Tristan BOURHIS
Promotion P2023 – 5ème année
Majeure MIN

Mémoire de Projet de Fin d'Etudes

Software engineer financing

Lieu du stage : Société Générale - Ville - France
Dates du stage : 6 février 2023 – 28 juillet 2023
Service où s'est déroulé le stage : GBSU/FTB/RPP
Nom du tuteur industriel : Alexandre Chennebault, Cécile DEROUBAIX
Nom du tuteur pédagogique : Alexandre Chennebault, Cécile DEROUBAIX

REMERCIEMENTS

Je tiens à exprimer ma sincère gratitude envers plusieurs personnes qui ont grandement contribué à mon expérience de stage :

Tout d'abord, je remercie chaleureusement Alexandre CHENNEBAULT, mon tuteur de stage, ainsi que Cécile DEROUBAIX, ma tutrice de stage. Leur encadrement attentif, leurs conseils avisés et leur soutien constant ont joué un rôle crucial dans la réussite de ce stage.

Je souhaite également exprimer ma reconnaissance envers Jun-Wah LEE, notre référent technique, dont les orientations précieuses ont illuminé le chemin du développement tout au long du projet.

Un remerciement spécial va à Ludovic LE GOFF du support technique, dont l'aide rapide et efficace a été inestimable pour surmonter les défis techniques rencontrés.

Je tiens à exprimer ma gratitude envers Frédéric RENOUARD, l'analyste métier à l'origine du sujet de travail de ce stage, pour sa vision novatrice et ses idées constructives qui ont guidé mes efforts.

Je tiens à exprimer mes sincères remerciements à Tatiana MUNTEANU-COLCEV et Clément MORET, nos managers, pour leur leadership inspirant et leur guidance tout au long de ce projet. Leur soutien constant, leurs conseils éclairés et leur engagement envers notre réussite ont été des facteurs essentiels dans l'atteinte des objectifs fixés. Votre encadrement a créé un environnement propice à l'apprentissage et au développement professionnel, et je suis reconnaissant pour cette opportunité précieuse. Votre expertise et votre dévouement ont joué un rôle déterminant dans la réalisation de ce projet.

Un immense merci s'adresse à toute l'équipe FSI pour son accueil chaleureux, sa collaboration et son partage de connaissances. Votre contribution collective a été essentielle pour la réalisation de ce projet.

Je tiens également à remercier Zehira HADDAD, responsable de Majeure à l'école, ainsi qu'Amin ZAMMOURI, responsable de Majeure et tuteur pédagogique, pour leur encadrement et leur soutien dans le suivi de mon parcours académique et professionnel.

Enfin, je suis reconnaissant envers l'école EPF pour les solides fondations académiques et les opportunités qu'elle m'a offertes, me permettant de mettre en pratique les compétences acquises au fil de mes études.

Ces remerciements sincères sont une expression de ma gratitude envers tous ceux qui ont contribué à façonner cette expérience mémorable et formatrice. Votre appui a été inestimable et a enrichi mon parcours de manière significative.

Rappel de l'intitulé du stage

Software engineer financing

Résumé

Comprendre les besoins utilisateurs lors des workshops avec les différents acteurs métiers ; Coder et documenter ce qui est produit (interfaces, back end, API) ; Donner l'impulsion dans l'innovation (réalisation de Proof of Concept), proposer de nouvelles techniques et montrer une force de proposition ; Optimiser les solutions techniques si nécessaire (monitoring et amélioration de la performance : latence, mémoire, etc.) définir des tests automatiques ; Participer à l'automatisation de déploiements.

Mots clés :

Logiciel, développement, monitoring, Elasticsearch, finance, interfaces, logs, données

Abstract

Understand user needs during workshops with the various business players; Code and document what is produced (interfaces, back end, API); Give impetus to innovation (realization of Proof of Concept), propose new techniques and show strength of proposal; Optimize technical solutions if necessary (monitoring and performance improvement: latency, memory, etc.) define automatic tests; Participate in the automation of deployments.

Keywords :

Software, development monitoring, Elasticsearch, finance, interfaces, logs, data

Engagement à produire un texte original et à ne pas recopier d'autres documents sans citer les sources.

Sommaire

Liste des figures

Liste des tableaux

Glossaire

Acronyme	Détail
API	Application Programming Interface : interface permettant d'échanger des données.
AWS	Amazon Web Services : Services de la société Amazon permettant notamment de mettre en place des machines virtuelles.
CSR	Certificate Signing Request : certificat contenant une preuve d'authenticité et des informations d'identifications
DSM	Daily Stand-Up Meetings : réunion journalière rapide permettant d'établir les objectifs de la journée
ELK	Elasticsearch, Logstash, Kibana : terme utilisé pour désigner la suite de logiciel Elasticsearch, comprenant Logstash et Kibana .
FSI	Financing Services Interface : nom de l'application et de l'équipe dans laquelle se déroule le projet de stage.
FTB	Financing and Transaction Banking : sous-service de GBSU qui couvre les activités de GLBA et GTPS.
GBSU	Global Business Service Unit : service de la Société Générale dans lequel se déroule le projet de stage.
GLBA	Global Banking and Advisory : service de la Société Générale travaillant en étroite collaboration avec GBSU.
GTPS	Global Transaction Payement Services : service de la Société Générale en charge des transactions et des payements.
HTTP	HyperText Transfer Protocol : protocole de communication client-serveur permettant de transférer des données.
JSON	JavaScript Object Notation : modèle de format de données structurées
PFE	Projet de Fin d'Étude : projet réalisé qui est décrit et détaillé par ce document.
POC	Proof Of Concept : démonstration de faisabilité.
RPP	Risk Provision and Profitability : sous-service de FTB qui concerne la gestion des risques.
SG	Société Générale : entreprise au sein duquel se déroule le stage.
SQL	Structured Query Language : langage informatique permettant d'exploiter les bases de données.
SSL	Secure Socket Layer : protocole de sécurité internet basé sur le chiffrement pour garantir la confidentialité, l'intégrité des données et la confidentialité dans les communications internet.
TCP	Transmission Control Protocol : protocole de transmission informatique de données.
TLS	Transport Layer Security : protocole réseau permettant de sécurisation des échanges par réseau informatique
UAT	User Acceptance Testing : terme pour désigner une phase de développement logiciel afin d'effectuer des tests de validation.
URL	Uniform Resource Locator : adresse internet d'une resource.
VM	Virtual Machine : machine virtuelle
VPN	Virtuel private network : réseau internet privé et virtuel.
YAML	Yet Another Markup Language : format de représentation de données souvent utilisé pour des fichiers de configuration d'un logiciel.

Nomenclature

b	Bit : unité élémentaire d'information pouvant prendre deux valeurs (0 ou 1)
Go	Giga octet : unité de mesure de quantité d'information numérique valant 1 000 000 000 d'octets
o	Octet : unité d'information contenant 8 bits
….	

1 Introduction

Le stage se déroule au sein du service **GBSU/FTB/RPP**. **GBSU** (Global Business Service Unit) a pour but de délivrer au jour le jour des services à ses clients pour accélérer les transformations (plus d'informations sur les différents services et l'organisation du groupe sont dans la partie 2.2). Le sous service **FTB** (Financing and Transaction Banking) couvre les finances et les activités de transactions bancaires de deux autres services : GLBA (Global Banking and Advisory) et GTPS (Global Transaction & Payment Services).

Le projet se déroule dans le cadre d'une application interne nommé **FSI (Financing Services Interface)**. Cette application joue le rôle d'un distributeur de données au sein de la banque et de ses différentes applications internes.

Au sein de l'actuelle infrastructure, une perte de temps significative découle de l'absence d'un outil adapté pour satisfaire certains besoins fonctionnels spécifiques. Lorsqu'un collaborateur exprime le besoin d'accéder à des informations relatives aux données de FSI, il est contraint de solliciter le support, qui doit ensuite effectuer une série de recherches manuelles dans divers fichiers pour ensuite restituer ces informations. Cette procédure engendre une perte de temps considérable, d'autant plus que d'importantes quantités de données traversent quotidiennement ce service. La célérité de réaction devient critique pour éviter tout impact négatif. Le support fonctionnel de l'application FSI se trouve fréquemment confronté à ce type de requêtes, engendrant ainsi une charge de travail substantielle pour l'équipe du support ainsi qu'une perte de temps pour les collaborateurs initiateurs de ces sollicitations.

L'importance d'atténuer cette perte de temps ne saurait être sous-estimée. Les conséquences d'une telle inefficacité peuvent se propager dans divers domaines. En premier lieu, la productivité globale de l'entreprise pourrait en souffrir, affectant sa réactivité et sa capacité à répondre rapidement aux défis et opportunités émergents. De plus, une accumulation de retards dans la consultation et la transmission d'informations pourrait entraver la prise de décisions informées, potentiellement ayant un impact sur la qualité des opérations. Cette situation pourrait également engendrer une frustration croissante chez les collaborateurs qui se trouvent confrontés à des délais inutiles. En somme, l'inefficacité engendrée par ce manque d'outil adapté peut avoir des répercussions en cascade, soulignant l'impératif de remédier rapidement à cette problématique pour préserver l'efficacité globale et la réputation de l'entreprise.

À l'ère du numérique, où les données prolifèrent à un rythme exponentiel, la nécessité d'optimiser les processus d'analyse pour une prise de décision éclairée est plus pressante que jamais. Cependant, le défi réside dans la gestion de ces volumes massifs de données de manière rapide et efficace, sans être entravé par des tâches manuelles chronophages. Dans ce contexte, la problématique cruciale émerge : comment automatiser les processus d'analyse des données de manière à accélérer la disponibilité des données pour les utilisateurs finaux ?

Au cœur de cette problématique se trouve le besoin impérieux d'optimiser la chaîne d'analyse des données, depuis leur collecte initiale jusqu'à leur mise à disposition pertinente. Les tâches manuelles traditionnelles, souvent sujettes à des erreurs humaines et consommatrices de temps, freinent la réactivité de l'entreprise face aux opportunités et aux défis émergents. La recherche d'une solution efficace implique une transition vers l'automatisation des processus, permettant une analyse plus rapide et une mise à disposition quasi instantanée des informations critiques pour les parties prenantes.

Par conséquent, la quête pour automatiser les processus d'analyse des données s'impose comme une priorité stratégique dans le panorama actuel de l'entreprise moderne. Les avantages potentiels sont multiples, allant de la réduction des erreurs et des délais à la libération des ressources humaines pour des tâches à plus forte valeur ajoutée. Cependant, cette transition ne se limite pas à une simple implémentation technique, mais nécessite une approche holistique englobant des aspects technologiques, organisationnels et culturels. Dans cette optique, explorer

les méthodes et les technologies qui permettent de concrétiser cette automatisation tout en assurant une collaboration fluide entre les différentes parties prenantes devient essentiel pour répondre à cette problématique centrale.

Le problème est la perte de temps qui est occasionnée par le manque d'outil mis en place. Le fonctionnement de certaines parties de la banque ne doit pas être handicapé ou ralenti par cette perte de temps.

La question de la confidentialité entre aussi en jeu car l'utilisateur final ne doit pas avoir le droit d'accéder à toutes les informations de l'application mais qu'à celles qui le concernent.

L'objectif est donc de mettre en place une interface accessible par les différents acteurs de l'application FSI afin de répondre à des besoins fonctionnels et d'automatiser des tâches actuellement faites manuellement, ce qui permettra de diminuer les communications entre les clients et le support de l'application pour apporter un gain de temps considérable aux deux parties.

Une solution technique avec une interface doit être mise en place et déployée sur un serveur, celle-ci permettant d'accéder aux informations des filtres. Cette interface devra être accessible non seulement au support de l'application FSI mais aussi aux utilisateurs finaux de l'application pour leur éviter d'avoir à passer par le support.

L'interface mis en place devra être sécurisée et extensible, assez performante pour fonctionner avec un nombre élevé de données, respecter certaines normes de sécurité. Bien prévoir la quantité de données est aussi important pour éviter les problèmes de surcoût.

Ce rapport suit une structure organisée qui se décompose en plusieurs sections, fournissant une vue d'ensemble détaillée du projet au sein de l'entreprise.

La première section examine en détail l'entreprise, son contexte général et sa structure organisationnelle. Les politiques et procédures du groupe sont également abordées, couvrant des aspects tels que la sécurité de l'information, la conformité, la gestion des risques, les ressources humaines et la lutte contre la fraude et le blanchiment d'argent.

La seconde section se focalise sur le projet d'étude. Son contexte est présenté, ainsi que la problématique visée et les objectifs qui ont été définis. La méthodologie utilisée est ensuite exposée, et la solution technique mise au point est décrite en détail, englobant les outils employés, la preuve de concept (POC), le développement du projet, sa mise en production et l'exploitation d'outils internes.

La troisième section s'attache aux résultats obtenus et à leur analyse, mettant en avant les accomplissements et identifiant les domaines où des améliorations pourraient être apportées.

Enfin, la section de conclusion résume les implications techniques du projet et offre également une perspective personnelle sur cette expérience.

C'est ainsi que ce plan se déroule, mettant en lumière chaque phase et aspect de la démarche au sein de ce projet d'envergure.

2 Présentation de l'entreprise

2.1 Présentation générale

La Société Générale est une banque française créée le 4 mai 1864 à la suite de la signature du décret de financement de la Société Générale par Napoléon III. La mission de cette banque a toujours été de : « promouvoir le développement du commerce et de l'industrie en France ».

Actuellement, la SG (Société Générale) est présente dans 66 pays tout autour du globe et possède plus de 25 millions de clients et 117 000 collaborateurs avec un résultat net de 5,6 milliards d'euros.

Concernant les actifs totaux, il s'agit de la troisième banque française la plus grande, la sixième en Europe et la dix-huitième dans le monde.

Les activités de la SG concernent notamment :

- La banque de détail en France avec la Société Générale, Boursorama et Crédit du nord
- La banque de détail à l'étranger
- La banque de financement
- La banque privée avec gestion d'actifs
- Les services financiers spécialisés
- Les assurances

Le groupe Société Générale se décrit avec les valeurs suivantes :

- Esprit d'équipe : le groupe valorise l'écoute, le travail en équipe, la contribution et le fait d'être unis dans le succès mais aussi dans la difficulté.
- Responsabilité : en tant que banque, le groupe SG contribue à l'économie, au social et au développement environnemental durable de l'économie. Cela implique de grosse responsabilité notamment sur le plan éthique et sur le fait de faire attention à tous les différents risques et leurs aspects. Le groupe attache autant d'importance dans la manière d'achever ses objectifs et d'obtenir des résultats que dans les résultats eux-mêmes.
- Innovation : un des objectifs majeures et l'évolution constante pour adapter les solutions en travaillant avec le client et particulièrement en tirant partie de l'innovation technologique.
- Engagement : l'engagement de la Société Générale découle de la satisfaction continue des clients en s'efforçant quotidiennement de faire une différence qui contribue à la réussite de leurs projets et de ceux du groupe.

2.2 Organisation du groupe

2.2.1 Organisation générale

Le groupe SG se divise en trois unités principales qui sont :

- General Management : service dédié pour la stratégie du groupe et son application
- Business Units : qui regroupe toutes les entités liées au business
- Service Units : qui regroupe toutes les entités liées aux services

Ces deux dernières unités travaillent généralement ensemble, comme les unités de services mettant en place des outils dédiés au bon fonctionnement et à l'accomplissement des objectifs des unités de business. Au total il y a 25 unités différentes au sein de ces trois services. Chacune d'entre

elles est ensuite redivisé en différents services qui sont eux-mêmes composés de plusieurs sous-services.

Le groupe SG étant plutôt grand et par soucis de praticité, la suite se focalisera sur les unités en rapport avec ce PFE pour les présenter.

Comme évoqué précédemment dans l'introduction, le projet se déroule dans l'unité GBSU (Global Business Service Unit). L'objectif de cette entité est de délivrer des services au jour le jour aux entités de la banque de financement et d'investissement. GBSU se divise aussi en plusieurs sous-services. Celui dans lequel se déroule ce stage est FTB (Financing Transaction Banking) qui couvre les activités de deux Business Units : GLBA (Global Banking and Advisory) et de GTPS (Global Transaction Payment Services) qui concernent toutes les deux le financement de gros et le conseil en financement. Le dernier sous-service dans lequel le projet se situe est RPP (Risk Provision & Profitability) qui est notamment orienté vers l'assistance de GLBA et lié aux risques et à la rentabilité.

