



Rapport de projet 3ème année ISIMA

Filière : Génie Logiciel et Systèmes Informatiques

---

# Développement d'une application web dédiée à la recherche de stages étudiants à Polytech

---

*Présenté par :*

**M. Florian GONZALES**

**M. Bruno JOUSSE**

*Responsable de projet :*

**Mme Marinette BOUET**

Date de la soutenance :

13 Mars 2020

Durée du projet :

6 mois

Campus des Cézeaux. 1 Rue de la Chebarde.  
TSA 60125. 63178 Aubière CEDEX



# Remerciements

Nous souhaitons tout d'abord remercier notre tutrice de projet, Mme BOUET Marinette pour son accompagnement au cours du projet, ainsi que pour ses conseils sur la réalisation de ce rapport et la préparation de la soutenance.

Nous remercions également l'ensemble du personnel de l'école Polytech pour leur aide concernant certain aspect du développement de la solution, notamment la connexion LDAP.

Enfin, nous remercions les quelques étudiants ISIMA qui nous ont apporté leur soutien lorsque nous bloquions dans la réalisation du projet.

## Résumé

Durant ce projet de 3ème année, nous avons eu l'occasion de développer une application ayant pour but d'aider les étudiants de l'école Polytech Clermont lors de leur recherche de stage. Grâce à celle-ci, ils sont en mesure de consulter tous les stages précédemment effectués par d'anciens élèves de leur département et année. Ils peuvent affiner leur recherche en fonctions du lieu de travail, des thèmes abordés durant le stage, ... leur permettant de trouver des entreprises et stages correspondants à leurs attentes.

Nous avons dû développer cette application web en tant que **Full Stack Developers**. Elle a été codée en **HTML**, **CSS**, **SCSS**, **TWIG** et **Javascript** pour la partie **FrontEnd** (désigne les éléments d'un site que l'on voit à l'écran et avec lesquels on peut interagir depuis un navigateur). Le développement de la base de données a été réalisé avec l'aide du logiciel **Doctrine** en **Microsoft SQL Server**. Pour ce qui est du **BackEnd** (partie administrant l'application), nous avons utilisé le langage **PHP**. Nous nous sommes également servi du framework **Symfony** pour gérer l'organisation de notre code, ainsi que des outils de **Git** afin d'assurer la gestion des versions et le stockage du projet.

Ce projet de fin d'étude nous a permis de renforcer nos compétences dans les langages de programmation et de travailler sur toutes les phases de réalisation d'une application web .

## Abstract

During this 3rd year project, we had the opportunity to develop an application designed to help Polytech Clermont's students in their search for an internship. Thanks to it, they are able to consult all the internships previously carried out by former students of their department and year. They can search for internships according to the place of work, the themes covered during the internship, ... helping them to find companies and internships fitting their desires.

We had to develop this web application as **Full Stack Developers**. It was coded in **HTML**, **CSS**, **SCSS**, **TWIG** and **Javascript** for the **FrontEnd** part (designates the elements of a site that you see on screen and with which you can interact from a browser). The development of the database was done with the help of **Doctrine** software in **Microsoft SQL Server**. As for the **BackEnd** (the part administering the application), we used the **PHP** language. We also used the **Symfony** framework to manage the organization of our code, as well as **[Git]** tools to ensure version management and project storage.

This end-of-study project allowed us to reinforce our skills in programming languages and to work on all the phases of the realization of a web application

**Keywords** : Web application, Symfony, PHP, HTML, CSS, JS, MySQL, Git, Agile, LDAP

# Table des matières

<b>Introduction</b>	<b>11</b>
<b>1 Présentation de l'école</b>	<b>13</b>
<b>2 Présentation du sujet de projet</b>	<b>15</b>
2.1 L'environnement . . . . .	15
2.2 Choix des technologies . . . . .	15
2.3 Notre sujet de projet . . . . .	16
2.3.1 Les étudiants . . . . .	17
2.3.2 Le responsable de département . . . . .	17
2.3.3 L'administrateur . . . . .	18
2.3.4 L'observateur . . . . .	18
<b>3 Développement de la solution</b>	<b>19</b>
3.1 La gestion du projet . . . . .	19
3.2 Ordonnancement du projet . . . . .	20
3.3 Conception de l'application . . . . .	22
3.3.1 L'analyse logicielle . . . . .	22
3.3.2 Conception de la base de données . . . . .	25
3.3.3 Création des vues de l'application . . . . .	26
3.4 Développement du code . . . . .	27
3.4.1 Développement du front-end et back-end . . . . .	27
3.4.2 Phase de test . . . . .	27
3.4.3 Maintien du code . . . . .	28

<b>4</b>	<b>Présentation de la solution réalisée</b>	<b>29</b>
4.1	Recherche des stages . . . . .	29
4.1.1	Filtrer les stages . . . . .	30
4.1.2	Liste des stages . . . . .	31
4.1.3	Afficher les stages sur la carte . . . . .	33
4.2	Importer des stages . . . . .	34
4.2.1	Enregistrer des stages à partir de fichiers tableurs . . . . .	34
4.2.2	Déterminer les coordonnées d'une adresse . . . . .	34
4.2.3	Ajouter les champs manquants . . . . .	35
4.3	Administrer les données . . . . .	35
4.4	Navigation . . . . .	36
4.4.1	Icônes de droite . . . . .	36
4.4.2	Icônes de gauche . . . . .	36
4.5	Connexion . . . . .	37
4.6	Déploiement de l'application . . . . .	38
	<b>Conclusion et perspectives</b>	<b>39</b>

# Table des figures

3.1	Schéma du principe de méthode Agile . . . . .	19
3.2	Diagramme de Gantt . . . . .	21
3.3	Diagramme de cas d'utilisation . . . . .	23
3.4	Diagramme de classe du projet . . . . .	24
3.5	Schéma de la base de données du site . . . . .	25
3.6	Vue présentée pour l'interface de recherche des stages . . . . .	26
4.1	Affichage recherche de stages . . . . .	30
4.2	Affichage du formulaire de filtrage . . . . .	30
4.3	Affichage liste des stages . . . . .	31
4.4	Détails d'un stage . . . . .	32
4.5	Affichage de la carte . . . . .	33
4.6	Affichage importation de fichier . . . . .	34
4.7	Affichage administration des données . . . . .	36
4.8	Barre de navigation . . . . .	36





# Liste des sigles et acronymes

<b>IDE</b>	<i>Integrated Development Environment</i>
<b>PHP</b>	<i>Hypertext Preprocessor</i>
<b>CSS</b>	<i>Cascading Style Sheets</i>
<b>SCSS</b>	<i>Langage dynamique de génération de CSS</i>
<b>TWIG</b>	<i>Moteur de template pour PHP</i>
<b>SQL</b>	<i>Structured Query Language</i>
<b>MySQL</b>	<i>Système de gestion de base de données SQL</i>
<b>MVC</b>	<i>Model View Controller</i>
<b>CSV</b>	<i>Comma Separated Values</i>
<b>XLS</b>	<i>Extension de fichier Microsoft Excel</i>
<b>PDF</b>	<i>Portable Document Format</i>
<b>PNG</b>	<i>Portable Network Graphics</i>
<b>XAMPP</b>	<i>(X(cross) Apache MariaDB Perl PHP)</i>
<b>API</b>	<i>Application Programming Interface</i>
<b>GPS</b>	<i>Global Positioning System</i>
<b>JSON</b>	<i>JavaScript Object Notation</i>
<b>HTTP</b>	<i>HyperText Transfer Protocol</i>
<b>LDAP</b>	<i>Lightweight Directory Access Protocol</i>
<b>VM</b>	<i>Virtual Machine</i>

Les acronymes utilisés dans ce rapport sont indiqués par des \*.



# Introduction

Pour ce projet de dernière année d'ISIMA, nous avons travaillé avec l'école Polytech située à Clermont-Ferrand. Ce projet s'insère dans le cadre d'une soumission de projet par Polytech Clermont au réseau Polytech. Ayant été retenu, Polytech Clermont souhaite maintenant proposer un prototype qui sera présenté en Juin lors d'un compte-rendu de tous les projets acceptés. Il pourra alors peut être repris pour la gestion des stages de tout le réseau Polytech. Ainsi, ce prototype doit être une application aboutie utilisable dans un premier temps par les étudiants de Polytech Clermont, mais aussi facilement modifiable afin de permettre son utilisation sur tout le réseau Polytech.

Le projet en question est une application web ayant pour but d'aider les étudiants de Polytech Clermont durant leur recherche de stage. En effet, ils doivent être en mesure, grâce au site, de trouver tous les stages réalisés les années précédentes par des étudiants de l'école qui étaient dans le même département et la même année qu'eux. Ils peuvent filtrer les résultats en renseignant certains champs comme la localisation ou le nom de l'entreprise. Cette application sera déployée sur les serveurs internes à Polytech Clermont et accessible depuis leur réseau interne. Les élèves ne seront pas les seuls à utiliser cette application. En effet, les responsables de stage pourront importer les stages réalisés chaque années, et des observateurs pourront avoir accès à tous les stages dans le but d'émettre des statistiques par exemple.

