



RAPPORT DE STAGE

Développeur Full Stack



BARTLOMIEJ GRZADZIEL
DE FÉVRIER À AOÛT 2025

Tutrice de stage : Andréa Croc

Superviseur académique : Guillaume Sanahuja

Entreprise d'accueil : Steamulo (14 Rue Auber, 75009 Paris)

Etablissement : Université Technologique de Compiègne (Rue du Docteur Schweitzer, 60200 Compiègne)

Remerciements

Je souhaite dans un premier temps remercier Andréa Croc, développeuse Full Stack, qui m'a pris en charge et accompagné tout au long de mon stage. La qualité de sa formation, sa bienveillance et son soutien m'ont permis de progresser rapidement et d'enrichir mes compétences techniques tout au long de mon stage.

Je tiens également à exprimer ma gratitude à Antoine Ruchot et Dorothée Schved, membres de l'équipe en charge du projet RNE. Dès mon arrivée, ils ont fait preuve d'une grande bienveillance et ont partagé avec générosité leur expertise dans le domaine du développement web. Leur accueil chaleureux et leur disponibilité ont largement contribué à mon intégration, ainsi qu'à mon épanouissement personnel et professionnel.

Je remercie aussi Sylvain Gourio, le CTO, ainsi que Christian de Nadaillac, CEO de Steamulo, pour m'avoir offert l'opportunité d'effectuer ce stage enrichissant au sein de l'entreprise.

Enfin, je souhaite remercier M. Guillaume Sanahuja, mon superviseur de stage à l'UTC, ainsi que Mme Maryse Loranti, responsable du service des stages, pour leur accompagnement attentif et leur disponibilité tout au long du processus de préparation et de suivi de mon stage.

Sommaire

Résumé technique.....	4
Introduction.....	5
I. Une expérience de plus de 12 ans, l'histoire de Steamulo.....	6
II. L'analyse de mon expérience au sein de l'entreprise.....	9
III. Les missions réalisées durant mon stage.....	15
Conclusion.....	49
Table des matières.....	50
Sitographie.....	51
Glossaire.....	52
Annexe.....	54

Résumé technique

Pour mon TN10, j'ai effectué mon stage de fin d'année chez Steamulo, une entreprise fondée en 2013 par Christian de Nadaillac. Steamulo est une digital & IA factory spécialisée dans la conception et le développement de solutions web, mobiles et d'intelligence artificielle sur mesure. Elle accompagne ses clients de bout en bout, de la phase de conception jusqu'au déploiement opérationnel, grâce à ses différents pôles d'expertise.

Durant ces six mois, j'ai principalement travaillé sur la maintenance évolutive du site du Registre National des Entreprises. J'y ai développé de nouvelles fonctionnalités, tout en améliorant certaines déjà existantes. Mon travail a couvert un large éventail de tâches : de l'enrichissement de la base de données avec de nouvelles collections à l'implémentation de tests unitaires et end-to-end, en passant par l'optimisation de fonctionnalités existantes. J'ai aussi travaillé sur le site du Don Volontaire de la Moelle Osseuse, ce qui m'a permis de découvrir Drupal comme CMS, sur lequel j'ai principalement effectué des ajouts côté front-end. Enfin, j'ai pu découvrir la création de workflows avec n8n, un projet interne de l'entreprise. Ce stage m'a ainsi permis de découvrir et de maîtriser les outils essentiels au quotidien d'un développeur full stack.

Ce rapport présentera d'abord l'entreprise Steamulo, avant de détailler les différentes missions et projets sur lesquels j'ai pu intervenir. Il se conclura par un bilan de mon expérience, mettant en lumière les compétences acquises et les enseignements tirés.

Introduction

Dans le cadre de mon parcours en Programme Grande École à l'Université de Technologie de Compiègne, j'ai eu l'opportunité d'effectuer un stage en tant que développeur Full Stack. Ce stage s'est déroulé au sein de l'entreprise Steamulo, située au 14 rue Auber, à Paris.

J'ai choisi cette entreprise car elle occupe une place importante dans le domaine du développement web et mobile. En tant qu'entreprise de services numériques (ESN), c'est la diversité des projets menés qu'elle a menés qui m'a particulièrement attiré. De l'élaboration d'une application d'assistance pour les personnes malvoyantes dans les gares de la SNCF à la création et à la maintenance du portail Open Data de l'Institut National de la Propriété Industrielle (INPI), Steamulo démontre une capacité à répondre à des besoins variés et exigeants. Pour ma part, j'ai eu la chance de contribuer au projet du Registre National des Entreprises (RNE), développé pour l'INPI, ainsi qu'au développement du site du Don Volontaire de la Moelle Osseuse (DVMO) pour l'Agence de la biomédecine (ABM).

Ces expériences concrètes m'ont permis de découvrir les réalités du métier de développeur Full Stack, tout en approfondissant mes compétences en programmation. J'ai ainsi appris à maîtriser de nouveaux langages, frameworks et CMS, comme TypeScript, ReactJS, PHP Symfony ou encore Drupal. J'ai également été amené à utiliser de nouveaux outils tels que Xdebug pour le débogage.

Ce stage m'a permis d'évoluer dans un environnement de travail stimulant, favorisant la collaboration et les échanges réguliers avec les autres développeurs. J'ai notamment découvert l'organisation agile Scrum avec la mise en place de réunions quotidiennes, qui permettent de suivre l'avancement du projet et de renforcer la coordination au sein de l'équipe. En outre, cette expérience a renforcé ma motivation à poursuivre une carrière dans le développement web et à approfondir mes compétences dans ce domaine.

I. Une expérience de plus de 12 ans, l'histoire de Steamulo

1. Activité, secteur et environnement de l'entreprise

Steamulo est une entreprise de service numérique spécialisée dans le développement de solutions digitales sur mesure. Fondée en 2013, Steamulo se concentre sur l'accompagnement des entreprises dans leur transformation numérique, offrant des services tels que le développement et la création d'application web et mobile, l'hébergement, le conseil en stratégie digitale, ainsi que l'intégration de solutions technologiques adaptées aux besoins spécifiques.

Steamulo possède deux bureaux. Le premier est situé en plein cœur de Paris dans le quartier de l'Opéra, le deuxième dans le quartier des affaires de la Part-Dieu à Lyon. L'équipe regroupe à ce jour 80 experts, composés de développeurs, designers, experts devops, architectes logiciel et consultants. Historiquement, Steamulo a réalisé 250 projets pour plus de 40 clients différents, se positionnant comme un acteur majeur dans le secteur du développement de services sur mesure.

2. Le positionnement de Steamulo, ses services et outils

a. Les pôles au sein de Steamulo

Steamulo se divise en trois pôles d'expertises majeurs : le Pôle Conseil & Transformation, le Pôle Développement & TMA, et le Pôle Hébergement & Devops.

 <p>Conseil & Transformation Transformons vos idées en solutions digitales concrètes, viables et impactantes. Ensemble, concevons vos futures applications web, mobile, IA, ...</p> <p>→</p>	 <p>Développement & TMA Des ingénieurs talentueux, des équipes Agiles, des apps incroyables ! On y trouve : savoir-faire, qualité, exigence, passion, formation, challenge, bienveillance...</p> <p>→</p>	 <p>Hébergement & Devops Automatisation des déploiements, mise en ligne sécurisée, infrastructure robuste et résiliente à la montée en charge, la Cloud factory vous accompagne dans la...</p> <p>→</p>
--	---	--

Figure 1 : Les différents pôles de Steamulo

- Le pôle Conseil & Transformation

Le pôle Conseil se concentre sur l'élaboration de la stratégie digitale, notamment à travers des audits, des analyses et des benchmarks. Ce pôle a pour mission de conseiller les clients sur l'approche à adopter et la stratégie à mettre en place avant la phase de production du service.

Le pôle Design, quant à lui, est axé sur la conception UX/UI des produits, c'est-à-dire sur l'aspect visuel et l'expérience utilisateur des futurs sites et applications mobiles.

- Le pôle Développement & TMA

Ce pôle, auquel j'ai été rattaché durant mon stage, est responsable du développement des applications web et mobiles. L'expertise de Steamulo lui permet de prendre en main tout type de projet, qu'il s'agisse de sites d'envergure, comme le Guichet Unique, ou d'applications plus modestes, comme Objectif Brevet.

- Le pôle Hébergement & Devops

Ce pôle se concentre sur l'hébergement cloud des services développés. La mise en ligne dans le cloud et la sécurisation des données sont prises en charge, et la maintenance des projets est assurée sur la durée. Le logiciel est déployé dans le cloud afin de garantir un fonctionnement 24h/24 et 7j/7, assurant ainsi une disponibilité à tout moment.

Enfin, il existe un pôle à taille réduite que je n'ai pas mentionné : le pôle Service. Constitué de trois développeurs, sa mission est de concevoir et développer des outils internes destinés à optimiser les processus de l'entreprise. Certains de ces outils sont également préparés pour une mise sur le marché ultérieure, visant à étendre l'offre de Steamulo.

b. Les outils de Steamulo

Collabs : Encore en développement, offre une solution sécurisée pour simplifier la gestion du cycle professionnel des collaborateurs au sein de l'entreprise. Ses fonctionnalités clés incluent la gestion des onboarding, des plannings, des congés, du matériel, du reporting, des carrières, des notes de frais ainsi que la préparation simplifiée des paies.

C'est sur cet outil que je devais indiquer chaque jour, en précisant le temps alloué sous la forme de ¼ de jour, les tâches sur lesquelles j'ai travaillé.

Steaminterview : Simplifie la gestion du processus de candidature en fournissant une plateforme sécurisée d'évaluation, avec des tests personnalisables pour le service de recrutement. Elle permet la personnalisation des tests, un outil de CRM, des générations

automatiques de CV et d'autres fonctionnalités modernes comme le scraping de CV via l'intelligence artificielle.

AutoScribe : Simplifie la création des comptes-rendus de réunion grâce à l'IA. En enregistrant des réunions, autoscribe génère automatiquement des résumés adaptés au type de réunion sélectionné. L'application fonctionne en trois étapes : Enregistrement des réunions, puis la retranscription et enfin la génération des comptes rendus.

Opera : Cette application, destinée aux anesthésistes, chirurgiens et infirmiers, a été conçue pour simplifier la gestion des vacations au bloc opératoire. Elle permet la planification hebdomadaire, le suivi des gardes des anesthésistes, la gestion des absences et des remplacements, ainsi qu'un système de notification pour rester informé en temps réel des changements de planning.

Le portail Client : Le portail client offre une interface regroupant l'ensemble des données monitorées des applications d'une entreprise. L'utilisateur peut y contrôler ses serveurs, les alertes, la consommation CPU, ainsi que de nombreuses autres métriques. Parmi les avantages figurent un accès facilité aux contrats, le contrôle des SLA, etc. Une gestion simplifiée des demandes et des changements concernant les applications et leurs infrastructures est également disponible. Enfin, des rapports en temps réel, incluant un indicateur de performance (KPI) relatif à la consommation des ressources CPU, RAM, etc., sont proposés.

c. Les partenaires de Steamulo

Steamulo, ancien membre de l'alliance composée de Cardiweb, Eulidia, Ipsosenso, Coreoz, Reelian, Datatorii et Engeance, a quitté le groupe pour se développer de manière autonome. Toutefois, Steamulo reste en bon contact avec ces différentes entreprises, et en cas de besoin, des échanges de ressources peuvent être effectués pour répondre à une demande. Le principal partenaire de Steamulo Paris est Steamulo Lyon, où les développeurs collaborent sur des projets communs, comme celui du Guichet Unique.

d. Les dernières évolutions de l'entreprise et réglementations

Steamulo prépare la mise en place d'un nouveau pôle, sous forme de laboratoire, dédié aux projets liés à l'intelligence artificielle. L'objectif est d'expérimenter différentes technologies liées à l'intelligence artificielle (IA) et de créer à terme des solutions réutilisables pour les clients, adaptées à plusieurs types de projets.

Ce pôle doit permettre à Steamulo de rester compétitive face à l'évolution rapide du marché et à la montée en puissance de l'IA dans tous les secteurs d'activité.

II. L'analyse de mon expérience au sein de l'entreprise

Dès mon arrivée chez Steamulo, j'ai bénéficié d'un mois complet de formation concernant les technologies utilisées par l'entreprise. Cette formation portait principalement sur les langages de développement : TypeScript avec la bibliothèque ReactJS pour la partie client, et PHP Symfony pour la partie serveur, tout en respectant l'architecture REST.

J'ai également assisté à une vingtaine de midinales. Il s'agit de cours d'une heure durant la pause déjeuner, lors desquels les développeurs de l'entreprise nous forment sur divers sujets, allant des notions générales, comme les bases de GIT, à des sujets plus complexes, tels que l'utilisation de Docker côté DevOps.

Après ce mois de formation, j'ai pu rejoindre les équipes du pôles de développement & TMA afin d'œuvrer sur de vrais projets.

1. Le sujet de stage

En tant que stagiaire développeur Full Stack, mon sujet de stage portait principalement sur la maintenance applicative du site du Registre National des Entreprises (RNE) et du Don Volontaire de la Moelle Osseuse (DVMO). Durant ces six mois, j'ai été amené à réaliser de nombreuses fonctionnalités, qui peuvent être regroupées en deux grandes catégories.

Tout d'abord, les nouvelles fonctionnalités, qui consistent à ajouter de nouvelles possibilités d'interactions sur les sites, comme de nouveaux onglets, de nouvelles routes, mais aussi des traitements côté serveur des données. Pour le RNE, c'est principalement sur ces derniers que j'ai travaillé. En effet, étant donné que le RNE est avant tout une base de données, il existe une centaine d'actions possibles à effectuer sur celle-ci. J'ai notamment contribué à l'archivage des fichiers, un point que je détaillerai plus tard.

Ensuite, lorsque je ne développais pas de nouvelles fonctionnalités, je corrigeais des bugs ou adaptais certains éléments en fonction des modifications demandées. Ces tickets, que je traitais généralement au fil de l'eau, étaient le plus souvent assez rapides à résoudre.

2. L'environnement de travail

Steamulo est situé au 14 rue Auber, dans le 9e arrondissement de Paris, en plein cœur du quartier de l'Opéra. Les locaux, situés au 6^e étage, se composent d'un grand espace de bureaux, de plusieurs salles de réunion ainsi que d'une cuisine. Une deuxième équipe est présente à Lyon, au 49 Boulevard Marius Vivier Merle.

L'équipe de Paris compte environ 60 collaborateurs, tandis que celle de Lyon en regroupe une vingtaine, soit un total d'environ 80 collaborateurs. Cette taille humaine favorise une

communication rapide et claire entre les employés, créant ainsi un fort sentiment de cohésion au sein d'équipes soudées travaillant sur différents projets.

Les locaux comprennent une grande salle de développement où sont installés tous les postes de travail. Il ne faut que 10 secondes pour traverser la salle, ce qui rend la communication entre les employés rapide et efficace. Personnellement, j'étais installé à un poste proche de ma tutrice et des autres développeurs travaillant sur les mêmes projets que moi, ce qui facilitait les échanges.

Par ailleurs, plusieurs salles de réunion sont dédiées aux réunions quotidiennes.

Enfin, une salle à manger équipée d'un frigo et d'un micro-ondes permet de réchauffer ses plats et de déjeuner en groupe. L'entreprise étant située dans un quartier d'affaires, il existe de nombreuses options pour déjeuner dehors.

Mes journées commençaient à 9h30 et se terminaient à 18h, avec une pause déjeuner obligatoire d'une heure et demie pour les stagiaires. Ce créneau me permettait d'aller à la salle de sport le midi, ce qui me libérait du temps le soir.

3. Les méthodes de travail

Les équipes de Steamulo utilisent en général la méthodologie AGILE pour développer leurs projets. Concernant les projets auxquels j'ai participé, la technique Agile Scrum était utilisée, qui se divise en trois rôles :

- **Le Scrum Master** est la personne qui veille à la bonne application de la méthode Scrum au sein de l'équipe. Il facilite la communication, supprime les obstacles qui freinent le travail de l'équipe et s'assure que les réunions quotidiennes ou les rétrospectives se déroulent efficacement. Il accompagne l'équipe vers plus d'autonomie et protège ses membres des distractions extérieures.
- **Le Product Owner** est responsable de la définition des besoins du produit et est le contact direct entre le client et l'équipe de développement. Il recueille les attentes des parties prenantes, rédige les user stories, et priorise les tâches dans le backlog. Il prend les décisions sur ce que l'équipe doit développer en priorité et valide le travail livré à la fin de chaque sprint.
- **L'équipe de développement** qui conçoit le produit.

