|  |
| --- |
| SMILE - EPSI |
| Dossier professionnel RNCP 35584 - Expert en Informatique et en Système d’Information |
| Promotion 2021-2023 |

|  |
| --- |
| Ryan Murati  08/01/2023 |

Table des matières

[I- Remerciements 2](#_Toc133768782)

[II- Introduction 3](#_Toc133768783)

[III- Environnement professionnel 4](#_Toc133768784)

[A) Présentation de l’entreprise SMILE 4](#_Toc133768785)

[i) Organisation de la structure 4](#_Toc133768786)

[ii) Présentation du SI, systèmes informatiques 4](#_Toc133768787)

[B) Activités de la structure 4](#_Toc133768788)

[i) Les différentes équipes de SMILE 4](#_Toc133768789)

[ii) Les objectifs des activités 5](#_Toc133768790)

[iii) Les clients réguliers 5](#_Toc133768791)

[C) Mon activité à SMILE 5](#_Toc133768792)

[i) Place dans les équipes 5](#_Toc133768793)

[ii) Missions confiées 5](#_Toc133768794)

[IV- Valorisation des compétences 6](#_Toc133768795)

[A) Projet WSO2 6](#_Toc133768796)

[i) Explication en détail du projet 6](#_Toc133768797)

[ii) Quelles compétences validées par le projet ? 6](#_Toc133768798)

[iii) Renseignement sur la compétence et bonnes pratiques 6](#_Toc133768799)

[B) Projet INRAE 7](#_Toc133768800)

[i) Explication en détail du projet 7](#_Toc133768801)

[ii) Contexte de réalisation, pourquoi avoir besoin de cette compétence dans l’activité? Comment ça a été réalisé et assimilé ? Quelles ressources utilisées ? As-t’elle été traversée par des évolutions majeures ces 10 dernières années ? 7](#_Toc133768802)

[iii) Renseignement sur la compétence et bonnes pratique 7](#_Toc133768803)

[C) Projet Java 7](#_Toc133768804)

[i) Explication en détail du projet 7](#_Toc133768805)

[ii) Contexte de réalisation, pourquoi avoir besoin de cette compétence dans l’activité? Comment ça a été réalisé et assimilé ? Quelles ressources utilisées ? As-t’elle été traversée par des évolutions majeures ces 10 dernières années ? 7](#_Toc133768806)

[iii) Renseignement sur la compétence et bonnes pratique 7](#_Toc133768807)

[V- Contexte de réalisation 8](#_Toc133768808)

[VI- Conclusion 9](#_Toc133768809)

[A) Résumé 9](#_Toc133768810)

[B) Ouverture sur l’avenir 9](#_Toc133768811)

# Remerciements

Je tiens à remercier toutes les personnes qui m’ont accompagnées lors de mes deux années en alternance à SMILE, ainsi que toutes celles qui m’ont aidé lors de la rédaction de ce rapport. En premier lieu, j’adresse mes sincères remerciements à **Sophie DEMOIS**, ainsi que toute l’équipe pédagogique de l’EPSI, dont le suivi et l’aide apportée a été d’une grande ressource lors de la rédaction de ce rapport. J’aimerai ensuite remercier mes deux manageuses de SMILE, **Cécile WALKENER** et **Thi** **thuy Linh LE**, qui m’ont suivi tout au long de mon alternance, et m’ont introduit les collaborateurs avec lesquels j’allais travailler.

Je remercie également toutes les personnes du pôle GED de SMILE, notamment **Nicolas Vezian**, qui ont su m’intégrer à l’entreprise lors de mon arrivée, et qui m’ont aidé durant l’entièreté de mon alternance lorsque j’avais des questions ou problèmes.

Un grand merci à mes anciens équipiers de l’INRAE, particulièrement **Nicolas Brun** avec qui j’ai pu travailler pendant 8 mois, et qui m’ont formé avec patience et bienveillance.

# Introduction

Vous vous trouvez au début de ce rapport de professionnalisation.

Ce dossier sera composé d’une présentation de l’entreprise SMILE, de son SI, de ses équipes et des missions qu’elle propose en général.