Une fois la recherche terminée, les étudiants auront accès à des détails précis sur le stage et pourront contacter leur responsable de stage pour entrer en contact avec l'entreprise si cette dernière propose toujours des stages similaires.

Ce projet est réalisé en PHP\*, CSS\* et JavaScript en utilisant le framework Symfony (version 5.2) et grâce à l'IDE\* PHPStorm.

Nous retrouverons dans ce rapport une présentation de l'école ayant soumis cette demande de projet, suivi d'une présentation de l'environnement de travail et les objectifs de notre travail. Ensuite, nous verrons comment s'est organisé le développement de cette application. Enfin, nous examinerons les résultats de nos travaux.



# Chapitre 1

## Présentation de l'école

L'école polytechnique universitaire de l'Université Clermont-Auvergne, appelée également Polytech Clermont-Ferrand ou encore Polytech-Clermont, est l'une des 205 écoles d'ingénieurs françaises accréditées au 1er septembre 2018 à délivrer un diplôme d'ingénieur.

Elle se situe sur le campus des Cézeaux, sur les communes de Clermont-Ferrand et d'Aubière, dans le département du Puy-de-Dôme en région Auvergne-Rhône-Alpes. L'école est membre de la conférence des grandes écoles (CGE) et du réseau Polytech.

Polytech Clermont-Ferrand est une école polytechnique universitaire, d'un effectif total de 600 élèves-ingénieurs répartis sur trois années. Elle délivre ainsi plus de 200 diplômes par an. En 2019, on dénombre 6000 ingénieurs déjà formés par Polytech-Clermont.

Le Diplôme d'Ingénieur de Polytech-Clermont est délivré dans une des six spécialités suivantes :

- Génie biologique (GB)
- Génie civil (GC)
- Génie électrique (GE)
- Génie mathématique et modélisation (GMM)
- Génie physique (GP)
- Génie des systèmes de production (uniquement par la voie de l'apprentissage à Montluçon)



# Chapitre 2

## Présentation du sujet de projet

### 2.1 L’environnement

Pour ce projet, nous avons utilisé nos ordinateurs personnels et sur la fin du projet un portable fourni par l’école pour travaillé sur la connexion au réseau interne Polytech-Clermont. Pour le choix des technologies, nous étions uniquement restreints à du PHP\* ([PHP ]) et MySQL, et nous pouvions choisir toutes les autres technologies, sous réserve de l’accord de notre responsable de projet. Ceci nous a permis par exemple de développer l’application avec la dernière version de Symfony, actuellement en 5.2.

Nous avons utilisé le logiciel PHPStorm afin de développer notre site, ainsi que le framework Symfony en version 5.2 (version la plus récente du framework). Pour pouvoir développer l’application, nous avons mis en place un serveur local sur notre machine à l’aide de XAMPP\*, nous permettant d’avoir un serveur Apache et une base de données MySQL.

### 2.2 Choix des technologies

Pour développer l’application, nous avons utilisé le framework css **Bootstrap 4** ([Boo ]) pour plus facilement gérer la construction des pages. Nous avons également opté pour le framework Symfony car il apporte des bundles très utiles et permet d’avoir une organisation claire du code source suivant l’architecture Modèle Vue Contrôleur (MVC).

L’architecture MVC sépare le code en plusieurs parties, permettant une conception et une organisation **claire** et **efficace**. L’application développée est ainsi plus facilement maintenable par d’autres développeurs si notre travail est amené à évoluer dans le futur. Le

framework Symfony permet de suivre cette architecture ainsi que d'utiliser des **bundles** disponibles sur internet. Ceci permet d'éviter de recoder des fonctionnalités qui ont été réalisées un nombre incalculable de fois et donc de gagner du temps sur le développement. Parmi ces Bundles, les bundles les plus importants sont :

- Doctrine ([Doc ]) facilitant grandement la gestion de la base de données, et apportant un lien entre entité objet et entité base de données,
- Twig ([Twi ]) facilitant le développement du front-end, car il s'agit d'un langage de template apportant une sur-couche au HTML\* en y ajoutant de la logique et des variables,
- Forms facilitant la génération et la gestion des formulaires, en permettant de les manipuler de manière orientée objet.

Au début de ce sujet, nous n'avions pas le même niveau de connaissances sur Symfony. En effet, Florian avait déjà travaillé sur cette technologie pendant 5 mois au cours de son stage de 2ème année. Bruno quant à lui n'avait jamais utilisé Symfony et a donc dû se former en début de projet. Afin de se former rapidement sur ces technologies, Bruno a beaucoup consulté la documentation officielle de Symfony : [Sym ]. Il a également suivi de nombreux cours du développeur **Grafikart** durant les vacances d'hiver pour approfondir sa formation : [Gra ] et être au niveau pour commencer le développement. Cependant nous avons tous les deux poursuivi notre apprentissage durant la réalisation de ce projet, avec l'aide d'internet en fonction des problèmes que nous rencontrions.

## 2.3 Notre sujet de projet

Le sujet de notre projet était de créer une nouvelle application web, destinée à être utilisée par les étudiants et certains employés de l'école Polytech-Clermont.

Afin d'aider les étudiants en recherche de stage, les responsables de stage de l'école fournissent des listes de stages effectués par d'anciens étudiants. Cependant, rechercher un stage à travers ces fichiers est chronophage et peu intuitif pour les étudiants. De plus, cela requiert un certain nombre d'interactions avec le responsable de stage.

Le principe général de cette application consiste à aider ces étudiants durant leur recherche de stage, en leur donnant la possibilité de rechercher tous les stages réalisés les années précédentes par d'anciens élèves et correspondant à leurs critères de recherche (localisation, sujet, entreprise, gratification, ...). Si un ancien sujet de stage les intéresse, ils seront en mesure de consulter une description précise du stage, ainsi que de contacter le responsable Polytech-Clermont de ce stage pour savoir si l'entreprise propose encore des stages similaires, ou bien contacter directement l'ancien tuteur entreprise de ce stage s'il a bien voulu que ses coordonnées soient transmises.



Pour créer cette nouvelle application, nous avons un cahier des charges regroupant précisément le sujet du projet ainsi que toutes les fonctionnalités qui doivent être présentes pour chaque type d'utilisateur dans la version finale. Mais comme dans tout projet, nous avons dû le compléter à mesure de la conception car nous avons rencontré des situations ou des questionnements qui n'étaient pas attendus dès le début.

### 2.3.1 Les étudiants

Le site doit permettre aux étudiants d'afficher les listes des stages précédemment réalisés. Ces résultats seront affichés en fonction de plusieurs critères :

- **l'étudiant** : les stages affichés doivent correspondre au département et à l'année de l'étudiant connecté sur l'application.
- **La recherche** : les résultats correspondront à la recherche de l'étudiant, à travers des données géographiques, des mots-clés, des technologies, ...

Les résultats de recherche seront affichés sur une carte du monde avec des points sur les lieux où des stages ont été effectués et sur une liste affichant les données synthétiques du stage.

Une fiche pour chaque stage doit être disponible en sélectionnant un point sur la carte ou une ligne de la liste. Cette fiche contient des informations détaillées sur le stage en question. Cependant, selon les stages, ils n'auront parfois pas accès aux informations concernant le tuteur de l'entreprise.

### 2.3.2 Le responsable de département

Ces utilisateurs ont les mêmes droits que les étudiants et disposent de quelques fonctionnalités supplémentaires sur le site. Ils peuvent rajouter des stages dans la base de données en important des fichiers tableurs (ex : Excel) listant les stages de leur département. En effet, Polytech-Clermont utilise un logiciel nommé Konosys qui leur permet de sauvegarder les informations sur les stages effectués dans des fichiers tableurs. Certaines informations comme les thèmes ne sont pas présents lors de la génération des fichiers tableurs, ils seront donc aussi en charge d'ajouter les champs manquants aux stages par le biais de l'application.

### **2.3.3 L'administrateur**

C'est l'utilisateur possédant le plus de privilèges. Il doit pouvoir rechercher les stages de tous les départements confondus et a un contrôle total de la base de données (stages, thèmes, entreprises, adresses) via l'interface du site.

### **2.3.4 L'observateur**

C'est un utilisateur ayant pour fonction de produire des statistiques sur les stages. Ainsi, il doit pouvoir rechercher comme l'administrateur les stages de tous les départements.

