



Rapport de stage

Développeur front

Augustin Marie
10 mai – 25 juin

Entreprise : All You Need
Tuteur : M. Maxime Zanchi

Je tiens tout d'abord à remercier **Maxime Zanchi** pour sa confiance, pour la liberté qu'il m'a offert de me former et de me documenter sur différents sujets au cours du stage. Je remercie aussi plus largement toute l'équipe d'All You Need pour leur accueil.

Je remercie également **Rémi Staelen**, stagiaire lui aussi qui a développé le back de l'application pour m'avoir recommandé auprès de Maxime Zanchi. Je joins à lui **Julien Vanhauwaert**, je les remercie pour les 7 semaines de travail passées ensemble. Sans leur sérieux, le projet n'aurait jamais pu arriver au stade où il est aujourd'hui.

Sommaire

Introduction	4
I. Présentation de la mission	6
A-Contexte professionnel de la mission	6
B-Problèmes à résoudre, contraintes et limites	7
II. Organisation du travail	9
A-Processus et méthode suivie	9
B-Organisation des journées	10
III. Réalisation des missions	11
A-La page de connexion et de création de compte	11
B-Conception de la charte graphique de l'application	12
C-Travail sur les API	14
D-Séparation claire du back et du front	17
E-Les fiches de renseignement	19
F-La création automatique de PDF	21
IV. Évaluation des compétences mobilités	22
A-Adéquation du travail	22
B-Compétences mises en œuvre	24
Conclusion	25

Introduction

En tant qu'étudiant à l'epsi, j'ai pu effectuer un stage dans l'entreprise All You Need. Stage qui a duré du 10 mai au 25 juin.

All you need est un cabinet de recrutement situé à Lille qui met en relation des entreprises et des candidats en recherche de nouveaux emplois ou d'évolution professionnelle. Il sur un accompagnement des candidats et entreprise, et sur la parfaite compréhension de leurs enjeux, de leur buts et besoins pour comprendre ce qui est vraiment important pour eux.

Mettre en contact des cadres et des entreprises ayant des projets concordant n'est pas toujours facile en passant par les moyens traditionnels. C'est pourquoi passer par un agent permet aux entreprise de s'assurer un collaborateur réellement intéressé par le projet et qui possède des compétences adaptés aux postes proposés et aux collaborateurs d'entrer dans un projet qui leur correspond au seins d'une entreprise en bonne santé dans laquelle ils pourront s'épanouir professionnellement le tout sans passer des heures à chercher dans des sites d'annonce de recrutement.

De plus, certains cadres n'aiment pas afficher clairement leur volonté de chercher un nouvel emploi pour ne pas se rendre vulnérables dans leur entreprise actuelle et de leur côté les entreprises n'expriment pas toujours publiquement toutes leurs annonces. Passer par un agent permet donc d'aller chercher des postes et des candidats qui ne sont pas visibles autrement.

Enfin d'un point de vue des candidats, All you need propose un accompagnement pour pouvoir se mettre en valeur et sublimer leur parcours lors des entretiens augmentant ainsi les chances d'obtenir le poste qu'ils désirent.

All you need s'occupe de recrutement dans plusieurs domaines avec entre autres le domaine de la santé, de l'IT, le marketing et même plus récemment dans l'environnement.

Le principal but de ce stage était de développer, avec 2 autres stagiaires, la plateforme SIRH (Système d'Information Ressources Humaines) imaginée par le directeur, Maxime Zanchi qui n'a jamais eu le temps de la développer par lui-même. L'idée était donc de faire une première version, un poc (proof of concept) fonctionnel pour voir si sa solution imaginaire marche comme il le veut dans la pratique.

Ayant appris les fondamentaux de html5 et de css3, j'ai pu communiquer avec M. Maxime Zanchi sur les compétences en développement que je possédais déjà. Le but de cette communication était pour moi d'évoluer sur les techno importantes pour moi et pour lui de voir quelles étaient les techno importantes dont je n'avais pas encore la maîtrise.

Ayant conclu qu'une connaissance du java script était nécessaire au bon déroulement du stage, j'ai suivi un formation en ligne de 15 heures financée par l'entreprise quelques semaines avant le début du stage

Dans ce rapport, j'expliquerai les étapes et les moments clés qui ont permis de développer cette première version du logiciel et ce en 7 semaines je parlerai plus en détail du projet et des fonctionnalités attendues pour le logiciel. Ensuite, je présentai l'environnement de travail et la méthode suivie pour la création des différentes fonctionnalités. Enfin je passerai par toutes les sous missions et fonctionnalités demandées en parlant des problèmes rencontrés, des solutions retenues et des raisons pour lesquelles j'ai décidé d'implémenter la fonctionnalité comme je l'ai fait. La dernière partie contiendra les compétences apprises et celles que je n'ai pas encore maîtrisées et, à l'issue de ce compte rendu, je donnerai mon point de vue sur les conditions de travail et les expériences acquises. Je parlerai ici également du futur du poc créé et de mes espérances pour mon prochain stage.

I. Présentation de la mission

A-Contexte professionnel de la mission

Pour que l'entreprise puisse tenir son rôle dans des conditions optimales, le groupe all you need a besoin de gérer candidats en entreprises efficacement. Cela passe par l'enregistrement des candidats sur une plateforme ou la création de PDF pour résumer les différents entretiens.

Un logiciel de ce type est déjà utilisé par l'entreprise, mais n'est pas une solution optimale et ce, pour plusieurs raisons. Parmi les remarques qui ont été faites sur le logiciel, on trouve les suivantes. Il peut être compliqué à utiliser, pas ou pas suffisamment intuitif ce qui fait qu'en cas d'intégration d'un nouveau membre à l'équipe all you need celui-ci doit apprendre à utiliser. Celui-ci manque aussi de certaines fonctionnalités qui aideraient beaucoup le groupe all you need.

L'objectif est donc de créer un poc, un prototype fonctionnel, d'un nouveau logiciel pour l'entreprise simple, fait sur mesure et qui appartiendrait au groupe.

Celui ci pourrait permettre aux entreprises et aux candidats de se créer un compte pour prendre un premier contact avec all you need et, à terme, en utilisant des identifiants comme linkedIn, all you need pourriez accéder très rapidement à des informations sur le profil du candidat pour gagner du temps et ne pas avoir à remplir manuellement tous les champs.

