonomie, j’ai travaillé sur le projet des architectures de référence qui a pour objectif de réaliser les étapes suivantes pour chaque architecture :

**Analyse** : le but est d’identifier les dépendances avec des autres architectures, les technologies à utiliser pour le développement de l’application de démonstration, et enfin les besoins en matière d’infrastructure.

**Développement** : développer une application minimaliste qui permet la réalisation de l’architecture.

**Déploiement** : la configuration de la Toolchain (chaine d’outils) Devops pour déployer l’application sur le Cloud.

**Documentation** : la rédaction de la documentation qui contient les détails de toutes les étapes cités ci-dessus, ce qui permet de faciliter la réalisation de l’architecture pour des prochains projets.

Jusqu’à la rédaction de ce rapport, les architectures qui ont été vérifiées sont les suivantes :

* L’accès d’un conteneur au COS à travers d’un Système de fichier local (PVC)
* Déclenchement d’application Batch à la réception d’un fichier dans le COS
* L’accès d’un utilisateur au COS pour chargement et téléchargement de fichiers

Avant de détailler ces architectures, je vais donner les définitions nécessaires pour la bonne compréhension :

**Batch**:

En informatique, un Batch est un programme qui s'exécute sans aucune interaction avec l'utilisateur. Une demande d'impression ou l'analyse des logs d'un site Web sont des exemples de Batch. Dans les grandes entreprises, les Batchs sont de deux types, ceux qui sont lancés par un utilisateur du système et ceux qui s'exécutent automatiquement à un moment donné.

**Kubernetes** :

Kubernetes est une plateforme open source d'orchestration de conteneurs qui automatise une grande partie des processus manuels impliqués dans le déploiement, la gestion et la mise en échelle (Scaling) des applications conteneurisées.

**Pod :**

C’est le plus petit composant de Kubernetes. Il représente une instance unique d'un processus en cours d'exécution dans un cluster, sur un Pod on peut tourner un ou plusieurs conteneurs tels que les conteneurs Docker. Lorsque plusieurs conteneurs s'exécutent sur un Pod commun, ils sont considérés comme une seule entité et partagent les ressources. En règle générale, un Pod exécute un seul conteneur mais ça existe des cas d’usage où lancer multiple conteneurs sur le même pod est nécessaire mais ça reste un cas d’usage avancée.

**Cloud Object Storage COS :**

C’est le service d’Ibm Cloud de stockage d’objets. Accessible via une interface web ou CLI (Commande line interface), ce service propose le stockage et la protection des fichiers dans des ”Buckets ” (sceaux).

Pour utiliser ce service, il faut commencer par créer un Bucket pour y stocker des fichiers au format objet. L’accessibilité des fichiers est assurée de n’importe où, il suffit juste d'avoir accès à internet, c’est l’avantage du Cloud..

**Bucket :**

Un Bucket COS est une ressource de stockage dans le IBM Cloud disponible dans le COS (Cloud Object Storage), une solution pour le stockage d'objets. Les Buckets des COS sont l’équivalent des dossiers dans un système de fichiers, stockent des objets, qui se composent de données et de leurs métadonnées.

**Docker:**

Docker est une plateforme open source pour le développement, la livraison et l'exécution d'applications. Afin de faciliter la livraison des applications, Docker sépare les applications de l’infrastructure, Docker permets une gestion simplifiée de l’infrastructure et de l'application. Aussi, l’application de la méthodologie de Docker pour livrer, tester et déployer le code, peut réduire considérablement le délai entre l'écriture du code et sa mise en production.

**Image Docker :**

Une image Docker est un fichier statique non modifiable. Ce dernier encapsule les éléments nécessaires pour l’exécution de l’application sur la plateforme de conteneurisation Docker

**Conteneur:**

La virtualisation traditionnelle ( virtualisation matériel) s’appuie sur un [hyperviseur](https://www.ionos.fr/digitalguide/serveur/know-how/conteneurs-informatiques-virtualisation-sans-emulation/serveur/know-how/quest-ce-quun-hyperviseur/) (VMware, virtualBox …), qui répartit les ressources de l’hébergeur sur les systèmes hôtes. Dans le cas de la virtualisation par conteneur, aucun système d’exploitation supplémentaire n’est démarré. Au lieu de cela, l’OS commun forme des instances isolées. Ces conteneurs virtuels fournissent aux applications un environnement complet d’exécution.

