

ECONOMIQUE



Informatique
Campus Guillemins

Mémoire de fin d'études

en vue de l'obtention du titre de

Bachelier en Informatique de Gestion

Année académique 2016-2017

Développement d'un portail web pour la gestion d'accès des sous-traitants en milieu industriel

Defimedia

Rue Phocas Lejeune, 32
5032 Gembloux
Belgique

Présenté par **David Ratz**

Date de la défense
Juin 2017



defimedia
agence digitale



Mémoire de fin d'études

en vue de l'obtention du titre de

Bachelier en Informatique de Gestion

Année académique 2016-2017

Développement d'un portail web pour la gestion d'accès des sous-traitants en milieu industriel

Defimedia

Rue Phocas Lejeune, 32
5032 Gembloux
Belgique

Présenté par **David Ratz**

Date de la défense
Juin 2017

Remerciements

En premier lieu, je souhaite remercier tout particulièrement mon maître de stage, Monsieur Luc Ponsard, directeur des opérations chez Defimedia pour m'avoir guidé et conseillé dans la réalisation de mon projet de stage.

Je remercie également toute l'équipe de la société Defimedia de m'avoir accueilli pendant ces 15 semaines de stage.

Je remercie plus précisément, Monsieur Olivier Detaille, Analyste – web développeur , pour son aide dans la réalisation de mon projet de stage et la rédaction de mon mémoire.

Je remercie aussi Monsieur Nicolas Hendrikx, mon promoteur, pour son suivi et ses conseils relatifs notamment à la planification de mon travail et à la rédaction de ce mémoire.

Je remercie tous les professeurs de HELMo pour la formation durant ces trois années d'études.

Enfin, je souhaite remercier toutes les personnes qui prendront le temps de lire ce mémoire.

Table des matières

Introduction	7
1. Présentation de l'entreprise	8
1.1. Histoire	8
1.2. Localisation de l'entreprise	8
1.3. Services proposés	8
1.4. Clients	9
2. Présentation de Dinec International.....	10
3. Présentation du projet.....	11
3.1. Problématique.....	11
3.2. Solution	12
3.3. Objectif du stage	13
3.3.1. Front-office	13
3.3.2. Back-office	24
4. Technologies.....	27
4.1. Spring.....	27
4.1.1. Comparaison Spring vs Symfony	28
4.2. PostgreSQL	29
4.3. Hibernate	29
4.4. Freemarker	30
4.5. Bootstrap.....	30
4.6. JQuery	30
5. Outils.....	31
5.1. Maven.....	31
5.2. PgAdmin	32
5.3. Balsamiq.....	32
5.4. Modelio	32
5.5. JMerise	32
5.6. Wamp Server.....	33
5.7. Qualité du code.....	33
5.7.1. Règle d'ergonomie	33
5.7.2. Règle de sécurité.....	37
5.7.3. Performance	41
6. Implémentation du projet.....	42
6.1. Accueil	42
6.2. Connexion	43

6.3. Inscription	45
6.4. Profil de l'entreprise	47
6.5. Liste des travailleurs	47
6.5.1. Différence par rapport à la maquette du point 3.3.2.1.....	48
6.6. Inscription et modification d'un travailleur	48
6.7. Affichage d'un travailleur.....	51
6.8. Liste des projets	51
6.8.1. Différence avec le dossier d'analyse.....	52
6.9. Ajout et modification d'un projet	52
6.10. Fiche détaillée d'un projet	52
6.11. Liste des accès	53
6.11.1. Différence dossier d'analyse.....	53
6.12. Ajout et modification d'un accès	53
6.13. Affichage d'un accès	54
6.14. Ajout d'une entreprise sous-traitante.....	54
6.15. Phase de tests	55
6.15.1. Problème rencontré.....	59
6.15.2. Solution trouvée.....	59
7. Conclusion	60
8. Bibliographie.....	61
8.1. Internet	61
8.2. Images	62

Table des illustrations

Figure 1 - Logo de Defimedia	8
Figure 2 - Logo de Dinec International.....	10
Figure 3 - Situation avant le projet.....	11
Figure 4 - Situation après le projet.....	12
Figure 5 - Partie publique	13
Figure 6 - Page d'accueil.....	14
Figure 7 - Connexion.....	15
Figure 8 - Inscription d'une entreprise	16
Figure 9 - Espace entreprise.....	17
Figure 10 - Fiche détaillée de l'entreprise	18
Figure 11 - Fiche détaillée d'un travailleur.....	19
Figure 12 - Modification mot de passe.....	21
Figure 13 - Gestion des travailleurs.....	21
Figure 14 - Ajout d'un travailleur.....	22
Figure 15 - Gestion des superviseurs	24
Figure 16 - Gestion des entreprises.....	25
Figure 17 - Gestion des travailleurs.....	26
Figure 18 - Architecture Spring.....	28

Figure 19 - Logo PostgreSQL.....	29
Figure 20 - Logo Hibernate.....	29
Figure 21 - Logo Maven	31
Figure 22 - exemple POM.xml	32
Figure 23 - Navigation site web	33
Figure 24 - Organisation page web.....	34
Figure 25 - exemple gestion erreur	35
Figure 26 - exemple responsive design.....	36
Figure 27 - mauvais exemple formulaire.....	37
Figure 28 - bon exemple formulaire	37
Figure 29 - métadonnée HTML	41
Figure 30 - Vérification taille image travailleur.....	41
Figure 31 - Page d'accueil.....	42
Figure 32 - Page d'accueil connecté	42
Figure 33 - Connexion au portail.....	43
Figure 34 - logo Spring Security	43
Figure 35 - code spring-security.xml.....	44
Figure 36 - Inscription entreprise	45
Figure 37 - Table entreprise	46
Figure 38 - Utilisation BCryptPasswordEncoder	47
Figure 39 - Fiche détaillée de l'entreprise	47
Figure 40 - Liste travailleurs.....	47
Figure 41 - Inscription modification travailleur.....	48
Figure 42 - code ajout travailleur base données	49
Figure 43 - code chemin documents travailleur	49
Figure 44 - Algorithme AES	50
Figure 45 - Table travailleur	50
Figure 46 - Fiche détaillée travailleur	51
Figure 47 - Liste projet	51
Figure 48 - Ajout modification projet.....	52
Figure 49 - Jeton unique csrf.....	52
Figure 50 - Fiche détaillée projet	52
Figure 51 - Liste accès	53
Figure 52 - Ajout modification accès.....	53
Figure 53 - Accès travailleur.....	54
Figure 54 - Fiche accès	54
Figure 55 - type entreprise.....	54
Figure 56 - Configuration test Spring	55
Figure 57 - Test ajout entreprise	56
Figure 58 - méthode ajout entreprise.....	56
Figure 59 - Requete http post	57
Figure 60 - Réponse requête http post.....	57
Figure 61 - test ajout travailleur	58
Figure 62 - Requete http post	58
Figure 63 - Reponse requete http post.....	59

Introduction

Ce mémoire est l'aboutissement de trois années de formation en vue de l'obtention du titre de Bachelier en Informatique de Gestion au sein de la Haute École Libre Mosane située au Campus Guillemins à Liège lors de l'année académique 2016-2017.

Ce dernier a été rédigé dans le cadre de mon stage de fin d'études au sein de la société *Defimedia*, une agence Web digitale wallonne spécialisée dans le développement de sites internet sur-mesure et applications métiers.

Le but de mon stage est de réaliser un portail Web pour la gestion d'accès des sous-traitants en milieu industriel

Ce travail de fin d'études est composé de plusieurs parties distinctes :

1. La présentation de l'entreprise
2. La présentation de Dinec International
3. La présentation du projet
4. Les technologies utilisées
5. Les outils utilisés
6. L'implémentation du projet

1. Présentation de l'entreprise



Figure 1 - Logo de Defimedia

1.1. Histoire

La société a été fondée en 1992, sous le nom *IMmedia* par Madame Anne Marie Heller et Monsieur Luc Ponsard, d'abord active dans le monde du multimédia et qui a ensuite évolué vers le Web. C'est en 2005 qu'*IMmedia* s'associe avec une autre société, *DefinITION* créée par Monsieur Olivier de Wasseige et change de nom pour devenir *Defimedia*.

Pionnière dans le monde du Web et de l'expérience digitale, *Defimedia*, aujourd'hui rattaché au groupe *Contraste Europe*, spécialisé en IT, déploie son expérience tant en Wallonie qu'à l'international.

Defimedia dispose d'une équipe d'une vingtaine de spécialistes faisant partie d'un groupe international de 330 personnes et avec une présence à Namur et Liège, mais aussi à Bruxelles, Luxembourg, Genève, Paris et Lille.

1.2. Localisation de l'entreprise

La vingtaine de personnels est répartie sur deux principaux sites :

- Un site : rue Phocas Lejeune, 32 à 5032 Gembloux.
C'est le siège social de l'entreprise.
- L'autre site est situé rue de Mulhouse, 36 à 4020 Liège.
J'ai réalisé mon stage sur ce lieu.

1.3. Services proposés

Defimedia est une entreprise qui offre des services informatiques pour des sociétés et des particuliers.

Forte de son expérience et de celle du groupe *Contraste Europe*, *Defimedia* est idéalement positionnée pour offrir un service tant aux spécialistes du marketing et de la communication qu'aux responsables IT.

Parmi ces activités, nous pouvons citer :

- Le référencement

Elle permet de gagner en visibilité sur les requêtes des internautes. Aussi bien avant le développement du site Web que pendant sa conception ou après sa mise en ligne, ainsi elle permet d'obtenir une visibilité immédiate sur les moteurs de recherche.

- Les canaux digitaux

Elle aide au choix du meilleur canal de communication (site Web, e-mail, réseau social, applications mobiles...) et donne des orientations sur la stratégie à adopter en fonction du canal choisi

- La stratégie de contenu

Elle développe le contenu des sites Web en fonction des objectifs du client, du public cible et des canaux.

- L'Analyse

Elle effectue une analyse stratégique qui permet de définir les objectifs business du client et du public cible, de réaliser un benchmark concurrentiel, et de structurer les contenus de sites Web sous forme de wireframes.

1.4. Clients

Defimedia possède de nombreux clients tels que :

- Wallonia Clusters

Approche Mobile First pour le site des Clusters !

Sous la houlette de la Direction générale opérationnelle de l'Économie, de l'Emploi et de la Recherche, *defimedia* avait mis en ligne en 2011 une nouvelle version du portail des Pôles de compétitivité et des Réseaux d'entreprises wallons (Clusters).

En 2017, *defimedia* a été chargé d'aligner le portail aux tendances Web actuelles. Notre équipe de Digital Experience Consultants a réalisé une restructuration des contenus et a défini les nouveaux wireframes de la plateforme. Le tout dans une approche Mobile First.

Le site est donc désormais adapté à toutes les tailles d'écrans et est optimisé pour une navigation sur mobile.

- Clinique André Renard

Nouveau site Web pour la Clinique André Renard

La Clinique André Renard est une clinique de proximité qui prône la qualité et l'humanisation des soins hospitaliers. Son bâtiment principal est situé à Herstal mais elle dispose également de 4 polycliniques en région liégeoise.

Afin d'assurer une présence sur le Web et de présenter l'étendue de ses services, la Clinique André Renard a fait appel aux équipes de *defimedia* pour la réalisation de son premier site Web.

- Fidusud

Defimedia a développé le nouveau site Web de *Fidusud*, société spécialisée dans la gestion de dossiers de contentieux. Le site a été réalisé avec le CMS Wordpress et est disponible en 4 langues.

Dans le cadre de cette refonte, un travail de réécriture Web a également été réalisé par notre équipe Webmarketing.

Sur ce nouveau site Web vous trouverez, entre autres, toutes les informations sur les services proposés par *Fidusud* et un extranet.

- SOWACCESS

Un nouveau site Web en Drupal pour la SOWACCESS

La SOWACCESS, interlocuteur neutre pour la transmission ou l'acquisition d'entreprise en Wallonie, a remis à neuf son site Web.

Basé sur une approche centrée sur l'utilisateur, ce nouveau site internet offre directement toutes les informations nécessaires aux personnes désireuses de vendre ou d'acheter une entreprise.

2. Présentation de Dinec International



Figure 2 - Logo de Dinec International

« Dinec International est le partenaire de référence en matière de sécurité électronique, d'offre globale de surveillance des bâtiments ainsi qu'en gestion de temps. Notre présence de plus de 30 ans sur les marchés internationaux est un gage de pérennité et de sérieux pour vos investissements sécuritaires. »¹

« Dinec International développe des solutions de sécurité adaptées à vos besoins : »²

- Contrôle d'accès
- Système anti-intrusion
- Chaîne du froid
- Gestion de bâtiments
- Gestion de temps
- Gestion énergétique

« Les produits Dinec sont distribués et installés par des partenaires en Belgique, France, Danemark, Pays-Bas, Luxembourg, Madagascar, Maroc, Sénégal, Côte d'Ivoire, Cameroun, Bénin et Algérie. »³

Les installations des partenaires dans le secteur industriel, dans le secteur sportif, des loisirs et du tourisme, dans le secteur de la grande distribution, dans le secteur public, dans le secteur hospitalier, dans le secteur bancaire.

Le siège social de *Dinec international* se situe : rue de la gare, 30 à 1420 Braine-l'Alleud.

¹ <http://www.dinec.be/fr/dinec-international-qui-est-ce.html?IDC=832>

² <http://www.dinec.be/fr/index.html?IDC=2>

³ <http://www.dinec.be/fr/references.html?IDC=161>

3. Présentation du projet

3.1. Problématique

Le schéma ci-dessous explique le fonctionnement d'une demande de projet et d'accès à un chantier avant la réalisation de mon projet.

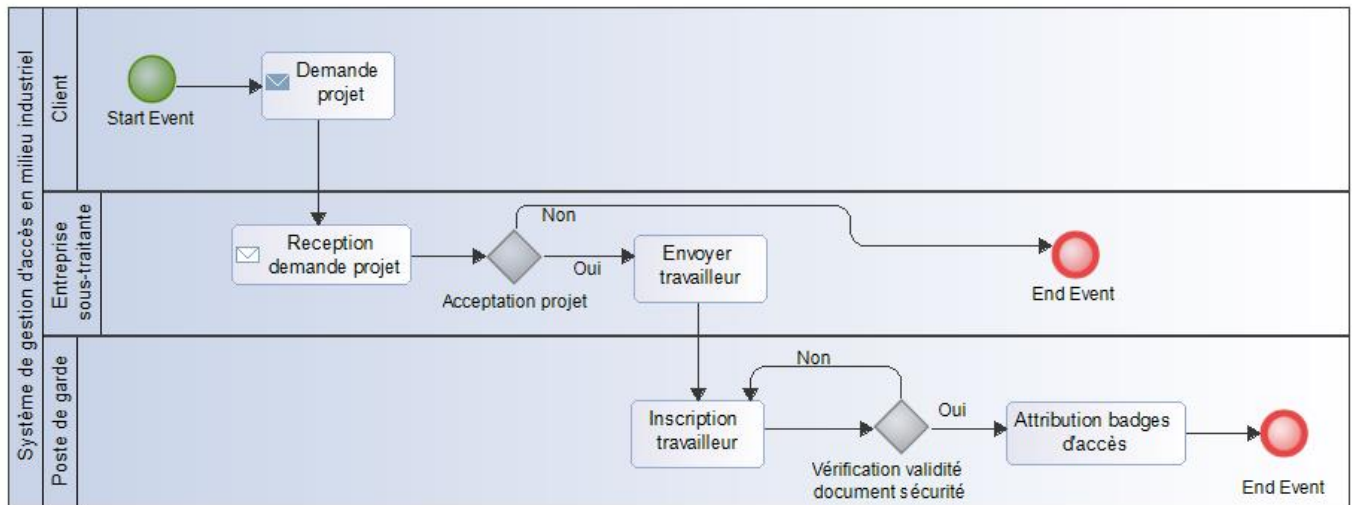


Figure 3 - Situation avant le projet

Dans le milieu industriel, lorsqu'une entreprise⁴ reçoit un bon de commande pour un projet, celle-ci est amenée à envoyer des travailleurs pour effectuer les tâches de construction ou de maintenance.

Lorsque ces personnes se présentent au poste de garde, celles-ci doivent s'inscrire avant de pouvoir accéder au site. Les travailleurs doivent aussi apporter des documents tels que l'E101⁵, Limosa, Certificat médical, copie passeport... et le gardien doit vérifier leurs validités.

Étant donné le nombre de travailleurs envoyés par une entreprise et que ceux-ci sont souvent étrangers, il n'est pas toujours facile d'en gérer le flux (vu les quantités) d'autant plus que les difficultés de langue ne facilitent pas la communication.

À cette problématique s'ajoute le fait que certaines entreprises mandatées font appel à des sous-traitants.

⁴ Lorsque je parlerai de « Entreprise » dans la suite de ce mémoire, cela fait référence à une entreprise sous-traitante.

⁵ Le formulaire E101 est utilisé pour attester de la législation applicable à un travailleur qui n'est pas affilié dans le pays de travail.

3.2. Solution

Le schéma ci-dessous explique le fonctionnement d'une demande de projet et d'accès à un chantier après la réalisation de mon projet.

