



UNIVERSITÉ  
CAEN  
NORMANDIE

Université Caen Normandie  
- UFR DES SCIENCES (CAMPUS 2)  
6 Bd Maréchal Juin, 14000 Caen



Groupe de recherche en informatique,  
image, automatique et instrumentation  
de Caen  
Campus 2, Bâtiment F, 6 Boulevard  
Maréchal Juin, 14000 Caen

# Rapport de stage

## Développeur Front-End

Simon CARDOSO

Du Lundi 10 Octobre 2022 au Vendredi 17 Février 2023

### Maîtres de stage :

Christophe ROSENBERGER  
Emmanuel GIGUET

Année : 2022 / 2023

# Sommaire

<b>Introduction</b>	<b>3</b>
<b>Présentation de l'entreprise</b>	<b>3</b>
<b>Projet de stage</b>	<b>4</b>
Les attentes du GREYC	4
Les contributeurs	4
Finalité	5
L'existant	6
Etat de l'art	7
<b>Mise en oeuvre du cahier des charges</b>	<b>8</b>
Les grands axes	8
Problématique	9
Ma contribution	14
Technologique	14
Technique	16
Difficultés rencontrées	18
<b>Axes d'amélioration</b>	<b>18</b>
<b>Les plus-values du stage</b>	<b>19</b>
<b>Conclusion</b>	<b>20</b>

# Introduction

Au cours de ma période de stage au sein du Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen (GREYC), j'ai eu l'opportunité enrichissante de travailler au sein de l'équipe SAFE en tant que développeur Front-end. Cette expérience m'a offert un aperçu passionnant du monde professionnel et m'a permis d'appliquer mes compétences en développement web dans un environnement de recherche de pointe.

L'équipe SAFE, spécialisée dans la sécurité informatique, la cryptographie et les systèmes distribués, m'a accueilli chaleureusement et m'a impliqué dans des projets stimulants dès le début de mon stage. En tant que développeur Front-end, j'ai eu l'occasion de collaborer avec des chercheurs et des développeurs expérimentés, tout en contribuant activement aux solutions logicielles qui jouent un rôle crucial dans la réalisation des objectifs de l'équipe.

Je vais présenter le GREYC, décrire les missions et les objectifs de l'équipe SAFE, ainsi que les projets spécifiques sur lesquels j'ai travaillé en tant que développeur Front-end. Je vais également mettre en évidence les compétences que j'ai acquises et les enseignements que j'ai tirés de cette expérience immersive au croisement de la recherche et du développement informatique.

Mon stage au GREYC a été une occasion exceptionnelle d'apprendre, de contribuer et de grandir professionnellement. Les défis auxquels j'ai été confronté et les réalisations que j'ai accomplies ont largement enrichi ma compréhension du développement Front-end et ont renforcé mon désir de poursuivre une carrière dans ce domaine dynamique et en constante évolution.

## Présentation de l'entreprise

Le GREYC est un laboratoire de recherche dans les domaines des sciences humaines et sociales et le domaine des interactions de l'informatique avec les mathématiques et les sciences de l'ingénieur. Le GREYC est situé dans plusieurs villes telles que Alençon, Caen, Cherbourg, Lisieux, Saint-Lo, Vire. J'ai réalisé mon stage sur le site du GREYC de Caen dans le bâtiment F situé sur le campus 2. Le laboratoire est constitué de plusieurs équipes Electronique, Image, Safe, CODAG, AMACC et MAD qui sont réparties sur les différents sites. Le bâtiment F est constitué d'une partie de l'équipe Electronique, Image et Safe. C'est l'équipe Safe

(sécurité, architecture, forensique et biométrie) que j'ai rejoint pour réaliser mon stage qui porte essentiellement sur le projet GDIP.

## Projet de stage

Quelles ont été les source de motivation poussant le GREYC à concevoir GDIP :

La conception : créer une application logicielle et matérielle ambitieuse et novatrice dans le domaine du forensique.

La recherche : le projet permet de capitaliser sur le savoir-faire du Greyc en traitement automatique de la langue, traitement d'image, benchmarking d'outils forensiques, recherche/création de corpus d'évaluation et proposition de méthode.

L'enseignement : GDIP serait utilisé comme support pour des modules tel que Identité numérique pour les 3A de l'ENSICAEN et Forensique pour les étudiants de master informatique à l'UNICAEN.

L'usage : GDIP, pourrait être utilisé par la section de recherche de la police de Caen afin de leur faire gagner du temps sur tout ce qui touche la fouille d'outils numérique. De plus, il permettrait d'analyser les traces numériques d'artistes contemporains pour l'IMEC.

## Les attentes du GREYC

Pour ce projet le Greyc cherchait un développeur frontend Vue, qui soit à l'aise avec l'utilisation des APIs, la création de visualisation avec D3 et la gestion de grands lots de données. Cela afin de créer une application web ergonomique et rapide qui puisse interagir avec des données stockées en base de données.