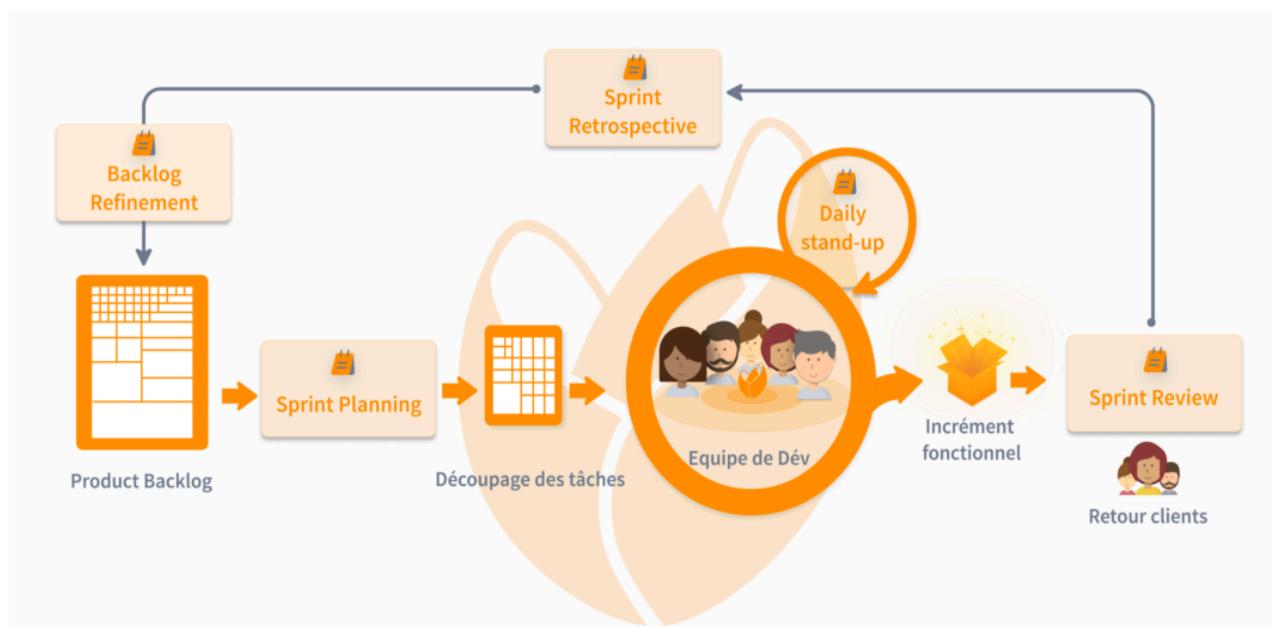


Figure 2 : Illustration de la méthode AGILE SCRUM

L'illustration ci-dessus permet de comprendre en détail le fonctionnement de la méthode. Pour le backlog, nous utilisions Jira, et les développeurs s'attribuaient eux-mêmes les tâches qui arrivaient. Puis, à la fin, un Product Owner vérifiait les fonctionnalités ajoutées, et si c'était bon, elles passaient en préproduction pour que le client teste la fonctionnalité à son tour.

Cependant, sur mes projets, nous n'avions pas de Scrum Master, étant donné que l'équipe de développement était réduite. Nous n'avions pas non plus de période de sprint, étant donné qu'il s'agissait souvent de maintenance, même si nous devions respecter des délais de livraison pour les nouvelles versions demandées. Nous faisions néanmoins des réunions quotidiennes afin de pouvoir échanger avec le client directement par visioconférence et discuter des problématiques et besoins.

4. Les outils utilisés

Au sein de Steamulo, la réutilisation des outils déjà maîtrisés est privilégiée afin d'éviter la gestion de multiples technologies différentes. Cela permet d'améliorer la cohérence des projets et de faciliter la collaboration entre les membres des équipes. Sur ces six mois, j'ai été amené à utiliser plusieurs outils et technologies clé :

Gitlab : Gitlab est un système de contrôle de version décentralisé pour gérer les projets via des dépôts. On distingue les dépôts locaux, présents sur l'ordinateur de l'utilisateur, et les dépôts distants, stockés sur un serveur. Chaque dépôt contient des branches et des copies du projet permettant le développement



et les tests de nouvelles fonctionnalités sans impacter le projet principal. C'est l'outil qu'on utilise chez Steamulo pour la collaboration simultanée sur un projet.

Mattermost : Steamulo utilise le service de chat français Mattermost pour sa communication interne. Ce dernier permet de communiquer facilement en messages privés ou bien au travers de canaux thématiques. Cependant, si Mattermost a été choisi comme outil de communication, c'est notamment pour sa flexibilité avec d'autres outils utilisés par l'équipe. En effet, l'application a été programmé telle que plusieurs canaux sont gérés par des bots qui envoient des messages comme des résultats de jobs lorsqu'une pipeline est passée.



Jira : Pour la gestion de projet, Steamulo utilise Jira, un outil très complet qui permet de suivre l'avancée des tâches de manière claire et organisée. Chaque membre de l'équipe peut voir ce sur quoi les autres travaillent, et les tickets sont mis à jour régulièrement pour refléter l'état d'avancement, allant de "à cadrer" jusqu'à "Validation" finale. C'est l'un des outils que j'ai eu le plus à utiliser, notamment pour parcourir les différents tickets afin de choisir sur quelle fonctionnalité je vais travailler.



MongoCompass : L'interface utilisateur graphique de MongoDB était un outil que j'utilisais aussi quotidiennement. Dans des projets comme le RNE, où la base de données est un élément central de l'application, ce genre d'outil est indispensable pour faciliter les tests et vérifier rapidement si les données sont bien enregistrées ou modifiées comme prévu.



PgAdmin : J'ai également utilisé PgAdmin, l'interface graphique pour PostgreSQL, qui est une base de données relationnelle. J'en faisais un usage similaire à MongoDB Compass, mais adapté à un autre type de données.



Visual Studio Code : Visual Studio Code est un éditeur de code open-source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'autocomplétion, la coloration syntaxique, le débogage, et les commandes git. C'est le principal éditeur de texte que j'ai eu à utiliser durant mon stage, éditeur que j'étais déjà habitué à utiliser à l'UTC.



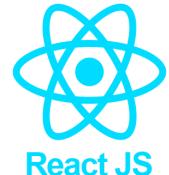
Collabs : Comme précisé précédemment, Collabs est l'outil interne de Steamulo sur lequel je pouvais récupérer mes fiches de paies, faire mon reporting, etc.



Langage de programmation :

Durant le stage, j'ai eu la chance de pouvoir utiliser et apprendre divers langages de programmation, présentés ci-dessous :

TypeScript & ReactJS : TypeScript est un langage de programmation développé par Microsoft, qui étend JavaScript en ajoutant un système de typage statique. Avec nous utilisons ReactJS qui est une bibliothèque JavaScript développée par Meta, qui permet de créer des interfaces utilisateur dynamiques et interactives.



Php Symfony : Symfony est un framework PHP open-source utilisé pour développer des applications web robustes et maintenables. Il repose sur une architecture MVC (Modèle-Vue-Contrôleur) et propose de nombreux composants réutilisables, ce qui en fait un outil très structurant pour les projets back-end.



Cypress : Cypress est un framework JavaScript de “end-to-end testing”, il permet de tester le rendu et les actions utilisateurs de l’application à l'aide de scénarios qui réalisent des actions une par une, simulant un réel utilisateur.



Drupal : Drupal est un CMS (Content Management System) qui aide les utilisateurs à structurer et à gérer leur contenu. Il peut être utilisé pour créer des sites ou des applications web et être intégré à de nombreux sites et services tiers. Il repose sur PHP et utilise une architecture basée sur des modules et des thèmes.



Prismic : Prismic est un CMS en ligne dit “headless”, ce qui signifie qu'il permet de gérer du contenu indépendamment de la manière dont il sera affiché. Il est particulièrement utilisé avec des frameworks comme React ou Next.js pour séparer la gestion du contenu du code de l'application. J'ai été amené à l'utiliser lors de mon basejump afin de créer plus facilement une base de données de livres qu'on pouvait récupérer à travers un endpoint.



5. L'ambiance chez Steamulo

Steamulo étant une entreprise à taille humaine, son effectif réduit crée une ambiance conviviale et accueillante. Par exemple, dès mon arrivée, les stagiaires ont eu droit à un petit déjeuner de bienvenue.

Tous les midis, les équipes se retrouvent autour d'une table pour manger ensemble dans une ambiance conviviale. Du café et des canettes de soda sont en permanence à disposition gratuitement, et des soirées sont organisées une fois par mois autour de différents thèmes comme les jeux vidéo, le karaoké, etc. En général, nous avons aussi une sortie dans un bar le jeudi soir, pour faire du team building. Le vendredi est une journée assez spéciale, puisqu'un Festify est lancé et que chacun peut y contribuer en ajoutant sa propre musique, quand bien même certains sont contre cette initiative étant donné que près de 30% de la playlist est composée de Gims. J'ai même eu la chance de participer au séminaire annuel, organisé chaque année dans un lieu différent. Cette année, nous sommes partis à Palma de Majorque en Espagne, et nous avons eu la chance de faire du quad ainsi qu'une sortie sur un catamaran.

Les membres de l'équipe sont eux aussi très impliqués pour favoriser une cohésion des collaborateurs. Par exemple, à l'occasion de l'anniversaire de l'un des collaborateurs, il est courant qu'il ou elle apporte un assortiment de croissants et de pains au chocolat, ou même un gâteau fait maison, pour partager avec les collègues.

Après avoir présenté l'entreprise et ses différents outils, il s'agira désormais d'aborder les missions que j'ai réalisées au cours de ces six mois.

III. Les missions réalisées durant mon stage

Au cours de mes six mois chez Steamulo, j'ai eu l'opportunité de travailler sur quatre projets variés : d'un projet de formation personnelle, en passant par des projets internes, jusqu'à de véritables missions pour des clients.

1. Le BaseJump

Le BaseJump est un parcours initiatique à travers lequel doit passer chaque nouveau développeur non expérimenté arrivant chez Steamulo, qu'il s'agisse d'un stagiaire ou d'un jeune diplômé. Le but de ce parcours est de nous donner un aperçu des méthodes de travail utilisées ainsi que de nous faire découvrir la stack applicative : Typescript, Java, PHP, Symfony, et l'architecture REST.

Il s'agit en quelque sorte d'un mini plan de formation d'un mois à l'environnement de travail des projets Steamulo. L'objectif est d'intégrer pas à pas le maximum de connaissances dans le temps imparti. Le sujet se présente de la façon suivante :

La ville de Trou-perdu-en-Isère désire réaliser une version web de sa bibliothèque municipale. Lors de la consultation du site, quatre parties doivent être accessibles :

- une partie institutionnelle (contenus éditoriaux) ;
- une partie permettant la consultation et la réservation des livres ;
- une partie « mon compte » affichant les informations de l'utilisateur identifié en front-office ;
- une partie « administration » permettant de gérer les utilisateurs.

Avant de commencer le développement, j'ai élaboré un plan de charge à partir du cahier des charges. Il s'agissait d'un découpage du travail en une liste de tâches auxquelles étaient affectées des durées exprimées en fractions de journée (1 jour, 0,5 jour, etc.). Ce plan était mis à jour quotidiennement.

Cet exercice, en apparence simple, s'est révélé être un véritable casse-tête. N'ayant jamais codé dans ces langages auparavant, je devais estimer la durée de chaque tâche, à savoir la création de la base de données et son alimentation, la création des pages, des API, etc.

Après une semaine, comme illustré sur la figure 3 ci-dessous, j'ai rapidement constaté que la plupart de mes estimations étaient erronées. Soit je sous-estimais largement l'ampleur de la tâche, soit je la terminais au contraire plus rapidement que prévu.

Projet		Début	Fin														
		10/02/2025															
D	Phases du projets	Evalué	Consommé	RAF	Dépass.	10/02	11/02	12/02	13/02	14/02	15/02	11/02	12/02	13/02	14/02	15/02	16/02
D		13,75	18	2	45,45%												
P	1 Initialisation du projet	2,25	6,25	0	177,78%												
E	1.1 Déclaration des routes et vues associées	2,25	2,25	0	0,00%												
T	1.1.1 Barre navigation permettant l'accès à chaque pages afin de construire le squelette du site	0,5	1	0	100,00%												
T	1.1.2 Création de la base de données	0,75	0,75	0	0,00%												
T	1.1.3 Test de la page des livres (juste afficher les livres sans fonctionnalités) afin de bien établir le lien Site-BDD	0,25	0,25	0	0,00%												
T	1.1.4 Footer avec lien vers -> CGU/ règlement -> recap en BDD	0,75	0,25	0	-66,67%												0,25
E	2 Implémentations des fonctionnalités principales	7,5	6,5	0													
E	2.1 Implémentation des utilisateurs	2,5	3	0	20,00%												
T	2.1.1 Interface inscription -> Ajouter un utilisateur	0,5	1	0	100,00%												1
T	2.1.2 Connexion -> Vérification si l'utilisateur existe dans la BDD	0,25	0,75	0	200,00%												0,25, 0,5
T	2.1.3 Implémentation du système de connexion -> Récupérer les infos de l'utilisateur connecté pour enregistrer ses actions	1	0,5	0	-50,00%												0,5
T	2.1.4 Implémentation de la page "Mon compte" avec modifications	0,75	0,75	0	0,00%												0,25, 0,25, 0,25
E	2.2 Implémentation de la page "Nos livres"	3	1,25	0	+58,33%												
T	2.2.1 Lister les livres, barre de recherche, filtres (catégories)	2	1	0	-50,00%												0,25, 0,25, 0,5
T	2.2.2 Création de la page "Fiche livre" : page spécifique au livre sélectionnée	1	0,25	0	-75,00%												0,25
E	2.3 Implémentation des différents types d'utilisateur	2	0	0													
T	2.3.1 Création des pages : Visualiser les utilisateurs + moteur de recherche	1	1,25	0													0,25, 0,25, 0,25
T	2.3.2 Pages des emprunts : Permet d'autoriser les emprunts	1	0,5	0													0,5
P	2 Implémentation complète des pages secondaires	4	0,5	0	-50,00%												
E	2.4	0,5	0,5	0	-50,00%												
T	2.4.2 Implémentation de la page Horaire/Règlement	0,5	0,5	0	0,00%												0,5
T	2.1.3 Implémentation de la page CGU	0,5	0,5	0													0,5
P	3 Vérification et améliorations	3	8,75	2	258,33%												
E	3.1 CSS	1	3,75	0	275,00%												
T	3.1.1 Amélioration de l'UI	1	3,75	0	275,00%												0,5, 0,5
E	3.2 Autres corrections	2	5	0	150,00%												
T	3.2.1 Débuggage, Amélioration sur retour	2	5	2	250,00%												0,25, 0,5, 0,5, 0,25
F						1	1	0,75	1	1	1	1	1	1	1	1	1

Figure 3 : Plan de charges de mon BaseJump

Le stack technique imposé était l'API REST. Une API REST est une interface de programmation d'application qui utilise les principes de l'architecture orientée ressources pour permettre la communication entre différentes applications via le protocole HTTP. Elle repose sur plusieurs principes clés, notamment l'utilisation des verbes HTTP standard tels que GET, POST, PUT et DELETE pour effectuer des opérations sur les ressources, qui sont exposées sous forme de route comme livres/1. Ainsi, si on veut supprimer le livre qui a pour identifiant 1, il suffit d'appeler cette API avec la méthode DELETE, et si on veut simplement le récupérer, on l'appelle avec la méthode GET.

La figure 4 ci-dessous illustre à la fois la déclaration et le fonctionnement d'une API REST, avec des échanges exclusivement en JSON, et la déclaration des routes sous format de ressources.