Pour résumer, le projet se déroule dans le service GBSU/FTB/RPP qui a pour but de délivrer des services à GLBA concernant les risques.

2.2.2 Organisation au sein de l'équipe

L'organisation au sein de l'équipe reflète une démarche agile, axée sur la collaboration et l'efficacité. La méthodologie agile est rigoureusement suivie, orchestrant les efforts par le biais de pratiques clés telles que les Daily Stand-Up Meetings (DSM), les réunions hebdomadaires (weeklys), ainsi que des sprints de trois semaines. Cette structure de travail bien établie favorise une communication transparente et constante, tout en permettant une flexibilité nécessaire pour répondre aux besoins changeants des projets.

L'équipe, composée d'un groupe multidisciplinaire, comprend des développeurs, des analystes métier, un soutien technique spécialisé en DevOps et un soutien fonctionnel. Cette diversité de compétences et de rôles au sein de l'équipe enrichit les perspectives et optimise la capacité à aborder chaque aspect du projet de manière holistique.

Les réunions DSM quotidiennes permettent à chacun d'être au courant des progrès, des défis et des objectifs de la journée, permettant ainsi une coordination optimale. Les réunions hebdomadaires sont des opportunités précieuses pour discuter des résultats de chaque sprint et planifier les prochaines étapes. Les sprints de trois semaines maintiennent un rythme de travail concentré tout en permettant des ajustements en fonction des besoins émergents.

Cette approche agile favorise un environnement collaboratif où les idées et les solutions sont échangées de manière continue. Elle assure également une réactivité accrue face aux changements et aux exigences évoluant au sein du projet. Grâce à cette organisation agile et à la composition diversifiée de l'équipe, l'équipe est en mesure de déployer des solutions efficaces et adaptées, en ligne avec les besoins et les objectifs de l'entreprise.

2.3 Politiques et procédures du groupe

2.3.1 Politique de sécurité de l'information et de gestion de données

La banque doit élaborer des politiques et des procédures pour protéger les informations confidentielles de l'entreprise et de ses clients contre les cyber-attaques, les violations de données et les fuites d'informations. Le groupe étant une entreprise majeure reconnue internationalement, et en vue du nombre de ses clients et des sommes d'argent en jeu, elle est constamment prise comme cible d'attaque.

Le groupe forme ses collaborateurs en matière de sécurité informatique dans le but de diminuer les risques encourus. Il met aussi en place des procédures rigoureuses en matière de sécurité et d'infrastructure afin de maintenir cette sécurité.

La confidentialité est aussi très importante, il ne faut pas donner plus d'accès que nécessaire à une personne pour la réalisation de ses tâches.

L'entreprise doit aussi être capable d'assurer la continuité de ses activités en cas de perturbation majeure telle qu'une cyber-attaque ou une catastrophe naturelle.

2.3.2 Politique de conformité

Le groupe doit garantir la conformité aux lois et aux réglementations applicables. En tant qu'institution financière, la Société Générale est soumise à de nombreuses réglementations qu'elle se doit d'appliquer. Cela peut concerner la protection des consommateurs, le respect de la vie privée et la lutte contre la fraude, mais cela touche aussi chaque transaction financière ou chaque projet en fonction des différents acteurs qui sont en jeux.

2.3.3 Politique de gestion des risques

La Société Générale se doit de limiter les risques en identifiant, évaluant et gérant les risques associés à ses activités et opérations auxquelles ils se produisent.

La première étape concerne l'identification. Ces risques peuvent être d'ordre financier (crédits, marché, liquidité…), opérationnel (transactions, matériels, procédures…), juridique, de réputation ou de conformité réglementaire.

Pour chacun de ces risques identifiés, il faut évaluer l'impact et la probabilité qu'il se produise afin de déterminer les risques critiques à traiter en priorité.

Il y a ensuite une phase de gestion des risques qui consiste à mettre en place des mesures pour réduire les risques identifiés. Ces mesures peuvent inclure des contrôles internes, des processus de surveillance continue, des politiques et procédures de conformité, des audits et des vérifications.

Un suivi et une évaluation de la gestion des risques et de leurs contrôles sont aussi mis en place pour suivre et évaluer régulièrement l'efficacité des mesures de gestion mises en place.

La communication et la formation régulière sur les risques et mesures de contrôle mises en place jouent un rôle majeur dans la diminution des risques.

2.3.4 Politique de ressources humaines

La politique des ressources humaines dans une grande banque est un aspect crucial de la stratégie globale de l'entreprise qui emploie de nombreux professionnels, la gestion des ressources humaines est essentielle pour attirer, retenir et développer les talents. Cette politique se concentre sur plusieurs aspects clés, notamment la gestion des performances, la rémunération, les avantages sociaux, la formation et le développement professionnel ainsi que la diversité et l'inclusion.

Concernant la gestion des performances il est essentiel d'évaluer les performances individuelles par des notations, ce qui aide l'entreprise à atteindre ses objectifs de performance. Des formations sont aussi mises en place pour aider les employés à acquérir les compétences nécessaires pour la réussite de leurs tâches. Ces formations peuvent inclure des séminaires, des programmes de coaching, des programmes de développement de carrière ou tout simplement des formations en ligne.

La rémunération et les avantages sociaux sont des points clés des ressources humaines. Ils doivent être compétitifs et répondre aux besoins des employés pour attirer les talents les plus qualifiés.

Enfin la diversité et l'inclusion sont des éléments importants de cette politique. Le groupe met en place des environnements de travail inclusifs et cherche à recruter des employés de divers horizons culturels et ethniques.

2.3.5 Politique de gestion des conflits d'intérêts

Cette politique vise à prévenir et à gérer les conflits d'intérêts qui peuvent survenir entre le groupe, ses employés et ses clients pour maintenir la confiance du public dans l'intégrité et l'impartialité de la banque. Cette politique doit être appliquée à tous les niveaux de la banque et doit être mise en œuvre de manière proactive pour détecter, évaluer et gérer les potentiels conflits d'intérêts.

Cela inclut la mise en place de mesures préventives telles que la séparation des fonctions, la divulgation des conflits, la formation des employés et la surveillance de l'application des politiques. Le rôle des employés est essentiel car ils doivent être en mesure d'identifier et signaler tout conflit d'intérêts potentiel à leur supérieur hiérarchique ou à la direction conformément aux procédures établies.

2.3.6 Politique de lutte contre la fraude et le blanchiment d'argent

Le groupe s'efforce de maintenir un environnement bancaire sûr et fiable pour ses clients. C'est pour cela que l'entreprise met en place une série de politiques et de procédures pour lutter contre les pratiques illégales.

Tout d'abord, la banque s'assure de respecter les réglementations en vigueur en matière de lutte contre la fraude et le blanchiment d'argent. Ces politiques et procédures ont été élaborées en étroite collaboration avec les régulateurs et les autorités compétentes pour garantir une conformité totale avec les lois en vigueur.

La banque met en place un ensemble de mécanismes de surveillance avancés pour détecter les activités suspectes. Des logiciels de détection de la fraude sont utilisés pour analyser les transactions bancaires permettant de détecter au plus rapidement possible les activités suspectes. Les employés sont eux aussi formés et sensibilisés aux risques de fraude afin de mieux comprendre les signaux d'alerte et de les signaler immédiatement.

Des enquêtes internes sur les cas de fraudes présumés ou confirmés sont mises en place et les résultats sont transmis aux autorités compétentes, si nécessaire. Cependant, des programmes de coopération sont aussi mis en place pour échanger des informations et des pratiques permettant ainsi de renforcer cette lutte.

3 Projet d'étude

3.1 Présentation du projet

3.1.1 Présentation du contexte et de la problématique

Le projet se déroule dans le cadre d'une application interne nommée FSI.

Il s'agit d'une application qui a pour but de recevoir des données depuis plusieurs sources et dans différents formats, de les traiter et de les redistribuer à d'autres acteurs ou applications sous les formats demandés. Cela fonctionne selon le processus suivant :

- Réception des données depuis différentes sources et dans différents formats
- Couplage des données
- Application de filtres pour répondre au besoin de chaque consommateur
- Partage de ces données à d'autres systèmes sous le format qu'ils demandent

Il y a plusieurs processus qui tournent en parallèle avec ce fonctionnement, ces processus sont appelés des « chaînes », qui sont divisées en plusieurs étapes. Chacune de ces étapes peut contenir un filtre ou un rejet :

- Filtre : filtre fonctionnel précisé avec les acteurs métiers
- Rejet : rejet technique, par exemple une erreur d'un contrôle de format d'un fichier bloquant le traitement

Deux types de chaînes sont distinguables : les chaînes journalières et mensuelles (s'exécutant respectivement tous les jours et tous les mois).

Lors du traitement des données, l'application traite des Deals, qui sont eux-mêmes décomposés en Facilités. C'est sur ces Deals et ces Facilités que sont appliqués les filtres.

Pour simplifier : lors du traitement d'une chaîne, des filtres sont appliqués ou non aux Deals et Facilités de la chaîne.

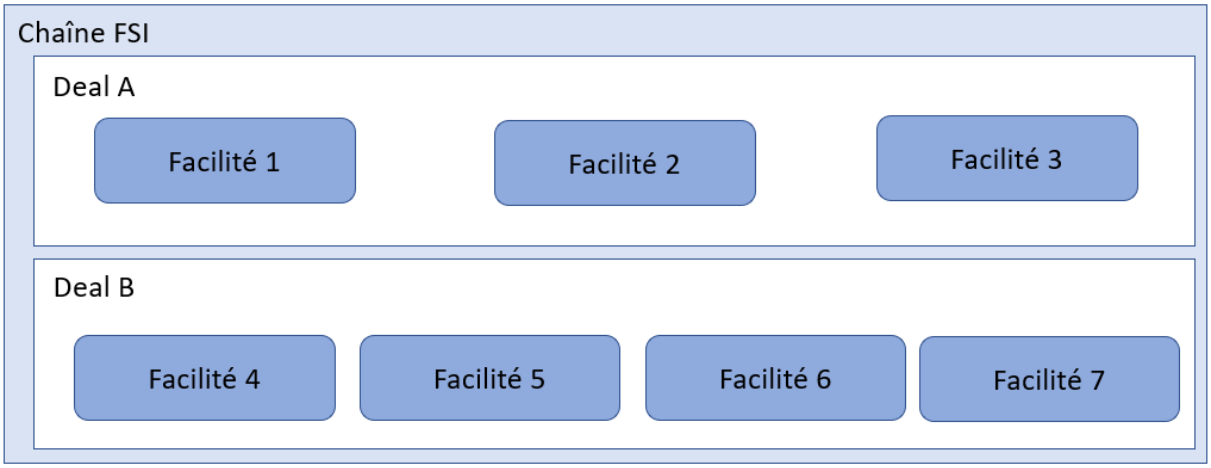


Figure 1. Chaîne FSI, traitant des Deals et des Facilités

L'application redistribue ensuite les données avec les filtres qui ont été appliqués.

La problématique du projet est qu'actuellement aucun outil ne permet de retracer les traitements concernant les filtres réalisés par l'application. Ces informations sont cependant nécessaires, ce qui mène les collaborateurs à devoir contacter le support FSI pour obtenir ces informations.

De même au niveau du support, il n'y a pas d'outils permettant de récupérer ces informations. Actuellement voici ce qu'il se passe :

Un collaborateur a parfois besoin de connaître des informations relatives aux filtres qui ont été appliqués par l'application FSI. Dans ce cas il doit communiquer avec le support FSI pour expliquer son besoin.

Le support FSI doit ensuite :

- Rechercher les informations manuellement au sein de plusieurs fichiers de production, comme les fichiers de logs ou les fichiers traités par l'application
- Faire une synthèse des informations obtenues
- Retourner ces informations au collaborateur

Cela occasionne d'énormes pertes de temps que ce soit pour les membres du support qui passe du temps à répondre à ces demandes, mais aussi pour les collaborateurs qui doivent attendre le temps d'obtenir ces informations.

3.1.2 Objectifs

L'objectif est de mettre en place une interface accessible par les collaborateurs. Cette interface doit leur permettre de récupérer les informations dont ils ont besoin à tout moment, sans avoir à contacter le support de l'application FSI.

Les informations accessibles doivent être :

- Les filtres appliqués aux Deals
- Les filtres appliqués aux Facilités
- Les relations entre un Deal et ses Facilités enfants.

Pour obtenir ces informations, l'utilisateur doit pouvoir les rechercher selon :

- L'identifiant du Deal ou l'identifiant de la Facilité
- La fréquence de la chaîne (journalière ou mensuelle)
- La date de traitement
- La date inventaire
- L'application source

La date inventaire d'une donnée représente le moment où elle a été initialement enregistrée ou collectée. En revanche, la date de traitement indique quand cette donnée a été soumise à des opérations, des analyses ou des transformations ultérieures.

FSI traitant des données d'un énorme nombre d'applications différentes, la portée des applications sources se limitera à celles suscitant le plus de requêtes au niveau du support. Les informations relatives aux traitements qui devront être gérées par l'interface concernent un total de 18 applications sources différentes, chacune faisant l'objet de chaînes journalières et mensuelles. Il y a donc un total de 36 chaînes qui doivent être traitées.

L'utilisateur doit être autonome pour retrouver les informations requises. Il ne doit plus avoir besoin de contacter le support de l'application FSI.

La solution technique mise en place devra garantir une certaine sécurité. Les données ne doivent être accessibles que par les personnes concernées.

L'application devra aussi être extensible. D'autres chaînes provenant d'autres applications sources doivent pouvoir être intégrées par la suite et d'autres cas utilisateurs aussi.

3.2 Méthodologie

Le projet est piloté de manière agile avec des « sprints » d'une durée de trois semaines.

La première phase aura pour but d'établir et de prendre en main les moyens et ressources à utiliser pour mettre en place une solution puis de réaliser un cas d'utilisation précis qui servira de POC (Proof Of Concept).

L'objectif est de pouvoir récupérer les informations des filtres appliqués aux Deals et aux Facilités pour un fichier de log précis et de les visualiser dans une interface graphique.

Ce POC a pour but d'établir la faisabilité du projet et/ou de définir les éventuelles tâches annexes à mettre en place pour permettre la réalisation du projet. Tout sera réalisé sur une seule machine du réseau (en local), dans des conditions permettant de simuler les conditions de production.

S'en suivra une phase de développement pour répondre aux différents besoins, dans laquelle il faudra aussi étendre le périmètre d'applications sur les 36 chaînes définies dans le besoin.

Cette phase de développement se déroulera cette fois-ci dans des conditions similaire à l'environnement de production, qui sera détaillé par la suite.

Suite à cela, se trouvera une période de tests durant laquelle des premiers utilisateurs auront accès à l'application afin de déterminer si elle répond aux besoins et si elle fonctionne correctement.

Si aucun problème n'est détecté, l'application pourra passer en production.

Les prochaines parties expliqueront de manière chronologique les tâches effectuées et le travail réalisé pour la réalisation de ce projet conformément à la méthodologie présentée.

3.3 Présentation du développement et de la solution technique

3.3.1 Présentation des outils utilisés

Pour répondre aux objectifs, il est nécessaire d'utiliser un outil de stockage et de gestion de données et d'avoir une interface graphique facile d'utilisation pour les utilisateurs. Le choix concernant la solution technique s'est porté sur la suite Elasticsearch qui permet de répondre aux besoins.

Elasticsearch est une base de données qui fonctionne en NoSQL. C'est-à-dire qu'elle n'utilise pas le langage SQL pour gérer les données. Elasticsearch utilise donc son propre système de requête qui fonctionne grâce à un système de poids qui sont attribué à chaque mot. Ces poids servent à établir un score qui est déterminé grâce à un modèle mathématiques à base de vecteurs et de fonctions trigonométriques.

De nombreux progiciels font partie de la suite Elasticsearch. Il s'agit de logiciels pouvant être intégrés autour d'Elasticsearch et d'en faciliter les interactions. Il faut donc sélectionner ceux qui serviront à répondre aux besoins exprimés.

Premièrement, il y a le besoin de récupérer les informations concernant le fonctionnement de l'application FSI. L'application émet des fichiers de logs dans lesquels se trouvent les informations requises pour la réalisation du projet, à savoir : les informations concernant les filtres appliqués aux Deals et aux Facilités. Pour récupérer ces informations, l'utilisation d'un autre logiciel de la suite Elasticsearch répond à ce besoin : Filebeat.