# Chapitre 3

## Développement de la solution

### 3.1 La gestion du projet

Lors de la réalisation du projet, nous avons suivi une méthode Agile [figure 3.1]. Ce modèle de pilotage de projet est une succession d'itérations :

- **Backlog du sprint** : sélection des tâches qui devront être effectuées durant le sprint
- **Sprint** : réalisation des fonctionnalités, design, tests et déploiement. L'application doit être fonctionnelle à la fin de chaque sprint.
- **Revue de sprint** : Observations de ce qui a été fait durant le sprint, les fonctionnalités implémentées, manquantes et les difficultés rencontrées.

La durée de chaque sprint était d'une à deux semaines, à la fin duquel nous avons une réunion avec notre responsable de projet pour faire la revue de sprint et préparer les backlogs du sprint suivant.



FIGURE 3.1 – Schéma du principe de méthode Agile

## 3.2 Ordonnancement du projet

Le projet a été attribué au milieu du mois de Novembre, nous avons de suite pris connaissance avec notre responsable de projet afin de commencer à travailler le plus tôt possible. La fin du projet est avant la soutenance, pendant la semaine de 9 Mars, même si nous allons sans doute continuer quelques jours supplémentaires afin de rendre une version stable avec une documentation riche. Comme vu précédemment, nous avons appliqué une méthode Agile. C'est pour cela que nous obtenons le diagramme de gantt suivant [figure 3.2], où chaque semaine un certain nombre de tâches est à faire, et où des tests sont effectués une fois la tâche finie afin d'assurer que l'application est stable à chaque fin de sprint. On peut voir les 3 phases principales :

- La conception qui nous a pris un mois. En effet, nous voulions être sûr de partir sur de bonnes bases et d'avoir bien compris les besoins et contraintes,
- Le développement qui a duré presque 4 mois. C'est pendant cette période que nous avons mis en place les fonctionnalités conceptualisées précédemment.
- Les tests et la documentation : cette phase est plus particulière car elle a eu lieu durant tout le développement, à chaque fin de sprint.



## 3.3 Conception de l'application

Nous avons débuté par une longue phase de conception pour s'assurer que nous avons compris toutes les exigences du client.

Une fois le cahier des charges analysé et les technologies à utiliser pour mener à bien ce projet définies, nous avons pu commencer à concevoir notre solution. Cette phase est l'une des plus importantes. En effet, elle permet de proposer notre vision détaillée du travail à réaliser au client. Ainsi, on peut s'assurer que la production qu'on s'apprête à développer est bien en accord avec les demandes du projet et permet de répondre aux besoins initiaux.

### 3.3.1 L'analyse logicielle

Une étape essentielle de la conception est la modélisation de la solution logicielle, car elle permet de synthétiser les informations importantes et de préparer les fondations du développement. Cette modélisation passe par différents diagrammes.

## Diagramme de cas d'utilisation

Ce genre de diagramme [figure 3.3] permet de clairement définir les différents types d'utilisateurs qui seront amenés à utiliser le site. On retrouve ainsi sur le schéma suivant les quatre types d'utilisateurs évoqués lors du chapitre 2 ainsi que leurs actions sur le site :

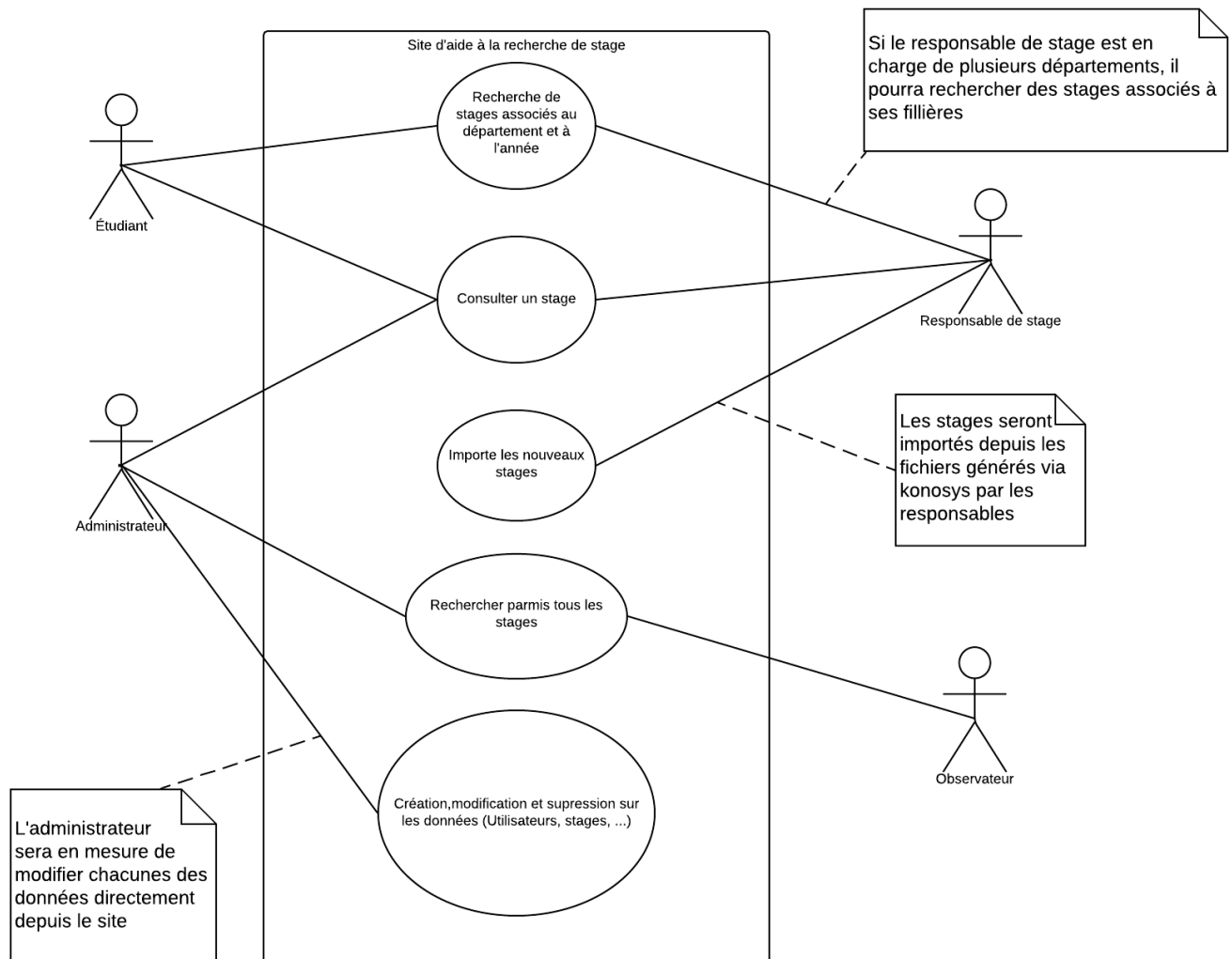


FIGURE 3.3 – Diagramme de cas d'utilisation

## Diagramme de classes

Ce diagramme [figure 3.4] permet de présenter les différentes entités que nous avons définies à partir du cahier des charges, ainsi que leurs relations entre elles :

