

IUT Nancy Charlemagne

Université de Lorraine

2 ter boulevard Charlemagne

BP 55227

54052 Nancy Cedex

Département Informatique



**Réalisation d’un site de demande de partenariat, sponsoring, mécénat**

Rapport de stage de DUT Informatique

Entreprise : Demathieu Bard

Ian Bialo

Tuteur : Anne-Cécile Schneider

Année universitaire 2017-2018



IUT Nancy Charlemagne

Université de Lorraine

2 ter boulevard Charlemagne

BP 55227

54052 Nancy Cedex

Département Informatique

**Réalisation d’un site de demande de partenariat, sponsoring, mécénat**

Rapport de stage de DUT Informatique

Entreprise : Demathieu Bard

Ian Bialo

Tuteur : Anne-Cécile Schneider

Année universitaire 2017-2018

# **Remerciements**

Avant de commencer la présentation de ce rapport, je tiens à remercier toutes les personnes qui ont contribué au succès de mon stage et qui m’ont aidé lors de la rédaction de ce rapport.

Tout d’abord je tiens à remercier ma tutrice de stage Mme Anne-Cécile Schneider qui m’a encadré tout le long de mon stage en entreprise et qui n’a pas hésité à m’apporter son aide lorsque j’en avais besoin. Je tiens à adresser également mes remerciements à toute l’équipe pédagogique de mon Université qui m’ont apporté les connaissances nécessaires pour la réalisation de ma mission durant mon stage au sein de l’entreprise.

Je voudrais également remercier vivement mon tuteur de stage M. Pierre George pour le temps précieux qu’il m’a accordé, pour son accueil mais aussi pour toutes les connaissances nouvelles qu’il a pu m’apporter durant le stage. Je voudrais également remercier toute l’équipe du département informatique de l’entreprise Demathieu Bard dans son ensemble pour leur accueil, leur gentillesse, leur esprit d’équipe qui ont toujours été présents lorsque j’avais besoin de renseignements.

Je voudrais également remercier la responsable du département de la communication au sein de l’entreprise ayant proposé le sujet de stage Mme Françoise Philibert pour avoir pris du temps pour suivre mon avancement sur le projet et pour avoir répondu à toutes mes questions.

Je tiens également à remercier mon père M. Michal Bialo qui m’a aidé à me mettre en relation avec l’entreprise dans laquelle j’ai effectué mon stage ainsi que la responsable des ressources humaines régional Mme Nadine Repovy qui m’a contacté pour me proposer une offre de stage au sein de l’entreprise.

Enfin, je tiens à remercier toutes les personnes qui m’ont conseillé et relu lors de la rédaction de ce rapport de stage c’est-à-dire ma famille ainsi que mes amis et mes camarades de promotion.

**Table des matières**

[**Remerciements** 2](#_Toc516673828)

[**Introduction** 5](#_Toc516673829)

[**I. L’entreprise Demathieu Bard** 6](#_Toc516673830)

[1) Présentation de l’entreprise et du secteur d’activité 6](#_Toc516673831)

[A. Historique de l’entreprise 6](#_Toc516673832)

[B. Secteur d’activité de l’entreprise 6](#_Toc516673833)

[2) Le département informatique 6](#_Toc516673834)

[3) Mise en relation avec l’étudiant 7](#_Toc516673835)

[**II. Réalisation et évolution du projet** 9](#_Toc516673836)

[1) La mission de stage 9](#_Toc516673837)

[A. La problématique 9](#_Toc516673838)

[B. Les objectifs 10](#_Toc516673839)

[2) Technologies utilisées 11](#_Toc516673840)

[3) Réalisation du projet 12](#_Toc516673841)

[A. Sprint 0 – Introduction au projet initial 13](#_Toc516673842)

[B. Sprint 1 – Développement de l’application 15](#_Toc516673843)

[C. Sprint 2 – Mise en place sur le serveur distant 20](#_Toc516673844)

[D. Sprint 3 – Finalisation du projet 23](#_Toc516673845)

[4) Structure et fonctionnement final du projet 27](#_Toc516673846)

[A. Structure du projet 27](#_Toc516673847)

[B. Fonctionnement final de l’application 30](#_Toc516673848)

[**III)** **Disposition du site et fonctionnalités associées** 33](#_Toc516673849)

[1) Le FrontOffice 33](#_Toc516673850)

[A. Le formulaire 33](#_Toc516673851)

[B. L’après-formulaire 35](#_Toc516673852)

[2) Le BackOffice 36](#_Toc516673853)

[A. Le menu de connexion 37](#_Toc516673854)

[B. La liste des projets 37](#_Toc516673855)

[C. Les détails du projet 39](#_Toc516673856)

[D. La recherche de projet 42](#_Toc516673857)

[E. La gestion de compte 42](#_Toc516673858)

[F. Création de compte 44](#_Toc516673859)

[G. Complément 45](#_Toc516673860)

[3) Page d’erreur 404 45](#_Toc516673861)

[**Conclusion** 47](#_Toc516673862)

[**Bibliographie** 48](#_Toc516673863)

[**Annexe** 49](#_Toc516673864)

# **Introduction**

Dans le cadre du DUT Informatique, un stage de fin d’étude est imposé aux étudiants pour pouvoir valider leur diplôme. Ce stage a pour but d’apporter une première transition pour l’étudiant du milieu scolaire vers le monde du travail tout en lui permettant de mettre en œuvre les connaissances reçues durant ses deux années à l’IUT au sein d’un équipe de professionnel avec des objectifs précis. J’ai choisi d’effectuer mon stage d’une durée de dix semaines au sein de l’enrteprise Demathieu Bard pour pouvoir approfondir mes connaissances en développement web mais aussi découvrir de nouvelles notions liées à celui-ci.

L’entreprise Demathieu Bard reçoit de la part de multiples prestataires externes divers demandes d’aides. Ces demandes peuvent être des partenariats, du sponsoring, ou encore des demandes de mécénat. Toutes ces demandes sont envoyées au département de la communication de l’entreprise qui est chargée de traiter toutes ces demandes. Cependant, elles sont envoyées par mail et ne sont donc pas structurées, la lecture de celles-ci peut donc être parfois difficile et il est possible de voir des informations inutiles apparaitre dans ces mails.

Le département de la communication de l’entreprise souhaite donc pouvoir structurer ces informations à l’aide d’un formulaire à compléter sur une page web par les organismes externes contenant toutes les informations nécessaires à l’entreprise qui pourront être visible sur une autre page web réservée exclusivement au département de la communication. La page du formulaire serait accessible au travers d’un onglet à partir de la page déjà existante de l’entreprise.

Dans un premier temps, nous verrons l’entreprise dans sa généralité, son secteur d’activité ainsi que son département informatique, dans un second temps nous verrons en détails les missions qui m’ont été confiées lors du stage pour finalement voir dans un troisième temps le résultat du travail réalisé avant de dresser un bilan du stage dans une brève conclusion.

# C:\Users\ibialo\Downloads\LOGO_DB_2012_horizontal_2COUL_ok.png**I. L’entreprise Demathieu Bard**

## **Présentation de l’entreprise et du secteur d’activité**

### Historique de l’entreprise

Demathieu Bard est une entreprise française du secteur de la construction crée en 1861 à Rohrbach-lès-Bitche (entre Metz et Strasbourg, en Moselle) par Pierre et Julien Demathieu. Initialement, l’entreprise avait pour nom Demathieu avant d’être changée en Demathieu et Bard en 1945 après que Raymond Bard, jeune ingénieur des Travaux publics n’entre dans l’entreprise Demathieu pour finalement subir une refonte en 2013 pour l’actuel Demathieu Bard.

### Secteur d’activité de l’entreprise

Le siège d’activité de Demathieu Bard est situé à Montigny-lès-Metz mais l’entreprise possède plus agences implémentées aux quatre coins de la France ainsi qu’en Île de France mais aussi à l’international dans des pays tels que le Canada, le Luxembourg ou encore l’Allemagne. L’entreprise a commencé à se spécialiser au départ dans le génie civil et les ouvrages d’art pour progressivement élargir son activité vers le bâtiment et l’industrie. Avec son chiffre d’affaire de 900 millions d’euros en 2014 et ses 2 600 employés, il s’agit d’une des principales entreprises indépendantes du Bâtiment Travaux Public (BTP) en France.

## **Le département informatique**

Parmi tous les départements situés au sein du siège social de l’entreprise se trouve un département informatique dit le département des « systèmes d’information ». Il s’agit, comme son nom l’indique, de la section chargée de traiter tous les besoins informatiques de l’entreprise. Il est divisé en plusieurs pôles composé de deux à trois personnes. Parmi ces pôles, on retrouve des personnes chargées de :

* L’administration réseau
* L’administration Système
* La gestion de parc et Achats
* La maintenance (ce sont les techniciens informatique)

On retrouve à la tête des différents pôles les différents responsables :

* Le responsable d’Exploitation
* Le responsable Applications de Gestion
* Le responsable d’outils collaboratif

A la tête de la direction des systèmes d’informations se trouve le directeur des systèmes d’informations qui est M. Pierre George.

Le département informatique est situé sur un étage entier du siège social (plus précisément au deuxième) et est divisé dans différentes pièces selon les pôles vu précédemment. Chaque personne peut donc travailler au sein d’une même pièce avec d’autres personnes s’il s’agit de personnes issues du même pôle ou comme dans mon cas seul. Chaque employé à son propre bureau avec son propre ordinateur ainsi que d’autres outils selon les besoins du pôles (des micros pour les personnes chargées de la maintenance par exemple). On retrouve également sur cet étage des salles de réunion ainsi qu’une salle où il est possible de se désaltérer et de prendre une pause. Il est à noter que le département informatique à pour coutume de désigner chaque semaine une personne chargée de ramener les vendredis des croissants pour tout le département.

## **Mise en relation avec l’étudiant**

Lors de ma recherche de stage, j’ai eu la chance d’avoir été mis en relation avec cette entreprise par le biais de la responsable des ressources humaines régional Mme Nadine Repovy qui connait mon père M. Michal Bialo qui lui aussi travaille dans l’entreprise et qui m’a apporté son aide dans la recherche de l’entreprise.

Pour mon stage de fin d’étude, j’ai décidé de rejoindre cette entreprise puisque, contrairement aux autres propositions de stage que j’ai reçu, celle-ci proposait le sujet le plus intéressant. En effet, la réalisation d’une application web est quelque chose qui m’intéresse énormément et en plus de l’enseignement reçu à l’IUT, je voulais avoir un approfondissement de ces connaissances et je voulais également découvrir de nouvelles choses que ce soit de nouvelles fonctionnalités ou de nouvelles notions. De plus, travailler pour un département informatique au sein d’une entreprise n’ayant pas son activité tournée vers l’informatique est également quelque chose que je voulais découvrir. En effet, il est intéressant de découvrir comment fonctionne un département informatique contrairement à une entreprise entièrement orientée vers l’informatique puisque les objectifs devaient, à mon sens, pas être les mêmes.

J’ai été accueilli le premier jour par mon tuteur au sein de l’entreprise qui m’a fait visiter le bâtiment et m’a présenté à l’ensemble des personnes responsables du service informatique. Il m’a ensuite expliqué quel était son rôle au sein de l’entreprise ainsi que le rôle de chacune des personnes du service informatique. Il m’a finalement montré mon poste de travail et m’a expliqué les premières tâches à réaliser. Durant tout le stage, j’ai travaillé seul dans une pièce située à côté du bureau de mon tuteur avec à ma disposition un bureau, un ordinateur contenant initialement aucune application de développement sur lequel j’effectuais le développement d’application mais aussi une ardoise pour pouvoir effectuer des croquis et prendre des notes lorsque j’en avais besoin. J’ai également eu un téléphone avec un annuaire intégré me permettant de contacter à distance les personnes suivant de près l’évolution du projet mais aussi une adresse mail ([ian.bialo@demathieu-bard.fr](mailto:ian.bialo@demathieu-bard.fr)) me permettant également de communiquer avec les personnes suivant de près l’évolution du projet ainsi que de recevoir par ces personnes les instructions détaillés vis-à-vis du projet.

# **II. Réalisation et évolution du projet**

## **La mission de stage**

### La problématique

L’entreprise Demathieu Bard reçoit de la part de multiples prestataires externes divers demandes d’aides. Ces demandes peuvent être des partenariats, du sponsoring, ou encore des demandes de mécénat. Toutes ces demandes sont envoyées au département de la communication de l’entreprise qui est chargée de traiter toutes ces demandes. Initialement, toutes ces demandes sont faites par mail, c’est-à-dire que chaque personne voulant envoyer une demande se contente d’envoyer un mail à l’entreprise en y incluant les éléments de leurs demandes ainsi que des pièces jointes pour valoriser la demande. Cependant, recevoir toutes ces demandes par mail n’est pas la meilleure des solutions puisque cela possède plusieurs contraintes majeures :

* La non-organisation des informations dans le mail rend la lecture de celui-ci plus difficile.
* Tous les éléments inscrits dans le mail ne sont pas forcément tous pertinent, il est donc nécessaire de trier toutes les informations pour faire ressortir tous les éléments intéressants. Cela concerne également les documents mis en pièce jointe.
* Il est nécessaire de structurer et stocker toutes les informations ailleurs (dans un tableur par exemple) pour pouvoir avoir une sauvegarde de ceux-ci et ainsi ne pas avoir à constamment devoir retrouver les mails concernant les demandes.

Le département de la communication de l’entreprise souhaite retravailler le système de demandes de manière à supprimer toutes les contraintes vues précédemment. Pour cela, le département a envisagé de remplacer le système de demandes par mail pour privilégier les demandes au sein d’un formulaire présent sur une page web. En effet, il a été imaginé un onglet spécifique aux demandes qui serait présent sur la page web de l’entreprise déjà existante qui redirigerait vers une nouvelle page contenant un formulaire avec quelques indications lié à celui-ci.

Ce formulaire serait un formulaire classique où il sera possible de remplir des champs pour la quasi-majorité obligatoires qui seront eux-mêmes situés dans des catégories spécifiques. Le formulaire contenant toutes les informations nécessaires pour l’entreprise, cela éviterait le souci des informations inutiles. Dès lors que l’utilisateur aurait fini de remplir le formulaire et le validerait, il serait redirigé vers une page lui indiquant que sa demandé a bien été envoyé et qu’il sera contacté par l’entreprise dans les plus brefs délais ou au contraire, une erreur est survenue et qu’il serait invité à recommencer la procédure ultérieurement. En parallèle, toutes les informations enregistrées par l’application lors de la création des demandes serait visible dans une partie du site réservée pour le département de la communication ce qui permettrait d’effectuer un suivi sur les différentes demandes.

### Les objectifs

L’objectif du stage est donc de réaliser une application web permettant de traiter les demandes de la part de prestataire externes au sein d’une page contenant un formulaire et de réaliser une page permettant le suivi de tous les projets enregistrés par l’application. En plus de la réalisation de l’application web, il est demandé d’effectuer des tests de celui-ci vis-à-vis de la sécurité de l’application mais il est également demandé de mettre en place le projet sur un serveur web distant. Il est également important que les sites puissent être responsif, c’est-à-dire adaptées pour l’utilisation mobile (tablette, smartphone, …)

L’application web doit être une application écrite avec le langage de programmation orienté objet PHP qui doit pouvoir être liée avec une base de données vis-à-vis de l’enregistrement des différentes demandes.

Mon tuteur en entreprise M. Pierre George ainsi que l’assistante de direction au sein du département de la communication de l’entreprise et réceptionniste du projet Mme Françoise Philibert sont les personnes qui m’annoncent les objectifs précis concernant les spécificités du projet et qui ont effectuent un suivi de près de celui-ci.

## **Technologies utilisées**

Comme il a été dit précédemment, j’ai effectué tout mon travail sur un poste ne contenant aucune application initialement installée. J’étais donc libre d’installer n’importe quelle application de développement pour pouvoir réaliser au mieux le travail qui m’a été demandé.



Beaucoup de ces outils sont des outils qui ont été utilisés lors de l’apprentissage à l’IUT. Parmi ces outils, on retrouve :

* Eclipse : un environnement de développement permettant l’écriture du code PHP de manière rapide et efficace.
* Xampp : un ensemble de logiciels permettant la mise en place d’un serveur web.
* Git : un logiciel permettant d’effectuer du versionnage d’application liée au service d’hébergement github.
* Composer : un gestionnaire de bibliothèques PHP. C’est avec composer qu’il a donc été possible d’installer des bibliothèques externes tels que :
* Eloquent : micro-Framework issu du framework PHP Laravel utilisé pour interagir avec la base de données avec l’utilisation d’un modèle d’architecture ActiveRecord.
* Slim (v2) : micro-Framework utilisé en tant que Dispatcher ou Routeur d’URLs au sein du projet. Concrètement, il permet de lier certaines routes à des méthodes de classe pour la génération personnalisée de code HTML.
* PHPMailer : librairie alternative à la fonction d’envoi de mail de base de PHP.
* Password\_compat : libraire permettant la création de mot de passe cryptés en PHP.



Il est également important de mentionner l’utilisation du Framework CSS Materialize pour la mise en place de l’aspect visuel du site pour la partie réservée au personnes issus du département de la communication.