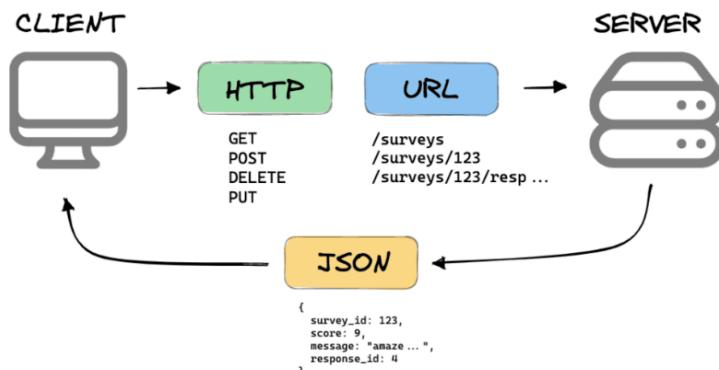


Figure 4 : Schéma du fonctionnement de l'API REST

A noter que les appels sont stateless, de manière qu'un appel ne peut en aucun cas dépendre d'un autre. Toutes les informations nécessaires doivent être présentes à chaque requête.

a. La réalisation du BaseJump

Pour ce projet, j'ai adopté une approche dite transversale, en développant simultanément la partie front-end et back-end. Cette méthode m'a permis de mieux appréhender le fonctionnement global d'un site web, en facilitant la compréhension des interactions entre les différentes couches de l'application.

Lors de mon premier stage (TN09), j'avais déjà eu l'occasion de développer certaines fonctionnalités, mais principalement côté front, en HTML et JavaScript. Par la suite, l'UV SR10 m'a permis de découvrir l'ensemble du processus de conception web, depuis la réalisation de maquettes et la création de la base de données, jusqu'à la mise en ligne du site final.

Huit mois plus tard, je me suis retrouvé de nouveau face à une tâche similaire, cette fois-ci avec de nouvelles technologies que je n'avais encore jamais utilisées : PHP avec le framework Symfony pour le back-end, et ReactJS pour le front-end.

b. La structuration du front-end

Pour le projet BaseJump, j'ai dû suivre une architecture simple que j'avais déjà eu l'occasion d'utiliser à l'UTC. Celle-ci consiste à diviser le code en plusieurs dossiers : services, interfaces, pages, assets, constantes, et components. Cette organisation permet une meilleure lisibilité et une meilleure gestion du projet sur le long terme.

Services	Contiennent la logique métier et les appels aux API
Interfaces	Définissent les types et structures des données utilisées
Pages	Regroupent les vues principales accessibles dans l'application
Assets	Regroupe les ressources statiques (images, styles, etc.)
Constantes	Stocke les valeurs fixes utilisées dans l'application (par exemple, les URLs, les clés, etc.)
Components	Contient les composants réutilisables dans les pages

Fonctionnalités :

Étant donné qu'il s'agit d'une bibliothèque, plusieurs fonctionnalités étaient indispensables : recherche par nom de livre, pagination des résultats, filtrage des ouvrages selon leurs catégories et leur disponibilité. L'image ci-dessous présente le rendu final de la liste des livres avec les différentes options de filtrage accessibles à l'utilisateur.

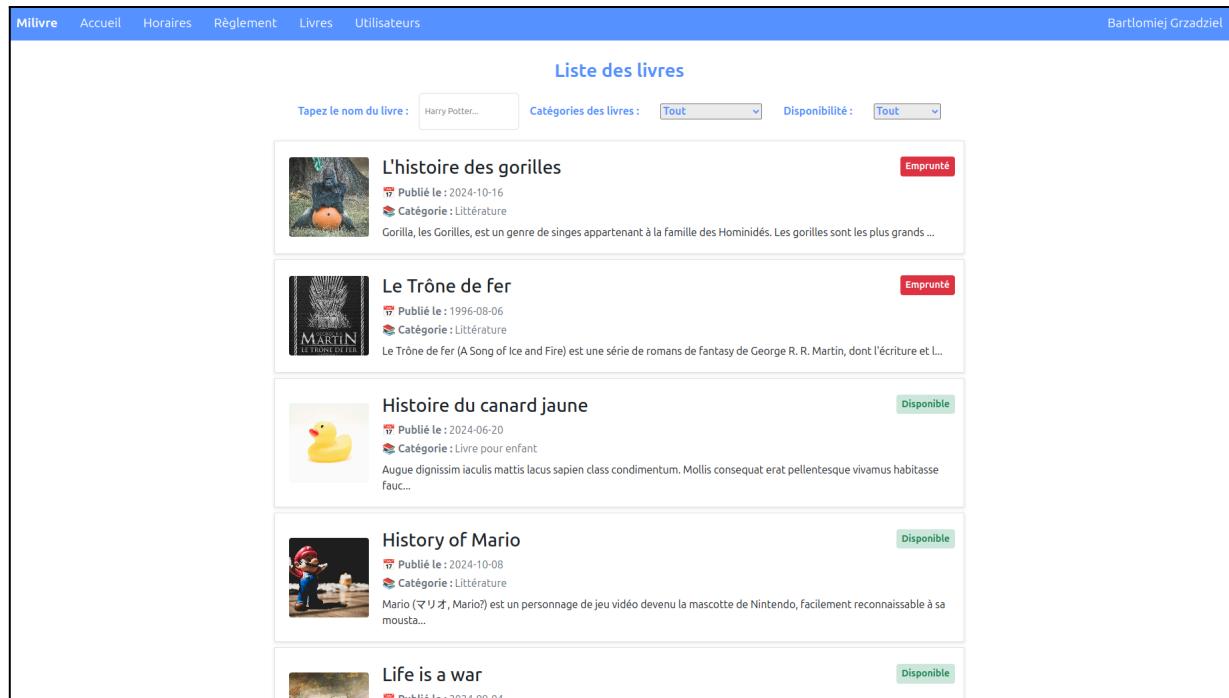
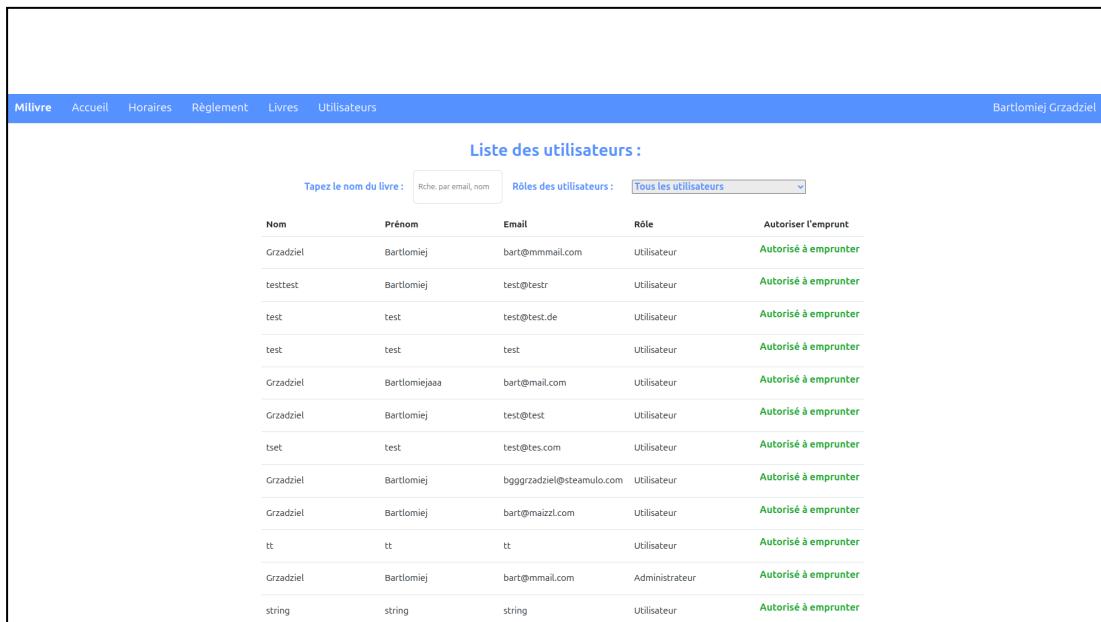


Figure 6 : Page de la liste des livres

Une page utilisateur permet à chaque usager de modifier ses informations personnelles et de consulter facilement la liste de ses emprunts. Par ailleurs, la figure 7 représente la page dédiée aux administrateurs qui permet de gérer les droits d'emprunt des utilisateurs et de trier la liste en fonction de leurs permissions.



Nom	Prénom	Email	Rôle	Autoriser l'emprunt
Grzadziel	Bartłomiej	bart@mmmail.com	Utilisateur	Autorisé à emprunter
testtest	Bartłomiej	test@testr	Utilisateur	Autorisé à emprunter
test	test	test@test.de	Utilisateur	Autorisé à emprunter
Grzadziel	Bartłomiejaaa	bart@mail.com	Utilisateur	Autorisé à emprunter
Grzadziel	Bartłomiej	test@test	Utilisateur	Autorisé à emprunter
tset	test	test@tes.com	Utilisateur	Autorisé à emprunter
Grzadziel	Bartłomiej	bgggrzadziel@steamulo.com	Utilisateur	Autorisé à emprunter
Grzadziel	Bartłomiej	bart@maizzl.com	Utilisateur	Autorisé à emprunter
tt	tt	tt	Utilisateur	Autorisé à emprunter
Grzadziel	Bartłomiej	bart@mmail.com	Administrateur	Autorisé à emprunter
string	string	string	Utilisateur	Autorisé à emprunter

Figure 7 : Page de la liste des utilisateurs

La réalisation de ces différentes pages m'a permis d'améliorer mes compétences en design web. Etant donné qu'aucune maquette n'était imposée, nous avons pu laisser libre cours à notre créativité. J'ai personnellement opté pour un style épuré aux tons bleus, même si, rétrospectivement, cette couleur ne correspond pas forcément à l'image classique d'une bibliothèque.

c. Structuration du back-end

Pour le back-end, j'ai procédé de manière similaire en divisant l'architecture en dossiers : constantes, controllers, DTOs, entités, repositories, et services :

Constantes	Stockent les valeurs fixes (URLs, clés API, messages d'erreur, configurations)
Controllers	Gèrent les requêtes HTTP, les autorisations par rôle et orchestrent les appels aux services
DTOs	Définissent les structures de données pour les échanges API (validation des entrées, formatage des sorties)
Entités	Représentent le modèle de données et la structure des tables PostgreSQL
Repositories	Permet l'accès aux données avec les opérations CRUD et requêtes personnalisées
Services	Contiennent la logique métier et les règles de gestion

Le back-end est plus complexe puisqu'il constitue la colonne vertébrale du site. J'ai procédé progressivement en commençant par définir les routes. Une fois celles-ci opérationnelles, j'ai pu mettre en place les rôles et droits afin de restreindre l'accès à certaines routes selon le rôle attribué à l'utilisateur.

Pour la gestion des utilisateurs et des emprunts, j'ai utilisé une base de données PostgreSQL, visualisée et administrée via pgAdmin. Côté serveur, j'ai bénéficié du mapping objet-relationnel (ORM) offert par Doctrine, intégré à PHP Symfony, ce qui m'a permis de manipuler facilement les données sans avoir à écrire manuellement les requêtes SQL.

Concernant les livres, Prismic a été utilisé pour créer une API permettant d'alimenter directement la base de données depuis le site. Cela permettait l'ajout et la modification rapide des livres sans nécessiter d'intervention sur le code, pour que des non développeurs puissent aussi contribuer au site.

Enfin, j'ai mis en place une documentation interactive avec Swagger. Il s'agit d'un endpoint listant toutes les routes disponibles, avec le détail des paramètres attendus ainsi que des réponses possibles, comme illustré sur la figure 8 ci-dessous.

The screenshot shows the Basejump Symfony API documentation generated by Swagger. The interface is organized into sections:

- Authentification**: Contains a single endpoint: `POST /api/login` to obtain a JWT token.
- Books**: Contains two endpoints: `GET /api/books` to return all books from Prismic, and `GET /api/books/{id}` to return a specific book through its ID.
- Loans**: Contains five endpoints: `GET /api/loans` to return all loans from the database, `POST /api/loans` to add a new loan to the database, `GET /api/loans/books/{bookId}/is-borrowed` to check if a specific book is borrowed, `GET /api/loans/users/{userId}` to return all loans of a specific user, and `DELETE /api/loans/{bookId}` to delete a loan through its book ID.
- PagesInfos**: Contains two endpoints: `GET /api/pagesinfos` to return all pages from Prismic, and `GET /api/pagesinfos/{pageName}` to return a page through its name.
- Users**: Contains five endpoints: `GET /api/users` to return all users from the database, `POST /api/users` to add a new user to the database, `GET /api/users/{id}` to return a specific user through its ID, `PUT /api/users/{id}` to modify a connected user, and `PATCH /api/users/{id}` to authorize a loan for a user.

Figure 8 : Documentation des API avec swagger

Une fois le Basejump terminé, environ 1 mois après mon arrivée, j'ai pu rejoindre l'équipe chargée de la maintenance évolutive du Registre National des Entreprises.

2. Le Registre National des Entreprises - TMA

Le **Répertoire National des Entreprises** (RNE) est une base de données centralisée qui regroupe l'ensemble des informations légales, administratives et économiques relatives aux entreprises enregistrées en France. Elle a pour objectif de simplifier l'accès et la manipulation des données fiables et à jour sur les entreprises. Elle a été initialement conçue pour regrouper les données de différents registres tels que l'Institut National de la Statistique et des Etudes Economiques (INSEE), le Répertoire National des Métiers (RNM), le Registre des actifs agricoles (RAA),... pour centraliser les données à un seul endroit.

Le **Guichet Unique**, mis en place par l'Institut National de la Propriété Industrielle (INPI), est un portail numérique centralisé qui est devenu le principal point d'entrée pour créer, modifier et corriger une entreprise, ainsi devenant la principale source d'alimentation du RNE. Lorsqu'une entreprise effectue une déclaration via le Guichet Unique, les informations saisies sont automatiquement transmises au RNE, qui devient la source officielle de ces données. Cette liaison entre le guichet unique et le RNE permet d'assurer la cohérence et la mise à jour en temps réel des informations sur les entreprises, tout en réduisant la complexité des formalités pour les entrepreneurs.

Le site du RNE est donc une interface permettant directement d'interagir avec la base de données titanique. Le site comprend un front-office pour les clients qui peuvent uniquement consulter les données d'entreprises, et un back-office qui permet de gérer les utilisateurs de la plateforme ainsi que de réaliser des actions de modifications avancées sur les données d'entreprises.

L'image suivante permet de représenter les différents flux entrants et sortants du RNE.

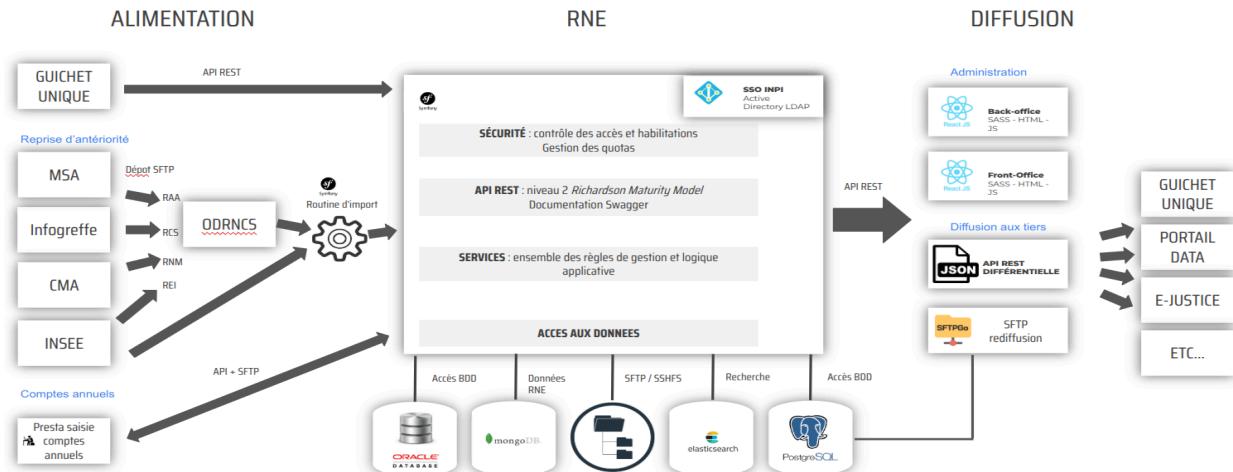


Figure 9 : Alimentation et Diffusion à travers le RNE

Le RNE est alimenté par diverses sources, dont la principale est le Guichet Unique, qui reçoit plus de 100 000 documents chaque jour. Pour mettre à jour les informations des entreprises, le RNE récupère également des formalités provenant d'organismes tels que l'INSEE. Par ailleurs, les comptes annuels, un ensemble de documents comptables que chaque entreprise doit fournir annuellement, permettent de présenter leur situation financière.