Les enjeux pour l'entreprise sont multiples. Le premier et peut-être le plus important est le gain de temps. Si dans un premier temps utiliser un logiciel mal adapté provoque une inefficacité, le fait qu'il manque des fonctionnalités désirées peuvent forcer à faire manuellement des tâches pourtant automatisables.

Par exemple, la création de PDF pour résumer les entretiens se fait manuellement alors que la plupart des informations a déjà été rempli sur la plateforme existante. C'est donc une tâche redondante et longue pour les agents all you need. De plus, avoir des fonctionnalités comme l'utilisation de linkedIn pour le pré-remplissage des champs permettrait d'aller à l'essentiel lors des entretiens et de passer plus rapidement sur les informations de base.

Un autre but pour l'entreprise est la création de statistiques. En effet, il pourrait être intéressant pour elle de savoir quel genre de profil a eu le plus de succès auprès d'une certaine entreprise ou bien encore d'accéder rapidement aux candidats respectant certaines caractéristiques (âge, technologies maîtrisées).

All you need desservant un large éventail de professions, avoir la main mise sur sa solution logicielle permettrait aussi de l'adapter d'un secteur d'activité à l'autre. Cela permettrait de facilement diviser, adapter le logiciel, ou faire des groupes de candidats / entreprises en fonction du secteur.

B-Problèmes à résoudre, contraintes et limites

Le premier à résoudre a été partir de rien d'autre que le pitch du logiciel pour créer le front de ce logiciel à partir de rien et ce en utilisant le java script que je n'avais jamais utiliser avant. Je parle par exemple de la collaboration avec les autres personnes sur le projet. Nous étions trois stagiaires et nous avons dû nous organiser pour travailler efficacement sur un sujet d'ampleur inédite pour nous (6 semaines à raison de 35 heures par semaine). Le but étant de ne pas forcer le maître de stage à répéter des informations déjà données et de savoir précisément qui devait faire quoi.

Un autre problème auquel nous nous sommes très vite heurtés est l'absence de github centralisé, le projet a très monter en complexité et il est devenu très vite compliqué de se maintenir à jour simplement en s'échangeant les différents fichiers.

Ce problème a été heureusement très vite résolu mais j'aurai l'occasion d'en reparler par la suite dans la section II.

Nous avons été aussi confronté à des problèmes liés à des connaissances que l'on avait pas encore. En effet, beaucoup des fonctionnalités du site qui nous avaient été demandées étaient des choses que l'on ne savait pas encore faire (utiliser des identifiants google, utiliser des API, des fonctionnalités non présentes dans le java script par défaut. Nous avons donc dû monter en compétences, rechercher des librairies, des API et de la documentation sur celles-ci pour surmonter les différents problèmes posés.

Enfin, la dernière difficultés était de coder une application web adaptés aux besoins et à la façon de travailler de l'entreprise alors que j'ai commencer le stage en étant complètement extérieur à celle ci, il a donc été nécessaire de beaucoup parler pour comprendre les enjeux et les fonctionnalités clés pour gagner en autonomie et avoir un rendu aussi adapté que possible.

Nous avons eu assez peu de contraintes, étant les seuls à travailler sur le sujet, nos seules consignes étaient de rédiger un code propre et commenter pour pouvoir être facilement exploitable par la suite. J'ai donc dû réécrire et réorganiser mon code plusieurs fois pour pouvoir le rendre compréhensible et lisible pour les développeurs qui feront la version définitive par la suite mais aussi pour rendre le suivi de l'évolution et la modification du site plus simple.

La deuxième contrainte était d'avoir quelque chose de fonctionnel, faire un site qui fonctionne correctement était plus important que l'implémentation de tout le cahier des charges qui avait été énoncé à l'oral. Je devais donc faire attention à bien s'assurer que les scripts que je produisais ne faisaient pas d'erreur quelque soit mes manipulations.

La principale limite à laquelle nous nous sommes confrontés est l'absence de serveurs dédiés, cela a rendu complexe le partage de la base de donnée dans un premier temps mais nous nous sommes rendu compte que cela pouvait aussi poser problèmes avec certaines technologies côté serveur.

On voulait par exemple permettre la connexion via google et demander l'autorisation pour ajouter des rendez vous automatiquement sur les calendriers mais pour faire ceci, nous devions stocker un code d'authentification sur un serveur ce qui était impossible puisque l'on n'en avait pas.

II. Organisation du travail

A-Processus et méthode suivie

La principale mission que l'on a eu à été découpée en plusieurs sous missions ce qui permettait de se concentrer sur une partie du site a chaque fois. Chaque mission partait d'une fonctionnalité à intégrer dans le site (un système d'authentification, la génération de fiche clients, la génération automatique de PDF...). Mais parfois les missions consistaient à juste se documenter, vérifier si quelque chose était possible, ou trouver des API / librairie. Ce type de mission ne donnait pas forcément lieu à une production mais était tout de même nécessaire pour faire avancer le projet dans la bonne direction.

Pour chaque mission, on commençait en groupe à discuter de comment bien l'intégrer et de quelles parties étaient nécessaires (quelles librairies, requête...). Une fois le sujet bien compris on pouvait se répartir les tâches entre les 3 stagiaires évidemment il y avait toujours moyen de s'aider les uns les autres.

Si il nous manquait quelques compétences ou que l'on devait apprendre de nouvelles choses pour implémenter la fonction, on avait plusieurs options. La première c'est de passer par des forums comme stackoverflow ou de regarder quelques vidéos sur un site comme youtube. Si les connaissances manquantes étaient trop importantes, la deuxième solution était de passer par des formations en ligne.

Après tout ça, nous pouvions commencer a intégrer la fonctionnalité, il y avait parfois des problèmes à propos de la concordance des noms de variable entre le php et le js (parfois on envoyer des noms de variable en js qui n'étaient pas ceux attendus en php) c'est pourquoi il était nécessaire de rester attentifs entre nous et continuer de communiquer régulièrement.

Tout au long de l'implémentation, on effectuait des commit réguliers sur le github avec des messages significatifs par rapport aux modifications effectués ce qui pouvait aussi servir à faire du versioning. La prise en main de github a été un peu compliquée mais une fois que l'on avait appris à s'en servir, cela a permis de grandement simplifier le développement.

B-Organisation des journées

Pour expliquer l'organisation du travail, il me semble important de détailler le déroulement d'une journée type.