Il s'agit d'un logiciel permettant de lire des fichiers. Il permettra de lire les fichiers de logs générés par l'application FSI lors de son fonctionnement. Filebeat transmettra ensuite les données qu'il a récolté dans les fichiers. Pour cela il utilise le protocole TCP.

Cependant Filebeat transmet les données des fichiers de log sans les traiter. Les données seront dans un état brut. L'objectif est de traiter ces données et de les indexer. C'est-à-dire organiser les données sous forme d'une structure de données pour pouvoir les gérer efficacement.

Il existe un logiciel dans la suite Elasticsearch qui permet de faire cela : Logstash. Ce logiciel peut être configurer pour recevoir des données en entrée, pour les traiter et les transmettre à

Elasticsearch (avec le protocole TCP). Logstash permet aussi de récupérer des données depuis une base de données. Il pourra aussi être utilisé dans ce sens si besoin.

Il y a donc Filebeat qui récupère les données des fichiers de log, Logstash qui les indexe et Elasticsearch qui joue le rôle de la base de données.

À la suite de cela, il reste le besoin de visualiser les données de manière simple et efficace. Un dernier logiciel répond à ce besoin : Kibana. Il s'agit encore d'un logiciel de la suite Elasticsearch. Kibana est une application Web (logiciel qui s'exécute dans un navigateur Web) qui permet de communiquer de manière simple avec Elasticsearch, mais aussi de créer des visuels de ces données (comme des tableaux de bord à titre d'exemple).

Pour résumer, les différents logiciels seront organisés et communiqueront de la manière suivante :

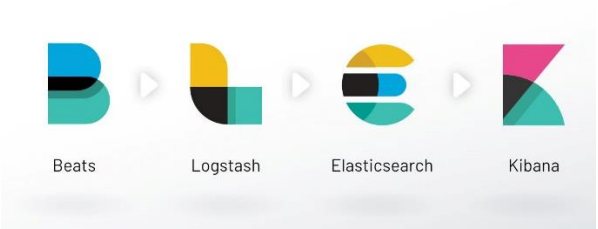


Figure 2. Organisation des logiciels utilisés

Ces 4 logiciels fonctionnent avec des fichiers de configuration en YAML.

Concernant le développement, Git a été utilisé en tant que gestionnaire de version.

Différents outils annexes seront utilisés pour faciliter le développement tel que SQLDeveloppeur, Notepad++, Putty, WinScp, Cyberduck. Pour la communication et la documentation, les outils de la suite Microsoft seront utilisés.

3.3.2 POC (Proof of Concept)

3.3.2.1 Objectif du POC

En premier lieu, il est préférable d'établir la faisabilité du projet et de déterminer si les outils choisis permettent vraiment d'atteindre les objectifs définis.

L'objectif de ce POC est donc de répondre aux besoins fonctionnels de la partie **3.1.2** pour un cas précis. En partant d'un fichier de log généré par l'application FSI, le but est de retrouver les informations de ce fichier dans une interface graphique. L'utilisateur doit pouvoir retrouver les filtres appliqués à un Deal ou une Facilité et doit pouvoir rechercher par rapport à :

- L'identifiant du Deal ou l'identifiant de la Facilité
- La fréquence de la chaîne (journalière ou mensuelle)
- La date de traitement
- La date inventaire
- L'application source

L'utilisateur devra être en mesure de consulter les filtres appliqués mais aussi de pouvoir faire le lien entre un Deal et ses Facilités enfant.

En supposant par exemple qu'un Deal A est filtré pour plusieurs raisons pendant l'exécution d'une chaîne.

L'utilisateur doit être en mesure de déterminer quels filtres ont été appliqués à ce Deal, mais aussi quels filtres ont été appliqués aux Facilités enfants de ce Deal (pour rappel un Deal peut contenir plusieurs Facilités)

3.3.2.2 Architecture réseau

FSI est une application critique au sein de la banque. Il est impératif de ne pas altérer ou ralentir son fonctionnement. C'est pourquoi l'objectif est de modifier au minimum le fonctionnement du serveur sur lequel tourne l'application FSI. Pour cela, l'utilisation d'un deuxième serveur de production, détaché du serveur de production FSI, est requis pour ce projet.

Cependant il faut tout de même récupérer les logs du serveur de production. Il est donc nécessaire d'établir une communication entre les deux serveurs dans le but de récupérer ces fichiers de log. C'est pourquoi le logiciel Filebeat sera sur le serveur de production. L'architecture utilisé sera la suivante :

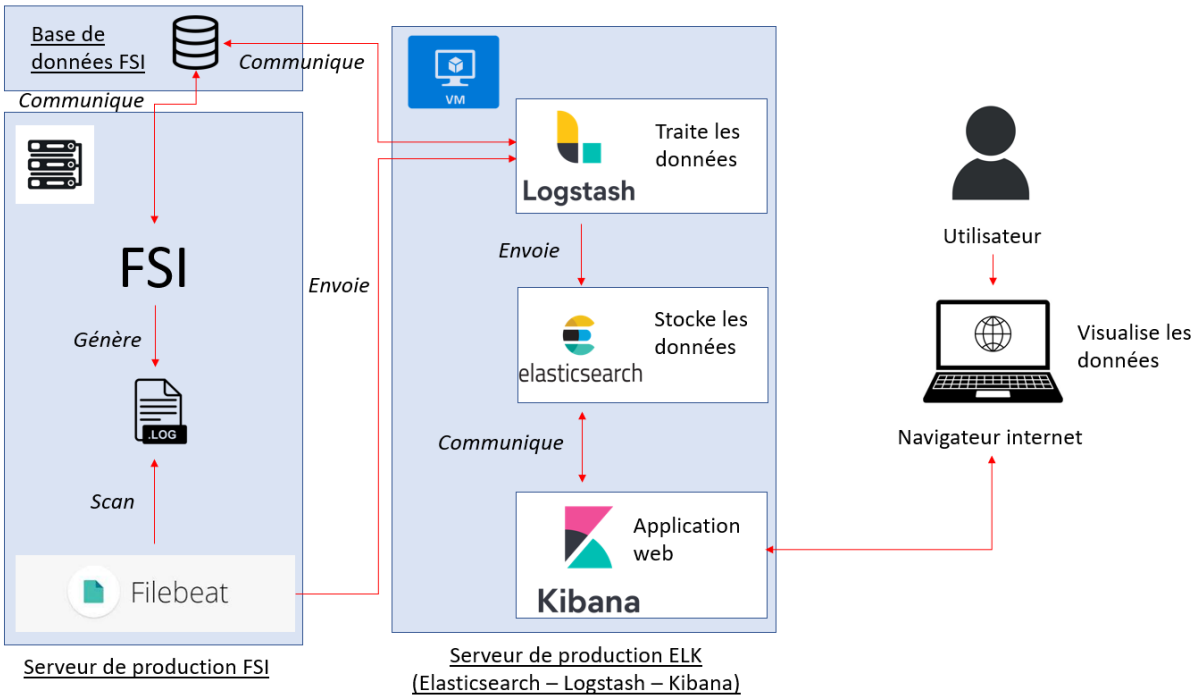


Figure 3. Schéma de l'architecture réseau de la solution technique

Concernant le serveur de production ELK, une machine virtuelle sera mise en place et fera office de serveur.

L'utilisateur accédera aux données via un navigateur internet grâce à une URL (Uniform Resource Locator). Il aura ainsi accès à une interface utilisateur pour visualiser les données et pour rechercher des données précises selon son besoin.

L'ensemble de cette infrastructure est compris dans l'intranet de l'entreprise (réseau privé interne) et n'est donc pas accessible depuis l'extérieur du réseau, à moins d'avoir recours à un VPN (Virtual Private Network).

Cependant cette architecture ne sera pas utilisée immédiatement pour le développement. En premier lieu tout sera fait en local sur une machine détaché du serveur de production suivant le schéma ci-dessous :

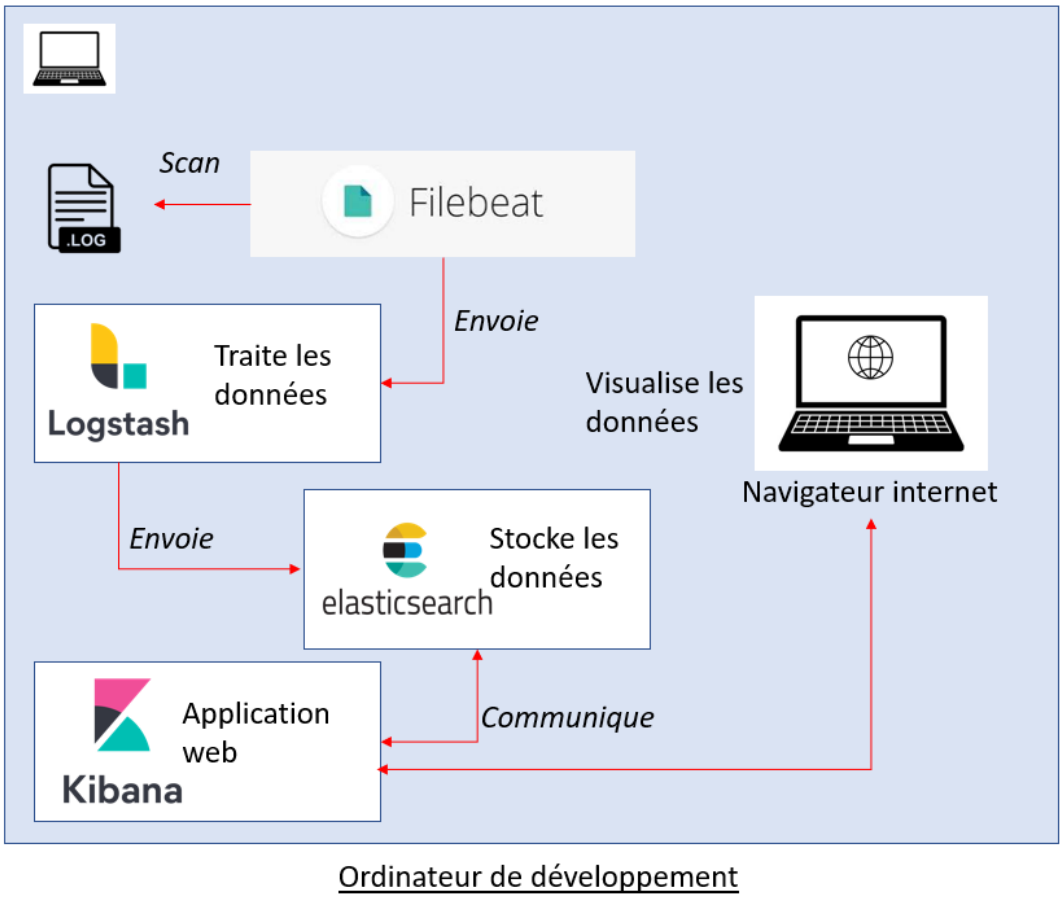


Figure 4. Schéma de l'environnement de développement du POC

L'élaboration du premier POC se fera conformément à cette architecture. Tout sera ensuite transposé selon l'architecture définitive.

3.3.2.3 Récupération des fichiers de logs

Comme expliqué précédemment, Filebeat récupère les fichiers de logs sur le serveur de production FSI et les envoie à Logstash.

Le fonctionnement se fait de la manière suivante :

Filebeat peut scanner un ou plusieurs répertoires à la recherche de fichier respectant un pattern défini au préalable. Filebeat va ainsi scanner tous les fichiers respectant ce pattern se trouvant dans les répertoires indiqués.

Quand Filebeat détecte un fichier, il scannera les informations contenues à l'intérieur de celui-ci et gardera en mémoire ce qui a été scanné. Il envoie ensuite les données vers la sortie qui a été définie.

Si un fichier a déjà été scanné et qu'il est modifié par la suite, Filebeat le détectera et récupérera seulement les nouvelles informations ou les nouvelles lignes du fichier avant de les envoyer.

L'application FSI génère un fichier de log à chaque fois qu'une chaîne est traitée. Ces fichiers sont toujours générés dans le même répertoire de stockage. Cependant, pour débuter le but est de reproduire un cas d'utilisation précis qui est de pouvoir retrouver les informations concernant les filtres des Deals et des Facilités, informations correspondantes à l'exécution d'une chaîne à une date précise.

Un fichier de log est donc récupéré manuellement sur le serveur de production pour ce POC. Il sera placé dans un répertoire sur la machine que servira pour le développement. Le répertoire

dans lequel se trouve le fichier de log sera ciblé avec Filebeat pour récupérer les informations comprises dans le fichier.

3.3.2.4 Traitement et indexation des logs

3.3.2.4.1 Récupération des données

C'est le logiciel Logstash qui va permettre d'indexer les logs.

Le but est d'envoyer les données de Filebeat à Logstash pour pouvoir les traiter. Pour cela, Filebeat permet d'envoyer des données en utilisant le protocole TCP en définissant un port logiciel en sortie, port vers lequel les données seront envoyées après. Il s'agit d'une « porte » donnant accès au système d'exploitation d'une machine (Linux, Windows…). Ces ports permettent de communiquer avec des interlocuteurs, en l'occurrence des logiciels, et sont identifiables par des numéros.

Un port est donc défini pour faire la connexion entre les deux logiciels.

3.3.2.4.2 Pipeline Logstash et indexation des données

Pour traiter les données avec Logstash, il faut concevoir ce qui s'appelle un « pipeline ».

Ce pipeline est composé de trois parties :

- Une entrée, qui permet de récupérer des données
- Un filtre, permettant de traiter et indexer les données
- Une sortie, pour envoyer les données après application des filtres.

Chaque partie peut être présente plusieurs fois. C'est-à-dire qu'il peut y avoir plusieurs entrées provenant de plusieurs sources différentes, plusieurs filtres et plusieurs sorties.

Il est aussi possible de créer plusieurs pipelines différents en parallèle.

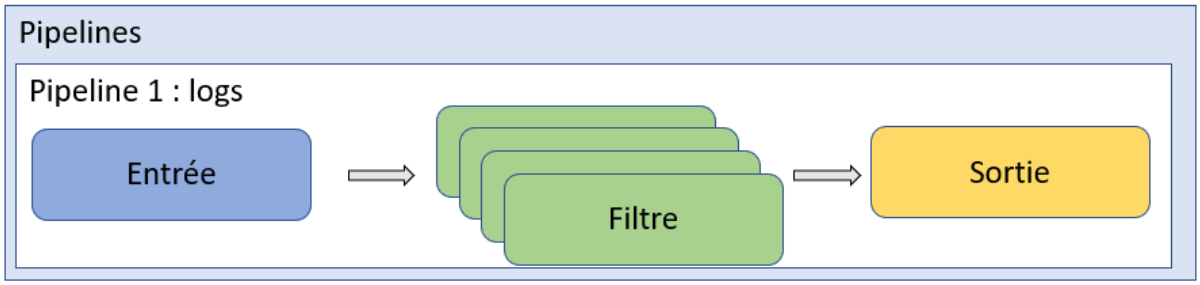


Figure 5. Schéma des pipelines Logstash

Dans le cas présent, le pipeline mit en place prend en entrée les données envoyées par Filebeat grâce au port logiciel qui a été défini au préalable.

Il faut ensuite indexer les informations qu'il y a dans les logs. Selon le besoin qui a été définit, l'utilisateur doit pouvoir rechercher un filtre selon plusieurs critères :

- L'identifiant du Deal ou l'identifiant de la Facilité
- La fréquence de la chaîne (journalière ou mensuelle)
- La date de traitement
- La date inventaire
- L'application source

L'objectif est de récupérer toutes ces informations dans les logs. Pour cela il faut d'abord identifier l'endroit où il est possible de les récupérer.

En étudiant le fichier de log mis à disposition et en parcourant le code de l'application FSI au niveau de l'écriture des logs, il est possible de déterminer où sont écrites les informations dans le fichier et selon quel format.

Pour les Facilités, voici le format des logs :

"YYYY-MM-DD HH:mm:SS,UUU [Thread-$thread_name] [INFO]- chemin.vers.le.fichier.du.code.FSI- SystemId: Splitter.xslt Line: 767 - Facility 1 is filtered for $filter_reason"

Par défaut, Filebeat indexe les données dans un seul champ : « message ». Pour chaque ligne du fichier de log, il existe un champ « message », qui contient tout le texte écrit dans la ligne. Logstash permet de créer un indexage personnalisé et précis.