Je décrirais aussi quelles compétences j’ai acquis lors des projets que j’ai mené dans l’entreprise, en décrivant les différentes tâches effectuées, les difficultés que j’ai pu rencontrer et comment je les ai surmontées, tout en gardant un regard critique sur les possibles améliorations de mon travail.

Lors de mon arrivée à SMILE, j’ai été placé dans une équipe de deux personnes, afin de me former sur une technologie que je ne connaissais pas. Après quelques mois sur cette technologie, j’ai proposé de développer un logiciel de démonstration, permettant de présenter aux potentiels clients les possibilités que la technologie offrait.

Après la réalisation de ce projet solo, j’ai rejoint l’INRAE en régie, supposément pendant un an, afin de me former sur leurs technologies et leurs projets pendant 6 mois et être autonome les 6 mois suivants.

Après mes 6 mois de « formation », je me suis rendu compte que je projet ne me stimulais pas, j’ai donc fait une demande a l’INRAE et a SMILE afin de changer de projet plus tôt, ce qu’ils ont accepté après délibération. J’ai donc continué deux mois à l’INRAE avant de rejoindre un projet le 01/03/2023 pour le gouvernement, dans une équipe de 7 personnes en Agilité SCRUM.

Je parlerais aussi, pour démontrer l’acquisition de certaines compétences, d’anciens projets effectués lors de mon cursus d’apprentissage.

# Environnement professionnel

## Présentation de l’entreprise SMILE

### Organisation de la structure

SMILE est une société spécialisée dans le développement de solutions open source. Crée en 1991 à Paris, elle est maintenant implantée dans 17 agences réparties dans 9 pays différents. Offrant des solutions à des clients importants comme la préfecture de Lyon, la SNCF, etc.

La société emploie plus de 2000 collaborateurs, et l’agence de Lyon est composée de 150 collaborateurs, dispersés en différentes équipes que nous retrouverons plus tard dans le dossier.

Voici l’organigramme de la structure lyonnaise lors de mon arrivée, **[montrant les différentes équipes et les personnes dirigeant ces équipes] :**

**ORGANIGRAMME**

Le groupe SMILE m’a embauché en octobre 2021 en tant que développeur java apprenti, et m’a confié différentes missions, que je décrirais tout au long de ce rapport.

### Présentation du SI, acteurs, informations et flux

## Activités de la structure

### Les différentes équipes de SMILE

Comme présenté précédemment, SMILE contient plusieurs équipes et travaille sur plusieurs projets différents. Cette partie va détailler les objectifs et les missions des équipes proches de mon travail.

Les équipes peuvent-être séparées en trois pôles :

**Data**

Les équipes dans le pôle data suivent des projets dont le but est de mettre en œuvre un système de gestion de données. Cela se traduit par :

* De la relation client afin de trouver quels sont les bonnes technologies pour implémenter ce système de gestion.
* Une création de systèmes qui correspondent aux demandes clients, cherchant à convenir au mieux aux besoins, en restant le plus optimisé possible.
* Un accompagnement client lors de la restitution du projet.

Les collaborateurs de ces équipes peuvent être amenés à utiliser diverses technologies comme *Talend*, *Power BI*, *Jedox* ou encore des API Manager tel que ***WSO2 API Manager.***

**Java/Backend**

Le pôle Backend est composé de multiples équipes, travaillant sur de nombreux projets, pour des clients ou en interne. Les collaborateurs de ces équipes fonctionnent en Agilité (SCRUM) dans des équipes d’environ 6 à 8 personnes constituées de développeurs backend, frontend, de testeurs, et d’un ScrumMaster.

Leur travail consiste à développer de nouvelles features sur un projet donné en Java, de participer aux créations des US (User Stories), faire de la relecture de code ainsi que de participer aux temps d’équipe agile.

**Régies**

Il est possible que des clients fassent appel a SMILE afin de trouver des personnes pour leur projet. Dans ce cas, le collaborateur smilien est « envoyé » chez le client afin de travailler dans leurs équipes. On appelle cela un projet en régie. C’est une coutume assez fréquente chez SMILE que j’ai pu effectuer pour rejoindre l’INRAE (**I**nstitut **N**ational de **R**echerche pour l’**A**griculture l’alimentation etl’**E**nvironnement).