En plus de tous ces logiciels déjà-vu lors de l’apprentissage à l’IUT, deux nouvelles applications ont été découvertes lors du stage :

* PuTTY : logiciel permettant d’accéder au terminal d’un serveur distant
* WinSCP : logiciel permettant d’accéder aux fichiers d’un serveur distant.

Ces nouveaux logiciels m’ont été utiles lors de la mise en place de l’application web sur le serveur distant.

## **Réalisation du projet**

Lors de la première réunion avec le tuteur de stage en entreprise, il a été convenu que l’évolution et le suivi du projet de stage sera effectué avec des sprints review, c’est-à-dire des revues de projet effectuées selon un certain un temps de la même manière que la méthode de gestion de projet : la méthode agile (vu durant l’enseignement à l’IUT). Il s’agit donc de se fixer des objectifs sur une certaine période (dans le cadre du stage sur trois semaines en moyenne) avec une revue à la fin de la période avec le tuteur.

Chaque période est donc divisé en « sprint ». Sur l’intégralité des dix semaines de stage, quatre sprints ont été effectués avec, pour chaque sprint, des objectifs définis par le tuteur mais aussi le project owner (le réceptionniste du projet) qui est, pour rappel, l’assistante de direction au sein du département de la communication de l’entreprise.

### Sprint 0 – Introduction au projet initial

Lors de l’arrivé en entreprise c’est-à-dire le 9 avril, aucune consigne n’a été apportée par la personne concernée par l’application web c’est-à-dire le project owner puisque toutes les modalités du projet devaient m’être présentés lors d’une réunion qui devait se dérouler le jeudi 12 avril. Le sprint 0 consiste donc à un sprint d’introduction de quatre jours en attendant les premières fonctionnalités annoncées lors de la réunion. Ce sprint est composé en deux parties distinctes :

* Mise en place d’un environnement de développement
* Réalisation du squelette du projet ainsi que des premières fonctionnalités en prévision de la réunion du 12 avril.

La mise en place d’un environnement de développement consiste en l’installation de toutes les applications nécessaires pour la réalisation de mon travail. Chaque outils installé (vu au point II. 2)) sur le poste de travail a été choisis de manière efficace puisqu’il s’agit d’outils qui correspondent parfaitement à un développement d’applications web. De plus, il s’agit d’outils déjà utilisés durant les cours de développement web à l’IUT, la manière dont il faut utiliser ces outils est donc déjà connu et ne nécessite pas de rechercher des outils nouveaux.

Premier problème : Par soucis de sécurité d’entreprise au niveau du réseau informatique, l’entreprise utilise un Proxy, ce qui a pour effet d’empêcher l’accès à des intrus au sein du réseau, le terme intrus désignant les personnes malintentionnées voulant accéder aux données de l’entreprise ou encore la possibilité la réception de données spécifiques venant de l’extérieur. De ce fait, le téléchargement d’application est limité et une des applications ne peut pas être téléchargée : composer.

Solution : Ayant eu un entretien avec mon tuteur qui m’a expliqué qu’il était très compliqué de modifier les protocoles de sécurité, ma seule option est de télécharger composer en dehors du réseau de l’entreprise et de le lier avec le projet en l’intégrant avec git.

La réalisation du squelette permet de poser les bases du projet en intégrant tout les dossiers et fichiers nécessaires pour une bonne structuration et fonctionnement de celui-ci. Initialement, la structure se fait manuellement mais il est également possible de structurer son projet avec des outils tels que les Framework PHP.

Réflexion : Dois-je utiliser un Framework PHP (laravel en particulier) pour pouvoir facilement structurer mon projet et bénéficier de ses fonctionnalités ?

Conclusion : Il est inutile d’utiliser un Framework PHP pour un petit projet comme celui-ci étant donné que la compréhension du Framework prendrait du temps et toutes les fonctionnalités ne sont pas utiles.

Les premières fonctionnalités ont été données par mon tuteur de stage en entreprise. L’objectif est de développer un début d’application web pouvant répondre aux besoins du département de la communication c’est-à-dire une application permettant de traiter l’authentification à une page à l’aide d’un identifiant et d’un mot de passe pouvant être créé sur la page. Il s’agit de donner un accès à un utilisateur voulant se connecter à l’application plusieurs possibilités :

* En tant qu'utilisateur non identifié, je suis redirigé vers une page de login si j'essaie d'accéder au site.
* En tant qu'utilisateur en possession d'un compte (login (email) / mot de passe), je peux accéder au site.
* En tant qu'utilisateur non identifié, je peux créer un compte : login (email), mot de passe, nom de l'organisme, nom / prénom contact, adresse, téléphone).
* En tant qu'utilisateur ayant demandé un compte utilisateur, je reçois un mail pour valider mon compte.
* En tant qu'utilisateur identifié, je peux : modifier mes informations, changer mon mot de passe, supprimer complètement mon compte.

Ces fonctionnalités ont été entièrement implémentées et la fin du sprint 0 est marqué par la réunion du 12 avril.

### Sprint 1 – Développement de l’application

Après une réunion d’une demi-heure, le project owner et assistante de direction au sein du département de la communication de l’entreprise Françoise Philibert m’a annoncé toutes les attentes concernant le projet : réaliser une application web contenant deux parties distinctes :

* Une partie accessible pout les utilisateurs externes dit « FrontOffice ». Il s’agit d’une page contenant un formulaire avec toutes les informations listées nécessaires pour le département de la communication, ainsi qu’une possibilité de joindre des fichiers au formulaire qui respecte la charte graphique du site déjà existant. Cette partie doit également comporter une page qui indique, après complétion du formulaire, que la demande a été envoyé et que l’entreprise prendra contact avec l’organisme dans les plus brefs délais ainsi qu’une page spécifique en cas d’erreur.
* Une partie accessible uniquement pour les employés de Demathieu Bard dit « BackOffice ». Il s’agit d’une page contenant un système d’authentification, une page permettant de répertorier tous les formulaires ainsi qu’une page permettant d’accéder aux détails d’un formulaire. Le BackOffice ne nécessite pas une charte graphique spécifique.

Il s’agit donc de reprendre le squelette ainsi que certaines fonctionnalités du sprint 0 tout en laissant de côté ce qui a été réalisé pour pouvoir le réutiliser plus tard si besoin.

Avant de commencer à développer, un modèle de base de données est nécessaire pour pouvoir stocker toutes les données du formulaire.



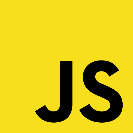
**Première version du schéma de la base de données**

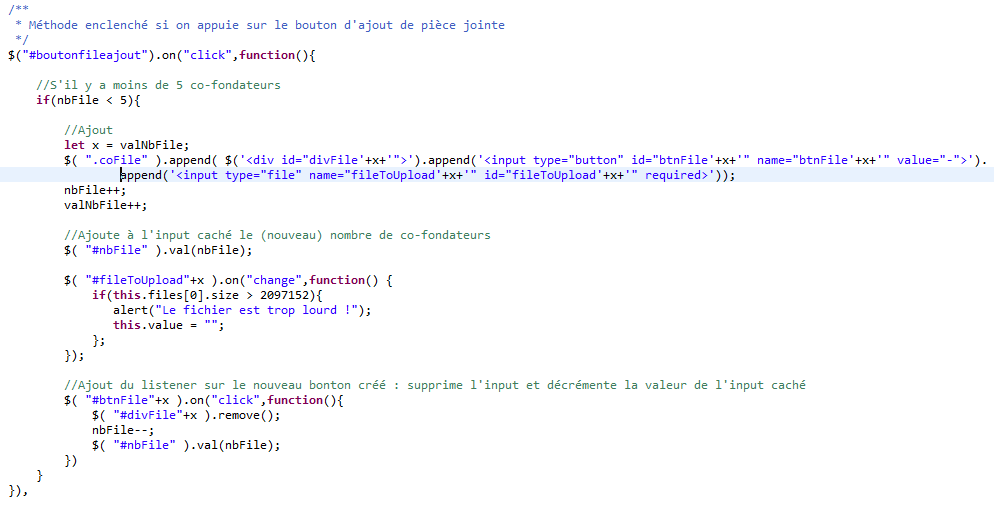
Tous les éléments nécessaires inscrits dans le schéma de la base de données (qui ne sont pas des clefs étrangères c’est-à-dire des éléments liés à une autre table) correspondent à un champ dans le formulaire, regroupés dans différentes tables exactement comme dans des groupes de champs dans le formulaire.

Le début de la réalisation de l’application web a été avec la réalisation du FrontOffice.

Réflexion : Comment l’utilisateur accède FrontOffice et comment un employé de l’entreprise accède au BackOffice ?

Conclusion : Si on imagine une adresse déjà existante de l’entreprise avec comme adresse [**https://www.sitedb.com**](https://www.sitedb.com)**,** on pourrait imaginer une adresse pour le FrontOffice [**https://www.sitedb.com/partenariat**](https://www.sitedb.com/partenariat)et pour le BackOffice (la deuxième adresse étant n’étant pas accessible à partir d’une redirection) [**https://www.sitedb.com/partenariat/admin**](https://www.sitedb.com/partenariat/admin).

La réalisation du FrontOffice consiste donc en la création d’une page de formulaire couplée avec du JavaScript lié à l’ajout et la modification dynamique des fichiers de valorisation de projet dans le formulaire



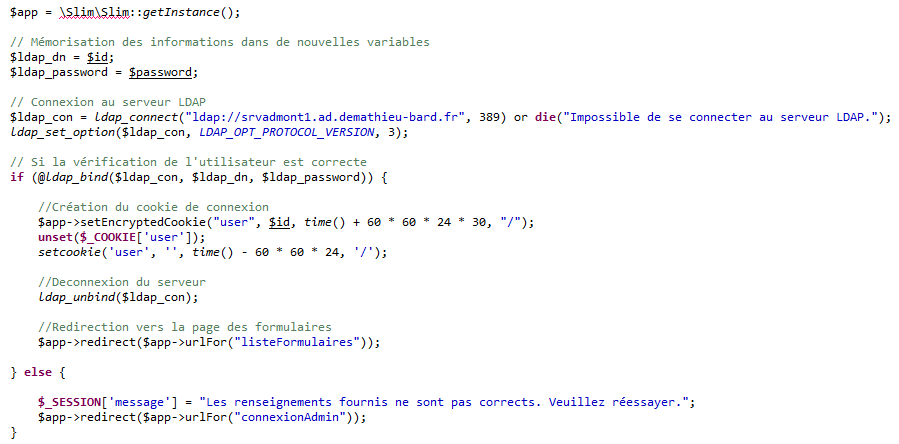
**Code JavaScript lié à l’ajout de fichier dans un formulaire**

En plus de l’ajout de fichier, la réflexion sur l’envoi de mail est nécessaire pour pouvoir informer le département de la communication lors de la réception de mail. Deux possibilités sont possibles :

* Utiliser la fonction de mail intégrée à PHP, cela nécessite néanmoins la configuration du serveur web.
* Importer une libraire liée à l’envoi de mail, les configurations se font dans le code PHP de l’application.

Solution : Utiliser la seconde option est plus intéressante puisqu’il ne sera pas nécessaire de configurer le serveur qui accueillera l’application. La fonction envoi un mail pour chaque adresse inscrite dans une liste statique

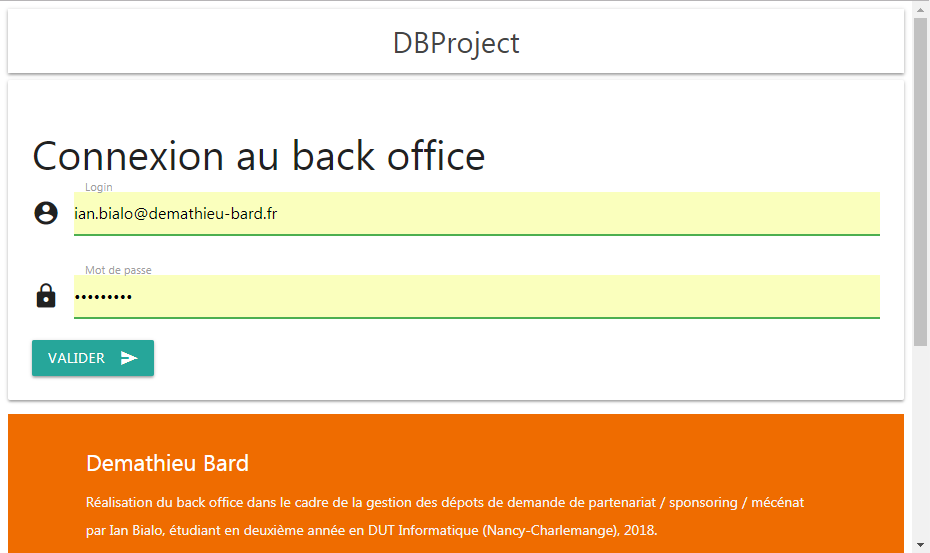
Lors du début de la réalisation de la partie BackOffice, il était nécessaire de se demander comment les utilisateurs c’est-à-dire les employés de Demathieu Bard et plus particulièrement les personnes issues du département de la communication allaient se connecter à cette partie du site. Après entretien avec mont tuteur, une solution m’a été imposée : la connexion LDAP Active Directory. Il s’agit concrètement d’un système de connexion à partir d’un annuaire et dans le cas présent l’annuaire de l’entreprise Demathieu Bard.



Code PHP lié à la connexion LDAP dans la partie BackOffice

L’avantage de ce type de connexion est qu’il n’est pas nécessaire de se créer un compte spécifique au BackOffice, il suffit de se connecter avec les identifiants de l’entreprise sans pour autant que l’application n’y ait accès : l’application se contente juste de demander à l’annuaire si les informations fournies sont correctes. Ne pas devoir créer de compte signifie également qu’il n’est pas nécessaire de créer de table dans la base de données et par conséquent le volume de données est réduit.

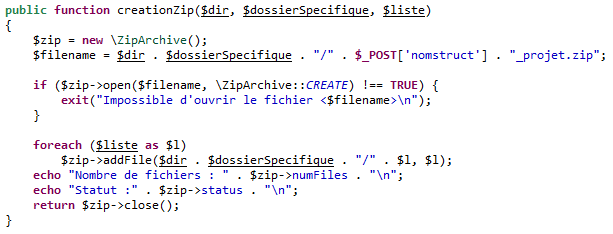
La partie BackOffice, contrairement au FrontOffice, intègre le Framework CSS Materialize permettant une mise en place des éléments sur la page. Il s’agit d’un Framework qui j’ai déjà utilisé auparavant et que je décide de réutiliser pour la partie BackOffice.



**Exemple d’utilisation de Materialize – Page de connexion au BackOffice**

La dernière difficulté dans le sprint 1 a été de comprendre le fonctionnement d’upload (envoi) de fichier au sein de l’application et de pouvoir les télécharger plus tard.

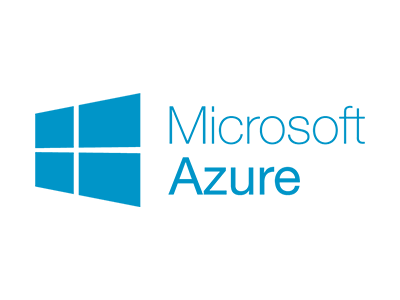
Une fonctionnalité supplémentaire qui n’a pas été mentionné que j’ai ajouté est de pouvoir télécharger tous les fichiers d’un formulaire directement compressé dans une archive au format .zip ce qui évite de devoir télécharger un à un tous les fichiers d’un projet.



**Code PHP lié à la création d’une archive compressé au format zip**

La majeure partie du sprint 1 a donc été l’écriture du code HTML ainsi que du code PHP et JavaScript permettant de modifier dynamiquement les pages web. La recherche de nouvelles fonctionnalités a été également un gros point permettant l’implémentation de nouvelles fonctionnalités (envoi de mail, connexion LDAP, création d’archive zip, …). Les entrevues avec mon tuteur ainsi que le project owner ont permis de savoir si le projet respecte les bonnes consignes et connaître les nouvelles fonctionnalités à implémenter dans la partie BackOffice. Le sprint a pris fin lorsque les fonctionnalités demandées ont été finalisées.

### Sprint 2 – Mise en place sur le serveur distant

La majeure partie du sprint 2 a été dédié au déploiement de l’application sur une copie du serveur original situé sur un serveur distant. L’entreprise Demathieu Bard possède des contrats chez Microsoft leur permettant d’avoir accès à un ensemble de service cloud : Azure. Parmi ces services, Azure propose son service d’hébergement de serveur à distance pour les entreprises : Microsoft se charge du maintien et de l’hébergement des serveurs à distance et Demathieu Bard peut accéder à ces serveurs et les contrôler au travers d’un tableau de bord. Il est, à partir de ce tableau de bord, facile d’effectuer des copies de serveurs ce qui a été fait dans mon cas pour que je puisse mettre en place l’application web sur un serveur distant sans qu’il y ait des risques vis-à-vis du site déjà existant.

C’est à partir du sprint 2 que j’ai commencé à utiliser les outils de gestion de serveur à distance : WinSCP et PuTTy. Après un entretien avec mon tuteur, il a été convenu que durant ce sprint que je puisse arriver à installer mon projet parmi les fichiers de la copie du site existant en y intégrant tout ce qui était nécessaire (script de déploiement d’application, création de la base de données, …).

Le serveur web distant tourne sous bitnami : il s’agit d’une solution pour les entreprises pour l’installation d’application sur les serveur web. L’application principale installée sur le serveur bitnami est Drupal qui est un CMS, c’est-à-dire un système de gestion de contenu libre. Drupal, de la même manière que des applications comme Wordpress de faciliter la gestion et la création de contenu web.

