

---

## Concrétisation disciplinaire Projet ESTER

---

*ESTER :*

Natacha FOUQUET

Anna LLOYD

*Chefs de projet (M2) :*

Hugues DUMONT

Guillaume HUET

Zineb LOUKILI

*Référents :*

Vincent BARICHARD

Laurent GARCIA

David GENEST

Jean-Michel RICHER

*Équipe de développement (M1) :*

Nidal BEDYOUCH

Imane BELHOUARI

Théo DÉZÉ

Charles MALLET



# Table des matières

<b>1</b>	<b>Organisation du projet</b>	<b>3</b>
1.1	Présentation du projet	3
1.2	Choix technologiques	3
1.2.1	Technologies côté serveur	3
1.2.2	Technologies côté client	3
1.2.3	Base de données	4
1.3	Planification et répartition des tâches	4
1.3.1	Outils utilisés	4
1.3.2	Diagramme de Gantt	5
1.3.3	Répartitions des rôles	5
<b>2</b>	<b>Back-end</b>	<b>6</b>
2.1	Base de Données	6
2.1.1	Utilisateur ESTER	6
2.1.2	Questionnaire	7
2.1.3	Réponse	7
2.1.4	Sécurité	7
2.2	Connexion	7
2.2.1	Génération chaîne de caractères aléatoire	7
2.2.2	Formulaire de première connexion du patient	9
2.3	Administration e-mail	9
2.3.1	Envoi de l'e-mail	9
2.3.2	Réinitialisation du mot de passe	10
<b>3</b>	<b>Front-end</b>	<b>11</b>
3.1	Structure du site et style	11
3.2	Questionnaires	12
3.2.1	Création de questionnaires	12
3.2.2	Questionnaire Eval_Risk_TMS	15
3.3	Résultats	16
3.3.0.1	DWR	16
<b>4</b>	<b>Problèmes rencontrés et perspectives</b>	<b>17</b>
4.1	Problèmes au cours du projet	17
4.1.1	Modification des Scripts	17
4.1.2	Versions Bootstrap et JQuery	19
4.2	Perspectives d'amélioration	19



# Remerciements

Nous souhaitons, tout d'abord remercier nos chefs de projet qui nous ont aidés et soutenus tout le long de cette réalisation. Sans leur participation, nous n'aurions pas été en mesure d'aboutir au site que nous avons actuellement. Grâce à leur implication, nous avons pu mieux intégrer le sujet et nous améliorer dans nos domaines respectifs. D'un autre côté, nous voudrions les remercier pour nous avoir donné leur avis sur ce rapport ainsi qu'avoir participé à sa relecture.

Dans un deuxième temps, nous voudrions remercier nos professeurs qui sont restés présents à nos côtés pendant tout le développement, qui nous ont écoutés et conseillés lorsque nous en avons besoin.

Enfin, nous tenions à remercier l'équipe ESTER pour leur retour à la mi-novembre concernant la réunion de présentation de nos travaux d'alors.

# Introduction

Au cours de ce semestre de Master 1 Informatique, nous avons participé, en concrétisation disciplinaire, au développement d'une application ou d'un site web. Nous devions répondre au besoin d'un client, en collaboration avec les Master 2. Ces derniers avaient pour rôle d'organiser les tâches et de gérer toute l'organisation du projet, ainsi que la structure globale. À ce titre, ils étaient les organisateurs et chefs de projet. En tant que Master 1, nous étions chargés du développement de l'application. Nous devions nous assurer de la bonne implémentation des fonctionnalités et du bon fonctionnement de l'application. Dans ce rapport, vous pourrez découvrir l'avancée de notre projet durant ce semestre, selon quatre axes. En premier lieu, nous parlerons dans ce compte-rendu de l'organisation de projet suivi du projet côté serveur. Puis, nous poursuivrons avec la partie côté client et nous parlerons des difficultés et problèmes rencontrés avant de conclure à propos de ce cycle de développement.

# Chapitre 1

## Organisation du projet

### 1.1 Présentation du projet

En lien avec l'équipe ESTER, nous avons reçu pour objectif de réaliser un site web permettant de créer des questionnaires médicaux auxquels pourraient répondre les salariés de diverses entreprises. Selon le cahier des charges et tout en respectant le secret professionnel, nous devons faire en sorte que les patients (salariés) puissent répondre aux questions que les médecins auraient préparé dans des questionnaires, enregistrés dans une base de données. Une fois créé et enregistré, le corps médical devait pouvoir les attribuer en fonction des cas. Par la suite, des résultats devaient être calculés et affichés aux personnels soignant afin qu'ils puissent se rendre compte des chiffres.

### 1.2 Choix technologiques

#### 1.2.1 Technologies côté serveur

**NodeJS** : NodeJS permet de créer des services web en utilisant du JavaScript. NodeJS est une technologie moderne avec beaucoup de documentation récente. Elle permet de n'utiliser quasiment que du JavaScript. NodeJS est quelque peu complexe car il utilise un mécanisme de callback et de promesses. Ils sont dus au fonctionnement asynchrone de NodeJS qui rend le tout très performant.

**JEE** : JEE est la version de Java orienté pour la création d'application pour les entreprises et permet de développer des services web. JEE présente l'avantage d'être simple, car il utilise des mécaniques proche de celle du Java que nous connaissons.

Nous avons proposé d'utiliser NodeJS côté serveur, mais nos chefs de projet ont privilégié l'utilisation de JEE car c'est une technologie qu'ils connaissaient et qui nous sera utile lors du prochain semestre. Ils estimaient, par ailleurs, qu'il s'agissait d'une solution facile à apprendre, en comparaison de NodeJS.

#### 1.2.2 Technologies côté client

**JavaScript** : JavaScript est un langage de programmation développé par Netscape en 1995 sous le nom de LiveScript. Il s'agit d'un langage de script léger, orienté objet, principalement connu comme le langage de script des pages web. JS nous permet d'effectuer des opérations côté client, modifier l'affichage ou lier des fonctions avec des boutons.

**CSS** : « Cascading Style Sheets » ce qui signifie « feuille de style en cascade ». Il s'agit d'un langage informatique utilisé pour mettre en forme les fichiers HTML ou XML. CSS permet de modifier l'affichage pour mieux correspondre aux besoins du client, en plus de Bootstrap qui a déjà un CSS de défini. Il permet d'améliorer la qualité visuelle de l'interface utilisateur.

**Bootstrap** : Correspond à une collection d'outils, développée depuis 2010, utiles pour la création de sites web. Cette collection contient des codes HTML et CSS ainsi que des extensions JavaScript. Il nous permet d'avoir un aspect unifié et de faciliter la mise en place de page adaptatif entre

les différents mobiles et les ordinateurs. Nous avons été amenés à utiliser Bootstrap lors de nos divers travaux.

**Highcharts** : Highcharts est une librairie JavaScript qui permet de générer des graphiques interactifs. Le projet nous demandait une certaine mise en forme des résultats. Les paramétrages s'effectuent en JSON et offrent la possibilité d'exporter dans les différents formats CSV, PDF, PNG et autres ainsi qu'être compatible avec les PCs/Tablettes/Smartphones. Nous l'avons choisi pour sa facilité à exporter les données et car il proposait des exemples pour faciliter son intégration.

### 1.2.3 Base de données

Pour le projet, nous avons eu besoin d'une base de données modulaire car un des besoins était que les questionnaires pouvaient évoluer : création, modification ou suppression de questions. Un autre des besoins était que l'utilisateur puisse faire des sauvegardes partielles pour reprendre le questionnaire en cas de problème ou si l'utilisateur souhaitait faire une pause.

En plus des besoins spécifiques pour la sauvegarde des questionnaires et des réponses, il y a des besoins plus génériques comme la gestion des comptes que nous verrons plus en détails dans une autre partie.