En utilisant des expressions régulières (regex), il est possible de définir des patterns de données. A titre d'exemple le pattern suivant : « (?:[0-5][0-9]) » correspond à deux chiffres, le premier étant compris entre 0 et 5, et le deuxième entre 0 et 9. Il récupère ainsi un nombre compris entre 00 et 59. Ce pattern permet donc d'avoir les informations correspondantes aux minutes d'une date. Le résultat peut ensuite être stocké dans une variable ou non.

En identifiant les patterns d'écriture des logs, il est possible de décomposer complétement une ligne d'information et de l'indexer.

A titre d'exemple, le pattern regex suivant correspond aux dates telles qu'elles sont écrites dans les logs (« YYYY-MM-DD HH:mm:SS,UUU ») :

« (?>\d\d){1,2}-(?:0?[1-9]|1[0-2])-(?:(?:0[1-9])|(?:[12][0-9])|(?:3[01])|[1-9]) (?!<[0-9])(?:2[0123]|[01]?[0-9]):(?:[0-5][0-9])(?::(?:(?:[0-5]?[0-9]|60)(?:[:.,][0-9]+)?))(?![0-9]) ».

Ainsi à partir du format des lignes de logs et en établissant des patterns en regex qui correspondent à ces formats des données, la ligne de log est indexée grâce à Logstash de la manière suivante :

Tableau 1. Exemple d'indexage d'un log

Nom du champ	**Contenu**
processing_date	YYYY-MM-DD HH:mm:SS,UUU
thread	thread_name
status	INFO
classe	chemin.vers.le.fichier.du.code.FSI
log_message	SystemId: Splitter.xslt Line: 767 - Facility A is filtered for $filter_reason
facility	1
filter	filter_reason

En se référant au besoin exprimé, le constat est qu'il manque encore des informations comme la date d'inventaire, la fréquence de la chaîne et le nom de l'application source.

De la même manière que précédemment, en déterminant l'endroit où se trouve ces informations dans les logs, en identifiant le format des données et en appliquant des patterns, il est possible de récupérer ces informations.

Voici une ligne de log avec des données requises pour répondre au besoin :

« YYYY-MM-DD HH:mm:SS,UUU [Thread-$thread_name] [INFO]- chemin.vers.le.fichier.du.code.FSI- Launching Simple transformation [name=$transformation_name,invDate=20211028,id=15] with Id Event [98]"

Ainsi, à partir de la ligne de log précédente, il est possible grâce au regex de déterminer les champs suivants :

Tableau 2. Suite de l'indexage des logs

Nom du champ	Contenu
Name	transformation_name
inventory_date	2021-10-28
id_transformation	15
id_event	98

De la même manière, il est possible de récupérer toutes les autres informations requises et de les indexer.

Il faut cependant pouvoir relier ces données. En effet, ces données proviennent de lignes de log différentes et par défaut elles vont être stockées dans des objets différents. Il faut indiquer à Logstash de regrouper ces informations sous le même objet car elles sont liées.

Pour cela il est possible d'injecter du code en Ruby dans Logstash. Il s'agit d'un langage de programmation orienté objet. Le Ruby va permettre de stocker des données provenant d''une ligne de log et de transmettre ces données à d'autres lignes. De cette manière, il est possible fusionner les différents tableaux et de les combiner en un seul.

3.3.2.5 Stockage des données

A ce stade, les données requises pour répondre au besoin sont récupérées dans les fichiers de logs et indexées. Il faut maintenant communiquer les données indexées à Elasticsearch pour les stocker. Pour rappel, Elasticsearch joue le rôle d'une base de données, à la différence que cela fonctionne sans le langage SQL.

Elasticsearch comprend un serveur HTTP pour communiquer. Ce serveur est relié à un port logiciel. Il est possible d'interagir avec Elasticsearch via ce port. C'est donc ici que les données provenant de Logstash vont être envoyées.

En reprenant le pipeline Logstash, il reste la sortie à définir. Dans cette sortie il faut définir :

- L'URL vers laquelle envoyer les données
- Le nom de l'index dans lequel stocker les données (l'équivalent du nom d'une table dans une base de données SQL)

Concernant l'URL, il faut renseigner le protocole utilisé, la machine ciblée ainsi que le port logiciel. Il faut donc utiliser le protocole HTTP en ciblant la machine sur laquelle se trouve l'instance d'Elasticsearch.

Les données seront ensuite stockées sous l'index portant le nom définit par Logstash. Si l'index n'existe pas, il sera automatiquement créé.

Dorénavant, les informations du fichier de log sont stockées dans la base de données conformément au besoin du POC.

Pour communiquer avec Elasticsearch, il faut passer par des requêtes HTTP.

Voici ce qu'il se passe en se connectant à l'instance d'Elasticsearch avec un navigateur internet :

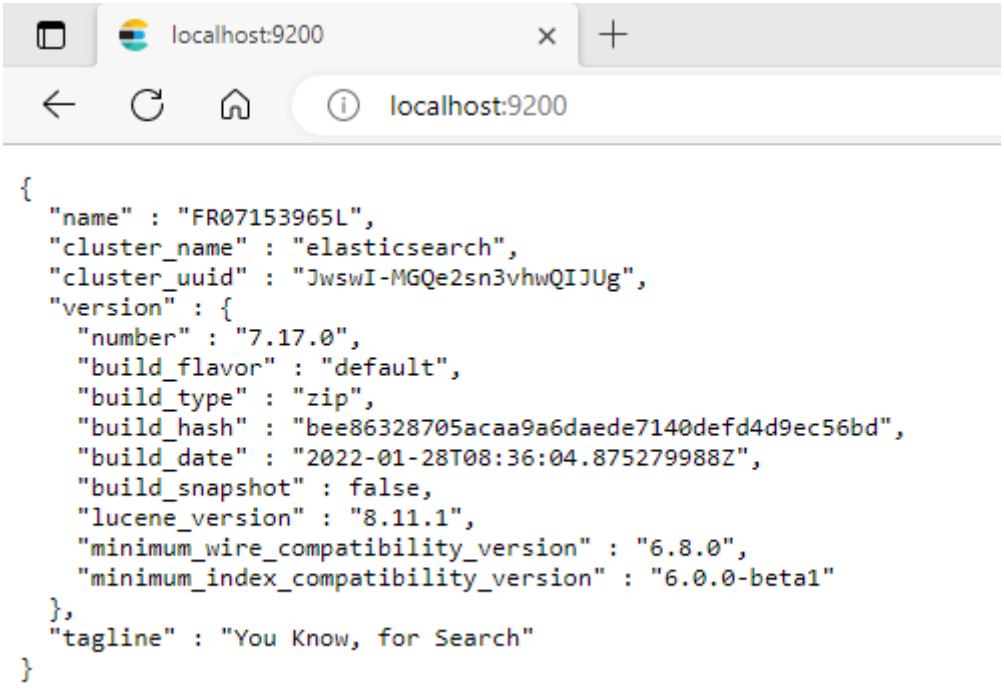


Figure 6. Cluster Elasticsearch

Elasticsearch retourne des informations concernant l'instance comme le nom du cluster, la version utilisée.

Elasticsearch dispose d'une API permettant de faire ces requêtes de manière simple et efficace. Il est par exemple possible d'avoir le nom de tous les index de stockage de cette manière :

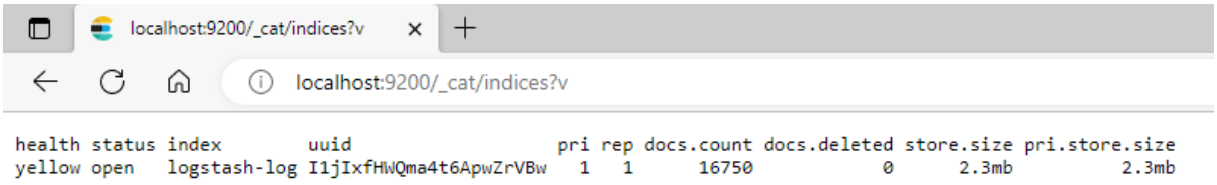


Figure 7. Index Elasticsearch

Il est possible d'observer le nom ou encore l'index dans lequel les données des logs ont été envoyées depuis Logstash.

3.3.2.6 Visualisation des données

3.3.2.6.1 Présentation de Kibana

Désormais, il reste le besoin de pouvoir rechercher les données concernant le filtre appliqué à la Facilité 1. Il faut aussi mettre à disposition une interface facile d'utilisation pour retrouver ces informations.

Elasticsearch dispose d'une API permettant d'accéder aux données, cependant l'objectif est d'avoir une interface simple d'utilisation.

C'est ici qu'intervient le logiciel Kibana. Il va permettre de créer facilement des tableaux de bord qui fera des requêtes automatiques à l'API d'Elasticsearch.

Le logiciel Kibana, à l'instar d'Elasticsearch, comprend un serveur HTTP mais a l'avantage de proposer une interface.

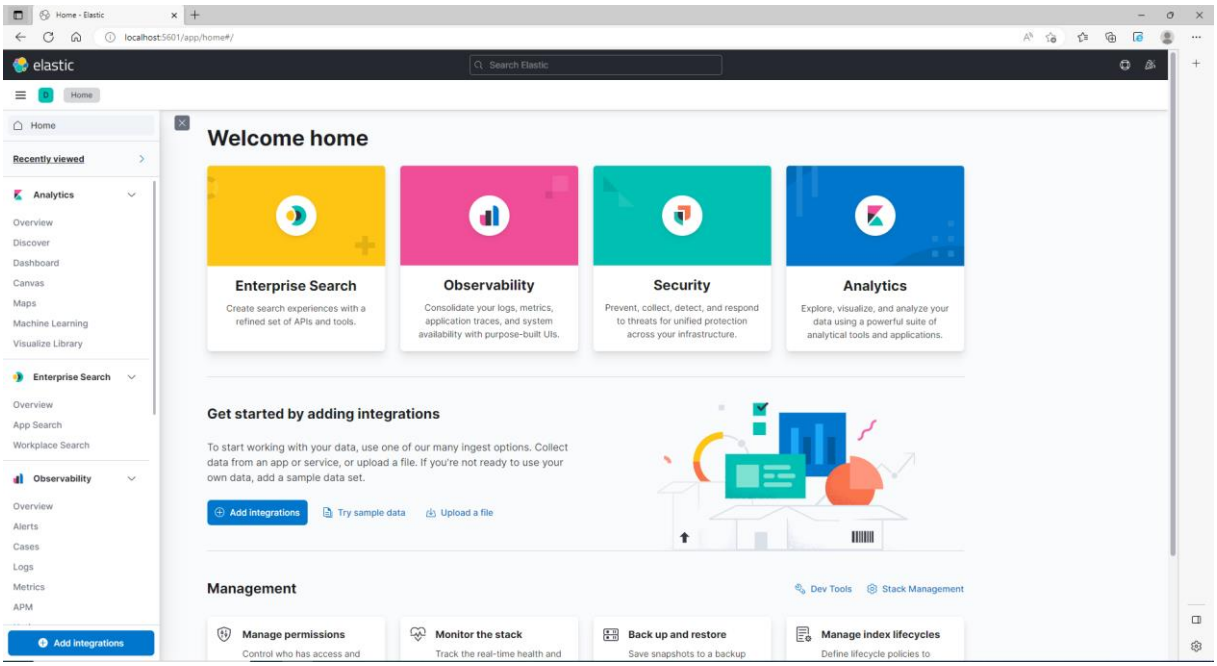


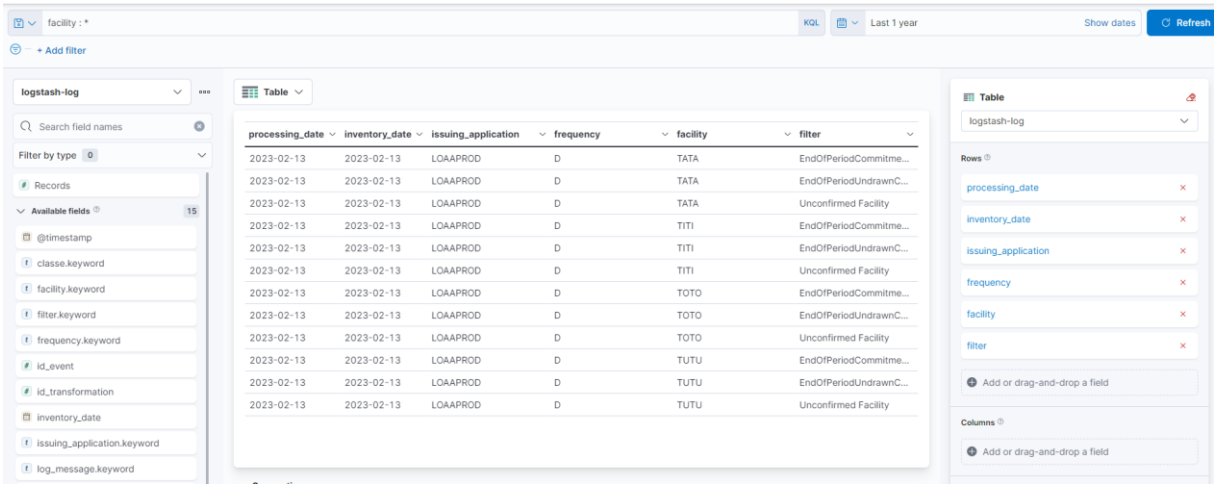
Figure 8. Interface Kibana

Cette interface donne accès à de nombreux outils de visualisation mais aussi de développement.

3.3.2.6.2 Création d'un tableau de bord

Parmi les nombreux outils de Kibana, il y a la création de tableaux de bord qui peuvent implémenter tout type de tableau ou de graphique avec un requêtage automatique vers l'API Elasticsearch.

Grâce à l'outil de création et pour répondre au besoin, un tableau de bord contenant une table est créé, table regroupant les différents champs de recherche exprimés dans le besoin pour retrouver une Facilité et ses filtres.



processing_date	inventory_date	issuing_application	frequency	facility	filter
2023-02-13	2023-02-13	LOAAPROD	D	TATA	EndOfPeriodCommitme...
2023-02-13	2023-02-13	LOAAPROD	D	TATA	EndOfPeriodUndrawnC...
2023-02-13	2023-02-13	LOAAPROD	D	TATA	Unconfirmed Facility
2023-02-13	2023-02-13	LOAAPROD	D	TITI	EndOfPeriodCommitme...
2023-02-13	2023-02-13	LOAAPROD	D	TITI	EndOfPeriodUndrawnC...
2023-02-13	2023-02-13	LOAAPROD	D	TITI	Unconfirmed Facility
2023-02-13	2023-02-13	LOAAPROD	D	TOTO	EndOfPeriodCommitme...
2023-02-13	2023-02-13	LOAAPROD	D	TOTO	EndOfPeriodUndrawnC...
2023-02-13	2023-02-13	LOAAPROD	D	TOTO	Unconfirmed Facility
2023-02-13	2023-02-13	LOAAPROD	D	TUTU	EndOfPeriodCommitme...
2023-02-13	2023-02-13	LOAAPROD	D	TUTU	EndOfPeriodUndrawnC...
2023-02-13	2023-02-13	LOAAPROD	D	TUTU	Unconfirmed Facility

Figure 9. Outil de création de tableaux de bord

De même au niveau des Deals, une table est réalisée de la même manière. Les différents composants créés sont regroupés et organisés dans un tableau de bord présentable de la manière suivante :

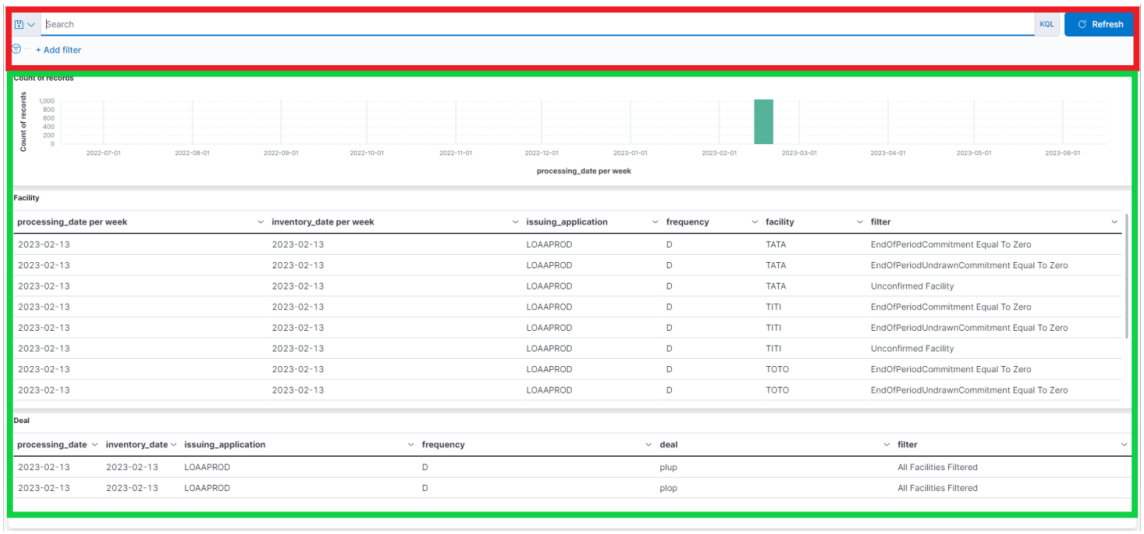


Figure 10. Tableau de bord pour Facilités et Deals

Le tableau est découpé en plusieurs parties, en haut se trouve les outils de recherche (encadré en rouge sur la Figure 9) en dessous les différents composants créés (encadré en vert sur la Figure 9).