Problème : La copie du serveur web distant s’avère utiliser une version antérieure de PHP (5.4) à la version de développement de l’application (7.2.3). Après vérification avec une version alternative de xampp contenant PHP (5.4), certaines libraires importées avec composer ne sont plus compatibles. Comment résoudre le problème de compatibilité ?

Solution : Il s’agit simplement de regénérer les bonnes versions des libraires de manière à les faire fonctionner sous PHP 5.4.

Le serveur web distant étant une machine drupal bitnami, les dispositions de fichiers diffèrent un peu par rapport à un serveur web classique. Lors du déploiement de l’application, le tuteur m’a indiqué qu’il fallait créer un dossier nommé « partenariat » dans le dossier contenant tout le projet du site web déjà existant et d’y intégrer le projet. De ce fait, il serait facile d’accéder au projet par l’URL <https://url_de_la_machine.com/partenariat> (url\_de\_la\_machine n’étant pas le vrai URL).

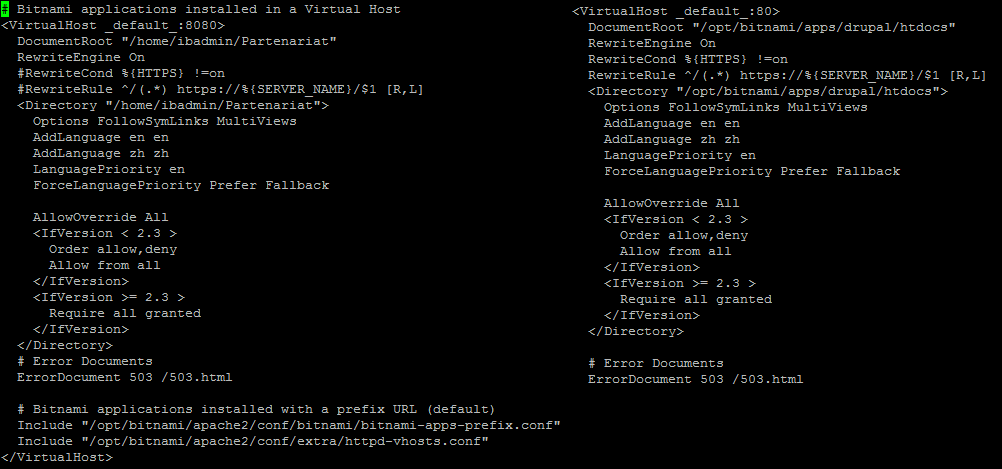
Problème : Le projet étant intégré avec Drupal, le système de « route » d’URL devient dépendant de celui-ci uniquement. Par conséquent, le micro-Framework Slim qui permet la création de routes au sein du projet ne peut plus fonctionner. Il s’agit donc de trouver une solution pour indiquer à Drupal que le contenu du dosser « partenariat » doit être traité de manière différente vis-à-vis des autres dossiers du projet.

Proposition : à l’aide du fichier .htaccess situé à la racine du projet Drupal, j’indique à Drupal que le dossier est indépendant. Néanmoins, cette option ne fonctionne pas.

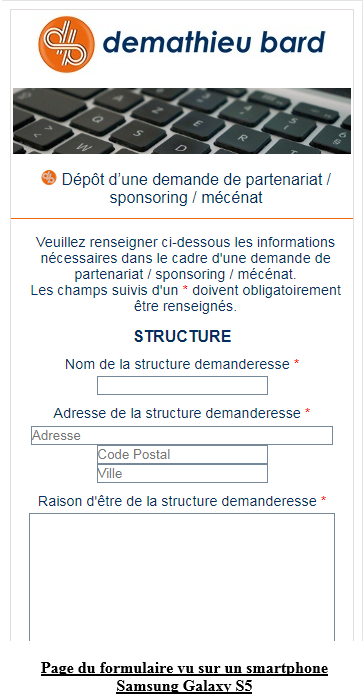
Solution : Dans les fichiers de configuration du serveur web, par le procédé des Virtual hosts, on indique deux types de chemin possible selon l’utilisation des ports :

* <https://url_de_la_machine.com/> : URL pour accéder à la page principale de l’application web déjà existante. (le port utilisé est donc par défaut le port 80 redirigé vers le 443 pour une connexion sécurisée, d’où le « s » après « http ».
* <http://url_de_la_machine.com:8080/> URL pour accéder à la page de partenariat

Le dossier contenant le projet de partenariat peut donc être placé où bon me semble étant donné que je peux préciser dans les fichiers de configuration du serveur web où est placé le dossier du projet (dans le cas présent le projet est placé dans le dossier /home/ibadmin/Partenariat visible dans la colonne de gauche, à la ligne 2 ci-dessous. Cependant, cette solution n’est que provisoire puisqu’il n’est pas correct d’utiliser ce type de procédé pour accéder à une page spécifique.



**Fichier de configuration de la copie du serveur web distant**

En plus de la mise en ligne du projet, une entrevue avec le product owner m’a permis d’implémenter une nouvelle fonctionnalité dans la partie du BackOffice de l’application web : un module de suivi. En plus de pouvoir avoir un accès à tous les projets de demande avec les détails des champs ainsi que les fichiers enregistrés, il est également nécessaire d’avoir un module de suivi permettant ainsi aux personnes issues du département de la communication de connaître l’avancement du traitement du projet. Cet ajout nécessite également l’ajout d’une nouvelle table dans la base de données liée au suivi.

Le product owner m’a également suggéré lors de l’entrevue de m’inspirer du site existant pour pouvoir réaliser l’habillage du FrontOffice. De ce fait, selon les pages <https://www.demathieu-bard.fr/crédits> et <https://www.demathieu-bard.fr/nous-contacter>, j’ai réalisé l’habillage correspondant à l’habillages des autres pages déjà existant. Etant donné que les autres pages ont réalisé avec Drupal contrairement à la page du formulaire, l’habillage ne peut pas entièrement être le même. Il respecte cependant la responsivité, c’est-à-dire le fait qu’il soit adapté également aux tablettes ou encore aux smartphones.

Les nouvelles fonctionnalités implémentées et le projet intégré sur le serveur distant un nouveau souci fait son apparition : la connexion LDAP ne fonctionne plus.

Problème : Le serveur web étant situé en dehors du réseau de l’entreprise, il n’est plus possible pour l’application situé sur ce serveur d’accéder à l’annuaire interne de l’entreprise. Comment effectuer de ce fait une connexion au BackOffice ?

Proposition : Je récupère toutes les données de l’annuaire de l’entreprise et je les intègre dans la base de données du projet. Cependant, avec les départs et les arrivées de nouvelles personnes dans l’entreprise, la table contenant les données utilisateurs risque d’être vitre obsolète.

Solution : Après entretien avec mon tuteur, il s’avère que je n’ai plus le choix : je suis obligé d’effectuer un système de connexion interne. Cela n’est pas une mauvaise alternative puisque peu de personnes utiliseront l’application et de ce fait les données dans la base de données concernant les utilisateurs ne seront pas encombrantes.

La fin du sprint 2 a été marqué par l’entretien avec le tuteur concernant la connexion LDAP.

### Sprint 3 – Finalisation du projet

L’intégralité du sprint 3 a été consacré à la finalisation du projet ainsi qu’à la correction de certaines erreurs et la documentation. Une entrevue avec le project owner a permis de définir les nouvelles fonctionnalités à implémenter vis-à-vis de l’ensemble du BackOffice.

Le nouveau système d’authentification a sollicité un menu de gestion de comptes ainsi qu’un menu de création de compte. Cela engendre, en plus des demandes du product owner, un système de droits pour les utilisateurs. Le BackOffice différencie deux types de comptes :

* Les comptes administrateurs
* Les comptes dit normaux.

Chacun de ces comptes ne possèdent pas les mêmes privilèges puisque les comptes administrateurs possèdent plus de droits que les comptes dit normaux. Il existe également un troisième compte dit « super administrateur » qui est un compte administrateur ordinaire mais qui ne peut pas être modifié par les autres comptes administrateurs : il s’agit de l’unique compte qui est créé initialement et qui est présent en cas d’inaccessibilité pour un compte administrateur dans le BackOffice.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Normal | Administrateur | Super administrateur |
| Consulter les projets | ✖ | ✖ | ✖ |
| Effectuer un suivi sur un projet | ✖ | ✖ | ✖ |
| Modifier ou supprimer un compte normal ou administrateur |  | ✖ | ✖ |
| Modifier un compte super administrateur |  |  | ✖ |

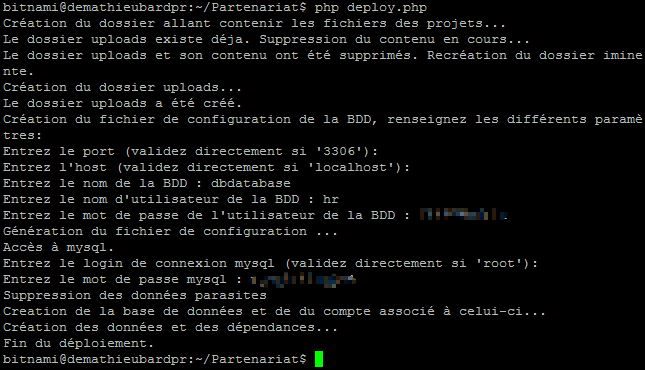
Tableau récapitulatif des droits selon le type de compte de l’application

Il est à noter qu’aucun compte ne peut se supprimer lui-même, il est nécessaire d’utiliser un autre compte pour cela (cela ne prend pas en compte le cas du super administrateur). Dès lors que l’utilisateur est supprimé, il ne peut plus accéder aux pages réservées aux personnes connectées et est redirigé vers la page de connexion au BackOffice. L’intégralité du système de droit a été pensé par soucis de sécurité étant donné que tout le mode peut accéder à la partie BackOffice de l’application. La mise en place de comptes nécessite la création d’une nouvelle table dans la base de données nommé « user ».

Problème : Lors de la modification du système d’authentification, il a été constaté l’absence de la librairie de création de mot de passe hashé (c’est-à-dire les mots de passe cryptés) ainsi que la lecture des mots de passe hashés étant donné qu’elle fait apparition à la version 5.5 de PHP et que le projet a été transformé pour qu’il soit adapté pour la version 5.4.

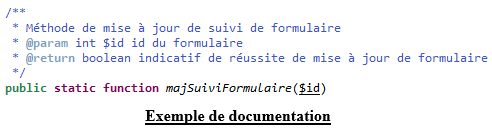
Solution : Une libraire externe permet pour les utilisateurs utilisant les version 5.3 et 5.4 de PHP d’implémenter un système d’authentification similaire à celui de la version 5.5.

En plus de la mise en place des nouvelles fonctionnalités et de la correction d’erreurs, la mise en place d’un script de déploiement et des indications d’installation ont été nécessaire pour permettre une installation facile et rapide de l’application. Le script de déploiement doit être lancé après avoir installé une copie du projet dans le dossier allant être lu par le serveur web. Il permet la vérification de dossier pouvant potentiellement exister comme le dossier allant contenir tous les fichiers envoyés par l’application lors d’un remplissage de formulaire par exemple en le supprimant complètement et en le recréant. Le script permet également de (re)créer une base de données liée au projet avec un fichier de configuration dédié au bon lien entre l’application et la base de données.



**Exécution complète du script de déploiement sur le serveur distant**

Il est à noter que la seule donnée enregistrée dans la base de données avec le script de déploiement est le compte super administrateur permettant ainsi une connexion au BackOffice.

En plus de la réalisation du rapport de stage en fin de sprint 3, l’écriture du code au sein du code est nécessaire pour donner le plus d’indications possible dans le cas où le projet devrait être amené à être modifié.

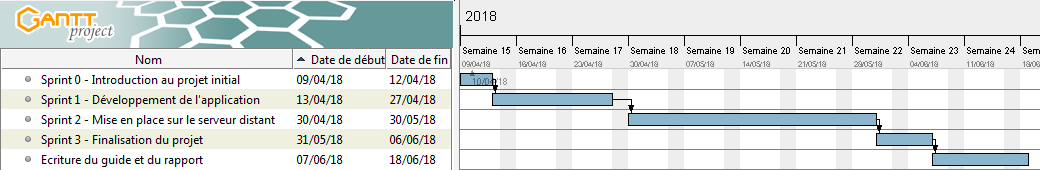
L’écriture d’un guide d’un cinquantaine de pages a également été réalisé dans le but d’expliquer pour les personnes voulant modifier l’application mais aussi pour les personnes voulant l’utiliser comment elle fonctionne dans son intégralité (que ce soit techniquement ou visuellement).

L’écriture du guide et du rapport de stage a mis fin au sprint 3 et par conséquent le développement de l’application. Aucune modification de code ainsi que d’ajout de documentation au sein de celui-ci n’ont été effectué après la fin du sprint.



**Page de couverture et exemple de page d’explication du FrontOffice du guide d’utilisation**

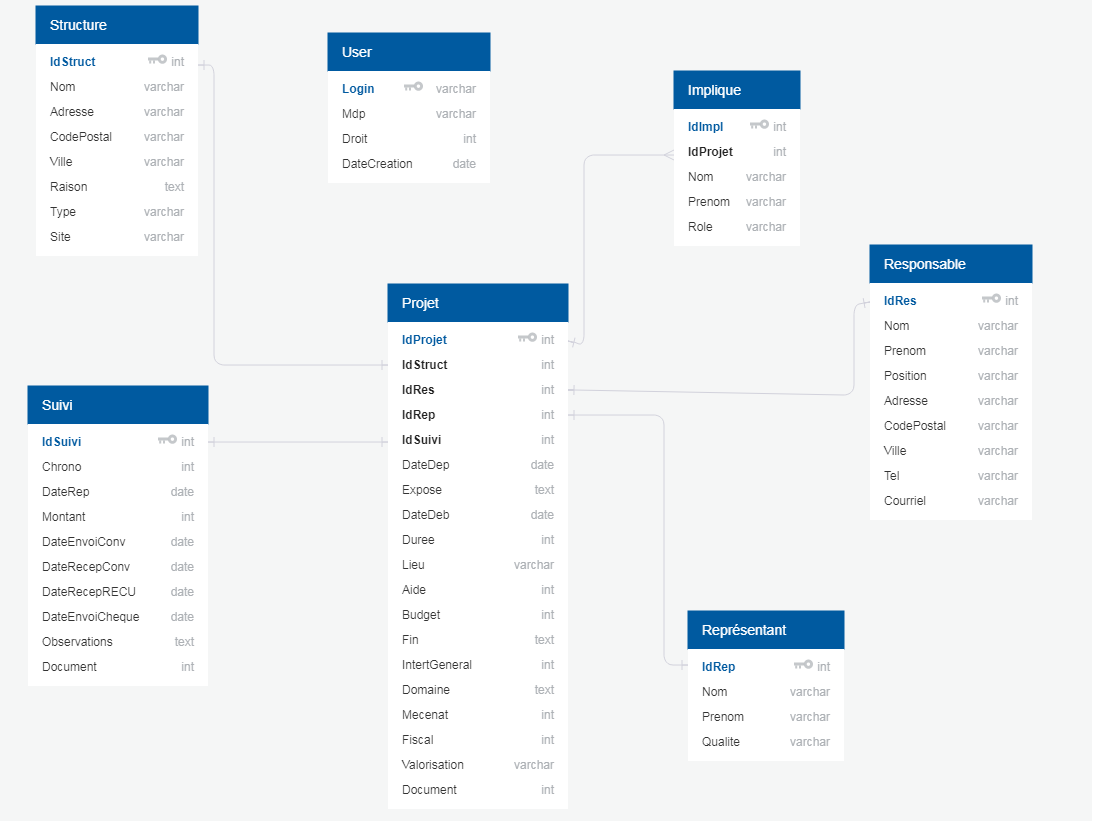
L’intégralité du stage a donc été scindé en différente parties selon les différents sprints avec, à chaque fois, des réalisation différentes (développement, intégration sur serveur, …). L’utilisation de la méthode agile a permis d’effectuer un suivi intéressant sur le projet et a permis de fixer de nouveaux objectifs en même temps que le projet avançait.



**Diagramme de gant liée à l’évolution de la réalisation des tâches durant le stage**

## **Structure et fonctionnement final du projet**

L’ensemble des éléments du projets ont évolués au fil du développement de l’application que ce soit la structure mais aussi la structure de la base de données.

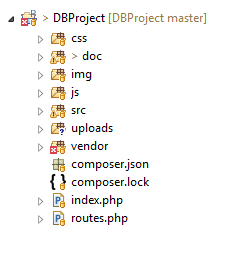


**Version finale de la base de données de l’application**

### Structure du projet

#### Structure à la racine

La structure de l’application web a été pensée de manière à suivre une l’architecture logicielle MVC (Modèle-Vue-Contrôleur). On retrouve tous les fichiers nécessaires au bon développement de l’application dans les dossiers dédiés.