**MongoDB** : Nous sommes partis sur du MongoDB qui fait partie de la mouvance NoSQL s'écarter du paradigme classique des bases relationnelles. Cela nous permet de nous affranchir d'une des contraintes des bases de données SQL qui est de prévoir un schéma prédéfini. Nous sommes quand même partis d'un schéma de base pour avoir des données en partie structurée.

En offrant une plus grande flexibilité, en permettant de gérer des données non structurées, MongoDB est particulièrement utile pour les questionnaires et les réponses car si un questionnaire est modifié, il était nécessaire que les anciennes réponses restent en partie utilisables.



FIGURE 1.1 – Logo de MongoDB

Ce choix du type de la base de données a été proposé par nos chefs de projet. La raison du choix de MongoDB s'explique par le fait qu'il est le membre le plus sécurisé de la famille NoSQL et facilement maintenable.

## 1.3 Planification et répartition des tâches

### 1.3.1 Outils utilisés

**Github** : Git est un outil très utilisé de nos jours dans les entreprises et projets nécessitant un partage de fichiers. Grâce à un espace de stockage à distance et à des fichiers en local, chacun peut travailler sur sa partie et la partager aux autres de manière efficace. Sitôt une partie réalisée, les utilisateurs de Git peuvent envoyer en ligne leurs tâches ce qui permet à leurs collègues de récupérer le code et de l'ajouter au leur par fusion.

**Trello** : Trello est un site internet que nous avons utilisé régulièrement afin de nous attribuer des tâches. Il nous permettait de connaître les parties de chacun et ainsi de mieux nous diriger lorsqu'il était nécessaire de demander des fonctionnalités, les uns les autres.

**Slack** : Nous disposions tout au long du cycle de développement d'un Slack afin d'échanger respectivement nos remarques et interrogations. Toutefois, ne disposant pas d'une version premium, nous ne l'avons pas utilisé pour partager des informations importantes car elles auraient été perdues.



**E-mail** : Comme explicité précédemment, nous n'avons pas pu nous servir de Slack pour échanger de documents importants ou pour garder des conversations importantes. C'est pourquoi nous avons privilégié les adresses e-mails.

### 1.3.2 Diagramme de Gantt

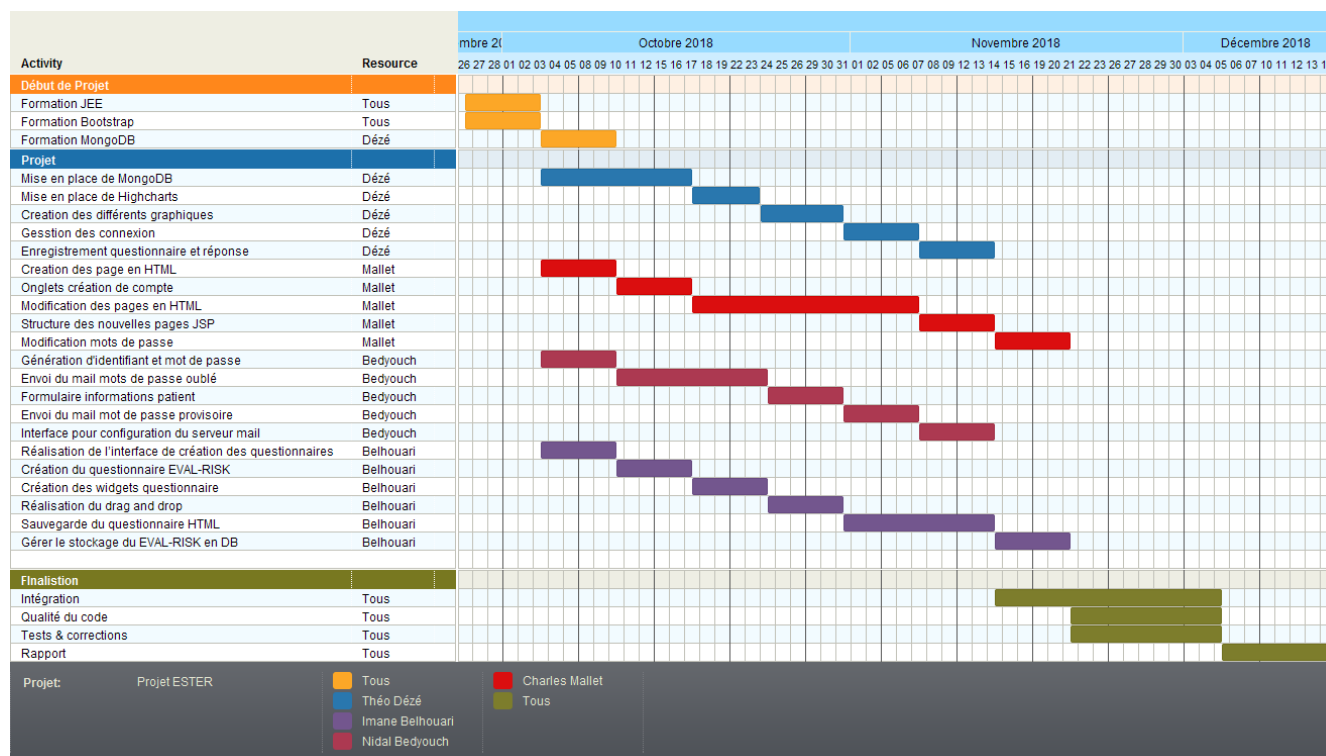


FIGURE 1.2 – Diagramme de Gantt

### 1.3.3 Répartitions des rôles

Nos chefs de projet nous ont demandé dès le début du semestre de leur indiquer nos préférences concernant nos tâches dans le cycle de développement. Imane et Charles se sont occupés principalement du Front-End, Théo et Nidal se sont chargés du Back-End.

Par ailleurs, nos Master 2 souhaitaient que nous nous attribuions un rôle de groupe :

**Nidal** : Chargée de décisions

**Imane** : Chargée d'organisation

**Théo** : Référent technique

**Charles** : Chargé de communication

## Chapitre 2

# Back-end

### 2.1 Base de Données

La base de données est divisée en plusieurs collections.

Pour l'implémentation, nous avons utilisé le driver officiel proposé par MongoDB pour le Java. Nous avons créé une classe mère en Java utilisée par héritage et qui permet de définir des interfaces simplifiant l'utilisation du reste du code.

#### 2.1.1 Utilisateur ESTER

Clé	Valeur
Identifiant	Chaîne de caractères, peut être utilisé à la place du mail lors de la connexion
Nom	Chaîne de caractères
Prénom	Chaîne de caractères
Première connexion	Boolean
Mot de passe	Chaîne de caractères chiffrée
Mail	Chaîne de caractères
Statut	Chaîne de caractères, représente le statut de l'Utilisateur (Médecin, Administrateur, Préventeur, Assistant)

FIGURE 2.1 – Tableau de la collection Utilisateur

La collection correspond à celle prévue par nos chefs de projet comme les collections Entreprise et Salarié qui sont des collections proches (L'entreprise ne possède pas d'email et le salarié n'a pas de mot de passe mais a la liste des questionnaires répondus ou à répondre) de Utilisateur ESTER donc nous ne nous répéterons pas.

### 2.1.2 Questionnaire

Clé	Valeur
Nom	Nom du questionnaire
Identifiant	Version simplifiée du nom
Identifiant ESTER	Identifiant de la personne qui a soumis le questionnaire
Date de soumission	Date
Mail	Chaîne de caractères
HTML	Chaîne de caractères (HTML brut)

FIGURE 2.2 – Tableau de la collection Questionnaire

### 2.1.3 Réponse

Clé	Valeur
Identifiant salarie	Identifiant du salarie qui a répondu
Identifiant questionnaire	Identifiant du questionnaire répondu
Reponses	Tableau associatif qui lie les identifiants des réponses avec leurs réponses

FIGURE 2.3 – Tableau de la collection Réponse

### 2.1.4 Sécurité

Nous stockons des mots de passe dans la base de données, nous avons eu besoin de sécuriser les mots de passe pour ne pas les stocker en clair. Nous avons comparé plusieurs technologies MD5, SHA256, SHA512, PBKDF2, BCrypt et SCrypt. Ce sont toutes des fonctions de hachage mais certaines proposent des bases de salage (PBKDF2, BCrypt, SCrypt) qui permettent de se protéger contre les attaques utilisant des rainbows tables ou par dictionnaire. Certaines fonctions de hachage sont difficilement optimisables ce qui permet de rendre plus difficile les attaques par la force brute. De ce fait, cela donne le temps à l'utilisateur de changer son mot de passe.