## Les contributeurs

GDIP est un projet qui a eu comme contributeur un certain nombre d'étudiants que ce soit à travers des projets ou bien des stages. Les premiers étudiants à coopérer sur ce projet étaient des étudiants de 3A de l'Ensicaen. Ils ont travaillé sur la structure de la plateforme et la création de filtres. Par la suite, un étudiant en Licence 3 a travaillé sur la conception d'une interface graphique. Il y a également un élève de

Master qui a coopéré afin d'optimiser le platform en Python. Il est désormais ingénieur et continue son travail sur la plateforme. Parmi les derniers arrivés, il y a un élève de Master 2 qui réalise une application web, et un ingénieur qui travaille sur la création de nouveaux filtres. Les managers du projet sont Emmanuel Giguet et Christophe Rosemberger

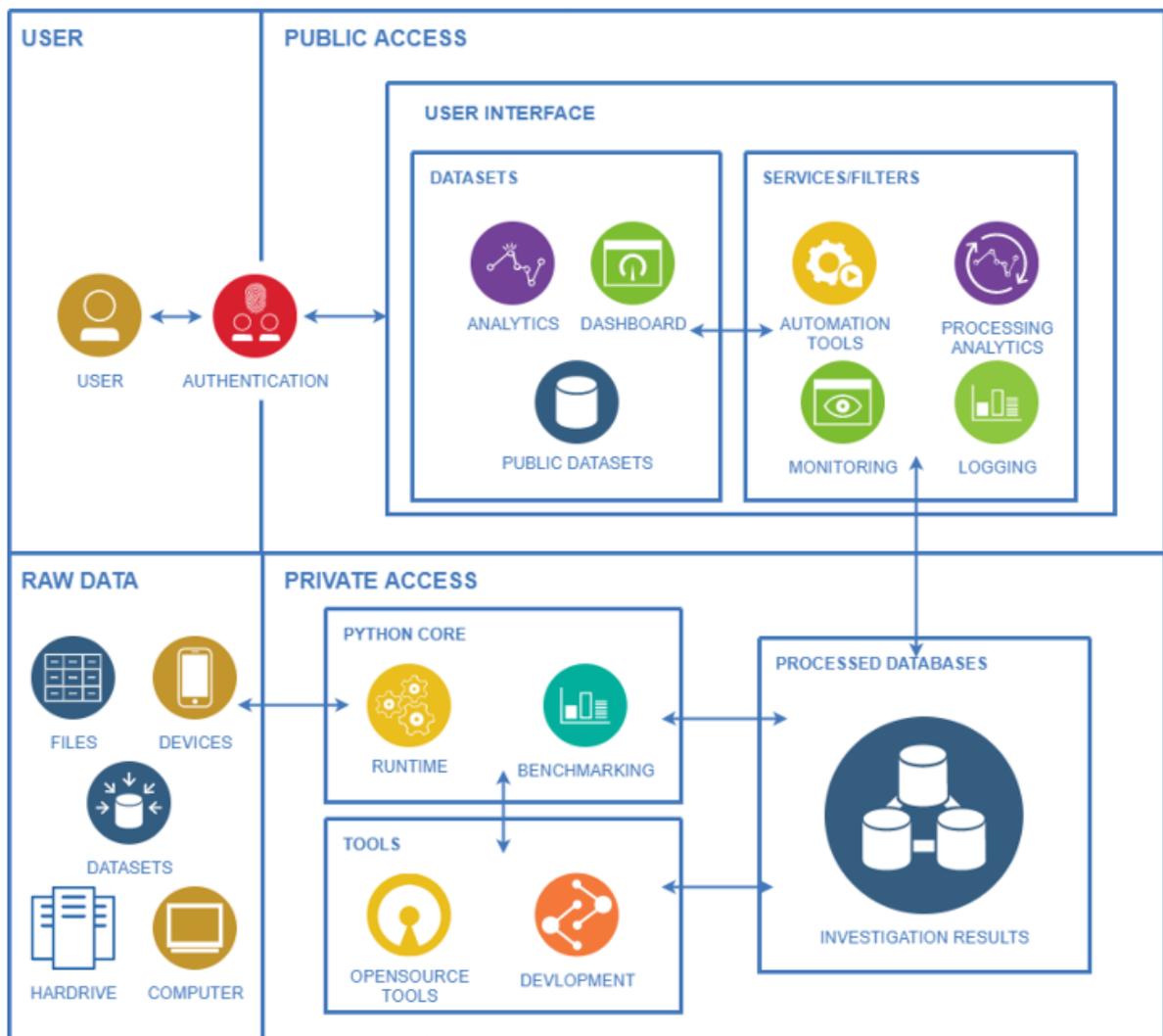
## Finalité

GDIP est une application forensique conçue dans le but de simplifier considérablement les recherches de fichiers pour les outils numériques. Elle se compose de deux principaux projets : la partie privée, appelée Backend, et la partie publique, nommée Frontend.