Nous avions rendez-vous tous les matins sauf le vendredi à 9h15 devant les locaux de l'entreprise. Un fois rentrés il y avait un débrief avec le maître de stage, sur les sujets qui allaient être traités aujourd'hui, tout était fait à l'oral, mais je gardais une trace des éléments importants soit sur ordinateur soit sur un tableau blanc. J'utilise également ce temps là pour m'assurer que tout le monde savait quoi faire.

Ensuite, le travail de la journée pouvait commencer, les matinées étaient généralement peu chargées puisque la pause du midi était à 12h00 ce que l'on avait généralement un peu plus de 2 heures de travail sur la matinée. Pendant la pause du midi, on parlait de ce qui avait été fait dans la matinée, pour voir les difficultés de tout le monde et pour s'assurer que personne n'était bloqué.

Les horaires de l'après-midi étant de 13 à 17 heures, il était nécessaire de penser à prendre des pauses pour ne pas perdre son rythme de travail. Il nous arrivait d'être bloqué sur un problème et dans ces moments là, il est toujours mieux de s'arrêter quelques minutes avant de reprendre. La journée se terminait donc à 17 heures par un débrief de ce qui avait été fait dans la journée. Les objectifs étant encore une fois de s'assurer de la bonne avancée du projet. Il m'est cependant arrivé de rester un peu plus longtemps pour exposer un élément au maître de stage.

Le vendredi, nous étions en télétravail à cause des obligations du maître de stage. L'emploi du temps restait cependant similaire puisque les débrief se faisaient sur WhatsApp aux mêmes moments que d'habitude. Une fois terminé, la communication avec les deux autres stagiaires se faisait sur discord. À noter tout de même que la communication à distance est toujours plus complexe et que généralement, ces journées étaient moins productives.

En plus de la communication difficile, ces sessions de travail m'ont confirmé ce que j'avais déjà remarqué pendant le confinement, c'est à dire qu'avoir un environnement dédié au travail permet une meilleure concentration.

III. Réalisation des missions

A- La page de connexion et de création de compte

La première mission a été la création des pages d'authentification et de création de compte. J'étais chargé de faire le front c'est-à-dire html, css et le java script. La principale fonctionnalité était de vérifier si les champs étaient remplis correctement avant d'envoyer le formulaire vers le php pour créer le compte (ou vérifier l'authentification).

Les vérifications à effectuer étaient les suivantes : s'assurer que le numéro de téléphone ait un bon format, que l'on trouve bien un "@" et un "." dans l'adresse mail, qu'il n'y ait pas de chiffres dans le nom ou le prénom et que le mot de passe est une sécurité suffisante (8 lettres, 1 minuscule, 1 majuscule, un chiffre et un caractère spécial). De plus, j'ai ajouté un champ "vérification mot de passe" et "vérification adresse mail" pour s'assurer qu'il n'y ait pas d'erreur comme le font la plupart des sites.

Ce qu'il se passe concrètement c'est que quand on clique sur le bouton envoyer, le script va "bloquer" le comportement par défaut du bouton (qui serait d'envoyer le formulaire). Ensuite on commence une batterie de tests, si il y a le moindre problème, le champ ayant été mal rempli sera encadré en rouge avec un



Adresse mail :

augustin.marieepsi.fr

Une adresse mail doit contetnir un '@'.

Vérification adresse mail :

augustin.marie@epsi.fr

Les deux adresses mail doivent être identiques

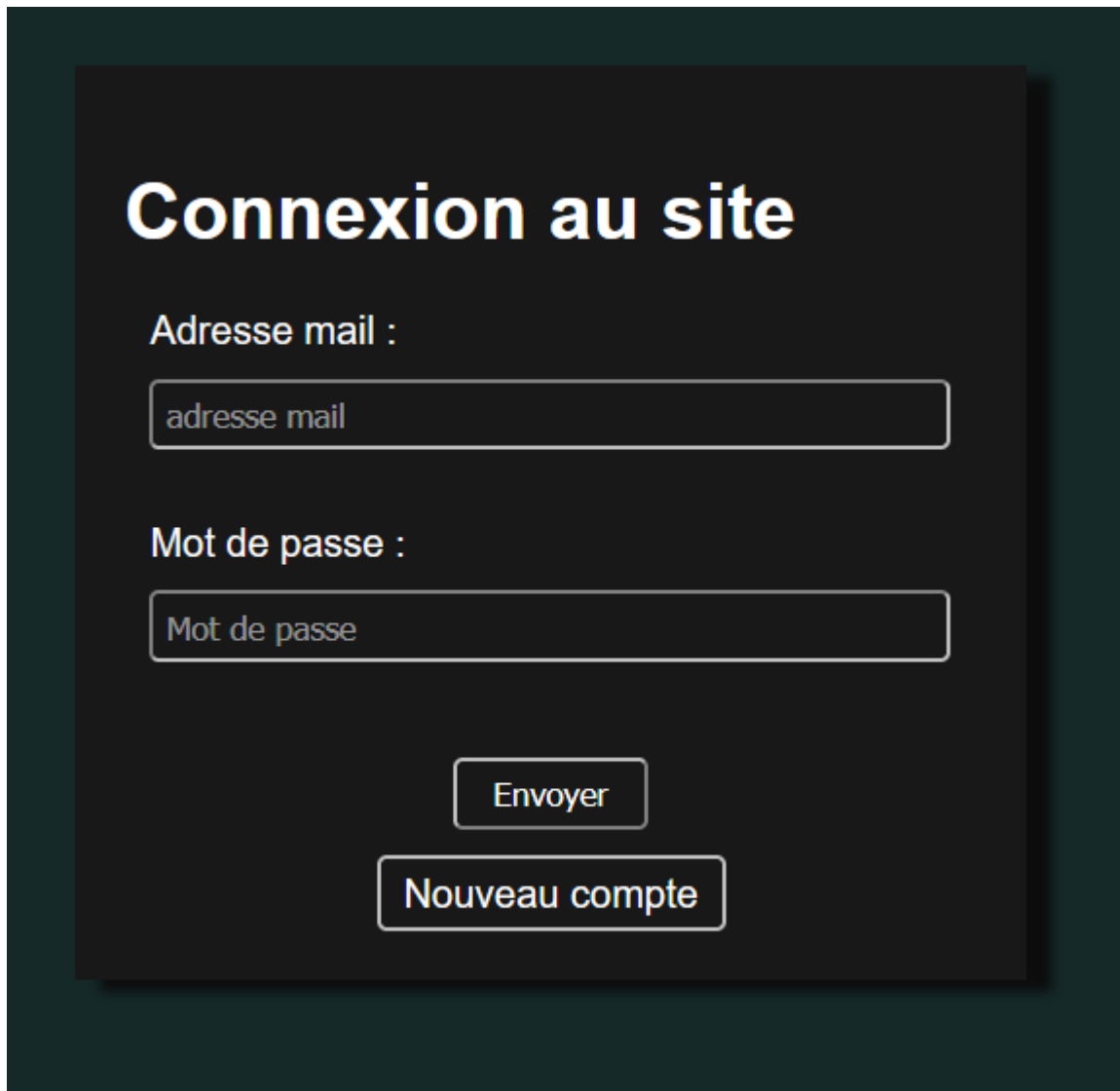
texte expliquant l'erreur.

