2022/2024

Amine NAKHIL

Ministères Sociaux

2022/2024

Rapport d’alternance



Table des matières

[Table des matières 1](#_Toc104978333)

[1. Remerciements 2](#_Toc104978334)

[2. Introduction 3](#_Toc104978335)

[3. Problématique 3](#_Toc104978336)

[4. Présentation du Ministère 3](#_Toc104978337)

[4.1 Historique du Ministère 4](#_Toc104978338)

# Remerciements

En premier lieu, je tiens à remercier mon tuteur d’apprentissage ***Mme Julie MARSHALL et Mme Catherine LOZE-FAELENS***. Un grand merci pour leur accueil chaleureux au sein du ministère auquel ils m’ont acclimaté, ainsi que pour leur patience et leurs précieux conseils. Elles m’ont beaucoup appris sur le ministère. Elles ont toujours su me faire relever des défis tout en comprenant les enjeux qui me semblent incertains et ont toléré nombre de mes absences pour motif médical.

Je saisis également cette occasion pour adresser mes profonds remerciements aux responsables et au personnel de **l*‘ETNA.*** Ils ont toujours été à mon écoute et ont su m’apporter un soutien sans faille, notamment en ce qui concerne les aménagements relatifs à mon handicap.

Je désire aussi plus spécialement remercier les professeurs de **l*‘École ETNA***, qui m’a fourni les outils nécessaires au bon déroulement de mon alternance. Je tiens particulièrement à remercier ***Mme Linda THAP, et Mme. Marie NIANG,*** qui furent les premiers à me soutenir dans ma démarche d’emploi.

Un grand merci à ma **mère** et mon **père**, pour leurs conseils, ainsi que pour leur soutien inconditionnel, à la fois moral et économique.

Pour finir, je voudrais remercier, ***M.******Henri FAGEBAUME***, mon chef, pour ses conseils avisés sur la conduite à tenir au sein du Ministère.

# Introduction

Ce mémoire a pour objet de présenter le déroulé de l’année en alternance que j’ai effectuée dans la cadre du Master of Science, et plus particulièrement de l’application de mes connaissances théoriques réalisée en tant qu’alternant aux Ministères Sociaux. Cela impliquera d’analyser et d’étudier les activités réalisées en milieu professionnel.

Ce mémoire détaillera l’ensemble des éléments nécessaires à la compréhension des enjeux d’informatique au sein des Ministères Sociaux ainsi que les clefs de réponse à la problématique susmentionnée.

# Présentation du Ministère

## Historique du Ministère

Le Ministère du Travail a vu le jour en 1906, en rassemblant plusieurs directions qui étaient auparavant sous la tutelle du Ministère du Commerce et du Ministère de l'Intérieur. Ces directions étaient notamment celles du Travail, de l'Assurance et de la Prévoyance sociale, ainsi que de la Mutualité.

Depuis 1966, le Ministère du Travail a souvent changé de nom pour devenir le Ministère des Affaires sociales, notamment entre 1966 et 1969, 1983 et 1984, et 1986 et 1988. Il y a eu des périodes où le Ministère du Travail et le Ministère des Affaires sociales ont fonctionné comme des entités distinctes, comme entre 1984 et 1986 et 1988 et 1995. De 2002 à 2010, les questions de Solidarité et/ou de Cohésion sociale étaient principalement gérées par le Ministère du Travail, tandis que la Santé était traitée séparément. Entre 2010 et 2012, les deux ont été regroupés, et un nouveau Ministère des Solidarités et de la Cohésion sociale a été créé. Depuis 2012, le Ministère du Travail et le Ministère des Affaires sociales, renommé Ministère des Solidarités à partir de 2017, fonctionnent à nouveau comme des entités séparées.

* 1. **L’organisation des Ministères**
     1. **Le Ministère du Travail, du Plein Emploi et de l’Insertion**

Le Ministère du Travail en France est une institution gouvernementale qui se consacre à la régulation du marché du travail, à la protection des travailleurs et à la promotion de l'emploi. Le ministère intervient dans un large éventail de domaines allant du droit du travail aux questions de formation professionnelle et de relations sociales. Le Ministère du Travail joue un rôle clé dans la création et la mise en œuvre des politiques publiques relatives à l'emploi, aux relations de travail et à la protection sociale.