Bien qu’ils existent, je ne vais pas détailler les autres pôles dans lesquels je n’ai pas activement participé à leur développement, voici cependant une liste des autres pôles existant chez SMILE :

Javascript,Drupal, PHP, E-commerce, GED/ECM, Fonctionnel et testing, Project Management

### Les objectifs des activités

Gagner de l’argent, fidélité client, utiliser les compétences des collaborateurs à bon escient

### Les clients réguliers

Grâce à son expertise et a son ancienneté, SMILE à pu obtenir la confiance de nombreuses entreprises et intervient donc chez de nombreux clients différents. Certains font régulièrement appel aux services de SMILE, voici une présentation non exhaustive de ces clients :

* beIN SPORTS

Le défi de beIN SPORTS était de créer un site web flexible capable d’assurer la publication de contenus multilingues et multisites, permettant à 60 journalistes et éditorialistes, répartis dans 4 salles de rédaction, de pouvoir facilement contribuer sur le site.

Après 2 ans de développement SMILE produit une plateforme open source développé dans 4 pays et 7 langues, permettant l'intégration de nombreux prestataires.

Avec ce site SMILE remporte en 2015 en Symphony Awards du « *best B2C website of the year* »

* Suez

Lors de leur affiliation, pendant 2 ans SMILE intervient chez Suez sur les technologies *Symphony* et *eZ publish.*

Dans cette période Suez recherche des experts afin de concevoir une plateforme web optimisée en seulement un mois. Ce projet sera donné à une équipe Smilienne, qui développera la solution en seulement 30 jours.

Un an après ce projet Suède décide d'accorder sa confiance à SMILE en lui confiant la conception et la maintenance de ces 3 sites web.

* EDF

Depuis 2016 SMILE occupe la fonction d'experts référent au sein du département SPENCER – ESA (département centralisé des besoins informatiques du groupe).

Actuellement plus de 60 tickets sont gérés chaque mois sur des technologies diverses et variées.

* Total

SMILE et total sont en lien depuis de nombreuses années. En effet 4 équipes sont mobilisées en parallèle contenant 20 collaborateurs, 7 projets sont réalisés par année avec total et plus de 500 tickets sont traités chaque année.

* Vitam

Vitam est un projet demandé par l'État français afin de créer un socle commun d'archivage open source permettant la facilité d'accès et la documentation. Missionnés sur le projet, SMILE se lance dans la conception d'un projet hautement sécurisé réutilisable et pérenne.

Engagé sur 20 ans ce fait le choix de la flexibilité technologique c'est-à-dire que la solution vitam est conçue de façon modulaire permettant l'optimisation ou la modification de certains pans de l'application

## Mon activité à SMILE

### Place dans les équipes

Chez SMILE, j’ai été placé dans 3 équipes différentes que j’expliquerais plus en détail dans la partie IV. Voici malgré tout une courte introduction à ma place dans ces-dites équipes :

*Equipe WSO2*

Lors de mon projet sur *WSO2 API Manager (APIM ou AM)* j’étais apprenti **« API Constructer »** dans une équipe composée de 2 personnes à mon arrivée.

*Régie INRAE*

Pendant ma mission en régie a l’INRAE, j’ai rejoint une équipe de 7 personnes afin de développer en Agilité une solution de gestion de rôles et habilitations utilisant la solution *OpenIDM* de forgerock.

*Equipe Java*

J’ai rejoint un projet java en tant que second développeur Backend. Ce projet consiste en un développement d’une application web permettant la création de formulaires lors d'accidents entre véhicules contenant des matières dangereuses. Ce projet est destiné au ministre de l’Écologie et a été développé par une équipe de 8 personnes : 2 développeurs backend, 4 développeurs front-end un Scrum Master et notre Product Owner.