Si tous les champs sont correctement remplis, alors le formulaire s'envoie vers le back du projet (les pages php). Du côté du php, on va vérifier si l'adresse mail est déjà utilisée (par un admin, un candidat ou une entreprise) et si ce n'est pas le cas, alors on va finalement envoyer le formulaire dans la base de données.

B-Conception de la charte graphique de l'application

Après avoir construit les premières pages html, il s'est posé la question de la charte graphique. Évidemment il ne s'agit que d'un prototype d'application mais le confort visuel de l'utilisateur reste très important.

J'ai donc regardé un peu ce qu'il se faisait sur internet et ce qui me semblait le plus adapté, j'ai décidé de partir sur un thème sombre pour le confort visuel et j'ai aussi arrondi les boutons et input et boutons pour donner un aspect moins brute a ces éléments la.



Connexion au site

Adresse mail :

Mot de passe :

Envoyer

Nouveau compte

Enfin, tout les boutons du site changent légèrement de couleur quand on passe la souris dessus car en règle générale, avoir un site qui réagit a l'utilisateur est toujours plus agréable.

Je me suis aussi imposé la règle d'utiliser les mêmes codes couleurs quand je concevais le css des autres pages pour garder une intégrité dans le site d'une page à l'autre.

C-Travail sur les API

Le problème qui m'a amené à étudier les API est le suivant, on veut que les entreprises entrent le siret et le nom de leur entreprise cependant, on veut s'assurer que les deux correspondent bien et que l'entreprise auquel ils correspondent soit bien en activité (puisque l'inscription d'une entreprise fermée sur ce genre d'outils est inutile). Comment pouvons nous faire ces vérifications alors que nous n'avons pas ces données?

La réponse à ce problème a été une API mise à disposition par insee.fr. Une API (ou application programming interface) est, comme son nom l'indique, une interface mise à notre disposition par un organisme, une entreprise ou un site. Elle permet d'aller utiliser des services d'un tiers. Dans ce cas, celle-ci a permis de faire ces vérifications en appelant l'api siren V3 développée par l'insee pour accéder à une base de données contenant ces informations.

Pour l'utiliser il faut une clé d'autorisation fournie par l'API (en se créant un compte) et depuis le javascript, il faut utiliser ce qu'on appelle une requête AJAX (pour Asynchronous JavaScript and XML), c'est une fonction particulière qui prend plusieurs paramètres pour pouvoir aller chercher des informations en passant par

```
$.ajax({  
  url: "https://api.insee.fr/entreprises/sirene/V3/siret/" + siret,  
  beforeSend: function(xhr) {  
    xhr.setRequestHeader("Authorization", "Bearer " + cle)  
  },  
  success: (data) => {  
    //Instructions  
  }  
});
```

le protocole http.

Url contient l'url de l'api à appeler, setRequestHeader sert à paramétrer le header de la requête http et success(data) est la fonction qui est appelée (si la requête aboutit), data étant la réponse de l'API (généralement en Json). Mais il existe beaucoup d'autres paramètres que l'on peut définir en fonction des prérequis de l'API.

Quelques difficultés ont été rencontrées avec l'intégration de l'API dans la vérification des champs à cause du concept de "script asynchrone". Par défaut, les requête AJAX s'exécutent de cette manière ce qui veut dire que quand le script de vérification des champs appelle l'API, au lieu d'attendre sa réponse

(comme une fonction classique avec un return) la requête AJAX s'exécute en parallèle du script principal ce qui fait que quand le script principal a besoin des données de l'API, celle ci n'a potentiellement pas encore envoyé sa réponse ce qui peut provoquer des bugs. La solution à ce problème a été de passer par les promesses qui permettent de forcer à attendre le résultat avant d'exécuter la suite du code.

D'autres API ont été étudiées au vu d'imaginer des solutions à certains problèmes de l'entreprise bien que ces recherches n'ont pas nécessairement donné lieu à une production notamment l'API Google et LinkedIn.

Pour l'API google, le but était de permettre de poser automatiquement des rendez vous sur le calendrier google des personnes inscrites ayant donné leur accord au préalable mais aussi de trouver des créneaux de disponibilité (l'API permettant de lire les événements déjà présent). Elle a été beaucoup plus complexe à utiliser puisque stricte d'un point de vue sécurité.

Évidemment, pour pouvoir créer un événement sur le calendrier de quelqu'un en son nom, on ne peut pas enregistrer son nom d'utilisateur et son mot de passe, cela serait dévastateur d'un point de vue sécurité. Il faut donc passer par un protocole appelé OAuth. Il fonctionne de la manière suivante, on demande à un utilisateur de se connecter, en échange, OAuth fournit une clé qui permet d'accéder aux autorisations acceptées par l'utilisateur.

Tout ceci permet deux choses tout d'abord, comme personne n'enregistre le mot de passe ou le nom d'utilisateur, ces informations restent sécurisées, deuxièmement, la clé ne permet d'accéder qu'aux fonctionnalités que l'utilisateur a accepté (on ne peut donc pas faire ce que l'on veut sur son compte).

Le problème de tout cela, c'est que l'utilisation de OAuth est très complexe et que les tutoriels officiels de google ne sont pas toujours clair il m'a fallu beaucoup de temps pour faire fonctionner l'API (inscrire un événement dans mon calendrier depuis une page html ou bien lire ceux que j'avais déjà inscrit manuellement).