**Cluster Kubernetes :**

Un cluster Kubernetes est un ensemble de machines permettant l’exécution des applications conteneurisées. Au minimum, il contient un nœud Control Plane et une ou plusieurs autres nœuds. Le nœud Control Plane est responsable du maintien de l'état souhaité du cluster, Les autres nœuds exécutent les applications.

**Toolchain Devops :**

Une Toolchain DevOps est un ensemble d'outils permet l’automatisation des tâches de développement et de déploiement des applications. Ce qui est très important dans le cadre des projets d’applications à une complexité importante.

**PVC** :

Au sein d’un Cluster Kubernetes, Réservation de volume persistant (PVC) est une demande envoyée à IBM Cloud permettant de mettre à disposition du stockage avec un type et une configuration spécifiques. En fonction de la configuration définie dans la classe de stockage, l'unité de stockage physique est commandée et mise à disposition du Cluster.

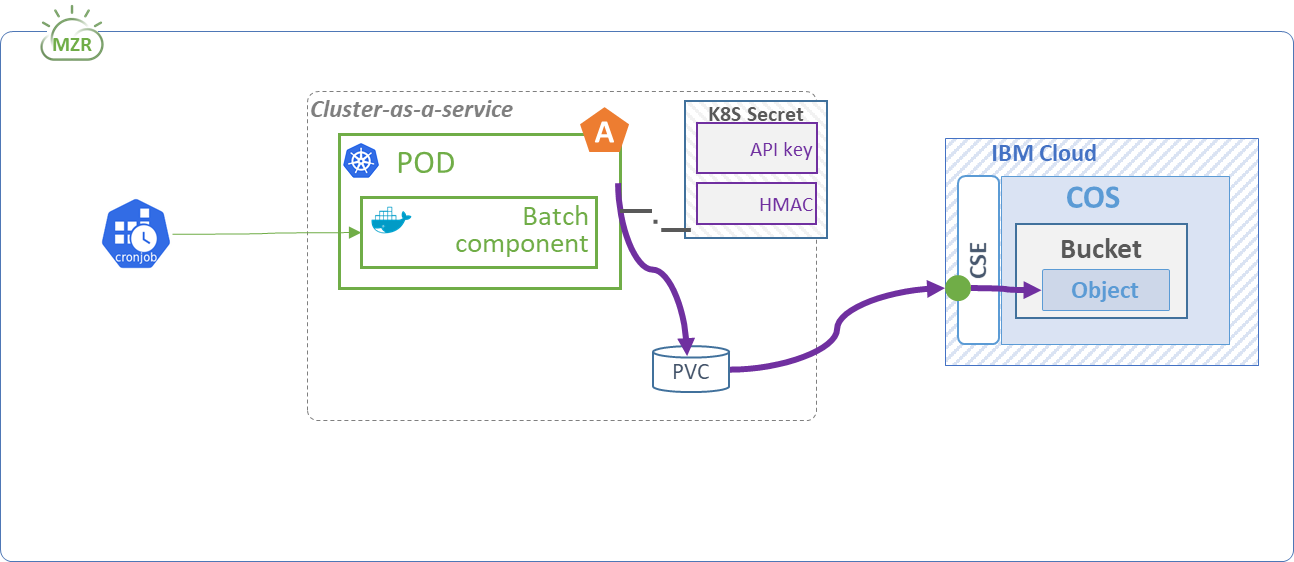
**CI/CD pipeline :** Un pipeline CI/CD est une série d’étapes qui doivent être effectuées afin de délivrer une application. Les pipelines d’intégration continue/livraison continue (CI/CD) sont une pratique axée sur l’amélioration de la livraison de logiciels en se basant sur l’approche Devops.