Afin de gérer ce volume et cette diversité de données, le RNE utilise plusieurs outils adaptés : MongoDB stocke l'ensemble des documents, PostgreSQL gère les utilisateurs, leurs rôles et plus globalement les relations entre différentes données, et Elasticsearch optimise la recherche grâce à l'indexation des données. La diffusion des données s'organise en plusieurs niveaux :

- L'administration, qui comprend deux sites distincts pour un même projet : un back-office réservé aux collaborateurs internes de l'INPI, permettant des actions avancées selon les rôles des utilisateurs, et un front-office accessible à tout détenteur d'entreprise pour des opérations simples comme la recherche.
- La diffusion aux tiers, qui concerne la transmission des données à différents clients, dont notamment le Guichet Unique.

Après avoir expliqué le fonctionnement du RNE en tant que base de données, il s'agira désormais de présenter les différentes fonctionnalités que j'ai réalisées dans le cadre de ce projet.

a. Le front-end

Rejoindre un tel projet n'a pas été simple. En effet, l'envergure du site et du code le rend difficile à prendre en main, et demande un temps conséquent de lecture et compréhension.

Tout d'abord, le front-end est structuré d'une manière que je ne connaissais pas : l'architecture Atomique. Elle divise la structure du code en 4 niveaux :

- Atomes : les éléments les plus simples, comme des boutons ou des champs de saisie
- Molécules : des combinaisons d'atomes formant une unité fonctionnelle (ex : un champ de recherche avec son bouton)
- Organismes : des groupes plus complexes composés de molécules et d'atomes (ex : un header avec logo, menu, barre de recherche)
- Vues : des pages entières construites à partir des éléments précédents

Le principe est simple : découper le code en briques réutilisables, pour un développement plus modulaire et bien plus facile à maintenir. Honnêtement, j'étais un peu sceptique au début. L'approche me paraissait trop lourde avec une courbe d'apprentissage trop raide pour ce qu'elle apporte. Mais à force d'ajouter de nouvelles fonctionnalités, j'ai compris tout l'intérêt de cette structure : le code reste cohérent, réutilisable et les modifications deviennent bien plus rapides à gérer.

L'illustration ci-dessous, qui illustre l'assemblage modulaire d'une page Instagram, permet de mieux comprendre le fonctionnement de l'architecture atomique :

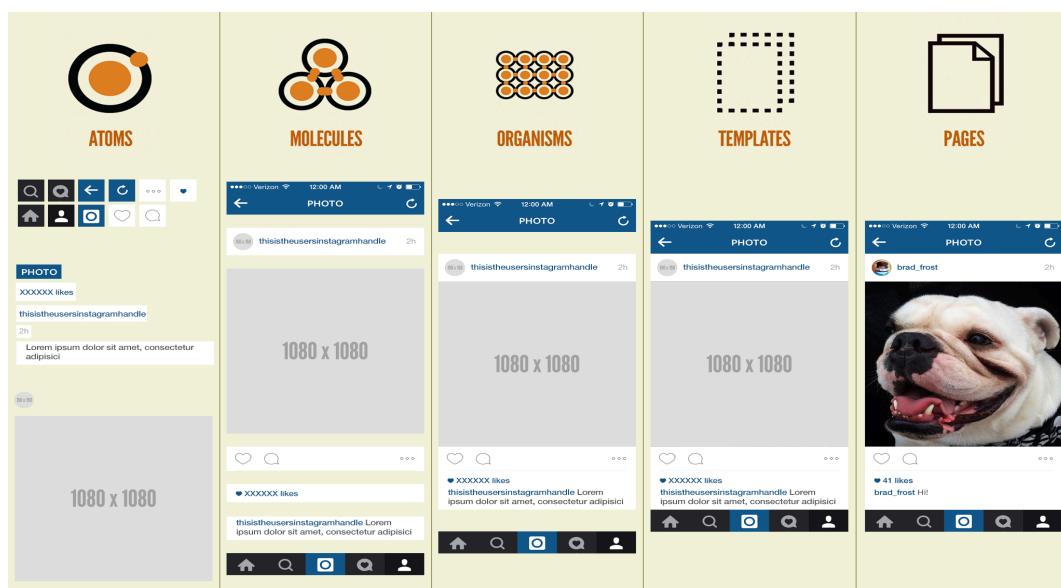


Figure 10 : Représentation de l'architecture atomique

Cette architecture devient particulièrement pertinente dans des projets tels que le RNE où les pages présentent une forte similitude et partagent de nombreux éléments entre elles, comme l'illustre la figure 11 ci-dessous.

The screenshot shows the INPI REGISTRE NATIONAL DES ENTREPRISES website. The top navigation bar includes links for 'Rechercher une entreprise doublon', 'Rechercher une association', 'Rechercher une liasse', 'Gestion des utilisateurs', 'Gestion des internes INPI', 'Gestion des entreprises', 'Gestion des super groupes', 'Statistiques d'import', 'Statistiques de purge', 'Gestion des clés API', and 'Onglet Super Admin'. The user is logged in as 'Admin Steamulo'. The main content area is titled 'Consulter les entreprises' and features a search bar with placeholder text 'Rechercher par Siren, nom ou dénomination sociale ...' and a 'Rechercher' button. Below the search bar is a link 'Effectuer une recherche avancée >'. A message indicates '22 éléments trouvés'. Three search results are listed in a grid:

	NOM	SIREN	N° DE GESTION	DATE DE DÉPÔT	NUMÉRO NATIONAL	TYPE DE PERSONNE
	GENERALI VIE	539637231		23/12/1960		Morale
	GENERALI VIE	602062482		23/12/1960		Morale
	OSCAR TAVOLARO	434783767		01/01/1991		Physique

Figure 11 : Capture d'écran du site du RNE

Chaque page réutilise le même style de police, de bouton, et la même couleur. Ainsi, la réutilisation est clé et permet de rendre plus rapide l'implémentation de nouvelles interfaces.

b. Tests E2E

Pour m'immerger rapidement dans le code et comprendre l'architecture du projet, j'ai débuté par l'implémentation de tests avec Cypress. Cette approche m'a permis de prendre en main le projet en analysant d'abord le fonctionnement de chaque onglet et bouton du site, puis en les retrouvant dans le code afin de pouvoir les tester de bout en bout. J'ai utilisé le framework Cypress, reconnu pour ce type de tests, et également maîtrisé par l'entreprise.

Les tests end-to-end (E2E) permettent de vérifier le bon fonctionnement des fonctionnalités du site en simulant un parcours utilisateur complet. L'objectif de cet ajout était d'avoir des tests pour le côté front-end du site et de lancer ces tests à chaque fois avant de migrer de nouvelles branches dans la version de recette. Les tests ont donc été intégrés aux pipeline CI/CD directement pour automatiser le processus de test.

J'ai ainsi développé un scénario utilisateur d'environ 3 minutes, couvrant les actions principales : navigation sur les boutons, appels aux API, création de compte, ainsi que la vérification des restrictions selon les rôles utilisateurs. Chaque test était conçu pour ne laisser aucune trace, notamment par la suppression systématique des comptes créés, afin de ne pas polluer la base de données.

c. Le back-end

Le back-end représente un niveau d'importance supérieur dans le projet, car le RNE s'appuie principalement sur ses fonctionnalités côté serveur plutôt que sur son interface actuelle. Étant avant tout une interface pour interagir avec la base de données, le système compte des centaines de routes, avec de nouvelles fonctionnalités à développer en continu.

J'ai donc commencé par des tickets simples, avant de progresser vers des tâches plus complexes. En raison du caractère évolutif et de maintenance du projet, j'ai eu l'opportunité de toucher à quasiment tous les aspects du développement back-end. Voici un aperçu des principales fonctionnalités sur lesquelles j'ai travaillé :

- ***Création de commande de rectification sur un json***

L'INPI constate parfois que des entreprises stockées au RNE sont erronées et ainsi que certaines données ne sont pas valides, fiables. Pour y remédier, des commandes de rectification sont créées afin de parcourir ces entreprises et modifier les valeurs concernées. Durant mon stage, j'ai notamment développé une commande qui mettait à jour la date de fermeture des établissements secondaires d'une entreprise principale lorsqu'elle possédait elle-même une date de fermeture.

Ce travail m'a permis de mieux maîtriser le fonctionnement de PHP tout en contribuant concrètement au projet. Par ailleurs, chaque commande ajoutée devait être accompagnée d'un test unitaire garantissant son bon fonctionnement. En effet, le client impose un taux de couverture de code minimum de 80 %. Ainsi, chaque nouvelle fonctionnalité back-end est systématiquement couverte par des tests unitaires pour maintenir cette exigence.

Ces tests ont pour but de tester la nouvelle fonctionnalité implémentée et que son comportement attendu soit respecté. Souvent, l'écriture de ces tests prend plus de temps que l'implémentation de la fonctionnalité même. Ces derniers se dévoilent surtout essentiels puisqu'ils sont lancés à chaque merge request, qui est une demande d'intégrer un nouveau code à la branche principale, et qu'ils permettent de tester les nouvelles fonctionnalités implémentées et vérifier que ces dernières ne cassent pas le fonctionnement d'autres éléments. Nous pouvons voir directement sur Jenkins si ces dernières ont réussies ou pas et voir ce qui a potentiellement cassé. L'ensemble des tests prennent trente minutes

à s'exécuter ce qui témoigne déjà de la taille colossale du code. L'image suivante montre l'exécution des tests sur une pipeline Jenkins.

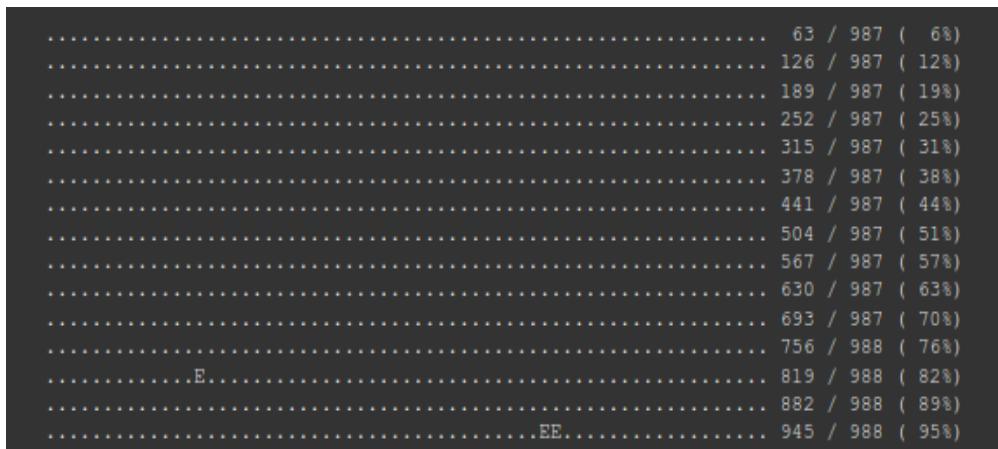


Figure 12 : Représentation des tests unitaires

Chaque point correspond à un test : Un point signifie que le test a réussi, E est pour Error et F pour Fail.

Error généralement quand il y'a tout simplement un bug dans le code, une erreur qui casse l'exécution.

Failures correspond à un échec d'un test : un test qui devait renvoyer un certain résultat mais qui en a renvoyé un autre.

En outre, concernant Jenkins, il est essentiel d'approfondir un autre aspect crucial du développement back-end : les pipelines CI/CD.

Un pipeline d'intégration et de déploiement continu (CI/CD) est une série d'étapes précises à réaliser en vue de la distribution d'une nouvelle version d'un logiciel. Les pipelines CI/CD permettent d'améliorer la distribution des logiciels tout au long du cycle de développement via l'automatisation. Elles permettent en quelque sorte de vérifier avant que le commit passe, que tout le code est exécutable, qu'il respecte les règles de code, qu'il ne présente pas d'erreurs et que tous les tests unitaires soient passés. Ces derniers ont été en quelque sorte mon "enfer" à moi lors du début du stage. La majorité de mes premiers commit ne passaient pas et présentaient des erreurs. J'ai mis bien du temps à maîtriser le sujet et comprendre comment les éviter.

Comme illustré sur la figure 13 ci-dessous, le job qui parcourt la pipeline doit valider plusieurs étapes, et ici après avoir réussi à passer les deux premières étapes, le job casse à l'exécution des tests unitaires, signifiant que la fonctionnalité implémentée en casse une autre.



Figure 13 : Différentes étapes d'une pipeline Jenkins

Après avoir présenté le fonctionnement du site côté client et côté serveur, je vais maintenant revenir sur deux fonctionnalités majeures que j'ai réalisées.

d. L'archivage des documents

Le RNE traite environ 100 000 documents par jour. Ces documents se décomposent sous trois formats :

- Imr -> formalités : démarches administratives permettant la création, la modification ou la cessation d'une entreprise
- Bil -> Comptes annuels : documents comptables obligatoires que les entreprises déposent chaque année
- Act -> Actes :documents juridiques officiels liés à la vie de l'entreprise

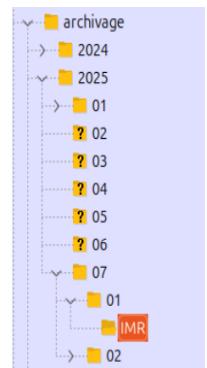
Il y a un besoin de conserver ces documents sur le long terme pour des raisons légales, réglementaires et métiers. Ainsi, un système d'archivage électronique sécurisé VITAM a été intégré, VITAM étant le système d'archivage officiel du gouvernement.

Plus précisément, l'archivage consiste à récupérer chaque jour les documents envoyés par le Guichet Unique et à les stocker sur un serveur SFTP nommé SFTP archivage. Ce dernier repose sur le protocole SSH (Secure Shell), qui permet de transférer des fichiers de manière sécurisée sur un réseau. Le SFTP chiffre les données ainsi que l'authentification, empêchant toute interception ou modification pendant le transfert. Cela garantit la confidentialité, l'intégrité et l'authentification des fichiers, ce qui est indispensable pour des documents sensibles liés aux entreprises françaises.

Les formalités à archiver sont classées en catégorie référencée par le trigramme CCC dans la suite du document. Chaque catégorie correspond à un type de procédure différent, lui-même en fonction du type de formalité.

Catégorie	Type de procédure	Type de formalité
ACT	Actes isolés	A : Acte
BIL	Comptes & Bilans	B-S : Social B-C : Consolidé
IMR	Immatriculation	C : Création M : Modification R : Cessation Y : Correction Z : Complétion

Ainsi, dès qu'une formalité est récupérée depuis le Guichet Unique, elle est automatiquement stockée dans le répertoire interne du système d'archivage du RNE, à l'adresse suivante : `\data\rne\archives\AAAA-MM-JJ`. Chaque fichier compressé (ZIP) déposé dans ce dossier donne lieu à la création d'une entrée dans la table d'archivage de la base PostgreSQL, précisant notamment : la date d'intégration, le chemin dans le SFTP, les métadonnées à indexer, et le statut de l'archivage.



Les fichiers archivés sur le SFTP interne doivent ensuite être transférés vers le SFTP du connecteur. Le SFTP connecteur fonctionne de la même manière que le SFTP archivage, mais est destiné à l'intermédiaire Vitam qui s'occupe de récupérer les archives déposées par le RNE pour qu'elle puisse les traiter et les archiver de leurs côtés.

Un CRON tourne tous les matins à 1h du matin effectue cette opération de téléversement de documents. Les archives sont déposées dans un dossier correspondant à la date d'intégration en base, et le statut associé est mis à jour dans la base de données.

Le schéma macro des flux ci-dessous permet d'illustrer simplement le cheminement des documents, depuis le RNE vers le serveur SFTP archivage en rouge, jusqu'à leur vérification par Vitam et le retour après traitement.

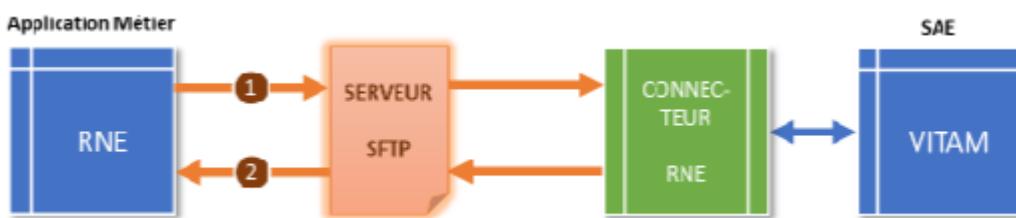


Figure 14 : Schéma macro des flux de l'archivage

Après avoir mis en place la première version du système d'archivage, il m'a été demandé une deuxième version (V2). J'ai eu l'opportunité d'y contribuer activement, et je vais donc

présenter les principales évolutions apportées dans cette nouvelle version et expliquer en quoi elles améliorent le fonctionnement initial.