Le Backend a pour rôle d'acquérir les données à partir d'un ensemble de données, de les enregistrer dans une base de données et de les enrichir de manière entièrement automatisées. GDIP se distingue par son caractère novateur et ambitieux dans le domaine de la forensic, ce qui se reflète dans les filtres qu'il propose. Il permet non seulement de rechercher des fichiers par leur nom, leur taille ou leur extension, mais également de rechercher des entités spécifiques dans le contenu des fichiers. Par exemple, il est possible de rechercher des adresses e-mail, des noms d'entreprises, ainsi que de déterminer si une photo a été prise en intérieur ou en extérieur, et la même chose pour une vidéo. Ces filtres enrichissent les informations relatives aux documents traités.

Quant au Frontend, son objectif principal est de rendre l'information claire et facilement accessible. Pour ce faire, l'interface propose différentes visualisations, telles que le pourcentage de chaque type de fichier, le nombre de fichiers créés ou téléchargés par année, mois ou jour, ainsi que la représentation de l'architecture des datasets observés. De plus, il permet de lister tous les fichiers du dataset avec une icône correspondante, offrant ainsi à l'utilisateur une idée rapide du contenu de chaque fichier. Il est également possible de filtrer les données à l'aide de filtres novateurs, ce qui permet de restreindre l'espace de recherche.

La figure ci-dessous illustre la structure du projet et son utilisation, offrant ainsi une meilleure compréhension visuelle de celui-ci.



## L'existant

Lorsque je suis arrivé, toute la partie backend de GDIP était déjà en place. Cela comprenait les scripts Python qui permettaient de parcourir un dataset et d'extraire des informations telles que les noms de fichiers, les extensions et leur taille. De plus, il y avait des scripts et des bibliothèques qui étaient utilisés pour générer des données pour des filtres plus complexes, comme celui permettant de détecter la présence d'une personne dans une image ou celui permettant de rechercher des images prises en intérieur ou en extérieur. Enfin, il y avait un script chargé de peupler la base de données.

## Etat de l'art

Voici une liste non exhaustive d'applications de forensique qui permet également de mener des investigations sur une collection, avec en détail leurs avantages et inconvénients :

## **EnCase Forensic :**

- Avantages :
  - Vaste expérience et réputation dans le domaine de la forensique numérique.
  - Fonctionnalités complètes pour la collecte, l'analyse et la présentation des preuves.
  - Prise en charge de divers systèmes de fichiers et types de médias.
  
- Inconvénients :
  - Coût élevé, ce qui peut limiter l'accès pour les petites organisations.
  - Interface complexe nécessitant une formation approfondie.

## **AccessData FTK (Forensic Toolkit) :**

- Avantages :
  - Interface utilisateur conviviale et intuitive.
  - Capacités d'analyse et de récupération puissantes.
  - Prise en charge des formats de disques physiques et virtuels.
  
- Inconvénients :
  - Coût élevé pour la version complète.
  - Moins d'options pour l'analyse de la mémoire vive.

## **X-Ways Forensics :**

- Avantages :
  - Haute performance avec utilisation efficace des ressources système.
  - Flexibilité dans l'analyse de données, de l'analyse légère à complète.
  
- Inconvénients :
  - Interface basée sur des lignes de commande, moins conviviale pour certains utilisateurs.

- Moins d'outils d'automatisation que certaines autres solutions.

### **Autopsy :**

- Avantages :
  - Open source, gratuit et personnalisable.
  - Interface graphique conviviale, adaptée aux utilisateurs de tous niveaux.
  - Possibilité d'ajouter des modules et des plugins.
- Inconvénients :
  - Moins robuste que des solutions commerciales.
  - Moins de fonctionnalités avancées pour l'analyse de mémoire vive.

## Mise en oeuvre du cahier des charges

### Les grands axes

- Création de prototypes : J'ai participé à la création de prototypes pour différentes fonctionnalités du projet. Cela a permis de visualiser et d'évaluer les idées avant leur implémentation complète, ce qui a favorisé l'itération et l'amélioration des concepts.
- Création d'une API : J'ai développé une interface de programmation d'application (API) qui facilite l'échange de données entre le frontend et le backend. Cette API permet de transférer efficacement les informations nécessaires pour le bon fonctionnement de l'application.
- Création d'une application web : J'ai contribué à la création d'une application web dans le cadre du projet. Cela inclut la mise en place de fonctionnalités spécifiques, l'intégration de composants et la gestion de l'interface utilisateur.
- Création d'une bibliothèque de composants : J'ai également participé à la mise en place d'une bibliothèque de composants réutilisables. Ces

composants permettent d'améliorer l'efficacité du développement en offrant des solutions prêtes à l'emploi pour des fonctionnalités couramment utilisées.

- Modification de la base de données : J'ai contribué à l'évolution de la structure de la base de données en créant de nouvelles tables et en modifiant certaines existantes. Ces modifications ont permis d'adapter la base de données aux besoins spécifiques du projet et d'améliorer son efficacité.
- Migration de la base de données SQL à NoSQL : J'ai participé à la migration de la base de données du système de gestion de base de données SQL vers une solution NoSQL. Cette migration a nécessité des ajustements et des adaptations pour assurer la compatibilité des données et le bon fonctionnement du système.