##### Analyse :

###### Dépendance :

Maintenant que les définitions nécessaires sont données, Voici la première architecture sur laquelle j’ai travaillé :

* Déclenchement d’application Batch à la réception d’un fichier dans le COS

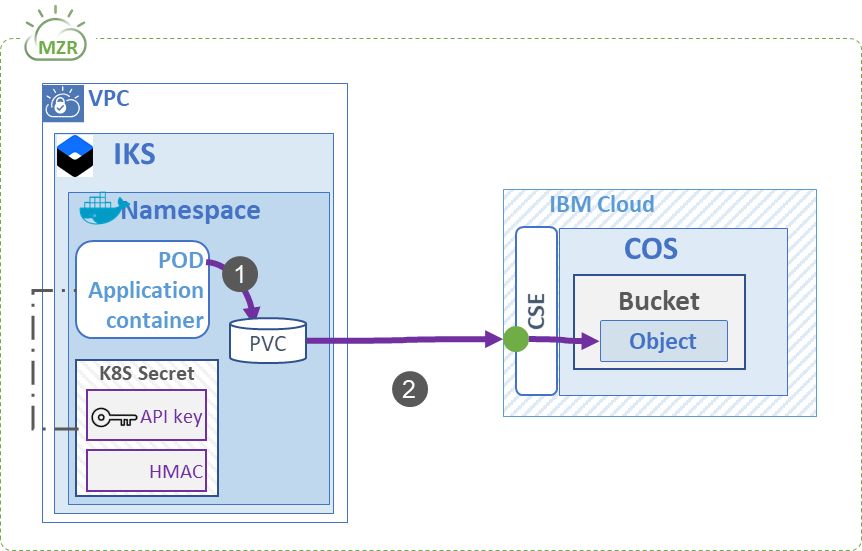


Voici le fonctionnement décrit par cette architecture :

* Le composant batch est déployé périodiquement par un CronJob Kubernetes quelques minutes après le dépôt planifié du fichier dans le COS (à des jours et heures bien précis).
* L'application Batch télécharge le fichier à partir du Bucket via l'API S3.
* L’application Batch supprime le fichier du Bucket quand son traitement est fini.

Après l’analyse de cette architecture, il est constatable qu’elle est reliée à deux autres architectures, la première définit l’accès au COS à partir d’un Conteneur, et la deuxième traite le chargement de fichiers dans un COS.

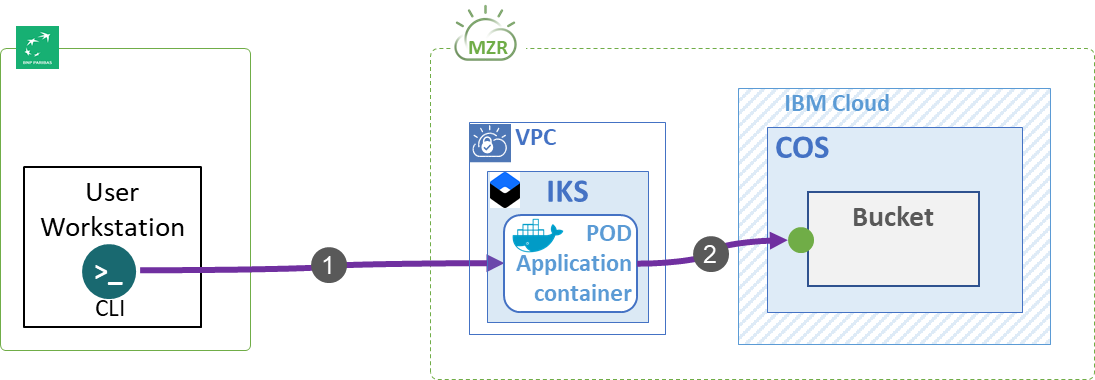
* L’accès d’un conteneur au COS à travers d’un Système de fichier local (PVC)



le fonctionnement décrit par cette deuxième architecture est le suivant :

1. Dans le fichier de déploiement du POD, la PVC précédemment créée est spécifiée. Lorsque le POD est déployé, il monte la PVC (donc un Bucket COS) en tant que système de fichiers local.
2. Kubernetes initie une connexion HTTPS à l'URL du Bucket via son CSE (Cloud Service Endpoint). Il s'authentifie au Bucket via son HMAC