Si l'utilisateur souhaite retrouver les informations concernant une Facilité, il lui suffit d'écrire son nom dans la barre de recherche et le tableau des Facilités se filtrera automatiquement pour n'afficher que les résultats correspondants à sa recherche.

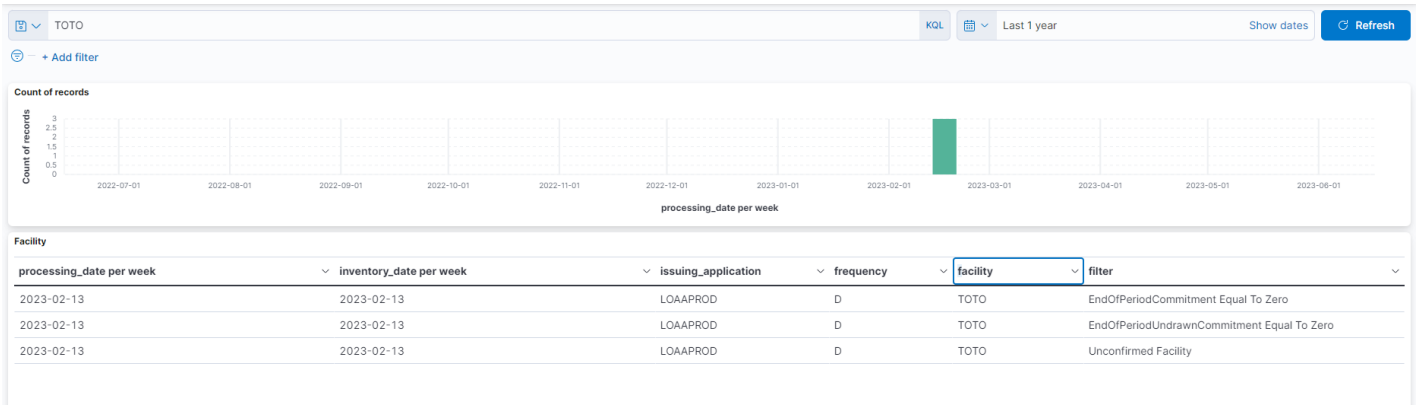


Figure 11. Recherche d'information sur un tableau de bord

3.3.2.7 Bilan du POC

En comparant les objectifs initialement prédits et ce qui a été réalisé, le constat est le suivant :

- Il est possible de récupérer les données de traitement générées par l'application FSI et de mettre en place une interface pour y accéder.
- L'utilisateur peut rechercher un Deal ou une Facilité et obtenir toutes les informations demandées.
- Il n'est actuellement pas possible à partir des informations présentes dans les fichiers de logs de déterminer le lien entre un Deal et ses Facilités enfants avec les informations actuelles.

Voici les objectifs qui en découlent pour la suite du projet :

- Trouver un moyen de faire le lien entre un Deal et ses Facilités enfants
- Mettre en place une architecture de développement similaire à celle de production

- Etendre les cas d'utilisations à toutes les chaînes définies dans les objectifs (le POC s'était concentré sur un seul fichier de log d'une seule chaîne)
- Sécuriser les données

3.3.3 Développement du projet

3.3.3.1 Lien entre Deal et Facilités enfants

Après analyse des fichiers de logs, il a été évalué qu'il n'est pas possible d'établir un lien entre un Deal et ses Facilités enfants. Pour avoir ces informations, le support procède en regardant l'arborescence d'un fichier spécifique.

Il a été déterminé que la meilleure solution est de modifier le format des fichiers de logs pour rajouter l'information qu'il manque. Ainsi, il suffira de modifier la manière d'indexer les logs grâce à Logstash.

Les lignes de logs concernant les filtres et les Facilités suivent désormais le format suivant :

"YYYY-MM-DD HH:mm:SS,UUU [Thread-$thread_name] [INFO]- chemin.vers.le.fichier.du.code.FSI- SystemId: Splitter.xslt Line: 767 - Facility A is filtered for $filter_reason – From Deal [1]"

Ainsi chaque Facilité pourra être reliée avec son Deal parent.

La génération des logs de l'application FSI a donc été modifiée pour ajouter ces informations. Il a pour cela fallu déterminer l'ensemble des endroits dans le code de l'application dans lesquels les logs sont écrits. L'ensemble de ces modifications doit être effectif sur les 36 chaînes du périmètre qui a été déterminé dans les objectifs du projet.

3.3.3.2 Architecture

Jusqu'ici, tout se faisait en local sur une machine pour le développement. Il faut maintenant se placer dans un environnement similaire à l'environnement de production explicité sur la ***figure 3*** dans la partie ***3.3.2.2 Architecture réseau***.

Le développement se fera donc selon en environnement UAT (test d'acceptation par l'utilisateur) comme sur le schéma suivant :

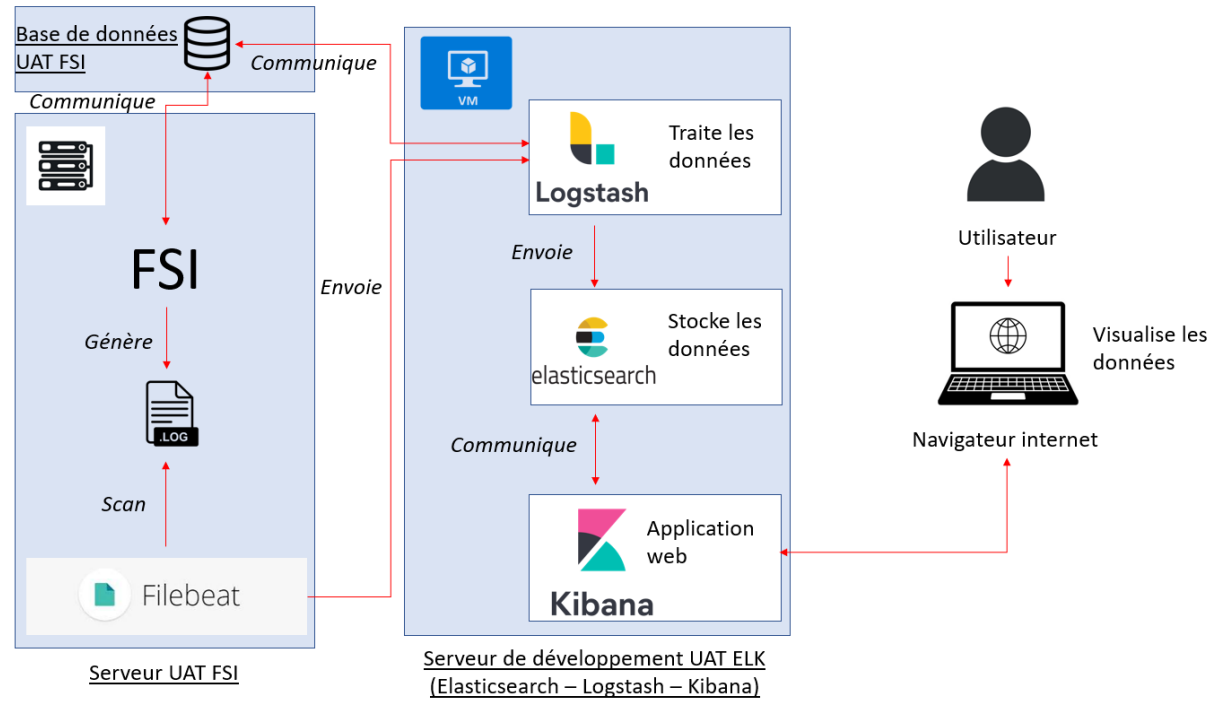


Figure 12. Schéma d'architecture réseau UAT

Cet environnement permettra de reproduire à l'identique l'architecture de production.

3.3.3.3 Extension du périmètre d'utilisation

3.3.3.3.1 Récupération des logs avec Filebeat

Le POC a été réalisé avec un fichier de log d'une chaîne précise. Cependant le besoin du projet ne s'étend ni sur cette seule chaîne ni sur toutes les chaînes de l'application FSI mais sur un total de 36 chaînes.

Il faut donc effectuer une sélection pour récupérer seulement les fichiers de log concernant ces chaînes à l'aide du logiciel Filebeat.

Lors de la génération d'un fichier de log par l'application FSI, un nom est attribué au fichier en fonction du nom de la chaîne et de la date inventaire correspondante. Afin de récupérer seulement les fichiers souhaités, il faut donc appliquer des filtres pour ne lire que les fichiers qui correspondent aux objectifs.

Pour cela il est possible de définir des patterns dans Filebeat au niveau des noms des fichiers.

Après analyse des différents noms possibles pour les fichiers de logs, des patterns sont donc définies pour correspondre aux noms des fichiers de logs correspondant uniquement aux 36 chaînes qui doivent être traitées.

Filebeat sera paramétré pour suivre l'algorigramme suivant :

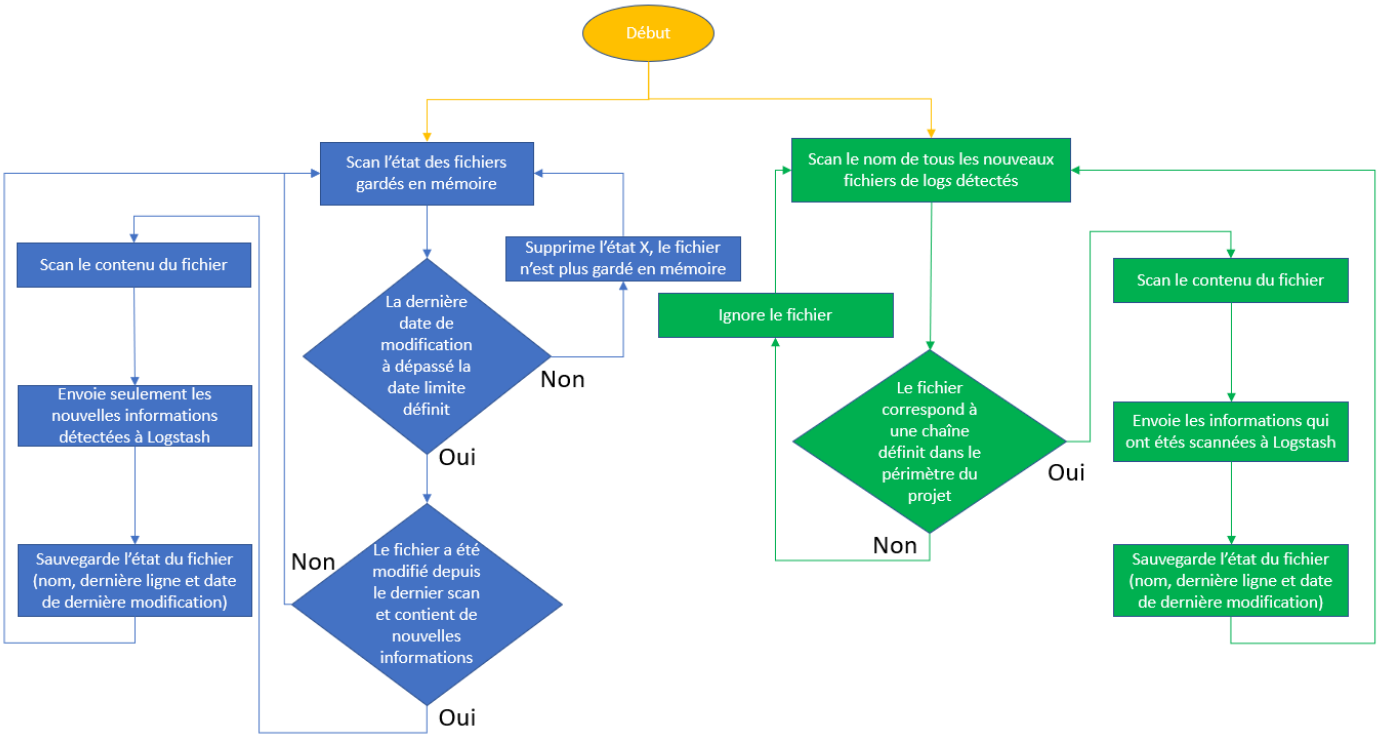


Figure 13. Algorigramme de Filebeat

Son fonctionnement contient deux processus en parallèle, le premier (en vert) correspond aux scans des nouveaux fichiers de logs détectés et le second (en bleu) concerne les fichiers qui ont déjà été scannés.

Grâce à cette configuration, les fichiers de logs sont récupérés conformément aux objectifs.

3.3.3.3.2 Indexation des logs avec Logstash

Pour donner suite à la modification des fichiers de logs dû au manque d'informations, il faut mettre à jour le traitement des logs dans Logstash pour prendre en compte cette amélioration.

Afin d'améliorer l'expérience utilisateur, il a aussi été jugé utile de rajouter des informations provenant de la base de données FSI. En effet, l'application FSI y sauvegarde des données qui ne sont pas contenues dans les logs.

Un deuxième pipeline est alors mis place sur Logstash pour se connecter à la base de données et en extraire les informations souhaitées. Pour cela, il est possible de définir des requêtes SQL à effectuer. Les données récupérées seront ensuite traitées selon le besoin puis envoyées vers Elasticsearch.

Logstash fonctionne désormais avec deux pipelines en parallèle conformément au schéma suivant :

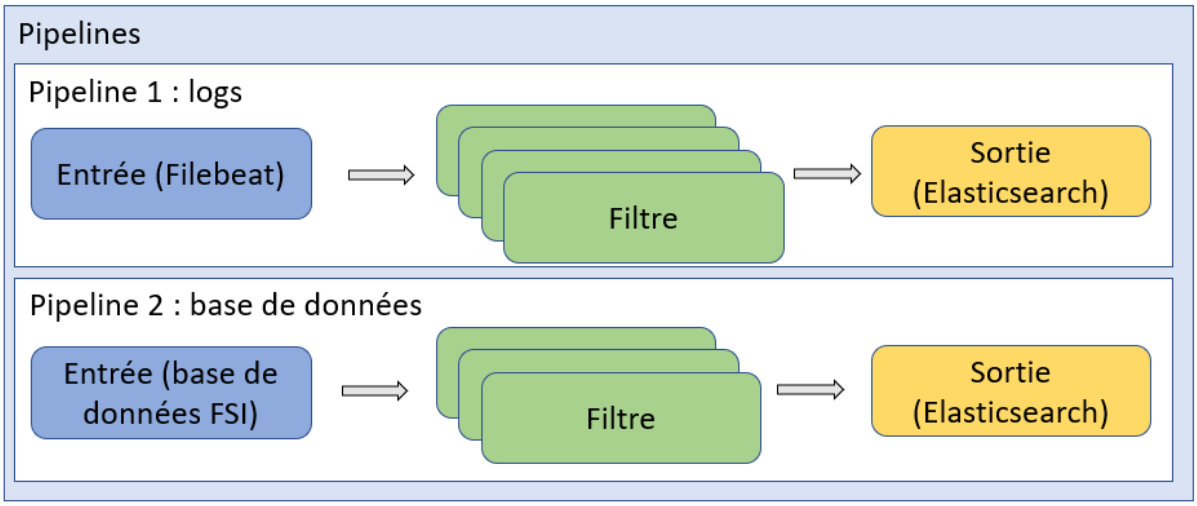


Figure 14. Les pipelines de Logstash

3.3.3.3.3 Stockage des données dans Elasticsearch

Les données sont stockées sont forme d'index (l'équivalent d'une table en SQL). Dans le cas présent, Elasticsearch recevra des données provenant de deux pipelines et les stockera dans deux index différents :

- Logs
- Database

Cependant il faut penser plus à long terme. En effet une instance Elasticsearch possède une quantité de stockage maximum recommandée qui est de 30 Giga octets. Il y aura donc un moment où il faudra procéder à un nettoyage des données (automatique ou manuel) lorsque la limite des 30 Go sera proche.