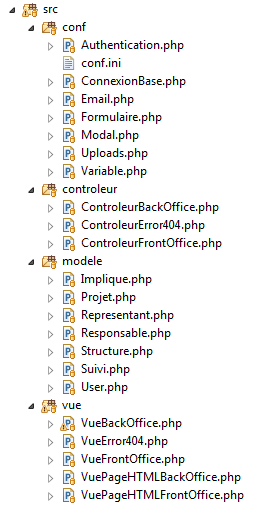
**Structure générale de l’application web**

A la racine du projet on retrouve :

* Le dossier css comprenant toutes les feuilles de style en cascade (plus communément appelés fichiers CSS) utilisés pour la mise en place des éléments sur toutes les pages de l’application.
* Le dossier doc comprenant toutes la documentation liée à l’application, dont le guide d’utilisation, le rapport de stage mais aussi les différents schémas d’utilisation.
* Le dossier img comprenant toutes les fichiers au format .jpeg et .png utilisées sur les différentes pages.
* Le dossier js comprenant tous les fichiers JavaScript utilisés en complément de toutes les fonctionnalités qu’apporte PHP utilisées sur les pages web (côté client).
* Le dossier uploads qui contient tous les fichiers uploadés par les utilisateurs de l’application (il est donc initialement vide).
* Le dossier vendor comprenant tous les outils supplémentaires au bon développement de l’application comme les bibliothèques générées avec composer.
* Le fichier index.php qui permet d’instancier tous les éléments de l’application.
* Le fichier routes.php qui définit toutes les routes de l’application (ce fichier utilise le micro-Framework Slim).
* Tous les fichiers de configuration dont le fichier composer.json.

Il est à noter que dans le cas où une version d’un module ne serait plus utilisable à cause d’une mauvaise version ou d’une mauvaise interaction des différent modules (ce qui n’est pas censé arriver), il suffit juste de modifier les paramètres du fichier composer.json et de regénérer les modules à l’aide de composer.

#### Structure au dossier source



**Structure du dossier src de l’application web**

Au niveau du dossier de source de l’application web, on retrouve les éléments regroupés selon l’architecture MVC vu précédemment. Etant donné que l’application web est divisée en une partie liée au FrontOffice et une partie liée au BackOffice, il est nécessaire de faire une distinction de ceux-ci dans les fichiers de l’application. Il y a donc :

* Le dossier conf comprenant tous les fichiers PHP avec des méthodes statiques utilisées dans l’application selon les disponibilités. Ces fichiers sont :
* Authentication.php : fichier qui permet à l’utilisateur de se connecter, de créer un compte, de se déconnecter de la partie BackOffice de l’application.
* ConnexionBase.php : fichier qui permet d’établir la connexion à la base de données MySQL (ce fichier utilise le Framework Eloquent).
* Email.php : fichier permettant la gestion d’envoi de mails au travers de l’application.
* Formulaire.php : fichier permettant la gestion des formulaires lorsqu’un utilisateur décide de créer un nouveau projet au travers du formulaire du FrontOffice de l’application.
* Modal.php : fichier utilisé dans les vues pour éviter la réutilisation la réécriture de code inutile. Il permet la génération de boîte de confirmation.
* Uploads.php : fichier permettant la gestion d’enregistrement et de suppression de fichier au sein de l’application.
* Variable.php : fichier contenant toutes **les variables pratiques et modifiables** de l’application. Cela signifie que lors de la configuration de l’application sur un serveur web, ce fichier doit être manipulé pour pouvoir modifier le fonctionnement de l’application selon les nécessités.

En plus de ces fichiers, on retrouve un fichier de configuration conf.ini **inexistant** lors de la récupération du projet du dépôt GitHub. Il contient toutes les informations de connexion à la base de données et est utilisé dans le fichier ConnexionBase.php. Il est généré avec le fichier de déploiement de l’application.

* Le dossier controleur qui contient distinctivement les fichiers de contrôle de la partie FrontOffice et BackOffice de l’application mais aussi le fichier de contrôle pour gérer le cas des erreurs 404 (ressource demandée inexistante) venant de l’application web.
* Le dossier modele qui contient les modèles de l’application. Un modèle correspond à une table dans la base de données et ce sont ces modèles qui sont appelés dès lors que la manipulation des données est nécessaire. Ces fichiers utilisent Eloquent vu précédemment.
* Le dossier vue qui contient toutes les vues de l’application. En plus des vues liées au BackOffice, FrontOffice et la Vue liée à l’erreur 404, on retrouve les fichiers VuePageHTML qui correspondent à des fichiers ayant des méthodes statiques permettant de générer selon différents cas de figures les balises <head> ou encore <footer>.

### Fonctionnement final de l’application

L’application Web fonctionne pour chaque cas de figure de la même manière. On différencie l’exécution selon les types de requêtes effectuées c’est-à-dire les méthode GET et POST. Il est à noter que l’application différencie les URLs selon les différentes requêtes, ce qui signifie qu’une URL peut être identique du moment que les méthodes sont différentes.

#### Fonctionnement vis-à-vis d’une méthode GET

Lorsqu’une méthode GET est envoyée au serveur, il réagit de cette manière :

* Premièrement, il vérifie que l’URL est une URL qui est enregistrée. L’application accède au fichier index.php pour ensuite accéder au fichier routes.php et chercher la route qui lui a été indiquée.
* Si la route a été trouvée, il accède à la fonction anonyme liée à la route et l’exécute.
* Si la route n’a pas été trouvée, il accède à une route par défaut qui lance la fonction notFound de Slim et qui accède et exécute la fonction anonyme liée.
* Ensuite l’application exécute tout ce qui se trouve dans la fonction anonyme.
* Il va premièrement chercher la première partie du code HTML souhaitée selon les différents cas de figures.
* Il va ensuite accéder au controleur spécifique.
* Il va finir par chercher la dernière partie du code HTML souhaitée.
* Pendant l’appel du controleur spécifique, l’application prépare les différentes données qui seront réutilisées dans l’application. Dans le cas du BackOffice, l’application va également effectuer des tests de sécurité pour contrôler les personnes qui accèdent au controleur : cela est fait par exemple avec la vérification de la connexion de l’utilisateur à l’application. Si tout s’est bien déroulé, l’application accède à la vue qui lui est associée.
* Pendant l’appel de la vue, un code HTML est généré et retourné dans le controleur. Cette vue fait appel aux données de la base de données dans le cas du BackOffice. Dans tous les cas, les vues sont générées dynamiquement.
* Finalement, le controleur ayant reçu la vue il peut l’afficher pour constituer la partie centrale du code HTML.

De cette manière, le code HTML complet est généré par l’application et est renvoyé au client pour l’afficher dans son navigateur.

#### Fonctionnement vis-à-vis d’une méthode POST

Lorsqu’une méthode POST est envoyée au serveur, elle réagit d’une manière similaire lorsqu’une méthode GET est envoyée à la différence que l’application n’affiche rien et que aucune vue n’est sollicitée.

* Premièrement, il vérifie que l’URL est une URL qui est enregistrée. L’application accède au fichier index.php pour ensuite accéder au fichier routes.php et chercher la route qui lui a été indiquée.
* Si la route a été trouvée, il accède à la fonction anonyme liée à la route et l’exécute.
* Si la route n’a pas été trouvée, il accède à une route par défaut qui lance la fonction notFound de Slim et qui accède et exécute la fonction anonyme liée.
* Ensuite l’application exécute tout ce qui se trouve dans la fonction anonyme. Il exécute dans tous les cas un appel à une méthode du controleur spécifique.
* L’application exécute le code de la fonction appelée pour finalement faire appel à une méthode GET. Cela permet donc de rediriger l’application vers une méthode qui permettra d’effectuer un affichage pour informer l’utilisateur de la réussite ou non de l’application après l’action de la méthode POST.

#### Complément de fonctionnement

Il est important de rappeler que Slim permet de générer des URLs et d’effectuer des redirections au sein de l’application. Ses méthodes sont constamment utilisées dans l’application et permettent un accès facile aux URLs de l’application. Lors de l’exécution d’une méthode POST, il s’agit de la méthode redirect qui est appelée pour rediriger l’application vers une méthode GET.

# **Disposition du site et fonctionnalités associées**

L’ensemble des captures d’écrans associées aux éléments de cette partie sont visible dans l’annexe pour éviter de polluer le corps du mémoire avec un surplus de captures d’écran. Pour la suite, nous allons imaginer que l’application web est mis en place et qu’il est accessible à l’adresse [**https://www.sitedb.com/partenariat/**](https://www.sitedb.com/partenariat/). Bien évidemment ce chemin redirige vers une page inexistante puisqu’il s’agit d’une adresse imaginaire qui sert juste d’exemple et qui n’est en aucun cas utilisé par l’entreprise.

## **Le FrontOffice**

Le FrontOffice est, pour rappel, la partie accessible à toutes les personnes souhaitant remplir un formulaire de demande de partenariat, de sponsoring ou encore de mécénat. Il s’agit donc d’une partie publique qui est rattachée au site existant de l’entreprise accessible à l’adresse <https://www.sitedb.com/partenariat/>.

### Le formulaire

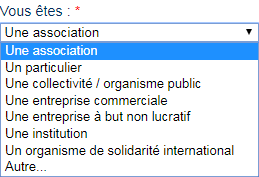
**(Voir la figure 1 dans l’annexe)**

La page du formulaire se distingue en deux parties : la partie supérieure qui, en plus du logo et d’une image permettant d’aérer la page, donne des indications pour les personnes souhaitant remplir le formulaire et la partie inférieure qui contient le formulaire.

Le formulaire est structuré selon trois catégories :

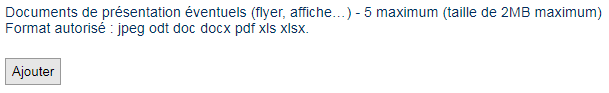
* Structure : coordonnées de la structure concernée par la demande. **(Voir la figure 2 dans l’annexe)**
* Personnel : coordonnées des personnes concernées par la demande. **(Voir la figure 3 dans l’annexe)**
* Projet : descriptif du projet pour la demande. **(Voir la figure 4 dans l’annexe)**

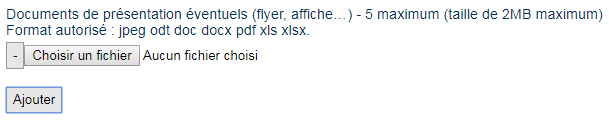
Chaque champ comprenant un petit texte décrivant ce qui doit être inséré. Dans le cas où un texte se finit par une étoile en rouge (\*), cela signifie que le champ doit être obligatoirement complété. On retrouve des champs pouvant être complété au clavier mais aussi des champs proposant des choix comme le champ associé au type de structure dans un formulaire par exemple.



**Champ associé au type de structure dans le formulaire**

Néanmoins il est possible de choisir « Autre » comme type de structure pour faire apparaître un nouveau champ obligatoire qui demande à l’utilisateur de préciser le type de structure.





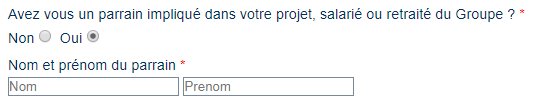
**Champ associé à l’ajout de fichier au projet**

Il est possible d’ajouter des fichiers au projet. Pour cela, il suffit d’appuyer sur le bouton « Ajouter » pour ensuite faire apparaître un champ obligatoire d’ajout de fichier. Il est possible d’ajouter des fichiers en respectant différentes conditions :

* Le nombre de fichier doit être au plus de cinq.
* La taille maximale de chaque fichier doit être de deux MB au plus.
* Le format doit être un des suivants : jpeg, odt, doc, docx, pdf, xls, xlsx.

Pour supprimer un fichier, il suffit d’appuyer sur le bouton « - » à côté du fichier en question. Ce procédé est identique pour l’ajout de co-financeur (dans la catégorie Projet).





**Champs associés à l’ajout d’un parrain au projet**

L’ajout de parrain diffère un peu. Pour ajouter un parrain, il suffit de mentionner le fait qu’un parrain est impliqué dans le projet en cochant l’option « Oui » et deux nouveaux champs obligatoires feront leurs apparitions. Pour faire disparaître ces champs, il suffit de cocher l’option « Non ».

Lorsque le formulaire est complété dans son intégralité, pour le valider il suffit d’appuyer sur le bouton « Valider et envoyer votre demande » situé en bas de page. Si aucune redirection n’a été faite, cela signifie qu’un champ n’a pas été renseigné et la page web vous redirigera vers le champ incomplet.

### L’après-formulaire

Dès lors que le formulaire a été complété et validé, l’application web se charge de récupérer toutes ces données et de les insérer dans la base de données mais aussi de stocker les fichiers envoyés par l’utilisateur :

* L’application récupère les informations concernant la structure, le responsable, la structure et les insères dans la base de données.
* L’application insère un nouveau suivi lié au projet et l’insère dans la base de données.
* L’application crée un nouveau projet, récupère les informations, le lie à la structure, au responsable et à la structure créée et insérée dans la base de données et l’insère à son tour dans la base de données.
* L’application crée les personnes impliquées (les co-financeurs et les parrains), les lient au projet et les insère dans la base de données.
* L’application récupère les fichiers envoyés par l’utilisateur, les analyse et les enregistre dans le dossier prévu à cet effet.
* L’application notifie les personnes chargées de traiter ces demandes au travers d’un mail.

**(Voir la figure 5 dans l’annexe)**

Si tout se déroule comme il a été mentionné ci-dessus, alors l’application redirige l’utilisateur vers une nouvelle page où un message explique que l’enregistrement a été correctement effectué et que l’entreprise se chargera de contacter au plus vite l’utilisateur.

**(Voir la figure 6 dans l’annexe)**

Dans le cas contraire, un message notifie l’utilisateur de l’échec avec la redirection de celui-ci vers une page dédiée.

## **Le BackOffice**

Le BackOffice est, comme il a été dit précédemment, la partie réservée au personnel de l’entreprise et plus particulièrement le personnel du département de la communication. Il s’agit donc d’une partie privée même s’il peut être accédé par n’importe quel individu mais il faudra néanmoins avoir un compte délivré par l’entreprise pour pouvoir s’y connecter. Contrairement au FrontOffice, le BackOffice adopte une charte graphique propre mis en place à l’aide du Framework CSS Materialize.

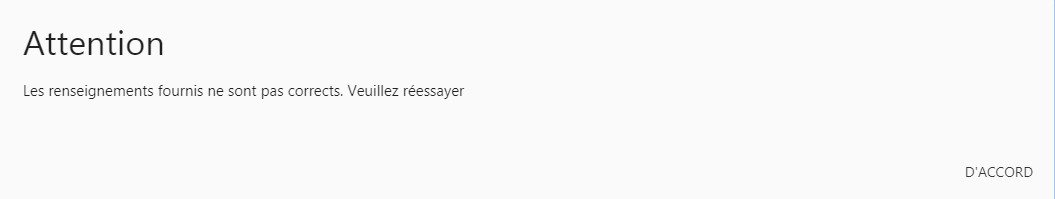
L’accès au BackOffice se fait simplement en ajoutant le chemin d’accès **/admin** après l’adresse principale de l’application : dans le cadre de ce rapport l’adresse est donc <https://www.sitedb.com/partenariat/admin>. Toutes les pages de la partie BackOffice auront donc cette adresse principale avec en complément un chemin d’accès en fin d’adresse. Par exemple, pour une page de recherche, l’adresse serait donc <https://www.sitedb.com/partenariat/admin/recherche>.

### Le menu de connexion

L’accès au menu de connexion s’effectue en accédant à l’adresse de la racine du BackOffice, c’est-à-dire <https://www.sitedb.com/partenariat/admin>.

**(Voir la figure 7 dans l’annexe)**

Lorsqu’un utilisateur accède au menu de connexion du BackOffice, il accède directement à une page l’invitant à rentrer ses identifiants à savoir son login (qui est donc une adresse mail) ainsi que son mot de passe. Il est donc nécessaire à l’utilisateur souhaitant accéder au BackOffice d’avoir un identifiant créé par un administrateur.



**Message d’erreur à la connexion au BackOffice**

Après avoir rempli les champs et validé, si les informations rentrées ne sont pas correctes, l’utilisateur reste sur la page de connexion et un message d’erreur apparaît à l’écran. Dans le cas contraire, il est redirigé vers la page contenant la liste des projets.

### La liste des projets

La page contenant la liste des projets est accessible à l’adresse <https://www.sitedb.com/partenariat/admin/formulaire>. Dans le cas où l’utilisateur n’est pas connecté au BackOffice, il sera automatiquement renvoyé vers la page de connexion.

L’utilisateur accède à cette page dès lors qu’il se connecte à l’application. Si l’utilisateur essaye d’accéder à la page de connexion alors qu’il est déjà connecté à l’application, il est automatiquement renvoyé vers cette page.