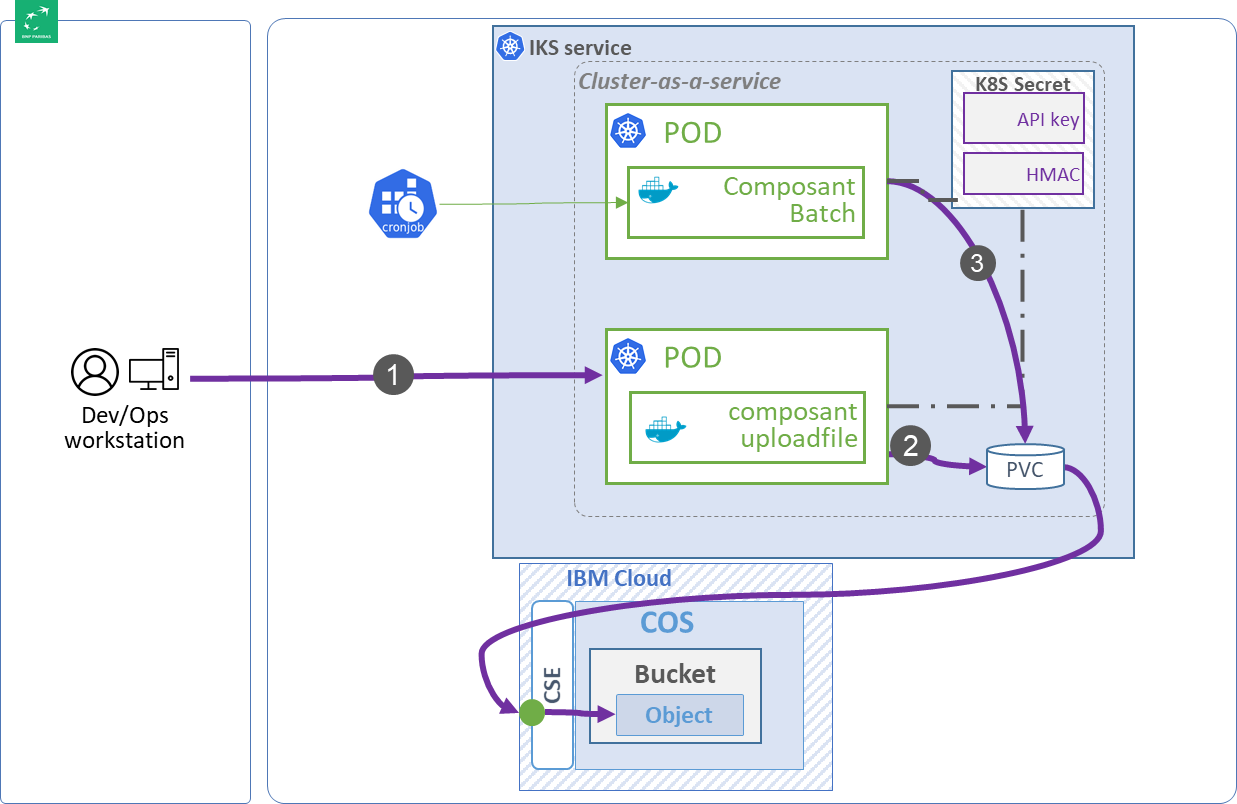
* L’accès d’un utilisateur au COS pour chargement et téléchargement de fichier



Voici le fonctionnement décrit par cette dernière architecture :

1. La station de travail de l’utilisateur se connecte à une application qui agit comme une passerelle et expose des API. L’utilisateur s’authentifie via les clés API ou les certificats de l’application
2. L’application peut accéder au COS de 2 manières différentes :
   1. Accès HTTP direct via l’API S3
   2. À partir d’un PVC monté en tant que système de fichiers local sur le conteneur (deuxième architecture)

Les trois architecture présentées ci-dessus sont complémentaires, en les combinant, on a l’architecture suivante :



Cette architecture complète permet :

1. La connexion au composant UploadFile exposant une API et qui agit comme une passerelle vers le COS
2. Le chargement des fichiers dans le COS.
3. Le déploiement programmé du composant batch qui retire le fichier du COS pour le traiter

###### Choix de technologie :

Toujours dans l’étape d’analyse des architectures, après l’identification des dépendances entre les architectures et la réalisation de l’architecture complète pour la vérification du bon fonctionnement de l’ensemble des architectures en question, il faut identifier les technologies à utiliser pour le développement de l’application.