### Missions confiées

Création d’API, apprentissage WSO2, Développement solution démo client en java WSO2

INRAE = OpenIDM forgerock, création d’API, gestion de rôles et habilitation, contact client

Java : création de formulaire pour le gvt recensant les accidents de transports contenant des matières chimiques

Puis autre projets (CGI MCO puis dev projet Xénon – Canada – MSPR)

# Valorisation des compétences

## Projet WSO2

### Explication en détail du projet

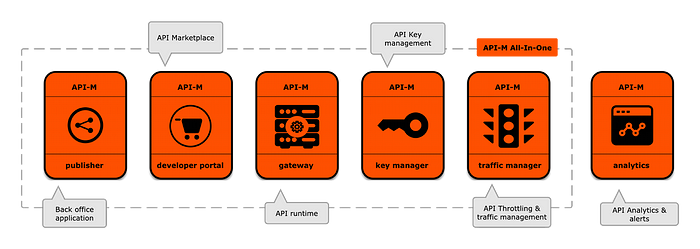
Lors de mon arrivée chez SMILE, contrairement à ma fiche de poste (développeur java junior), je n’ai pas rejoint un projet java. En effet, j’ai été placé dans une équipe composée de deux personnes (Stéphane et Lihn), dont l’expertise était le logiciel *WSO2 AM*.

Le but de cette équipe « WSO2 » était d’installer, chez des clients demandant, la solution WSO2 AM, de former les collaborateurs de ces clients afin de leur permettre de comprendre le logiciel ainsi que de modifier les APIs de ces clients pour qu’elles soient utilisables par WSO2 AM ; c’est-à-dire en format swagger API.

**Qu’est-ce qu’est *WSO2 API Manager* ?**

*WSO2 AM* est un gestionnaire d'API, c’est-à-dire un logiciel permettant aux entreprises de créer, publier et gérer des APIs. Il prend en charge la gestion du cycle de vie des API, le développement d'applications, le contrôle d'accès, la limitation de débit et l'analyse dans un seul système. L’exposition des APIs est gérée via un outil d’édition et de publication ; et la consommation via une plateforme sur laquelle l’utilisateur pourra s’inscrire aux APIs souhaitées. Le transfert d’information est réalisé via une passerelle (Gateway) permettant l’interopérabilité des SI des deux parties (éditeur et souscripteur).

Voici un schéma d’explication :



Comme dit précédemment, j’ai rejoint ce projet dès mon arrivée à SMILE malgré mon titre de développeur java junior. J’ai donc dû me former sur cette solution avant de pouvoir moi aussi entamer des processus d’installation chez les clients.

Pour cela, j’ai suivi une auto-formation, interne a SMILE, pendant 6 semaines, afin d’apprendre au mieux le fonctionnement de la solution. Pendant cette formation, j’ai appris à créer et utiliser des APIs en swagger, les ajouter a WSO2 AM, créer des « applications » grâce à ces APIs, créer des rôles, des gateways et à sécuriser les APIs.

**[PARLER DE LA FORMATION, QUOI D’APPRIS, QUOI DE COMPLEXE, OVERCAME SMTH ?]**

J’ai par ailleurs rédigé aux alentours de décembre 2021 un article interne pour SMILE, expliquant comment configurer Auth0 comme « Key Manager ». Vous pouvez le retrouver en Annexe.

Cette présentation m’a permis d’affirmer la fin de mon autoformation et de lancer mon autonomie sur les différents projets qui pourraient nécessiter ma présence.

Malheureusement, après quelques peaufinages de l’article en lien avec le responsable WSO2 de SMILE, Gregory Eve, les clients demandant une installation WSO2 APIM se fîtes de plus en plus rare, et les projets que mes collègues avaient nécessitaient des compétences dans des technos que je ne maitrisais pas, comme Talend ou Elasticsearch.

J’ai donc demandé un point avec mon manager, mes coéquipiers ainsi que Gregory, afin de leur proposer le développement d’une application en **Java**, utilisant l’**api manager** de **WSO2**, qui permettrai de faire des démonstrations des possibilités d’utilisation de WSO2 APIM dans un contexte de projet.

Cette proposition a été très bien reçue par mon manager et Gregory, puis après quelques délibérations nous sommes venus à un objectif plus précis : développer une application en java, permettant de démontrer l’utilisation de 3 APIs intégrés sur une gateway WSO2 APIM.