Le Ministère du Travail est composé de plusieurs divisions et sous-divisions, y compris :

Cabinet du Ministre: Le Ministre du Travail est le chef du ministère et est responsable de l'orientation générale des politiques.

Directions Générales: Il existe plusieurs directions générales au sein du ministère, telles que la Direction Générale du Travail (DGT) et la Direction Générale à l'Emploi et à la Formation Professionnelle (DGEFP). Ces directions sont responsables de divers aspects tels que la réglementation du travail, l'inspection du travail, l'emploi, la formation professionnelle et l'apprentissage.

Organismes attachés: Plusieurs organismes et institutions sont rattachés au ministère, comme Pôle Emploi qui s'occupe de l'accompagnement des demandeurs d'emploi, ou encore l'Agence Nationale pour l'Amélioration des Conditions de Travail (ANACT).

Services déconcentrés: Les Directions Régionales des Entreprises, de la Concurrence, de la Consommation, du Travail et de l'Emploi (DIRECCTE) sont des services extérieurs du ministère qui agissent au niveau régional pour mettre en œuvre les politiques nationales.

Fonctions et Responsabilités

Réglementation et Contrôle: Le ministère élabore le code du travail et veille à son application à travers des inspections du travail.

Formation et Emploi: Le ministère est chargé de la formation professionnelle et de l'apprentissage. Il supervise également les politiques d'emploi et de chômage.

Relations Sociales: Le ministère joue un rôle dans la gestion des relations sociales, y compris les relations entre les employeurs et les employés, la négociation collective et la résolution des conflits.

Protection Sociale: Le ministère participe à l'élaboration des politiques de protection sociale liées au travail, comme les politiques de santé et de sécurité au travail.

Gestion des Crises: Le ministère peut également jouer un rôle dans la gestion des crises économiques ou sociales qui ont un impact sur l'emploi ou les conditions de travail.

Le Ministère du Travail en France est une institution clé pour la réglementation du marché du travail et la protection des droits des travailleurs. Sa structure complexe lui permet de gérer un large éventail de questions, allant de la réglementation du travail et de l'emploi à la formation professionnelle et aux relations sociales. Le ministère joue ainsi un rôle vital dans le maintien de l'équilibre entre les besoins des employeurs et les droits des travailleurs.

* + 1. **La DNUM des Ministères Sociaux**

**La Direction du Numérique des Ministères Sociaux (DNUM) de France est une entité gouvernementale qui joue un rôle central dans la transformation numérique des services publics dans les domaines de la santé, de l’action sociale et de la sécurité sociale. Elle est placée sous l'autorité des ministres chargés de ces secteurs et est dirigée par un délégué au numérique en santé, assisté d'un adjoint, tous deux nommés par décret et arrêté respectivement​1​.**

**Les missions de la DNUM sont vastes et essentielles pour l'administration des affaires sociales en France. Elle est responsable de la proposition, la mise en œuvre et le suivi de la feuille de route du numérique en santé, assurant ainsi que les projets numériques dans le domaine de la santé sont en cohérence avec la stratégie numérique globale de l'État. La DNUM anime également la gouvernance du numérique en santé, qui rassemble divers acteurs, y compris des représentants des usagers et professionnels du secteur, afin de discuter et de guider les politiques numériques en santé. Elle propose des évolutions du cadre juridique pour les services numériques en santé, assure la mise en conformité des outils numériques utilisés dans le secteur, supervise la mise en œuvre de certaines plateformes numériques essentielles et coordonne les actions dans le domaine du numérique en santé à l'échelle européenne et internationale​1​.**

**La DNUM agit comme un acteur majeur dans l'appui des Directions de politiques publiques et vise à renforcer la performance des structures et du personnel des ministères des Solidarités et de la Santé et du Travail. En leur fournissant conseils et appui, en concevant des produits et services numériques, et en développant l'environnement de travail numérique des agents, la DNUM facilite une plus grande efficacité et mobilité des services publics​2​.**