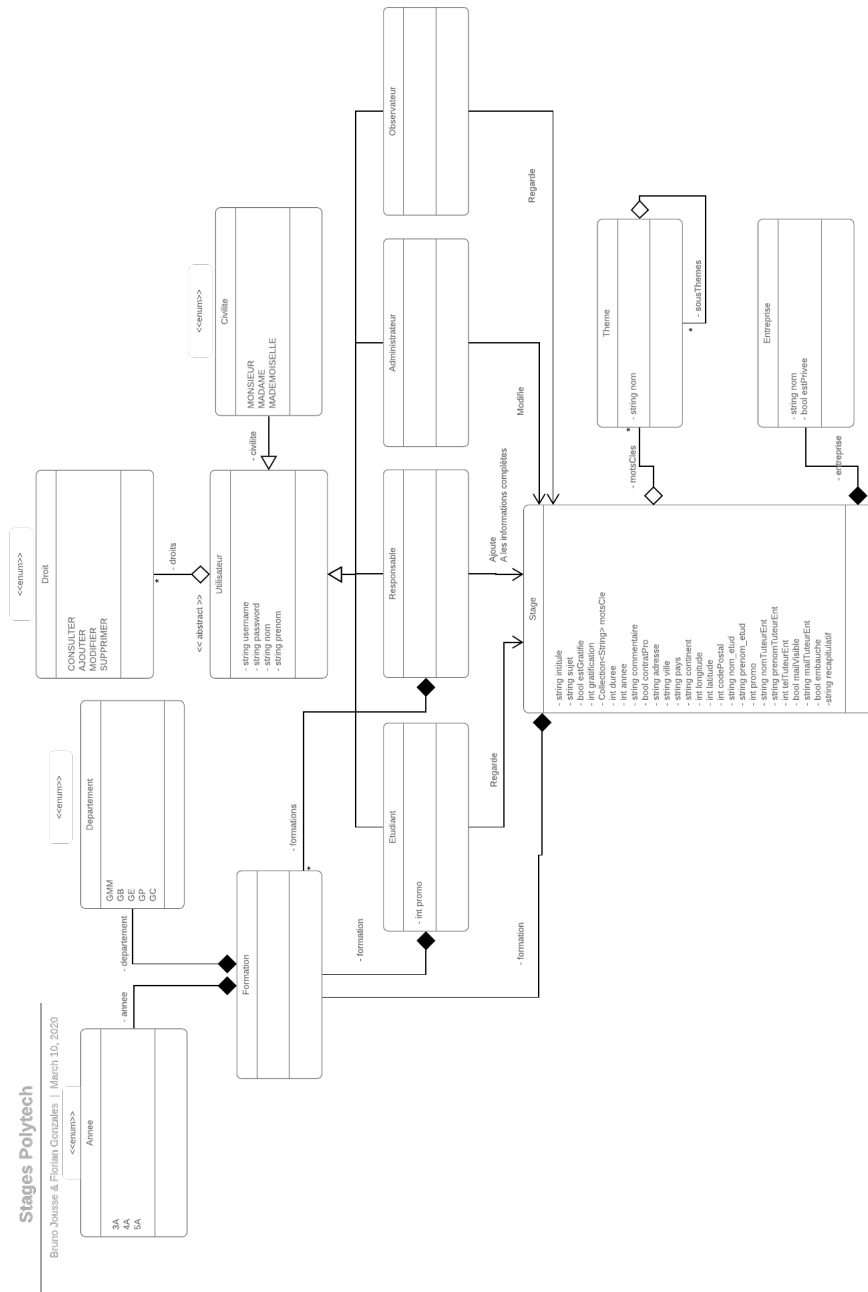


FIGURE 3.4 – Diagramme de classe du projet



### 3.3.2 Conception de la base de données

A partir des fichiers csv produits par Polytech Clermont et des réunions avec notre responsable de projet, nous avons conçu une base de données [figure 3.5] correspondant aux besoins de notre application :

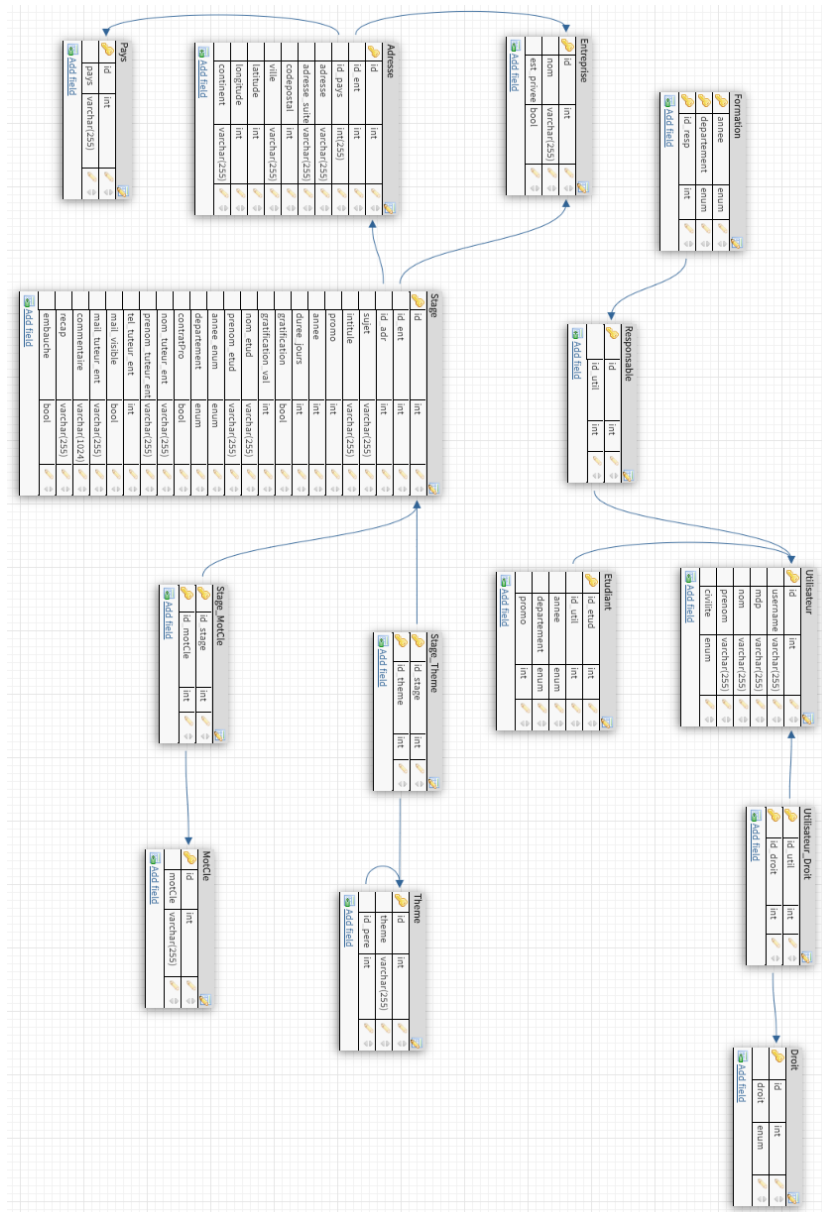


FIGURE 3.5 – Schéma de la base de données du site

### 3.3.3 Création des vues de l'application

Un moyen idéal de présenter notre vision du produit final au client est la mise en place de maquettes présentant la vue finale des différentes pages de l'application web. Pour réaliser ces vues, nous nous sommes servis du site **Moqups** permettant d'élaborer assez simplement le design d'un futur site. La figure [figure 3.6] montre une des maquettes des vues de notre application (voir les autres en annexes).

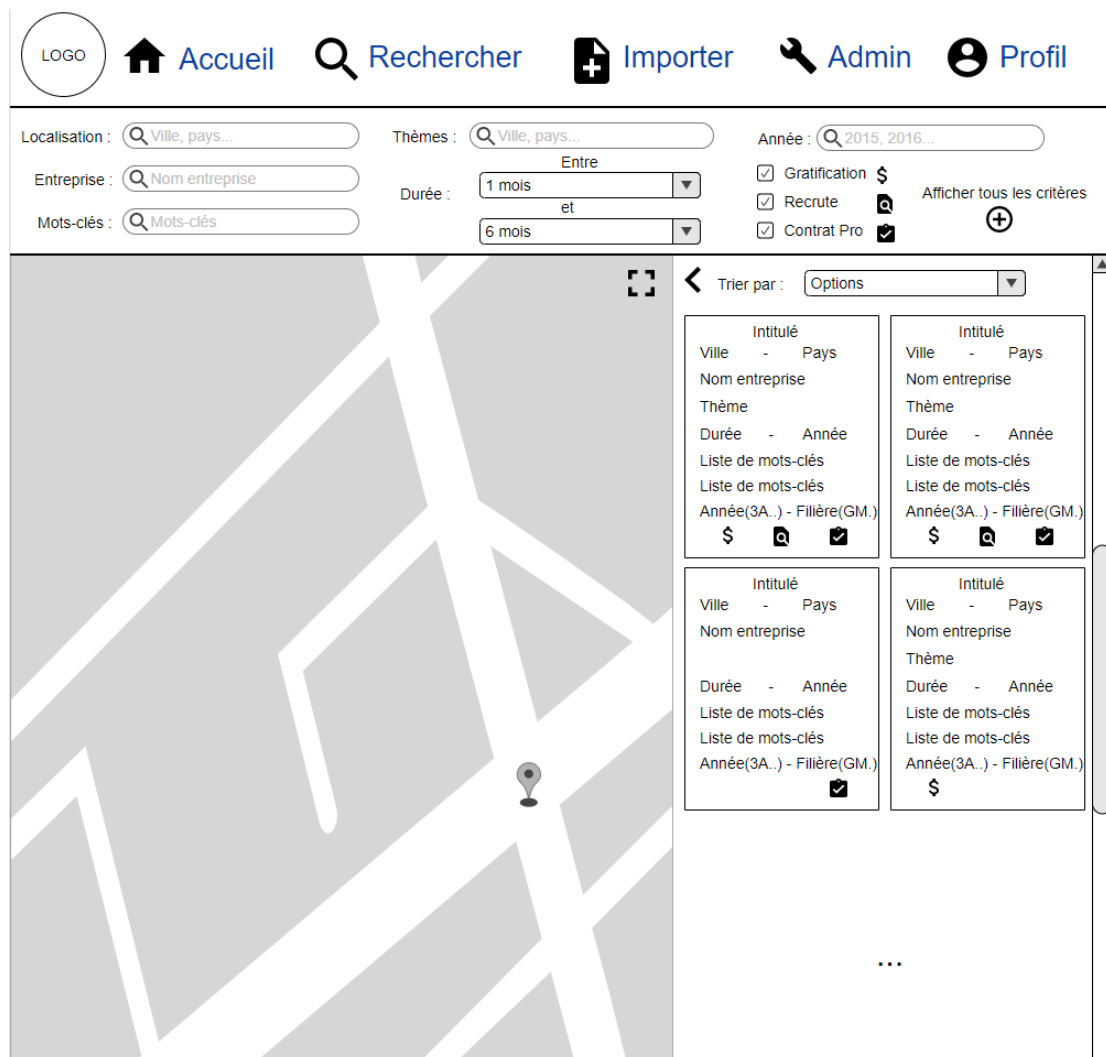


FIGURE 3.6 – Vue présentée pour l'interface de recherche des stages

Ces maquettes permettent de s'assurer que notre vision de l'application correspond à celle du client et que toutes les fonctionnalités requises sont présentes. Cela nous permet aussi de spécifier le résultat final auquel on souhaite aboutir. Cette page de recherche correspond à la majeure partie de l'application web puisqu'elle permettra aux utilisateurs

de chercher les stages en fonction de critères de recherche et de les afficher sous forme de liste avec également un point géographique sur la carte.