## Problématiques

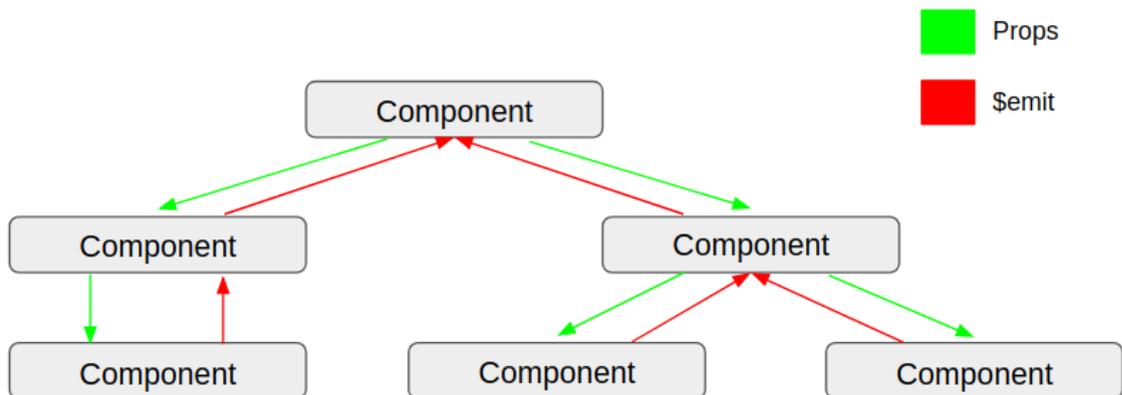
1. Réutilisabilité : L'équipe SAFE a constaté que le projet GDIP partage de nombreux aspects communs avec d'autres projets. Dans le but d'optimiser le temps de développement et de maintenance, il m'a été demandé de créer des composants hautement réutilisables et transférables vers d'autres projets. Ainsi, j'ai entrepris de développer une bibliothèque de composants externes au projet principal.

Pour assurer une réutilisabilité maximale, il est essentiel de concevoir des composants abstraits. L'idée est donc d'extraire toute la logique métier des composants, afin qu'ils ne se concentrent que sur des tâches génériques pouvant être appliquées à différents cas. Ces tâches sont principalement liées à l'affichage des données.

Dans ce but, Vue met à notre disposition deux attributs clés : Props, qui permet de transmettre des données du composant parent au composant enfant, et Emit, qui permet de remonter des événements vers le parent. Grâce à Props, les données métier ne dépendent pas directement du composant abstrait, offrant ainsi une grande flexibilité quant à la structure des données transmises. Le composant enfant n'a plus qu'à utiliser les données fournies.

Cependant, dans certains cas, il peut être nécessaire de modifier les données en réponse à un événement. Cette opération requiert généralement des informations sur le type de données et leur signification. C'est là qu'intervient la propriété Emit, qui permet de remonter un événement vers le composant parent. Ainsi, les modifications sont effectuées par le parent qui détient la

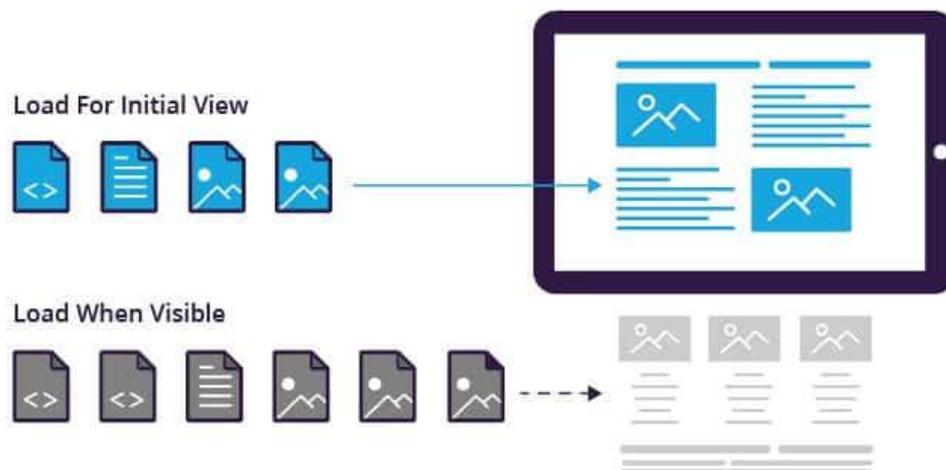
logique métier. Pour mieux comprendre le fonctionnement des propriétés Props et Emit, voici un schéma illustrant leur utilisation.



En mettant en place cette approche, nous favorisons la réutilisabilité des composants et offrons une grande flexibilité lors de leur utilisation dans différents projets.

2. Gestion du temps : Étant donné que nous traitons de grands volumes de données et que certaines opérations peuvent être longues, il est essentiel de fournir à l'utilisateur une indication visuelle du déroulement des opérations. Pour cela, deux solutions sont envisageables : créer une animation de chargement ou mettre en place une barre de progression. Dans notre cas, la barre de progression est la solution privilégiée, car elle permet de visualiser en temps réel l'avancement des opérations, offrant ainsi à l'utilisateur une meilleure compréhension du temps d'attente.