Afin de faciliter ce nettoyage, il serait pratique de pouvoir supprimer ou non les données en fonction de leurs dates. Actuellement, avec les index définit, il est difficile d'effectuer ce nettoyage facilement et rapidement. C'est faisable mais coûteux en temps et en ressources.

C'est pourquoi il a été jugé utile de mettre en place une indexation dynamique en fonction de la date. Cela permet de supprimer directement les anciens index de manière très rapide et efficace.

Les logs seront donc désormais indexés en fonction de la date de traitement. Le stockage se fera donc comme ceci :

- Database
- Logs-2023-05
- Logs-2023-06
- Logs-2023-07

De cette manière, en regroupant les logs par mois, il sera aisé de les traiter. En procédant par exemple, à un nettoyage automatique tous les mois, qui ira supprimer les index datant de plus de 6 mois. Les informations concernant la base de données n'ont pas été indexées par date car en étudiant la quantité de stockage des données récupérées dans la base de données, la quantité a été jugé assez faible pour être négligé.

Maintenant que les index sont dynamiques, il est intéressant de se plonger dans leurs fonctionnements pour le comprendre et pouvoir l'optimiser.

Par défaut, les données sont stockées de la manière suivante :

Figure 15. Modèle de stockage de données par défaut d'Elasticsearch

Ainsi lorsqu'on souhaite effectuer une recherche de données sur cet index, la recherche s'appliquera sur toutes les données. Cela peut poser des problèmes lorsqu'il y a un grand nombre de données, les performances seront atténuées et les résultats mettront plus de temps à sortir.

Néanmoins il existe une solution pour éviter ces soucis. Les données au sein d'un index peuvent être réparties dans ce qui s'appelle des « shards ». A titre d'exemple, en répartissant les données de la figure 15 en 4 shards, voici comment les données seront stockées :

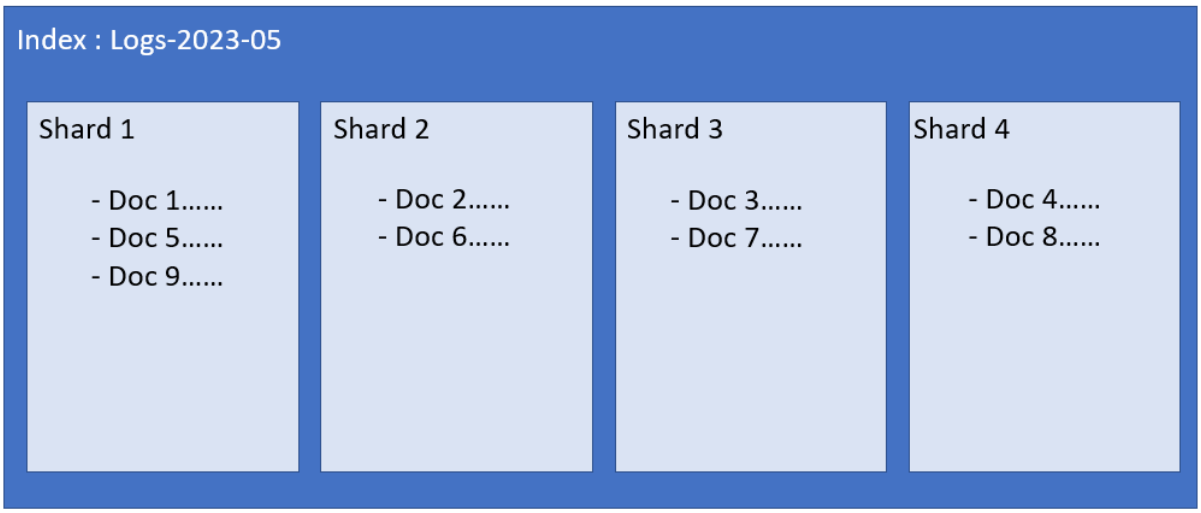


Figure 16. Index Elasticsearch avec des shards

Les données sont ainsi réparties de manière égale dans différents blocs. Dorénavant, lorsque qu'une recherche sera faite sur l'index, il y aura 4 recherches qui seront effectuées en parallèle, 1 recherche par shard, ce qui améliore les performances.

3.3.3.3.4 Affichage des données dans Kibana

Concernant l'affichage des données, les informations supplémentaires récupérées sont intégrées. Par soucis de confidentialité, les données du tableau de bord ont été masquées par des carrés bleus sur la figure ci-dessous.

Figure 17. Tableau de bord final sur Kibana

Il est désormais possible de faire le lien entre un Deal et sa Facilité enfant grâce à la table à gauche.

Une autre table a été ajoutée tout en bas permettant de visualiser les informations provenant de la base de données.

3.3.3.4 Gestion des comptes et des droits utilisateurs

3.3.3.4.1 Comptes techniques

La sécurité prend une place centrale dans ce projet. En effet, en traitant et en mettant à disposition des données fonctionnelles, il faut respecter des normes de sécurité.

C'est pourquoi, il faut mettre en place un certain nombre de paramètres et de fonctionnalité pour minimiser les risques concernant la confidentialité, l'intégrité la disponibilité et l'intégrité des données.

La première sécurité à mettre en place concerne les utilisateurs et les droits afin d'empêcher n'importe qui d'accéder aux informations mais aussi d'ajouter ou d'altérer les données. Premièrement sont mis en place des profils utilisateurs dit « techniques ». Il s'agit des comptes techniques à mettre en place obligatoirement dans toute infrastructure Elasticsearch. Parmi ces utilisateurs techniques, il y a par exemple « elastic », qui est le super-utilisateur possédant tous les droits, ou encore « logstash_system » qui a les droits pour écrire des données dans Elasticsearch. Pour chacun de ces utilisateurs techniques, il faut attribuer un mot de passe.

Ainsi, à partir de ce moment, toutes les requêtes effectuées vers Elasticsearch vont demander un utilisateur et un mot de passe et la requête ne sera validée que si l'utilisateur est valide et possède les bons droits. Il faut donc apporter des modifications aux logiciels communiquant à Elasticsearch pour intégrer ces mots de passe.

Les deux logiciels en communication sont Logstash et Kibana. Concernant Logstash, les pipelines sont mis à jour pour intégrer l'utilisateur « logstash_system » ainsi que le mot de passe définit

avec. De même avec Kibana, les fichiers de configurations sont mis à jour pour utiliser le compte utilisateur « kibana » lui octroyant tous les droits dont le logiciel a besoin.

3.3.3.4.2 Comptes utilisateurs

La gestion des utilisateurs se décompose en 3 parties :

- Les espaces
- Les rôles
- Les utilisateurs

Un espace est un environnement dans lequel on peut insérer des tableaux de bord, des index et plus encore. Il est par exemple possible de créer des espaces différents possédant des tableaux de bord différents. Un utilisateur n'ayant accès qu'à un seul espace ne pourra pas consulter les tableaux de bord présents sur le deuxième espace et inversement.

Il faut ensuite créer des rôles. Les rôles correspondent à des profils de droits. A titre d'exemple, il est possible de créer un rôle donnant accès en lecture aux tableaux de bord de l'espace 1. Ces rôles représentent un ensemble de droits pouvant être attribué aux utilisateurs.

Pour répondre au besoin, conformément au schéma ci-dessous, un espace contenant le tableau de bord et les index des données est créé. Un rôle donnant accès en lecture au tableau de bord est aussi créé. Lorsqu'un utilisateur est créé, il n'y a plus qu'à lui attribuer le droit de lecture grâce au rôle qui a été créé.

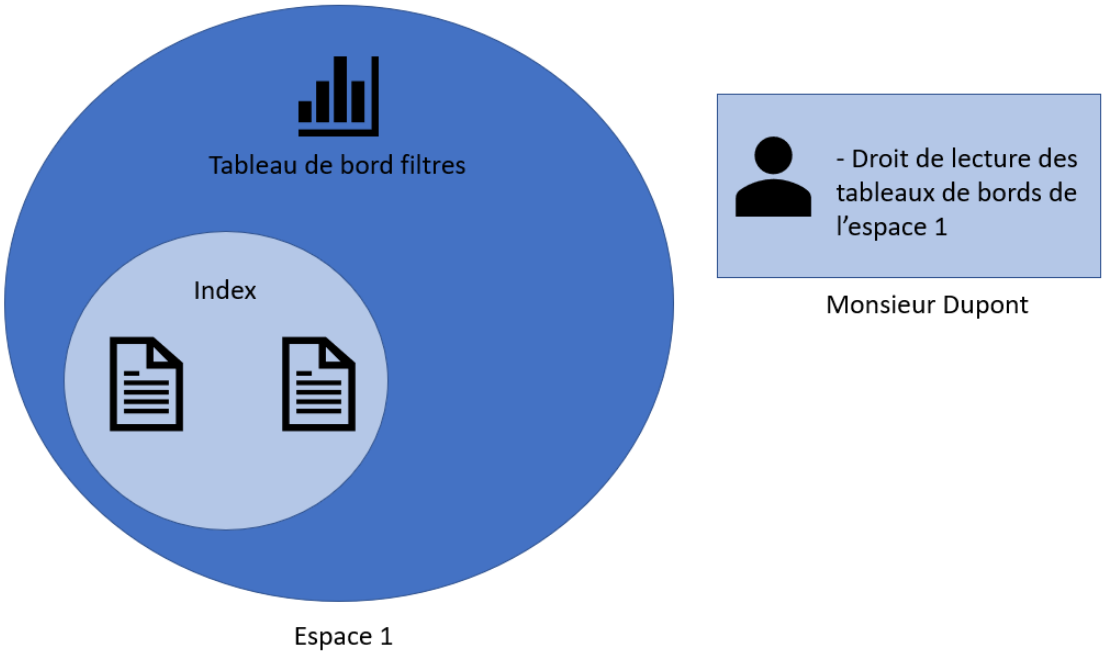


Figure 18. Droits utilisateurs, espace et rôle

Il est aussi possible de créer les comptes utilisateurs, les espaces et les rôles de deux façons différentes :

- En faisant des requêtes HTTP directement au logiciel Elasticsearch grâce à son API intégré
- Depuis un navigateur web en utilisant l'interface Kibana qui fera des requêtes automatiquement à Elasticsearch (comme sur les deux images ci-dessous)

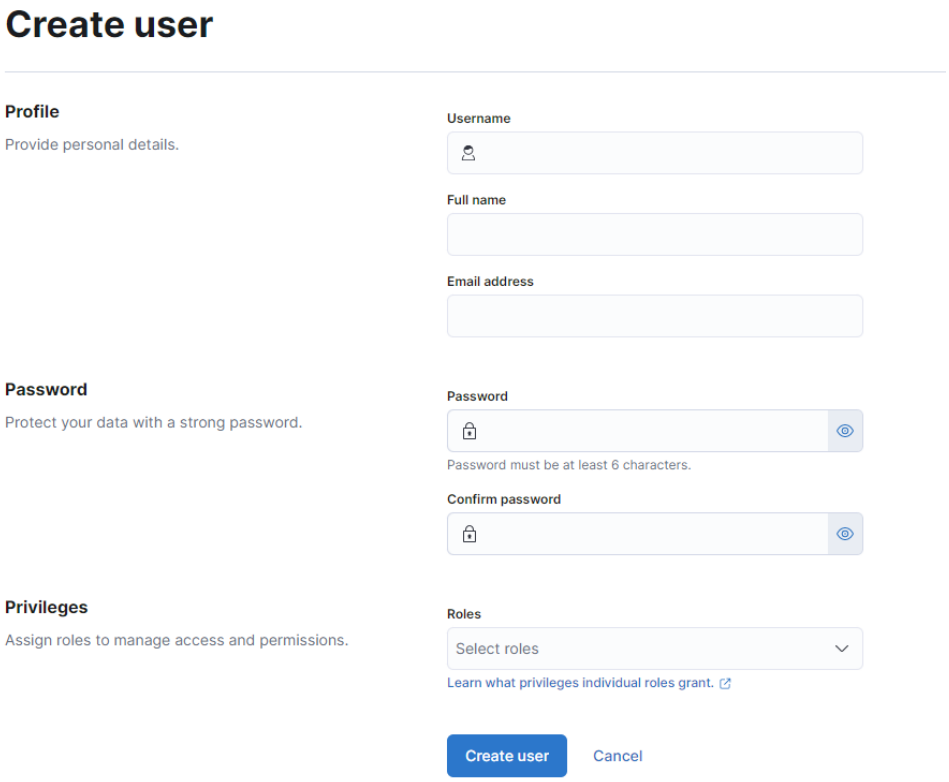


Figure 19. Page de création d'utilisateurs sur Kibana

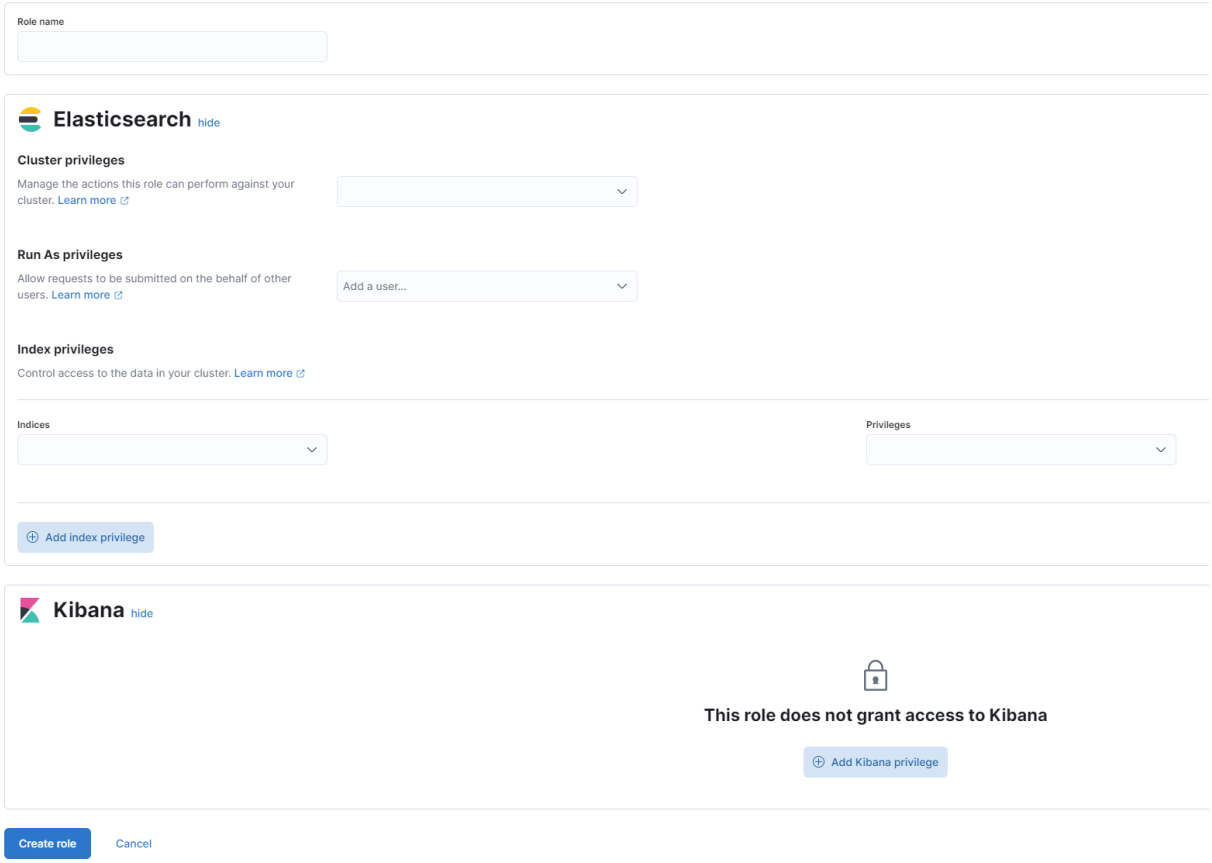


Figure 20. Page de création de rôles sur Kibana

3.3.3.4.3 Connexion à Kibana

Concernant l'accès à Kibana par un utilisateur, il faut désormais se connecter avec des identifiants. En se connectant à l'application web de Kibana (depuis un navigateur internet), il faudra rentrer ses identifiants sur une page de connexion pour accéder aux informations.