Nous avons choisi d'utiliser BCrypt car il propose une implémentation en java et il fait partie des algorithmes les plus sécurisés.

## 2.2 Connexion

### 2.2.1 Génération chaîne de caractères aléatoire

- Problématique : Lors de la création d'un utilisateur ESTER par un autre (un médecin peut créer un compte infirmier, assistant, préventeur, et un administrateur peut tout créer), un mot de passe provisoire est communiqué à l'utilisateur pour sa première connexion. D'autre part, le patient se connecte en utilisant un identifiant unique de 5 caractères communiqué par son médecin.
- Implémentation : Pour l'implémentation, nous avons utilisé une instance de la classe `SecureRandom`<sup>1</sup> permettant de tirer aléatoirement les caractères de la chaîne et de la permuter aléatoirement afin d'être imprédictible.

---

1. Cette classe fournit un générateur de nombres aléatoires (RNG) de chiffrement fort.

- Bilan : Le générateur de chaîne de caractères aléatoires est fonctionnel et sera utilisé pour la génération de mot de passe provisoire. Tandis que, l'identifiant du patient sera abandonné vu que la complexité de vérification de l'unicité est haute. Les M2 se chargeront de définir un algorithme générant des identifiants cryptés et uniques.

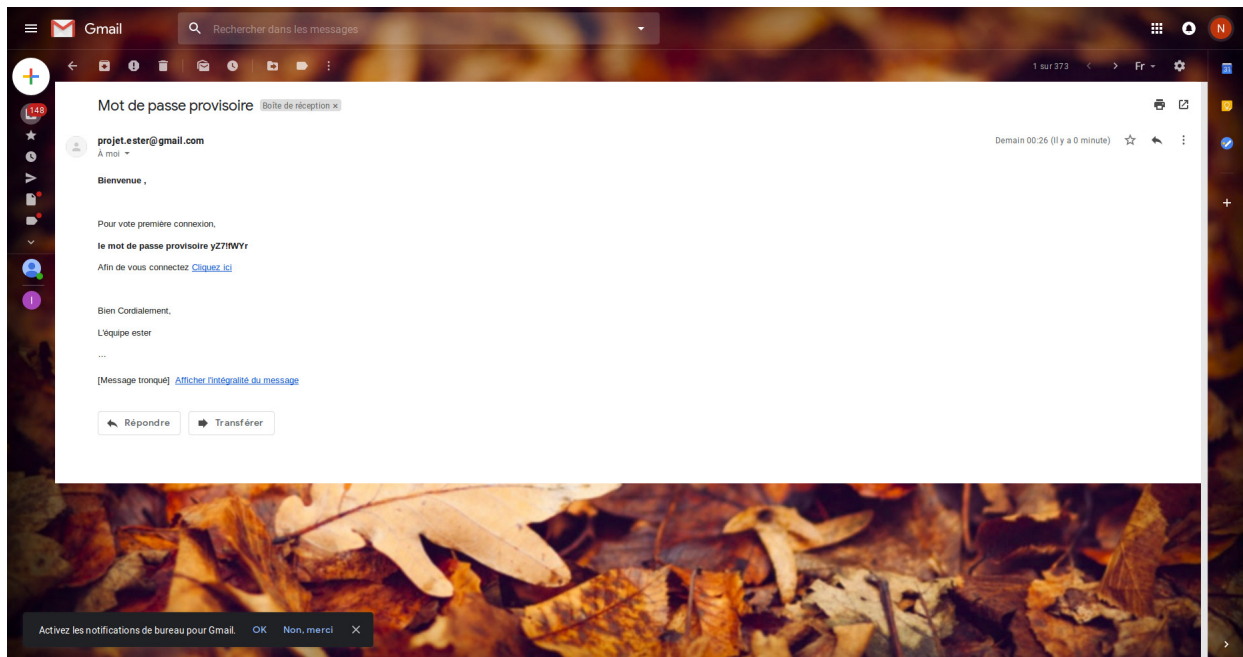


FIGURE 2.4 – E-mail pour la première connexion

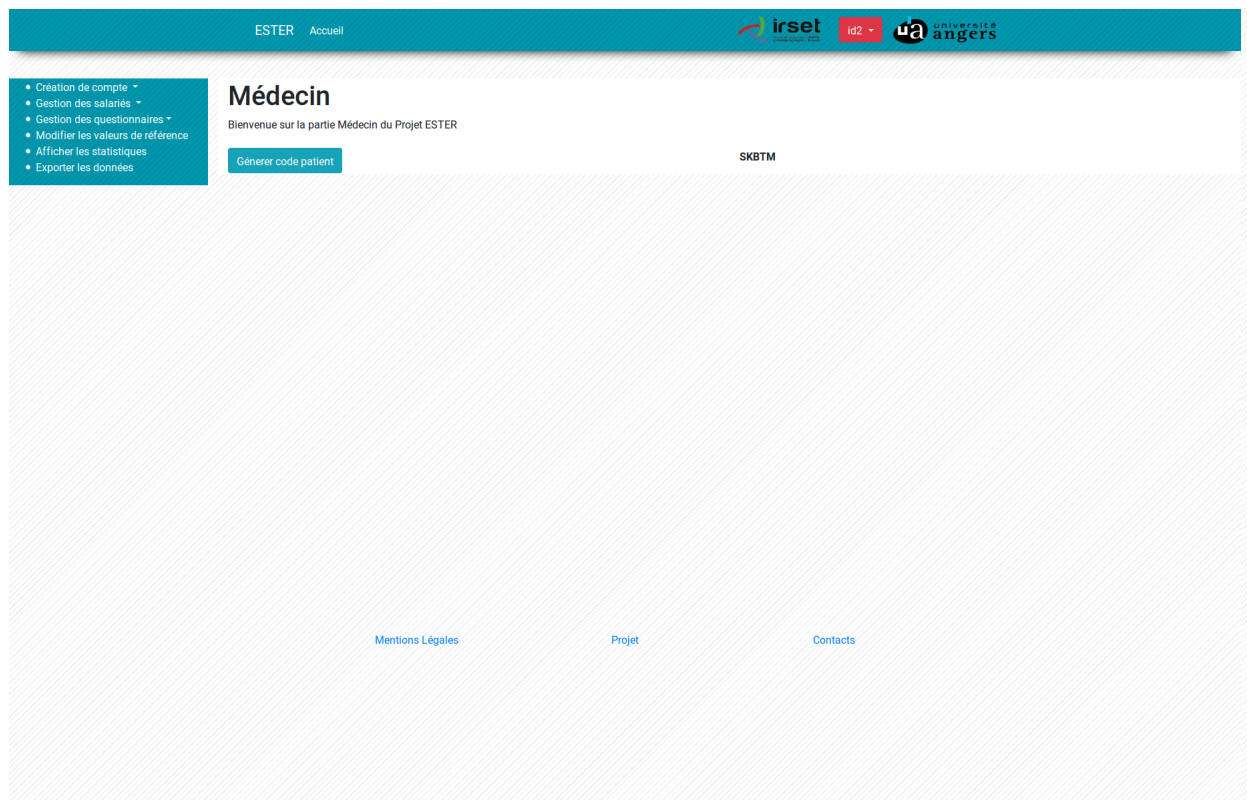


FIGURE 2.5 – Générateur de codes patients

## 2.2.2 Formulaire de première connexion du patient

Lors de la création d'un compte patient seul l'identifiant est enregistré en base de données. Et afin de compléter les informations manquantes au profil à titre d'illustration : le sexe, l'âge quinquennal (pour préserver l'anonymat du patient), poste de travail, secteur d'activité et département, le patient est redirigé lors de sa première connexion vers le formulaire ci-dessous.