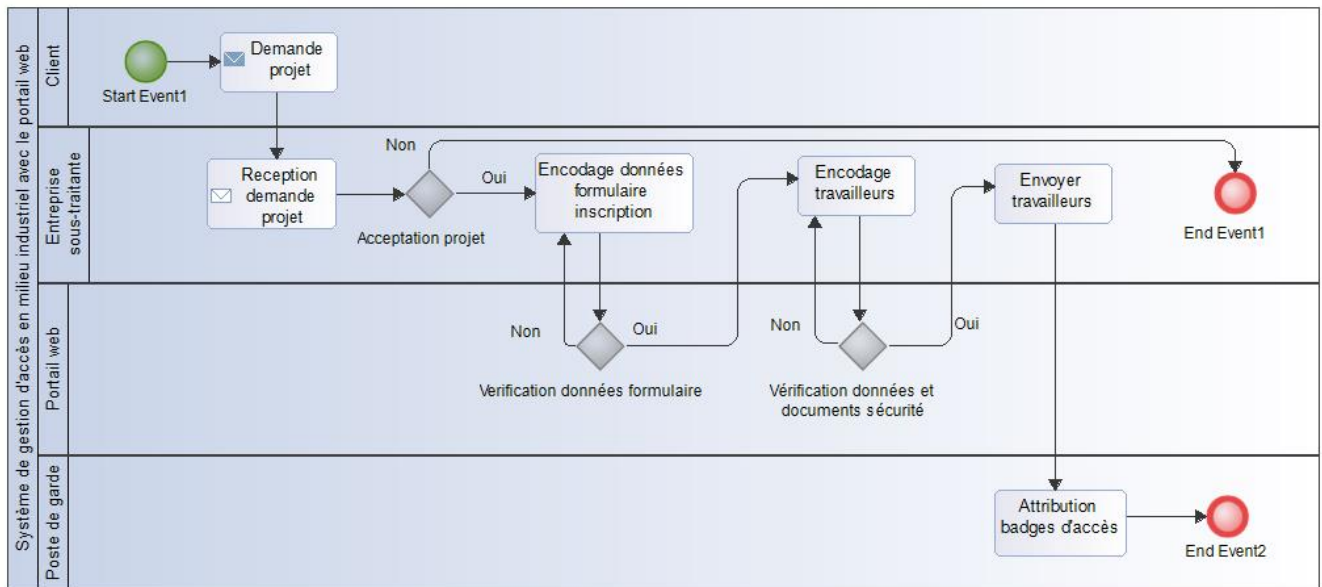


Figure 4 - Situation après le projet

La société *Dinec International* qui gère le contrôle d'accès souhaiterait une solution permettant aux entreprises qui envoient des travailleurs chez un client d'enregistrer elles-mêmes leurs travailleurs. Le client se chargeant de valider les enregistrements.

Le système demandé devra être accessible au travers d'Internet, il se fera sous la forme d'un portail Web.

Ce portail Web permettra :

- ✓ D'optimiser l'accueil des fournisseurs en gérant les accès
- ✓ De préparer les travailleurs à intervenir sur le site du Client en leur communiquant l'information de sécurité nécessaire
- ✓ D'anticiper les problèmes d'accès liés à l'expiration des droits d'accès, notamment de type Limosa

Ce portail sera utilisé sur différents types d'appareil, celui-ci devra donc être un site Web adaptatif et respecter les règles d'ergonomie (celles-ci sont expliquées à la section 5.7 Qualité du code) afin de s'adapter à n'importe lequel de ces appareils.

Étant donné que le portail stockera les données personnelles de la société (numéro BCE, nom, statut de la société, etc) et les données confidentielles des travailleurs (nom, prénom...), ainsi que les documents de ceux-ci.

Afin de sécuriser mon portail contre les attaques, je respecte les règles de sécurité OWASP⁶.

⁶ Les règles de sécurité OWASP (Open Web Application Security Project) sont expliquées à la section 5.7 Qualité du code.

3.3. Objectif du stage

L'objectif de ce stage est de réaliser un portail Web pour la gestion d'accès des sous-traitants en milieu industriel pour la société *Dinec International*.

Après avoir parcouru l'offre de *Defimedia* pour le projet de « Développement d'une solution d'inscription de visiteurs en milieu industriel », j'ai réalisé un dossier d'analyse me permettant de bien cerner les attentes du client dont le contenu est expliqué ci-dessous. Ce dossier a ensuite été validé par un analyste de la société. (Voir le schéma et les explications des tables de la base de données en annexe).

Voici les fonctionnalités attendues pour ce projet :

3.3.1. Front-office

Ce portail possède une partie front-office avec une partie publique pour tous les utilisateurs et une partie privée pour le responsable de l'entreprise.

3.3.1.1. Partie publique

Le visiteur en accédant sur le portail pourra effectuer les fonctionnalités suivantes :

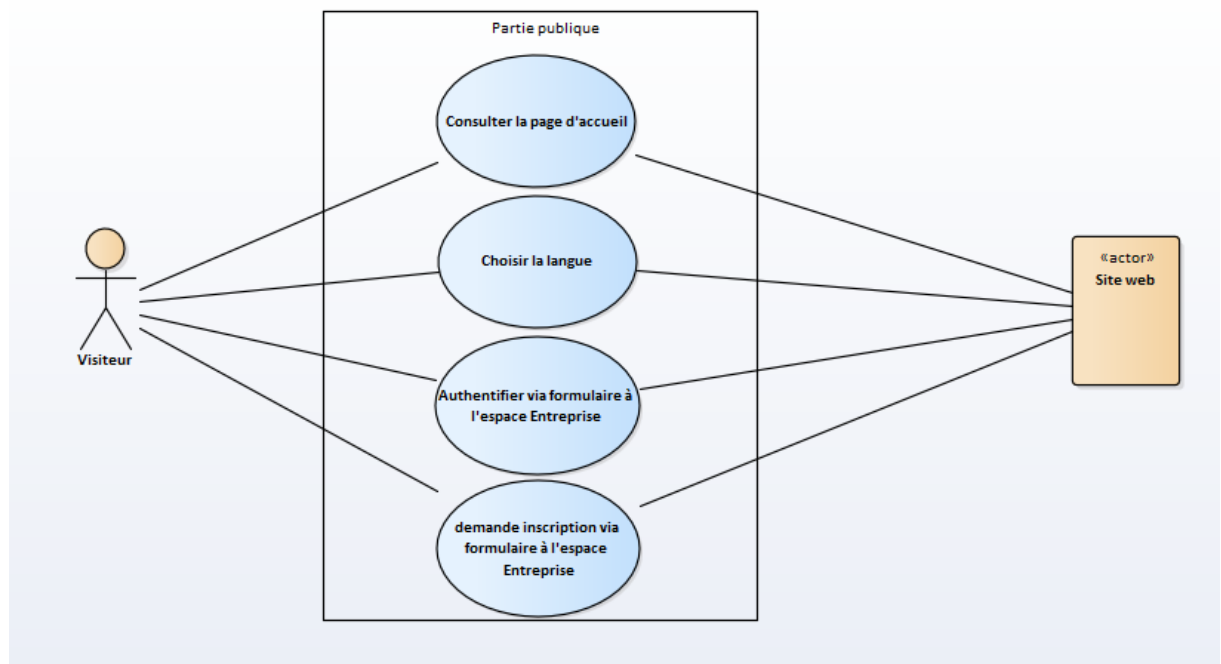


Figure 5 - Partie publique

3.3.1.1.1. Consultation de la page d'accueil

En tant qu'utilisateur du portail, je souhaite pouvoir consulter la page d'accueil du portail afin de me connecter ou de m'inscrire à l'espace Entreprise.

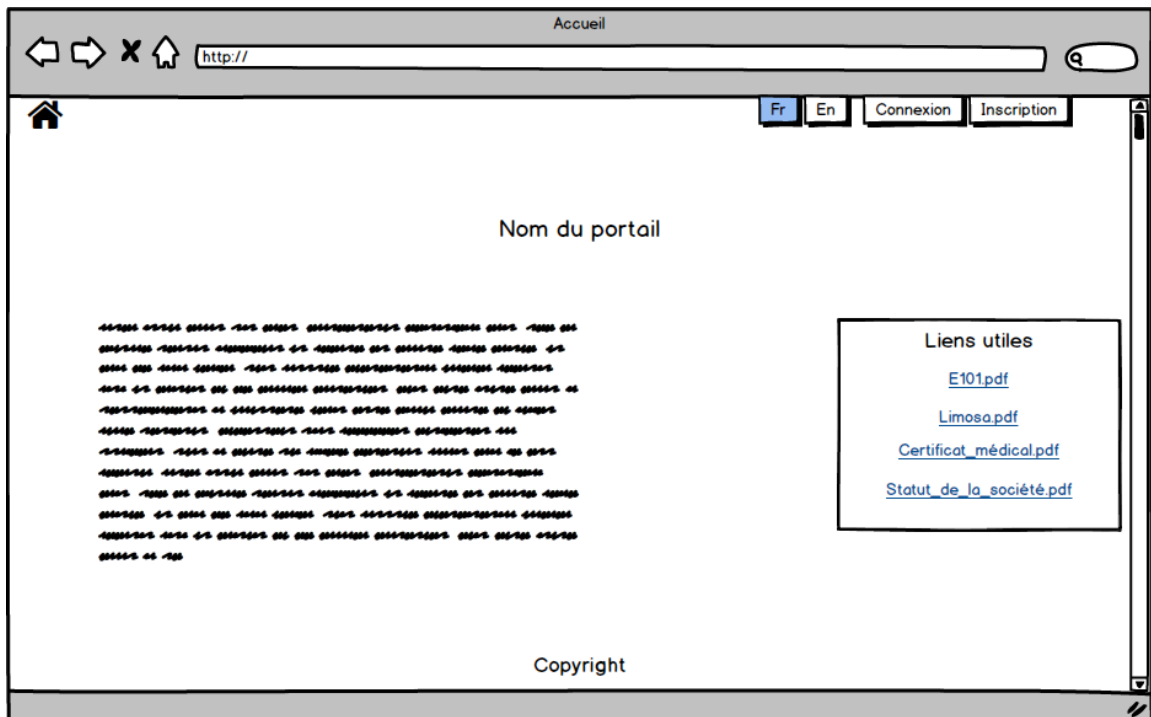


Figure 6 - Page d'accueil

L'utilisateur pourra se connecter au portail en cliquant sur le bouton « Connexion » (voir Figure 7).

Il pourra aussi s'inscrire au portail en cliquant sur le bouton « Inscription » (voir Figure 8).

Celui-ci pourra choisir la langue du portail en cliquant sur celle désirée « Fr » pour français ou « En » pour anglais.

Le visiteur trouvera aussi sur la droite de l'écran des liens permettant de télécharger les modèles de documents à remplir et à ajouter pour chaque travailleur (voir Figure 11).

3.3.1.1.2. Authentification à l'espace entreprise

En tant qu'utilisateur, je souhaite me connecter à l'espace Entreprise afin d'y ajouter mes travailleurs.

Connexion espace Entreprise

Fr En Connexion Inscription

Connexion

Adresse email : *

Mot de passe : *

Connexion

[Mot de passe oublié ?](#)

Vous n'êtes pas encore inscrit : [Inscrivez-vous ici](#)

Copyright

Figure 7 - Connexion

L'utilisateur pourra se connecter au portail en encodant l'adresse email et le mot de passe encodés lors de l'inscription (voir Figure 8).

Si celui-ci n'a pas encore inscrit son entreprise sur le portail, il pourra accéder au formulaire d'inscription d'une entreprise via le lien à droite.

Le responsable de l'entreprise pourra utiliser le formulaire de mot de passe oublié en cliquant sur le lien en dessous du bouton « Connexion » si celui-ci a oublié son mot de passe.

3.3.1.1.3. Inscription à l'espace entreprise

En tant qu'utilisateur, je souhaite inscrire une entreprise afin de pouvoir me connecter au portail et ajouter des travailleurs.

The screenshot shows a web browser window titled 'Inscription Entreprise'. The address bar shows 'http://'. The page has a header with a home icon, a language selector (Fr, En), and buttons for 'Connexion' and 'Inscription'. The main heading is 'Inscription entreprise'. The form is divided into several sections:

- Inscription entreprise**:
 - Número BCE: * (text input)
 - Nom de l'entreprise: * (text input)
 - Personne de contact**:
 - Nom: * (text input)
 - Prénom: * (text input)
 - Téléphone/GSM: * (text input)
 - Email: * (text input)
 - Adresse du siège social**:
 - Adresse: * (text input)
 - Code postal: * (text input)
 - Localité: * (text input)
 - Pays: * (text input)
 - ☐ Société sous-traitante maître ☐ Sous-traitante d'une société maître
 - Raison de la demande: * (dropdown menu)
 - Numéro de référence: * (text input)
 - Remarques: (text input)
 - Documents**:
 - Bon de commande: * (text input)
 - Statut de la société: * (text input)
 - Mot de passe: * (password input)
 - Confirmation mot de passe: * (password input)
 - ReCaptcha *: ☐ I'm not a robot (with reCAPTCHA logo and links to Privacy and Terms)
- Liens utiles**:
 - [E101.pdf](#)
 - [Limosa.pdf](#)
 - [Certificat_médical.pdf](#)
 - [Statut_de_la_société.pdf](#)

At the bottom right of the form is an 'Inscription' button. Below the form, the word 'Copyright' is displayed.

Figure 8 - Inscription d'une entreprise

L'écran de la page précédente (voir figure 8) permet à l'utilisateur d'inscrire son entreprise.

Le responsable devra encoder les informations de l'entreprise suivantes :

- Numéro BCE de l'entreprise
- Le nom de l'entreprise
- Les informations de la personne de contact
 - Nom
 - Prénom
 - GSM/Telephone
 - Email
- L'adresse du siège social
 - Rue
 - Numéro
 - Code postal
 - Localité
 - Pays
- Le type d'entreprise
- La raison de la demande d'inscription
- Le numéro de référence du projet à réaliser
- Les remarques (optionnel)
- Les documents concernant la société :
 - Le bon de commande reçu du client
 - Le statut de la société
- Le mot de passe

Ensuite, il cliquera sur le bouton « Inscription » afin d'inscrire son entreprise sur le portail.

3.3.1.2. Espace entreprise

Le responsable de l'entreprise pourra effectuer les fonctionnalités suivantes :

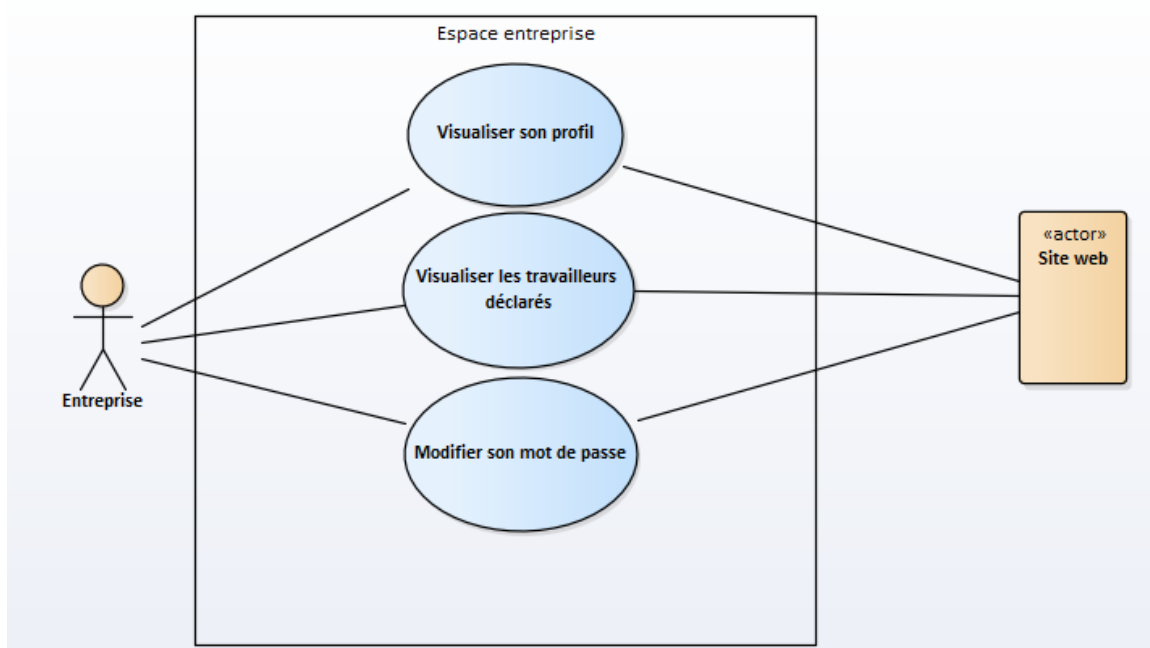


Figure 9 - Espace entreprise

3.3.1.2.1. Visualisation du profil de l'entreprise

En tant que responsable de l'entreprise, je souhaite visualiser les informations de mon entreprise afin de vérifier ou modifier mes informations.

The screenshot shows a web browser window titled 'Espace Entreprise'. The address bar shows 'http://'. The page has a header with 'Fr', 'En', 'Nom entreprise connectée ici', and a 'Deconnexion' button. The main content area is titled 'Espace Entreprise' and contains a 'Fiche détaillée entreprise' form. The form has several sections: 'Nom de l'entreprise', 'Personne de contact', 'Adresse du siège social', 'Type', 'Nom du sous-traitant maître', 'Etat', 'Date de validation', 'Commentaire (si refusée)', 'Raison de la demande', 'Numéro de référence', 'Remarques', 'Documents', and 'Liste des travailleurs déclarés'. The 'Liste des travailleurs déclarés' section contains a table with 8 columns: Nom, Prenom, Nationalite, Langue, Adresse email, Date de début, Date de fin, and Etat. The table has 2 rows of data. Below the table, there are fields for 'Date et heure de création', 'Utilisateur ayant créé le travailleur', 'Date et heure de dernière modification', and 'Utilisateur ayant effectué la dernière modification'. At the bottom, there is a button 'Ajouter nouveau travailleur' and two buttons 'Modifier' and 'Supprimer'.