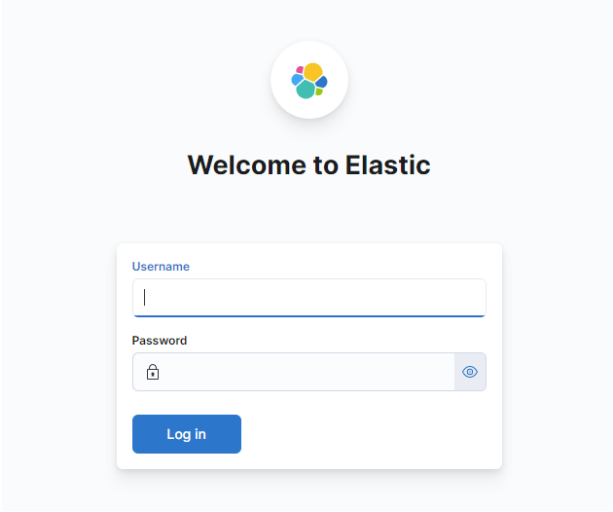


Figure 21. Page de connexion sur Kibana

Il est possible de se connecter à l'interface Kibana avec un compte technique ou un compte utilisateur.

Kibana et Elasticsearch utilisent les mêmes utilisateurs.

3.3.4 Mise en production

3.3.4.1 Machine virtuelle de production : résilience et confidentialité

L'objectif est de déployer l'infrastructure mise en place conformément à la figure 3. Il est important de respecter les normes en vigueur en matière de sécurité et de confidentialité.

La sécurisation des données et leur résilience est d'une importance cruciale lors de la mise en production. Les données constituent l'actif le plus précieux d'une entreprise, et leur protection est essentielle.

Il est nécessaire de se conformer aux réglementations. Une application vulnérable expose les informations sensibles à des risques tels que le vol, la modification ou la destruction. La mise en œuvre de mesures de sécurité robustes, telles que le chiffrement des données, l'authentification et l'autorisation appropriées permet de minimiser ces risques et de préserver l'intégrité des données.

La résilience des données est essentielle pour assurer une continuité de service en cas de défaillance ou de catastrophe. Des stratégies de sauvegarde, de réplication et de redondance doivent être mises en place pour garantir le fonctionnement du système de manière continue.

De même, la confidentialité des données joue un rôle majeure. Il existe au sein de la Société Générale un système de notation pour déterminer le niveau de confidentialité des données. Il a été établi que les données de ce projet sont d'un niveau de confidentialité critique. Ces données doivent impérativement rester en interne. Il n'est donc pas possible de déployer la solution sur une machine virtuelle ou sur un cloud extérieur à l'entreprise comme AWS.

Il existe en interne de l'entreprise un système permettant de déployer des machines virtuelles hébergées sur les serveurs de la Société Générale. Il est aussi possible de dupliquer l'instance de

cette machine virtuelle pour avoir plusieurs machines virtuelles en temps réel. Dans le cas présent les données seront présentes sur trois machines virtuelles distinctes. Cela implique que si une machine ne fonctionne plus, une autre prend le relais automatiquement et immédiatement. Cela permet de réduire tous les problèmes provenant de dysfonctionnement d'une VM. Ainsi, la solution mise en place sera conforme aux normes de sécurité et de confidentialité.

3.3.4.2 Sécurité et communications des progiciels ELK

Avant de passer en production, il faut respecter certaines normes en matière de sécurité du coté des progiciels de la suite ELK. Il y a plusieurs couches de sécurité à mettre en place qui sont les suivantes :

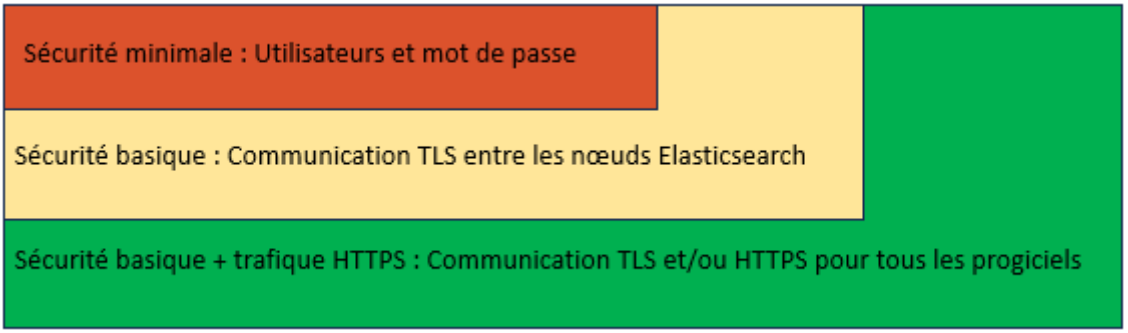


Figure 22. Couches de sécurités ELK

La première section comprend la configuration des utilisateurs et des mots de passe, étape déjà achevée dans le passé. La deuxième phase consiste à instaurer le protocole TLS (Transport Layer Security) pour sécuriser la communication entre les nœuds Elasticsearch. Actuellement, le système fonctionne avec un seul nœud, rendant cette étape optionnelle pour le moment. Cependant, en vue de faciliter les évolutions ultérieures, cette démarche est prise en compte.

Pour mettre en place la communication TLS, le processus suivi comprend :

- La création d'une autorité de certification.
- La génération de clés privées pour chaque nœud, ces clés étant ensuite signées par l'autorité de certification préalablement instaurée pour confirmer leur authenticité. Ces clés privées englobent à la fois un certificat de nœud et un certificat d'autorité.
- L'ajout de la clé privée générée à la configuration du progiciel Elasticsearch.

Les deux dernières étapes susmentionnées doivent être exécutées pour chaque nœud (toutefois, dans ce contexte, elles sont réalisées une seule fois). Dans l'optique de faciliter la génération des clés et des certificats, la suite Elasticsearch comprend des exécutables binaires paramétrables pour cette tâche.

La troisième et dernière couche de sécurité englobe la communication entre les différents progiciels, impliquant :

- Le chiffrement de la communication HTTP en HTTPS pour Elasticsearch.
- Le chiffrement de la communication client HTTP vers HTTPS pour Kibana.
- Le chiffrement des échanges entre Filebeat, Logstash et Elasticsearch.

Concernant la communication HTTPS pour Elasticsearch, cela nécessite la création d'une demande de signature de certificat (CSR, Certificate Signing Request) qui agit comme un certificat d'identité numérique pour sécuriser la communication. Cette certification contient une preuve d'authenticité qui garantit l'intégrité. Générée via les outils inclus dans la suite ELK, cette certification est ensuite intégrée à Elasticsearch via un module appelé Xpack, dédié à la gestion des paramètres de sécurité.

En appliquant une logique similaire, des certificats utilisant le protocole SSL sont employés pour établir une communication HTTPS avec Kibana. Par conséquent, lorsque les utilisateurs accèdent à Kibana via un navigateur, le protocole HTTPS assure la communication sécurisée.

La sécurisation des échanges entre les divers progiciels constitue l'étape ultime. Cela requiert le chiffrement des communications et le blocage des échanges non autorisés. Ainsi, en utilisant des clés de chiffrement publiques et des clés de déchiffrement privées, les transmissions entre Filebeat et Logstash, ainsi qu'entre Logstash et Elasticsearch, sont protégées.

À l'issue de ces étapes, les trois couches de sécurité sont mises en place. Néanmoins, un élément crucial demeure à traiter en matière de sécurité : la protection du contenu des fichiers de configuration. Il est impératif d'éviter la présence d'informations compromettantes, telles que les mots de passe, dans ces fichiers. À cet effet, toutes ces informations sont stockées dans des keystores.

3.3.4.3 Automatisation du déploiement

L'automatisation joue un rôle critique dans la rationalisation des processus informatiques. Un script Shell dédié à l'automatisation du déploiement représente un outil essentiel pour garantir une mise en production fluide et fiable. En éliminant la nécessité d'interventions manuelles répétitives, ce script réduit de manière significative le risque d'erreurs humaines. Les erreurs manuelles, bien que courantes, peuvent avoir des conséquences coûteuses en termes de temps, de ressources et de stabilité du système.

En automatisant le déploiement à l'aide d'un script, les étapes sont exécutées de manière cohérente et précise, minimisant ainsi les chances d'incohérences ou de configurations incorrectes. Cette automatisation permet aux équipes de se concentrer sur des tâches à plus forte valeur ajoutée, tout en garantissant un environnement de déploiement stable, sécurisé et hautement performant.

Le script qui a été conçu automatise non seulement le déploiement exhaustif d'une stack ELK (Elasticsearch, Logstash, Kibana), mais va bien au-delà de cette fonctionnalité basique. Il est soigneusement programmé pour installer la stack ELK et la configurer en accord avec les configurations préétablies en récupérant les fichiers du référentiel GitHub associé. De plus, il intègre de manière essentielle des mesures de sécurité robustes. Cela inclut la création de keystore, l'établissement de communications sécurisées en utilisant TLS, ainsi que la configuration du protocole HTTPS entre les diverses instances.

Le résultat de cette exécution englobe donc une stack ELK entièrement prête à l'emploi pour automatiser les processus d’analyse de données et elle est configurée selon les meilleures pratiques de sécurité.

3.3.5 Utilisation des outils internes de l’entreprise

Bien que la stack ELK soit un outil puissant pour la gestion et l'analyse des données, elle présente une limitation dans la capacité à tracer les connexions utilisateurs. Cette limitation devient critique compte tenu des exigences stipulées dans la charte de l’entreprise, qui stipulent clairement la nécessité de tracer ces connexions lorsque des données fonctionnelles sont mises à disposition. Dans cette perspective, afin de satisfaire cette exigence fondamentale de traçabilité, une orientation est prise en faveur de l'exploitation des outils internes propres à l'entreprise. Cette démarche vise à garantir que chaque interaction avec les données fonctionnelles soit minutieusement suivie et enregistrée, tout en respectant les directives et les normes en matière de sécurité et de traçabilité de l’entreprise.

La mise en place d'une solution qui répond de manière complète au besoin de traçabilité des connexions utilisateurs ainsi qu'à la création de tableaux de bord interactifs s'appuie sur un outil interne essentiel de l'entreprise. Cet outil permet non seulement de créer des tableaux de bord interactifs, mais également de gérer l'accès via les droits utilisateurs de l'entreprise. Ce système

garantit que seules les personnes autorisées ont accès aux tableaux de bord , tout en assurant que les demandes d'accès pour les utilisateurs non autorisés sont refusées conformément aux protocoles établis.

Le cœur de cet outil réside dans sa capacité à tracer les connexions utilisateurs. Chaque interaction est consignée, fournissant une piste claire et détaillée de l'activité des utilisateurs. Cette fonctionnalité répond parfaitement aux exigences de traçabilité stipulées par la charte de l'entreprise.

Afin de fournir des données à ces tableaux de bord interactifs, la création d'une interface entre l'outil et les données entreposées dans Elasticsearch est nécessaire. Cette tâche est accomplie par la mise en place d'une API. Plutôt que d'opter pour une démarche de développement à partir de zéro, les compétences internes de l'entreprise sont exploitées à travers l'utilisation d'un outil interne capable de générer automatiquement une API à partir d'un format JSON.

La phase suivante de la démarche implique la création d'une fonction en Python. Cette fonction requêtera l'API Elasticsearch pour extraire les données requises. Ces données devront ensuite être traitées pour correspondre au format JSON attendu. À ce stade, l'outil interne prend le relais pour créer automatiquement une API, cette fois-ci conçue spécifiquement pour être compatible avec les tableaux de bord interactifs.

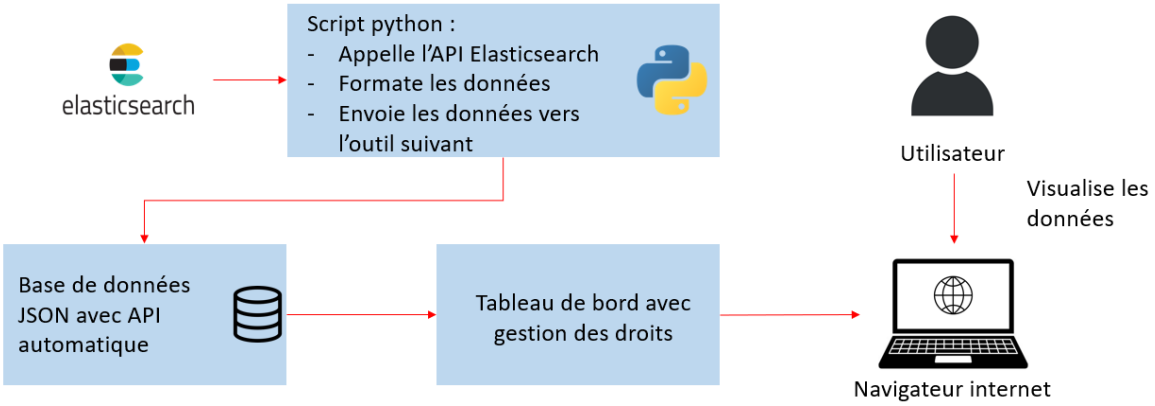


Figure 23. Architecture de la solution technique utilisant les outils internes

En combinant l'outil de création de tableaux de bord interactifs, l'API générée et la fonction de traitement des données en Python, il est possible d'atteindre l'objectif de mettre en place une solution qui répond parfaitement aux besoins de traçabilité des connexions utilisateurs tout en fournissant des tableaux de bord interactifs basés sur les données d'Elasticsearch. Cette approche enchaînée et intégrée tire parti des ressources et des compétences de l'entreprise pour aboutir à une solution cohérente et performante.

Néanmoins, même si la suite ELK mise en place ne sera pas utilisée par les utilisateurs finaux pour des raisons de traçabilité des connexions utilisateurs en raison, la stack continuera de jouer un rôle fondamental en interne au sein de l'équipe. Cette décision est guidée par le souci de respecter scrupuleusement les directives de sécurité de l'entreprise.

L'utilisation de la suite ELK au sein de l'équipe apportera un avantage significatif en optimisant l'efficacité des analystes métier. En fournissant des données de manière plus accessible et en automatisant certaines tâches d'analyse, l'outil permettra aux analystes de gagner un temps précieux dans leurs processus d'analyse, tout en contribuant à des prises de décision plus informées et rapides.

3.3.6 Documentation technique et fonctionnelle

La rédaction de documentation technique et fonctionnelle occupe une place fondamentale dans l' approche de travail. La documentation technique, en particulier, assure la traçabilité des

processus de développement, rendant chaque étape compréhensible et reproductible. Elle constitue un pilier clé pour garantir la pérennité des solutions mises en place, facilitant la maintenance future et la résolution efficace de problèmes. De plus, elle renforce la communication au sein de l'équipe en offrant une référence commune, ce qui est essentiel dans un contexte multidisciplinaire.

La documentation fonctionnelle, quant à elle, joue un rôle central dans la transmission cohérente des objectifs et des caractéristiques des projets. Elle établit un pont entre les exigences des utilisateurs et les solutions techniques, en garantissant que chaque membre de l'équipe partage une compréhension uniforme des attentes. Cette documentation est essentielle pour éviter les malentendus et les déviations par rapport aux objectifs fixés.

Une documentation technique exhaustive a été rédigée en anglais et mise à disposition sur GitHub. Cette documentation constitue une ressource inestimable pour l'équipe, fournissant des instructions détaillées sur l'installation et la configuration de l'instance logicielle. Elle assure une référence fiable pour tout membre de l'équipe impliqué dans le déploiement et la maintenance du système. Grâce à cette documentation, les procédures complexes sont simplifiées, permettant d'économiser du temps précieux et d'éviter les erreurs potentielles lors de l'installation.