Deux composants sont à développer, le composant Batch et le Composant UploadFile, Voici les Fonctionnements attendus de chaque composant, la technologie choisie, et la justification des choix :

**Le composant Batch :**

Il doit pouvoir récupérer un fichier du volume PVC (Bucket du COS) qui est monté à son conteneur comme un système de fichier local, il doit par la suite faire un traitement quelconque sur le fichier et écrire dans les logs ce qui a été fait, et à la fin du traitement il doit supprimer le fichier du volume.

Spring Batch est un Framework open source pour le développement des Batchs. Il s'agit d'une solution légère et très répandue dans les entreprises, notamment à BNP Paribas. Cela fait de lui le bon choix pour le développement du composant Batch.

**Le composant UploadFile :**

Il doit exposer un API qui permet aux utilisateurs de charger un fichier dans le COS. Ce qui est Facile à développer avec Spring Boot. Aussi, Spring Boot est répandu dans l’entreprise, donc c’est la technologie choisie pour le développement du composant UploadFile

##### Développement :

**Le composant Batch :**

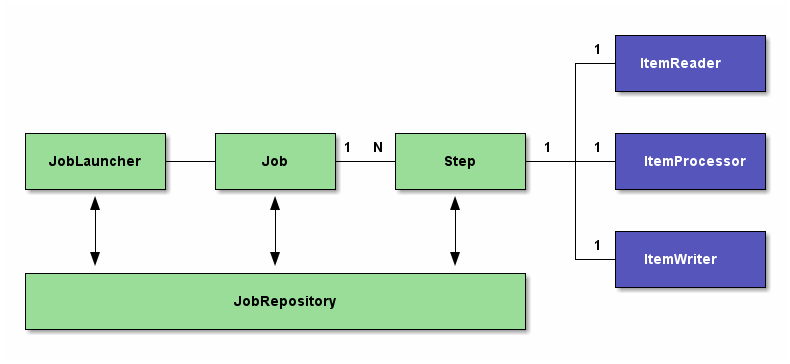
L’objectif est de développer un service qui récupère un fichier CSV du Système de fichiers locale, lire les données dans le fichier, et effectuer une transformation sur ces derniers.

Le fichier CSV a toujours le même Nom « data.CSV » et ses données sont écrites sur n lignes et 2 colonnes, chaque ligne représente une personne, la première colonne représente le nom de la personne et la Deuxième colonne représente le prénom de la personne. La transformation à effectuer sur les données du fichier est de transformer le nom et le prénom de la personne de la minuscule au majuscule.

Ce composant est développé en Spring Batch, donc avant de parler du code, c’est nécessaire de comprendre comment le Spring Batch fonctionne.

Pour le développement en Spring Batch on utilise principalement les trois outils suivants:

* Le **JobLauncher** : son rôle est de lancer le Batch. Il est configurable pour se lancer automatiquement ou pour être déclenché par un évènement
* Le **Job** : il s'agit du composant qui représente la tâche à qui on délègue la responsabilité du besoin métier traité dans le programme. Il est chargé de lancer de façon séquentielle une ou plusieurs étapes (Step).
* La **Step** : c'est le composant qui représente le besoin métier à traiter. Il définit le trois sous-composants suivants :
  + **ItemReader** : responsable de la lecture des données d’entrée.
  + **ItemProcessor** :  responsable de la transformation des données lues.
  + **ItemWriter** : responsable de la sauvegarde des données après transformation.
* **JobRepository** : c'est le composant responsable de l’enregistrement des statistiques issues du monitoring



Dans le cas de notre besoin, le développement est simple, il faut un JobLauncher pour lancer le Job automatiquement lors du lancement du programme, un JobLIstener pour ouvrir le fichier CSV et lire les données, un JobProcessor qui transforme les données de la minuscule en majuscule.