Fiche détaillée entreprise

Nom de l'entreprise

Numéro BCE : Numéro BCE ici

Personne de contact

Nom : Nom ici

Prenom : Prenom ici

Téléphone / GSM : Tel / Gsm ici

Email : Email ici

Adresse du siège social

Adresse : Rue, numéro ici

Code postal : Code postal ici

Localité : Localité ici

Pays : Pays ici

Type : Sous-traitant maître ou sous-traitant de sous-traitant maître

Nom du sous-traitant maître : Nom ici

Etat : En attente, validée ou refusée

Date de validation : Date ici

Commentaire (si refusée) : Commentaire ici

Raison de la demande : Raison ici

Numéro de référence : Numéro de référence ici

Remarques : Remarques ici

Documents

Bon de commande : Fichier bon de comande

Statut de la société : Fichier statut de la société

Liste des travailleurs déclarés

Nom	Prenom	Nationalite	Langue	Adresse email	Date de début	Date de fin	Etat
Nom	Prenom	Français	Fr	test@mail.be	01/01/17	31/12/17	<input type="checkbox"/>
Nom 2	Prenom 2	Belge	Nl	test@gmail.com	01/01/17	31/12/17	<input checked="" type="checkbox"/>

Date et heure de création : Date et heure de création ici

Utilisateur ayant créé le travailleur : Utilisateur ayant créé le travailleur ici

Date et heure de dernière modification : Date et heure de dernière modification ici

Utilisateur ayant effectué la dernière modification : Utilisateur ayant effectué la dernière modification ici

Ajouter nouveau travailleur

Modifier Supprimer

Copyright

Figure 10 - Fiche détaillée de l'entreprise

Le responsable de l'entreprise, une fois connecté, pourra visualiser les informations de son entreprise en cliquant sur le nom de son entreprise (en haut à droite).

Sur cette page, celui-ci pourra aussi visualiser la liste de ses travailleurs déclarés et la possibilité d'en ajouter un nouveau en cliquant sur le bouton « Ajouter nouveau travailleur ».

Le responsable pourra visualiser quand et qui a ajouté et modifié l'entreprise.

Il pourra aussi modifier ou supprimer son entreprise en cliquant sur le bouton « Modifier » ou « Supprimer »

3.3.1.2.2. Visualiser les travailleurs déclarés

En tant que responsable de l'entreprise, je souhaite pouvoir visualiser les travailleurs déclarés afin de vérifier si ceux-ci sont bien en ordre pour accéder au site du client.

The screenshot shows a web browser window titled 'Espace Entreprise'. The address bar shows 'http://'. The page has a header with a home icon, language buttons 'Fr' and 'En', a login field 'Nom entreprise connectée ici', and a 'Deconnexion' button. The main content area is titled 'Espace Entreprise' and contains a 'Fiche détaillée travailleur' section. This section is divided into 'Informations personnelles' and 'Documents'. The 'Informations personnelles' section includes a placeholder for a photo, a 'Numéro de référence', 'Entreprise', 'Nom', 'Prenom', 'Nationalité', 'Langue', and 'Adresse email'. The 'Documents' section lists four document types: 'E101', 'Limosa', 'Certificat médical', and 'Copie passeport', each with a 'Fichier' upload field and a 'Date de validité du document' field. Below these, there are several administrative fields: 'Date de début de validité', 'Date de fin de validité', 'Etat' (with options 'En attente, validée ou refusée'), 'Date de validation', and five 'Réponse question' fields. At the bottom of the card, there are 'Date de validité du contrôle d'accès', 'Travailleur sur liste noire', 'Date et heure de création', 'Utilisateur ayant créé le travailleur', 'Date et heure de dernière modification', and 'Utilisateur ayant effectué la dernière modification'. At the very bottom of the card are 'Modifier' and 'Supprimer' buttons. The footer of the page says 'Copyright'.

Espace Entreprise

Fr En Nom entreprise connectée ici Deconnexion

Espace Entreprise

Fiche détaillée travailleur

Informations personnelles

Numéro de référence : Numéro de référence ici

Entreprise : Nom entreprise ici

Nom : Nom ici

Prenom : Prenom ici

Nationalité : Nationalité ici

Langue : Langue ici

Adresse email : Adresse email ici

Date de début de validité : Date de début de validité ici

Date de fin de validité : Date de fin de validité ici

Etat : En attente, validée ou refusée

Date de validation : Date de validation ici

Réponse question 1 : Réponse question 1 ici Réponse question 2 : Réponse question 2 ici

Réponse question 3 : Réponse question 3 ici Réponse question 4 : Réponse question 4 ici

Réponse question 5 : Réponse question 5 ici

Documents

E101

Fichier : Fichier ici

Date de validité du document : Date de validité du document ici

Limosa

Fichier : Fichier ici

Date de validité du document : Date de validité du document ici

Certificat médical

Fichier : Fichier ici

Date de validité du document : Date de validité du document ici

Copie passeport

Fichier : Fichier ici

Date de validité du document : Date de validité du document ici

Date de validité du contrôle d'accès : Date de validité du contrôle d'accès ici

Travailleur sur liste noire : Travailleur sur liste noire ici

Date et heure de création : Date et heure de création ici

Utilisateur ayant créé le travailleur : Utilisateur ayant créé le travailleur ici

Date et heure de dernière modification : Date et heure de dernière modification ici

Utilisateur ayant effectué la dernière modification : Utilisateur ayant effectué la dernière modification ici

Modifier Supprimer

Copyright

Figure 11 - Fiche détaillée d'un travailleur

Le responsable arrivera sur cette page après avoir cliqué sur une ligne du tableau de travailleur de la page de profil de l'entreprise. Il pourra visualiser les informations du travailleur sélectionné suivantes :

- La photo
- Le numéro de référence
- Le nom de l'entreprise
- Le nom
- Le prénom
- La date de naissance
- La nationalité
- La langue
- Le numéro de GSM
- L'adresse email
- Les dates de début et de fin de validité.
- L'état de la demande et la date de validation.

Les documents E101, Limosa, certificat médical et copie de passeport du travailleur et leurs dates de validités.

Il pourra aussi visualiser les informations suivantes :

- La date du contrôle d'accès
- Si l'utilisateur se trouve en liste noire.
- La date et heure de création et modification
- La personne ayant ajouté et modifié le travailleur

Le responsable de l'entreprise pourra modifier ou supprimer le travailleur en cliquant sur le bouton « Modifier » ou « Supprimer ».

3.3.1.2.3. Modification du mot de passe

En tant que responsable de l'entreprise, je souhaite modifier mon mot de passe actuel.

Modification mot de passe

Fr En Connexion Inscription

Modifier mot de passe

Mot de passe actuel : *

Nouveau mot de passe : *

Confirmation nouveau mot de passe : *

Modifier

Copyright

Figure 12 - Modification mot de passe

Le responsable de l'entreprise pourra modifier son mot de passe encodé lors de l'inscription.

3.3.1.3. Gestion des travailleurs

Cette partie présente les fonctionnalités attendues pour la gestion des travailleurs.

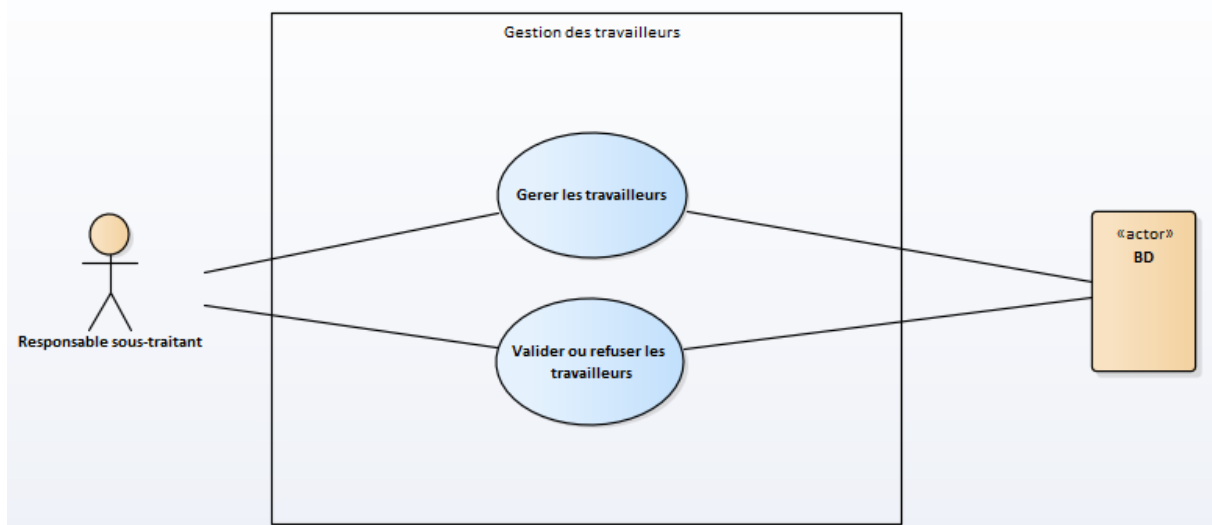


Figure 13 - Gestion des travailleurs

3.3.1.3.1. Ajout d'un travailleur

En tant que responsable de l'entreprise, je souhaite pouvoir ajouter les travailleurs de ma société moi-même afin de vérifier que les données sont correctement encodées et de limiter le nombre d'utilisateurs ayant accès à ce portail.

The screenshot shows a web browser window titled 'Espace Entreprise'. The address bar shows 'http://'. The page has a header with a home icon, language buttons 'Fr' and 'En', the text 'Nom entreprise connectée ici', and a 'Deconnexion' button. The main content area is titled 'Espace Entreprise' and contains a section 'Ajout travailleur'. This section has a sub-section 'Informations personnelles' with a placeholder for a profile picture (a square with an 'X') and several text input fields: 'Entreprise : *' (pre-filled with 'Nom entreprise connectée ici'), 'Nom : *', 'Prenom : *', 'Nationalité : *', 'Langue : *', and 'Adresse email : *'. Below these are two date input fields: 'Date de début de validité : *' and 'Date de fin de validité : *'. The 'Documents' section follows, containing four sub-sections: 'E101', 'Limosa', 'Certificat médical', and 'Copie passeport'. Each sub-section has a 'Fichier : *' input field and a 'Date de validité du document : *' input field. An 'Ajouter' button is located at the bottom right of the 'Ajout travailleur' section. The footer of the page says 'Copyright'.

Figure 14 - Ajout d'un travailleur

Après avoir cliqué sur le bouton « ajouter nouveau travailleur » de la page de profil de l'entreprise, le responsable arrivera sur cette page (voir figure 14 sur la page précédente) et il pourra encoder les informations d'un travailleur suivantes :

- La photo
- Le nom
- Le prénom
- La date de naissance
- La nationalité
- La langue
- Le numéro de GSM
- L'adresse email
- Les dates de début et de fin de validité (celles-ci représenteront les dates d'un projet)
- Les documents E101, Limosa, certificat médical et copie passeport ainsi que leurs dates de validités.

Il ajoutera le travailleur en cliquant sur le bouton « Ajouter ».

3.3.1.3.2. Modification d'un travailleur

En tant que responsable de l'entreprise, je souhaite pouvoir modifier les travailleurs de ma société moi-même afin de modifier leurs données.

L'écran est le même que celui pour l'ajout d'un travailleur (voir Figure 14).

3.3.1.3.3. Suppression d'un travailleur

En tant que responsable de l'entreprise, je souhaite pouvoir supprimer les travailleurs de ma société moi-même afin d'être sûr que ceux-ci n'existent plus.

3.3.2. Back-office

Ce portail possède aussi une partie back-office pour les superviseurs avec une partie administration.

3.3.2.1. Gestion des superviseurs

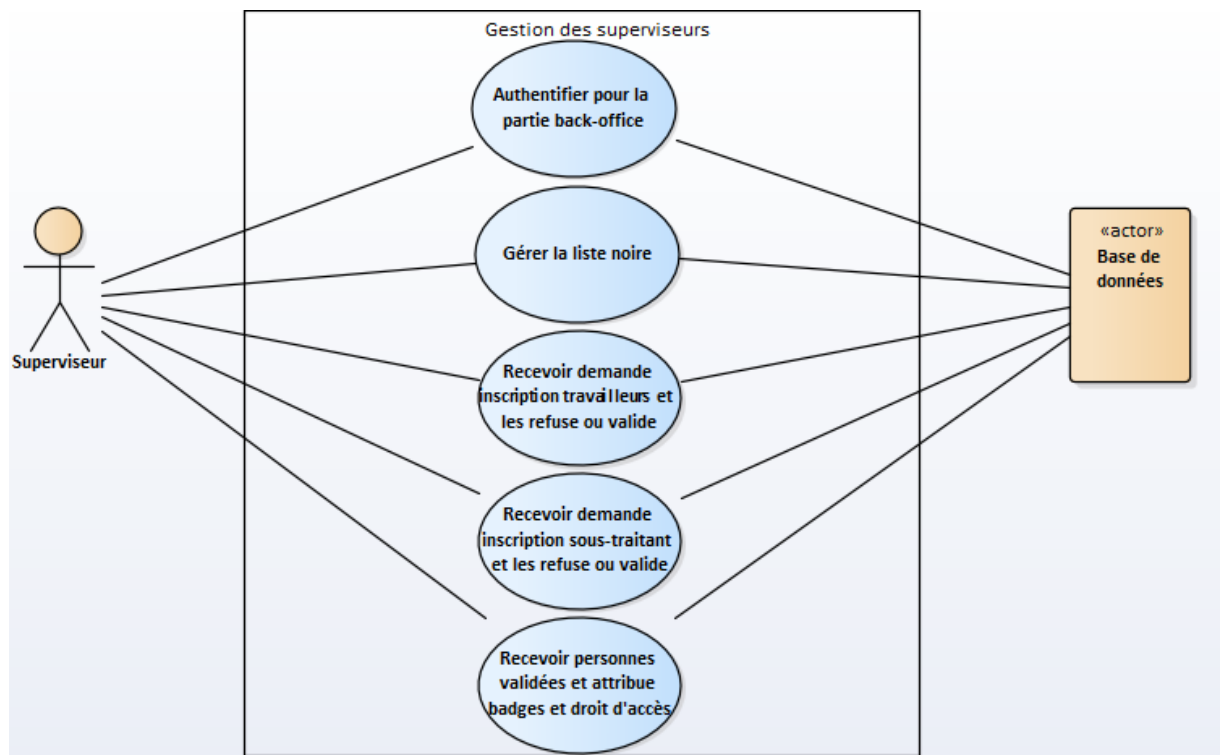


Figure 15 - Gestion des superviseurs

3.3.2.1.1. Authentification au back-office

En tant qu'Administrateur (Superviseur), je souhaite pouvoir m'authentifier au portail afin de valider ou refuser les demandes d'inscription des entreprises ou des travailleurs.

L'écran est le même que celui de connexion d'une entreprise (voir Figure 3).

Sauf que le rôle de l'utilisateur doit être de type « superviseur ».

Il y a quatre types de superviseurs :

- *Superviseur1* : reçoit les demandes d'inscription des sous-traitants et les refuse ou les valide
- *Superviseur2* : reçoit les demandes d'inscription des travailleurs et les refuse ou les valide
- *Superviseur3* : reçoit les personnes validées et doit leur attribuer un badge et des droits d'accès
- *Superviseur4* : gère la liste noire des travailleurs de l'entreprise sous-traitante

3.3.2.1.2. Gestion des entreprises

En tant qu'Administrateur (Superviseur), je souhaite pouvoir gérer les entreprises enregistrées sur le portail afin de modifier ou supprimer celles-ci.

The screenshot shows a web browser window titled 'Espace BackOffice'. The browser's address bar shows 'http://'. The page has a header with a home icon, language buttons 'Fr' and 'En', a field for 'Nom superviseur ici', and a 'Deconnexion' button. The main content area is titled 'Espace BackOffice' and contains two sections: 'Filtrer les entreprises' and 'Liste des entreprises'.

The 'Filtrer les entreprises' section includes the following fields:

- Nom de l'entreprise :
- Nom du sous-traitant maître :
- Type : (dropdown menu)
- État : (dropdown menu with options: Validée, Refusée)
- Numéro de référence du travailleur :
- Nom du travailleur :

A 'Filtrer' button is located at the bottom right of the filter section.

The 'Liste des entreprises' section features a table with the following columns: Nom, Nom sous-traitant maître, Type, Code postal, Localité, Pays, and État. The table is paginated with '50 éléments par page' and an 'Ajouter sous-traitant' button. The first three rows of the table are visible:

Nom	Nom sous-traitant maître	Type	Code postal	Localité	Pays	État
Nom	Nom sous-traitant maître	Sous-traitant maître	4020	Liège	Belgique	<input type="checkbox"/>
Nom	Nom sous-traitant maître	Sous-traitant de sous-traitant maître	4000	Liège	Belgique	<input checked="" type="checkbox"/>

The page footer contains the text 'Copyright'.