En plus de l'aspect technique, la documentation en anglais fournit une explication sur la manière dont les progiciels fonctionnent. Elle guide les membres de l'équipe à travers les différentes fonctionnalités et modules, en fournissant une compréhension approfondie du fonctionnement interne du système. Cela s'avère extrêmement précieux pour les développeurs, les analystes fonctionnels et tous les membres de l'équipe travaillant en étroite collaboration sur le projet.

Parallèlement à la documentation technique, un guide d'utilisation utilisateur a été élaboré en anglais, visant à familiariser les utilisateurs finaux avec le logiciel d'une manière fonctionnelle. Ce guide est conçu pour permettre aux utilisateurs de prendre en main le logiciel de manière autonome, en fournissant des instructions claires et étape par étape pour chaque fonctionnalité.

Ce document est également enrichi par des exemples de cas d'utilisation concrets qui correspondent à des besoins fonctionnels spécifiques. Ces exemples offrent une approche pratique pour mettre en œuvre des solutions en utilisant le logiciel, ce qui peut s'avérer particulièrement utile pour les utilisateurs moins familiers avec les aspects techniques. En somme, cette documentation technique et fonctionnelle en anglais joue un rôle essentiel dans l'efficacité de l'équipe et dans la satisfaction des utilisateurs finaux, en fournissant une source complète et accessible d'informations et de guides.

3.4 Résultats et analyse

3.4.1 Période de test

La mise en place de la solution a été soumise à une phase cruciale d'évaluation, coïncidant avec une période stratégique marquée par un arrêté mensuel. Durant cette période, les utilisateurs finaux ont eu l'opportunité de tester activement la solution et de fournir des retours concrets sur son fonctionnement et son utilité. Pour faciliter cette transition, une session d'initiation à l'utilisation des outils a été organisée, visant à guider les utilisateurs dans la manipulation de l'interface utilisateur nouvellement mise à disposition.

Les retours des utilisateurs ont été majoritairement positifs, témoignant de la pertinence et de la facilité d'utilisation de la solution. Bien que des problèmes techniques mineurs aient été identifiés au départ, l'équipe a rapidement réagi pour les résoudre dès leur détection. Cette réactivité a grandement contribué à maintenir une expérience utilisateur fluide et sans interruption.

3.4.2 Améliorations possibles

Cependant, les retours des utilisateurs ont également identifié des axes d'amélioration importants pour optimiser davantage la solution. Trois points majeurs ont été relevés : tout d'abord, la nécessité d'ajouter des informations concernant les rejets effectués par l'application. Alors que les filtres étaient déjà un objectif, les utilisateurs ont souligné que l'inclusion d'informations spécifiques sur les rejets améliorerait la clarté des résultats.

Ensuite, une demande majeure émanant de l'équipe interne concerne la possibilité d'importer leurs propres fichiers de logs, au-delà des fichiers de logs de production. Cette extension permettrait aux membres de l'équipe d'effectuer des tests plus précis et approfondis en utilisant des données spécifiques à leur contexte. Cette amélioration offrirait une flexibilité accrue et une utilisation plus personnalisée de la solution.

Une remarque pertinente émanant des utilisateurs portait sur la nécessité d'améliorer la clarté des informations fournies par les filtres. Bien que les filtres aient déjà été inclus dans les objectifs, les utilisateurs ont exprimé le souhait d'une meilleure explicitation des informations correspondant à chaque filtre. Cette amélioration permettrait une compréhension plus rapide et plus précise des résultats affichés par la solution.

Pour répondre à cette demande, l'équipe envisage de collaborer étroitement avec les utilisateurs lors d'un meeting dédié. L'objectif de ce meeting serait de créer un mapping détaillé entre les noms des filtres et les informations spécifiques auxquelles ils se réfèrent. En travaillant main dans la main avec les utilisateurs, l'équipe s'efforcera d'établir une correspondance précise entre les filtres et les données sous-jacentes, garantissant ainsi que chaque filtre soit clairement associé à son contexte pertinent.

Dans l'ensemble, cette phase de test a été essentielle pour identifier les points forts et les domaines d'amélioration de la solution. Les retours des utilisateurs ont été précieux pour orienter les évolutions futures, renforçant ainsi la capacité de la solution à répondre aux besoins spécifiques de l'équipe et à fournir une expérience utilisateur optimale.

3.5 Extension des cas d'utilisations

3.5.1 Ajout des rejets

Suite aux retours pertinents des utilisateurs lors de la phase d'évaluation, une évolution significative a été apportée aux cas d'utilisation de la solution. Concrètement, cette évolution a consisté à intégrer de manière explicite les informations relatives aux rejets au sein des résultats affichés. Cette nouvelle dimension renforce la clarté et la pertinence des résultats, permettant aux utilisateurs de mieux comprendre les données traitées et les rejets potentiels.

Cette modification reflète un engagement à répondre de manière proactive aux besoins exprimés par les utilisateurs, améliorant ainsi la qualité globale de l'expérience utilisateur et renforçant l'efficacité des analyses effectuées au sein de l'équipe.

Pour ajouter ces informations, cela nécessite premièrement une période de recherche pour déterminer où et comment récupérer les informations concernant les rejets. Il a été déterminé que ces données sont présentes dans la base de données. Pour les intégrer à la solution il est donc requis d'ajouter un nouveau Pipeline dans Logstash. Dorénavant les pipelines Logstash seront les suivantes :

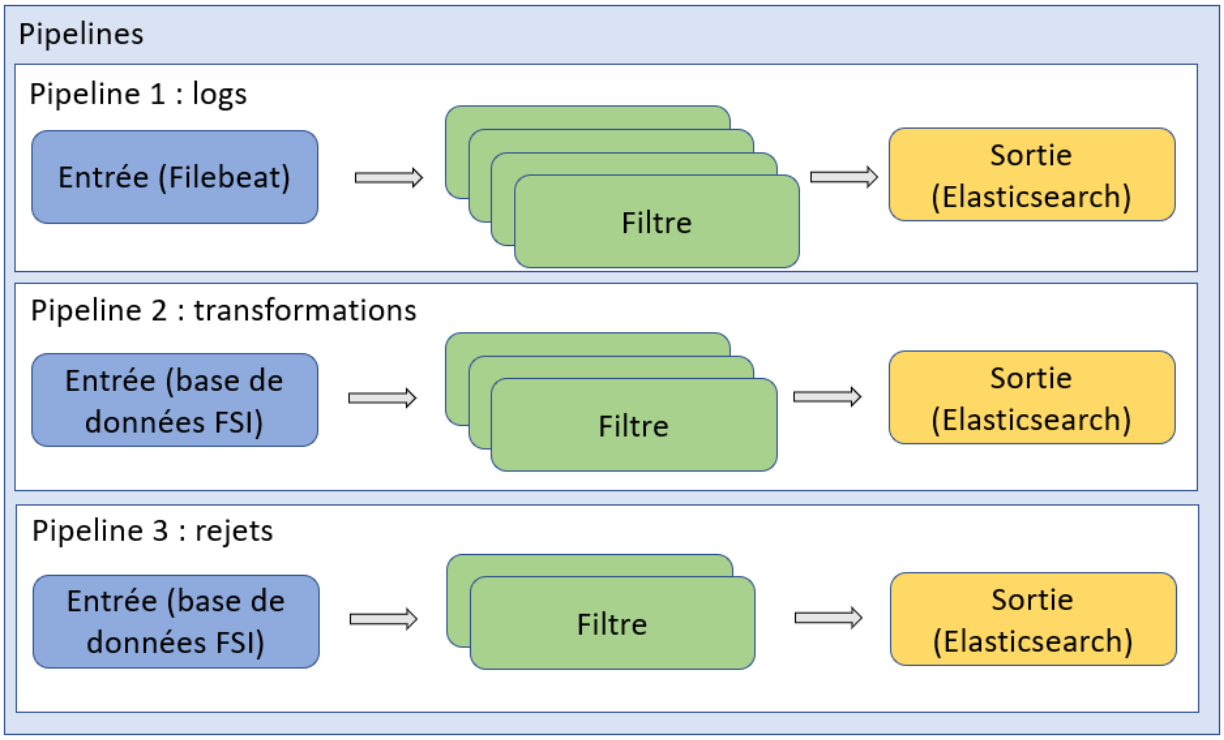


Figure 24. Pipelines Logstash avec les rejets

Tout changement apporté aux rejets doit être reflété à plusieurs niveaux pour garantir une cohérence et une utilisation harmonieuse.

En premier lieu, la modification des rejets implique une mise à jour de l'interface utilisateur de Kibana, qui joue un rôle central dans le processus d'analyse en interne au sein de l'équipe. Cette interface doit être adaptée pour prendre en compte les nouveaux paramètres de rejet, assurant ainsi que les utilisateurs internes disposent d'une représentation précise des données.

De même, le script Python utilisé pour récupérer les données depuis l'API Elasticsearch doit être ajusté pour refléter l'ajout au niveau des rejets. Cette synchronisation est essentielle pour garantir que les données récupérées soient cohérentes avec les nouveaux critères de rejet, et que l'analyse et les résultats restent précis.

En parallèle, l'interface du tableau de bord destinée aux utilisateurs externes à l'équipe doit également être modifiée pour refléter les rejets. Cette étape est cruciale pour assurer que les utilisateurs externes accèdent aux données conformément aux critères mis à jour.

Cette coordination d'ajustements à travers plusieurs composantes de la solution souligne la complexité des interactions entre les différents éléments. Cependant, elle garantit également l'intégrité et la cohérence des données et des résultats à tous les niveaux de l'utilisation. Cette approche méthodique et multidimensionnelle renforce la fiabilité de la solution et maintient sa capacité à répondre avec précision aux besoins spécifiques de l'équipe, tout en offrant une expérience homogène pour les utilisateurs internes et externes à l'équipe.

3.5.2 Instance UAT

Pour répondre aux besoins spécifiques de l'équipe et améliorer davantage l'efficacité des processus, une deuxième instance de la suite ELK a été déployée sur une machine virtuelle distincte. Cette nouvelle instance se distingue par sa source de données : au lieu de récupérer les logs à partir du serveur de production, elle se connecte au serveur UAT (User Acceptance Testing). Cette modification stratégique permet aux membres de l'équipe d'importer leurs propres fichiers de logs en les plaçant sur le serveur UAT, sans perturber l'environnement de production.

En fournissant un environnement distinct pour les fichiers de logs de test, cette deuxième instance ELK renforce la flexibilité de l'équipe pour effectuer des essais, des analyses et des expérimentations. Les membres de l'équipe peuvent ainsi explorer les fonctionnalités de la solution, tester des scénarios spécifiques et simuler différentes situations, tout en préservant l'intégrité et la stabilité de l'environnement de production.

Cette initiative a prouvé son utilité en offrant un espace sécurisé pour les membres de l'équipe afin d'explorer et d'expérimenter, sans compromettre la qualité ni l'efficacité de l'environnement de production. En somme, cette deuxième instance ELK ouvre de nouvelles perspectives pour une analyse approfondie et une expérimentation ciblée, renforçant ainsi la capacité de l'équipe à innover et à optimiser la suite d'outils ELK.

3.5.3 Passation de connaissances

La passation de connaissances au sein de l'équipe est une étape cruciale qui découle naturellement de l'engagement envers la documentation. Le transfert de connaissances assure que chaque membre de l'équipe est capable de contribuer de manière significative et d'assurer la continuité en cas de rotations ou d'évolutions au sein de l'équipe. Cette approche réduit la dépendance envers un individu particulier et renforce la résilience de l'équipe dans son ensemble.

Les sessions de passation de connaissances, qui se produisent régulièrement au cours des sprints, permettent de partager les meilleures pratiques, les méthodologies de travail et les enseignements tirés des expériences passées. Cela contribue à une montée en compétences constante et favorise un environnement collaboratif où chaque membre peut apporter sa contribution unique. En fin de compte, la documentation et la passation de connaissances ne sont pas seulement des pratiques, mais des investissements stratégiques dans la croissance de l'équipe et la réussite à long terme des projets.

4 Conclusion

4.1 Conclusion technique

En termes de conclusion technique, les objectifs établis ont été atteints avec succès. Les processus d'analyse de données ont été automatisés et une approche CI/CD (livraison et intégration continue) a été mise en œuvre pour réduire les tâches manuelles et accélérer la disponibilité des données pour les utilisateurs finaux, engendrant ainsi des améliorations substantielles en matière d'efficacité opérationnelle. L'implémentation de la stack ELK, associée à une interface intuitive destinée à la consultation des données de traitement, a renforcé la visibilité et la traçabilité des informations essentielles au sein de l'entreprise.

La réussite de ce projet a été grandement influencée par la collaboration étroite avec divers intervenants, dont les développeurs, les analystes métiers, le support technique, le support fonctionnel et les utilisateurs finaux. Le suivi d'une méthodologie agile, incluant des réunions de synchronisation quotidiennes, des réunions hebdomadaires et des cycles de développement de trois semaines, a maintenu un alignement constant avec les objectifs et les besoins changeants de l'entreprise.

L'introduction d'outils internes pour gérer les connexions utilisateurs, créer des tableaux de bord interactifs et générer automatiquement une API a grandement amélioré la convivialité et la flexibilité de la solution. Les retours positifs des utilisateurs finaux, ainsi que les ajustements effectués en réponse à leurs remarques, ont confirmé l'impact positif des efforts fournis.

Dans l'ensemble, cette initiative a montré la valeur tangible de l'automatisation et de la mise en œuvre d'approches modernes de développement et de déploiement. Les résultats obtenus s'alignent étroitement sur les objectifs stratégiques de l'entreprise en matière d'efficacité opérationnelle et de qualité des services offerts aux utilisateurs finaux. Forts de ces réalisations, une confiance s'affirme quant à l'impact positif et durable de cette solution dans le cadre de l'amélioration continue des processus de l'entreprise.

4.2 Conclusion personnelle

En conclusion personnelle, cette expérience a été enrichissante à bien des égards. La participation à ce projet m'a permis d'approfondir mes compétences en matière de développement, d'automatisation et de gestion de données, tout en me donnant l'opportunité de travailler au sein d'une équipe multidisciplinaire. La collaboration avec des experts issus de divers domaines a favorisé l'apprentissage et l'échange de connaissances, tout en renforçant ma compréhension des besoins et des contraintes propres à chaque rôle.

L'application des méthodologies agiles m'a également permis d'apprécier l'importance de la flexibilité et de l'adaptabilité dans un environnement en constante évolution. Le suivi régulier des progrès et l'ajustement continu des objectifs ont été cruciaux pour maintenir la cohérence et l'efficacité de notre démarche.

Cependant, cette expérience n'a pas été sans ses défis. La complexité technique et les ajustements nécessaires en cours de route ont nécessité une résolution de problèmes constante et une prise de décisions informées. Ces défis ont toutefois renforcé ma capacité à gérer l'incertitude et à persévérer face aux obstacles.

Dans l'ensemble, ce projet a solidifié ma conviction en l'importance de l'automatisation intelligente et de l'approche itérative pour améliorer les processus opérationnels. Cette expérience a été une étape significative dans ma croissance professionnelle et m'a inspiré à continuer à explorer des moyens innovants pour résoudre les défis complexes auxquels les entreprises sont confrontées.

5 Bibliographie

1. Auteurs anonymes, "Société Générale" [en ligne], Wikipédia, Lien : https://fr.wikipedia.org/wiki/Société_générale (consulté le 14.04.2023)

2. Auteurs anonymes, "Société Générale : présentation" [en ligne], Société Générale, Lien : https://www.societegenerale.com/fr/le-groupe-societe-generale/identite/presentation (consulté le 14.04.2023)

3. Auteurs anonymes, "ÉTHIQUE ET GOUVERNANCE : NOS ENGAGEMENTS" [en ligne], Société Générale, Lien : https://www.societegenerale.com/fr/responsabilite/ethique-et-gouvernance (consulté le 21.04.2023)

4. Olivier Jan, "Introduction à ELK : Elasticsearch, Logstash et Kibana" [en ligne], Wooster, Lien : <https://wooster.checkmy.ws/2014/04/elk-elasticsearch-logstash-kibana/> (consulté le 02.05.2023)

5. Shay Banon, "Elasticsearch Reference Guide" [en ligne], Elastic, Lien : <https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html> (consulté le 14.06.2023)

6. Shay Banon, "Manually configure security" [en ligne], Elastic, Lien : <https://www.elastic.co/guide/en/elasticsearch/reference/current/manually-configure-security.html> (consulté le 12.07.2023)