Pour réduire le temps d'attente, j'ai mis en place deux stratégies. La première consiste à utiliser le lazy loading (chargement différé). Cette approche permet de traiter de plus petits lots de données, simplifiant ainsi le traitement et l'envoi des données. De plus, elle permet de charger les données uniquement lorsque cela est nécessaire, ce qui contribue à optimiser les performances.



La deuxième stratégie est la factorisation, qui permet de réduire le volume de caractères émis lors de chaque transaction. L'idée est de remplacer les champs redondants d'un message qui transite de l'API vers le frontend par une valeur plus concise. Étant donné que le frontend doit être capable de reconstituer les données d'origine, nous envoyons une liste de tableaux de transformation. Voici un exemple concret de ce qui est réalisé.

Table File\_info, état initiale

id	name	type	path
1	image0.png	Image	.../.../..
2	image1.png	Image	.../.../..

Table File\_Type, créer pour effectuer les changement

id	type
1	Image
2	Audio

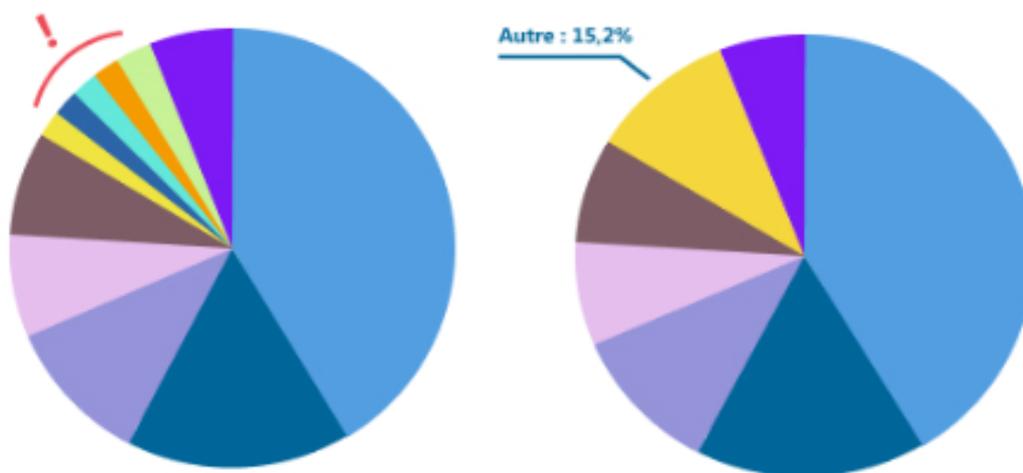
Table File\_info, après modification

id	name	type	path
1	image0.png	1	.../.../..
2	image1.png	1	.../.../..

En effectuant ces opérations sur un grand nombre de lignes et plusieurs colonnes, nous parvenons à réduire le temps de transfert des données.

3. Visualisation : Lorsque nous traitons un grand volume de données, cela peut entraîner des visualisations illisibles ou incompréhensibles. Pour remédier à cette situation, plusieurs solutions sont envisageables.

Dans le cas des graphiques en forme de donut ou de camembert, si une part est trop importante, cela peut rendre les plus petites parties invisibles ou plus difficiles à distinguer. Afin de rendre ces graphiques plus lisibles, on peut supprimer la valeur la plus élevée du graphique, ce qui laissera davantage de place pour les parts les plus petites, ou bien regrouper toutes les petites parts en un ensemble unique. Vous trouverez ci-dessous une image qui permettra de mieux comprendre cette idée.



Ensuite, pour éviter de perdre des données, il est possible de mentionner les noms des parties incluses dans la catégorie "Autre" ainsi que leur valeur correspondante.

Dans le cas des graphiques représentant des arbres ou des cartes, il est intéressant de considérer la possibilité de changer l'échelle des données. Cela permet d'éviter une surcharge d'informations qui pourrait rendre la visualisation difficile à comprendre. Vous trouverez ci-dessous un exemple illustrant cette approche.

Dans l'exemple sans mise à l'échelle, la carte devient illisible, tout comme les données en bleu.



Mise à l'échelle réalisée, la carte est lisible ainsi que les données



4. Gestion du Document Object Model (DOM) : Dans le contexte de notre application, la structuration soignée du DOM revêt une importance cruciale. En effet, l'abondance de nœuds au sein du DOM peut compromettre la fluidité de la navigation et détériorer l'expérience utilisateur. Pour pallier à cette problématique, l'utilisation de techniques de windowing s'avère pertinente. Cette stratégie est généralement mise en œuvre sur des éléments itérables