Figure 16 - Gestion des entreprises

Cet écran permet à l'administrateur de rechercher une entreprise, selon certains critères, parmi toutes les entreprises inscrites sur le portail.

Les critères sont les suivants :

- Le nom de l'entreprise
- Le nom de l'entreprise sous-traitante maître
- Le type de l'entreprise
- L'état de la demande
- Le numéro de référence du travailleur
- Le nom du travailleur

La liste des entreprises sera paginée (exemple : 50 entreprises par page).

L'administrateur pourra aussi ajouter une nouvelle entreprise en cliquant sur le bouton « ajouter sous-traitant » au-dessus de la liste des entreprises.

Gestions des travailleurs

En tant qu'Administrateur (Superviseur), je souhaite pouvoir gérer les travailleurs de toutes les entreprises enregistrées sur le portail afin de modifier ou supprimer ceux-ci.

[illegible]

Figure 17 - Gestion des travailleurs

Cet écran permet à l'administrateur de rechercher un travailleur, selon certains critères, parmi tous les travailleurs déclarés sur le portail.

Les critères sont les suivants :

- Le nom de l'entreprise
- Le nom du sous-traitant maître
- L'état de la demande
- Le numéro de référence du travailleur
- Le nom du travailleur
- Le document E101 ou Limosa est expiré

La liste des travailleurs sera paginée (exemple : 50 entreprises par page).

L'administrateur pourra aussi supprimer un ou plusieurs travailleurs en cliquant sur la case de la première colonne du travailleur ou sélectionner tous les travailleurs.

4. Technologies

4.1. Spring



« Spring est un framework Web libre pour construire et définir l'infrastructure d'une application java, dont il facilite le développement et les tests. Il est généralement utilisé pour le développement d'applications en entreprise.

Spring est considéré comme un conteneur dit « léger ». La raison de ce nommage est expliquée par Erik Gollot dans l'introduction du document Introduction au framework Spring :

« Spring est effectivement un conteneur dit « léger », c'est-à-dire une infrastructure similaire à un serveur d'applications J2EE. Il prend donc en charge la création d'objets et la mise en relation d'objet par l'intermédiaire d'un fichier de configuration qui décrit les objets à fabriquer et les relations de dépendances entre ces objets. Le gros avantage par rapport aux serveurs d'application est qu'avec Spring, les classes n'ont pas besoin d'implémenter une quelconque interface pour être prises en charge par le framework (au contraire des serveurs d'applications J2EE et des EJBs). C'est en ce sens que Spring est qualifié de conteneur « léger ». » »⁷

Les trois concepts clés de Spring sont les suivants :

1. L'inversion de contrôle est assurée de deux façons différentes :
 - 1.1. La recherche de dépendance : consiste pour un objet à interroger le conteneur, afin de trouver ses dépendances avec les autres objets. C'est un cas de fonctionnement similaire aux EJBs.
 - 1.2. L'injection de dépendances : cette injection peut être effectuée de trois manières possibles :
 - L'injection de dépendance via le constructeur.
 - L'injection de dépendance via les modificateurs (setters).
 - L'injection de dépendance via une interface.
2. La programmation orientée aspect.
3. Une couche d'abstraction.
 - 3.1. La couche d'abstraction permet d'intégrer d'autres frameworks et bibliothèques avec une plus grande facilité. Cela se fait par l'apport ou non de couches d'abstraction spécifiques à des frameworks particuliers. Il est ainsi possible d'intégrer un module d'envoi de mails plus facilement.

« Les deux premières sont les plus utilisées par Spring.

Ce framework, grâce à sa couche d'abstraction, ne concurrence pas d'autres frameworks dans une couche spécifique d'un modèle architectural Modèle-Vue-Contrôleur mais s'avère un framework multi-couches pouvant s'insérer au niveau de toutes les couches : modèle, vue et contrôleur. »⁷

⁷ [https://fr.wikipedia.org/wiki/Spring_\(framework\)](https://fr.wikipedia.org/wiki/Spring_(framework))

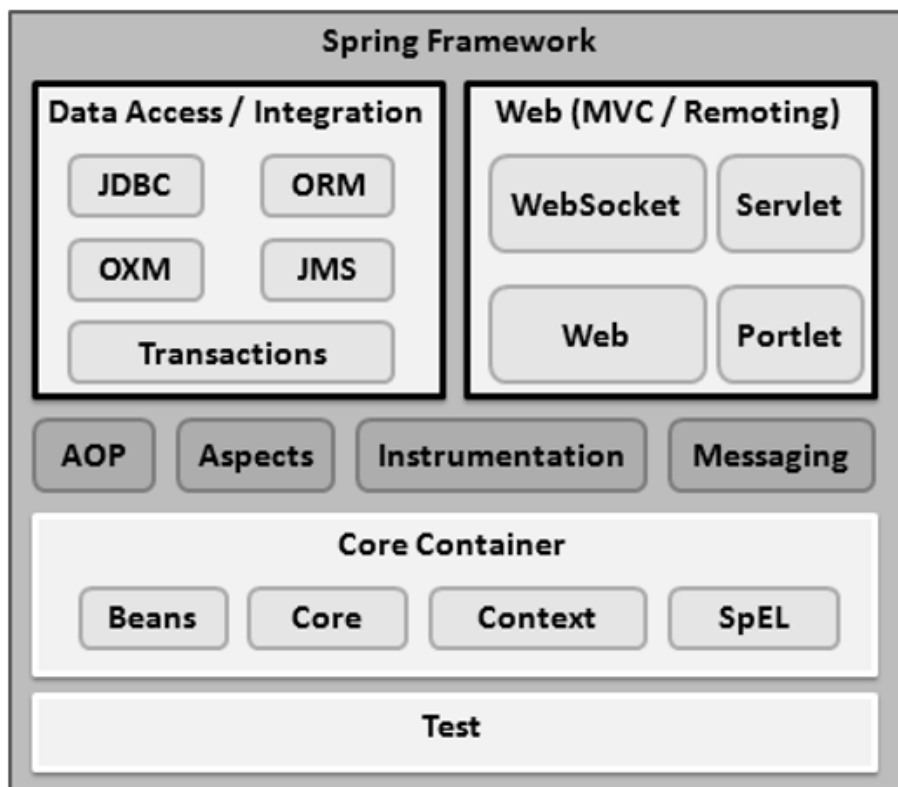


Figure 18 - Architecture Spring

Spring est modulaire, ce qui veut dire que l'on peut choisir les modules que l'on souhaite utiliser, sans devoir intégrer le reste au projet. Le framework **Spring** fournit environ 20 modules qui peuvent être utilisés en fonction de l'exigence de l'application.

J'ai utilisé **Spring** car il s'agit du framework utilisé par l'entreprise.

Pour apprendre **Spring**, mon maître de stage et un analyste de la société m'ont demandé de réaliser une application minimaliste de gestion d'une entreprise.

4.1.1. Comparaison Spring vs Symfony

« **Symfony** est un ensemble de composants PHP ainsi qu'un framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulables et adaptables qui permettent de faciliter et d'accélérer le développement d'un site web. »⁸

J'ai utilisé **Symfony** lors de mon projet de développement web de 3^{ème}. Je vais donc faire une comparaison de ce dernier avec le CMS spring :

	Spring	Symfony
Java	X	
PHP		X
Open-source	X	X
framework MVC	X	X
Entreprise	X	
Bonne documentation	X	
Doctrine		X
Freemarker / Mustache	X	
Facile à apprendre	X	X

⁸ <https://fr.wikipedia.org/wiki/Symfony>

Pour moi, J'ai une préférence pour **Spring** car

- Le framework est très complet.
- Il y a une meilleure documentation.
- On peut utiliser un langage comme freemarker ou mustache qui permet de faire des templates d'habillage qui est plus simple à utiliser pour les graphistes et permet de mieux partager le travail.
- Le langage de programmation est Java.

4.2. PostgreSQL



Figure 19 - Logo PostgreSQL

« **PostgreSQL** est un système de gestion de base de données relationnelles objet (ORDBMS). Il supporte une grande partie du standard SQL tout en offrant de nombreuses fonctionnalités modernes comme les clés étrangères, les requêtes complexes, etc. »⁹

PostgreSQL est le SGBD (système de gestion de base de données) principal de la société.

Je l'ai donc aussi utilisé pour la gestion de ma base de données.

4.3. Hibernate



Figure 20 - Logo Hibernate

« **Hibernate** est une solution open source de type ORM (Object Relational Mapping) qui facilite le développement de la couche persistante d'une application. »¹⁰

Hibernate permet donc de représenter les tuples de la base de données en objets Java et vice versa.

« **Hibernate** facilite la persistance et la recherche de données dans une base de données en réalisant lui-même la création des objets et les traitements de remplissage de ceux-ci en accédant à la base de données. »⁸

La quantité de code ainsi épargnée est très importante d'autant plus que ce code est généralement fastidieux et redondant.

⁹ <http://docs.postgresql.fr/9.4/pg94.pdf>

¹⁰ <https://www.jmdoudoux.fr/java/dej/chap-hibernate.htm>

J'ai utilisé **Hibernate** pour lire, ajouter, modifier et supprimer des éléments en base de données.

4.4. Freemarker

« **Freemarker** est un moteur de template ou gabarits qui permet de générer automatiquement des fichiers textes à partir d'un modèle de données de type tableau associatif. »¹¹

C'est une bibliothèque logicielle écrite en Java, souvent utilisée pour générer des pages Web ou du code source.

J'ai utilisé **Freemarker** comme moteur de template pour la réalisation de mes vues.

4.5. Bootstrap

« **Bootstrap** est une collection d'outils utile à la création du design (graphisme, animation et interactions avec la page dans le navigateur...) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. »¹²

Bootstrap permet de gérer automatiquement les adaptations d'interfaces sur un écran, une tablette ou un appareil mobile.

J'ai utilisé **Bootstrap** pour réaliser le design de mon portail.

4.6. JQuery

JQuery est une bibliothèque JavaScript rapide, libre, multi-plateforme et riche en fonctionnalités.

La bibliothèque contient notamment les fonctionnalités suivantes :

- Parcours et modification de documents HTML
- Manipulation des feuilles de style en cascade
- La gestion d'événements
- Les effets visuels et l'animation
- Ajax
- Fonctionne dans une multitude de navigateurs.

Ces fonctionnalités permettent de rendre les choses beaucoup plus simple et plus facile à utiliser.

J'ai utilisé **JQuery** lorsque je devais masquer ou afficher des éléments en fonction du choix de l'utilisateur ou bien lors de l'ajout de la photo du travailleur pour que celle-ci s'ajoute à la place de la photo par défaut.

¹¹ <https://fr.wikipedia.org/wiki/Freemarker>

¹² [https://fr.wikipedia.org/wiki/Bootstrap_\(framework\)](https://fr.wikipedia.org/wiki/Bootstrap_(framework))

5. Outils

5.1. Maven



Figure 21 - Logo Maven

« **Maven** est un outil pour la gestion et l'automatisation de production des projets logiciels Java en général et Java EE¹³ en particulier.

Celui-ci est géré par l'organisation Apache Software Foundation.

Il utilise un modèle de document XML connu sous le nom de Project Object Model (POM) afin de décrire un projet logiciel, ses dépendances avec des modules externes et l'ordre à suivre pour sa production. Il est livré avec un grand nombre de tâches prédéfinies, comme la compilation de code Java ou encore sa modularisation.

Un élément clé et relativement spécifique de **Maven** est son aptitude à fonctionner en réseau. Une des motivations historiques de cet outil est de fournir un moyen de synchroniser des projets indépendants : publication standardisée d'information, distribution automatique de modules jar. Ainsi en version de base, **Maven** peut dynamiquement télécharger du matériel sur des dépôts logiciels connus. Il propose ainsi la synchronisation transparente de modules nécessaires. »¹⁴

¹³ Java Enterprise Edition : spécification pour la technique Java d'Oracle plus particulièrement destinée aux applications d'entreprise

¹⁴ https://fr.wikipedia.org/wiki/Apache_Maven

5.1.1. Project Object Model (POM)

Chaque projet ou sous-projet est configuré par un POM qui contient les informations nécessaires à Maven pour traiter le projet (nom du projet, numéro de version, dépendances vers d'autres projets, bibliothèques nécessaires à la compilation, noms des contributeurs, etc.). Ce POM se matérialise par un fichier pom.xml à la racine du projet.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>PortailInscription</groupId>
  <artifactId>PortailInscription</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>
  <properties>
    <encoding>UTF-8</encoding>
    <spring.version>4.3.6.RELEASE</spring.version>
  </properties>
  <build>
  </build>
  <dependencies>
    <dependency>
      <groupId>org.freemarker</groupId>
      <artifactId>freemarker</artifactId>
      <version>2.3.23</version>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-test</artifactId>
      <version>${spring.version}</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>
```

Figure 22 - exemple POM.xml

5.2. PgAdmin

Pgadmin est la plate-forme d'administration et de développement Open Source, la plus populaire, pour PostgreSQL.

J'ai utilisé cet outil afin de créer ma base de données PostgreSQL et de visualiser les données de celle-ci.

5.3. Balsamiq

Balsamiq est l'outil que j'ai utilisé pour créer mes maquettes pour mon dossier d'analyse.

5.4. Modelio

Modelio est l'outil que j'ai utilisé pour créer mes diagrammes UML tels que les diagrammes de cas d'utilisation ou les diagrammes BPMN ¹⁵ (Business Process Model and Notation).

5.5. JMerise

J'ai utilisé **JMerise** afin de réaliser mon schéma de base de données.

¹⁵ BPMN veut dire en français « modèle de procédé d'affaire et notation » et c'est une méthode de modélisation de processus d'affaires pour décrire les flux d'activités et les procédures d'une organisation sous forme d'une représentation graphique standardisée. Plus d'informations : https://fr.wikipedia.org/wiki/Business_process_model_and_notation

5.6. Wamp Server

« **Wamp Server** est une plateforme de développement Web de type WAMP¹⁶ (Windows Apache MYSQL PHP). »¹⁷ Il permet de faire fonctionner localement mon projet avec Apache Tomcat¹⁸ et d'utiliser phpPgAdmin¹⁹ pour gérer la base de données PostgreSQL.

5.7. Qualité du code

5.7.1. Règle d'ergonomie

Les règles d'ergonomie expliquées ci-dessous sont issues des sites Web suivant : <https://www.nobilito.fr/2016/11/les-grandes-regles-dergonomie-de-votre-site-web/>, <http://ludismedia.com/12-regles-ergonomie-web/> et [http://www.infowebmaster.fr/116, news-optimiser-ergonomie-formulaires-web.html](http://www.infowebmaster.fr/116-news-optimiser-ergonomie-formulaires-web.html).

Ces règles permettent de faciliter l'utilisation du portail pour tous les utilisateurs sur tous types d'appareils.

5.7.1.1. Architecture de l'information

Les informations doivent être organisées afin que l'utilisateur puisse trouver facilement ce qu'il cherche.

- Grouper les contenus cohérents
- Mettre en évidence les contenus clés
- Créer un menu pour éviter de perdre l'internaute

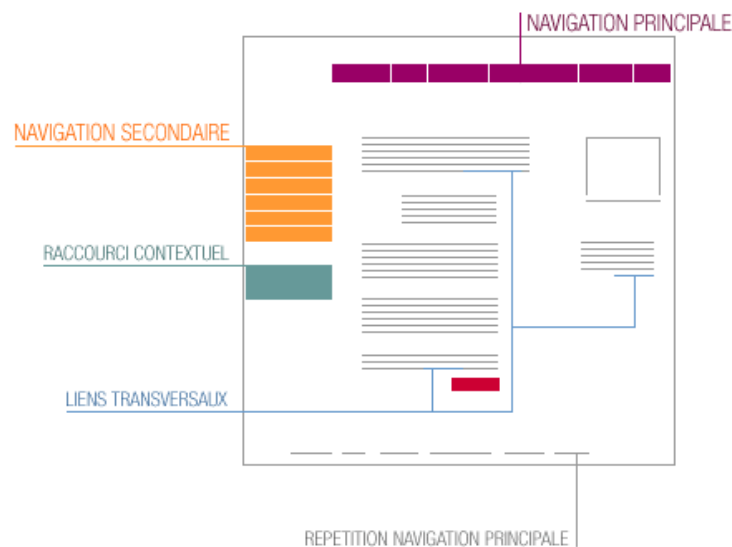


Figure 23 - Navigation site web

La figure 23 ci-dessus montre comment organiser la navigation sur un site.

5.7.1.2. Organisation visuelle

Il faut considérer une page Web comme un environnement à part entière.

- Ne jamais surcharger vos pages Web
- Hiérarchiser visuellement votre page Web pour la rendre compréhensible rapidement
- Privilégier les textes foncés sur fond blanc

¹⁶ <https://fr.wikipedia.org/wiki/WAMP>

¹⁷ <https://fr.wikipedia.org/wiki/WampServer>

¹⁸ https://fr.wikipedia.org/wiki/Apache_Tomcat

¹⁹ <https://fr.wikipedia.org/wiki/PhpPgAdmin>



Figure 24 - Organisation page web

La figure 24 ci-dessus montre comment représenter les éléments sur une page web. Il y a une zone header, une zone de contenu et une zone footer.

La zone header doit reprendre le logo à gauche pour que ce soit la première chose que le visiteur puisse trouver.

Dans la zone principale, on retrouve le plus souvent une colonne sur la gauche, car généralement, le visiteur regarde la page en commençant en haut à gauche et finissant en bas à droite ce qui permet d'avoir le contenu au centre.

Et la zone footer, on y retrouve les informations sur le site comme le copyright, l'année de mise en ligne.

5.7.1.3. Information

Le site réagit aux actions de l'utilisateur et l'informe de ce qu'il se passe au bon moment.

- Informer l'utilisateur d'où il se trouve
- Détailler les raisons et conséquences des actions de l'utilisateur
- Fournir une réponse suivant l'action d'un utilisateur
- Rendre les zones cliquables faciles à viser
- Éviter que l'utilisateur ait à revenir en arrière après une action

5.7.1.4. Gestion des erreurs

Anticiper les erreurs commises par l'utilisateur lors de la navigation Web.

- Repérer où l'erreur a été commise et indiquer lui où elle s'est produite
- Expliquer précisément l'erreur commise
- Proposer une solution pour corriger celle-ci

The image shows a web form with the following elements:

- A label "Email : *" above a text input field.
- The input field contains the text "test".
- A label "Adresse du siège social" below the email field.
- A label "Rue : *" below the social address field.
- An empty text input field for the street address.
- A validation error message box with a yellow warning icon and the text: "Veuillez inclure "@" dans l'adresse e-mail. Il manque un symbole "@" dans "test"."

Figure 25 - exemple gestion erreur

La figure 25 ci-dessus montre une vérification du champ (Email) se lance à la sortie du champ et un message s'affiche indiquant l'erreur si c'est le cas.

5.7.1.5. Convention

- Convention de localisation : les emplacements des éléments sur votre site (logo, navigation, panier)
- Convention de vocabulaire : la page d'accueil de votre site Web s'appelle « Accueil », le panier reste « panier », la page de contact « contact »
- Convention d'interaction et de présentation : format de présentation des formulaires, les alertes, tunnel d'achat et le changement d'état d'un bouton au survol de la souris signifiant la fonctionnalité cliquable

5.7.1.6. Liberté d'action

- Donner la possibilité de stopper une action : revenir en arrière, arrêter une animation.
- Éviter l'affichage de pop-up ou de publicité intrusive

5.7.1.7. Accessibilité

Le site doit être accessible à tout le monde. Il y a 2 types d'accessibilité Web :

- L'accessibilité physique : votre site Web doit être conçu de manière à permettre la navigation à partir de logiciels spécifiques aux déficients visuels

- L'accessibilité technologique : le contenu du site Web doit être conçu en responsive design, c'est-à-dire que le contenu s'adapte automatiquement à la résolution et la taille de l'écran

Accueil

Inscription d'une entreprise

Les champs obligatoires sont suivis d'un astérisque (*)

Numéro BCE : *

Nom de l'entreprise : *

Personne de contact

Nom : *

Prenom : *

Figure 26 - exemple responsive design

La figure 26 ci-dessus montre comment le contenu d'un site s'adapte automatiquement à la taille de l'écran (par exemple, ici, sur un appareil mobile) (voir figure 42 plus bas pour voir la différence sur un écran d'ordinateur).

5.7.1.8. Satisfaction

Toutes les règles participent à l'atteinte de la satisfaction, si elles sont réunies, les visiteurs seront satisfaits de la qualité d'utilisation du site.

5.7.1.9. Positionnement des éléments dans le formulaire

5.7.1.9.1. Position des champs du formulaire

Lister les champs les uns au-dessus des autres. De même il convient de placer le label du champ juste au-dessus de la zone à compléter. Un tel placement des éléments permet de scanner plus facilement et rapidement le formulaire pour les raisons suivantes :

- L'utilisateur voit clairement s'il a oublié de remplir des champs.
- L'œil n'a pas besoin de faire de zigzag.
- Ça respecte plus facilement la loi de proximité (cf. les éléments proches sont en rapport les uns des autres).

- Ça respecte mieux la loi de Fitts²⁰.

La figure 27 ci-dessous montre un mauvais exemple de formulaire :

Figure 27 - mauvais exemple formulaire

La figure 28 ci-dessous montre un bon exemple de formulaire :

Figure 28 - bon exemple formulaire

5.7.1.9.2. Position du bouton d'envoi

- Selon la structure du formulaire, il faut placer le bouton d'envoi en bas à gauche (1 colonne) ou en bas à droite (2 colonnes).
- Placer le bouton d'envoi en bas à droite si c'est un formulaire de plusieurs pages ou de plusieurs sections.

5.7.2. Règle de sécurité

Les 10 règles de sécurité OWASP (Open Web Application Security Project) expliquées ci-dessous proviennent de ce site web : https://www.owasp.org/index.php/Main_Page. Lorsque ces règles sont appliquées, elles permettent de sécuriser l'application contre toutes les attaques possibles.

5.7.2.1. Injection

L'injection SQL est une attaque en envoyant du code SQL dans les requêtes Web et ainsi prendre le contrôle de la base de données, même si celle-ci n'est pas connectée à internet.

Exemple :

SELECT * FROM User where id ='sdfssd' or '1'='1'

Solution : Utiliser des requêtes avec des paramètres