**En termes d'organisation, la DNUM comprend plusieurs sous-directions et missions spécialisées, telles que le Département Appui et Ressources, la Cellule Communication, la Mission Fabrique numérique, la Mission Transformation numérique, la Sous-direction Projets et services numériques, la Sous-direction Environnement de travail numérique des agents, la Mission Confiance numérique, et le Service Systèmes d'information mutualisés des ARS​3​.**

**Pour plus d'informations détaillées sur ses missions et son organisation, vous pouvez consulter les décrets et arrêtés qui régissent la DNUM sur le site de Légifrance et sur le site des Ministères Sociaux.**

# Mon arrivée au Ministère

Mon arrivée au Ministère a été suivi de procédures, et de protocoles à respecter. J’ai d’abord été à la direction des ressources humaines pour toutes les formalités administratives que mon arrivée présupposait. Ensuite, il a fallu me conformer à la culture d’entreprise de la structure où je travaille, on m’a fait un briefing sur toutes les technologies utilisées au ministère Il a fallu ensuite socialiser avec mes différents responsables hiérarchiques, mes collègues de bureau et plus largement l’ensemble des personnes à qui je serais confronté lors de mon alternance. Enfin, comme je souffre d’un handicap, il a fallu faire une visite médicale pour évaluer mon aptitude à réaliser ma mission, dans mon cas, le médecin de prévention m’a donné des aménagements de mon poste, en l’occurrence un casque antibruit, une souplesse horaire, un télétravail thérapeutique ponctuel et une possibilité de m’absenter de manière périodique pour motif médical.

# Présentation de Cod’It

Cod'It est un outil développé par le Ministère du Travail, du Plein emploi et de l'Insertion en France. Il est conçu pour aider les utilisateurs à naviguer à travers les dispositions du code du travail. Le logiciel est à jour des dispositions du code du travail au 1er août 2022.

Fonctionnalités Principales

* **Organisation Thématique**: Cod'It est organisé en différentes feuilles de calcul qui permettent de consulter les dispositions du code du travail selon une organisation thématique.
* **Compatibilité**: Le logiciel est compatible avec Microsoft Excel et offre également une version compatible avec la suite Libre Office.
* **Recherche Avancée**: Il permet de voir les articles relatifs à un thème en un même emplacement et offre une fonction de recherche par arborescence thématique.
* **Réutilisabilité**: Les utilisateurs peuvent copier les articles pour les réutiliser dans d'autres applications.

Évolution Future

Le logiciel est en cours d'évolution, et une nouvelle version devrait être disponible à l'horizon 2024.

Participation des Utilisateurs

Plus de 300 personnes ont répondu au questionnaire de recueil de besoin en 2021, contribuant ainsi aux réflexions sur les améliorations futures du logiciel.

# Les outils que j’utilise au Ministère

## Talend : zoom sur les spécificités de cet outil ETL Open-SourceTalend — WikipédiaTalend

Le ministère utilise beaucoup un logiciel de type ETL (Extract Transform Load) pour l’exécution de script impliquant des données, et pour le traitement de celles-ci. C’est ainsi que j’ai appris à connaitre l’ETL Talend. Un ETL est un logiciel d’extraction, de traitement et de mappage (mise en relation) des données, généralement à des fins de synchronisation. Il s’appuie sur un langage de programmation, chez nous Java, et permet d’exécuter des scripts sans forcément connaitre le langage de programmation sur lequel le logiciel s’appuie. Bien que l’on puisse modifier manuellement le langage produit par l’ETL. Le logiciel Talend est extrêmement utilisé pour traiter les données au sein du ministère, il était donc incontournable d’apprendre à l’utiliser dans le cadre de mon alternance, pour me former au logiciel Talend, il a donc fallu que je m’exerce à l’utiliser, surtout que lorsque l’on est habitué à programmer de manière plus conventionnelle, programmer sur un ETL peut être très déconcertant.