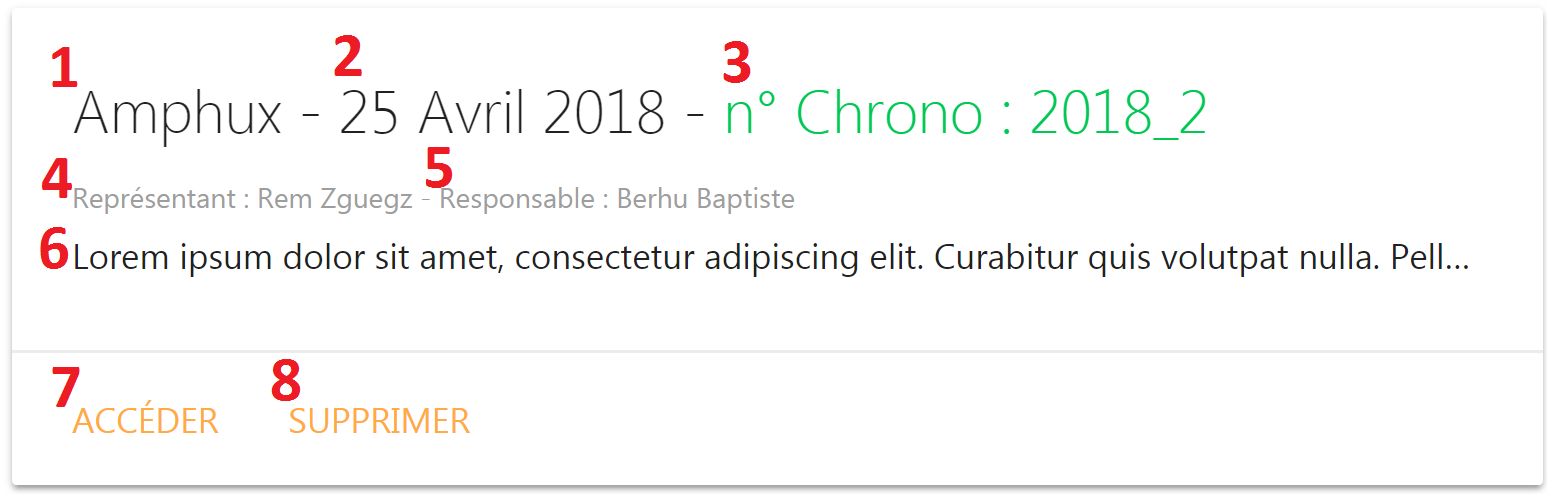
**(Voir la figure 8 dans l’annexe)**

La page de la liste des projets du BackOffice se distingue en trois parties : la partie du haut plus communément appelé « header », la partie centrale comportant des fonctionnalités permettant la recherche de projets et la partie du bas où l’on retrouve tous les projets qui ont été enregistrés dans l’application. Il est à noter que le « header » est la partie que l’on retrouve sur toutes les pages de BackOffice et son contenu (c’est-à-dire des fonctionnalités) varie selon si l’utilisateur est connecté ou non.

Dans la partie centrale on retrouve donc les fonctionnalités permettant la recherche de projets. Parmi ces fonctionnalités il en existe deux :

* Recherche par accès rapide : il permet de rechercher tous les projets existants dans une petite liste
* Recherche manuelle : il permet de rechercher un projet en rentrant le nom de la structure au clavier

Après avoir complété le champ et appuyé sur le bouton « Rechercher », l’utilisateur est redirigé vers la page de recherche de projet.



**Projet type affiché dans la liste des projets du BackOffice**

Chaque projet dans la liste des projets est affiché de la même manière avec les mêmes éléments. Vis-à-vis de la capture d’écran ci-dessus, on retrouve :

* 1 : Le nom de la structure
* 2 : La date de création de projet
* 3 : La numéro chrono complété par le biais de suivi
* 4 : Le nom et prénom du représentant de projet
* 5 : Le nom et prénom du responsable de projet
* 6 : Le descriptif de projet
* 7 : Le bouton d’accès au projet
* 8 : Le bouton de suppression de projet

Tous ces champs sont donc des champs qui ont été complétés par l’utilisateur du FrontOffice lors de la complétion du formulaire.

Le bouton « Accéder » (7) permet de rediriger l’utilisateur vers une page dédié au formulaire qu’il souhaite accéder, lui permettant d’avoir des détails des détails complets sur le projet.

Lorsque l’utilisateur souhaite supprimer un projet, il lui suffit d’appuyer sur le bouton « Supprimer » (8). Ceci fait, une boîte de dialogue apparaît à l’écran de l’utilisateur lui demandant s’il est sûr de vouloir supprimer son projet puisqu’il s’agit d’un acte irréversible. Il est important de rappeler que **supprimer un projet est un acte irréversible** puisqu’en aucun cas il sera possible de récupérer un projet supprimé quel que soit la manière envisagée.

### Les détails du projet

La page contenant les détails d’un projet est accessible à l’adresse <https://www.sitedb.com/partenariat/admin/formulaire/numero>, où numéro est l’identificateur de projet dans la base de données. Dans le cas où l’utilisateur n’est pas connecté au BackOffice, il sera automatiquement renvoyé vers la page d’erreur de page inexistante du FrontOffice. L’utilisateur accède à cette page dès lors qu’il sélectionne le lien « Accéder » d’un projet sur la page de liste des projets.

**(Voir la figure 9 dans l’annexe)**

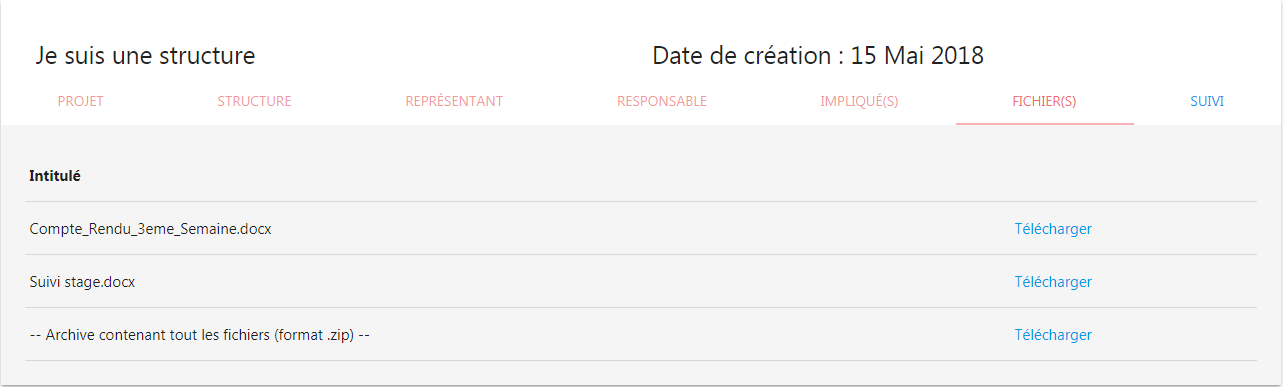
La page de détails d’un projet du BackOffice est la page qui permet de regrouper tous les éléments d’un projet dans des tableaux structurés et organisés selon des catégories spécifiques. N’importe quel utilisateur de l’application à le droit d’accès et de suivi de chacun des projets, qu’il soit utilisateur normal ou administrateur.

Comme il a été dit précédemment, chaque projet a ses données regroupées selon des catégories spécifiques dans des onglets : Projet, Structure, Représentant, Responsable et Impliqué(s) (ce le(s) co-fondateur(s) ainsi que le parrain (s’il y en a) en lien avec le projet).

Il est à noter que la plupart de ces catégories correspondent aux mêmes catégories que l’on retrouve sur la page du formulaire du BackOffice.

**(Voir la figure 10 à 14 dans l’annexe)**

Chacune des catégories regroupées dans des onglets permettent à l’utilisateur du BackOffice de se renseigner sur les différents éléments inscrits par les utilisateurs du FrontOffice lors de la création d’un projet. Chacune de ces catégories regroupes des données qui peuvent être modifiées selon les besoins de l’utilisateur du BackOffice. Pour cela, il suffit de modifier les champs souhaités et d’appuyer sur le bouton « Modifier » présent sur chaque onglet.

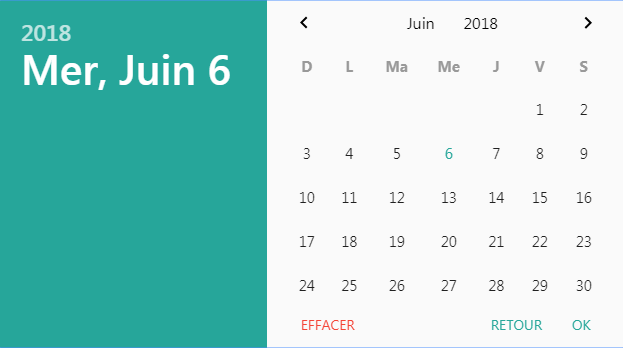


Fichiers liés au projet dans la page de détails d’un projet du BackOffice

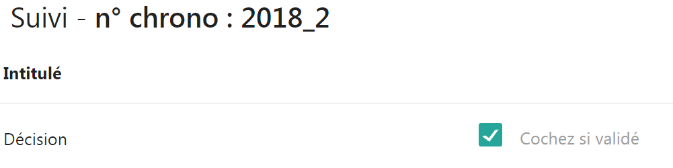
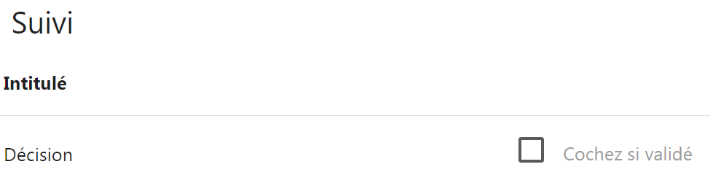
Lors de la création d’un projet dans la partie FrontOffice, il est possible de joindre d’éventuels documents permettant une valorisation supplémentaire d’un projet. Ces documents sont regroupés dans l’onglet « Fichier(s) de la page de détails d’un projet du BackOffice sous forme de liste. En plus des documents envoyés lors de la création d’un projet, on retrouve dans la liste de cet onglet une archive compressée au format zip contenant tous les fichiers envoyés lors de la création d’un projet. Tous ces documents sont téléchargeables en appuyant sur le lien « Télécharger » qui se situe sur la même ligne que le fichier à télécharger dans la liste des fichiers.

**(Voir la figure 15 dans l’annexe)**

En plus de tous les onglets regroupant toutes les données sur la page de détails de projet ainsi que les fichiers, il existe un dernier onglet qui correspond au suivi du projet. Le suivi de projet est unique, c’est-à-dire que chaque utilisateur a la possibilité de voir les mêmes données enregistrées dans cet onglet. L’onglet de suivi se distingue en deux parties : la partie de suivi de projet ainsi que la partie d’enregistrement de fichiers.

La partie de suivi est une fonctionnalité permettant le traitement facilité de dossier vis-à-vis des utilisateurs du BackOffice. Il s’agit de champs à compléter pouvant être modifié si nécessaire. Elle est composée de sept champs ainsi que d’une case à cocher. La plupart de ces champs permettent d’indiquer une date contrairement à d’autres où il s’agit simplement de renseignement à intégrer au clavier.

Initialement, tous les champs sont vides par défaut et les champs qui attendent une date indiquent avec une police grisée la date du jour. Lorsqu’un champ n’a jamais été modifié (donc qui est initialisé par défaut), le label explicatif de celui-ci indique d’une couleur rouge que la valeur est inchangée.



**Exemple de projet non-validé (à gauche) et validé (à droite) dans la partie de suivi de projet de la page de détails d’un projet du BackOffice**

L’unique case présente dans le suivi sert à préciser si le projet doit être pris en compte ou non. Lorsque celui-ci est coché, le projet passe en statut de « valide » et se voit être attribué par un numéro chrono. Le numéro chrono correspond à l’année de début de projet suivi de l’ordre dans l’année de création. Dans l’exemple ci-dessus, on comprend que le projet a été créé en 2018 et qu’il s’agit du deuxième projet dans cette année ayant un statut « valide ».

 En plus de manipuler des données pour le suivi d’un projet, il est également possible d’ajouter un ou plusieurs fichiers permettant ainsi de conserver des documents importants liés au projet dans l’application. Il n’existe pas de limite concernant l’ajout de fichiers et comme pour l’onglet « Fichier(s) » vu précédemment, il est également possible de télécharger les fichiers enregistrés. Tous les fichiers enregistrés sont téléchargeables en appuyant sur le lien « Télécharger » qui se situe sur la même ligne que le fichier à télécharger dans la liste des fichiers.Il est également possible de supprimer les fichiers enregistrés dans l’application. Pour cela, il suffit d’appuyer sur le bouton ayant comme symbole une corbeille situé à côté du fichier à supprimer. Il est important de souligner que **supprimer un fichier est un acte irréversible** puisqu’en aucun cas il sera possible de récupérer un fichier supprimé quel que soit la manière envisagée.

Dès lors que l’utilisateur a fini de modifier le suivi ou d’ajouter un ou plusieurs fichiers, il peut valider ces fichiers en appuyant sur le bouton « Valider » situé en dessous du lien d’ajout de fichier.

### La recherche de projet

La page contenant la recherche de projets est accessible à l’adresse <https://www.sitedb.com/partenariat/admin/formulaire/recherche/projet>, où projet est le contenu de la recherche de l’utilisateur. Dans le cas où l’utilisateur n’est pas connecté au BackOffice, il sera automatiquement renvoyé vers la page d’erreur de page inexistante du FrontOffice. L’utilisateur est automatiquement redirigé vers cette page dès lors qu’il effectue une recherche de projet depuis la page contenant la liste des projets.

Le contenu de la page de recherche dépend du contenu de la recherche de l’utilisateur. Trois cas de figures sont possibles :

* 1 : L’application ne trouve pas de projet lié avec le contenu de la recherche de l’utilisateur et l’informe au travers d’une page vide contenant seulement un message ainsi qu’un bouton permettant de revenir vers la page avec la liste des projets.
* 2 : L’application trouve plusieurs projets portant le nom du contenu de la recherche de l’utilisateur et affiche tous les projets dans la page de recherche. Cette page est similaire à la page listant tous les projets à la différence où il n’y est pas possible d’effectuer une recherche ou encore d’effectuer un tri. **(Voir la figure 16 dans l’annexe)**
* 3 : L’application ne trouve qu’un seul projet portant le nom du contenu de la recherche de l’utilisateur et redirige l’utilisateur vers la page détaillée de ce projet.

### La gestion de compte

La page contenant la gestion des comptes est accessible à l’adresse <https://www.sitedb.com/partenariat/admin/gestion/>. Dans le cas où l’utilisateur n’est pas connecté au BackOffice, il sera automatiquement renvoyé vers la page d’erreur de page inexistante du FrontOffice. L’utilisateur est redirigé vers cette page dès lors qu’il sélectionne l’option « gestionnaire de compte » dans la partie « header » depuis n’importe quel page du BackOffice (à condition qu’il y soit connecté).

La page de gestion des comptes diffère du type du compte de l’utilisateur connecté à l’application (compte administrateur ou compte normal).

#### Compte administrateur

**(Voir la figure 17 dans l’annexe)**

 Le compte administrateur ayant les droits de modifications sur tous les comptes enregistrés de l’application (sauf sur un compte super administrateur), l’accès à la page de gestion des comptes du BackOffice lui donne accès à un tableau de bord où sont renseignés tous les comptes. Tous les logins des comptes sont répertoriés dans une liste avec les droits associés. Les droits des comptes sont représentés par des pictogrammes qui sont situés à côtés de chacun des logins. Une légende située en haut de page permet de comprendre facilement chacun de ces pictogrammes.

**(Voir la figure 18 dans l’annexe)**

Lorsque l’utilisateur sélectionne un compte dans la liste des comptes enregistrés, des nouveaux champs apparaissent en dessous de celui-ci lui permettant d’effectuer une modification du compte : un champ de modification de droits et un champ de modification de mot de passe. Pour pouvoir modifier le mot de passe, il suffit de cocher la case demandant d’être cochée si l’utilisateur souhaite changer le mot de passe puis de rentrer le nouveau mot de passe dans les champs prévus à cet effet en dessous. Dès lors que l’utilisateur à fini de modifier un compte, il lui suffit d’appuyer sur le bouton « Modifier » situé en dessous des champs des mot de passes. Si l’utilisateur souhaite supprimer le compte, il lui suffit d’appuyer sur le bouton « Supprimer » situé à côté du bouton « Modifier » vu ci-dessus. Il est important de souligner que **supprimer un compte est un acte irréversible** puisqu’en aucun cas il sera possible de récupérer un compte supprimé quel que soit la manière envisagée.

 En plus de la liste des comptes enregistrés dans l’application, il est également possible d’ajouter un nouveau compte dans l’application. Ce bouton est situé en dessous de la liste des comptes enregistrés.

#### Compte normal

**(Voir la figure 19 dans l’annexe)**

Le compte normal n’ayant pas les droits de modifications sur tous les comptes enregistrés de l’application l’accès à la page de gestion des comptes du BackOffice lui donne accès à un tableau de bord où sont renseignés les informations de son compte seul. Il lui est ainsi impossible d’accéder à la liste complète des comptes enregistrés dans l’application.

On retrouve dans cette page deux types de champs : les champs non modifiables content toutes les informations de l’utilisateur, c’est-à-dire son login et son statut de compte, et les champs modifiables correspondant aux champs de modification de mot de passe.

### Création de compte

La page contenant la création de compte est accessible à l’adresse <https://www.sitedb.com/partenariat/admin/gestion/creation>. Dans le cas où l’utilisateur n’est pas connecté au BackOffice, il sera automatiquement renvoyé vers la page d’erreur de page inexistante du FrontOffice. Cette page est réservée exclusivement aux personnes qui sont connectés à l’application avec un compte ayant les droits administrateurs, si ce n’est pas le cas l’utilisateur est redirigé vers la page d’erreur de page inexistante du FrontOffice. L’utilisateur accède à cette page lorsqu’il appuie sur le bouton de création de compte sur la page de gestionnaire de comptes du BackOffice.