ESTER Accueil

irset SKBTM univ angers

**Salarié**  
Bienvenue sur la partie Salarié du Projet ESTER

• Modifier mon profil  
• Voir les questionnaires  
• Voir les statistiques

Veillez saisir vos informations avant de débiter les questionnaires :

Sexe : ☐ Homme ☒ Femme

Âge : 55-59

PCS : 311d Psychologues, psychanc

NAF : 86 Activités pour la santé hun

Region : 52 Pays de la Loire

Département : 49 Maine-et-Loire

Soumettre

Mentions Légales Projet Contacts

FIGURE 2.6 – Formulaire du patient à la première connexion

## 2.3 Administration e-mail

### 2.3.1 Envoi de l'e-mail

- Problématique : Pour la gestion des utilisateurs ESTER, nous avons eu besoin d'envoyer des emails afin de communiquer un mot de passe provisoire pour la première connexion ainsi qu'un lien permettant de réinitialiser le mot de passe.
- JavaMail API et protocole :
  1. JavaMail API : L'API JavaMail fournit une plateforme indépendante et un Framework de protocoles indépendant pour créer des mails et des applications de messagerie. L'API JavaMail fournit un ensemble de classes abstraites définissant des objets qui constituent un système de messagerie. Il s'agit d'un package optionnel (extension standard) pour la lecture, la composition et l'envoi de messages électroniques. JavaMail fournit des éléments utilisés pour construire une interface avec un système de messagerie, y compris des composants système et des interfaces. Bien que cette spécification ne définisse aucune implémentation spécifique, JavaMail inclut plusieurs classes qui implémentent les normes de messagerie Internet RFC822 et MIME. Ces classes sont livrées dans le cadre du package de classe JavaMail.

2. Protocole SMTP : SMTP est l'acronyme de Simple Mail Transport Protocol. Ce protocole défini par la recommandation RFC 821 permet l'envoi de mails vers un serveur de mails qui supporte ce protocole.
- Implémentation :
- Envoi d'e-mails : L'envoi d'e-mails en utilisant JavaMail API nécessite :
1. Une authentification nécessitant un e-mail, son mot de passe, un hôte et un port. Pour notre projet, nous nous sommes servis du serveur Gmail correspondant à 'smtp.gmail.com ' et le port 587 ;
  2. Une instance MimeMessage servant à indiquer destinataire, sujet et corps du message pouvant être en HTML ou chaîne de caractère.
  3. Pour Java9+, il est nécessaire d'ajouter le module activation.jar parce qu'il n'est plus activé par défaut.
- Configuration serveur mail : Un compte administrateur peut accéder à l'interface lui permettant de configurer l'envoi d'e-mails en indiquant un e-mail, son mot de passe, un hôte et un port.

### 2.3.2 Réinitialisation du mot de passe

Dans le cadre de la gestion des utilisateurs ESTER, il s'est avéré important de définir la fonctionnalité réinitialisation de mots de passe. Pour cette fin, nous avons défini deux interfaces, la première sert à indiquer l'e-mail du compte et la deuxième sert à définir un nouveau mot de passe, ainsi que l'email de réinitialisation expirant dans un délai prédéfini. Pour l'implémenter, nous avons opté pour l'utilisation d'un token unique dans l'URL envoyé en e-mail. Il est généré en utilisant la méthode Java `UUID.randomUUID()`, qui crée un UUID aléatoire afin d'éviter les collisions. Ainsi, ce token est enregistré en base de données en le liant à l'e-mail du compte et la date d'expiration qui, à son tour, est vérifié lors de l'accès au lien afin de définir sa validité.

## Chapitre 3

# Front-end

### 3.1 Structure du site et style

Pour la partie interface utilisateur, nous avons réalisé des pages web en JSP. Nous avons pour consigne de faire en sorte que celles-ci soient responsive grâce à Bootstrap. L'utilisation de scripts JSP représentaient un réel avantage dans cette tâche car elle permettait de rendre modulable le site tout entier grâce aux liaisons avec les servlets correspondants. Chaque page, selon un lien cliquable, peut afficher un nouveau champ et modifier des éléments du Back-end. Par ailleurs, nous avons défini deux fichiers JSP, servant de header et de footer. Nous les avons appelés dans la plupart de nos pages. Grâce au Header, il est aisé de se connecter ainsi que se déplacer à travers le site, selon l'utilisateur connecté. Le Footer, quant à lui, permet de trouver les mentions légales, un descriptif du projet et les contacts.

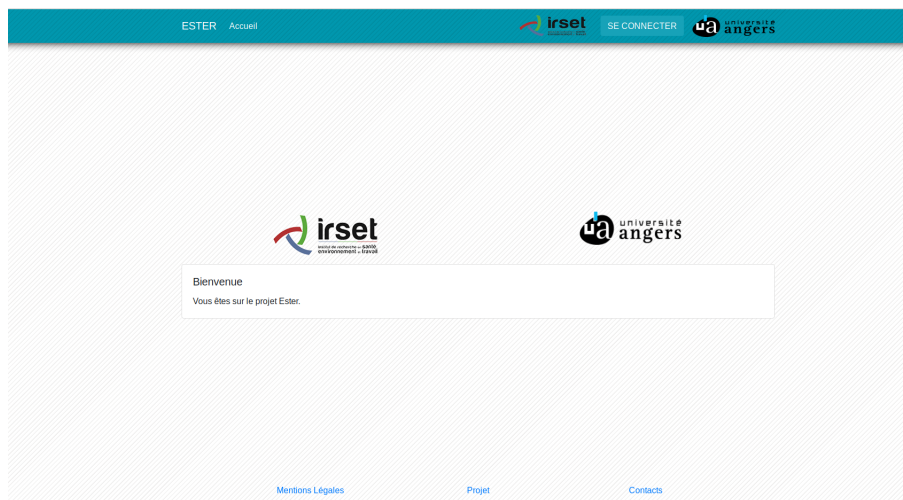


FIGURE 3.1 – Page d'Accueil ESTER

Afin de donner un exemple concernant l'aspect modulable de nos pages, nous pouvons regarder la page de Connexion. Sur cette dernière, en fonction du bouton sélectionné ("Salarié"/"Entreprise"/"Utilisateur"<sup>1</sup>), vous aurez des champs de saisie différents. Puis, une fois la vérification effectuée, l'utilisateur du site sera redirigé automatiquement vers sa page.

Une fois connecté, l'utilisateur se retrouve sur la page qui correspond à son statut. Il peut choisir ce qu'il souhaite faire à l'aide des liens dans le Menu à gauche. Pour certains liens, du contenu s'ajoutera sur sa page s'il les sélectionne. Il sera également redirigé vers une nouvelle page correspondante pour d'autres.