De plus, pour pouvoir lire ou inscrire des événements à partir d'une seule connexion de l'utilisateur (sans qu'il ai besoin de donner son accord à chaque fois qu'on veut le faire), il faut stocker cette clé sur un serveur. Il faut aussi déclarer son application comme étant publique et pas comme en développement depuis l'interface de sécurité google. Cette étude de l'API google c'est donc arrêter la en sachant que cette solution existait et était applicable.

Enfin la dernière API ayant été étudiée plus rapidement était l'API LinkedIn. L'entreprise voudrait utiliser cette API pour récupérer des informations sur la personne quand celle ci donne son LinkedIn (au lieu de devoir les recopier manuellement)

Cette fois si je n'ai pas réellement utiliser l'API, il m'a simplement été demander de regarder si cette utilisation était possible. Après quelques renseignements, j'en ai conclu qu'il existait deux solutions au problème de départ. La première était d'effectivement utiliser l'API LinkedIn avec des autorisations particulières, celle-ci acceptait de rechercher des informations sur un compte en particulier. Cependant LinkedIn n'aime pas que l'on utilise l'API pour rechercher des personnes, il accepte seulement l'utilisation de l'API pour rechercher quelqu'un en particulier ayant donné son compte au préalable.

Une deuxième solution intéressante qui a attiré mon attention est une technique reposant sur le scraping. Le principe du scraping c'est l'automatisation de recherches répétitives, on peut concevoir ce type d'outils pour naviguer à notre place à travers plusieurs pages en prenant un type d'information en particulier. Il serait donc envisageable de laisser un outil de scraping aller sur le profil LinkedIn donner par l'interlocuteur, le laisser se connecter a notre place et prendre les informations désirées.

Une autre utilisation du scraping serait d'accélérer la recherche de CV. Éplucher les CV en ligne pour chercher les candidats maîtrisant un sujet en particulier peut prendre beaucoup de temps aux différents agents All You Need, configurer un outil de scraping permettrait automatiquement chercher un grand nombre de CV pour trouver ceux qui contiennent l'élément que l'on recherche. J'ai moi même testé une méthode de scraping utilisant sélénium et java pour rechercher automatiquement le nom des articles sur un site avec leur nombre de vues pour pouvoir avoir un avis sur cette technologie.

D-Séparation claire du back et du front

A mesure que le php et le java script se complexifient, je me suis rendue compte qu'avoir du html et du php pouvait vite devenir illisible, de plus la moindre modification du html nécessitait de chercher à travers de lignes php ayant été écrites par quelqu'un d'autres. Pire encore, si le javascript avait besoin d'effectuer une requête php ou de réagir au résultat de celle-ci, on était obligé de l'avoir dans le même document. On pouvait donc avoir un fichier avec du php, du html et du javascript avec du front et du back mélanger. Ce n'était pas lisible, mal organisé et peut efficace mais au début du stage je ne savais pas comment régler ce problème.

Celui-ci a été réglé lors de mon travail sur les requêtes AJAX, en effet, on peut appeler des fichiers php via requête AJAX j'ai donc proposé de complètement séparer le html, le js et le php. Voici les avantages que j'y ai vu.

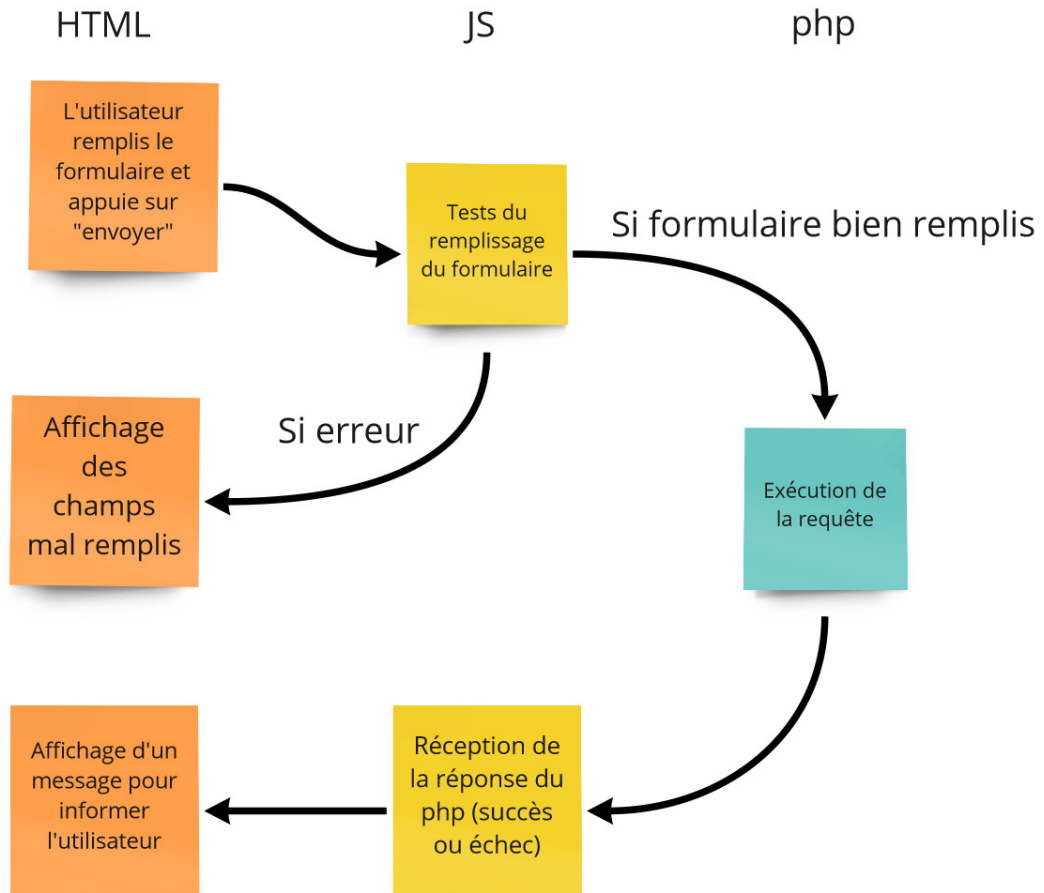
Premièrement, un meilleur découpage du code veut dire une meilleure lisibilité, le but premier de ce découpage est de ne plus avoir de code html dans le php, ni de balise `<script>`.