## 3.4 Développement du code

### 3.4.1 Développement du front-end et back-end

Une fois toutes ces étapes de conception terminées, il faut développer le code des différentes fonctionnalités que nous avons établies lors de la conception du projet. Cette phase est découpée en deux étapes :

#### Développement du back-end

Le **back-end** est la partie développement de la logique du code. C'est dans cette étape que l'on va modifier les entités et apporter la logique et l'ensemble de règles dont a besoin un élément pour correctement fonctionner. Cette partie est développée en **PHP** sur ce projet.

#### Développement du front-end

Le **front-end** est le développement de la partie visible de l'application, où l'on conçoit toutes les interfaces avec lesquelles l'utilisateur va communiquer. C'est ici que nous nous servons des maquettes produites précédemment pour produire une interface fonctionnelle, ergonomique et cohérente. Sur ce projet le front-end est développé en **CSS**, **SCSS**, **TWIG**, **HTML** et **Javascript**.

### 3.4.2 Phase de test

Une fois le code terminé, il est important de tester tous les cas possibles d'utilisation de notre nouvelle fonctionnalité. Ainsi on s'assure d'avoir un minimum d'erreurs et de retours négatifs de la part de l'utilisateur. Nous avons réalisé les tests nous même et nous avons aussi fait tester l'application web à d'autres personnes, extérieures au projet, en leur demandant de réaliser certaines actions. Ceci permet de savoir si notre travail est facilement utilisable par des personnes non habituées à utiliser nos interfaces. Les retours étaient plutôt positifs, chaque page était claire et les actions possibles étaient intuitives.

### 3.4.3 Maintien du code

Afin d’avoir un code propre, compréhensible et facilement maintenable par nous et par d’éventuels autres développeurs amenés à travailler sur ce projet, il est important d’accorder du temps pour nettoyer régulièrement le code et pour écrire des commentaires décrivant la logique.

Dès lors que beaucoup de fonctionnalités sont ajoutées et que le code devient peu lisible par endroits, nous avons été amené à le retravailler en apportant les modifications suivantes :

- **Limiter la duplication de code.** Cette étape consiste à écrire des fonctions intermédiaires afin d’éviter d’écrire plusieurs fois la même chose. Ainsi apporter des modifications sur ce code sera beaucoup plus facile et rapide.
- **Commenter les parties du code source les moins évidentes.** Ainsi si nous devons réexaminer une partie de notre travail ou si un autre développeur doit la modifier, la compréhension de la logique est beaucoup plus facile. Nous prévoyons aussi de fournir un manuel d’utilisation et de fonctionnement de l’application.

# Chapitre 4

## Présentation de la solution réalisée

Dans cette partie nous allons vous présenter l'état d'avancement actuel du projet et les résultats que nous avons obtenus. A partir du cahier des charges et de la phase de conception, nous avons été capable de réaliser la majorité des fonctionnalités prévues, même si certaines ne sont pas encore finalisées.

### 4.1 Recherche des stages

La fonctionnalité principale de l'application web est la recherche de stages effectués précédemment par des élèves ayant eu la même formation. Pour rappel, cette page [figure 4.1] doit permettre aux étudiants de rechercher des stages correspondant à leurs critères et d'afficher les résultats sous forme de liste et sur une carte du monde. Cette interface est accessible à tous les utilisateurs de l'application et se décompose en plusieurs espaces développés ci-dessous :

- **Filtrer les stages**
- **Affichage des résultats sur la carte**
- **Affichage des résultats en liste**

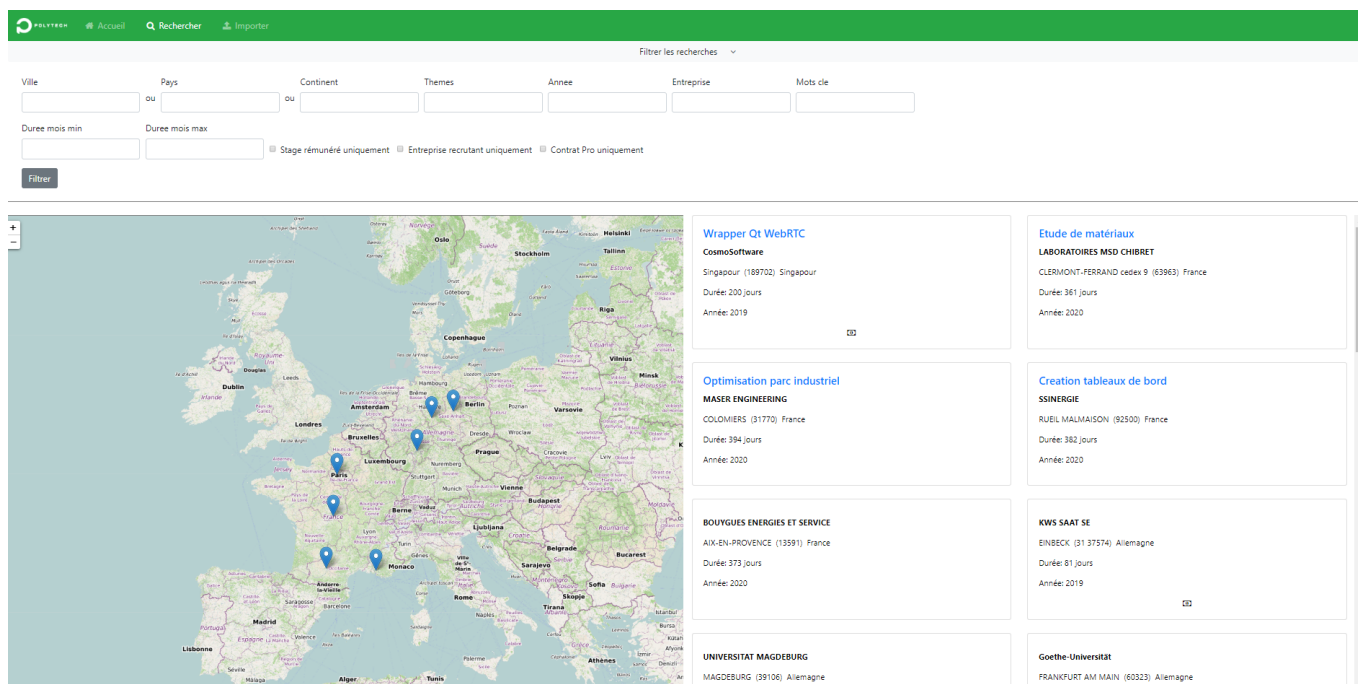


FIGURE 4.1 – Affichage recherche de stages

### 4.1.1 Filtrer les stages

Afin que les étudiants puissent rechercher des stages qui leur correspondent, une barre de filtrage des données a été mise en place [figure 4.2]. Nous avons dans notre cahier des charges les différents critères de recherche que les utilisateurs doivent pouvoir utiliser, comme par exemples la localisation du stage, le fait que le stage soit rémunéré, le sujet du stage...

Nous avons donc mis en place un formulaire possédant ces différents champs pour que l'utilisateur puisse affiner sa recherche affichée sur la **carte** et la **liste des stages**.

The screenshot shows the 'Filtrer les recherches' form. It has a grey header with the text 'Filtrer les recherches' and a dropdown arrow. Below the header, there are several input fields for filtering: 'Ville', 'Pays', 'Continent', 'Themes', 'Annee', 'Entreprise', and 'Mots cle'. The 'Pays' field is highlighted with a blue border. There are also checkboxes for 'Stage rémunéré uniquement', 'Entreprise recrutant uniquement', and 'Contrat Pro uniquement'. A 'Filtrer' button is located at the bottom left of the form.