1ère étape : Archiver documents entrants

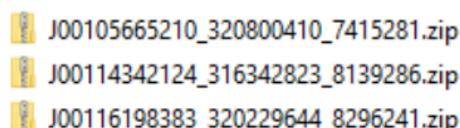
Ma tutrice s'est occupée de la première partie de la mise à jour du système d'archivage qui consiste à suivre une convention pour le nommage des fichiers ZIP entrants sur l'archivage.

Selon son point d'arrivée, chaque archive zip doit désormais être catégorisée et nommée selon son type :

L'archive zip d'une formalité est nommée comme suit :

- Archives IMR : <liasseNumber>_<siren>_<**archiveRneId**>.zip
- Archives ACT : <acteNumber>_<siren>_<**archiveRneId**>.zip
- Archives BIL : <liasseNumber>_<siren>_<**archiveRneId**>.zip

Exemple d'archives IMR :



2ème étape : Crédation et dépôt des tar journaliers sur le SFTP Connecteur

La seconde étape consiste à transférer les archives du jour sur le serveur SFTP Archivage. Pour cela, j'ai codé une commande qui permet de récupérer toutes les archives de la veille afin d'en créer un tar par type d'archive contenant un ensemble de zip et un fichier entries.json listant les formalités présentes dans le tar. Le fichier tar se construit comme ci :

lotId = <CCC>_<AAAAMMMJJ>_<NNNNNNN>.tar, avec :

- CCC la catégorie d'archives
- AAAAMMMJJ, le jour courant
- NNNNNNN, numéro incrémental basé sur une séquence quotidienne par catégorie

Les lots sont déposés dans l'arborescence *Entrants/{type d'archive}/AAAA-MM-JJ/* correspondant à la date du jour de l'entrée en base postgres. Lors du dépôt sur le connecteur, il faut toujours mettre à jour la base de données avec les états d'avancement du processus d'archivage.

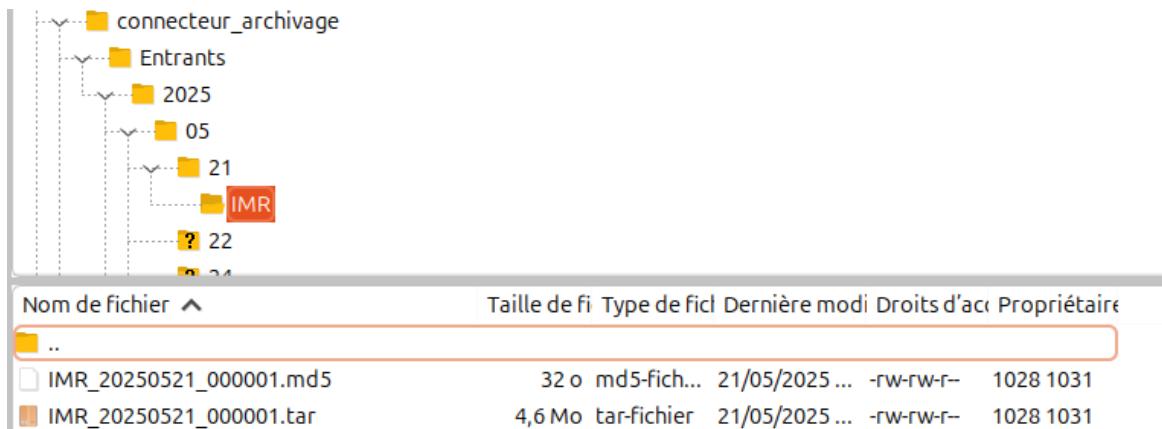


Figure 15 : Arborescence des tar entrants

A la création d'un tar, nous ajoutons une ligne dans la table archivage_lot avec les données suivantes :

- id du lot
- path sur le connecteur
- date d'envoi
- statut (CREATED à la création du tar puis SENT lors du dépôt du tar sur le SFTP Connecteur)

Cet ajout améliore significativement le traitement des fichiers entrants. La catégorisation par type dans chaque archive TAR permet non seulement une meilleure lisibilité des contenus, mais elle est surtout essentielle pour les étapes suivantes du processus d'archivage. En effet, cette classification conditionne les actions spécifiques qui seront appliquées à chaque catégorie de documents lors de leur traitement ultérieur.

3eme étape : Traitement des retours de Vitam

Après l'analyse des documents effectués par VITAM, ces derniers nous renvoient des fichiers xml, qui sont des accusés d'archivage qu'ils vont déposer sur le SFTP connecteur dans le dossier Retours/{type d'archive}/AAAAMMMJJ

Une structure globale est utilisée dans tous les cas des fichiers xml, qui se présente de la façon présentée ci-dessous :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<retour>
    <date>2025-03-17T10:01:957</date>
    <statut>OK</statut>
    <entites>
        <typeEntite identifiant="identifiant élément">
            ...
        </typeEntite>
    </entites>
</retour>
```

Figure 16 : Structure d'un fichier xml

Le statut de l'élément retour définit le statut global de toutes les entités. Les valeurs possibles sont :

- **OK** : Les éléments sont archivés dans VITAM
- **KO** : Une erreur a été rencontrée lors du traitement des entités.

Le statut précis de l'erreur est porté par la balise statut des éléments entités.

Vitam nous renvoie deux type d'accusés, le “type lot” et le “type formalité” :

- **L'accusé de type Lot** : Cet accusé n'est possible qu'en cas d'erreur sur le traitement d'un fichier tar en entrée.
- **Les accusés de type formalité**, se décomposant en deux sous accusés : erreur et succès. L'accusé de type Succès qui répertorie l'ensemble des documents traités avec succès, et l'accusé de type Erreur, qui liste tous les documents en échec de traitement en précisant pour chacun un code erreur ainsi qu'une description détaillée, comme illustré dans la figure ci-dessous :

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<retour>
    <date>2025-03-17T10:01:957</date>
    <statut>KO</statut>
    <entites>
        <formalite identifiant="1234567890">
            <statut>202</statut>
            <message>le fichier KBIS_2024_12_11-1.pdf est absent</message>
        </formalite>
        <formalite identifiant="1234567891">
            <statut>203</statut>
            <message>le fichier KBIS_2024_12_11-2.xls n'est pas listé</message>
        </formalite>
    </entites>
</retour>
```

Figure 17 : Structure d'un fichier xml de type formalité

Ayant travaillé sur le traitement de ces fichiers xml, l'arborescence du système d'archivage a évolué :

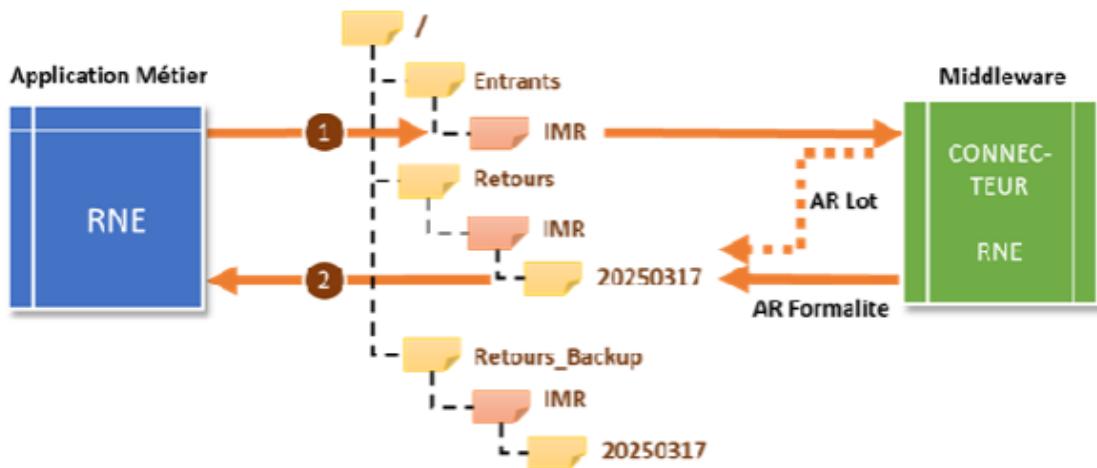


Figure 18 : Schéma macro des flux pour la V2 de l'archivage

Le nouveau processus demande désormais de mettre à jour la base de données à partir des fichiers xml renvoyés par Vitam. J'ai donc appris à récupérer des données depuis des fichiers xml et mettre à jour les champs suivants :

- status : DONE ou ERREUR
- archivage_return_id : l'id de VITAM en cas de succès (uniquement pour la table archivage)
- code_error : le code erreur renvoyé par VITAM
- error : le message d'erreur renvoyé par VITAM

Ensuite, une fois la base de données traitée, j'ai travaillé sur le processus de traitement des fichiers tars présents dans le dossier Retour :

- Lors du traitement des XML de succès, il faut supprimer le zip associé sur notre SFTP présent dans AAAA/MM/JJ/{type d'archive}
- Une fois que le XML a fini d'être traité, il faut le déplacer dans le dossier Retours_Backup/{type d'archive}/AAAAMMMJJ

Ces ajouts permettent d'avoir un système plus autonome qui sauvegarde les informations en bases de données et qui supprime les fichiers inutiles sur le serveur SFTP après traitement des fichiers xml.

La dernière évolution de l'archivage concerne le traitement des accusés d'archivage en erreur. Cela implique un processus de contrôle et de correction manuelle des fichiers

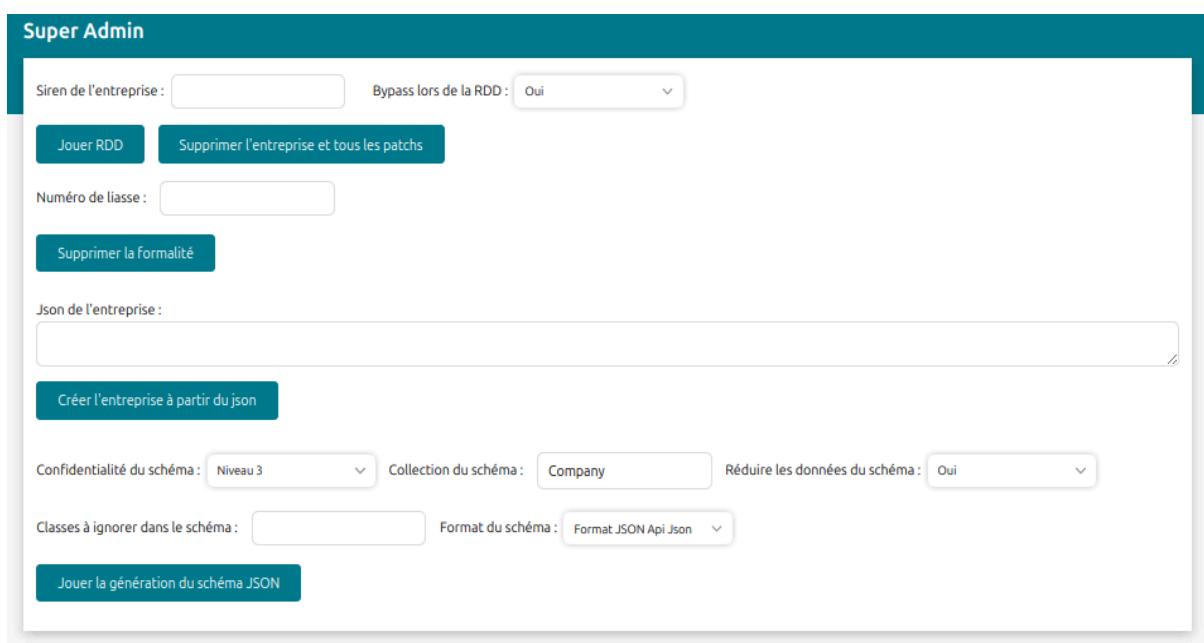
concernés, portant sur leur format, contenu et métadonnées, avant qu'ils ne soient retraités par Vitam. Je ne détaillerai pas davantage cette étape, car je n'y ai pas participé directement.

Pour un point de vue global et final sur l'archivage, un flux global représentant le système d'archivage complet est disponible en Annexe.

e. L'onglet Super Admin

La dernière fonctionnalité importante que je souhaite mentionner est l'implémentation d'un nouveau type d'utilisateur : le rôle Super-admin. Le RNE comporte déjà une dizaine de rôles différents, allant du niveau 1, le plus basique, au niveau 3, qui dispose de droits plus étendus. Il existait également un compte administrateur avec des droits complets. L'INPI a demandé la création d'un nouveau rôle, le Super-admin, qui devait cumuler les droits de l'administrateur tout en pouvant réaliser des actions jusque-là réservées à des manipulations directes dans la base de données, comme la suppression d'une entreprise via son SIREN ou l'ajout d'une entreprise en collant directement son JSON dans un champ dédié.

Pour répondre au besoin, j'ai commencé par développer les nouvelles routes permettant ces opérations spécifiques sur la base de données. Ensuite, j'ai créé ce rôle de Super-admin pour restreindre l'accès à ces routes aux seuls utilisateurs disposant de ce rôle. Pour tester ces nouvelles routes, j'ai utilisé HoppScotch, un outil permettant de tester et documenter les API du projet. Une fois les tests validés, je suis passé à la création de l'interface utilisateur. N'ayant pas de designer dédié et compte tenu de la simplicité du site, j'ai conçu moi-même cette interface en respectant le style général, plutôt basique, du site.



The screenshot shows a web-based administrative interface titled "Super Admin". The page contains several input fields and buttons:

- Inputs:**
 - "Siren de l'entreprise" (with a dropdown for "Bypass lors de la RDD" set to "Oui")
 - "Numéro de lieasse"
 - A large text area for "Json de l'entreprise"
 - "Classes à ignorer dans le schéma:" and "Format du schéma:" dropdowns (both set to "Format JSON Api Json")
- Buttons:**
 - "Jouer RDD" and "Supprimer l'entreprise et tous les patchs" (highlighted in blue)
 - "Supprimer la Formalité"
 - "Créer l'entreprise à partir du json"
 - "Jouer la génération du schéma JSON"
- Other controls:**
 - "Confidentialité du schéma:" dropdown set to "Niveau 3"
 - "Collection du schéma:" dropdown set to "Company"
 - "Réduire les données du schéma:" dropdown set to "Oui"

Figure 19 : Page dédiée aux super administrateurs

C'était une amélioration intéressante m'ayant permis de développer une fonctionnalité de bout en bout sur le site : de la création des routes et rôles sur le back-end, à la création de l'onglet sur le front. C'était aussi l'occasion de prouver que je pouvais mener une feature de A à Z en suivant l'architecture monolithique.

Cette fonctionnalité, très attendue par l'INPI, a été particulièrement bien accueillie à sa livraison. J'ai reçu de nombreux remerciements, ce qui fait toujours chaud au cœur. C'est une satisfaction importante de voir que le travail accompli est réellement apprécié.

f. Autres travaux

Sans entrer dans le détail de toutes les fonctionnalités que j'ai développées, je tiens à évoquer certaines évolutions que je trouve intéressantes, mais qui ne nécessitent pas un paragraphe dédié.

- ***Ajout d'une visionneuse de PDF***

Sur les pages de détail d'une entreprise ou d'un utilisateur, une section est dédiée aux fichiers associés à l'entité. Jusqu'à présent, la seule option pour consulter ces documents était de les télécharger, ce qui encombrait rapidement le disque dur des utilisateurs. La mise en place d'une visionneuse a donc été une fonctionnalité très attendue, car elle permet de visualiser les fichiers directement en ligne. Cela représente à la fois un gain de temps considérable et évite de surcharger les ordinateurs avec des téléchargements inutiles.

- ***Ajout d'un nouveau type de collection***

Une feature intéressante sur laquelle j'ai travaillé était l'ajout d'un nouveau type de collection pour les patchs des entreprises. Un patch companyPatch est une collection qui représente l'historique des modifications d'une entreprise, listant tous les changements qui ont été effectués dessus depuis l'ajout de l'entreprise au RNE.

J'ai ajouté la collection CompanyStructurePatch, une nouvelle collection de patch dédiée uniquement aux entreprises avec plus de 1000 établissements secondaires comme la Poste. L'objectif était de séparer les modifications structurelles d'une entreprise (Company) des modifications spécifiques à ses établissements (Establishments) afin de ne pas tout enregistrer dans une seule collection (companyPatch). Cette opération permet d'éviter de surcharger la collection companyPatch pour les grandes entreprises qui, avec le temps, accumulent tellement de modifications que les documents peuvent dépasser la limite de taille imposée par MongoDB qui est de 16 MO.