## 7.2 DBeaver

## Après avoir rejoint le Ministère, j'ai été rapidement initié à sa culture organisationnelle, qui inclut l'usage obligatoire du logiciel DBeaver. Ce programme est une plateforme exhaustive pour la gestion de bases de données, compatible avec une variété de systèmes de gestion de bases de données (SGBD), y compris celui en service au Ministère. L'adaptation n'a pas été facile, car j'étais novice dans ce domaine spécifique. DBeaver utilise un pilote JDBC et une API qui permet aux applications Java d'interagir avec des bases de données qui utilisent également JDBC.

## Au début, DBeaver était assez déroutant pour moi et a nécessité une période d'adaptation, surtout par rapport aux outils que j'avais utilisés pendant mes études. Ces derniers n'étaient pas toujours en adéquation avec les exigences techniques du Ministère. Par exemple, j'étais habitué à utiliser MySQL avec le framework PHPMyAdmin, ce qui est très différent de l'approche plus professionnelle de DBeaver pour la gestion et les requêtes de bases de données.

## DBeaver a été un outil éducatif pour moi, m'offrant des perspectives sur la gestion des données au Ministère ainsi que sur les standards SQL. Avant cela, j'étais plus à l'aise avec l'utilisation de SQL en ligne de commande ou via l'interface de PhpMyAdmin.

## PostgreSQL

Le Ministère suit des directives très ciblées en matière de gestion de bases de données, en raison de besoins spécifiques. C'est pourquoi PostgreSQL a été sélectionné comme le SGBD officiel au sein de l'Administration Centrale. Tous les collaborateurs sont donc tenus d'utiliser ce système. Il est généralement reconnu que PostgreSQL est le SGBD le mieux adapté aux exigences du Ministère.

# Modélisation Base de Données

Lorsque j'ai rejoint l'équipe du projet "Cod'It" au Ministère du Travail, je me suis rapidement rendu compte de l'importance cruciale de la modélisation des bases de données pour réussir notre mission. Le projet avait pour objectif de numériser et d'optimiser divers processus administratifs, et une base de données solide était au cœur de cette transformation.

Mcd de la base de données que j’ai réalisé

N'ayant jamais créé de Modèles Conceptuels de Données (MCD) auparavant, j'étais un peu intimidé. Heureusement, le Ministère avait organisé une série de formations internes sur PostgreSQL et la modélisation de bases de données. J'ai suivi ces sessions avec enthousiasme, prenant des notes détaillées et posant de nombreuses questions.

Après la formation, j'ai commencé à travailler sur les MCD avec l'aide de collègues plus expérimentés. Nous avons utilisé des outils de modélisation spécialisés pour esquisser les relations entre les différentes entités et attributs. C'était fascinant de voir comment un schéma bien conçu pouvait simplifier des processus complexes et améliorer l'efficacité de notre système.

Au fur et à mesure que le projet avançait, j'ai eu l'opportunité de mettre en pratique mes nouvelles compétences. J'ai participé à plusieurs réunions de revue de conception, où j'ai présenté mes MCD à des parties prenantes clés, y compris des chefs de département et des experts techniques. Leurs retours ont été précieux pour affiner mes modèles et les aligner avec les objectifs stratégiques du Ministère.

Finalement, nos MCD ont été intégrés dans la base de données PostgreSQL que nous avons développée. Le projet "Cod'It" a été un grand succès, en grande partie grâce à une planification et une modélisation soigneuse. Pour moi, ce fut une expérience d'apprentissage inestimable qui m'a non seulement enseigné comment créer des MCD, mais aussi comment ils peuvent être utilisés pour résoudre des problèmes réels dans un environnement professionnel.

# Programmation de Cod’It

## 8.1 Début : échec de la programmation en python

Logo du langage python



Lorsque j'ai été affecté au projet "Cod'It" au Ministère du Travail, l'objectif était clair : créer un logiciel qui rendrait le Code du Travail Français plus accessible et ergonomique, en le présentant sous un angle thématique. Python étant réputé pour sa facilité d'utilisation et sa polyvalence, il semblait être le choix idéal pour ce projet.

J'ai commencé par établir un environnement de développement Python et installer plusieurs librairies qui, je pensais, seraient utiles. Parmi elles, Flask pour le backend, SQLAlchemy pour la gestion de la base de données et React pour le frontend.