J’ai ainsi pu commencer le développement de mon projet, Interne à SMILE et en solo. Pour commencer j’ai effectué un benchmark des API que je pouvais utiliser, voulant rester dans un contexte professionnel et en lien avec l’essence de SMILE, je me suis dirigé vers des APIs OpenSource, des *Open APIs.*

Lors de ce benchmark, 3 API sont ressorties par leur praticité et le fait qu'elle soit open source.

La première API que j'ai choisie est une API de découpage administratif appelée **API Géo**, provenant du site api.gouv.fr, qui référence des informations sur toutes les villes françaises (comme son code INSEE, sa superficie, son nombre d'habitants, etc.) ; Ensuite, j’ai inclus au projet **CoronavirusTracker**, une API postée sur github, qui référence l'évolution de l'épidémie de COVID-19 dans le monde ou dans un pays demandé ; Enfin, pour ajouter des informations à la ville choisie par l'utilisateur, j'ai intégré **WeatherAPI**, qui permet d'avoir des informations sur la météo sur une ville donnée grâce aux retours de l’API Géo.

Ces 3 API me semblaient être un choix adapté et professionnels pour une démonstration client, et après présentation à mon équipe, ce choix a été validé. J'ai malgré tout abandonné une API lors de mes recherches, en effet la première appli que j'avais choisie pour l'épidémie de COVID-19 ne correspondais pas aux besoins du projet.

Cette étape de maquettage et de benchmark m’a pris environ deux semaines, et, dès l'accomplissement de la maquette, j'ai réuni mes 2 équipiers ainsi que mon manager afin d'avoir un retour avant de commencer le développement.

Voici à quoi ressemblait la maquette acceptée par mon équipe :

J’avais donc prévu que l’application soit constituée de deux écrans :

-Le premier contenant une liste déroulantes des villes de France, permettant à l’utilisateur de choisir sur quelle ville il voudrait un focus d’informations.

-Le second serait constitué de 3 parties distinctes, la première, en vert, contiendrait les informations « basiques » de la ville choisie, c’est-à-dire son nombre d’habitant, sa surface, sa région et ses code postaux (car une ville de France peut avoir plusieurs codes postaux, par exemple Metz en a trois). La deuxième partie, en bleu, contiendrait des informations météorologiques de la ville, les températures maximum et minimum, la vitesse des vents, l’humidité, etc. Et la troisième partie, en rouge, contiendrait les informations relatives aux Covid-19, l’épidémie encore assez présente dans la période de réalisation du projet.

Après l'étape de maquettage de benchmark j'ai dû intégrer les API que j'avais choisies dans une instance locale de WSO2 APIM, afin de me permettre d'exporter ces API en jdk.

Cela est possible grâce à WSO2, qui convertit automatiquement les fichiers swagger des APIs en fichiers utilisable en Java, python ou javascript (selon le choix du développeur).

Une fois les ressources des API exportées en classes Java, j'ai pu commencer le développement de mon logiciel en *Spring MVC*, une technologie que ma collègue Lihn m’avais conseillé pour ce genre de projet.

Voici une brève explication du fonctionnement de *Spring MVC*.

Comme son nom l'indique Spring MVC utilise le design pattern MVC (modèle-vue-contrôleur) afin de créer des applications web en Java Avec le Framework Spring.

Le design pattern MVC est constitué de trois éléments : le modèle qui définit la structure de la donnée, la vue qui est l’élément que nous allons présenter à l'utilisateur (l’UI), et le contrôleur qui contient les fonctions qui modifie les données de la vue et du modèle.

Au centre de *Spring MVC* se trouve la dispatcher servlet, qui gère toutes les requêtes et réponses http, c’est le contrôleur principal de notre projet. Dans les modèles, nous allons retrouver Les données qui seront transmises vers les vues. Les vues, dans *Spring MVC*, sont implémentées en Java ServerPages (JSP), conçues pour être implémenté de manière optimisée.

Lors de la création de mon projet, je n'avais jamais utilisé la solution *Spring MVC*, j'ai donc en premier lieu suivi un tutoriel de spring.io afin de me familiariser avec la technologie.