FIGURE 4.2 – Affichage du formulaire de filtrage

### 4.1.2 Liste des stages

La liste des stages se présente sous cette forme [figure 4.3] :

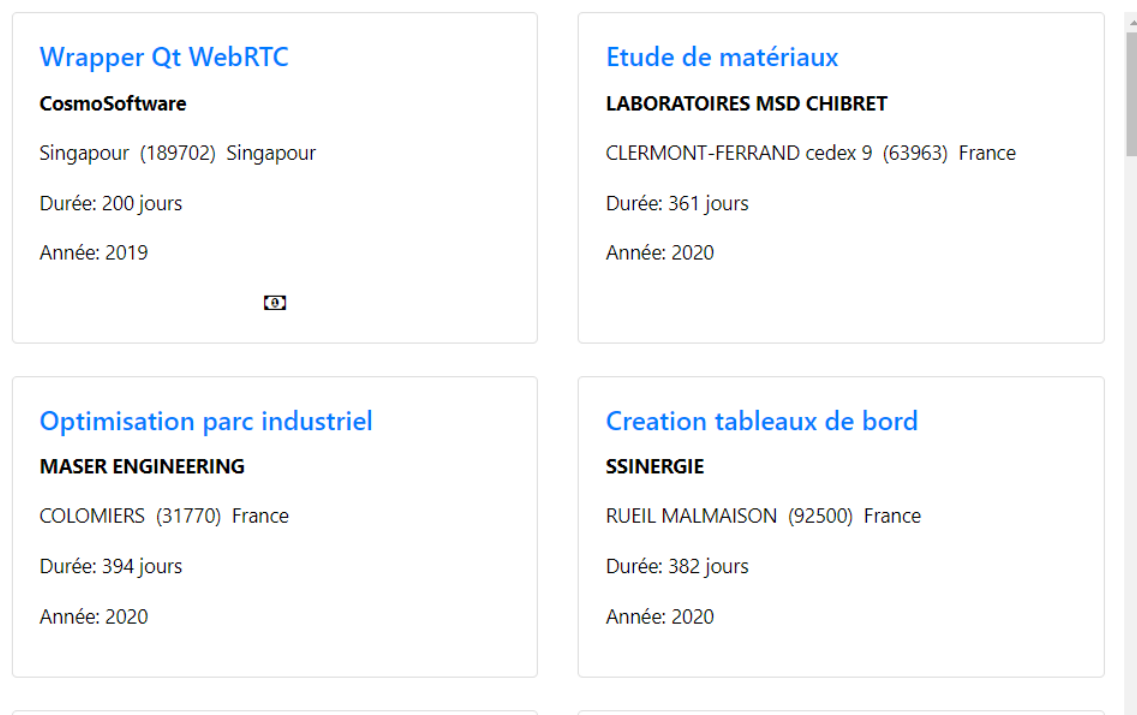


FIGURE 4.3 – Affichage liste des stages

Notre solution liste les stages correspondants aux critères précisés par l'étudiant dans la barre de filtre précédemment présentée. Chaque stage est présenté par une petite carte regroupant les informations essentielles sur le stage :

- Le sujet du stage,
- L'entreprise d'accueil,
- L'adresse,
- La durée en jours,
- L'année de réalisation du stage,
- Une liste d'icônes indiquant si le stage était rémunéré, si le stage a mené à une embauche ou encore si le stage est un contrat de professionnalisation (disponible uniquement en 5ème année).

L'étudiant peut ensuite accéder à un résumé détaillé du stage en cliquant sur la carte. Il obtiendra cette page [figure 4.4] :

## Développeur Symfony

**Sujet:** Un sujet de stage

Ce stage a été réalisé en **2017** et a duré **180 jours**

**Information sur l'étudiant lors de la réalisation du stage:**


Nom	Gonzales
Prénom	Florian
Promotion	2020
Année formation	4A
Département	G.P


**Information sur le stage:**


Entreprise	Credit Agricole (publique)
Adresse	32 rue de la Rotonde -
Ville	CLERMONT-FD (03310)
Pays	France
Continent	Europe

**Tuteur enseignant du stage:** Jousse Bruno


**Informations complémentaires:**


 Le stage est gratifié

 Le stage n'est pas un contrat pro

 Le stage a mené à une embauche

**Contacts du tuteur enseignant:**

 06 00 00 00 01

 fl.gonzales@gmail.com

**Mots-clé:** Aucun mots-clé associé

**Commentaire:**

Un petit commentaire sympa

[Accéder au récapitulatif](#)

FIGURE 4.4 – Détails d'un stage

Ici toutes les informations relatives au stage sont présentes. En plus des informations précédemment décrites, quelques détails supplémentaires sont présents :

- Les détails sur l'étudiant,
- Les contacts du tuteur de stage si ils sont accessibles pour les étudiants,
- Les mots-clés associés au stage,
- Les thèmes du stage,
- Un commentaire de l'étudiant sur le stage,
- Un lien vers un diaporama produit par les étudiants récapitulant les informations du stage.



### 4.1.3 Afficher les stages sur la carte

La figure [figure 4.5] montre l’affichage des résultats d’une recherche sur la carte :

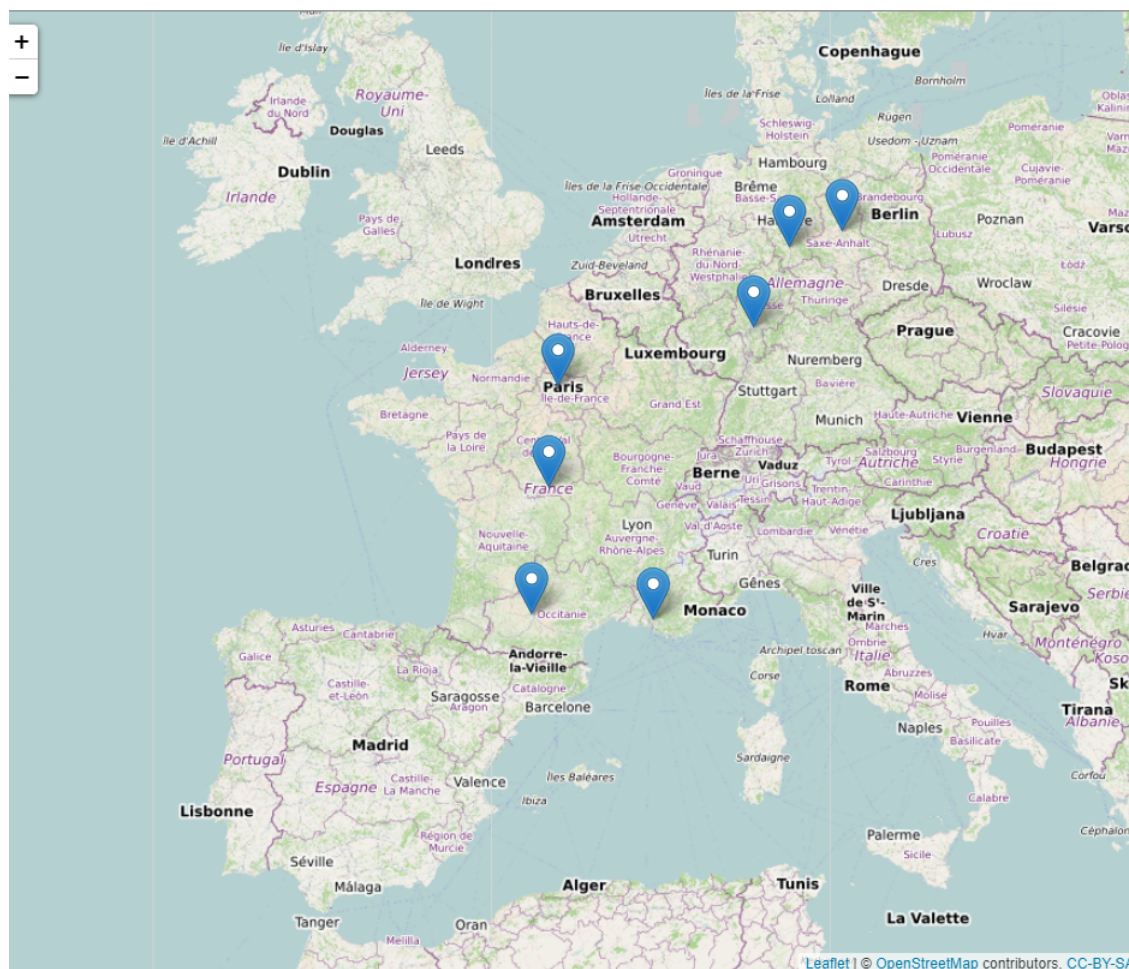


FIGURE 4.5 – Affichage de la carte

Il était primordial que notre solution permette aux utilisateurs de pouvoir visualiser les stages sur une carte mondiale. Dans cet affichage, l’adresse de chaque stage du résultat de la recherche est affichée sur la carte par un marqueur bleu. A partir de là, l’utilisateur peut cliquer sur un marqueur pour afficher une petite carte décrivant le stage. Enfin, si l’étudiant veut accéder aux informations complètes sur le stage, il peut à nouveau cliquer sur la description du stage pour aller sur sa fiche [figure 4.4].