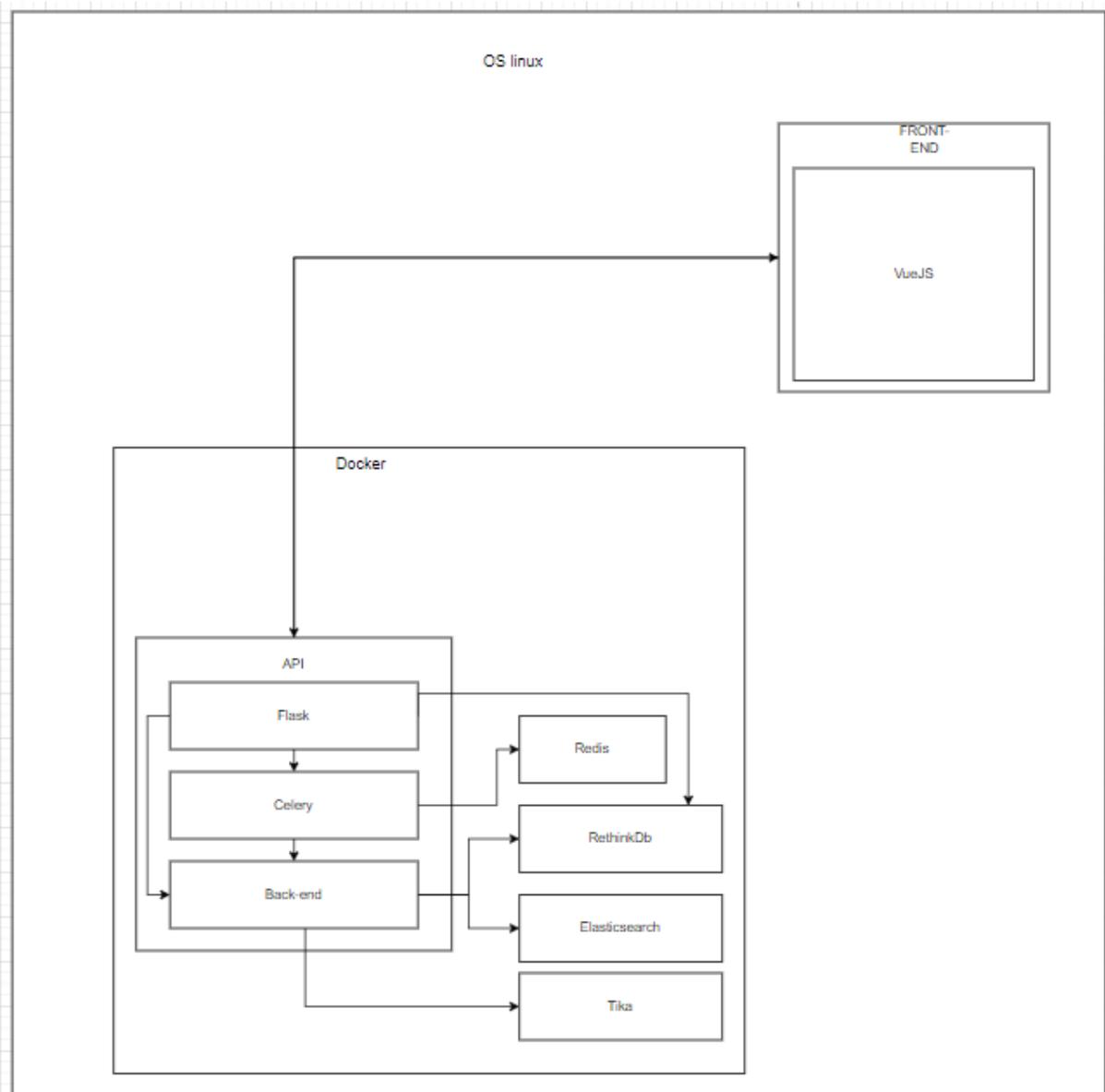
au sein de la page. L'idée fondamentale est d'ajouter au DOM uniquement un nombre prédéterminé d'éléments issus de cette liste itérable, et de permettre le défilement fluide entre eux, sans générer de nouveaux nœuds dans le DOM.

Le concept de windowing permet ainsi de substantiellement réduire la complexité de la structure d'arborescence HTML. De surcroît, il peut être associé au mécanisme de chargement différé (lazy loading), créant une synergie pour une performance accrue. Cette approche ingénieuse offre une solution élégante pour gérer efficacement le contenu et garantir une expérience utilisateur optimale, même lorsque le contenu à afficher est considérable.

## Ma contribution

### Technologique

Pendant la durée de mon stage, j'ai eu l'opportunité de prendre part à la sélection des technologies nécessaires à la mise en place de l'environnement de développement de notre application. Voici les conclusions auxquelles nous sommes parvenus :



Comme vous pouvez le constater sur ce schéma nous avons deux grandes entités qui communiquent entre elles :

- Nous avons intégré une interface Front-End ainsi qu'une API dans notre solution. Cette décision a été prise dans le but de permettre l'accès à la partie métier de l'application par d'autres applications potentielles.
- Nous avons opté pour le framework Vue.js pour le développement de la partie Front-end. Ce choix a été motivé par sa capacité à créer des applications de différentes tailles et complexités, offrant des temps de chargement performants tout en réduisant la nécessité de rafraîchir la page.
- Nous utilisons docker pour simplifier le déploiement de l'application sur diverses machines et systèmes d'exploitation.

