



Création d'un service de ticket sur Laravel
(Stage en entreprise du 31 janvier au 18 mars 2022)
BTS Services Informatiques aux Organisations

Entreprise d'accueil	Inovyn France, site de Tavaux
Maître de stage	M.Stéphane Pariset
Professeur référent	M.Michel Girard



Remerciements

Tout d'abord je remercie le groupe Inovyn de m'avoir accueilli pour faire mon stage de deuxième année au sein de leur entreprise.

Je remercie grandement mon maître de stage, Mr Stéphane Pariset, responsable Lims (Système informatisé de gestion de laboratoire), pour m'avoir aidé et accompagné durant ces 7 semaines dans une bonne ambiance.

Je remercie vivement Mr Stéphan Leruth technicien SI infrastructure, pour avoir été présent pour régler certains problèmes techniques.

Je remercie également Mr Carl Hervier, cadre du Laboratoire de Contrôle de Matières Plastiques, pour son accueil et ses conseils avisés.

Je remercie toutes les personnes rencontrées à Inovyn pour leur générosité et leur accueil durant ces 7 semaines de stage.

Je suis également reconnaissant envers Mr Gilles Darago, Site Manager chez Solvay à Jemeppe sur Sambre et Jean-noël Daloz, SAP Industrial Manager chez Inovyn, pour avoir appuyé ma demande de stage. Un grand merci à eux.

Enfin Je remercie mon professeur référant Mr Michel Girard pour son suivi et ses remarques pertinentes.

Sommaire

Table des matières

REMERCIEMENTS	2
GLOSSAIRE	4
INTRODUCTION	5
I-PRESENTATION DE L'ENTREPRISE	6
II-MA MISSION.....	10
III-MES REALISATIONS	12
IV-BILAN.....	21
LISTE DES FIGURES	22
ANNEXE 1 : VUE AERIENNE DU SITE DE TAVAUX	23
ANNEXE 2 : SCHEMA DU FONCTIONNEMENT DE GTL.....	23
ANNEXE 3 : DIAGRAMME DE LA BASE DE DONNEES TICKETING.....	24

Glossaire

LabWare	Système de gestion de l'information de laboratoire et des cahiers électroniques de laboratoire.
Lims	Système informatisé de gestion de laboratoire
Sql	Structured Query Language est un langage permettant de communiquer avec une base de données.
Bdd	Base de données
view	Page de mon site
Side bar	Barre de menu latérale
fr	Identifiant chez Inovyn
PVC	Polychlorure de vinyle
PVDC	Polychlorure de vinylidène
PVDF	Polyfluorure de vinylidène
VCM	Chlorure de vinyle monomère
UFP	Unités de Fabrication de Polymère
POC	Produits Organiques Chlorés
fr	Identifiant chez Inovyn
mdp	Mot de passe

Introduction

Dans le cadre de mon BTS services informatiques aux organisations, au lycée privé Saint-Bénigne à Dijon en deuxième année, j'ai réalisé un stage de deux mois du 31 janvier au 18 mars 2022 au sein de l'entreprise Inovyn implantée à Tavaux dans le Jura (39).

Inovyn produit sur le site de Tavaux de nombreux produits chimiques (chlore, VCM, soude caustique, organiques chlorés) et du PVC. Le numérique étant en constante évolution et le laboratoire engendrant énormément de données, il faut souvent résoudre des problèmes liés aux logiciels (bugs, nouvelle fonction, modification...).

Lors de mon stage j'ai été affecté dans le service d'analyse de l'usine où j'ai aidé Monsieur Stéphane Pariset (responsables Lims) à mettre en place une solution logicielle à destination des employés d'Inovyn.

Il m'a été demandé de créer un site-web de gestion de tickets à destination des employés d'Inovyn, pour avoir un aperçu direct de tous les problèmes liés au LabWare (logiciel qui gère les données d'échantillons) ou autre logiciel.

Dans un premier temps nous décrirons l'entreprise Inovyn, puis dans un second temps nous étudierons mes différentes réalisations durant ma mission. Enfin nous dresserons un bilan de ce stage en entreprise du point de vue professionnel puis personnel.

I-Présentation de l'entreprise

1. Maison mère : Le groupe Ineos

Ineos est une société anglaise née en 1998. Ineos est un fabricant mondial de produits pétrochimiques, spécialisé dans la chimie et les produits huileux. Le groupe possède une capacité de production de produits chimiques de 66 millions de tonnes.