Tout semblait bien se passer jusqu'à ce que je commence à intégrer ces différentes librairies. C'est là que les problèmes ont commencé à surgir. Des conflits de versions aux incompatibilités entre les librairies, chaque étape devenait de plus en plus compliquée. J'ai passé des jours à essayer de résoudre ces problèmes, mais sans succès.

Après plusieurs réunions avec l'équipe et les parties prenantes, nous avons pris la décision difficile mais nécessaire d'écarter Python comme langage de programmation pour le projet "Cod'It". Les problèmes d'implantation des librairies étaient devenus un obstacle insurmontable, et il était clair que nous perdions un temps précieux.

Nous avons finalement opté pour une autre pile technologique, plus adaptée aux besoins spécifiques de notre projet. Cela a nécessité une période d'adaptation, mais nous avons réussi à rattraper le temps perdu et à mettre le projet sur la bonne voie.

Bien que mon expérience avec Python pour ce projet ait été difficile, elle a été incroyablement instructive. Elle m'a appris l'importance de la planification minutieuse et de la prise en compte de tous les aspects techniques avant de s'engager dans un chemin particulier. Et même si nous avons dû abandonner Python pour ce projet, les leçons apprises ont été inestimables pour la réussite future de "Cod'It".

Après avoir écarté Python, nous avons dû réévaluer nos options technologiques. Plusieurs langages et frameworks étaient sur la table, mais Node.js s'est rapidement imposé comme un choix solide pour plusieurs raisons.

Premièrement, Node.js est particulièrement performant pour les applications en temps réel et à fort trafic, ce qui était crucial pour "Cod'It" étant donné que nous visions une large audience d'utilisateurs. De plus, Node.js utilise JavaScript, un langage que toute notre équipe maîtrisait déjà, ce qui a facilité la transition.

Deuxièmement, l'écosystème npm (Node Package Manager) offre une vaste gamme de modules et de bibliothèques prêts à l'emploi, ce qui pourrait accélérer notre développement. Contrairement à notre expérience avec Python, où les librairies ont posé des problèmes d'intégration, les modules npm sont souvent conçus pour être hautement compatibles et modulaires.

Troisièmement, Node.js nous a permis de créer un environnement de développement unifié. Étant donné que JavaScript serait utilisé à la fois sur le client et le serveur, cela a simplifié notre flux de travail et a rendu la maintenance du code beaucoup plus gérable.

Après avoir pris la décision, nous avons rapidement mis en place un environnement de développement Node.js et commencé à explorer les bibliothèques et les frameworks qui seraient les plus utiles pour "Cod'It". Nous avons opté pour Express.js pour le backend et avons continué à utiliser React pour le frontend, ce qui s'est avéré être une combinaison puissante.

La transition vers Node.js n'a pas été sans défis, mais elle s'est avérée être la bonne décision. Le projet "Cod'It" a non seulement rattrapé son retard, mais a également gagné en efficacité et en performance. Finalement, le logiciel a été lancé avec succès, recevant des éloges pour sa facilité d'utilisation et sa robustesse.

En résumé, bien que notre parcours initial avec Python ait été semé d'embûches, le passage à Node.js a été un tournant décisif qui a grandement contribué à la réussite du projet "Cod'It".

## 8.2 Recentrage de la programmation vers NodesJS

Le processus de redéveloppement du logiciel "Cod'It" en NodeJS a été une entreprise détaillée et structurée. Nous avons abordé cette transition en nous appuyant sur les leçons tirées de notre expérience avec Python et en nous concentrant sur une intégration fluide et efficace des technologies.

**Étape 1 : Analyse et Planification**

La première étape a consisté à analyser en profondeur les exigences du projet et les défis que nous avions rencontrés précédemment. Nous avons cartographié les fonctionnalités clés, les attentes des utilisateurs et les objectifs de performance. Chaque membre de l'équipe a été impliqué dans ce processus, garantissant une perspective diversifiée et une compréhension complète des défis à venir.