- Nous avons opté pour le framework flask afin de construire notre API, car il est développé en Python, tout comme le Backend déjà en place. Cette décision simplifie les interactions entre l'API et le Back-end.
- Nous avons adopté l'utilisation de Celery, une file de tâches asynchrones, dans notre système. Cette approche permet à notre API de gérer des tâches longues sans perturber la fluidité de l'interface Web. Concrètement, lorsque des opérations chronophages sont initiées depuis l'interface Web, au lieu de paralyser l'interface en attendant leur achèvement, Celery prend en charge leur exécution en arrière-plan. Par conséquent, les utilisateurs peuvent continuer à interagir sans subir de ralentissement ou de blocage, ce qui contribue significativement à une expérience utilisateur plus agréable et réactive.
- Nous avons intégré Redis, une base de données clé-valeur, dans notre configuration. Cette base de données est employée par Celery pour enregistrer le statut d'avancement des tâches en cours d'exécution.
- Nous exploitons RethinkDB, une base de données orientée document, au sein de notre système. Cette base de données nous offre la capacité de stocker et d'effectuer des recherches au sein des résultats d'analyses d'une collection donnée.
- Nous avons intégré Elasticsearch, un moteur de recherche full text, pour améliorer la réactivité et l'efficacité de nos recherches de texte au sein d'une collection de données.
- Nous faisons usage de Tika pour extraire le contenu textuel des fichiers lors de l'analyse du disque.

## Technique

De plus, j'ai eu l'occasion de mettre en place différentes fonctionnalités, parmi lesquelles je vais vous présenter celles qui me semblent les plus intéressantes pour le projet, que ce soit du point de vue technique ou pour répondre aux besoins.

- Création de miniatures : dans le cadre d'un projet d'investigation numérique, il était important de permettre aux utilisateurs de visualiser facilement le contenu des fichiers. J'ai donc développé une méthode qui génère des icônes, et si nécessaire, les ajoute à la base de données lors du lancement de l'API. Actuellement, il est possible de créer des icônes pour tous les types d'images, de vidéos, de fichiers PDF et certains fichiers texte. La mise en

œuvre de cette méthode n'est pas très intuitive, car elle implique la gestion de l'asynchrone, du synchrone et de la parallélisation, le tout dans le but d'obtenir des temps de création raisonnables et une écriture efficace dans la base de données. Pour les fichiers pour lesquels il n'est pas possible de créer une icône, une icône par défaut est attribuée en fonction de leur extension.

- Choix du dataset : étant donné qu'il peut y avoir plusieurs datasets disponibles et que leur nombre peut augmenter avec le temps, j'ai mis en place un composant qui permet aux utilisateurs de les rechercher avec un système d'autocomplétion. Cela permet de gagner du temps lors de la recherche ou en cas d'oubli. Les datasets proposés peuvent être publics et ne nécessitent pas de connexion, partagée ou privée, nécessitant une connexion pour vérifier s'ils sont partagés avec l'utilisateur ou s'ils lui appartiennent. Pour rendre cette fonctionnalité de recherche plus ergonomique, j'ai également prévu un affichage des différents datasets possibles, avec des informations telles qu'une miniature, un nom et une taille.
- Affichage de l'arborescence d'un dataset : cette fonctionnalité permet de visualiser l'architecture des médias numériques que nous examinons. Sa mise en place n'est pas très complexe, mais elle présente la particularité d'être récursive, ce qui simplifie grandement sa création.
- Filtrer les données : cette fonctionnalité est essentielle compte tenu de la taille des datasets. Elle permet à l'utilisateur de restreindre l'espace de recherche, ce qui lui fait gagner du temps et améliore la précision des données affichées.
- Visualiser les données : pour faciliter la compréhension des données, plusieurs visualisations sont disponibles. Par exemple, une visualisation expose la fréquence d'utilisation des fichiers dans le temps par rapport au nombre total de fichiers. Une autre visualisation permet de comprendre quels types de fichiers sont les plus présents dans l'outil numérique.
- Parcours des données : une autre fonctionnalité clé est la possibilité de parcourir les fichiers comme si nous avions accès à l'ensemble du dataset. Étant donné la taille importante des datasets, il est crucial de ne pas afficher toutes les données en même temps afin de ne pas ralentir la page. J'ai donc mis en place un défilement infini bidirectionnel qui supprime les données à l'opposé de la direction du défilement lorsque trop de données sont présentes

## Difficultés rencontrées

Migration de l'API vers Python : en effet, étant donné que ce n'est pas le langage que j'utilise habituellement, j'ai nécessité un certain temps d'adaptation pour produire du code de qualité.

Création d'un slide bidirectionnel : la conception de ce composant a été un défi en terme d'implémentation, car j'ai cherché à le rendre générique tout en conservant un code clair et épuré. Pour remédier à cela, j'ai factorisé ce composant en un ensemble de plusieurs composants. La création de plusieurs composants a permis une meilleure lisibilité et possibilité architecturale.