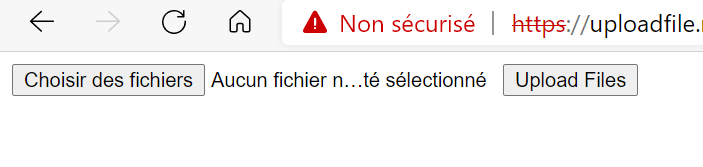
**Le composant UploadFile :**

Ce composant est pour but de permettre à un utilisateur de charger un fichier dans le COS, donc il agit comme une passerelle et expose une API à travers laquelle l’utilisateur se connecte pour charger un fichier.

Le composant UploadFile doit déposer le fichier dans la PVC qui serait montée au POD sur lequel le conteneur de l’application est déployé. En faisant ça, une fois déployé, le Composant Batch pourra récupérer le fichier pour le traiter.

Deux Solutions sont possibles, développer juste le Back-End et envoyer le ficher avec Postman ou un outil équivalent qui exécute des commandes post sur l’API exposée, ou, en plus du Back-End développer un Simple Front qui facilite le chargement coté utilisateur. La deuxième solution a été choisie, pour la facilité de son implémentation avec Spring Boot.

L’interface développée est la suivante :



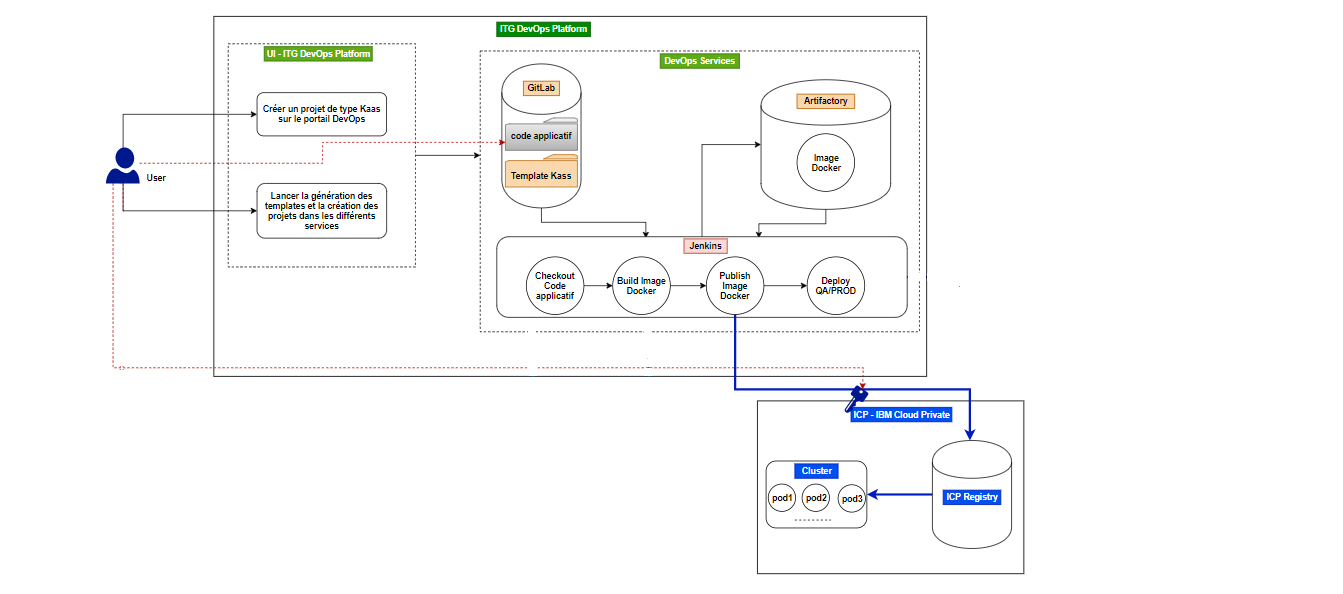
Concernant le Back-End, il est constitué de deux méthodes mappées a deux Paths : @RequestMapping("/")qui permet le chargement de la page du Front.

@RequestMapping("/upload") qui permet de vérifier l’intégrité du fichier chargé et le stocker dans le volume monté, dans les deux cas, succès ou échec, la fonction retourne un message au front.