Pour réaliser cette fonctionnalité, nous avons utilisé la bibliothèque Leaflet ([Lea]). Leaflet est la principale bibliothèque JavaScript à code source public pour les cartes interactives adaptées aux téléphones portables. Utiliser cette bibliothèque permet de facilement importer des cartes sur une application web et de placer des informations dessus comme par exemples des marqueurs. Par exemple, la bibliothèque propose des fonctions comme **addMarker()** permettant de placer un point sur la carte à l'endroit indiqué par les coordonnées entrées en paramètres de la fonction. Toutes ces fonctions permettent de facilement manipuler les cartes.

## 4.2 Importer des stages

Polytech-Clermont possède déjà une base de données répertoriant toutes les informations sur les stages effectués. Ces données sont exportées sous la forme de fichier XLS ou CSV. Ainsi l'ajout de données dans notre application passe par l'importation de tels fichiers, permettant de créer tous les stages du fichier dans notre base de données [figure 4.6].

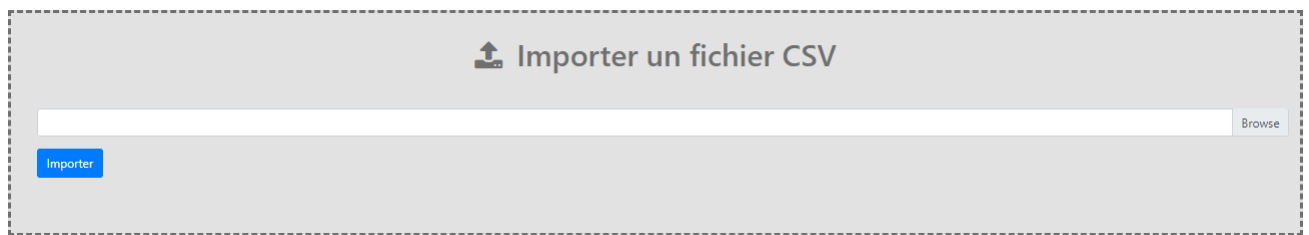


FIGURE 4.6 – Affichage importation de fichier

### 4.2.1 Enregistrer des stages à partir de fichiers tableurs

Pour lire les fichiers tableurs, nous avons utilisé la librairie PhpOffice. Elle crée un objet adapté au type de fichier (xls, csv) et nous permet de parcourir facilement les informations du fichier (ligne par ligne par exemple). Nous récupérons la première ligne du fichier pour obtenir le nom des colonnes, et ensuite nous créons un nouvel objet "Stage" pour chaque ligne du document s'il n'existe pas déjà dans la base de données.

### 4.2.2 Déterminer les coordonnées d'une adresse

Le framework Leaflet requiert des coordonnées GPS (longitude et latitude) afin de pouvoir afficher les points sur la carte. Or nous ne possédons pas ces points dans les données importées dans la base de données. C'est pourquoi nous avons dû les calculer à partir des

adresses, ce que l'on appelle faire du **geocode**. Lors de l'importation des données relatives aux stages, nous envoyons les adresses à une API nous retournant les coordonnées GPS. Nous avons comparé différentes API, en regardant principalement le coût (nous voulions une API gratuite) et son nombre d'appels possibles (nous voulions être sûr que les responsables pourraient importer tous les stages sans être limités). L'API qui correspondait au mieux à nos critères est **MapQuest Geocoding API** ([Map ]), qui est une API utilisable gratuitement permettant d'effectuer 15000 transactions par mois, ce qui est largement suffisant.

Son utilisation est très simple : nous effectuons une requête HTTP POST où nous fournissons en paramètres notre clé d'API et, au format JSON, la liste des adresses dont les informations sont réparties dans "rue", "ville" et "pays". Puis l'API nous renvoie au format JSON la liste des coordonnées GPS. Au moment d'ajouter les informations du stage dans la base de données, nous sauvegardons aussi ses coordonnées.

### 4.2.3 Ajouter les champs manquants

Nous n'avons pas encore eu le temps de réaliser cette partie, mais nous aimerions ajouter le fait que les responsables de stage puissent modifier les données qu'ils importent en ajoutant les champs qui ne sont pas présents dans le fichier tableur, le tout avec une interface lisible et simple d'utilisation. Par exemple nous aimerions qu'un responsable de stage puisse associer facilement des thèmes avec les stages importés.







## 4.3 Administrer les données

Toute application a besoin d'un côté administrateur où des utilisateurs ayant les privilèges requis peuvent manipuler les données à l'aide d'une interface sans avoir à écrire des requêtes SQL sur la base de données.

Nous avons donc réalisé cette implémentation [figure 4.7]. L'administrateur peut lister, éditer, ajouter et supprimer toutes données (stage, adresse, entreprise, mot clé, thème).

Stages Adresses Entreprises Mots-clé Thèmes

+ Rechercher :

Intitulé ↑	Adresse ↑	Pays ↑	Année ↑	Prénom étudiant ↑	Nom étudiant ↑	Promo ↑	Actions ↑
Community Manager Novice	32 rue de la Table MARSEILLE(13000)	France	2017	Florian	Gonzales	2020	 
Développeur Symfony	32 rue de la Rotonde CLERMONT-FD(03310)	France	2017	Florian	Gonzales	2020	 
Management d'une équipe	14 rue du Paradis PARIS(75000)	France	2017	Florian	Gonzales	2020	 

Affichage de l'élément 1 à 3 sur 3 éléments

Précédent 1 Suivant

FIGURE 4.7 – Affichage administration des données

## 4.4 Navigation

Il existe différents types d'utilisateurs sur notre application web, ayant chacun des droits différents. Par conséquent, nous avons donc du développer une barre de navigation évolutive [figure 4.8] en fonction de l'utilisateur connecté :



FIGURE 4.8 – Barre de navigation

### 4.4.1 Icônes de droite

Cette partie de la navigation est commune à tous les utilisateurs du site. Elle comporte **Ma liste**, qui redirige l'utilisateur vers la liste des stages qu'il a décidés d'ajouter dans cette liste et **le nom de l'utilisateur connecté**.

### 4.4.2 Icônes de gauche

En revanche, cette partie là de la navigation est amenée à évoluer en fonction des droits de l'utilisateur connecté.

### Pour les étudiants

Les étudiants auront accès aux deux premiers liens, permettant respectivement d'aller sur la page d'accueil du site, présentant les différentes fonctionnalités de l'application, et la page de recherche précédemment décrite.

### Pour les responsables de département

Ces utilisateurs ont accès à un lien supplémentaire : **l'importation des fichiers**, permettant d'importer des stages dans la base de données.

### Pour l'administrateur

Un dernier lien supplémentaire sera disponible pour cet utilisateur : **l'administration**, le redirigeant sur l'interface d'administration des données.

## 4.5 Connexion

L'application sera uniquement disponible depuis le réseau Polytech. Pour éviter aux utilisateurs de devoir créer un compte lors de leur première connexion à l'application, nous avons décidé de gérer la connexion avec **LDAP** ([LDA ]). LDAP nous permet d'aller chercher des informations sur la base de données de Polytech. Ainsi pour se connecter, les étudiants n'auront qu'à spécifier les identifiants qu'ils utilisent pour se connecter sur les machines de l'établissement. Cela nous permet d'avoir des informations sur l'utilisateur qui se connecte, comme par exemple son rôle ou encore le département auquel il est associé. Grâce à ces informations, on peut donc afficher à l'utilisateur les données qui lui correspondent.

## 4.6 Déploiement de l'application

Puisque l'application ne doit être disponible que sur le serveur de l'école, nous devons donc héberger le site sur ces serveurs. Pour cela nous avons créé une machine virtuelle à l'aide du logiciel **VMWare** :

Une fois notre application web hébergée sur cette machine virtuelle, nous la transférons sur les serveurs de l'école. Quand l'application sera sur les serveurs de l'école, elle sera accessible aux étudiants et prête à être utilisée.

# Conclusion et perspectives

Notre projet était de créer une application web permettant aux étudiants de pouvoir chercher les stages effectués les années précédentes par des étudiants ayant eu la même formation qu’eux.

C’est avec cet objectif en tête que nous avons réalisé l’application, de sa conception à son déploiement. En nous aidant de frameworks et de bibliothèques tels que Symfony ou Leaflet, nous avons été capable de mener à bien ce projet en ayant fini les fonctionnalités principales comme la recherche ou l’importation.