```
public Entreprise findEntrepriseByName(String nom) {
    TypedQuery<Entreprise> query = manager.createQuery("Select e From Entreprise e Where e.nom = :nom", Entreprise.class);
    query.setParameter("nom", nom);
    try {
        return query.getSingleResult();
    }
    catch (NoResultException e)
    {
        Log.info(e);
        throw new NoResultException(e.getMessage());
    }
}
```

²⁰ https://fr.wikipedia.org/wiki/Loi_de_Fitts

5.7.2.2. *Violation de gestion d'authentification et de session*

La gestion d'authentification et de session est souvent mal configurée ce qui permet aux attaquants d'exploiter ces failles afin d'usurper l'identité des utilisateurs.

Exemple : utiliser l'id de session dans l'url avec les données de connexion

5.7.2.3. *Exposition de données sensibles*

Certaines données sensibles ne sont pas protégées correctement (carte de crédit, identifiant d'impôt...). Pour éviter que celle-ci soit volée ou modifiée, elles méritent une protection supplémentaire (chiffrement statique ou en transit), ainsi que des précautions particulières lors de l'échange avec le navigateur.

Solution :

- Chiffrer les données sensibles
- Stocker les mots de passe au moyen d'un algorithme adapté
- Ne conserver que les données sensibles nécessaires
- Identifier les menaces contre lesquelles l'on souhaite se protéger

5.7.2.4. *Cross-site scripting (XSS)*

Les failles XSS se produisent lorsqu'une application accepte des données non fiables et les envoient sans les valider au préalable. Celles-ci permettent d'exécuter des scripts dans le navigateur pour détourner des sessions ou rediriger l'utilisateur vers des sites malveillants.

Exemple :

`www.exemple.com/index.html?name=guest<script>alert('attacked')</script>`

Solution : Pour éviter cette faille de sécurité, il faut échapper toutes les données non fiables basées sur le contexte HTML

5.7.2.5. *Références directes non sécurisées à un objet*

Cette faille se produit lorsqu'une référence à un objet d'exécution interne (fichier, dossier ou clé de base de données) est exposée et qu'il n'y a pas de contrôle d'accès ou de protection pour éviter que des personnes non autorisées accèdent à ces données.

Exemple :

```
String sqlquery = "SELECT * FROM useraccounts WHERE account = ?";
PreparedStatement st = connection.prepareStatement(sqlquery , ... );
st.setString( 1, request.getParameter("acct"));
ResultSet results = st.executeQuery( );

L'attaquant modifie la requête pour accéder aux informations de l'administrateur.
http://example.com/app/accountInfo?acct=admin
```

Solution :

- Implémenter des références indirectes, par utilisateur ou session : associer la référence indirecte par utilisateur à la valeur réelle de la clé sur le serveur.

Contrôler l'accès : vérifier que l'utilisateur qui accède à une entité non fiable a bien le droit d'accéder à l'objet demandé.

5.7.2.6. *Mauvaise configuration Sécurité*

Une bonne sécurité nécessite de disposer d'une configuration sécurisée définie et déployée pour l'application, contextes, serveur d'application, serveur Web, serveur de base de données et la plate-forme. Tous ces paramètres doivent être définis, mis en œuvre et maintenus, car beaucoup ne sont pas livrés sécurisés par défaut. Cela implique de tenir tous les logiciels à jour.

Exemple :

Le listage des répertoires est activé. L'attaquant le découvre et peut lister les répertoires et trouver les fichiers. L'attaquant trouve et télécharge vos classes java compilées qu'il décompile. Il identifie une faille de contrôle d'accès.

Solution :

Tous les environnements tels que Développement, QA et environnements de production devraient tous être configurés de manière identique en utilisant différents mots de passe utilisés dans chaque environnement qui ne peuvent pas être piratés facilement.

Assurez-vous qu'une architecture d'application forte est adoptée qui offre une séparation efficace et sécurisée entre les composants

Il peut également minimiser la possibilité de cette attaque en exécutant des analyses automatisées et en effectuant régulièrement des audits.

5.7.2.7. *Exposition de données sensibles*

Certaines données sensibles ne sont pas protégées correctement (carte de crédit, identifiant d'impôt...). Pour éviter que celle-ci soit volée ou modifiée, elles méritent une protection supplémentaire (chiffrement statique ou en transit), ainsi que des précautions particulières lors de l'échange avec le navigateur.

Solution :

- Chiffrer les données sensibles
- Stocker les mots de passe au moyen d'un algorithme adapté
- Ne conserver que les données sensibles nécessaires
- Identifier les menaces contre lesquelles l'on souhaite se protéger

5.7.2.8. *Manque de contrôle d'accès au niveau fonctionnel*

Vérifier les droits d'accès au niveau fonctionnel, mais aussi faire les mêmes vérifications de contrôle d'accès sur le serveur lors de l'accès à chaque fonction.

Exemple :

<http://example.com/app/getappInfo>

http://example.com/app/admin_getappInfo

Si un utilisateur non connecté peut accéder à la page d'administration, c'est un problème.

5.7.2.9. *Falsification de requête intersite (CSRF)*

Une attaque CSRF force le navigateur d'un utilisateur authentifié à envoyer une requête HTTP forgée à une application Web vulnérable. L'application vulnérable pense que ces requêtes émanent de l'utilisateur authentifié.

Exemple :

<http://example.com/app/transferFunds?amount=1500&destinationAccount=4673243243>

La requête de l'attaquant cachée dans une balise image :

```

```

Solution : Utiliser un jeton unique dans un champ caché, celui-ci sera envoyé dans le corps de la requête HTTP.

5.7.2.10. Utilisation de composants avec des vulnérabilités connues

Les composants vulnérables, tels que bibliothèques, contextes et autres modules logiciels fonctionnent presque toujours avec des privilèges maximums. Ainsi, si exploités, ils peuvent causer des pertes de données sérieuses ou une prise de contrôle du serveur. Les applications utilisant ces composants vulnérables peuvent compromettre leurs défenses et permettre une série d'attaques et d'impacts potentiels.

Exemple :

Les deux composants vulnérables suivants ont été téléchargés 22 millions de fois en 2011.

- *Apache CXF Authentication Bypass* - En ne fournissant pas de jeton d'authentification, les attaquants pouvaient faire appel à n'importe quels Web services avec l'ensemble des privilèges.
- *Spring Remote Code Execution* - Un abus de l'implémentation du langage d'expression de Spring permettait aux attaquants d'exécuter du code arbitraire et ainsi de prendre le contrôle du serveur.

Solution :

- Identifiez tous les composants et les versions qui sont utilisées dans les webapps et ne se limitent pas uniquement à la base de données / frameworks.
- Tenir à jour tous les composants tels que les bases de données publiques, les listes de diffusion des projets.
- Il est important d'ajouter des enveloppes de sécurité autour de composants de nature vulnérable.

5.7.2.11. Redirections et renvois non validés

Les applications Web réorientent et redirigent fréquemment les utilisateurs vers d'autres pages et sites internet, et utilisent des données non fiables pour déterminer les pages de destination. Sans validation appropriée, les attaquants peuvent réorienter les victimes vers des sites de phishing ou de malware, ou utiliser les renvois pour accéder à des pages non autorisées.

Exemple :

Une application possède une page « `redirect.jsp` » disposant d'un seul paramètre nommé « `url` ». Un attaquant forge une URL permettant de rediriger les utilisateurs vers un site malveillant.

<http://www.example.com/redirect.jsp?url=evil.com>

Solution :

- Éviter l'utilisation des redirections et des renvois.
- En cas d'utilisation, ne pas utiliser de valeur de destination dans les paramètres utilisateur. Ceci est généralement réalisable.
- Si une valeur de destination doit être spécifiée, vérifier que la valeur est valide et autorisée pour l'utilisateur.

5.7.3. Performance

Côté performance, j'ai dû essayer de réduire au maximum le nombre de requêtes faites à la base de données afin de ne pas ralentir inutilement le portail web.

Ensuite, j'ai ajouté une métadonnée dans mon code HTML qui permet de détecter quel périphérique l'utilisateur utilise et ainsi, **bootstrap** pourra gérer l'affichage en conséquence.

```
<meta content="width=device-width, initial-scale=1.0" name="viewport" />
```

Figure 29 – métadonnée HTML

J'ai aussi vérifié la longueur et la largeur de l'image ajoutée pour un travailleur afin d'empêcher l'utilisateur d'ajouter une image trop grande et ainsi causer des problèmes de lenteur sur le serveur où est déployé le portail.

```
int height = 0;
int width = 0;
try {
    Metadata metadata = ImageMetadataReader.readMetadata(photo);
    JpegDirectory directory = metadata.getFirstDirectoryOfType(JpegDirectory.class);
    height = directory.getImageHeight();
    width = directory.getImageWidth();
} catch (ImageProcessingException e) {
    Log.info(e);
    throw new IllegalArgumentException(e.getMessage());
}
```

Figure 30 - Vérification taille image travailleur

6. Implémentation du projet

6.1. Accueil

Cet écran représente la première page du portail, la page d'accueil. Sur celle-ci, l'utilisateur pourra inscrire une entreprise via le bouton « inscription » (voir figure 36) et se connecter via le bouton « Connexion » (voir figure 33).

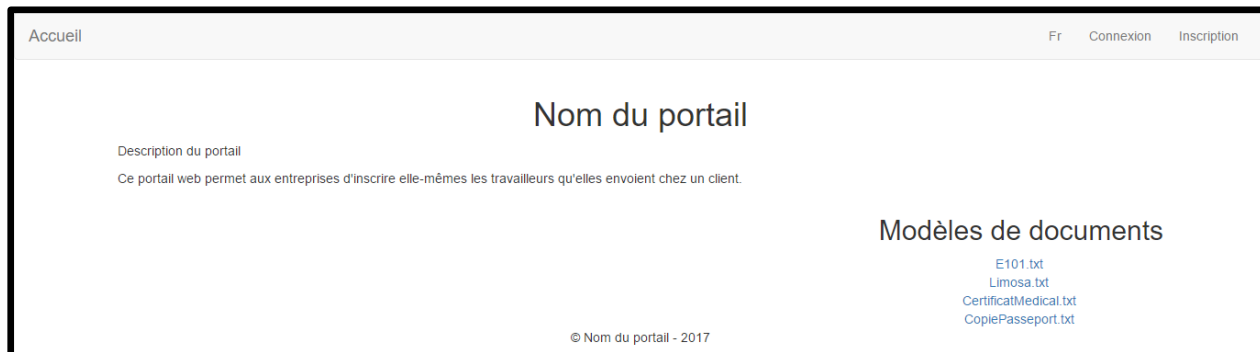


Figure 31 - Page d'accueil

L'utilisateur trouvera sur la droite des modèles de documents à télécharger et ajouter pour chaque travailleur.

Lorsque le responsable de l'entreprise est connecté, la page d'accueil ressemblera à ceci :

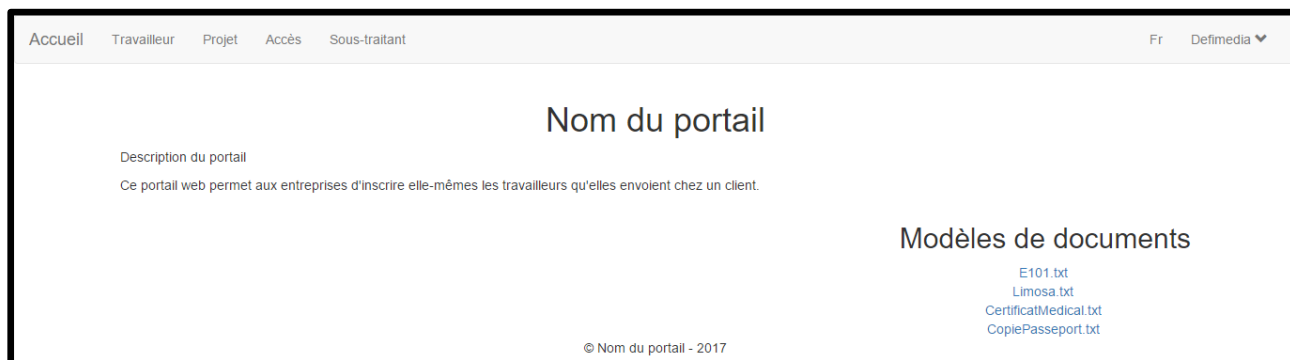


Figure 32 - Page d'accueil connecté

Le responsable peut ainsi via le menu de navigation : ajouter ses travailleurs, ajouter un projet, ajouter un accès pour un travailleur pour un certain projet et ajouter une entreprise sous-traitante.

Celui-ci peut également visualiser son profil et se déconnecter en cliquant sur le nom de sa société en haut à droite.

6.2. Connexion

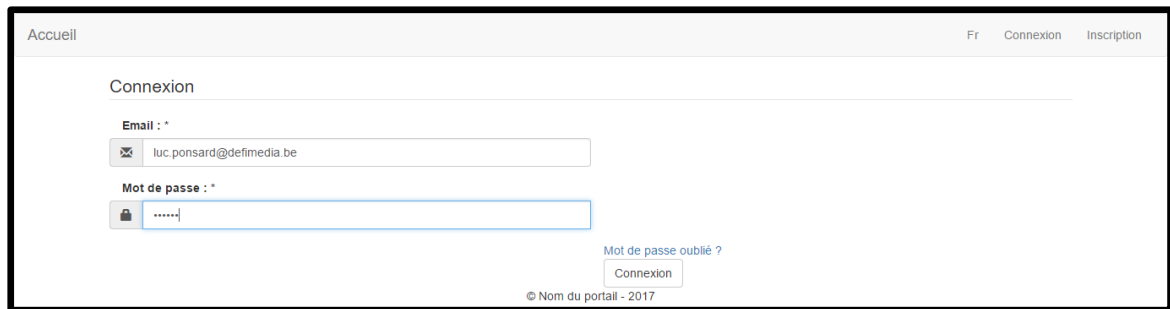


Figure 33 – Connexion au portail

Cet écran permet à l'utilisateur (responsable de l'entreprise) de se connecter au portail.

Pour se connecter, le responsable entrera son adresse email et son mot de passe encodés lors de l'inscription.

Si un utilisateur souhaite accéder à une page du portail, par exemple, la page d'administration, en écrivant l'url de celle-ci et que l'utilisateur n'a pas les droits d'accès ou n'est pas connecté. L'utilisateur sera redirigé vers cette page de connexion.

Afin de respecter la règle de sécurité OWASP n°7 : Manque de contrôle d'accès au niveau fonctionnel expliqué plus haut (voir 5.7.2.8) dans mon projet, j'ai utilisé un sous-projet de Spring qui s'appelle Spring Security.



Figure 34 - Logo Spring Security

Spring Security est un framework d'authentification et de contrôle d'accès.

Les deux concepts clés sont :

- L'authentification consiste à garantir que la personne connectée est bien celle qu'elle prétend être.
- Les autorisations consistent à vérifier que la personne connectée a bien les permissions d'effectuer une action donnée ou d'accéder à une ressource.

Il y a deux moyens pour configurer Spring Security via une classe en java ou sous forme de bean en XML.

Voici le code XML de mon bean : spring-security.xml

Ce code permet que si l'utilisateur accède à l'adresse suivante : <http://www.example.com/travailleur/ajoutTravailleur.html> mais que celui-ci n'a pas les droits nécessaires, il sera redirigé vers la page de connexion (connexion.html) précisée entre la balise *form-login*.