**(Voir la figure 20 dans l’annexe)**

Le menu de création d’un nouveau compte permet la création de nouveaux comptes au sein de l’application et plus spécifiquement du BackOffice. Il est constitué d’un petit formulaire où l’on retrouve les champs nécessaires à la création d’un compte à savoir le login et le mot de passe mais aussi un sélecteur permettant de définir le rôle de l’utilisateur dans l’application. Comme dans la gestion des comptes, il est nécessaire que le mot de passe rentré dans les champs prévus à cet effet ait une taille minimale de 6 et qu’elles contiennent au moins une lettre majuscule, une lettre minuscule et un chiffre. Dans le cas où le mot de passe ne respecte pas ces conditions, une boîte de dialogue apparaît pour expliquer que les mots de passe ne respectent pas les conditions.

 Dès lors que l’utilisateur à fini de remplir les champs sur la page, il lui suffit d’appuyer sur le bouton « Créer » situé en dessous du sélecteur de droits. Cela va effectuer la création du compte et rediriger l’utilisateur vers la page de gestion de comptes où il pourra voir, en plus des comptes déjà existant, le compte qu’il vient de créer.

### Complément

Comme il a été mentionné auparavant, malgré les différentes pages présentes dans le BackOffice il existe des éléments que l’on retrouve sur toutes les pages du BackOffice : ces éléments sont la partie supérieure et la partie inférieure des différentes pages, plus communément appelé « header » et « footer ».

Le footer est la partie du bas si on prend en compte le fait que le contenu principal est situé à la partie centrale qui, pour n’importe quelle page, ne voit pas son contenu changer. Le header quant à lui est la partie supérieure de la page. Son contenu diffère selon si l’utilisateur est connecté à l’application ou non. Dans le cas où l’utilisateur n’est pas connecté, il n’a pas d’utilité si ce n’est rendre la page plus esthétique.Lorsque l’utilisateur est connecté à l’application, deux nouveaux éléments font leurs apparitions dans le header comme cela est montré ci-dessus. Ces éléments sont :

* « Gestionnaire de compte » : il s’agit d’un lien permettant la redirection vers la page de gestion de compte.
* « Déconnexion » il s’agit d’une fonctionnalité permettant de se déconnecter de l’application. Lorsqu’un utilisateur appuie sur le lien de déconnexion, il est automatiquement redirigé vers le menu de connexion et il ne pourra plus accéder à aucune des fonctionnalités de l’application tant qu’il ne s’est pas reconnecté.

## **Page d’erreur 404**

Lors de l’utilisation de l’application, il est possible, selon différents cas de figures, que celui-ci ne retrouve pas la page que l’utilisateur cherche ou que les droits du compte auquel il est connecté au BackOffice ne lui permettent pas d’accéder aux fonctionnalités de la page en question.

**(Voir la figure 21 dans l’annexe)**

Dans ces cas-là, l’application redirige l’utilisateur vers une page spécialisée. Cette page est une page semblable à une page du FrontOffice ayant comme seule différence le contenu qui informe l’utilisateur que la page recherchée est inexistante.

L’application ne mentionne pas un souci d’inaccessibilité à une page due au manque de droit, cela est due vis-à-vis d’un souci de sécurité. Pour une personne externe essayant d’accéder à des pages du BackOffice, il lui sera plus difficile d’identifier les pages existantes, d’autant plus que la redirection s’effectue vers une page de FrontOffice et non du BackOffice.

# **Conclusion**

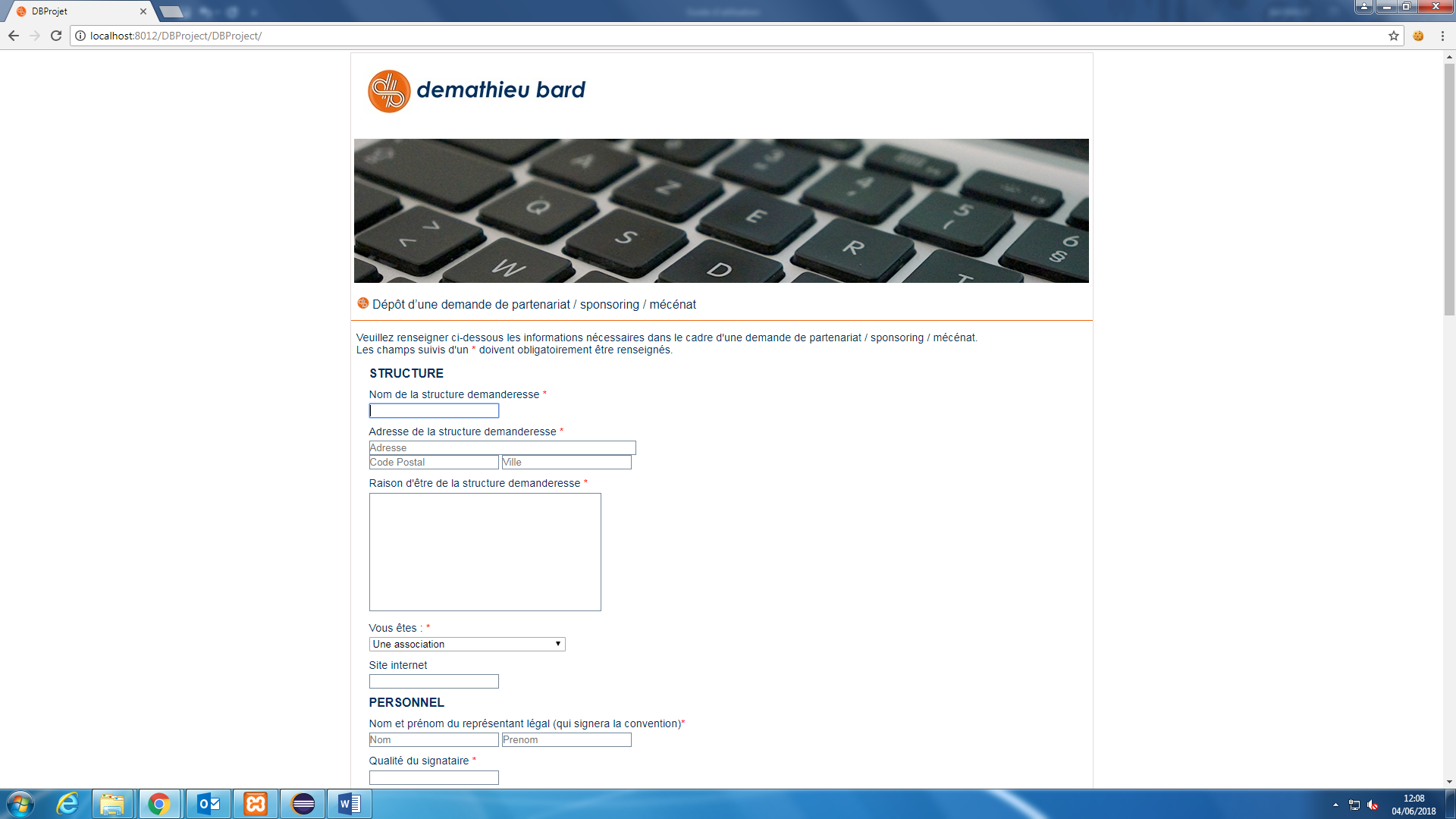
Ainsi, j’ai effectué mon stage de fin de DUT Informatique au sein de l’entreprise Demathieu Bard. Lors de ce stage de 10 semaines, j’ai pu mettre en avant les connaissances théoriques que j’ai obtenu lors de mon enseignement à l’IUT. Ce travail a été très enrichissant pour moi car il m’a permis de découvrir de nouvelles fonctionnalités en lien avec la programmation web mais aussi approfondir les connaissances que j’avais déjà acquis. Egalement, ce travail a été enrichissant puisque j’ai découvert de nouveaux aspects concernant la gestion de serveurs web à distance et l’implémentation de projets dans ceux-ci tout en étant limité aux contraintes de déploiement.

Le travail pour une entreprise et plus particulièrement pour un département spécifique m’a permis de découvrir de nouveaux aspects et de découvrir les potentiels difficultés liées à ceci. Ces difficultés peuvent être liées à l’environnement de développement qui ne permet pas d’effectuer le travail de la manière dont on le souhaite mais aussi les contraintes imposées par les responsables de projet. Le travail réalisé selon une méthode de gestion de projet (la méthode agile) pour la réalisation de l’application web a été le meilleur moyen pour moi de progresser puisque cela a permis à mon tuteur d’effectuer un meilleur suivi de mon travail.

Je pense que cette expérience en entreprise m’a apporté que de bonnes choses et m’a permis de préparer ma future insertion professionnelle dans le monde du travail. De plus, cette expérience enrichissante me conforte dans mon désir d’exercer un métier dans le domaine de l’informatique.

# **Bibliographie**

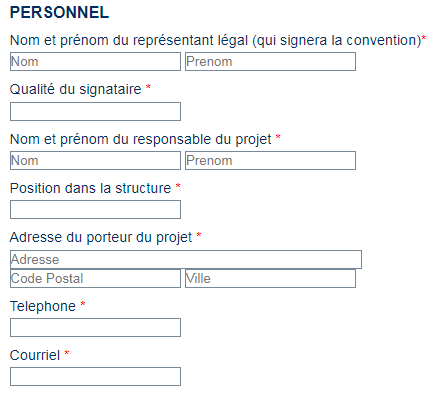
# **Annexe**



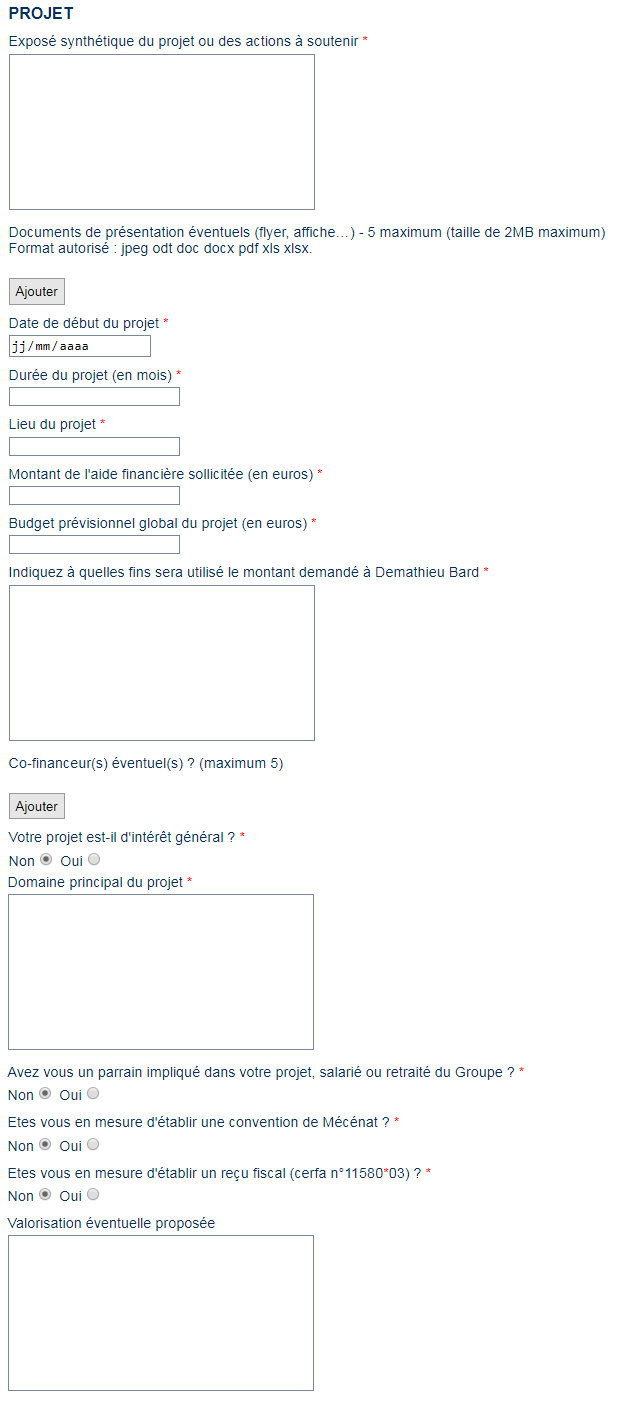
**Figure 1 : Page du formulaire du FrontOffice (coupée)**



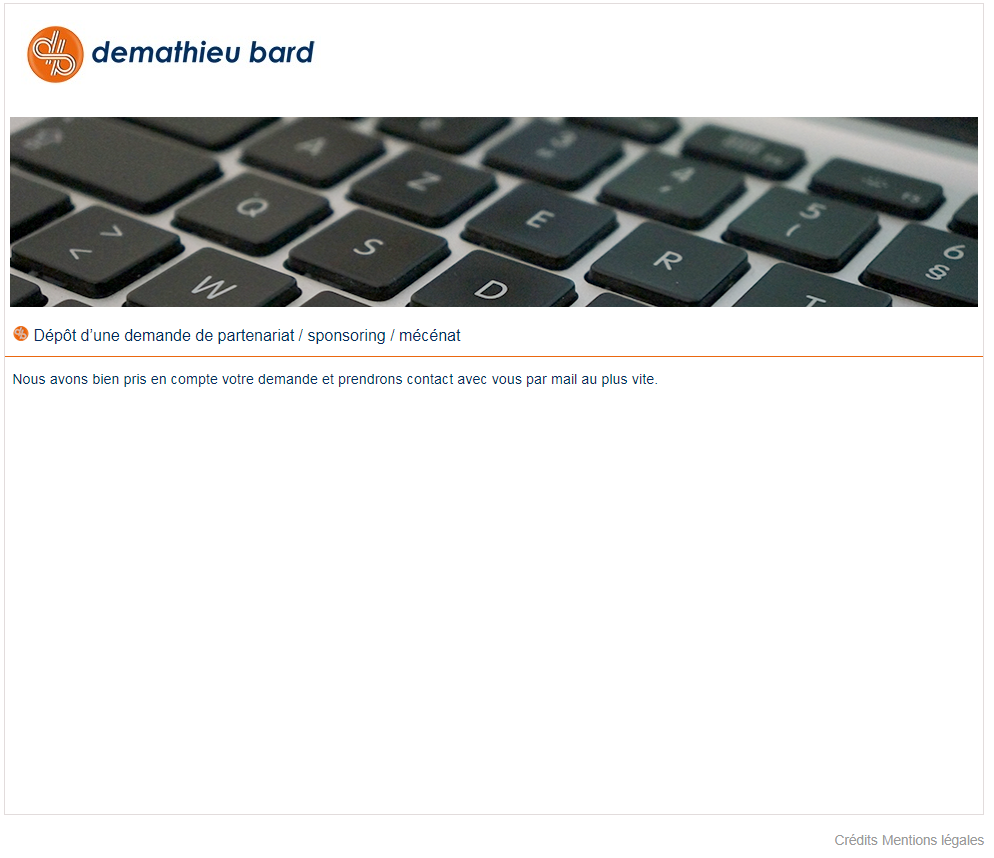
**Figure 2 : Champs associés à la structure dans le formulaire**



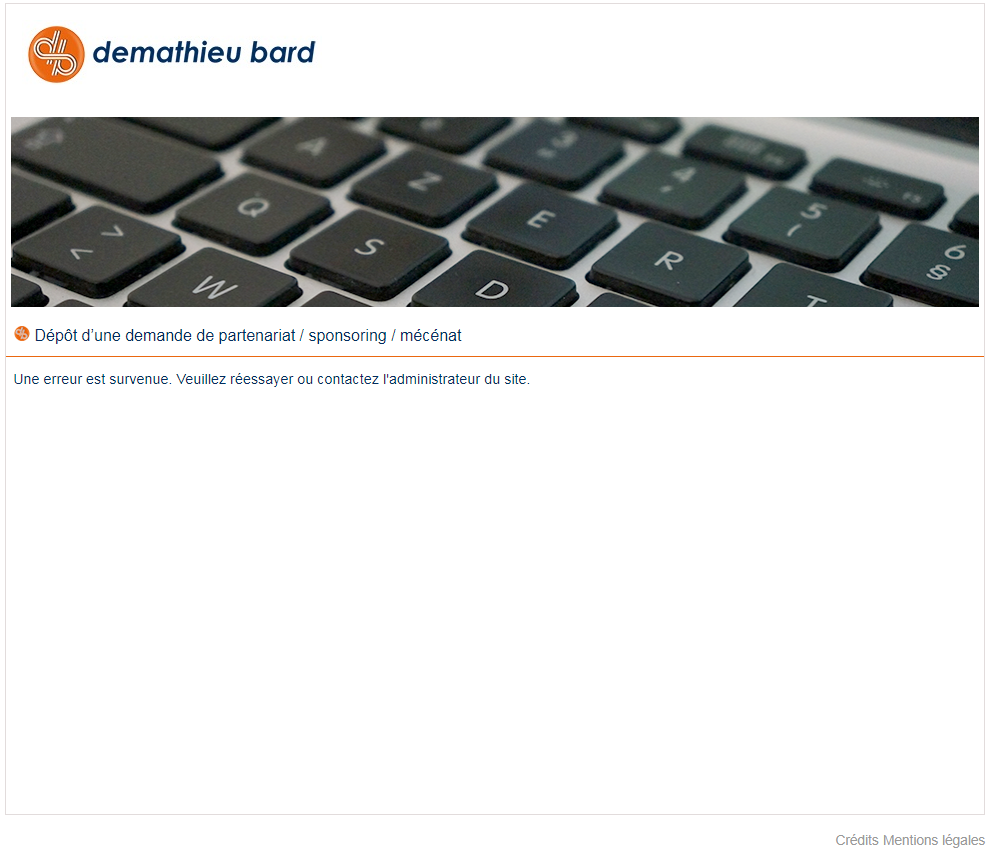
**Figure 3 : Champs associés au personnel dans le formulaire**