Cependant, certains points sont encore à améliorer, et nous avons eu l’idée d’autres fonctionnalités que nous n’avons pas eu le temps de développer. Nous pourrions par la suite ajouter une interface ergonomique pour permettre aux responsables de stage d’ajouter les champs manquants comme les thèmes d’un stage. Nous avons également pensé, pour les étudiants, à leur donner la possibilité d’enregistrer des stages dans un panier pour pouvoir les consulter par la suite plus facilement et les exporter en pdf pour les montrer à leur responsable de stage.

Ce projet nous a donné l’opportunité de travailler sur une application web complète : Florian a pu renforcer ses compétences dans le domaine du **développement web full-stack**, domaine dans lequel il sera amené à travailler plus tard. Les connaissances et compétences qu’il a consolidées en développement web le confortent dans son désir de se diriger dans le développement web au terme de cette troisième année.

Bruno quant à lui n’avait pas un excellent souvenir du développement web, mais il a pu le redécouvrir en apprenant le framework Symfony. Il s’est concentré sur le back-end de l’application et a pu utiliser la majorité des fonctionnalités propres à Symfony. Fort de cet apprentissage, il est désormais beaucoup plus intéressé à l’idée de poursuivre dans du développement web, en se concentrant sur le back-end.

Tous ces éléments font que nous avons particulièrement apprécié ce projet et que nous sommes impatients de le voir arriver à son terme avant le départ en stage.





# Annexes



FIGURE 9 – Maquette de l'accueil du site

LOGO

Accueil

Rechercher

Importer

Admin

Profil

Utilisateurs

Stages

Prénom :

Nom :

Année :

Filière :

...

+

Nom	Prénom	Rôle	Année	...	Action

FIGURE 10 – Maquette de la gestion des données pour l’administrateur

Stages Polytech

Connexion

Identifiant :

Mot de passe :

Se connecter

[Première connexion ?](#)  
[Mot de passe oublié ?](#)

FIGURE 11 – Maquette de la page de connexion

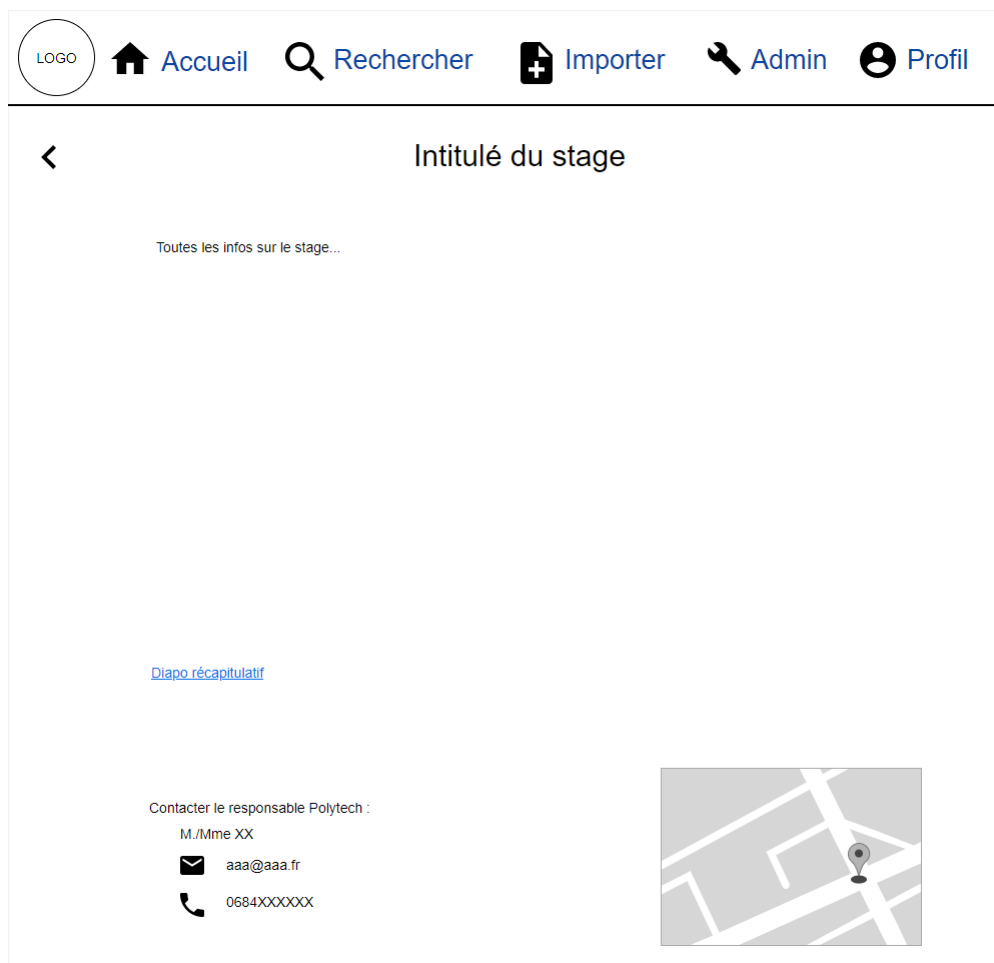






FIGURE 12 – Maquette de la fiche détaillée d'un stage


LOGO

 [Accueil](#)


 [Rechercher](#)

 [Importer](#)

 [Admin](#)


 [Profil](#)

Importer un fichier .csv



Sélectionner un fichier

Nom du fichier importer



Importer

FIGURE 13 – Maquette de l'importation

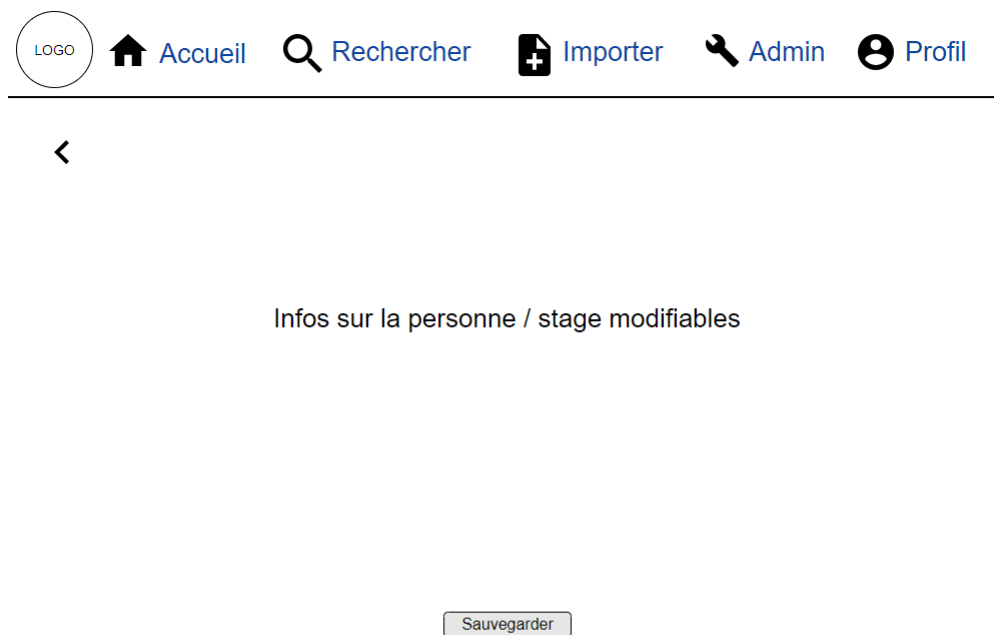


FIGURE 14 – Maquette de la modification des données d'un stage



# Bibliographie

- [Boo ] *Documentation de Bootstrap*. In [https ://getbootstrap.com/docs/4.4/getting-started/introduction/](https://getbootstrap.com/docs/4.4/getting-started/introduction/).
- [Doc ] *Documentation de Doctrine*. In [https ://symfony.com/doc/current/doctrine.html](https://symfony.com/doc/current/doctrine.html).
- [Git ] *Documentation de Git*. In [https ://git-scm.com/doc](https://git-scm.com/doc).
- [Gra ] *Site de formation de Grafikart*. In [https ://www.grafikart.fr/](https://www.grafikart.fr/).
- [LDA ] *Documentation de l'utilisation de LDAP en PHP*. In [https ://www.php.net/manual/en/book.ldap.php](https://www.php.net/manual/en/book.ldap.php).
- [Lea ] *Documentation de LeafLet*. In [https ://leafletjs.com/reference-1.6.0.html](https://leafletjs.com/reference-1.6.0.html).
- [Map ] *Documentation de MapQuest*. In [https ://developer.mapquest.com/documentation/](https://developer.mapquest.com/documentation/).
- [PHP ] *Documentation de PHP*. In [https ://www.php.net/manual/fr/index.php](https://www.php.net/manual/fr/index.php).
- [Sym ] *Documentation officielle du framework Symfony*. In [https ://symfony.com/doc/current/index.html](https://symfony.com/doc/current/index.html).
- [Twi ] *Documentation de Twig*. In [https ://twig.symfony.com/doc/3.x/](https://twig.symfony.com/doc/3.x/).