```
<http security="none" pattern="/resources/**"/>
<http use-expressions="true">
  <intercept-url pattern="/travailleur/*" access="hasRole('ROLE_SUP_1')"/>
  <intercept-url pattern="/projet/*" access="hasRole('ROLE_SUP_1')"/>
  <intercept-url pattern="/acces/*" access="hasRole('ROLE_SUP_1')"/>
  <access-denied-handler error-page="/403.html"/>
  <form-login
    login-page="/connexion.html"
    authentication-failure-url="/connexion.html"
    username-parameter="username"
    password-parameter="password"
  />
  <logout delete-cookies="JSESSIONID" logout-url="/deconnexion.html" logout-success-url="/index.html" />
</http>
<csrf/>
</authentication-manager>
  <authentication-provider user-service-ref="userDetailsService">
    <password-encoder hash="bcrypt"/>
  </authentication-provider>
</authentication-manager>
```

Figure 35 - code spring-security.xml

La balise *csrf* permet d'activer la protection csrf sur le site web.

La balise *logout* permet de définir la page de déconnexion ainsi que la page de redirection lorsque l'utilisateur est bien déconnecté.

La balise *authentication-manager* permet de spécifier le service de gestion des authentifications et des rôles pour accéder au site web.

La balise *password-encoder* signifie que l'algorithme de cryptage du mot de passe est Bcrypt.

6.3. Inscription

Cet écran permet au responsable de l'entreprise de faire une demande d'inscription pour son entreprise sur le portail.

Accueil

FrConnexionInscription

Inscription d'une entreprise

Les champs obligatoires sont suivis d'un astérisque (*)

Numéro BCE : *

BE044668905

Nom de l'entreprise : *

Defimedia

Personne de contact

Nom : *

Ponsard


Prenom : *

Luc

GSM/Telephone : *

0474488656

Email : *

 luc.ponsard@defimedia.be

Adresse du siège social

Rue : *

Phocas Lejeune

Numéro : *

32

Code postal : *

5032

Localité : *

Gembloux

Pays : *

Belgique

Type : *

☒ Société sous-traitante maître ☐ Société sous-traitante d'une société maître

Raison de la demande : *

Maintenance d'un serveur

Numéro de Référence : *

d4547d8s955a

Remarques :

Documents

Bon de commande : *

Choisissez un fichier

ben_commande.txt

Statut de la société : *

Choisissez un fichier

statut_societe.txt

Mot de passe : *

.....

Confirmation mot de passe : *

.....

Ajouter

© Nom du portail - 2017

Figure 36 - Inscription entreprise

Le responsable remplit le formulaire contenant les données suivantes :

- Numéro BCE de l'entreprise
- Le nom de l'entreprise
- Les informations de la personne de contact
 - Nom
 - Prénom
 - GSM/Telephone
 - Email
- L'adresse du siège social
 - Rue
 - Numéro
 - Code postal
 - Localité
 - Pays
- Le type d'entreprise
- La raison de la demande d'inscription
- Le numéro de référence du projet à réaliser
- Les remarques (optionnel)
- Les documents concernant la société : bon de commande reçu du client et le statut de la société
- Le mot de passe

Quand il aura encodé toutes les informations du formulaire, le responsable cliquera sur le bouton ajouter pour s'inscrire et il pourra ensuite se connecter au portail (voir figure 33).

Les données de l'entreprise seront stockées dans une table de la base de données. Cette table s'appellera « Entreprise » (voir ci-contre). Celle-ci sera liée à un utilisateur qui sera le responsable ayant encodé l'entreprise.

L'état actuel de la demande et la date de création seront aussi insérés.

La description de la table ci-contre est expliquée en annexe.

Entreprise		
PrK	Id	Auto_increment
	NumeroBCE	Varchar
	Nom	Varchar
	NomContact	Varchar
	PrenomContact	Varchar
	TelContact	Varchar
	EmailContact	Varchar
	Rue	Varchar
	Numero	Varchar
	CodePostal	Varchar
	Localite	Varchar
	Pays	Varchar
	Etat	Varchar
	Type	Varchar
	RaisonDemande	Varchar
	NumeroReference	Varchar
	Remarque	Varchar
	NumeroBCEMaitre	Varchar
	DateCreation	Datetime
	DateModification	Datetime
FrK	Id_Utilisateur	Int

Figure 37 - Table entreprise

Pour éviter d'exposer les données sensibles comme le mot de passe et respecter la règle de sécurité OWASP n°6 : Exposition de données sensibles (voir plus haut 5.7.2.7).

J'ai utilisé l'algorithme BCryptPasswordEncoder²¹ pour crypter le mot de passe encodé pour le responsable.

```
Role role = roleDao.findRoleByName("ROLE_SUP_1");
PasswordEncoder passwordEncoder = new BCryptPasswordEncoder();
user = new User();
user.setMotPasse(passwordEncoder.encode(entrepriseForm.getMotPasse()));
user.setRole(role);
user.setEnabled(true);
```

Figure 38 - Utilisation BCryptPasswordEncoder

6.4. Profil de l'entreprise

Cet écran affiche les informations de l'entreprise.

Accueil Travailleur Projet Accès Sous-traitant Fr Defimedia

Espace Entreprise

Fiche détaillée d'une entreprise

Profil

Defimedia

Numéro BCE : BE044668905

Personne de contact
Nom : Ponsard
Prénom : Luc
Tel / Gsm : 0474488656
Email : luc.ponsard@defimedia.be

Adresse du siège social
Rue de Mulhouse, 36
4 020 Liège
Belgique

Type : Société sous-traitante maître
Etat : **En attente**
Raison de la demande : Maintenance d'un serveur
Numéro de référence : 5d5fs0fs4ds8s

Documents
Bon de commande : bon_commande.txt
Statut de la société : statut_societe.txt

Modifier **Supprimer**

© Nom du portail - 2017

Figure 39 - Fiche détaillée de l'entreprise

Le responsable de l'entreprise peut modifier (bouton « modifier ») ou supprimer (bouton « supprimer ») son entreprise.

6.5. Liste des travailleurs

Cet écran liste les travailleurs de l'entreprise connectée ainsi que les travailleurs de l'entreprise sous-traitante à celle-ci.

Accueil Travailleur Projet Accès Sous-traitant Fr Defimedia

Espace Entreprise

Mes travailleurs

Ajouter

Afficher 10 éléments Rechercher :

#	Nom	Prenom	Nationalité	Langue	Adresse email
	Ratz	David	Belge	FR	david.ratz@hotmail.com

Affichage de l'élément 1 à 1 sur 1 éléments

Précédent 1 Suivant

© Nom du portail - 2017

Figure 40 - Liste travailleurs

²¹ plus d'informations sur PasswordEncoder : <http://docs.spring.io/spring-security/site/docs/current/apidocs/org/springframework/security/crypto/password/PasswordEncoder.html>

Le responsable, une fois connecté, pourra ajouter un travailleur en cliquant sur le bouton « ajouter » via l'écran ci-dessus et sera redirigé vers la page d'ajout d'un travailleur (voir figure 41).

6.5.1. Différence par rapport à la maquette du point 3.3.2.1

À l'issue d'une réunion avec un analyste de la société, nous avons décidé de déplacer la liste des travailleurs qui se situait sur la page de profil de l'entreprise sur une autre page afin d'épurer celle-ci.

6.6. Inscription et modification d'un travailleur

Cet écran permet au responsable de l'entreprise d'inscrire un travailleur.

The screenshot shows a web application interface titled 'Espace Entreprise' with a subtitle 'Ajout d'un travailleur'. At the top, there is a navigation bar with links: 'Accueil', 'Travailleur', 'Projet', 'Accès', 'Sous-traitant', and a user profile 'Fr Defmedia'. A message states: 'Les champs obligatoires sont suivis d'un astérisque (*)'. The form is divided into several sections:

- Informations personnelles:** Includes a photo upload button 'Ajouter une photo'. Fields for 'Nom : *' (Ratz), 'Prénom : *' (David), 'Date de naissance : *' (20/04/1995), 'Nationalité : *' (Belge), 'Langue : *' (FR), 'Gem : *' (0493130801), and 'Email : *' (david.ratz@hotmail.com).
- Documents:** Contains two sections: 'E101' and 'Limosa'. Each section has a 'Fichier : *' field with a file selection button and a 'Date de validité : *' field set to 19/05/2017.
- Certificat Medical:** Similar to the documents section, with a 'Fichier : *' field and a 'Date de validité : *' field set to 19/05/2017.
- Copie Passeport:** Similar to the documents section, with a 'Fichier : *' field and a 'Date de validité : *' field set to 19/05/2017.

At the bottom right, there is a green 'Ajouter' button and a footer '© Nom du portail - 2017'.

Figure 41 - Inscription modification travailleur

Sur l'ensemble des pages de mon portail, j'ai respecté au mieux les règles d'ergonomie, plus précisément, celle sur les formulaires (voir le point 5.7.1.9 plus haut).

Le responsable encodera les informations personnelles du travailleur suivantes :

- La photo
- Le nom
- Le prénom
- La date de naissance
- La nationalité
- La langue
- Le numéro de GSM
- L'adresse email

Il ajoutera ensuite les différents documents (E101, Limosa, certificat médical et copie de passeport) remplis pour le travailleur d'après les modèles téléchargés sur la page d'accueil. Il encodera pour chaque document la date de validité de celui-ci.

Le code ci-dessous montre la méthode d'ajout d'un travailleur en base de données.

```
@RequestMapping(value = "/ajouterTravailleur", method = RequestMethod.POST)
@Transactional(rollbackFor=Exception.class)
public String add(@RequestParam(name = "id", required = true) Integer id, @RequestParam("photoFichier") MultipartFile photoFichier,
    @RequestParam("file") MultipartFile[] files, @Valid @ModelAttribute("travailleurForm") TravailleurForm travailleurForm,
    BindingResult bindingResultTravailleurForm, @Valid @ModelAttribute("travailleur") Travailleur travailleur,
    BindingResult bindingResultTravailleur, Model model, HttpServletRequest request) throws IOException, MetadataException {
    Travailleur travailleur2;
    if(bindingResultTravailleurForm.hasErrors() || bindingResultTravailleurForm.hasErrors()) {
        model.addAttribute(LISTELANGUE, listLangue);
        model.addAttribute(LISTENATIONALITE, listNationalite);
        return "travailleur/ajouterTravailleur";
    }
    else if(uploadFile.imageInvalide(request, travailleurForm.getPhotoFichier(), travailleur.getId())) {
        model.addAttribute(LISTELANGUE, listLangue);
        model.addAttribute(LISTENATIONALITE, listNationalite);
        model.addAttribute("travailleurForm", new TravailleurForm());
        model.addAttribute("errorImage", "La taille de l'image est trop grande");
        return "travailleur/ajouterTravailleur";
    }
    else {
        try {
            entreprise = entrepriseDao.find(id);
            travailleur2 = new Travailleur(travailleur);
            travailleur2.setPhoto(travailleurForm.getPhotoFichier().getOriginalFilename());
            dateNaissance = dtl.parse(travailleurForm.getDateNaissance());
            travailleur2.setDateNaissance(dateNaissance);
            travailleur2.setEntreprise(entreprise);
            travailleur2.setDateCreation(dateTime.parse(dateTime.format(new Date())));
        } catch (ParseException e) {
            Log.info(e.getMessage());
            throw new IllegalArgumentException(e.getMessage());
        }
        uploadFile.uploadFiles(request, travailleurForm, travailleur.getId(), listFichier, travailleur2, documentDao, travailleurDao,
            travailleurDao.insert(travailleur2);
        return REDIRECTLISTETRAVAILLEURS;
    }
}
```

Figure 42 - code ajout travailleur base données

Une fois que le responsable a cliqué sur le bouton « Ajouter », cette méthode récupère toutes les informations du formulaire. Ensuite, elle ajoute ces informations dans un objet « Travailleur ». Puis, elle ajoute les documents du travailleur à la racine du projet (voir figure 43 ci-dessous) et enfin, elle ajoute le travailleur en base de données.

```
String idTravailleurCrypte = cipherUtil.encrypt(String.valueOf(idTravailleur));
String rootPath = request.getServletContext().getRealPath("/resources");
File dir = new File(rootPath + File.separator + "fichiers" + File.separator + idTravailleurCrypte);
```

Figure 43 - code chemin documents travailleur

Ici, j'ai utilisé l'algorithme de cryptage AES²² (Advanced Encryption Standard) (voir figure 44 ci-dessous) pour crypter le nom du dossier qui contiendra les documents du travailleur afin d'éviter qu'une personne malveillante accède à ce dossier en récupérant la structure de l'url du portail.