Les bénéfices étaient donc doubles : une meilleure scalabilité et lisibilité puisque les changements sont désormais catégorisés en fonction de la taille de l'entreprise. Mais aussi

de meilleures performances puisque les grandes entreprises ont leurs collections dédiées, ne surchargeant pas celles des petites.

Cela a été un travail intéressant au début de mon stage, m'ayant permis de mieux comprendre le fonctionnement du système d'historique des entreprises modifiées, mais aussi de créer mes premières routes et fonctionnalités destinées à de très grandes entreprises.

g. Analyse réflexive

La maintenance évolutive du RNE est un projet que j'ai apprécié par sa complexité. D'abord côté front-end, me confronter à une architecture atomique a été un défi en soi et l'ajout d'éléments en respectant cette dernière m'a permis de comprendre l'importance d'un code réutilisable et plus structuré. C'est définitivement une approche que j'essaierai de garder à l'avenir dans mes futurs projets, pas forcément de réaliser une architecture atomique mais de développer au maximum des éléments réutilisables s'ils apparaissent ne serait-ce qu'à deux endroits différents.

Mais c'est surtout la partie back-end qui était la plus intéressante. La complexité et diversité des API, des collections et tout simplement du côté métier des éléments du RNE était très challengeante en termes technique et de raisonnement. Respecter les nombreuses normes, interfaces, règles d'écriture m'a fait découvrir un aspect très strict du code, mais c'est ce qui fait aussi sa force et robustesse. Me voir progresser sur ce projet au fil des mois me procurait aussi une réelle satisfaction et un sentiment de progression.

Un autre aspect particulièrement intéressant au niveau du RNE était l'optimisation des traitements des données. Étant une base de données énorme, de nombreuses optimisations sont réalisées à tous les étages, et la création de la collection dédiée aux grandes entreprises était l'une d'elles.

Finalement, le RNE est un type de projet que l'on rencontre rarement mais qui permet de progresser du fait de sa complexité, et c'est du RNE que j'ai tiré les plus grandes leçons et bonnes pratiques durant mon stage.

Après quatre mois passés sur le projet RNE, celui-ci a été repris en interne par l'INPI. J'ai donc été affecté à un nouveau projet : la maintenance corrective et évolutive du site du Don Volontaire de la Moelle Osseuse.

3. Le Don Volontaire de Moelle Osseuse (DVMO)

Après quatre mois de développement sur le projet RNE, j'ai intégré un projet complètement différent : le site du Don Volontaire de Moelle Osseuse. Il s'agit du site officiel de l'Agence de la biomédecine, destiné à toute personne souhaitant devenir donneur de moelle osseuse.

Contrairement au RNE, qui s'adressait à des utilisateurs internes de l'INPI avec une interface principalement fonctionnelle, ce nouveau projet s'adresse à un public bien plus large. Il implique donc des attentes différentes, notamment en matière d'accessibilité, de design et d'ergonomie. L'interface est plus soignée, le parcours utilisateur plus guidé, et l'ensemble du site est pensé pour offrir une expérience fluide, intuitive et rassurante à des visiteurs non techniques. En ce sens, même si les fonctionnalités sont plus "basiques" techniquement que celles du RNE, les enjeux sont tout aussi importants, car ils concernent l'engagement et la confiance des futurs donneurs.

a. Le sujet du projet : Drupal - architecture Monolithique

Le site diffère non seulement par son public ciblé, mais aussi par les technologies employées. Celui-ci repose sur le CMS Drupal, avec une architecture monolithique.

De mon côté, j'étais jusqu'alors surtout habitué à des architectures headless, avec un front-end séparé du back-end et communiquant via des API. Cette logique, totalement opposée à celle du monolithe, m'a demandé un vrai effort d'adaptation. La prise en main de Drupal a été d'autant plus complexe que l'architecture est dense, parfois peu intuitive, et nécessite de comprendre comment les différents modules interagissent entre eux.

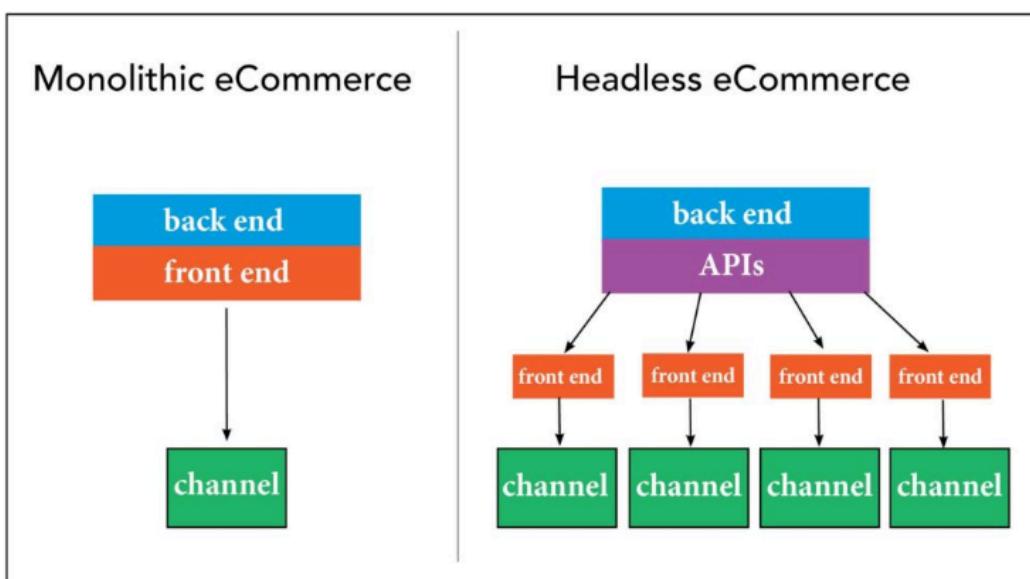


Figure 20 : Architecture Monolithique VS Headless

Avec Drupal en mode monolithe, tout est centralisé dans une seule base de code :

- Le CMS gère à la fois le contenu, la logique métier et la présentation ;
- Le front-end est généré côté serveur avec twig.

La figure suivante présente les différents outils que propose Drupal depuis l'interface.

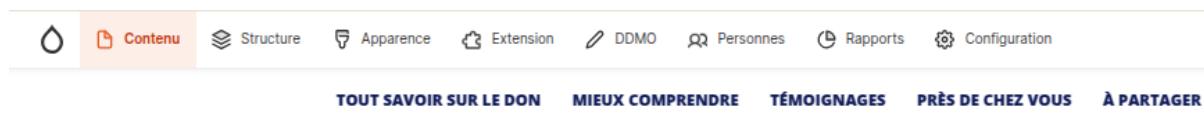


Figure 21 : Outils de contribution de Drupal

Contenu sert à gérer tout ce qui est publié sur le site, comme les articles, documents, images... **Structure** permet de définir les éléments structurels du site comme les types de contenus, les menus, les blocs ou les vues. **Apparence** gère tout ce qui est lié au thème visuel du site. **Extension** permet d'ajouter ou désactiver des modules qui apportent de nouvelles fonctionnalités. **DDMO** regroupe les pages spécifiques à notre projet, notamment celles liées à la gestion des dons. **Personnes** permet de gérer les comptes utilisateurs et leurs rôles. **Rapports** donne accès à différents journaux d'activité ou d'erreurs du site. Enfin, **Configuration** regroupe les réglages globaux comme la langue, les formats, la sécurité, etc. Ces outils permettent de construire très rapidement un site avec de nombreuses fonctionnalités déjà prêtées. C'est un avantage notable puisque pendant ma courte formation sur l'outil, j'ai pu créer une base de données, l'alimenter, faire une page dédiée pour une table avec pagination comprise en moins de trois heures.

Le problème provient davantage de la prise en main du code. En effet, étant donné que le front, back et la base de données y sont inclus, le code est difficile à lire déjà par la quantité de dossiers et fichiers présents. Sans trop rentrer dans les détails, Drupal fonctionne avec des modules qui viennent s'implémenter et ajouter des éléments au site. Chaque module se décompose en plusieurs fichiers qui définissent les routes, les droits, les fonctions, tout ça en php. Ainsi, pour une seule fonctionnalité ou page, c'est en moyenne sept fichiers qui sont créés pour assurer son bon fonctionnement.

b. La Relation Client

Étant donné qu'il s'agissait d'un nouveau projet, j'ai eu l'opportunité de le rejoindre dès le début et de m'y consacrer à 100 %, contrairement aux autres développeurs qui étaient également impliqués sur d'autres missions en parallèle. Cela m'a permis de rapidement prendre mes marques et d'avoir une certaine autonomie sur les développements. Après un mois de travail sur le projet, j'avais acquis une meilleure vision du temps nécessaire à la réalisation de certaines tâches, ce qui m'a permis de participer à l'estimation des tickets en vue du prochain bon de commande.

Lors des réunions hebdomadaires avec les clients, je me retrouvais ainsi régulièrement à présenter les avancées réalisées, à poser mes questions concernant certaines évolutions, ou encore à valider les orientations prises afin de m'assurer qu'elles correspondaient bien à leurs attentes. C'était un exercice nouveau sur le plan relationnel, car ces échanges étaient enrichissants ; ils permettaient de confronter plusieurs solutions, d'évaluer laquelle répondait le mieux aux besoins, et parfois de partir dans une nouvelle direction en fonction des retours. Il m'est aussi arrivé de devoir justifier certains délais, en expliquant pourquoi certaines fonctionnalités nécessitaient plus de temps de développement que prévu.

c. Mes réalisations

Ce projet m'a permis d'ajouter un nouvel outil à ma boîte à outils, puisque je n'avais encore jamais travaillé sur un site utilisant un CMS. J'ai ainsi eu l'occasion d'intervenir directement sur le contenu du site, afin de corriger la mise en forme, ajouter de nouveaux éléments, en retirer d'autres, et mieux comprendre la logique de fonctionnement d'un système aussi intégré que Drupal. Je vais maintenant présenter certaines fonctionnalités que j'ai mises en place sur le site du don de la moelle osseuse.

- ***Mise en place d'un lecteur vidéo***

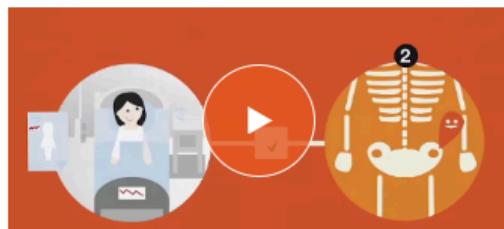
L'une des fonctionnalités attendues sur le site de la DVMO était la possibilité pour les contributeurs d'ajouter des vidéos directement depuis leur ordinateur.

Jusqu'alors, seules les vidéos hébergées sur YouTube pouvaient être intégrées. L'objectif était donc double : permettre l'intégration de vidéos locales à l'aide d'un lecteur intégré, et offrir la possibilité de les télécharger afin de faciliter leur partage, notamment sur les réseaux sociaux.

Pour répondre à ce besoin, j'ai dû me plonger dans les mécanismes propres à Drupal, en particulier le système de hooks et la manipulation des fichiers Twig, utilisés pour la structure du rendu HTML. La prise en main n'a pas été évidente, notamment à cause de problèmes avec mon éditeur (Visual Studio Code), qui ne détectait pas correctement les fichiers **.theme**. De plus, les fichiers Twig n'affichent pas les erreurs, rendant le débogage particulièrement difficile. À cela s'ajoute le fonctionnement spécifique de Drupal, qui nécessite de vider le cache à chaque modification, ce qui ralentit considérablement la phase de développement.

Malgré ces difficultés, j'ai pu progressivement comprendre la logique de l'outil en résolvant plusieurs tickets liés au lecteur vidéo. Cette expérience m'a permis de développer ma capacité à m'adapter à un environnement technique complexe et peu permissif en termes de retour d'erreurs.

FACILE DE PARTAGER À VOS PROCHES !



TEST SUPPORT 3



[Je télécharge](#)

Figure 21 : Interface du lecteur vidéo

Un des grands avantages de Drupal est la rapidité avec laquelle certaines fonctionnalités peuvent être mises en place. L'intégration d'un champ permettant d'ajouter une vidéo locale s'est faite très facilement via l'interface de configuration du CMS. En quelques clics, les contributeurs pouvaient désormais téléverser des fichiers vidéo depuis leur ordinateur.

La partie la plus complexe a concerné l'affichage correct de ces vidéos sur le site à l'aide de Twig. En effet, Drupal stocke les fichiers en leur attribuant des URIs internes, de la forme `public://videos/mon_fichier.mp4`, qui ne sont pas directement accessibles depuis un navigateur. Or, pour qu'une vidéo soit lisible dans un lecteur HTML et puisse être téléchargée, il faut lui associer une URL complète de type `https://example.com/sites/default/files/videos/mon_fichier.mp4`.

Pour résoudre ce problème, j'ai utilisé un service Drupal dédié à la conversion des URIs internes en URL publiques. Une fois cette étape franchie, j'ai pu intégrer le lecteur vidéo dans le rendu Twig, avec un affichage propre et une option de téléchargement fonctionnelle.

Ainsi, les contributeurs du site pouvaient enfin publier des vidéos directement, sans passer par des plateformes comme YouTube, et conserver l'hébergement de leurs contenus en interne.

- ***Ajout d'une bulle pour détailler une information***

Il s'agit du remplacement d'un lien de téléchargement d'un pdf détaillant la liste des maternités par une bulle d'information plus compacte et qui ne renvoie pas vers un fichier.

- Un liquide amniotique contenant les souches cellulaires du fœtus et de la mère. Il est également positif.
- Un liquide amniotique contenant les souches cellulaires du fœtus et de la mère. Il est également positif.
 - Une rupture de la paroi utérine.
 - Une souffrance fœtale.

LE RÉSEAU, C'EST QUOI ?

Le Réseau Français de cordon. Il est composé

Ces établissements sont autorisés à stocker des cellules souches hépatiques et pancréatiques, toutes agréées par le ministère de la Santé

Banques de sang placentaires	Maternités partenaires en France
Bordeaux	CHU Poitiers CHU Dupuytren Limoges GH Pellegrin-CHU Polycl Bourdeaux-Nord Aquitaine
Besançon	Centre Hospitalier Nord (Saint Priest en Jarez) Centre Hospitalier Universitaire Jean Minjoz (Besançon) Hôpital de la Croix Rousse (Lyon)
Montpellier	CHU Arnaud de Villeneuve (Montpellier) Clinique Clementville (Montpellier) Polyclinique Saint-Roch (Montpellier)
Nancy	CHRU de Nancy Polyclinique Majorelle (Nancy)
Saint-Louis	CH Intercommunal de Crétteil HP Antoine Béclère Hôpital Robert Debré (AP-HP) IH Franco-Britannique

ENTAIRE,

s dans l'activité de don de sang et de sang de cordon.

matériaux (poches contenant les partenaires réparties en France,

Figure 22 : Bulle d'information pour afficher les maternités

Le plus long a été la partie concernant le style. J'ai d'abord dû apprendre à identifier quel fichier CSS était chargé selon la page consultée. Ensuite, l'ajout de nouveaux styles s'est avéré très chronophage. Comme je l'ai mentionné précédemment, pour qu'une modification soit prise en compte, il faut d'abord la faire dans un fichier SCSS, puis reconstruire tous les fichiers CSS, ce qui prend environ 20 secondes. Ensuite, il fallait reconstruire le cache, ce qui prend aussi 30 secondes environ. Au total, il fallait en moyenne 40 secondes rien que pour voir l'impact d'un changement stylistique, ce qui est très long en développement.

○ Ajout de timecode sur les podcasts

Ajouter des timecodes sur les podcasts permet d'obtenir une information visuelle plus claire pour l'utilisateur sur le temps restant du podcast qu'il écoute, rendant le parcours utilisateur plus agréable.

ÉPISODE 1

MOELLE OSSEUSE ET MOELLE ÉPINIÈRE : MÊME COMBAT ?



Où se trouve la moelle osseuse ? Quel est son rôle ? Souvent lorsque l'on parle de moelle osseuse, on imagine qu'il s'agit de la moelle épinière... mais pas du tout ! Dans cet épisode, Cyrus vous éclaire sur ce qui différencie la moelle osseuse de la moelle épinière et met en lumière le rôle essentiel que joue la moelle osseuse dans notre corps.

[Lire la retranscription](#)

ÉPISODE 1

MOELLE OSSEUSE ET MOELLE ÉPINIÈRE : MÊME COMBAT ?