**Étape 2 : Formation et Adaptation**

Bien que notre équipe ait été familiarisée avec JavaScript, une session de formation intensive sur les spécificités et les meilleures pratiques de Node.js a été organisée. Cela nous a aidés à nous adapter rapidement au nouvel environnement de développement et à tirer le meilleur parti des fonctionnalités offertes par Node.js.

**Étape 3 : Développement Agile**

Nous avons adopté une approche de développement agile, avec des sprints courts et des réunions quotidiennes pour discuter des progrès, des défis et des ajustements nécessaires. La modularité de npm nous a permis d'intégrer facilement des bibliothèques et des outils, évitant les problèmes d'incompatibilité qui avaient entravé notre progression avec Python.

**Étape 4 : Tests Intensifs**

Un accent particulier a été mis sur les tests. Des tests unitaires, des tests d'intégration et des tests de charge ont été effectués à chaque étape du développement pour s'assurer que le logiciel était non seulement fonctionnel, mais aussi performant et fiable.

**Étape 5 : Optimisation Continue**

Node.js a facilité l’implémentation d’une culture d’optimisation continue. Grâce à son écosystème dynamique et à la facilité de mise à jour des modules, nous avons pu améliorer constamment le logiciel, en ajoutant de nouvelles fonctionnalités et en optimisant les performances existantes.

**Résultat**

"Cod'It" en Node.js est devenu un produit robuste, capable de gérer un trafic élevé et d'offrir une expérience utilisateur optimisée. Le code propre, la maintenance simplifiée et la performance améliorée ont validé notre décision de migrer vers Node.js. Nous avons non seulement surmonté les obstacles initiaux mais avons également créé un produit qui a dépassé les attentes.

**Réflexion**

La flexibilité et l'adaptabilité sont des éléments clés dans le monde du développement logiciel. Bien que le passage de Python à Node.js ait été un défi, c'était une opportunité d'apprentissage et de croissance. Nous avons appris que la sélection d'une pile technologique ne doit pas seulement être basée sur la popularité ou la familiarité, mais doit être une décision bien réfléchie, alignée avec les exigences spécifiques et les défis uniques de chaque projet.

Chaque échec et succès dans le parcours de "Cod'It" a renforcé notre équipe, affiné notre approche du développement logiciel et nous a préparés à surmonter les défis futurs avec une résilience et une expertise renouvelée. "Cod'It" est désormais un témoignage de notre engagement envers l'excellence, l'innovation et la résilience face aux obstacles.

## 8.3 Exercice 0.1



Cet exercice sert à récupérer des données dans des fichiers csv en entrée (il y en a cinq) pour unifier les flux d’information en une seule sortie. C’est comme l’Exercice 1 que j’explique juste en bas. Tout cela implique que l’on ait des colonnes dans les fichiers csv qui coïncident. Pour récupérer ces données, on utilise le composant « tFileInputDelimited », pour unir ces données, on utilise le composant « tUnite ».

Ensuite, dans le but de s’exercer, on va défaire ce que l’on vient de faire pour ramener les données à leur état d’origine. Pour ce faire, il faut utiliser le composant « tUniqRow », on pourra scinder le flux unifié de données en autant de flux de sortie que l’on veut, en l’occurrence deux. On récupère ces flux pour les mettre dans deux fichiers Excel en sortie et l’exercice est terminé.

## 8.4 Exercice 1



Ce Job Talend a pour but de m’exercer afin de m’acclimater au logiciel Talend. En effet, ce logiciel étant très utilisé, il a fallu m’exercer à utiliser des technologies que je ne maitrisais pas forcément. Le projet est d’apparence très simple, du haut de l’expérience que j’ai acquis depuis le début de mon alternance. En effet, dans cet exercice, nous avons deux fichiers csv en entrée, et nous souhaitant fusionner une partie des données qu’il y a à l’intérieur des fichiers csv. En effet, un fichier csv s’apparente à une base de données car l’en tête des feuilles corresponds aux noms des différentes colonnes, chaque feuille du csv représente une table et les lignes du tableur représentent les lignes des tables d’une base de données. On peut ainsi très bien utiliser un ETL pour traiter des données dans des fichiers csv. C’est ainsi que nous avions des colonnes qui coïncidaient l’une avec l’autre dans ces deux fichiers csv et nous voulions donc les unifier dans un même fichier csv. Ainsi, pour capter les données nous utilisions naturellement le composant « tFileInputDelimited » deux fois pour faire entrer les données en provenance de la base de données. Ensuite, pour unifier les flux, nous utilisons un composant Talend spécialisé nommé « tUnite », qui va unifier des données qui correspondent à une colonne en doublon dans deux référentiels. Cela va générer un flux unique de sortie que l’on captera et que l’on dirigera vers un composant fait pour transcrire les données du flux dans un fichier csv, ce composant est « tFileOutputDelimited ».