```
public class CipherUtilSecret {

    public static final String CIPHER_ALGORITHM = "AES";
    public static final String KEY_ALGORITHM = "AES";
    public static final byte[] SECRET_KEY = "DMOMVPRT5SJ7RANE" .getBytes(Charsets.UTF_8);

    public String decrypt(String encryptedInput) {
        try {
            Cipher cipher = Cipher.getInstance(CIPHER_ALGORITHM);
            cipher.init(Cipher.DECRYPT_MODE, new SecretKeySpec(SECRET_KEY, KEY_ALGORITHM));
            return new String(cipher.doFinal(Base64.getUrlDecoder().decode(encryptedInput)), Charsets.UTF_8);
        } catch (Exception e) {
            throw new RuntimeException(e);
        }
    }

    public String encrypt(String str) {
        try {
            Cipher cipher = Cipher.getInstance(CIPHER_ALGORITHM);
            cipher.init(Cipher.ENCRYPT_MODE, new SecretKeySpec(SECRET_KEY, KEY_ALGORITHM));
            return Base64.getUrlEncoder().encodeToString(cipher.doFinal(str.getBytes(Charsets.UTF_8)));
        } catch (Exception e) {
            throw new RuntimeException(e);
        }
    }
}
```

Figure 44 - Algorithme AES

Les données du travailleur seront stockées dans une table de la base de données. Cette table s'appellera « Travailleur » (voir ci-contre). Ce travailleur sera lié à une entreprise qui sera l'entreprise connectée. Le travailleur sera aussi lié à une liste noire si les dates de validités de ces documents sont dépassées.

La date de création sera aussi insérée.

La description de la table ci-contre est expliquée en annexe.

Travailleur		
PrK	Id	Auto_increment
	NumeroReference	Varchar
	Nom	Varchar
	Prenom	Varchar
	DateNaissance	Date
	Photo	Blob
	Nationalite	Varchar
	Langue	Varchar
	Telephone	Varchar
	AdresseEmail	Varchar
	DateCreation	Datetime
	DateModification	Datetime
FrK	Id_ListeNoire	Int
FrK	Id_Entreprise	Int

Figure 45 - Table travailleur

²² AES : <https://www.aescrypt.com/>

6.7. Affichage d'un travailleur

Cet écran affiche les informations du travailleur sélectionné via la liste des travailleurs (voir figure 40).

The screenshot shows the 'Espace Entreprise' interface. At the top, there is a navigation bar with links: Accueil, Travailleur, Projet, Accès, Sous-traitant. The main title is 'Espace Entreprise'. Below it, the section is 'Fiche détaillée d'un travailleur'. Under 'Informations personnelles', there is a profile picture of a man with glasses. To the right of the picture, the following information is listed:

Entreprise :	Defmedia
Nom :	Ratz
Prenom :	David
Date de naissance :	30 avr. 1995
Nationalité :	Belge
Langue :	FR
Email :	david.ratz@hotmail.com
Tel / Gsm :	0493138881

Below the personal information, there is a 'Documents' section with four entries:

- E101**
Fichier : e101.txt
Date de validité : 19 mai 2017
- Limosa**
Fichier : Limosa.txt
Date de validité : 19 mai 2017
- Certificat medical**
Fichier : certificat_medical.txt
Date de validité : 19 mai 2017
- Copie passeport**
Fichier : copie_passeport.txt
Date de validité : 19 mai 2017

Below the documents, there is a 'Les accès du travailleur' section. It contains a table with columns: Nom du projet, Date de début, Date de fin. Below the table, it says: 'Date et heure de création : 17 mai 2017 00:00:00' and 'Utilisateur ayant effectué la dernière modification : Ponsard Luc'. At the bottom, there are two buttons: 'Modifier' (orange) and 'Supprimer' (red). The footer says '© Nom du portail - 2017'.

Figure 46 - Fiche détaillée travailleur

6.8. Liste des projets

Cet écran liste les projets de l'entreprise connectée.

The screenshot shows the 'Espace Entreprise' interface. At the top, there is a navigation bar with links: Accueil, Travailleur, Projet, Accès, Sous-traitant. The main title is 'Espace Entreprise'. Below it, the section is 'Mes projets'. There is a green 'Ajouter' button. Below the button, there is a dropdown menu for 'Afficher' with '10' selected and the text 'éléments'. To the right, there is a search bar labeled 'Rechercher :'. Below the search bar, there is a table with columns: Nom, Rue, Numero, Code Postal, Localite, Pays. The table has one row of data:

Nom	Rue	Numero	Code Postal	Localite	Pays
Maintenance du serveur A	de Mulhouse	36	4 020	Liège	Belgique

Below the table, it says 'Affichage de l'élément 1 à 1 sur 1 éléments'. At the bottom, there are three buttons: 'Précédent', '1', and 'Suivant'. The footer says '© Nom du portail - 2017'.

Figure 47 - Liste projet

Le responsable peut ajouter un projet reçu par un client en cliquant sur le bouton « Ajouter » (voir figure 48).

6.8.1. Différence avec le dossier d'analyse

Au départ, il n'était pas prévu dans l'offre d'ajouter la gestion des projets. Nous avons décidé, mon maître de stage et moi, de l'ajouter comme ça le responsable de l'entreprise peut ajouter le projet qu'elle va réaliser et celle-ci pourra ainsi visualiser ces projets.

6.9. Ajout et modification d'un projet

Cet écran permet d'ajouter un projet pour l'entreprise connectée.

Accueil Travailleur Projet Accès Sous-traitant Fr Defimedia ▼

Espace Entreprise

Ajouter projet pour une entreprise

Les champs obligatoires sont suivis d'un astérisque (*)

Nom : *
Maintenance du serveur A

Type : *
Maintenance ▼

Rue : *
de Mulhouse

Numéro : *
36

Code postal : *
4020

Localité : *
Liège

Pays : *
Belgique ▼

Ajouter

© Nom du portail - 2017

Figure 48 - Ajout modification projet

Cet écran sera le même lors de la modification du projet.

Dans tous les formulaires de mon application, j'ai utilisé un jeton unique (voir figure 49 ci-dessous) afin de respecter la règle de sécurité n°8 (voir le point 5.7.2.9 plus haut) concernant la faille CSRF.

```
<input type="hidden" name="{_csrf.parameterName}"  
value="{_csrf.token}" />
```

Figure 49 - Jeton unique csrf

6.10. Fiche détaillée d'un projet

Cet écran permet de visualiser la fiche détaillée d'un projet choisi parmi la liste (voir figure 47).

Accueil Travailleur Projet Accès Sous-traitant Fr Defimedia ▼

Espace Entreprise

Fiche détaillée du projet de l'entreprise Defimedia

Nom : Maintenance du serveur A

Type : Maintenance

Rue de Mulhouse, 36

4 020 Liège

Belgique

Modifier Supprimer

© Nom du portail - 2017

Figure 50 - Fiche détaillée projet

Le responsable pourra modifier (même écran que figure 47) ou supprimer le projet en cliquant sur le bouton « Modifier » ou « Supprimer » prévu à cet effet.

6.11. Liste des accès

Cet écran permet au responsable d'ajouter une demande d'accès pour un de ses travailleurs pour projet.



Figure 51 - Liste accès

Le responsable pourra ajouter un accès en cliquant sur le bouton « Ajouter » et sera redirigé vers la page d'ajout d'une demande d'accès (voir figure 52 ci-dessous).

6.11.1. Différence dossier d'analyse

Avec l'ajout de la gestion de projets, nous avons décidé, mon maitre de stage et moi, d'ajouter la gestion des accès pour les travailleurs à un projet. Nous avons ajouté ceci pour que le responsable puisse voir si ces travailleurs ont bien accès à un chantier.

6.12. Ajout et modification d'un accès

Cet écran permet d'ajouter une demande d'accès pour un travailleur pour un certain projet.

Figure 52 – Ajout modification accès

Le responsable choisira le travailleur et le projet dont il veut faire une demande. Il encodera aussi les dates de début et de fin de l'accès qui font référence à la durée du projet.

Si le responsable ne trouve pas son travailleur dans la liste déroulante, c'est que celui-ci se trouve est en liste noire. La liste noire veut dire que les dates de validités de ces documents sont dépassées.

La demande sera acceptée et validée par le responsable de la société *Dinec International*.

Une fois la demande acceptée, le responsable pourra voir si son travailleur à un accès ou non via la liste de ses travailleurs.

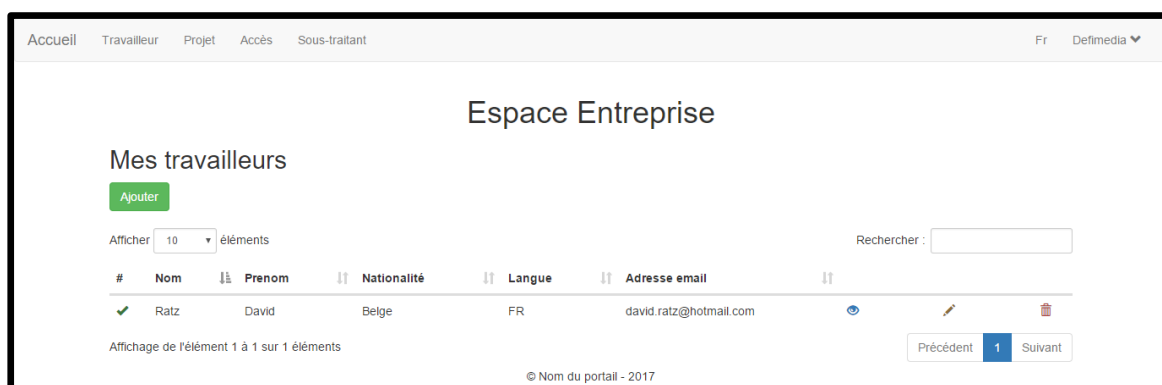


Figure 53 - Accès travailleur

Si le travailleur a un « V » sur sa ligne (voir figure 53 ci-dessus) alors celui-ci à un accès pour un projet.

6.13. Affichage d'un accès

Cet écran permet de visualiser l'accès d'un travailleur pour un projet.



Figure 54 - Fiche accès

Le responsable pourra modifier (même écran que figure 52) ou supprimer l'accès d'un travailleur pour un projet en cliquant sur le bouton « Modifier » ou « Supprimer » prévu à cet effet.

6.14. Ajout d'une entreprise sous-traitante

L'écran sera le même que celui de l'inscription d'une entreprise.

Lorsqu'on choisira comme type d'entreprise « Société sous-traitante d'une société maître », le responsable devra encoder le numéro de TVA de l'entreprise maître.

The form is titled 'Type : *'. It has two radio buttons: 'Société sous-traitante maître' (unselected) and 'Société sous-traitante d'une société maître' (selected). Below the radio buttons is a label 'Numéro BCE de la société maître : *' and a text input field containing the value 'BE044668905'.

Figure 55 - type entreprise

6.15. Phase de tests

À la suite de mon développement, j'ai décidé de tester mon application afin de pouvoir déployer mon portail sur le serveur de développement de l'entreprise et de faire une présentation au client sans risque de gros problèmes.

J'ai effectué des tests du côté serveur et j'ai utilisé **MockMVC**²³ qui est le point d'entrée principal pour le support de test pour Spring MVC côté serveur.

Voici la configuration indispensable pour exécuter un test en Spring ci-dessous.

```
@RunWith(SpringJUnit4ClassRunner.class)
@WebAppConfiguration
@ContextConfiguration(locations = {"classpath:applicationContext.xml", "file:WebContent/WEB-INF/spring-security.xml"})
@FixMethodOrder(MethodSorters.NAME_ASCENDING)
public class EntrepriseControllerTest{

    private MockMvc mockMvc;

    @Autowired
    private WebApplicationContext wac;

    @Before
    public void setup() {

        mockMvc = MockMvcBuilders.webAppContextSetup(this.wac).apply(springSecurity()).build();
    }
}
```

Figure 56 - Configuration test Spring

Il faut le contexte de l'application Spring ainsi que l'appelle à une méthode *springSecurity()* afin de configurer Spring security.

Un test se déroule en 2 étapes :

- La création de la requête HTTP simulée (URL, les en-têtes, paramètres, etc.)
- La vérification de la réponse HTTP (type de contenu, le contenu, etc.)

Le test suivant (voir figure 57 page suivante) permet de vérifier que lorsque l'utilisateur a cliqué sur le bouton « Ajouter » pour inscrire une entreprise (voir Figure 36), celui-ci est bien redirigé vers la page d'accueil.

²³Plus d'informations : <https://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/test/web/servlet/MockMvc.html>


```

@Test
public void ajoutPostEntreprisePage() throws Exception {
    entreprise = new Entreprise("BE123456789", "Defimedia", "Ratz", "lol", "0493138881", "k.ratz@gmail.com", "de mulhouse", "36",
        4000, "Liege", "Belgique", null, "Société sous-traitante maître", "rkrkrk", "5e54r4z1a2", null, null);
    EntrepriseForm entrepriseForm = new EntrepriseForm();
    entrepriseForm.setBonCommande("test.txt");
    entrepriseForm.setStatutSociete("lol.txt");
    entrepriseForm.setMotPasse("azerty");
    entrepriseForm.setConfirmation("azerty");

    ResultActions resultActions = mockMvc.perform(post("/entreprise/ajouterEntreprise")
        .with(csrf())
        .param("raisonDemande", entreprise.getRaisonDemande())
        .param("numeroBCE", entreprise.getNumeroBCE())
        .param("nom", entreprise.getNom())
        .param("nomContact", entreprise.getNomContact())
        .param("prenomContact", entreprise.getPrenomContact())
        .param("telContact", entreprise.getTelContact())
        .param("emailContact", entreprise.getEmailContact())
        .param("rue", entreprise.getRue())
        .param("numero", entreprise.getNumero())
        .param("localite", entreprise.getLocalite())
        .param("codePostal", String.valueOf(entreprise.getCodePostal()))
        .param("pays", entreprise.getPays())
        .param("type", entreprise.getType())
        .param("numeroReference", entreprise.getNumeroReference())
        .param("statutSociete", entrepriseForm.getStatutSociete())
        .param("bonCommande", entrepriseForm.getBonCommande())
        .param("motPasse", entrepriseForm.getMotPasse())
        .param("confirmation", entrepriseForm.getConfirmation())
        ).andDo(print());
    resultActions.andExpect(status().isFound());
    resultActions.andExpect(redirectedUrl("../index.html"));
    resultActions.andExpect(view().name("redirect:../index.html"));
}

```

Figure 57 - Test ajout entreprise

Tout d'abord, dans ce test, je crée un objet entreprise avec toutes les informations d'une entreprise. Je rajoute aussi les documents de l'entreprise et le mot de passe.

Ensuite, je simule la requête http post avec le jeton unique csrf et les paramètres de la méthode *ajouterEntreprise* du contrôleur (voir figure 58 ci-dessous) et j'affiche le résultat de la requête.

```

@RequestMapping(value = "/ajouterEntreprise", method = RequestMethod.POST)
@Transactional(rollbackFor=Exception.class)
public String add(@ModelAttribute("entreprise") @Valid Entreprise entreprise, BindingResult bindingResultEntreprise,
    @ModelAttribute("entrepriseForm") @Valid EntrepriseForm entrepriseForm, BindingResult bindingResultEntrepriseForm, Model model) throws ParseException {
    if(bindingResultEntreprise.hasErrors() || bindingResultEntrepriseForm.hasErrors()) {
        model.addAttribute("type", types);
        model.addAttribute(LISTEPAYS, listPays);
        return "entreprise/ajouterEntreprise";
    }
    else if(entrepriseDao.getEntrepriseByEmail(entreprise.getEmailContact()) != null) {
        model.addAttribute("type", types); model.addAttribute(LISTEPAYS, listPays);
        model.addAttribute("errorMail", "Cette adresse est déjà utilisée !");
        return "entreprise/ajouterEntreprise";
    }
    else {
        Role role = roleDao.findRoleByName("ROLE_SUP_1");
        PasswordEncoder passwordEncoder = new BCryptPasswordEncoder();
        user = new User();
        user.setMotPasse(passwordEncoder.encode(entrepriseForm.getMotPasse()));
        user.setRole(role);
        user.setEnabled(true);
        entreprise.setEtat("En attente");
        entreprise.setDateCreation(dateTime.parse(dateTime.format(new Date())));
        entreprise.setUtilisateur(user);
        document = new Document();
        document.setNom("Bon de commande");
        document.setNomFichier(entrepriseForm.getBonCommande());
        document.setEntreprise(entreprise);
        document2 = new Document();
        document2.setNom("Statut de la société");
        document2.setNomFichier(entrepriseForm.getStatutSociete());
        document2.setEntreprise(entreprise);
        userDao.insert(user);
        entrepriseDao.insert(entreprise);
        documentDao.insert(document);
        documentDao.insert(document2);
        return "redirect:../index.html";
    }
}

```

Figure 58 - méthode ajout entreprise

Enfin, je définis les attentes via *andExpect* par rapport au statut de la réponse, l'url de redirection et le nom de la vue retournée.

Voici la requête http post :

```
MockHttpServletRequest:
  HTTP Method = POST
  Request URI = /entreprise/ajouterEntreprise
  Parameters = {raisonDemande=[rkrkrk], numeroBCE=[BE123456789], nom=[Defitest], nomContact=[Ratz], prenomContact=[lol], telContact=[0493138881], emailContact=[s.ratz@gmail.com...
  Headers = {}

Handler:
  Type = com.portailinscription.controller.EntrepriseController
  Method = public java.lang.String com.portailinscription.controller.EntrepriseController.add(com.portailinscription.model.Entreprise,org.springframework.validation.BindingI ...

Async:
  Async started = false
  Async result = null

Resolved Exception:
  Type = null

ModelAndView:
  View name = redirect:../index.html
  View = null
  Attribute = entreprise
  value = com.portailinscription.model.Entreprise@223967ea
  errors = []
  Attribute = entrepriseForm
  value = com.portailinscription.form.EntrepriseForm@4a6a6a69
  errors = []

FlashMap:
  Attributes = null
```

Figure 59 - Requete http post

Sur la figure 59 ci-dessus, on peut voir que la requête http post récupère bien tous les éléments de l'objet *Entreprise* via la variable *Parameters*.

On peut aussi voir que la vue qu'il souhaite retournée est la page d'accueil et que les deux modèles utilisés par la vue sont *entreprise* et *entrepriseForm*.

Voici la réponse de requête http post :

```
MockHttpServletResponse:
  Status = 302
  Error message = null
  Headers = {Cache-Control=[no-cache, no-store, max-age=0, must-revalidate], Pragma=[no-cache], Expires=[0], X-XSS-Protection=[1; mode=block], X-Frame-Options=[DENY], ...
  Content type = null
  Body =
  Forwarded URL = null
  Redirected URL = ../index.html
  Cookies = []
```

Figure 60 - Réponse requête http post

Ici, sur la figure 60 ci-dessus, on peut voir que le statut de la réponse est 302, ce qui veut dire que la redirection vers la page d'accueil va être faite.

Le test est donc réussi puisqu'on s'attendait à recevoir comme statut de réponse 302 (*status.isFound()*), une redirection vers la page d'accueil et le nom de la vue soit le nom de la page d'accueil.

Le test suivant (voir figure 61 page suivante) permet de vérifier lorsque le responsable de l'entreprise a cliqué sur le bouton « Ajouter » pour ajouter un travailleur (voir figure 41), celui-ci est bien redirigé vers la liste des travailleurs (voir figure 42).