Où se trouve la moelle osseuse ? Quel est son rôle ? Souvent lorsque l'on parle de moelle osseuse, on imagine qu'il s'agit de la moelle épinière... mais pas du tout ! Dans cet épisode, Cyrus vous éclaire sur ce qui différencie la moelle osseuse de la moelle épinière et met en lumière le rôle essentiel que joue la moelle osseuse dans notre corps.

[Lire la retranscription](#)

Figure 23 : Comparaison entre ancien et nouveau timecode

d. Analyse réflexive

Je ne vais pas détailler toutes les améliorations que j'ai développé puisqu'elles se comptent par dizaines. Bien que ce n'étaient pas de grosses réalisations, ces ajouts m'ont permis de bien progresser sur Drupal et mais aussi de mieux comprendre comment améliorer le parcours utilisateur pour le rendre le plus agréable possible.

En effet, les fonctionnalités que j'avais traitées concernaient majoritairement une meilleure expérience utilisateur sur le site : informations disponibles dès le premier coup d'œil, meilleure navigation, contrôle de la data... Toutes ces améliorations sont des features que je garde en tête afin de les implémenter dans les futurs projets et simplement réutiliser ce que j'ai pu apprendre lors de mes expériences passées.

Ce projet m'a aussi permis de comprendre le fonctionnement d'un CMS et m'a permis de développer des compétences sur une technologie où la maîtrise se fait rare aujourd'hui, ce qui est un grand plus sur mon CV.

Après avoir présenté les deux projets majeurs de mon stage qui concernaient le développement web, je vais maintenant détailler un mini-projet un peu différent mais tout aussi intéressant qui concerne l'automatisation et la génération automatique de texte sur les sujets tendances sur LinkedIn.

4. Les Workflow n8n

J'ai reçu une demande de l'équipe commerciale visant à automatiser la veille sur LinkedIn, à travers le développement de deux workflows distincts. Le premier devait permettre de récupérer les sujets tendances de la semaine sur LinkedIn, afin d'en générer des résumés destinés à être diffusés aux différents pôles de l'entreprise. Le second workflow avait pour objectif de sélectionner l'un des dix sujets identifiés, puis de générer automatiquement un post LinkedIn, en s'appuyant sur une chaîne d'appels à différentes intelligences artificielles.

J'ai donc naturellement commencé par la mise en place du premier workflow. Après avoir suivi un tutoriel rapide sur l'outil utilisé, dont la prise en main s'est révélée assez intuitive, j'ai pu débuter le développement. Pour la récupération des sujets, je me suis appuyé sur une vidéo YouTube présentant l'utilisation d'un scraper LinkedIn. Cela m'a conduit à utiliser un scraper en ligne gratuit, bien que limité à 20 requêtes par mois.

a. Le premier workflow : Scraping de données

D'abord, nous devons définir les mots clés sur lesquels le scrapper va récupérer la data dans un tableau, ainsi que les mails à qui les sujets seront envoyés. La première partie se concentre donc sur la récupération de la data, ainsi que sa mise en forme par une IA (ici ChatGpt). La capture d'écran ci-dessous illustre ces différentes étapes :

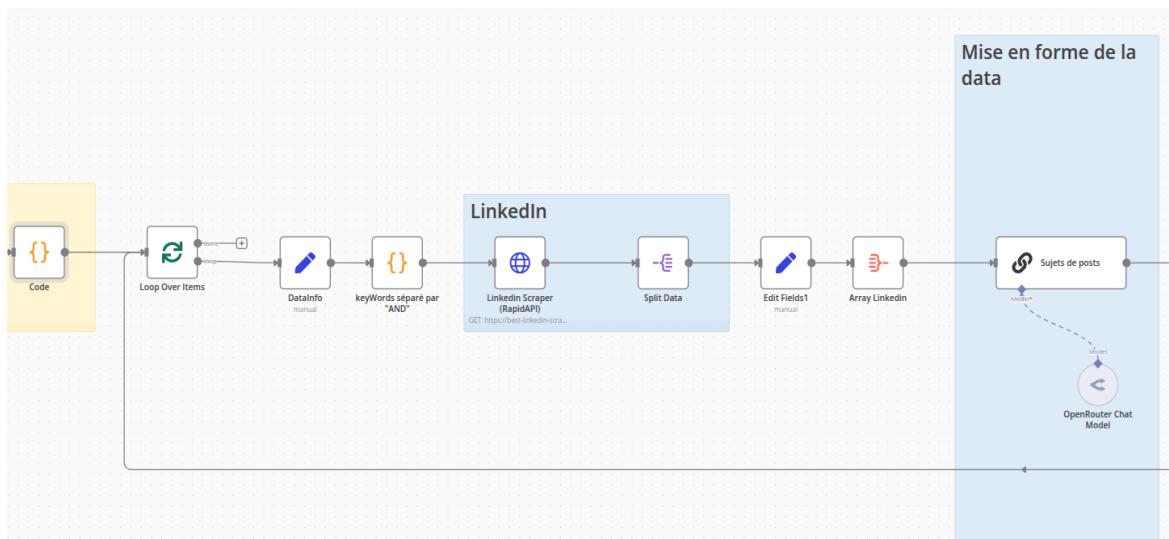


Figure 24 : Partie scrapping et génération du workflow

Les mots-clés à scraper sont donc définis dans la partie *Code*. Le *workflow* s'exécute ensuite pour chaque mot-clé de manière répétée. Après un léger formatage des données récupérées, celles-ci sont envoyées à une API qui renvoie les 10 sujets tendances LinkedIn de la semaine, en lien avec le mot-clé concerné. Les résultats étant retournés sous forme de

JSON, une étape supplémentaire est effectuée : les données sont envoyées à ChatGPT afin d'en générer des résumés clairs, composés d'un titre, d'une courte description pour chaque sujet et du lien url vers le post.

La prochaine étape, illustrée par la figure 25, est l'envoi par mail de ces résumés, ainsi que leur stockage dans une base de données NocoDb en ligne afin de pouvoir effectuer la génération des résumés depuis une interface plus claire.

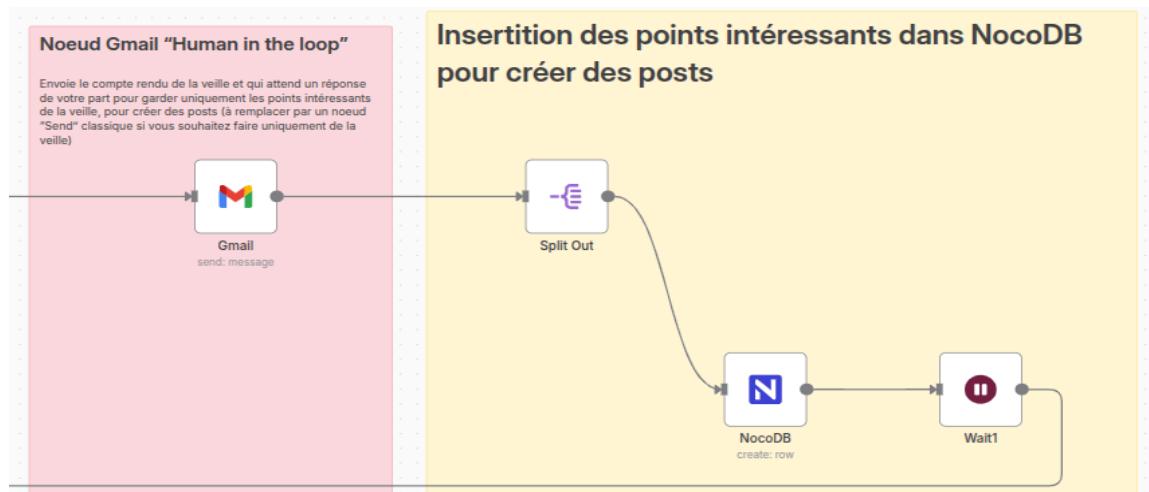


Figure 25 : Partie communication de la data

L'image suivante présente un screenshot d'un mail typique reçu après exécution du workflow avec pour mot clé “Devops et IA”.

1. Comment automatiser la localisation dans les pipelines CI/CD grâce à l'IA

Découvrez comment intégrer des outils d'automatisation de la localisation dans vos pipelines CI/CD pour gagner du temps et réduire les erreurs manuelles. Cette approche innovante facilite la gestion des versions multilingues dès le développement, avec un impact direct sur la qualité produit. Explorez les solutions capables de s'insérer dans vos workflows existants sans complexifier la chaîne DevOps.

[Découvrir le post](#)

2. Comment sécuriser et scalar des infrastructures multicloud via une orchestration IA-Friendly

L'utilisation d'IA pour orchestrer des environnements multicloud devient un levier précieux pour la scalabilité et la conformité. Ce contenu aborde les meilleures pratiques pour intégrer des outils hybrides alliant intelligence artificielle et gestion des politiques cloud, tout en respectant les contraintes de sécurité. Offrez des solutions concrètes pour pouvoir automatiser sans perdre en contrôle.

[Découvrir le post](#)

3. Le rôle grandissant des agents de test auto-réparants alimentés par l'IA dans DevOps

Les agents de test IA capables de s'adapter et de s'auto-corriger révolutionnent les processus d'assurance qualité. Ce sujet explore comment ces technologies diminuent le temps de maintenance des tests, augmentent la couverture et accélèrent les déploiements. Présentez aussi des retours d'expérience pratiques sur la mise en œuvre et la rentabilité de ces solutions.

[Découvrir le post](#)

Figure 26 : Exemple de contenu généré par le workflow

Un titre explicite sur le thème, une description pour plus de détails ainsi qu'un lien vers la source pour vérifier les informations. On peut effectivement voir que les sujets sont assez généraux et que c'est difficile d'aller plus loin avec aussi peu d'informations, le résultat est donc comme attendu : des sujets assez vagues qu'il faudrait pousser plus loin.

Enfin, ces informations sont triées sur la base de données NocoDB depuis laquelle on peut exécuter le deuxième workflow : la génération de post Linkedin.

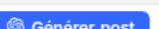
Title	Description	Email	Générer Post
La microformation mobile : Le futur de la formation continue ?	Plongez dans les innovations pédagogiques ...	bgrzadziel@steamulo.com	
Intelligence Artificielle et fonctions supports : un duo qui transfo...	Élaborez sur l'impact concret de l'intelligence...	bgrzadziel@steamulo.com	

Figure 27 : Data envoyée du workflow récupérée sur une table NocoDB

Après avoir approfondi la première partie de ce projet, je vais maintenant détailler la deuxième, qui consiste à la génération de posts Linkedin à partir des sujets récupérés.

b. Deuxième workflow : La création de posts LinkedIn.

Comme mentionné précédemment, les résumés générés restent assez succincts et ne fournissent pas toujours suffisamment d'informations pour rédiger un post pertinent. J'ai

ainsi développé un second workflow, visant à approfondir un sujet sélectionné et à récupérer des informations fiables disponibles sur internet.

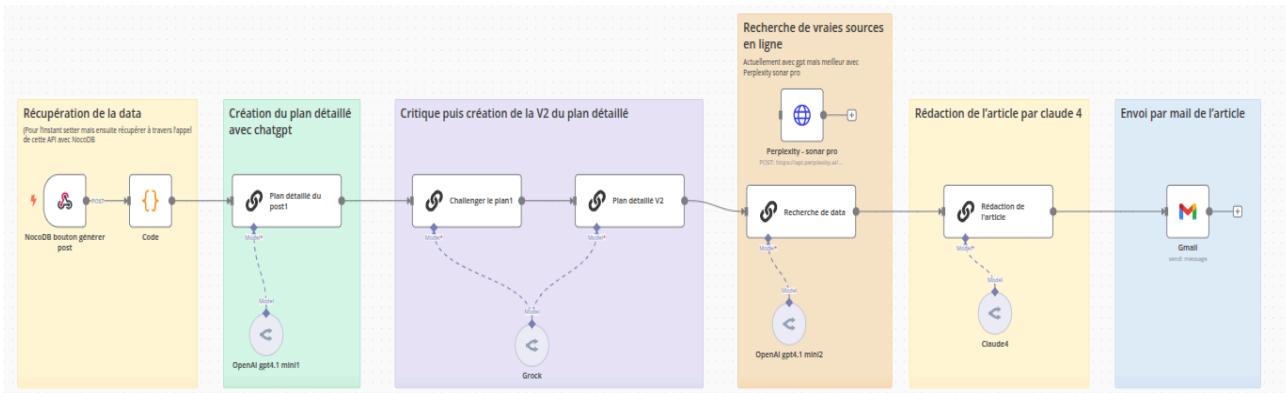


Figure 28 : Illustration des différentes étapes du deuxième workflow

Le workflow fonctionne de la manière suivante : un webhook est en attente d'une action utilisateur. Dès qu'un utilisateur clique sur "générer un post" depuis l'interface Nocodb, le workflow se déclenche automatiquement et enchaîne plusieurs étapes successives :

- Génération d'un premier plan détaillé via ChatGPT,
- Analyse critique de ce plan par Grok,
- Génération d'un plan détaillé V2 révisé à partir des retours de Grok,
- Recherche de sources fiables sur le sujet par ChatGPT,
- Rédaction complète de l'article à l'aide de Claude 4,
- Et enfin, envoi de l'article finalisé par e-mail à la personne concernée

J'utilise plusieurs intelligences artificielles dans ce workflow car chacune d'entre elles a ses points forts et ses limites.

D'abord, j'utilise **ChatGPT** pour générer un plan classique, car il est plutôt bon pour du raisonnement simple. Ensuite, j'enchaîne avec deux appels à **Grok** pour critiquer le contenu généré par ChatGPT. En effet, un problème récurrent concernant les IA est qu'elles ont du mal à critiquer et remettre en question ce qu'a écrit l'utilisateur. Grok est l'une des rares à pouvoir le faire assez librement, à condition de lui envoyer un prompt bien formulé qui l'encourage à critiquer sans se retenir.

Enfin, **Claude 4** est utilisé pour rédiger le post final, car il est reconnu pour ses capacités rédactionnelles. Une faiblesse de cet outil est sa tendance à générer un contenu excessif, sans savoir quand s'arrêter.

Ci-dessous, un exemple de mail tronqué que j'ai reçu au sujet de la micro formation mobile :

Microformation Mobile : Le Futur Incontournable de la Formation Continue ?

Et si vos équipes perdent 70% de leurs compétences en moins de 5 ans ? Dans un monde professionnel qui accélère sans cesse, la formation continue n'est plus une option, c'est une urgence vitale. Découvrez comment la microformation mobile révolutionne tout avec des formats courts, immersifs et ultra-accessibles. Prêt(e) à booster vos compétences sans perdre une minute ? Plongeons dans cette révolution pédagogique qui transforme déjà les entreprises les plus innovantes.

1. Pourquoi la Formation Continue doit se Réinventer d'Urgence

Face à l'évolution rapide du monde professionnel, la formation traditionnelle montre ses limites de manière criante. Trois constats alarmants s'imposent à toute organisation qui souhaite rester compétitive.

Premier choc : les compétences techniques et digitales s'effacent à une vitesse vertigineuse. Selon le *LinkedIn Learning Workplace Learning Report 2021* et le *World Economic Forum Future of Jobs Report 2020*, **70% des compétences deviennent obsolètes en seulement 5 ans**. Cette accélération s'explique par les innovations technologiques, l'automatisation et l'évolution des méthodes de travail.

Deuxième défi : les entreprises peinent à former efficacement sans bloquer la productivité ou exploser les budgets. Le rapport *Deloitte 2023 Global Human Capital Trends* révèle ce paradoxe : comment former vite et bien sans paralyser l'activité ? Les formations présentes classiques monopolisent du temps précieux et génèrent des coûts logistiques prohibitifs.

Exemple concret : Une PME industrielle française (source : rapport AGEFOS PME 2022) a effectivement perdu plusieurs contrats suite à un retard dans la mise à jour sur la norme ISO 45001. Ce retard de formation a causé un manque à gagner estimé à plus de **150 000 € sur un an** – un coût bien plus élevé que l'investissement initial en formation.

Cerise sur le gâteau : le *World Economic Forum* estime que **50% des employés auront besoin d'une requalification d'ici 2025**, soit dans moins de deux ans. Face à cette urgence, rester sur les méthodes traditionnelles équivaut à naviguer à vue dans un océan de défis.

⚠️ Le temps presse : former vite et bien n'est plus négociable !