---

1. Fait référence aux utilisateurs médicaux et aux administrateurs; par opposition aux entreprises et salariés.

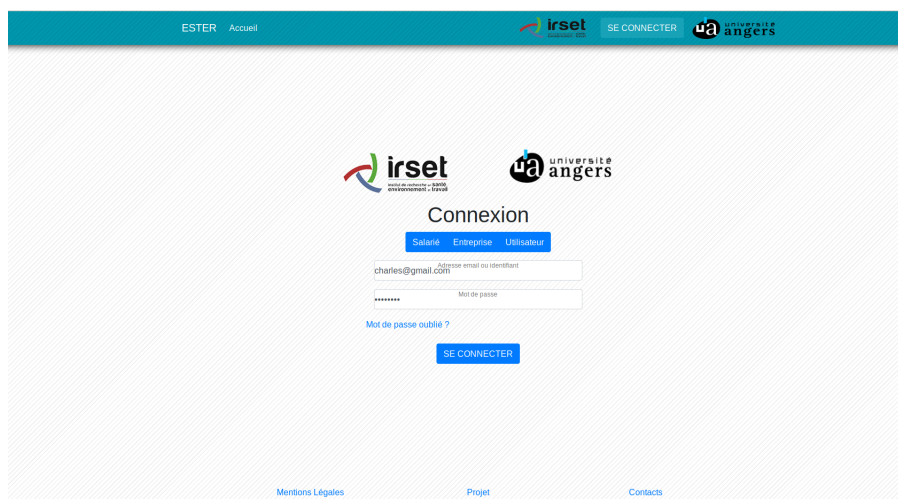


FIGURE 3.2 – Page de Connexion

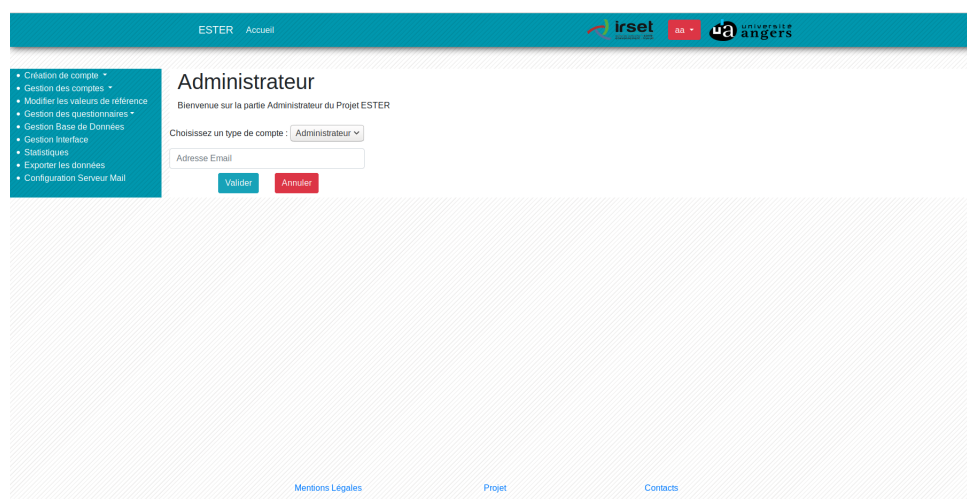


FIGURE 3.3 – Page de l'Administrateur

## 3.2 Questionnaires

### 3.2.1 Création de questionnaires

La problématique consistait à proposer un outil de création de questionnaires facile à utiliser. Cet outil était un besoin fondamental dans le projet qui nous a été donné. La solution que nous avons adoptée pour répondre au besoin formulé par le client, était de créer un générateur de questionnaires. L'utilisateur soignant, à l'aide de celui-ci, peut créer des formulaires HTML en glissé-déposé. Étant donné que les utilisateurs de cet outil sont des non-informaticiens, nous avons adapté cet outil aux besoins de ces derniers pour le rendre plus facile aussi bien à l'utilisation qu'à la compréhension de son fonctionnement.



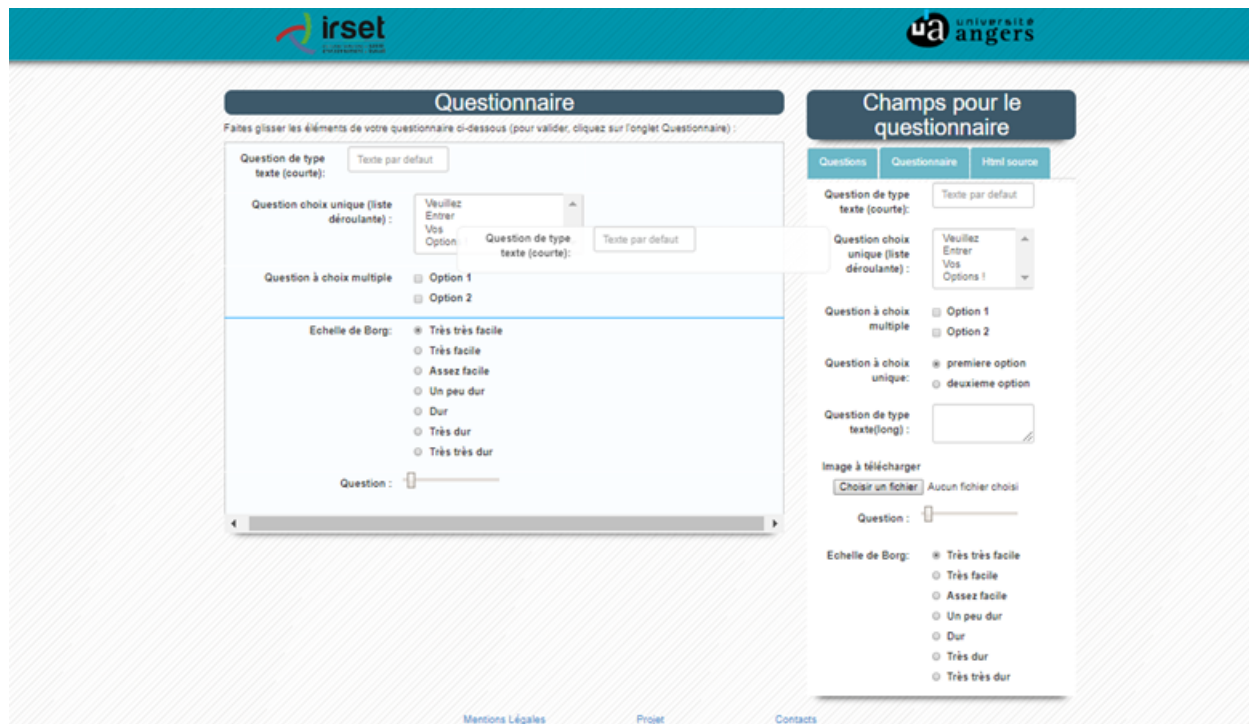


FIGURE 3.4 – Ajout d'une question dans le générateur de questionnaires

Cette interface permet à l'utilisateur de créer un questionnaire et paramétrer ses questions en glissant les questions de la partie droite vers la partie gauche et en cliquant sur la question déposée, un nouveau formulaire apparaît avec un nombre de zones de texte différent selon le type de question. Ce formulaire permet de modifier ou remodeler la question.

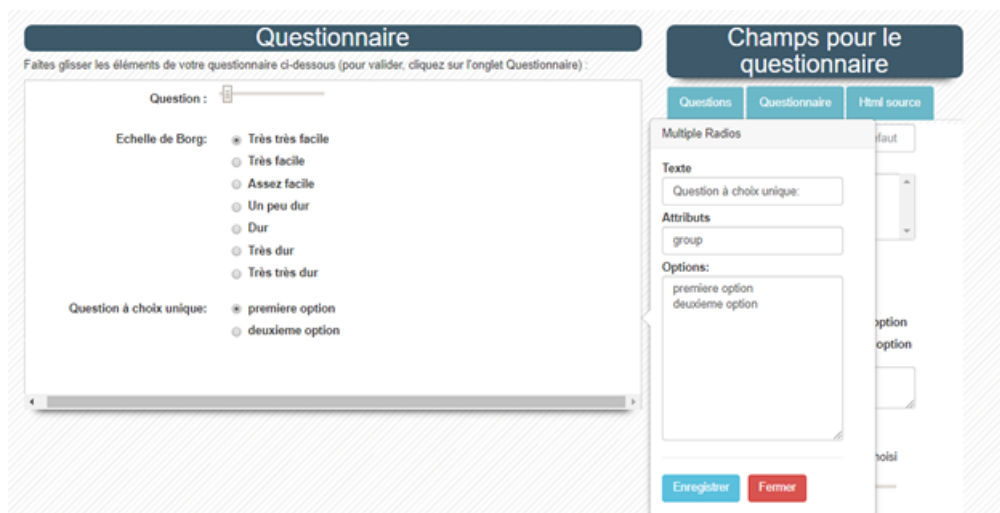


FIGURE 3.5 – Modification d'une question

Pour supprimer une question, il suffit de glisser la question hors du cadre. Cela permet de la retirer des autres questions.