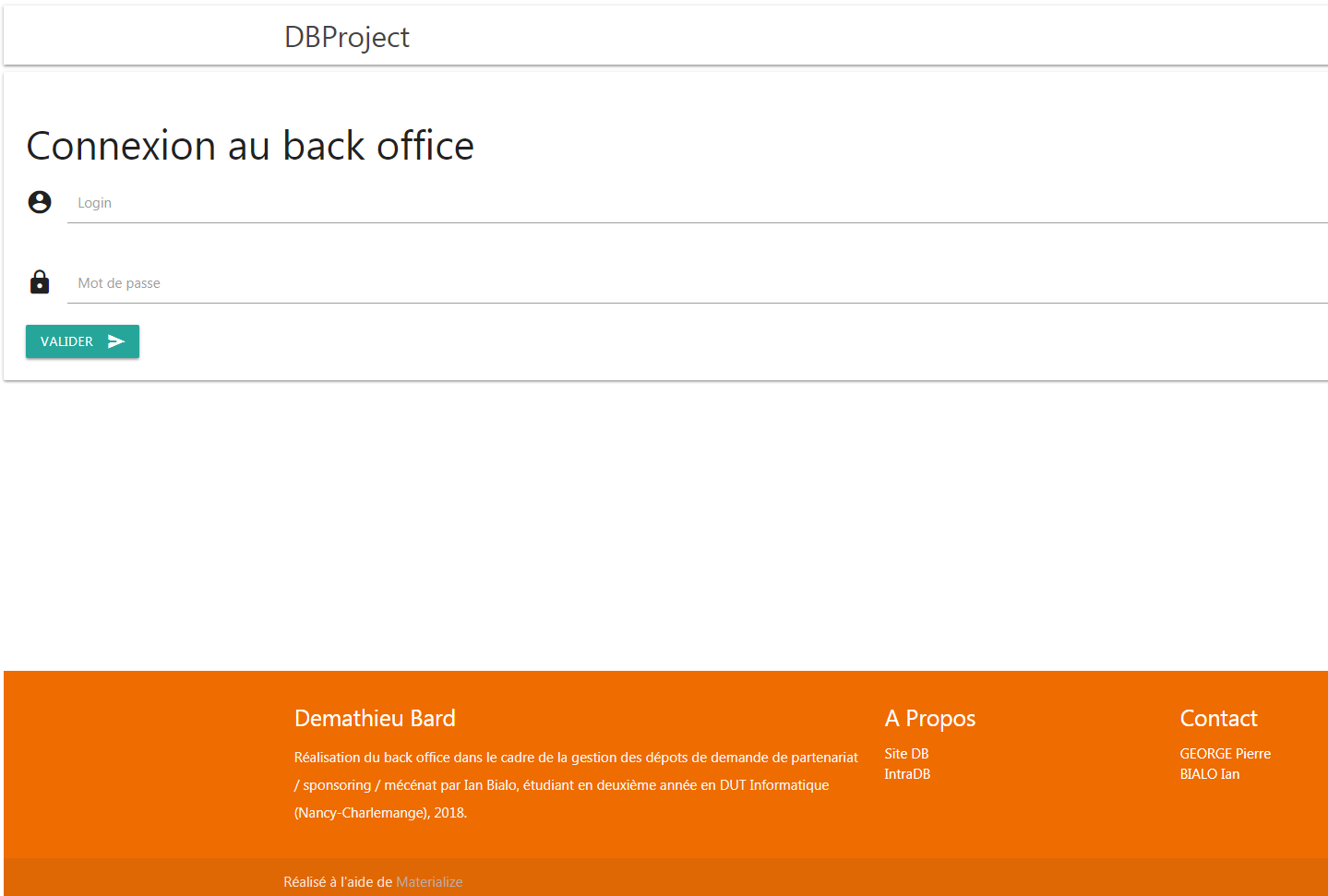
**Figure 4 : Champs associés au projet dans le formulaire**



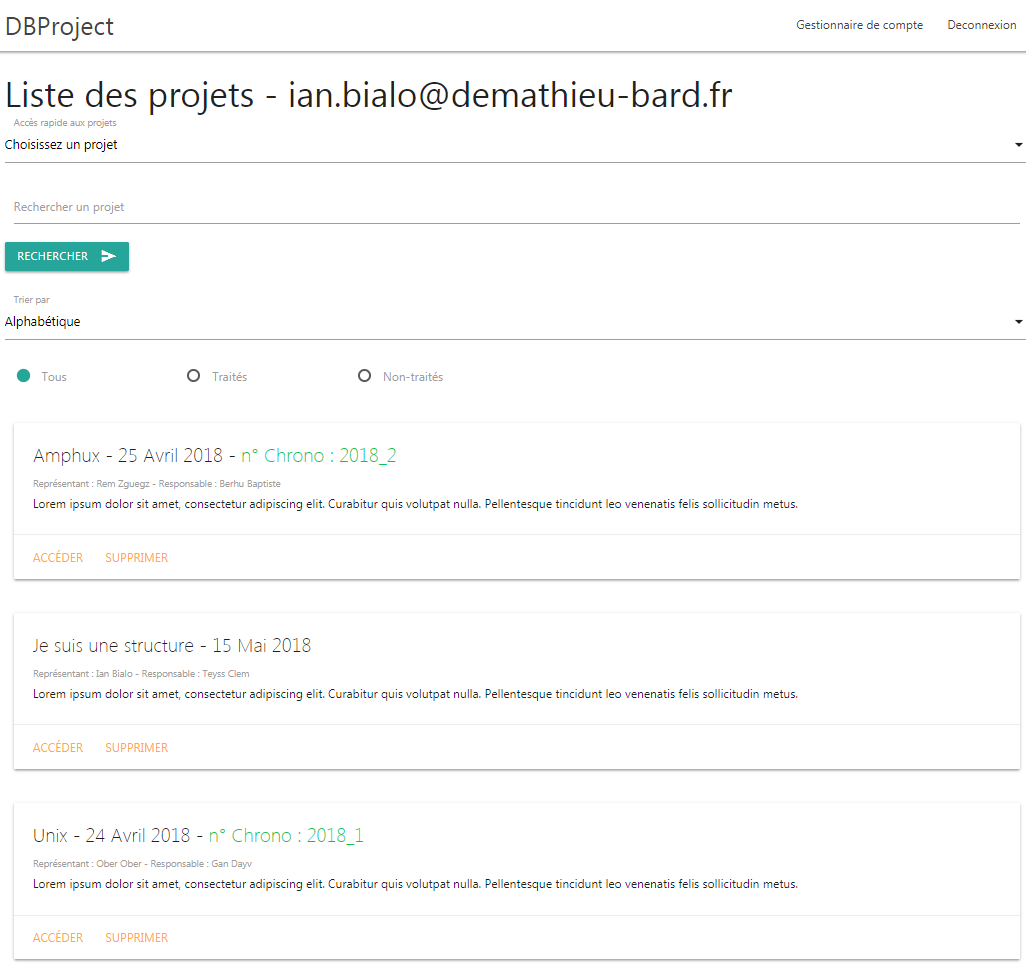
**Figure 5 : Page du message de réussite du FrontOffice**



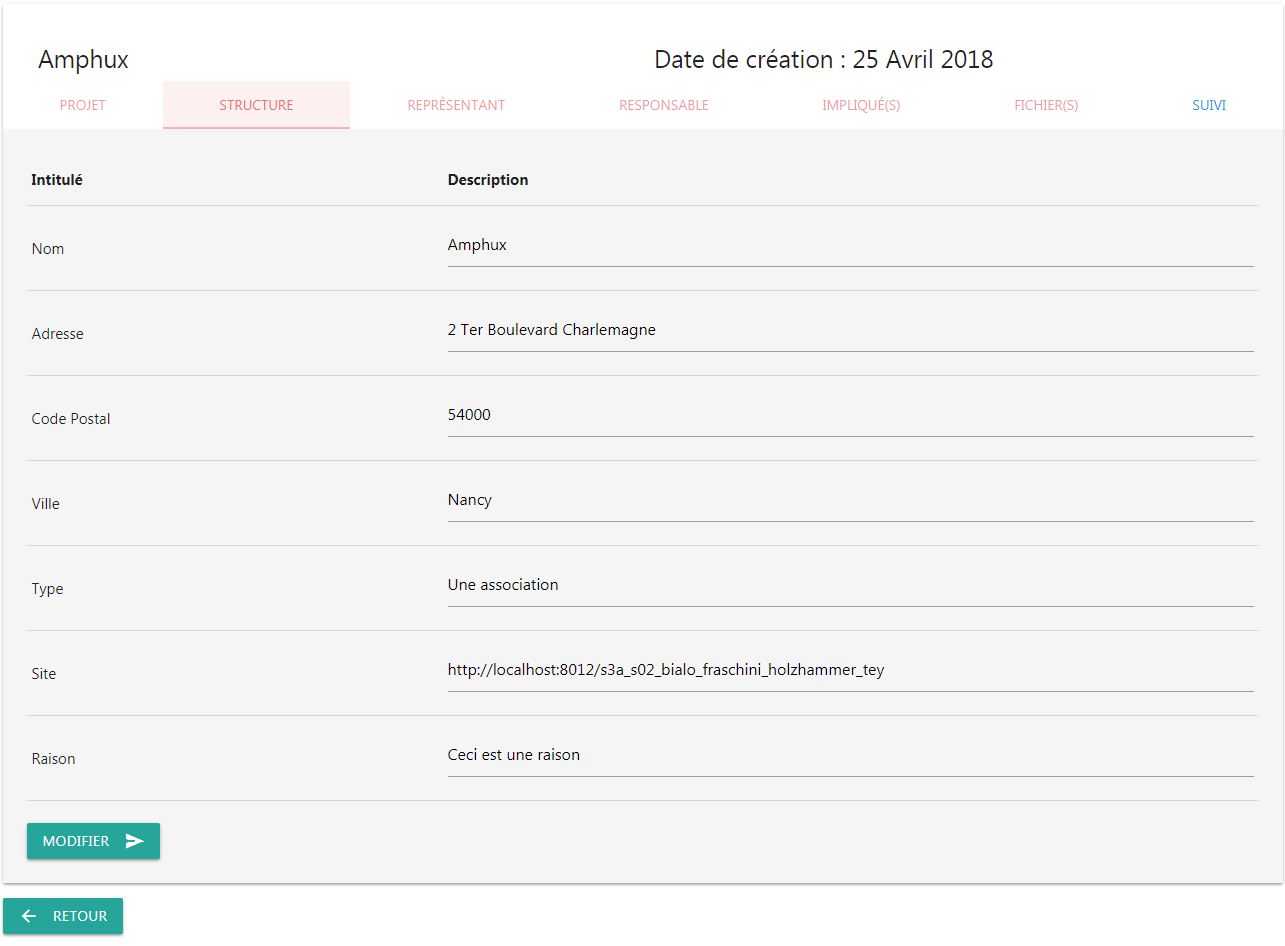
**Figure 6 : Page de message d’échec du FrontOffice**



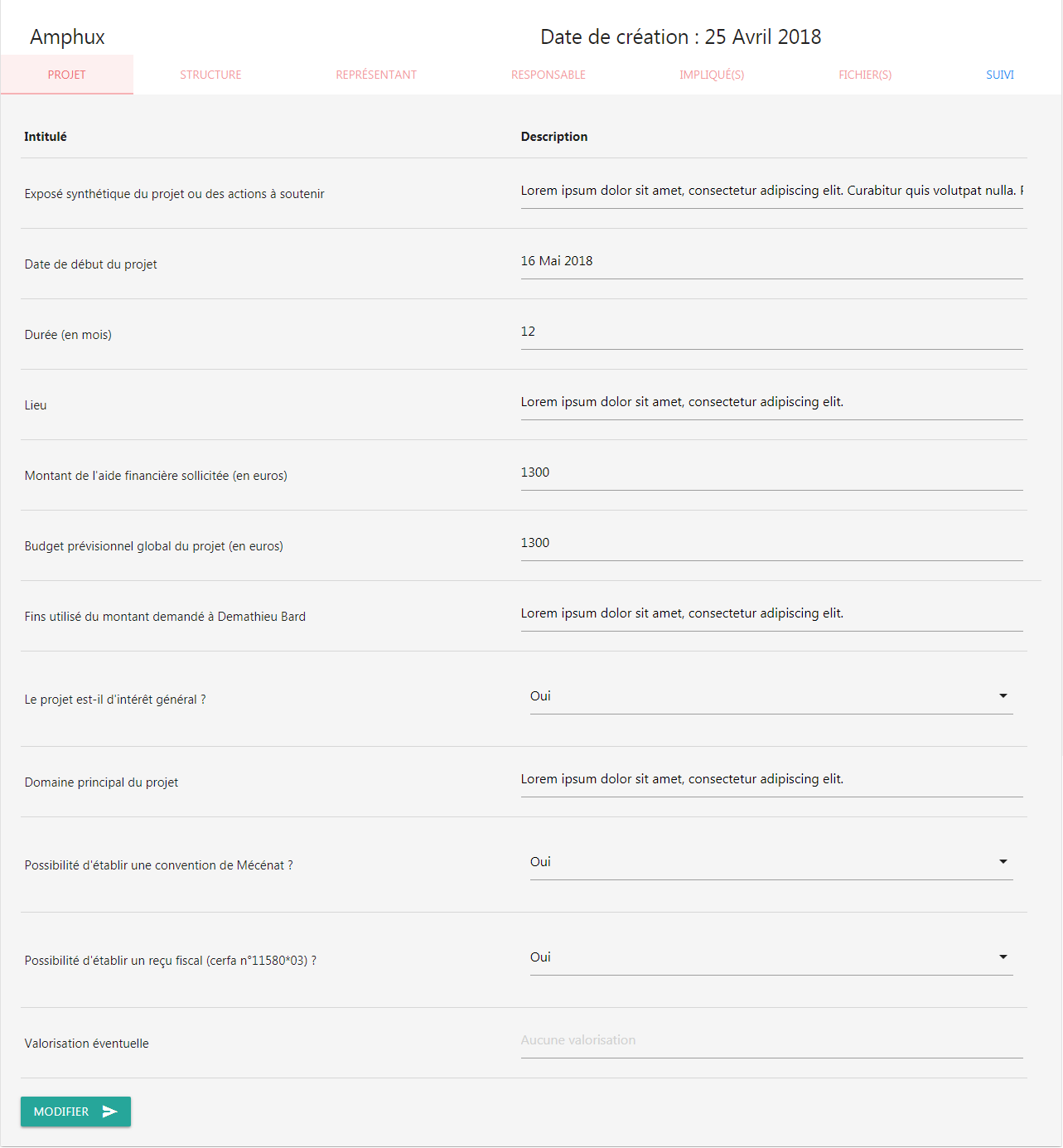
**Figure 7 : Page de menu de connexion du BackOffice**



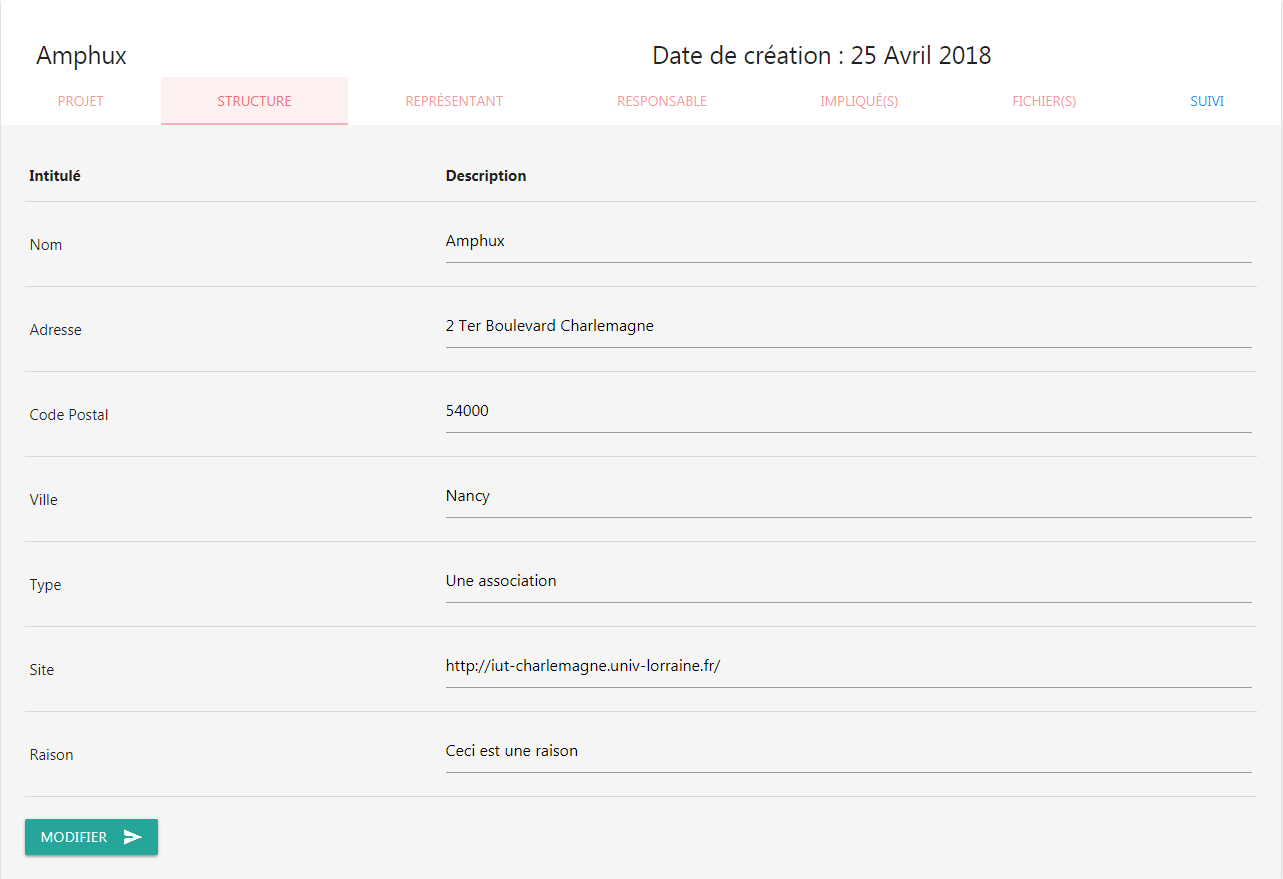
**Figure 8 : Page de la liste des projets du BackOffice**