##### Déploiement :

Une fois l’étape de développement est finie on passe à l’étape de déploiement. BNP Paribas possède une plateforme DEVOPS qui propose des services de DEVOPS, et parmi les Service proposées, il y a pipeline CI/CD pour le KaaS,et c’est ça le service le plus adapté qui facilite le déploiement des deux composants sur le cloud.

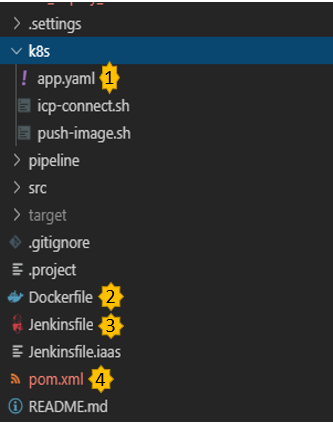
Un Namespace précédemment crée sur le Cluster Kubernetes de l’entité BPLS pour le projet d’architectures de référence



Fonctionnement du pipeline CI/CD pour le déploiement sur IBM Cloud Privé

Pour déployer les deux composants précédemment développés en utilisant le Pipeline CI/CD décrit ci-dessus, on commence par la création d’un projet de type Kaas sur la plateforme Devops interne à BNP Paribas. une fois le projet est créé l’utilisateur lance la génération du Template.

Le Template est généré sur GitLab et il contient les fichiers suivants :



Avant de modifier les fichiers 1,2 et 3, il faut d’abord créer les Jar (Java ARchive) pour les deux composant Batch et UploadFile, puis les pousser dans le Repo Git du Template. Une fois c’est fait voici les modifications à effectuer sur les fichiers :

1. app.yaml :

C’est le fihier de configuration Kubernetes, il contient l’état voulu du déploiement sur le Cluster, Ici on déclare les éléments suivants :

* + PersistentVolumeClaim : permet de réserver du stockage pour pouvoir le monté à un Pod sous la forme d’un volume
  + CronJob : permet de créer un job qui déploie le Pod Batch. Aussi, dans cette partie, on spécifie les ressources nécessaires au fonctionnement de l’application et on monte le volume créer précédemment.
  + Pod : permet de créer un Pod qui sera omniprésent tout au long du cycle de vie de l’application, sur lequel s’exécute le conteneur UploadFile, on spécifie Aussi les ressources nécessaires et on monte le volume créé précédemment
  + Service : permet d’exposer le Pod UploadFile sur le port 8080
  + Ingress : un objet Kubernetes permet de gérer le trafic http externe vers le service qui expose le Pod UploadFile

1. Dockerfile :

C’est le fichier qui permet la construction de l’image Docker, on précise l’image de base, on crée un dossier, on copie le jar de l’application dans le dossier et au finale une commande qui lance le Jar.

Ici on cherche à déployer deux composant donc on crée un autre fichier DockerFile qui aura les meme configurations

1. Jenkinsfile :

Le fichier Jenkinsfile du template généré décrit les étapes nécessaires pour la construction de la chaîne CI/CD (Configuration des variables d'environnement, Build, Publication de l'image Docker, Déploiement, ...).

Voici les étapes principales constituant le pipeline CI/CD pour déployer l’image Docker buildée sur KAAS.

**Création de l'image Docker :**

Dans l'étape (stage en anglais) "Build" du Jenkinsfile, on spécifie les différentes actions pour la création de l’image Docker, principalement les deux commandes : docker login et docker build. Les paramètres passés sont des variables d'environnement initialisées dans le bloc environnement au début du fichier Jenkinsfile.

**Publication de l'image Docker sur Artifactory.**

Suite à la création de l'image Docker, la publication de l'image se fait sur Artifactory, définie dans une nouvelle étape sous le nom "Publish image to Artifactory". Cette étape exécute les deux commandes : docker login et docker push. Aussi, les paramètres passés sont des variables d'environnement initialisées dans le bloc environnement au début du fichier Jenkinsfile.

**Déploiement de l'image sur le cloud privé**

Pour déployer l'image Docker sur KaaS Cloud privé, on choisit l'environnement de déploiement, puis, le script se connecte au cloud avec la clé API qu’on ajoutera par la suite à Jenkins secrets.