**Questionnaire**

Faites glisser les éléments de votre questionnaire ci-dessous (pour valider, cliquez sur l'onglet Questionnaire) :

Question :

Question à choix unique :

- ☒ première option
- ☐ deuxième option

Echelle de Borg :

- ☒ Très très facile
- ☐ Très facile
- ☐ Assez facile
- ☐ Un peu dur
- ☐ Dur
- ☐ Très dur
- ☐ Très très dur

**Champs pour le questionnaire**

Questions | Questionnaire | HTML source

Question de type texte (courte) :

Question choix unique (liste déroulante) :

Question à choix multiple :

- ☐ Option 1
- ☐ Option 2

Question à choix unique :

- ☒ première option
- ☐ deuxième option

Question de type texte(long) :

Image à télécharger :

Choisir un fichier : Aucun fichier choisi

Question :

FIGURE 3.6 – Suppression d'une question

Lorsqu'on souhaite repositionner les questions, nous avons fait en sorte qu'il suffise de glisser puis déposer les questions à l'endroit où nous souhaitons la placer (en restant dans le même cadre).

**Questionnaire**

Faites glisser les éléments de votre questionnaire ci-dessous (pour valider, cliquez sur l'onglet Questionnaire) :

Question de type texte(long) :

Question à choix multiple :

- ☐ Option 1
- ☐ Option 2

Echelle de Borg :

- ☐ Très très facile
- ☐ Très facile
- ☐ Assez facile
- ☐ Un peu dur
- ☐ Dur
- ☐ Très dur
- ☐ Très très dur

Question à choix unique :

- ☐ première option
- ☐ deuxième option

**Questionnaire**

Faites glisser les éléments de votre questionnaire ci-dessous (pour valider, cliquez sur l'onglet Questionnaire) :

Question de type texte(long) :

Question à choix multiple :

- ☐ Option 1
- ☐ Option 2

Echelle de Borg :

- ☐ Très très facile
- ☐ Très facile
- ☐ Assez facile
- ☐ Un peu dur
- ☐ Dur
- ☐ Très dur
- ☐ Très très dur

Question à choix unique :

- ☐ première option
- ☐ deuxième option

**Questionnaire**

Faites glisser les éléments de votre questionnaire ci-dessous (pour valider, cliquez sur l'onglet Questionnaire) :

Question de type texte(long) :

Question à choix multiple :

- ☐ Option 1
- ☐ Option 2

Echelle de Borg :

- ☐ Très très facile
- ☐ Très facile
- ☐ Assez facile
- ☐ Un peu dur
- ☐ Dur
- ☐ Très dur
- ☐ Très très dur

Question à choix unique :

- ☐ première option
- ☐ deuxième option

FIGURE 3.7 – Repositionnement des questions

Pour finaliser la création d'un questionnaire, il faut que l'utilisateur saisisse le nom et l'identifiant du questionnaire puis en cliquant sur le bouton "enregistrer". Le questionnaire va être sauvegardé en deux formes différentes, la première est sous forme d'un fichier HTML (figure 3.9) et la deuxième est dans la base de données qu'on peut consulter à partir de la liste des questionnaires (figure 3.10).

**Questionnaire**

Faites glisser les éléments de votre questionnaire ci-dessous (pour valider, cliquez sur l'onglet Questionnaire) :

Question choix unique (liste déroulante) :

Question à choix multiple :

- ☐ Option 1
- ☐ Option 2

Question à choix unique :

- ☒ première option
- ☐ deuxième option

Question de type texte(long) :

Echelle de Borg :

- ☒ Très très facile
- ☐ Très facile
- ☐ Assez facile
- ☐ Un peu dur
- ☐ Dur
- ☐ Très dur
- ☐ Très très dur

Question :

**Champs pour le questionnaire**

Questions | Questionnaire | HTML source

Nom du questionnaire :

Identifiant du questionnaire :

Enregistrer

FIGURE 3.8 – Saisie de données pour finaliser la sauvegarde

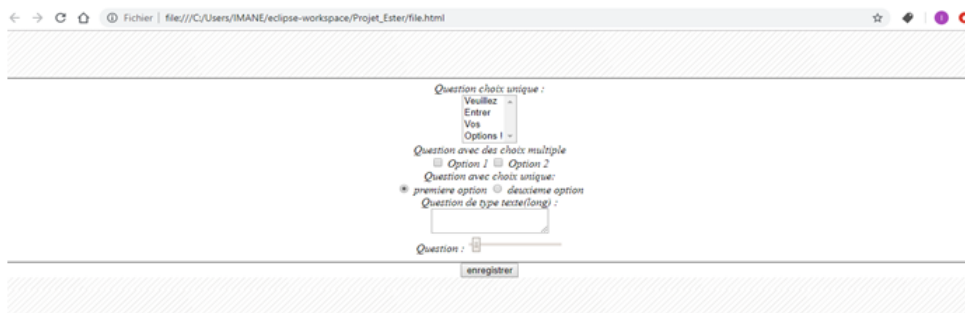


FIGURE 3.9 – Sauvegarde sous forme d'un formulaire HTML

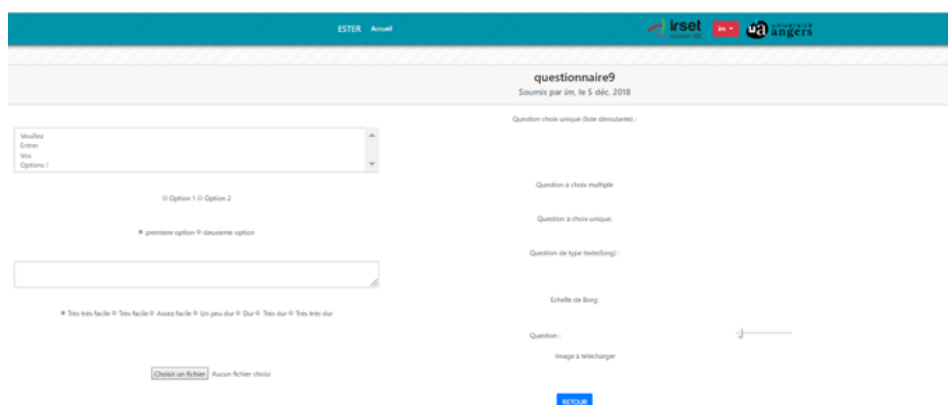


FIGURE 3.10 – Affichage du questionnaire enregistré dans la base de données

### 3.2.2 Questionnaire Eval\_Risk\_TMS

Dans le cahier des charges, il était spécifié qu'un questionnaire devait impérativement être créé : Eval\_Risk\_TMS est un questionnaire relatif à la médecine du travail. Il fut le premier créé, sous forme d'un script HTML. Il a été enregistré dans un fichier HTML ainsi que dans la base de données en tant que chaîne de caractères. Il est ensuite appelé dans la partie d'affichage des utilisateurs où les salariés peuvent y répondre ou encore le personnel soignant peut regarder les pages, une par une, dont Eval\_Risk\_TMS.



FIGURE 3.11 – Logo de Santé publique France

### 3.3 Résultats

L'implémentation de base de Highcharts a été assez simple, mais la création des lignes de séparation (exemple "D'accord vs Pas d'accord") et de coche pour les réponses du salarié ont demandé plus de temps.