**[J’AI DONC COMMENC2 PAR CREER MES CONTROLLEURS , BLAH BLAH BLAH]**

En premier lieu, j'ai créé différentes méthodes permettant, grâce aux classes créées par WSO2, de faire appel aux API citées ci-dessus grâce à la ville choisie par l'utilisateur dans le premier menu. Pour faire un appel le plus précis possible, j'avais l'idée de tout d’abord chercher la ville que l'utilisateur choisissait en fonction des données qu'il mettrait dans le formulaire de la première page. Afin que cela soit possible, Il me fallait un fichier contenant toutes les villes de France ainsi que certaines informations les concernant, permettant d’effectuer une recherche dans ce-dit fichier en fonction de ce que l'utilisateur entrais comme valeur. En cherchant brièvement sur github, j'ai trouvé un fichier Json d'environ 200000 lignes, composé de l'entièreté des villes françaises ainsi que des informations relatives à ces villes comme le code INSEE, le code postal, le nom de la commune, etc. (le lien du repo github est en annexe).

Lorsque j'ai trouvé ce fichier il me fallait faire un choix : fallait-il que je crée une base de données à partir du Json, contenant toutes les villes de France et leurs informations, ou était-il possible de garder ce fichier tel quel, malgré la probabilité d'un temps de latence lors de la recherche à cause du parsing d’un fichier très lourd.

Je n'ai pas voulu prendre ce choix seul, j'ai donc demandé à mes collègues quelle solution leur semblait adéquate dans le contexte de mon projet. Après délibération nous avons convenu que de garder le fichier tel quel ne serait pas une bonne idée mais qu'une base de données serait peut-être excessive pour un projet de démonstration. La solution que nous avons envisagée est donc la suivante : modifier le fichier json afin de garder les 10000 plus grandes villes de France, réduisant grandement le temps de charment lors du parsing du fichier, tout en gardant la possibilité d’effectuer des recherches des plus grandes villes de France.

Nous avions aussi pensé à stocker le fichier Json en cache dans le navigateur, mais cette solution n'était pas envisageable à cause du fait que l'application ne serait lancée que très rarement pour les démonstrations (c'est à dire que le stockage en cache devra se faire trop régulièrement pour que ce soit efficace).

Afin donc de modifier le fichier automatiquement, j'ai créé un script dont le but était de supprimer toutes les lignes après les dix millièmes plus grandes villes. À noter que par « grande » j'entends ici contenant le plus grand nombre d'habitants. Le script est retrouvable en annexe.

Le fichier ainsi créé me permet donc de récupérer les informations précise d'une ville demandée par l'utilisateur dans le premier formulaire de choix. Cette récupération s'organise grâce à une fonction de recherche via un parsing json en fonction des attributs que l'utilisateur à rempli dans le formulaire.

Pendant ce projet malgré le fait que je sois seul, mes collègues, Lihn et Stéphane, m’ont été d'une grande aide et m’ont donné un regard objectif sur mes idées et mes états d’avancement sur le projet.

Implication personnelle dans le projet, mon apprentissage et ce que j’ai pu produire

### Quelles compétences validées par le projet ?

### Renseignement sur la compétence et bonnes pratiques

## Projet INRAE

### Explication en détail du projet

### Contexte de réalisation, pourquoi avoir besoin de cette compétence dans l’activité? Comment ça a été réalisé et assimilé ? Quelles ressources utilisées ? As-t’elle été traversée par des évolutions majeures ces 10 dernières années ?

### Renseignement sur la compétence et bonnes pratique

## Projet Java

### Explication en détail du projet

### Contexte de réalisation, pourquoi avoir besoin de cette compétence dans l’activité? Comment ça a été réalisé et assimilé ? Quelles ressources utilisées ? As-t’elle été traversée par des évolutions majeures ces 10 dernières années ?

### Renseignement sur la compétence et bonnes pratique

# Contexte de réalisation

# Conclusion

## Résumé

## Ouverture sur l’avenir

# Annexe

Lien de l’article Auth0: <https://docs.google.com/document/d/1ix73uuw6ou-B6POphNXcAZHEMKA5nOYnNBFuzg0x0R8/edit#heading=h.rozdwj5vcws4>

Lien repo github : <https://github.com/high54/Communes-France-JSON>