```

@Test
public void ajoutPostTravailleurPage() throws Exception {
    entreprise = entrepriseDao.find(1);
    byte[] bytes = new byte[100]; String[] dates = new String[4]; MockMultipartFile[] files = new MockMultipartFile[4];
    MockMultipartFile photo = new MockMultipartFile("photoFichier", "David.png", "multipart/form-data", bytes);
    travailleur = new Travailleur(0, "Ratz", "David", "Belge", "FR", "0478484547", "d.ratz@gmail.com"); travailleurForm = new TravailleurForm();
    travailleur.setDateCreation(dateTime.parse(dateTime.format(new Date())));
    travailleur.setDateNaissance(dtl.parse("14/04/2017"));
    travailleur.setEntreprise(entreprise);
    travailleurForm.setDateNaissance("14/04/2017");
    travailleurForm.setPhotoFichier(photo);
    travailleurForm.setPhoto(travailleurForm.getPhotoFichier().getOriginalFilename());
    for (int i = 0; i < listFichier.size(); i++) {
        MockMultipartFile multipartFile = new MockMultipartFile("file", listFichier.get(i) + ".txt", "multipart/form-data", bytes);
        files[i] = multipartFile;
        dates[i] = dtl.format(dateTime.parse(dateTime.format(new Date())));
    }
    travailleurForm.setDate(dates); travailleurForm.setFile(files);
    ResultActions resultActions = mockMvc.perform(fileUpload("/travailleur/ajouterTravailleur?id=" + entreprise.getId())
        .file(files[0]).file(files[1]).file(files[2]).file(files[3]).file(photo).with(csrf())
        .param("numeroReference", String.valueOf(travailleur.getNumeroReference()))
        .param("nom", travailleur.getNom())
        .param("prenom", travailleur.getPrenom())
        .param("nationalite", travailleur.getNationalite())
        .param("langue", travailleur.getLangue())
        .param("telephone", travailleur.getTelephone())
        .param("adresseEmail", travailleur.getAdresseEmail())
        .param("dateCreation", dateTime.format(travailleur.getDateCreation()))
        .param("dateNaissance", dtl.format(travailleur.getDateNaissance()))
        .param("file", travailleurForm.getFile().toString())
        .param("photo", travailleurForm.getPhoto())
        .param("photoFichier", travailleurForm.getPhotoFichier().toString())
        .param("date", travailleurForm.getDate())
        .param("dateNaissance", travailleurForm.getDateNaissance())
        ).andDo(print());
    resultActions.andExpect(status().isFound());
    resultActions.andExpect(redirectedUrl("listeTravailleur.html"));
    resultActions.andExpect(view().name("redirect:listeTravailleur.html"));
}

```

Figure 61 - test ajout travailleur

Tout d'abord, je crée un objet *Travailleur*, un fichier pour la photo, les documents du travailleur.

Ensuite, je simule la requête http post *fileupload*²⁴ (pour une requête *multipart*) avec les différents fichiers *file*, le jeton unique csrf et les paramètres de la méthode *ajouterTravailleur* (voir figure 35 plus haut) et j'affiche le résultat de la requête.

Enfin, je définis les attentes via *andExpect* par rapport au statut de la réponse, l'url de redirection et le nom de la vue retournée.

Voici la requête http post :

```

MockHttpServletRequest:
  HTTP Method = POST
  Request URI = /travailleur/ajouterTravailleur
  Parameters = {id=[1], numeroReference=[0], nom=[Ratz], prenom=[David], nationalite=[Belge], langue=[FR], telephone=[0478484547], adresseEmail=[d.ratz@gmail.com] ...
  Headers = {Content-Type=[multipart/form-data]}

Handler:
  Type = com.portailinscription.controller.TravailleurController
  Method = public java.lang.String com.portailinscription.controller.TravailleurController.add(java.lang.Integer,org.springframework.web.multipart.MultipartFile...)

Async:
  Async started = false
  Async result = null

Resolved Exception:
  Type = null

ModelAndView:
  View name = redirect:listeTravailleur.html
  View = null
  Attribute = travailleurForm
    value = com.portailinscription.form.TravailleurForm@7168112e
    errors = []
  Attribute = travailleur
    value = Travailleur [nom=Ratz, prenom=David]
    errors = []

FlashMap:
  Attributes = null

```

Figure 62 - Requete http post

²⁴ <http://docs.spring.io/spring/docs/current/javadoc-api/org.springframework.test.web.servlet.request.MockMvcRequestBuilders.html>

Sur la figure 62 de la page précédente, on peut voir que la requête http post récupère bien tous les éléments de l'objet Entreprise via la variable *Parameters* et l'en-tête de la requête est bien de type *multipart*.

On peut aussi voir que la vue qu'il souhaite retournée est la liste des travailleurs et que les deux modèles utilisés par la vue sont *travailleur* et *travailleurForm*.

Voici la réponse de requête http post :

```
MockHttpServletResponse:
  Status = 302
  Error message = null
  Headers = {Cache-Control=[no-cache, no-store, max-age=0, must-revalidate], Pragma=[no-cache], Expires=[0], X-XSS-Protection=[1; mode=block], X-Frame-Options=[DENY]...
  Content type = null
  Body =
  Forwarded URL = null
  Redirected URL = listeTravailleur.html
  Cookies = []
```

Figure 63 - Reponse requete http post

Ici, sur la figure 63 ci-dessus, on peut voir que le statut de la réponse est 302, ce qui veut dire que la redirection vers la liste des travailleurs va être faite.

Le test est donc réussi puisqu'on s'attendait à recevoir comme statut de réponse 302 (`status.isFound()`), une redirection vers la liste des travailleurs et le nom de la vue soit le nom de la liste des travailleurs.

6.15.1. Problème rencontré

J'ai eu un gros problème au commencement des tests avec Spring car je ne savais pas comment écrire le test même avec l'aide de la documentation Spring.

Je n'arrivai pas à exécuter la bonne requête et trouver la bonne façon de l'écrire au niveau du *perform* (voir figure 61 ci-dessus).

6.15.2. Solution trouvée

J'ai donc décidé de créer un test sur le projet d'exemple que j'avais mis en place pour l'apprentissage de Spring. Après plusieurs tentatives, j'ai réussi à faire fonctionner ce dernier.

Le problème était que je n'utilisais pas correctement la requête à passer au test.

7. Conclusion

Ce mémoire est le résumé de mon stage dans la société *Defimedia* d'une durée de 15 semaines. Je vais conclure celui-ci en expliquant mon sentiment par rapport au stage, le respect des objectifs du stage et l'amélioration possible à apporter au projet.

Tout d'abord, ces 15 semaines m'ont permis de mieux comprendre comment fonctionne une entreprise en informatique et le rôle qu'à un analyste développeur dans celle-ci. J'ai aussi pu mettre en pratique les connaissances que j'ai acquises pendant ma formation à HELMo. J'ai pu découvrir de nouvelles personnes et un environnement de travail différent de celui de l'école mais aussi de nouvelles technologies.

Ensuite, l'objectif du stage était de développer un portail web pour la gestion d'accès en milieu industriel afin qu'une entreprise sous-traitante ayant reçu une demande d'un client s'inscrive sur le portail et ajoute elle-même ses travailleurs. L'objectif est atteint même s'il reste encore quelques améliorations à apporter au portail.

Enfin, l'amélioration possible à apporter au portail est la partie administration de celui-ci.

8. Bibliographie

8.1. Internet

Defimedia [en ligne]. [Consulté le 03 février 2017]. Disponible à l'adresse : <https://www.defimedia.be/>

Spring [en ligne]. [Consulté le 13 février 2017]. Disponible à l'adresse : <https://spring.io>

Spring Framework Reference Documentation [en ligne]. [Consulté le 13 février 2017]. Disponible à l'adresse : <https://docs.spring.io/spring/docs/current/spring-framework-reference/html/>

Spring (framework) [en ligne]. [Consulté le 24 avril 2017]. Disponible à l'adresse : [https://fr.wikipedia.org/wiki/Spring_\(framework\)](https://fr.wikipedia.org/wiki/Spring_(framework))

Spring Tutorial [en ligne]. [Consulté le 24 avril 2017]. Disponible à l'adresse : <https://www.tutorialspoint.com/spring/index.htm>

Postresql [en ligne]. [Consulté le 24 avril 2017]. Disponible à l'adresse : <https://www.postgresql.org/>

Documentation PostgreSQL 9.6.2 [en ligne]. [Consulté le 13 février 2017]. Disponible à l'adresse : <http://docs.postgresql.fr/9.6/pg96.pdf>

PostgreSQL Tutorial [en ligne]. [Consulté le 13 février 2017]. Disponible à l'adresse : <https://www.tutorialspoint.com/postgresql/index.htm>

Spring Security Reference [en ligne]. [Consulté le 24 avril 2017]. Disponible à l'adresse : <https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/>

[Spring Security] Authentification par certificat [en ligne]. [Consulté le 03 avril 2017]. Disponible à l'adresse : <https://blog.netapsys.fr/spring-security-authentification-par-certificat/>

Security Testing Tutorial [en ligne]. [Consulté le 24 avril 2017]. Disponible à l'adresse : https://www.tutorialspoint.com/security_testing/index.htm

Les 12 règles de l'ergonomie Web [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse : <http://ludismedia.com/12-regles-ergonomie-web/>

Les grandes règles d'ergonomie de votre site web [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse : <http://www.nobilito.fr/2016/11/les-grandes-regles-dergonomie-de-votre-site-web/>

Welcome to OWASP the free and open software security community [en ligne]. [Consulté le 12 mai 2017]. Disponible à l'adresse : https://www.owasp.org/index.php/Main_Page

OWASP TOP 10 – 2013 The ten most critical web application security risks
[Consulté le 12 mai 2017]. Disponible à l'adresse :
https://www.owasp.org/images/f/f8/OWASP_Top_10_-_2013.pdf

Introduction to Spring Testing [en ligne]. [Consulté le 03 avril 2017]. Disponible à l'adresse :
<http://docs.spring.io/spring/docs/current/spring-framework-reference/html/testing-introduction.html>

Introducing Spring MVC test framework [en ligne]. [Consulté le 03 avril 2017]. Disponible à l'adresse :
<https://blog.zenika.com/2013/01/15/spring-mvc-test-framework/>

8.2. Images

Logo de Defimedia [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
https://www.defimedia.be/sites/default/files/logo_1.png

Logo de Spring [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
<https://spring.io/img/spring-by-pivotal.png>

Logo de l'architecture de Spring [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
https://www.tutorialspoint.com/spring/images/spring_architecture.png

Logo de Spring Security [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
<https://ordina-jworks.github.io/img/spring-security-logo.png>

Logo de PostgreSQL [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
<http://logonoid.com/postgresql-logo/>

Logo de Hibernate [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
https://upload.wikimedia.org/wikipedia/fr/4/45/Logo_Hibernate.png

Logo de Dinec International [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
<http://www.dinec.be/servlet/Repository/logo-dinec-2015.png?ID=4410>

Logo de Maven [en ligne]. [Consulté le 22 mai 2017]. Disponible à l'adresse :
https://upload.wikimedia.org/wikipedia/commons/thumb/0/0b/Maven_logo.svg/2000px-Maven_logo.svg.png

Annexe

Schéma et explications des tables de la base de données

Le schéma de base de données du portail web sur la gestion d'accès des sous-traitants en milieu industriel.



La structure de la table « Utilisateur » est la suivante :

Champs	Description	Type
Id	Id unique de l'utilisateur Incrémentation automatique	Int
MotPasse	Le mot de passe de l'utilisateur	varchar(255)
Id Role	Le rôle de l'utilisateur	Int

Table « Entreprise »

La table « Entreprise » représente les entreprises ayant reçu un bon de commande de la part du client ou les entreprises sous-traitantes.

La structure de la table « Entreprise » est la suivante :

Champs	Description	Type
Id	Id unique de l'entreprise Incrémentation automatique	Int
NumeroBCE	Le numéro BCE de l'entreprise	varchar(13)
Nom	Le nom de l'entreprise	varchar(255)
NomContact	Le nom de la personne de contact	varchar(255)
PrenomContact	Le prénom de la personne de contact	varchar(255)
TelContact	Le téléphone de la personne de contact	varchar(255)
EmailContact	L'email de la personne de contact	varchar(255)
Rue	La rue où se trouve le siège social	varchar(255)
Numero	Le numéro où se trouve le siège social	varchar(255)
CodePostal	Le code postal où se trouve le siège social	varchar(255)
Localite	La localité où se trouve le siège social	varchar(255)
Pays	Le pays où se trouve le siège social	varchar(255)
Etat	L'état de validation de l'entreprise	varchar(255)
RaisonDemande	La raison de la demande	varchar(255)
NumeroReference	Le numéro de référence de l'entreprise	varchar(255)
Remarque	Remarque éventuelle sur l'entreprise	varchar(255)
NumeroBCEMaitre	Le numéro BCE de l'entreprise maître	varchar(13)
DateCreation	La date de création de l'entreprise	Date
DateModification	La date de modification de l'entreprise	Date
Id_Utilisateur	L'identifiant de l'utilisateur qui a créé l'entreprise	Int
Type	Le type d'entreprise (Société sous-traitante maître ou société sous-traitante d'une société maître)	varchar(255)

Table « Travailleur »

La table « Travailleur » représente le ou les travailleur(s) de l'entreprise sous-traitante maître ou l'entreprise sous-traitante de l'entreprise maître.

La structure de la table « Travailleur » est la suivante :

Champs	Description	Type
Id	Id unique du travailleur Incrémentation automatique	int
NumeroReference	Le numéro de référence du travailleur	varchar(255)
Nom	Le nom du travailleur	varchar(255)
Prenom	Le prénom du travailleur	varchar(255)
Photo	La photo du travailleur	varchar(255)
Nationalite	La nationalité du travailleur	varchar(255)
Langue	La langue du travailleur	varchar(255)
Telephone	Le téléphone du travailleur	varchar(255)
AdresseEmail	L'adresse email du travailleur	varchar(255)
DateCreation	La date de création du travailleur	Date
DateModification	La date de modification du travailleur	Date
Id_Entreprise	L'identifiant de l'entreprise où se trouve	int
Id_ListeNoire	L'identifiant de l'utilisateur qui a créé l'entreprise	int
DateNaissance	La date de naissance du travailleur	Date

Table « Document »

La table « Document » représente les documents de l'entreprise (« Bon de commande » et « Statut de la société ») et du travailleur (« E101 », « Limosa », « Certificat médical » et « Copie passeport »).

La structure de la table « Document » est la suivante :

Champs	Description	Type
Id	Id unique de l'entreprise Incrémentation automatique	Int
Nom	Le nom du document : « E101 », « Limosa », « Certificat Medical » ou « Copie Passeport »	varchar(255)
NomFichier	Le nom du fichier stocké sur le serveur	varchar(255)
DateValidite	La date de validité du document	Date
Id_Travailleur	L'identifiant du travailleur qui possède ce document	varchar(255)
Id_Entreprise	L'identifiant de l'entreprise qui possède ce document	varchar(255)

Table « Projet »

La table « Projet » représente le projet pour l'entreprise.

La structure de la table « Projet » est la suivante :

Champs	Description	Type
Id	Id unique du projet Incrémentation automatique	Int
Nom	Le nom du projet	varchar(255)
Rue	La rue où se déroule le projet	varchar(255)
Numero	Le numéro où se déroule le projet	varchar(255)
CodePostal	Le code postal où se déroule le projet	varchar(255)
Localite	La localité où se déroule le projet	varchar(255)
Pays	Le pays où se déroule le projet	varchar(255)
Id_Type	L'identifiant du type de projet	Int
Id_Entreprise	L'identifiant de l'entreprise qui réalise ce projet	Int

Table « Acces »

La table « Acces » représente les accès pour des travailleurs pour un projet.

La structure de la table « Acces » est la suivante :

Champs	Description	Type
DateDebut	La date de début du projet	Date
DateFin	La date de fin du projet	Date
DateValidationControleAcces	La date de validité de l'accès	Date
DateCreation	La date de création de l'accès	Date
EtatDemande	L'état de la demande d'accès	varchar(255)
Id_Projet	L'identifiant du projet pour lequel l'accès est obtenu	int
Id => id_Travailleur	L'identifiant du travailleur qui a accès au chantier	int

Table « Role »

La table « Role » représente le rôle d'un utilisateur.

La structure de la table « Role » est la suivante :

Champs	Description	Type
Id	Id unique du rôle Incrémentation automatique	Int
Nom	Le nom du rôle	varchar(255)

Table « Type »

La table « Type » représente le type d'un projet et d'un questionnaire.

La structure de la table « Type » est la suivante :

Champs	Description	Type
Id	Id unique du type Incrémentation automatique	Int
Nom	Le nom du type de projet	varchar(255)

Table « ListeNoire »

La table « ListeNoire » représente les utilisateurs ayant un document dont la date de validité est dépassée.

La structure de la table « ListeNoire » est la suivante :

Champs	Description	Type
Id	Id unique de la liste noire Incrémentation automatique	Int
Nom	Le nom du travailleur placé en liste noire	varchar(255)
Prenom	Le prénom du travailleur placé en liste noire	varchar(255)
DateNaissance	La date de naissance du travailleur placé en liste noire	Date