Ce temps en plus est dû au temps qu'a nécessité l'adaptation à nos besoins puisque certains n'avaient pas des fonctions par défaut pour effectuer les tâches citées ci-dessus.

#### 3.3.0.1 DWR

DWR (Direct Web Remoting) est une librairie Java qui permet de recevoir des résultats du serveur sur le principe Ajax (asynchronous JavaScript and XML). Celui-ci permet de faire des requêtes aux serveur, de manière simplifiée.

Cette librairie nous permet de faire le lien entre Highcharts et notre serveur pour récupérer les informations telles que les questions et leurs valeurs (réponses et pourcentages).

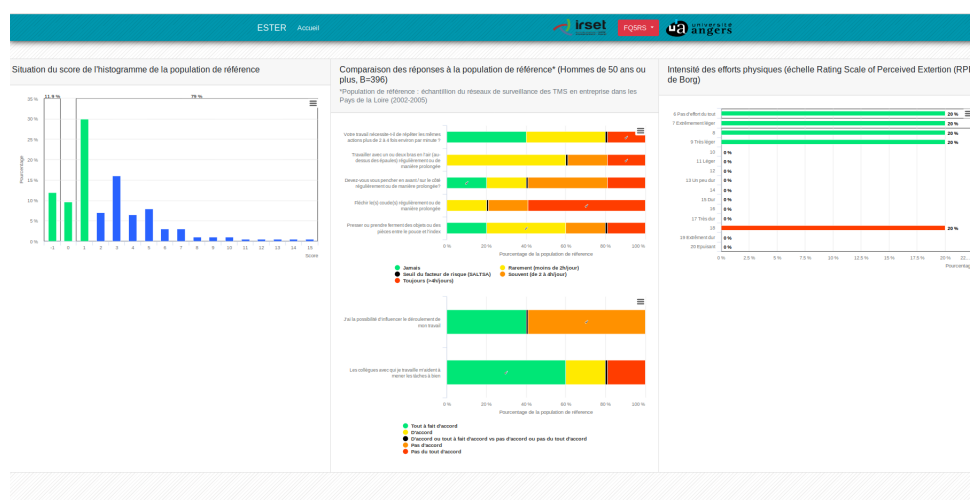


FIGURE 3.12 – Mise en forme des résultats finaux rendus

## Chapitre 4

# Problèmes rencontrés et perspectives

### 4.1 Problèmes au cours du projet

#### 4.1.1 Modification des Scripts

Dans un premier temps, nous avons créé des premières pages HTML afin d’avoir un rendu visuel et de définir un premier style CSS servant à voir, pour le développement, à quoi s’apparenterait les pages finales.

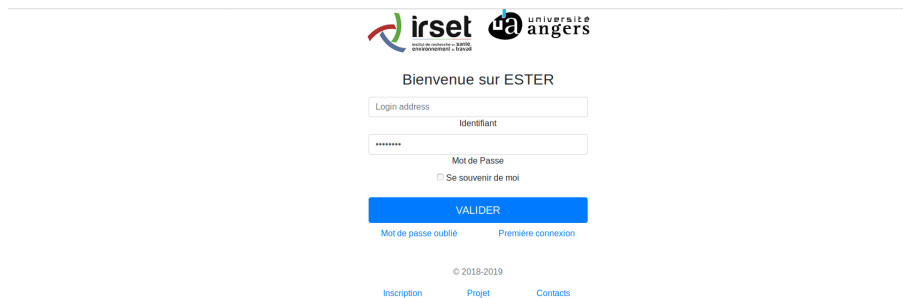


FIGURE 4.1 – Ancienne Page de Connexion

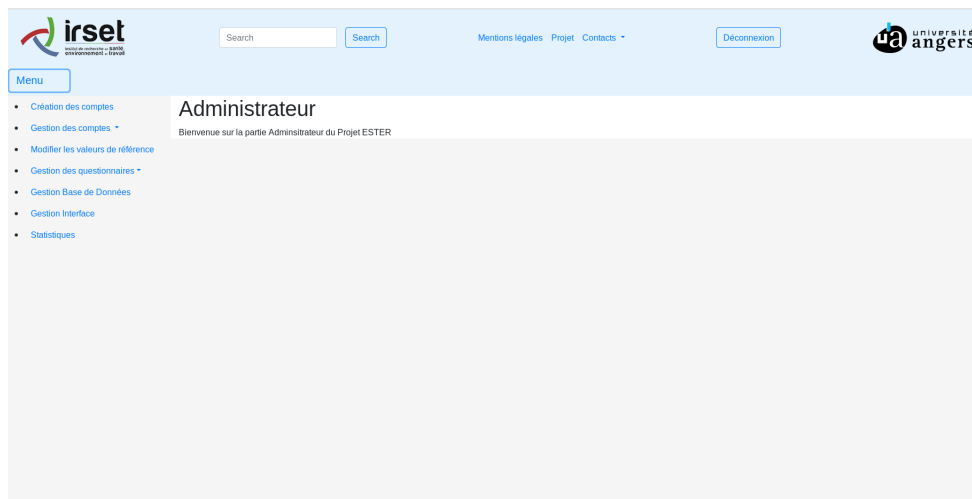


FIGURE 4.2 – Ancienne Page de l'Administrateur

Au sein de nos pages, nous avons défini des liens permettant d'appeler des pop-up de Bootstrap. Ces Modals devaient nous permettre d'une manière élégante de créer des comptes ou encore d'appeler la partie de modification de l'interface utilisateur par l'administrateur.

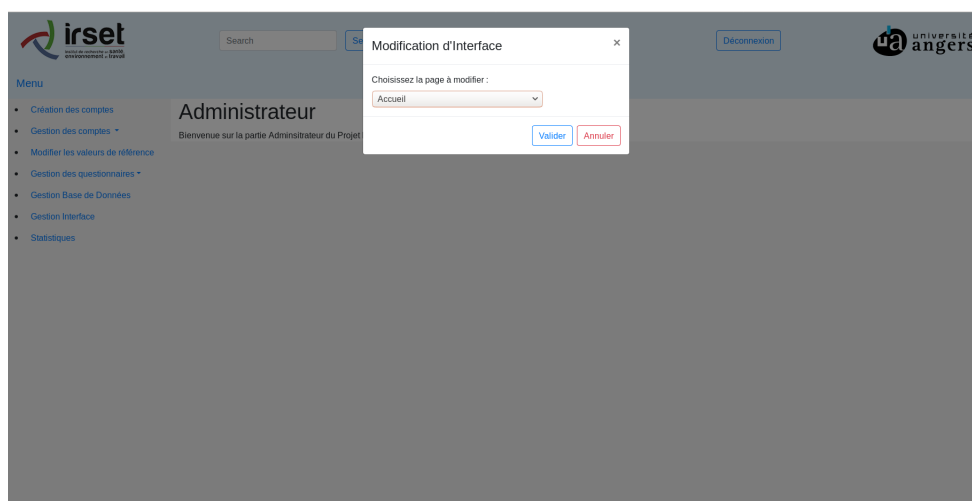


FIGURE 4.3 – Modal avec Bootstrap

L'interface de modification graphique permettait à l'administrateur de modifier à loisir chaque page du site comme il le souhaitait. Grâce à JQuery, nous récupérions la page choisie dans un modal et l'interface permettait de faire glisser de nouveaux éléments dans la page selon le désir de son utilisateur. Il pouvait également les supprimer ou les redimensionner, tout en gardant la page responsive.