Dans cette étape le fichier de configuration app.yaml sera évoqués pour définir le déploiement sur le Namespace ICP.

Une fois que la configuration du Pipeline sur Gitlab est terminée, il faut passer à l’instance jenkins créer par la plateform Devops et effectuer les opérations suivantes :

**Ajouter la clé API d’IBM Cloud :**

Pour que Jenkins puisse se connecter au Namespace sur le Cloud et déployer l’application il faut qu’il se dispose d’une clé API Valable, pour cela il faut générer une clé API sur le Cloud IBM puis l’ajouter à un secret sur Jenkins de portée Global, le nom de ce secret doit être le même que celui ajouté comme variable d’environnement au Jenkinsfile.

**Lancer le Pipeline :**

A cette étape-là, la configuration du Pipeline est complète, Jenkins récupère le Jenkinsfile ainsi que les autres fichiers de configuration du Gitlab, exécute le scripte de Jenkinsfile qui évoquera lui-même les autres fichiers de configuration dans les différentes étapes.

En , pour tester l’application il faut accéder au Pod UploadFile a travers le lien précisé dans le fichier de configuration app.yaml

# Bilan des acquis

## Difficultés rencontrées et solutions apportées

Avant de parler des difficultés rencontrées, il est important de rappeler comment s’est déroulé ce semestre d’une façon générale. Dans un contexte inédite marqué par la COVID19, le télétravail est instauré à l’entreprise. Pendant Le confinement je faisais cinq jours sur cinq en télétravail, et pendant le reste de la période je faisais deux jours sur bureau et le reste en télétravail. Tout d’abord le fait d’être en télétravail la majorité du temps était dure pour moi car je n’étais pas forcément habituée à ça. De plus, le fait de changer d’équipe et de commencer à travailler avec des nouveaux collaborateurs et faire leurs connaissances virtuellement n’était pas facile au début. Dans ce contexte, il a fallu que je m’adapte assez rapidement pour entamer mes missions et aussi faire preuve d’assiduité à l’université. Cela m’a permis de démontrer mes capacités d’adaptabilité et de réactivité.

Dans le cadre de mes missions, j’ai eu parfois des difficultés à comprendre les aspects techniques compliqués du Cloud. Pour aboutir à une réponse efficace j’ai demandé de l’aide aux membres de mon équipe. Ce qui m’a amené à développer et améliorer mon aptitude au travail collaboratif.

Ma principale difficulté est de gérer mon temps aussi bien pour réaliser mes tâches que pour gérer le calendrier d’alternance. Je trouve que c’est difficile de gérer son temps et d’organiser son travail. Une chose que j’ai pu surmonter « si je peux dire avec succès » étant donné que j’ai réalisé les missions qui m’ont été attribuées tant sur le plan professionnel que scolaire.

## Résultats obtenus/compétences acquises

J’ai la chance de travailler dans une grande entreprise telle que BNP PARIBAS. Ce qui me permis de comprendre de mieux en mieux le Fonctionnement des entreprises du leasing. A travers mes diverses missions, j’ai pu concrétiser et améliorer mes différentes compétences techniques typiquement en suivant de près les processus de déploiement d’applications.

Autrement, la diversité des missions tout au long de l’alternance a renforcé les compétences acquises durant mes études mais également m’a permis de toucher à plusieurs types de Métiers, notamment le métier d’architecte IT, ingénieur cloud ainsi qu’ingénieur DEVOPS . Pour conclure, ce semestre fut très riche et constructif tant d’un point de vue professionnel que personnel.

## Perspective d’avenir

Durant la période prochaine je travaillerai sur la réalisation des applications pour démontrer la faisabilité des patrons d’architecture recommandés par le Groupe, vu le nombre et la complexité de réalisation de ces applications, cette mission vas m’occuper tout au long des mois qui viennent jusqu’à la fin de mon alternance.

La réalisation des patrons d’architecture consiste dans le choix d’outils et le de langages de programmation, le développement de l’application, le déploiement et enfin la capitalisation de tous le processus. Cette mission me permettra d’acquérir des compétences nécessaires pour le travail d’un ingénieur DEVOPS.