Le back et le front de l'application sont donc complètement séparés. Ensuite comme le php ne dépend plus du html celui-ci peut être développé indépendamment.

De plus, si le php échoue dans l'exécution d'une requête, alors il affichera son message d'erreur sur la page sans aucun style particulier, ce qui n'est pas bon d'un point de vue utilisateur, passer par l'intermédiaire du js permet d'afficher les erreurs de php plus proprement.

Troisièmement, le fait que le php soit détaché permet de le réutiliser dans plusieurs pages html sans avoir à le recopier. Et finalement, envoyer un formulaire à du php sur la même page que le html nécessite de recharger la page alors que faisant appel au php sur le java script via requête AJAX nous n'avons pas à le faire.

Alors comment ça marche ? On appelle le php via une requête AJAX comme vu plus haut, en tant que lien on met le chemin vers le fichier php à appeler, on renseigne les potentielles données nécessaires puis on traite les données renvoyées par le php. Ensuite en fonction de celles-ci, on affiche un message sur la page html comme "Votre compte a bien été créé", ou "Une erreur est survenue". Voici le schéma de fonctionnement pour la page de création de compte par exemple.



miro

Tout ceci a forcé à coder, et à isoler des parties de codes mais il à été convenu avec les autres stagiaires que cette solution serait beaucoup plus lisible et plus facile à entretenir sur le long terme.

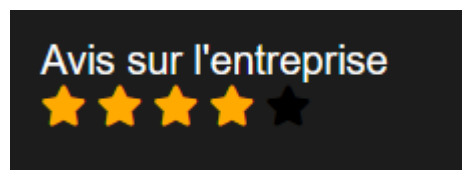
E-Les fiches de renseignement

Ceci est la fonctionnalité principale du site l'idée est de pouvoir renseigner des champs organisés par catégorie pour les candidats on veut pouvoir enregistrer un maximum d'informations sur eux et pour les entreprise on veut récupérer quelques informations mais aussi une liste de contacts au seins de celle ci, une liste des missions ouvertes par l'entreprise et les contrats signés avec elle pour avoir une estimation du chiffre d'affaire.

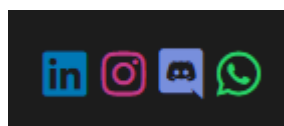
Pour pouvoir disposer les différents champs efficacement, je suis passer par un système de grid css. J'ai dû apprendre à utiliser cet aspect du css mais il a donné un fichier chargé, passer par du scss ou sacc par la suite permettra d'avoir un fichier mieux organiser.

Ensuite, j'ai eu besoin de trouver une façon de mettre des icônes dans le document. On voulait utiliser des icônes d'étoiles pour faire un système d'avis visuel et rapide à comprendre mais aussi mettre les logo de réseaux sociaux pour savoir (encore une fois très rapidement) a quoi correspondait les différents identifiants rajouter (savoir très rapidement si il s'agissait d'un identifiant instagram, discord ou encore LinkedIn).

En effectuant quelques recherches, je suis tombé sur fontawesome qui est une librairie js qui permet de faire exactement cela, on peut même modifier le style des icônes avec des propriétés css classiques. On peut même par la suite y ajouter des événements js ce qui a permis de faire un système de notation dans lequel on peut directement cliquer sur les étoiles pour changer la valeur (cela a aussi permis de faire le lien pour télécharger la fiche complète sous forme d'icône.



Pour ce qui est de la couleur des icônes, on peut très facilement la trouver sur les sites officiels des différentes marques. Ce qui rend sur le site comme ceci :



D'un point de vue fonctionnel des fiches, il a fallu créer un système pour permettre d'enregistrer régulièrement une fiche que l'on est en train de créer

(donc faire en sorte de savoir si on doit envoyer une requête d'ajout ou de modification au php). Pour se faire je suis passer par une variable qui s'initialise au chargement d'une page existante (si il trouve un id dans l'url) si c'est le cas, le script comprend que l'on doit modifier cet élément. Si on crée une nouvelle fiche, alors cette même variable est initialisée qu'après avoir effectué l'ajout de l'entreprise (c'est a dire quand on fait une requête d'ajout qui aboutit).

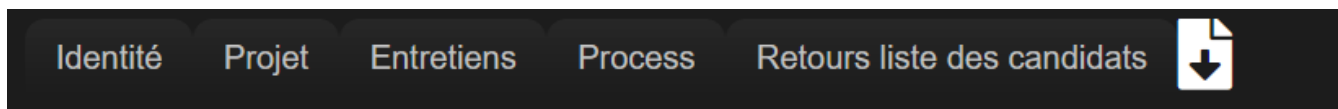
Il y a eu un problème similaire avec les tables liées aux entreprises (la table contact par exemple). La question était de savoir si l'utilisateur venait de créer un contact ou s' il en avait modifié un existant. Pour se faire nous chargeons l'id du candidat si il existe comme une classe sur un élément ce qui permet par la suite de savoir si le contact possède un id (et existe donc déjà dans la base de donnée) ou pas.

Nous avons ensuite créé une liste de clients et de candidat qui recense les informations les plus importantes et qui permet d'accéder à leur fiche (ou bien de rajouter un nouveaux client ou candidat).

F-La création automatique de PDF

C'était un des problèmes de base de l'entreprise, comment éviter de perdre du temps à faire des PDF avec des informations déjà entrées dans le logiciel. Principalement deux options ont été trouvées, la première est la fonction `print()` de javascript qui permet d'enregistrer la page ou de l'imprimer mais qui ne permet pas de prendre les couleurs d'arrière-plan. La deuxième est une librairie appelée `html2pdf.js` qui donne en plus de préciser plusieurs options comme par exemple la qualité du pdf, le format (a4) ou bien encore le nom du pdf.

Le fonctionnement des deux méthodes est similaire, il suffit de charger une page et de lancer la fonction. On ne peut télécharger la fiche qu'une d'une fiche que l'on a chargée ou que l'on vient d'enregistrer dans la base (on ne peut pas créer la fiche d'un candidat pas encore présent dans la base de donnée). Un logo apparaît alors qui, une fois cliqué, ouvre un nouvel onglet avec la page html qui sera convertie en pdf. Une fois le chargement terminé, il demande si on veut enregistrer le pdf ou non.