## 8.5. A Injecter Modif



Le but de ce job Talend est d’injecter des données venant de fichiers CSV dans une base de données que ma hiérarchie m’a donné. Le but est d’extraire les données des fichiers CSV en entrée, on a ainsi utilisé le composant Talend « tFileInputDelimited ». Mais il faut également réaliser une table pour que le format soit compatible avec l’injection dans la base de données, il faut donc utiliser le composant « tCreateTable », comme on a deux fichiers csv en entrée, on duplique les deux opérations. Après cela, on peut utiliser les composants « tDBOutput » pour injecter les données dans la base de données.

## 8.6 OPE



Ce job Talend sert à récupérer des données dans des tables de la base de données et à les unifier pour ensuite les mettre dans un fichier csv en sortie. En effet, nous avons utilisé les composants « tDBInput » pour récupérer les informations contenues dans deux tables, ensuite nous les avons mises en relation avec le composant « tMap » pour envoyer les deux flux en entrée de ce composant dans un même flux de sortie. Cela constitue une solution technique originale car on aurait très bien pu utiliser le composant « tUnite » qui aurait réalisé le travail, mais cette solution a permis d’explorer les différentes fonctionnalités du mappage sur Talend et je tire une satisfaction de cela.

# Suivi Camus

Cette mission m’a été assignée à la fin de mon alternance, et est à l’heure où j’écris, toujours en cours d’exécution. Elle vise à produire un programme en PHP. J’en profite pour dire que mon alternance exige une certaine polyvalence, car l’intitulé de mon offre ne spécifie pas une technologie particulière à réaliser, mais un projet sur lequel travailler, ce qui change tout car je dois maitriser toutes les technologies qui se rapportent de près ou de loin à ce projet. Ce qui exige une certaine polyvalence, car je n’avais pas l’occasion de travailler sur du PHP avant dans mon alternance. Je trouve que c’est un point important dans l’apport de mon alternance à ma formation professionnelle. C’est ainsi que le programme PHP que j’ai dû réaliser a pour but de modifier un autre fichier, un fichier JavaScript cette fois-ci.

# Conclusion

Ainsi, au terme de ce mémoire, nous avons analysés les différents outils que j’utilise pour travailler au Ministère, après avoir présenté le Ministère de l’Economie, des Finances et de la Relance pour lequel je travaille et avoir présenté le projet Camus sur lequel je travaille. Pour ensuite aborder les jobs Talend sur lesquels je travaille.

Sur le plan personnel les différents projets sur lesquels j’ai travaillé durant cette année de formation au sein du MEFR m’ont apporté de nombreuses satisfactions.

En effet pour réaliser ces différentes actions j’ai réalisé de nombreuses recherches d’informations aussi bien dans les différentes documentations techniques mises à ma disposition qu’au contact des chefs de projets et experts avec lesquels j’ai été amené à travailler.

Ces projets m’ont permis de développer et de compléter mes connaissances techniques en m’imprégnant des paramètres spécifiques liées aux réalités de terrain.

Enfin, sur un plan plus général, le projet Camus qui m’a été proposé m’a permis de mieux cerner la fonction d’apprenti au sein du projet Camus.

J’ai ainsi découvert la réalité de sa mission en réalisant un travail de fond basé sur les tâches quotidiennes mais également en réalisant, en amont, un travail de planification pour chaque intervention plus exigeante.

Nous avons ainsi répondu à la problématique posée sur l’application de la science informatique au Ministère de l’Economie, des Finances et de la Relance.