2. Microformation Mobile : La Révolution de l'Apprentissage Express

Imaginez pouvoir former vos équipes en **sessions de 5 à 10 minutes**, directement sur smartphone, avec un impact mesurable et durable. C'est exactement ce que propose la **microformation mobile**, une approche qui révolutionne la manière d'apprendre en entreprise.

Qu'est-ce que la microformation mobile ? Il s'agit d'un apprentissage fractionné, découpé en modules courts et ciblés, accessibles via smartphone ou tablette. Chaque session vise un objectif pédagogique précis : maîtriser une procédure, comprendre une réglementation, ou acquérir une compétence technique spécifique.

Sa force principale ? La flexibilité totale. Terminés les créneaux bloqués de 3 heures qui désorganisent les plannings ! La microformation s'intègre naturellement aux agendas chargés grâce aux notifications push intelligentes et aux modules disponibles à la demande. Vos collaborateurs apprennent entre deux réunions, dans les transports, ou pendant une pause déjeuner.

Cas d'école chez Siemens : selon leurs communiqués presse de 2021, cette entreprise industrielle a déployé une application de microlearning pour former **10 000 techniciens** sur une mise à jour logicielle critique. Résultat ? Formation bouclée en **3 semaines au lieu de 3 mois** avec les méthodes traditionnelles.

La recherche de Josh Bersin (Bersin by Deloitte, 2022) le confirme : la microformation mobile permet une **réduction de 30 à 50% du temps de formation** comparé aux formats classiques, avec une meilleure rétention des informations. Le chiffre de référence ? **40% de gain de temps en moyenne**.

✓ Format court, Impact maxi !

Figure 29 : Exemple post généré à l'issue du deuxième Workflow

Les résultats sont plutôt convaincants, et les sources démontrent la fiabilité de l'outil. Toutefois, pour des raisons internes, ce projet a été mis en standby, de sorte que Steamulo n'a pas encore été amené à l'utiliser.

c. Analyse réflexive

Une autre question peut se poser : celle de la responsabilité écologique de ce workflow, qui repose sur de nombreux appels à des IA pour produire du contenu. Rien que pour sa conception et les différentes vérifications de son bon fonctionnement, j'ai dû effectuer des dizaines d'appels, sachant que chaque nouvelle exécution entraîne encore davantage d'appels.

Une amélioration judicieuse consisterait à rendre le workflow plus durable en limitant le nombre d'appels, par exemple en évitant de faire appel aux IA lorsque le contenu de base est jugé non pertinent.

Au-delà de l'aspect écologique, un autre point à surveiller concerne la fiabilité des données utilisées. Toute la data récupérée provient de l'IA et du scrapper LinkedIn, encore faut-il qu'elle soit fiable. Même si l'IA produit ici le plus gros du travail, un travail de vérification doit toutefois être réalisé afin d'éviter de partager de fausses informations.

Néanmoins, je trouve que ce projet représente un outil fiable de veille technologique et qu'il peut être intéressant à intégrer dans les newsletter des entreprises. Et dans un contexte de développement rapide de l'intelligence artificielle, les workflow utilisant l'IA vont devenir la norme. Pour rester compétitives, les entreprises doivent constamment s'adapter à l'évolution du marché. Dans un monde où l'IA devient incontournable, il est essentiel d'apprendre à l'utiliser de manière stratégique et à en optimiser les usages.

Conclusion

Pour conclure, je tiens à exprimer ma grande satisfaction à l'égard de mon expérience professionnelle chez Steamulo. Que ce soit au niveau de la formation, de l'accompagnement ou des connaissances et compétences que j'ai acquises, cette expérience a été formatrice et m'a permis de grandir professionnellement. Je suis persuadé que les compétences acquises me seront très utiles dans la suite de mon projet professionnel.

La polyvalence des projets menés a particulièrement marqué mon expérience. En effet, pendant mes six mois de stage, j'ai pu travailler sur trois projets principaux différents qui m'ont chacun permis d'acquérir des compétences variées. Grande montée en compétence sur le développement côté serveur avec des règles métiers complexes sur le RNE, découverte du fonctionnement d'un CMS avec Drupal et respect d'un design établi pour le client, ainsi que la découverte des workflow pour un usage utile à l'entreprise.

L'organisation et la participation aux réunions a été aussi un nouveau défi pour moi, puisque contrairement à mon précédent stage, le cadrage et réunions étaient davantage respectés, avec des échanges quotidiens avec les clients. Cela m'a appris à adopter un comportement plus professionnel et me conformer aux attentes du monde du travail, à échanger avec le client et à débattre sereinement sur des propositions de solutions.

Ainsi, grâce à ce stage, j'ai pu améliorer et développer de nouvelles compétences. Tout d'abord, sur le plan des compétences sociales et de la synthèse, ma participation aux réunions d'équipe m'a amené à apprendre à condenser mon travail et à l'expliquer de manière concise devant les autres membres du projet. Ensuite j'ai acquis une rigueur accrue dans mon approche du travail : travailler pour un vrai client et non un projet interne m'a appris à respecter ses exigences et produire un travail conforme aux attentes.

Enfin, le développement de mes compétences en travail d'équipe a été une composante essentielle de cette expérience. Chaque contribution que je faisais devait être accessible et bénéfique pour l'ensemble de l'équipe. Cela a renforcé ma capacité à collaborer efficacement, à prendre en compte les besoins de l'équipe et à aligner mes efforts sur les objectifs communs.

Table des matières

Résumé technique.....	4
Introduction.....	5
I. Une expérience de plus de 12 ans, l'histoire de Steamulo.....	6
1. Activité, secteur et environnement de l'entreprise.....	6
2. Le positionnement de Steamulo, ses services et outils.....	6
a. Les pôles au sein de Steamulo.....	6
b. Les outils de Steamulo.....	7
c. Les partenaires de Steamulo.....	8
d. Les dernières évolutions de l'entreprise et réglementations.....	8
II. L'analyse de mon expérience au sein de l'entreprise.....	9
1. Le sujet de stage.....	9
2. L'environnement de travail.....	9
3. Les méthodes de travail.....	10
4. Les outils utilisés.....	11
5. L'ambiance chez Steamulo.....	13
III. Les missions réalisées durant mon stage.....	15
1. Le BaseJump.....	15
2. Le Registre National des Entreprises - TMA.....	22
3. Le Don Volontaire de Moelle Osseuse (DVMO).....	37
4. Les Workflow n8n.....	43
Conclusion.....	49
Table des matières.....	50
Sitographie.....	51
Glossaire.....	52
Annexe.....	55

Sitographie

Site de Steamulo : <https://steamulo.com>

Site du RNE : <https://registre.entreprises.gouv.fr>

Site du Guichet Unique : <https://procedures.inpi.fr>

Site du Don Volontaire de la Moelle Osseuse : <https://www.dondemoelleosseuse.fr>

Présentation de l'Architecture Headless VS Monolithique :

<https://digicommercegroup.com/insights/headless-commerce-vs-monolithic-commerce/>

Présentation de la Méthode Agile Scrum :

<https://www.tuleap.org/fr/agile/comprendre-methode-agile-scrum-10-minutes>

Présentation d'API REST : <https://topexemples.fr/api-rest-exemple/>

Présentation de l'Architecture Atomique :

<https://www.adimeo.com/blog/design-system-a-quoi-ca-sert-et-comment-le-faire>

Définition Doctrine :

<https://symfonycasts.com/screencast/symfony5-doctrine/entity>

Définition Framework :

<https://aws.amazon.com/fr/what-is/framework/>

NB : Les différents liens ont été consultés pour la dernière fois en date du 04/08/2025.

Glossaire

API : Une API (interface de programmation d'application) est une interface logicielle qui permet de « connecter » un logiciel ou un service à un autre logiciel ou service afin d'échanger des données et des fonctionnalités.

ESN : Une ESN, ou Entreprise de Services Numériques, est une société spécialisée dans la fourniture de solutions informatiques pour les entreprises.

CMS : CMS est l'acronyme de Content Management System, c'est-à-dire système de gestion de contenu. Il s'agit d'un logiciel en ligne grâce auquel il est possible de créer, de gérer et de modifier facilement un site web, sans avoir besoin de connaissances techniques en langage informatique.

Workflow : Un workflow, aussi appelé flux de travaux ou flux opérationnel, désigne une suite de tâches ou d'opérations liées au traitement de données. C'est un genre de circuit qui consiste à fournir à chacun des acteurs les informations nécessaires pour la réalisation de leurs tâches. Dans le cadre de mon stage, un workflow n8n est donc un circuit de tâche automatisée qui, à partir d'une donnée en entrée, en ressort une autre en sortie.

CRON : Un cron est un programme système qui permet aux utilisateurs de planifier des tâches qui vont s'exécuter automatiquement tous les X temps.

Fichier .tar : Un fichier .tar correspond à un fichier d'archive. Les fichiers .tar sont des conteneurs pouvant stocker des fichiers, répertoires et d'autres objets dans une archive. C'est comme un fichier .zip mais qui n'applique pas la compression à son contenu.

Chief Technical Officer (CTO) : Le Directeur des Nouvelles Technologies est le garant de l'innovation technique et de la mise en œuvre de nouvelles solutions technologiques au sein d'une entreprise.

Customer Relationship Management (CRM) : Il s'agit d'un logiciel de gestion de la relation client permettant de centraliser, organiser et gérer les échanges entre l'organisation et ses usagers.

Devops : L'ingénieur Devops vise à créer un environnement dans lequel la conception, les tests et diffusion de logiciels peuvent être réalisés rapidement. Il s'occupe en général de la partie hébergement et cloud d'une entreprise.

Scraping : Le scraping, ou extraction de données en français, consiste à extraire les informations d'un site web vers une feuille de calcul ou base de données. C'est une méthode efficace pour capturer une grande quantité d'informations à des fins d'analyse.

Doctrine : Doctrine est un ORM, c'est-à-dire un mapper d'objet relationnel. Chaque table dans la base de données aura une classe PHP correspondante avec ses attributs. Cela évite d'écrire des requêtes SQL pour interagir avec la base de données.

Framework : Un framework est un ensemble de composants réutilisables qui permettent de développer de nouvelles applications plus efficacement.

Librairie en code : Une librairie est un ensemble de fonctions et définitions pré-écrites qui peuvent être utilisées par les développeurs pour faciliter le développement d'applications.

Backlog : Un backlog est une liste hiérarchisée de tâches destinées à l'équipe de développement. Il permet de garder la trace des tickets effectués et des fonctionnalités à venir, et de suivre quelle personne travaille sur quel élément.

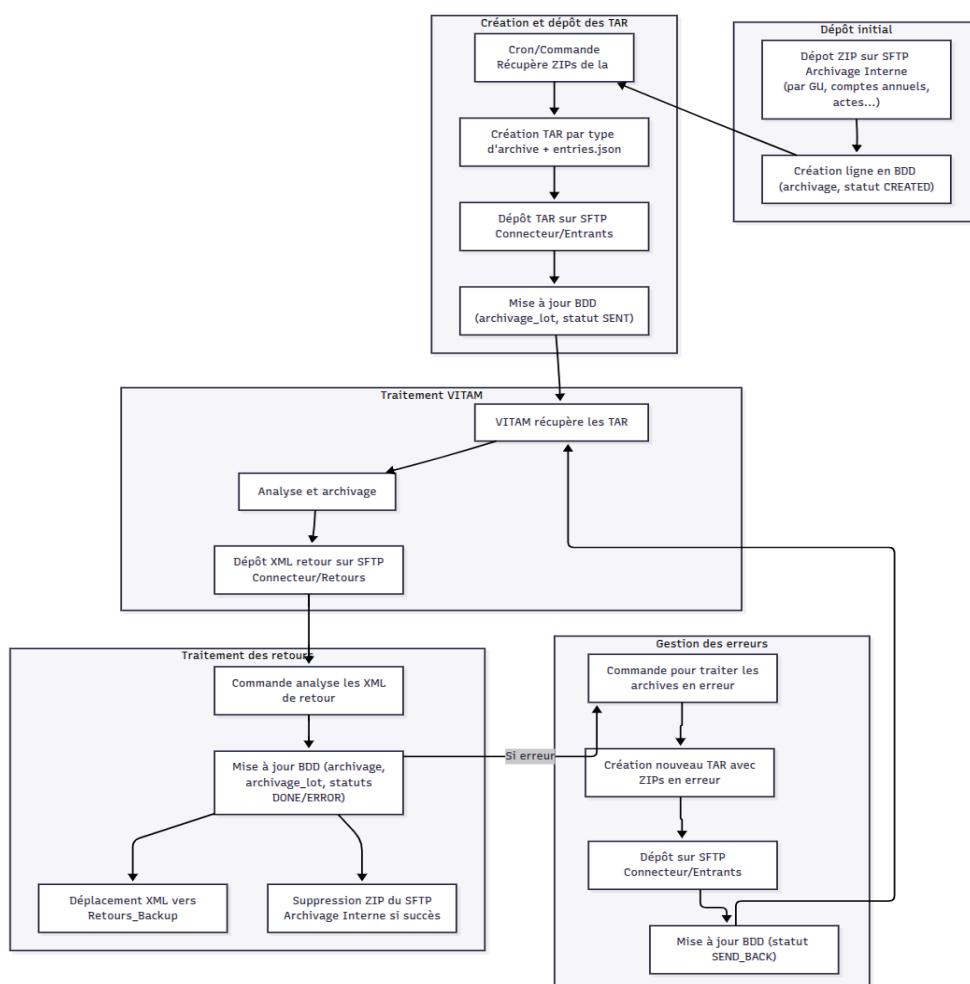
Twig : Twig est un moteur de template pour le langage de programmation PHP. Il permet de créer des modèles de pages web en utilisant des syntaxes simples pour insérer des données dynamiques dans les pages HTML.

Jenkins : Jenkins est un outil d'intégration et de déploiement continu. Il aide à automatiser les parties du développement logiciel liées au build, tests et au déploiement.

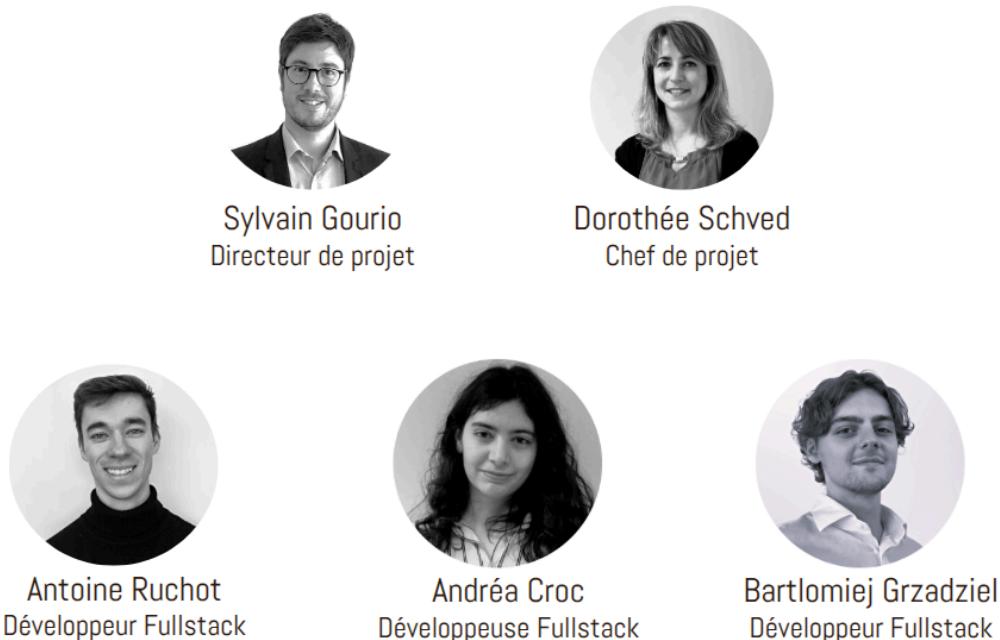
Annexe

Pièce 1 : Flux global de l'Archivage

- **Flux 1 :** Traitement documents à archiver et dépôt temporaire sur Sftp Archivage Rne
- **Flux 2 :** Génération de tar dans lesquels sont ajoutés les archives de la veille contenus dans le sftp Archivage. Puis ajout des tar dans le dossier Entrants du Sftp Connecteur
- **Flux 3 :** Traitement des accusés de traitements retournés par Vitam dans le dossier Retours du Sftp Connecteur
- **Flux 4 :** Traitement des erreurs pour pouvoir déposer des archives spécifiquement choisis sur le SFTP Connecteur



Pièce 2 : Organigramme de l'équipe du RNE



Pièce 3 : Schéma d'urbanisation du RNE