Dans l'enregistrement de la fiche la principale difficulté a été l'utilisation de la librairie, celle-ci avait en effet quelques difficultés avec les images et les icônes. Pour les images il a suffi de modifier quelques options pour choisir le bon type d'image et pour les icones il a fallu utiliser une copie locale de `font awesome` au lieu de passer par la version en ligne parce que la librairie avait un problème de chargement avec les ressources qui n'étaient pas en local.

IV. Évaluation des compétences mobilités

A-Adéquation du travail

A la fin de la période de stage, nous avons réussi à faire un prototype avec les fonctionnalités de base les plus importantes du point de vue de l'entreprise. Pour résumer, on avait la possibilité de créer des comptes clients et candidats soit par le portail de création de compte soit via la création de fiche. On avait aussi moyen de créer des pdf à partir de ces fiches automatiquement pour pouvoir transmettre facilement les informations entre les différents membres de l'entreprise. Le tout dans un code bien organisé. Nous avons aussi produit beaucoup de documentations sur différentes API pour pouvoir les intégrer rapidement par la suite (nous avons créé beaucoup de fichiers fonctionnels qui n'ont plus qu'à être intégrés).

Le tuteur de stage étant satisfait de l'avancée du projet, tout ceci permettra à l'entreprise de tester le poc et de l'améliorer rapidement en exploitant la documentation créée. Il pourra ensuite décider d'améliorer le poc en ajoutant différentes fonctionnalités, ou bien encore le faire développer par des équipes professionnelles. Ce qui est sûr c'est que cette application web lui permettra de mieux se rendre compte si son projet est réaliste ou non.

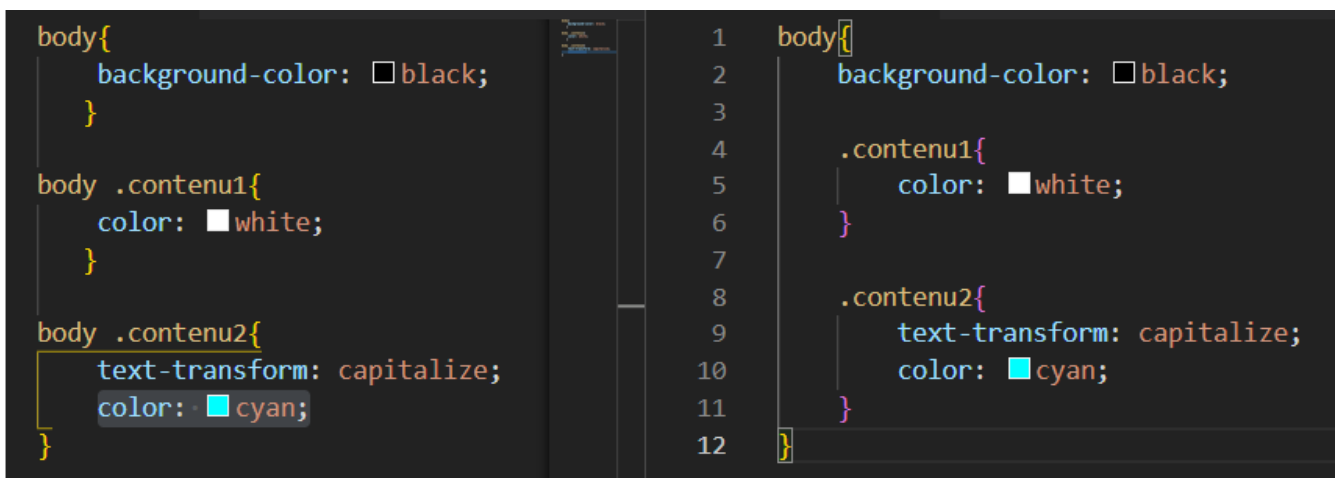
En terme d'amélioration, je peux voir plusieurs choses. D'une part, nous n'avons pas eu le temps d'intégrer toutes les API sur lesquelles je m'étais renseigné, la première amélioration logique serait donc de les intégrer.

Deuxièmement, pendant toute la durée du stage, il n'y avait pas de serveur centralisé, il faut bien sûr en rajouter un ce qui débloquerait beaucoup de possibilités. Notamment utiliser l'API google (qui nécessite un serveur pour l'utilisation que l'on veut en faire) pour consulter les possibilités de rendez-vous. Du point de vue des candidats, cela permettrait de proposer des créneaux cohérents, évidemment il ne s'agirait pas de manuellement regarder les rendez-vous ce qui serait très intrusif pour la vie privée de la personne... On aurait simplement un programme qui regarderait les horaires de rendez-vous présents pour renvoyer une liste de créneaux cohérents (donc aucune information personnelle ne serait enregistrée. Il ne s'agit pas non plus de prendre un rendez-vous sans rien demander au candidat, on parle vraiment ici de proposer des créneaux et pas d'en réserver sans leur accord. Une autre utilisation imaginable serait de s'en servir juste pour poser des créneaux une fois qu'il y a eu un accord de rendez-vous sans que le candidat ait à le faire lui-même (sans "lecture" du calendrier).

Une autre amélioration possible serait de passer par un framework comme react (car le front est en js pur). Pour ceux qui ne le savent pas, react est une librairie javascript de front développée par facebook qui a pour but de faciliter la création d'application web et de les rendre plus rapides. Elle est notamment

utilisée par Netflix, facebook (évidemment) ou d'autres réseaux sociaux comme instagram ou tiktok et peut être couplé à symphony pour le back. Cela permettrait de rendre le site plus fluide et agréable d'un point de vue utilisateur. De plus react peut être intégré sur seulement une partie du site, son intégration pourrait donc être progressive ou centrée sur les pages utilisés par les clients et candidats en priorité (les interfaces utilisateur)

Un dernier axe d'amélioration que je pourrais soulever est l'aspect graphique du site, des formulaires et des pdf. En effet même si j'ai passé beaucoup de temps à mettre au point un site agréable à utiliser, il m'est arrivé de devoir ajouter des éléments en cours de route ou de ne pas savoir quelle disposition était la plus adaptée pour les éléments il est donc important de mon point de vue, de faire une charte graphique détaillée pour créer une disposition plus en accord avec l'utilisation future de l'application. On pourrait également mettre en œuvre cette nouvelle charte en passant par un langage comme le scss ou sass qui permettent de faire du css mieux structuré et en ajoutant des fonctionnalités. Ces deux langages ont cependant besoin d'être transformés en css compréhensible par le navigateur.