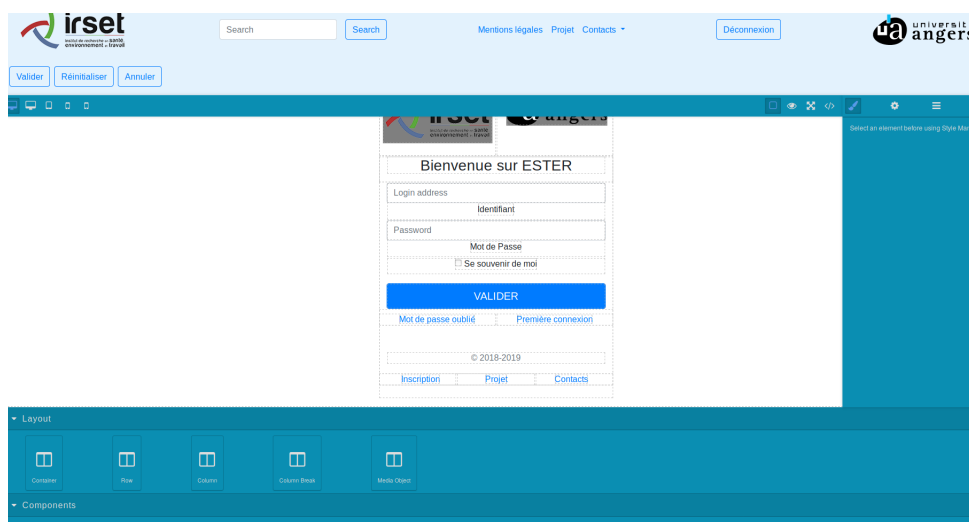


FIGURE 4.4 – Interface de modification graphique pour l’administrateur

Toutefois, les scripts HTML ont fini remplacés par des scripts JSP car plus avantageux comme dit plus tôt. De ce fait, les modals et l’interface utilisateur, n’étant plus compatibles avec les nouveaux scripts, ont été laissés. D’un autre côté, nous n’avons pas tous défini de la même manière nos styles CSS. Ce fut les membres d’ESTER qui décidèrent au cours de la réunion de novembre du visuel qu’ils préféraient. Nous avons ensuite appliqué les modifications voulues et directement réécrit dans des scripts JSP. Cependant, ces changements nous ont coûté un peu de temps qui aurait pu nous être utile dans l’implémentation d’autres fonctionnalités.

#### 4.1.2 Versions Bootstrap et JQuery

N’étant pas tous partis de la même origine, d’un même code, nous avons développé chacun nos tâches à part tel que nos chefs de projet nous les donnaient. Cependant, lors de l’intégration commune en novembre, nous nous sommes aperçus de certaines incompatibilités. Certaines parties étaient développées avec la dernière version de Bootstrap et la dernière de JQuery, tandis que d’autres utilisaient la précédente. De ce fait, il fut plus compliqué, plus laborieux de réunir chaque partie et de faire correspondre chaque page avec nos propres normes.

### 4.2 Perspectives d’amélioration

Malgré les nombreuses semaines à travailler sur ce projet, nous n’avons pas été en mesure de finir totalement le projet. Avec les difficultés listées précédemment et les tâches de plus en plus nombreuses, nous n’avons pu que faire au mieux. Le projet que nous avons à vous proposer, correspond donc avant tout aux principales fonctionnalités attendues. Toutefois, il serait envisageable d’intégrer davantage les entreprises dans le site puisqu’elles sont quelque peu délaissées. Également, les questionnaires créés par les médecins et administrateurs ne prennent pas en compte les scores qui seraient grandement utiles dans les calculs des statistiques. Nous pouvons penser qu’à terme, il serait remis une interface de modification graphique. Bien qu’elle fut avant tout penser en HTML, il devrait être possible de la réutiliser en correspondance avec le JSP actuel ou bien d’en recréer une nouvelle plus légère.

# Conclusion

Ce projet fut pour nous une véritable nouveauté. Encadrés par nos chefs de projets avec qui nous avons eu plusieurs réunions et avec qui nous avons souvent échangé, nous avons pu développer un site web relativement important. En dépit des difficultés rencontrées, nous avons essayé de mener notre travail jusqu'à son terme. Le projet ESTER fut pour nous l'occasion de découvrir certaines technologies et de les mettre en application, telles que Bootstrap ou surtout JEE.

Le cycle de développement que nous avons suivis, nécessitait une bonne organisation. Que ce soit de définir la base de données, de choisir les technologies ou d'implémenter les fonctionnalités dans le temps imparti, il était impossible de passer outre une certaine rigueur dans le planning. La réalisation de ce site nous a également demandé de faire preuve d'une certaine adaptation et de coordination. Nous avons pu constater que travailler dans son coin sur un projet d'une telle importance, n'est pas possible, tout comme prendre les tâches les unes après les autres sans compter sur celles à venir. Nous avons partagé nos connaissances, nos visions sur le projet et à ce titre, nous avons pu bien plus appréhender le sujet.

Par manque de temps, nous n'avons pas pu faire aboutir totalement le site. Tout de même, nous sommes parvenus à créer une interface utilisateur semblant correspondre aux attentes de notre client. Nous avons développé un générateur de questionnaires que les médecins pourront prendre en main assez aisément et enregistrer leur travaux. Les salariés peuvent répondre aux questionnaires et les utilisateurs sont en mesure d'accéder aux résultats calculés selon les réponses données par l'utilisateur. Enfin, plusieurs autres fonctionnalités ont été ajoutées aux principales afin que les utilisateurs puissent mieux s'approprier les pages de notre site.

Vous trouverez à ce lien le code du site web que nous avons développé : [https://github.com/nordinaryguy/Projet\\_ESTER](https://github.com/nordinaryguy/Projet_ESTER).



# Table des figures

1.1	Logo de MongoDB . . . . .	4
1.2	Diagramme de Gantt . . . . .	5
2.1	Tableau de la collection Utilisateur . . . . .	6
2.2	Tableau de la collection Questionnaire . . . . .	7
2.3	Tableau de la collection Réponse . . . . .	7
2.4	E-mail pour la première connexion . . . . .	8
2.5	Générateur de codes patients . . . . .	8
2.6	Formulaire du patient à la première connexion . . . . .	9
3.1	Page d'Accueil ESTER . . . . .	11
3.2	Page de Connexion . . . . .	12
3.3	Page de l'Administrateur . . . . .	12
3.4	Ajout d'une question dans le générateur de questionnaires . . . . .	13
3.5	Modification d'une question . . . . .	13
3.6	Suppression d'une question . . . . .	14
3.7	Repositionnement des questions . . . . .	14
3.8	Saisie de données pour finaliser la sauvegarde . . . . .	14
3.9	Sauvegarde sous forme d'un formulaire HTML . . . . .	15
3.10	Affichage du questionnaire enregistré dans la base de données . . . . .	15
3.11	Logo de Santé publique France . . . . .	15
3.12	Mise en forme des résultats finaux rendus . . . . .	16
4.1	Ancienne Page de Connexion . . . . .	17
4.2	Ancienne Page de l'Administrateur . . . . .	18
4.3	Modal avec Bootstrap . . . . .	18
4.4	Interface de modification graphique pour l'administrateur . . . . .	19

# Bibliographie

- [1] Bcrypt. Bcrypt java library and cli tool. <https://github.com/patrickfav/bcrypt>. [En ligne ; accédé le 7 décembre 2018].
- [2] Bootstrap. Documentation. <https://getbootstrap.com/docs/4.1/getting-started/introduction/>. [En ligne ; accédé le 7 décembre 2018].
- [3] grapesJS. Free and open source web builder framework. <https://grapesjs.com/>. [En ligne ; accédé le 7 décembre 2018].
- [4] Highcharts. Documentation. <https://www.highcharts.com/docs>. [En ligne ; accédé le 7 décembre 2018].
- [5] OpenClassroom. Créez votre application web avec java ee. <https://openclassrooms.com/fr/courses/626954-creez-votre-application-web-avec-java-ee>. [En ligne ; accédé le 7 décembre 2018].
- [6] OpenClassroom. Prenez en main bootstrap. <https://openclassrooms.com/fr/courses/1885491-prenez-en-main-bootstrap>. [En ligne ; accédé le 7 décembre 2018].