Collaboration en équipe : j'ai été confronté à des défis inhérents à la dynamique de travail en groupe. Pour remédier à cela, nous avons mis en place des réunions régulières ainsi que des moments formels d'échange. Ces mesures m'ont donné l'opportunité de poser des questions et de clarifier les objectifs et les attentes de chacun. Cela m'a permis de garantir une meilleure compréhension des objectifs et des besoins de l'équipe.

## Axes d'amélioration

- Interface pour le lancement des scripts : actuellement, le script d'acquisition des données et leur enrichissement est lancé depuis un terminal. Il serait bénéfique de créer une page dédiée à cette tâche, ce qui simplifierait son exécution et la rendrait accessible à tous.
- Personnalisation des filtres lors du lancement d'une collection : lorsque nous lançons une collection, il serait intéressant d'avoir la possibilité de choisir les filtres souhaités afin d'obtenir des résultats pertinents plus rapidement. Cela permettrait de gagner du temps et d'adapter les résultats en fonction des besoins spécifiques.
- Ajout de nouvelles visualisations : pour enrichir l'expérience utilisateur, il serait pertinent d'ajouter de nouvelles visualisations. Par exemple, un nuage de mots pour les données textuelles, une carte pour la géolocalisation des données, ou encore proposer des alternatives aux visualisations existantes.

Cela permettrait d'explorer les données sous différents angles et d'obtenir de nouvelles perspectives.

- Expansion des filtres pour le traitement des médias : il serait intéressant d'ajouter de nouveaux filtres pour le traitement des images, des vidéos, des textes et des sons. Cela permettrait d'améliorer les fonctionnalités d'analyse et de recherche, en offrant des possibilités de filtrage plus avancées et précises dans ces domaines spécifiques.

En implémentant ces pistes d'amélioration, le projet pourrait bénéficier d'une interface plus conviviale, d'une personnalisation accrue, de nouvelles perspectives visuelles et de fonctionnalités plus avancées pour le traitement de différents types de données.

## Les plus-values du stage

- Gain en autonomie : travailler sur des projets concrets et être responsable de certaines tâches m'a permis de gagner en autonomie. J'ai appris à prendre des décisions et à mener à bien des missions de manière indépendante, ce qui a renforcé ma confiance en mes compétences.
- Perfectionnement de mon niveau général en développement et conception : en travaillant sur le projet, j'ai eu l'occasion d'améliorer mes compétences en développement et en conception. J'ai pu mettre en pratique mes connaissances acquises lors de mes études et les enrichir grâce aux défis rencontrés lors du stage.
- Maîtrise de technologies spécifiques : j'ai eu l'opportunité de travailler avec des technologies spécifiques telles que d3.js, Vue.js et Node.js. Cette expérience m'a permis d'approfondir mes connaissances dans ces domaines et d'acquérir de nouvelles compétences techniques.
- Développement d'un sens critique : travailler sur un projet concret m'a amené à développer mon sens critique. J'ai appris à évaluer et remettre en question mes propres solutions, ainsi que celles de mes collègues, afin de proposer des améliorations et des innovations pertinentes.

- Gain en communication et travail d'équipe : Une partie essentielle de mon stage au Greyc a été la collaboration au sein de l'équipe. Participer à des réunions, échanger des idées avec mes collègues et discuter des défis techniques ont considérablement amélioré mes compétences en communication et ma capacité à travailler en équipe. J'ai appris à articuler mes pensées de manière claire et concise, à écouter activement les opinions des autres et à trouver des solutions collaboratives aux problèmes. Cette expérience m'a montré l'importance cruciale de la communication transparente pour la réussite d'un projet.

## Conclusion

Le stage que j'ai effectué au Greyc a représenté une étape essentielle dans mon parcours de développement professionnel. Cette expérience riche en enseignements m'a permis de concrétiser et de renforcer mes compétences dans les domaines du Web, de la gestion des données et de la visualisation de données.

En travaillant sur des projets concrets au sein de l'équipe, j'ai pu passer de la théorie à la pratique, tout en étant confronté à des défis stimulants. Cela m'a offert l'occasion de perfectionner mes compétences techniques, mais aussi d'affiner mon approche critique envers les solutions que je propose et les décisions que je prends.

L'interaction avec des technologies variées, telles que Vue.js, Node.js, Elasticsearch et d'autres, a contribué à élargir mon horizon professionnel. Cette immersion m'a permis de mieux comprendre les tendances actuelles et les innovations dans le domaine du développement, ainsi que les enjeux concrets auxquels font face les entreprises.

En résumé, ce stage a eu un impact significatif sur mon développement personnel et professionnel. Il m'a conféré une perspective plus complète, me permettant de prendre des décisions plus éclairées et de proposer des solutions pertinentes. Je suis reconnaissant envers l'équipe du Greyc pour cette opportunité et je suis convaincu que les compétences et les connaissances acquises auront une influence positive sur ma future carrière dans le développement.