The image shows a side-by-side comparison of CSS and SCSS code. On the left, standard CSS is shown with nested selectors: 'body' for the background, 'body .contenu1' for the first content block's color, and 'body .contenu2' for the second content block's text transform and color. On the right, the same styles are written in SCSS using the '@' symbol for selectors and curly braces for blocks. The SCSS version is more concise and uses color names like 'black', 'white', and 'cyan' with corresponding color swatches. Line numbers 1 through 12 are visible on the right side of the SCSS code block.

```
body{
  background-color: black;
}

body .contenu1{
  color: white;
}

body .contenu2{
  text-transform: capitalize;
  color: cyan;
}
```

```
1 body{
2   background-color: black;
3
4   .contenu1{
5     color: white;
6   }
7
8   .contenu2{
9     text-transform: capitalize;
10    color: cyan;
11  }
12 }
```

Comme on peut le voir dans cet exemple, il est beaucoup plus rapide de voir que l'on s'intéresse à des éléments dans l'élément body. Le sass et scss permettent également de faire des conditions et des boucles ce qui permet de ne pas avoir à répéter, coupler avec des variables, on peut obtenir un vrai programme qui permet de ne pas avoir à se répéter. Enfin, ils intègrent aussi des mixin qui sont comme des fonctions réutilisables partout dans le code.

B-Compétences mises en œuvre

Pendant ce stage, j'ai pu exploiter et développer certaines compétences mais je me suis aussi rendu compte qu'il y en avait d'autres qui me restait à travailler, je vais faire le point sur les plus importantes d'entre elles.

Vu le sujet de mon stage, j'ai pu exploiter et développer mes compétences techniques liées au développement front. Je ressors de ce stage avec une meilleure maîtrise du css, notamment des grilles que je n'avais jamais utilisé avant mais aussi quelques subtilités comme les variables css ou la fonction calc. J'ai aussi pu apprendre à utiliser le javascript avec des appels d'API et l'utilisation de bibliothèques. Ces compétences font qu'aujourd'hui, je me sens prêt à approfondir le développement web ce qui me permettra de m'intéresser au front ou à des framework à l'avenir.

J'ai pu aussi m'améliorer dans un grand nombre de compétences liées au travail en équipe tel que la communication, ou l'organisation. En effet il m'est souvent arrivé de prendre les devants pour aider à distribuer le travail à faire entre les trois stagiaires, prendre l'initiative de faire des points pour savoir où tout le monde en était et, de moi même, aller voir si tout se passait bien pour tout le monde. Il est parfois arrivé que quelqu'un bloque sur quelque chose sans forcément demander de l'aide en nous pensant trop occuper. J'ai donc pu me rendre compte de l'importance de communiquer dans un groupe de travail comme celui-ci.

Une des dernières compétences qui a été très importante dans ce stage a été l'autonomie. En effet, le maître de stage n'étant pas souvent présent, et les sujets étant régulièrement hors de nos connaissances actuelles, il m'a fallu apprendre à faire mes propres recherches, trouver mes propres solutions et comprendre rapidement les sujets et les problèmes évoqués pour pouvoir être efficace dans le travail demandé.

Cependant, je me suis également rendu compte des compétences à développer par la suite. Même si ce stage m'a permis de développer ma coordination avec mes collaborateurs, j'ai encore du travail à faire sur ce point là j'ai notamment pu avoir des soucis avec des changements dont je n'avais pas été averti par exemple. Cela m'a permis de me rendre compte de l'importance de github, en particulier d'avoir des commits réguliers avec des messages permettant de vite comprendre ce qui a été modifié.

Conclusion

J'aimerais commencer par donner mon analyse des conditions de travail. Tout d'abord, bien que le maître de stage était souvent occupé, il s'assure de rester attentif sur l'évolution du projet. Certains auraient vu dans ces obligations un problème mais pour moi, cela a surtout permis de renforcer mon autonomie.

Quelque chose qui m'a surpris était l'absence de template pour certains éléments (wireframes pour le site, ou template pour les pdf) cela m'a forcé à me poser moi même la question de comment faire les choses de manière à rendre le site agréable à utiliser.

Enfin, le fait de ne pas être intégré dans une entreprise spécialisée dans l'informatique et de travailler sur un sujet qui constitue l'avenir de cette entreprise m'a donné le sentiment de faire quelque chose d'utile et d'important pour celle-ci. Ce qui a été beaucoup plus motivant qu'une simple refonte de site par exemple ou quelque chose de donné "juste pour occuper le stagiaire", pour me sentir que je faisais quelque chose d'important à été une vraie motivation.

La mission qui a donc été réalisée à apporter à l'entreprise une première version d'essais du logiciel, qui pourra être testée, modifiée ou complexifiée pour améliorer sa cohérence au sein du système de fonctionnement de l'entreprise. D'ailleurs il est déjà prévu que d'autres professionnels reprennent le travail produit par notre groupe de stagiaires.

L'apport que nous avons donné à l'entreprise lui permettra à elle et à notre maître de stage de gagner du temps et concevoir la suite avec une idée plus précise de ce qui est à modifier pour avoir le meilleur logiciel possible pour leur utilisation.

Pour le prochain stage, j'envisage de revenir dans la même entreprise, j'ai trouvé ce stage très instructif et intéressant. Le maître de stage ayant exprimé sa satisfaction et son envie de nous reprendre en stage l'année prochaine, j'espère revenir dans la même entreprise.

J'espère également continuer d'apprendre, ce stage aura été plaisant de part la quantité de nouvelles choses apprises, que mon souhait de retourner dans l'entreprise All You Need se concrétise ou pas, j'espère pouvoir continuer de travailler sur des sujets que je ne maîtrise pas intégralement, pourquoi pas d'autres technologies de développement front mais étant seulement en première année, je ne me ferme pas la porte d'autres domaines de développement.

Enfin j'espère pouvoir être intégré dans un groupe lors de mon prochain stage. Je trouve que le travail avec d'autres a été une expérience enrichissante, et pouvoir compter sur d'autres permet d'avancer beaucoup plus vite, parfois une simple relecture du code permet de mettre le doigt sur une erreur évidente que l'on ne remarque pas forcément quand on travaille dessus depuis plusieurs heures. D'une manière plus générale, je n'aimerais pas être envoyé sur quelque

chose a faire de mon côté sans communication ou sans beaucoup de lien avec le reste des projets de l'entreprise.