**Figure 9 : Page de détails d’un projet du BackOffice**



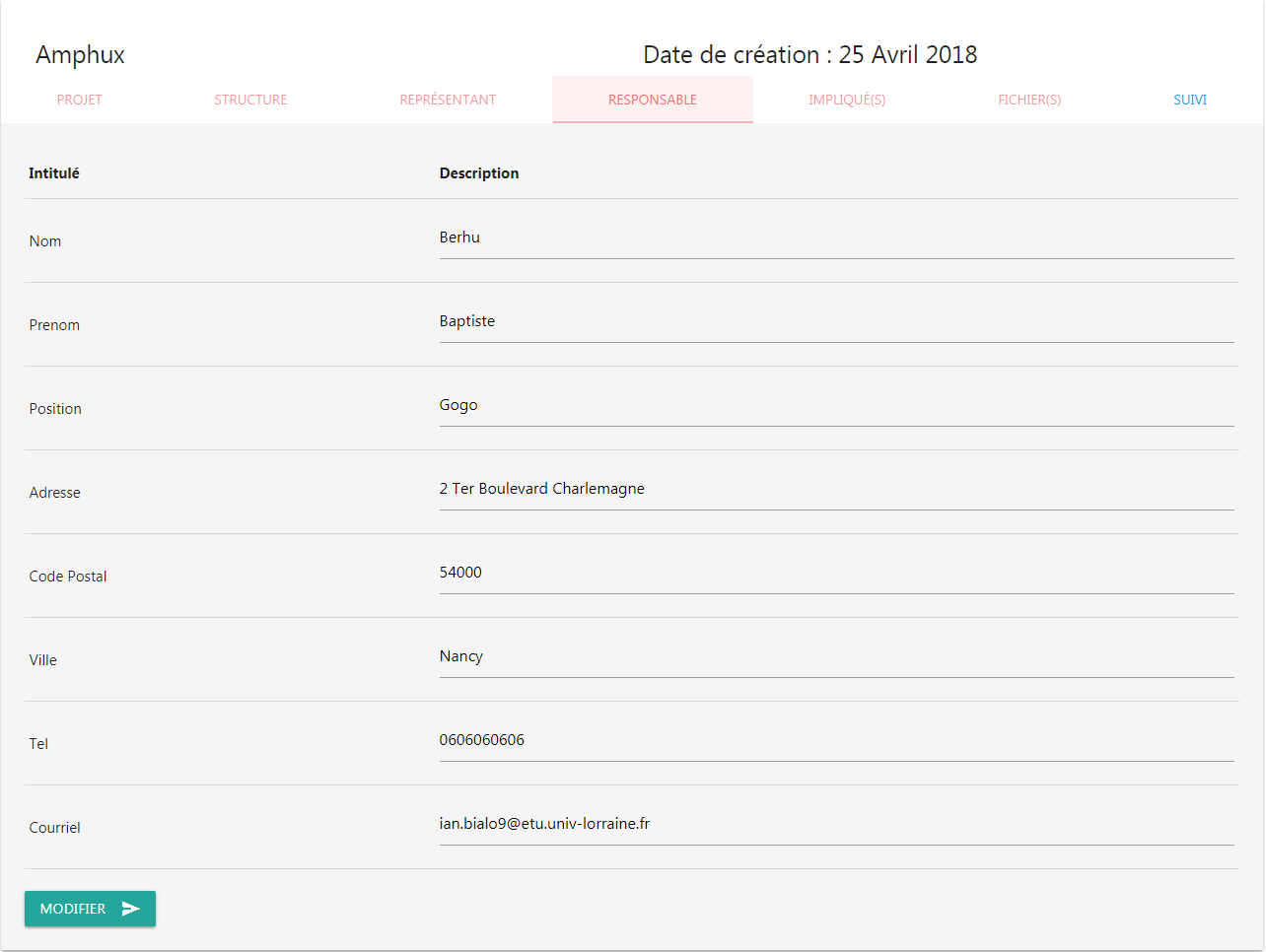
**Figure 10 : Données du projet de la page de détails d’un projet du BackOffice**



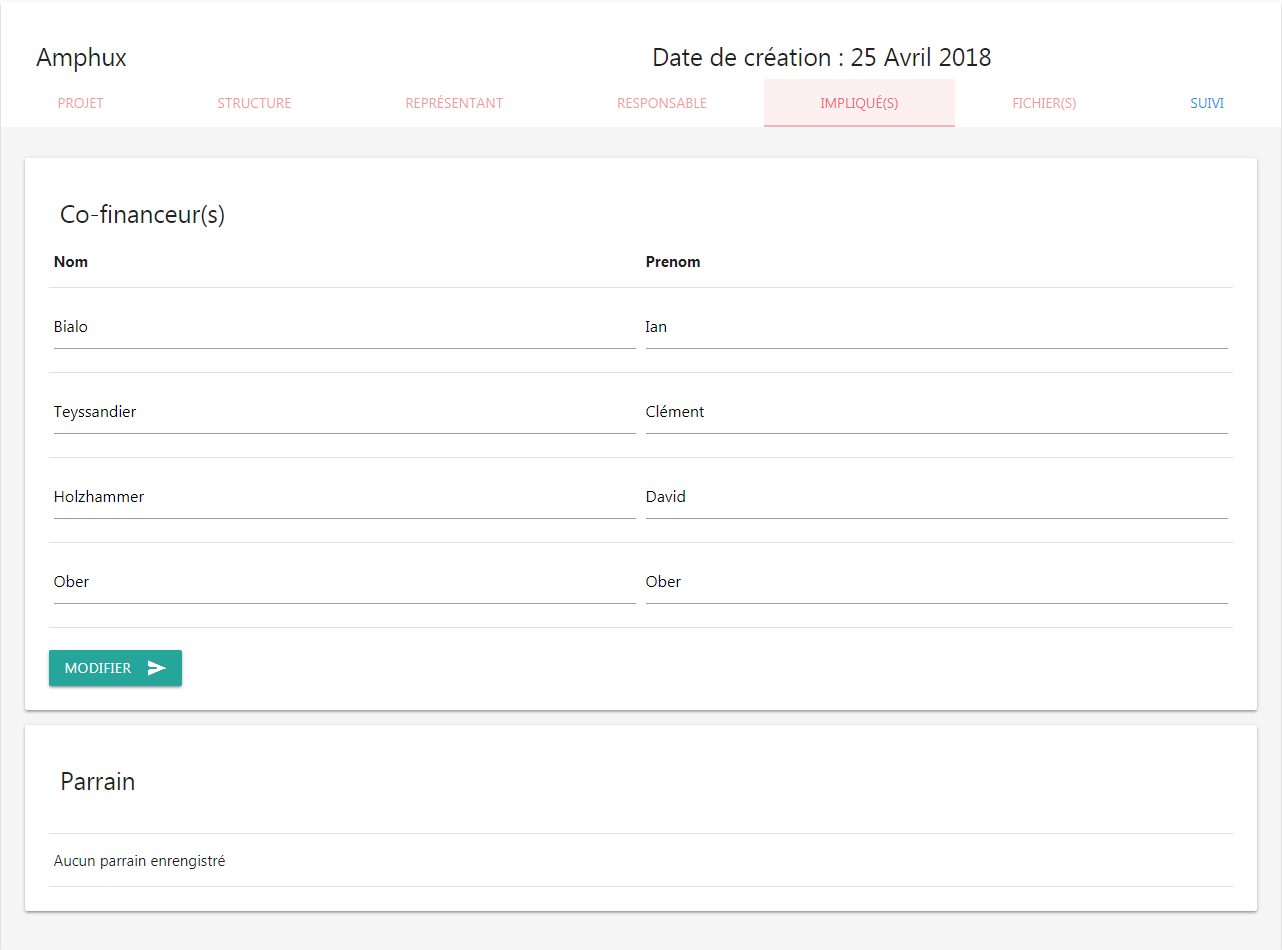
**Figure 11 : Données de la structure de la page de détails d’un projet du BackOffice**



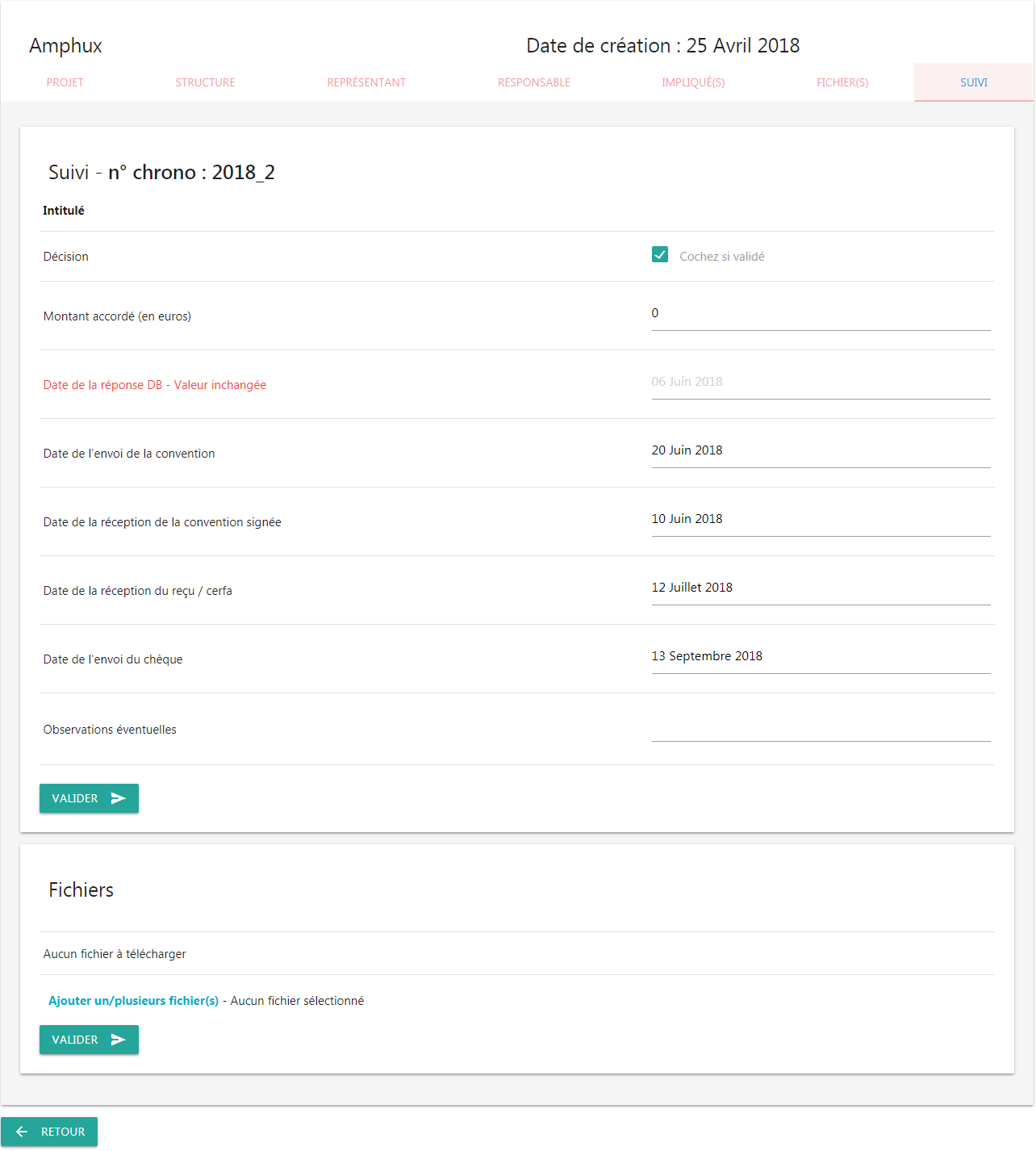
**Figure 12 : Données du représentant de la page de détails d’un projet du BackOffice**



**Figure 13 : Données du responsable de la page de détails d’un projet du BackOffice**



**Figure 14 : Données des personnes impliquées de la page de détails d’un projet du BackOffice**



**Figure 15 : Onglet de suivi de projet dans la page de détails d’un projet du BackOffice**

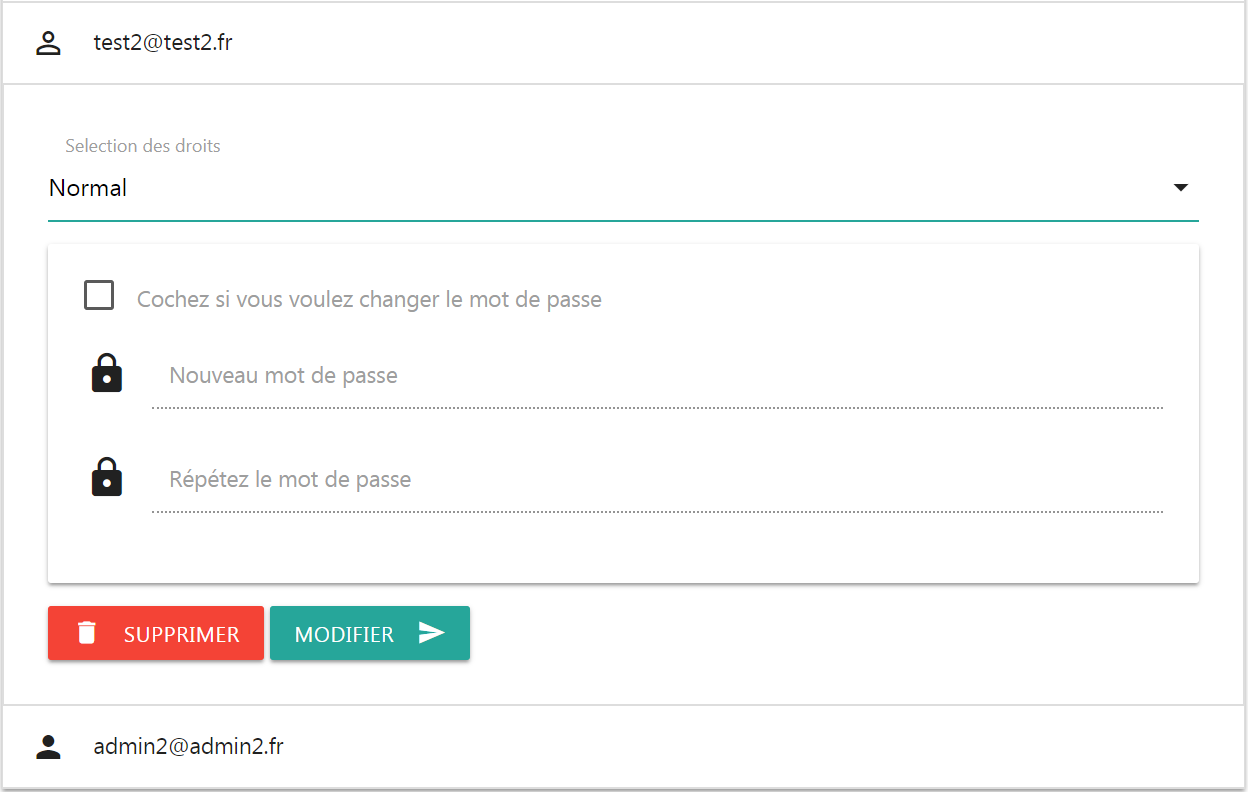


**Figure 16 : Page de recherche de projet du BackOffice avec résultats multiples**



**Figure 17 : Page de gestion des comptes du BackOffice avec un compte administrateur connecté**





**Figure 18 : Fonctionnalité de modification de compte dans la gestion de compte du BackOffice**



**Figure 19 : Page de gestion des comptes du BackOffice avec un compte normal connecté**



**Figure 20 : Page de menu de création de compte du BackOffice**



**Figure 21 : Page d’erreur de page introuvable du FrontOffice**