La société dispose de 194 sites de fabrication dans le monde, dans 29 pays. Elle génère 61 milliards de dollars de revenu annuel, grâce à ses 26 000 employés. Elle est la maison mère de 36 entreprises différentes.

En 2015, une coentreprise entre les chimistes Solvay et Ineos, nommée Inovyn, est créée. Au 1^{er} juillet 2016 Ineos rachète les parts de Solvay, Inovyn devient donc une filiale appartenant 100% à Ineos.

Le site Solvay de Tavaux ([voir annexes : 1](#)) devient à compter de cette date une plateforme industrielle, accueillant deux principaux exploitants : Inovyn qui couvre la chaîne des produits chlorovinylés fabriqués à Tavaux (PVC, soude caustique, solvants chlorés, ...) et Solvay Tavaux, centré sur la chimie de spécialités (PVDC, PVDF, organiques fluorés).

En 2016, Solvay se désengage de cette coentreprise, les installations chlorovinylées du site de Tavaux appartiennent donc désormais à Inovyn, et les autres installations à Solvay.



Figure 1 : Présentation des sites Ineos

2. L'entreprise Inovyn

2.1. Chiffres clés

Formé en juillet 2015, Inovyn est le leader Européen de PVC et se classe dans le top 3 mondial des producteurs de vinyles. Avec un chiffre d'affaires annuel de 3,1 milliards d'euros, Inovyn compte près de 4200 employés répartis sur 15 sites de production dans 8 pays Européen

2.2. Production

Le portefeuille d'activités d'Inovyn est organisé à travers 5 groupes de produits clés : Dérivés Organiques Chlorés, Chlore, Vinyles Application Générale, Vinyles Spéciaux et Technologies électrochimiques & vinyliques.



Figure 2 : Représentation des cinq groupes d'activités d'Inovyn

Inovyn fabrique une vaste gamme de produits chimiques utilisés comme matières premières dans de nombreux secteurs industriels :



Figure 3 : Utilisation des produits Inovyn

3. Le site de Tavaux

3.1. Chiffres clés

La plateforme industrielle de Tavaux est spécialisée dans la fabrication de produits chimiques et de matières plastiques nécessaires aux industries.

L'usine Inovyn de Tavaux est l'une des plus importantes du groupe, par sa taille, ses effectifs ainsi que la diversité des produits qui y sont fabriqués. Elle compte 800 employés. Le site est partagé avec Solvay qui compte environ 600 employés.

Les sociétés Inovyn et Solvay et quelques entreprises extérieures sont implantées à Tavaux sur un parc industriel d'une superficie de 200 ha, avec 32 km de route et 35 km de voies ferrées. Inovyn Tavaux fabrique et expédie plus d'un million de tonnes de produits chaque année dont 250 000 tonnes de PVC. Le site est classé Seveso seuil haut.

3.2. Organisation de l'entreprise

L'usine Inovyn de Tavaux est constituée de plusieurs pôles, services et secteurs. Le service analyse se divise en quatre grands secteurs :

- ❖ Laboratoire de Contrôle des matières Plastiques : Analyse des matières plastiques issues du site de Tavaux et de clients extérieurs.

- ❖ Contrôle fabrication : Analyses organiques chimiques, pôle des prélèvements, potentiomètre, UFP, POC.

- ❖ Spectroscopie et environnement, hygiène et qualité, analyse extérieure : Laboratoire environnement et hygiène industrielle, spectroscopie.

❖ Support et coordinateur : Laboratoire de chromatographie en phase vapeur pour des analyses environnementales et des composés organiques ; analyses diverses.

Toutes les analyses sont gérées informatiquement grâce à un Lims nommé LabWare
C'est dans cet environnement que se déroule mon stage.

II-Ma mission

1.Objectif du stage

1.1. Présentation du projet

Ma mission est de créer un système de tickets pour le Lims LabWare comme il en existe dans les services IT, ces tickets contiendront les informations utiles aux personnes qui les prendront en charge, pièce jointe, une description etc...

Cette application est destinée à être quotidiennement utilisée par les employés d'Inovyn pour faire remonter des problèmes ou requêtes.

Cette application se nommera GTL (Gestion des Tickets LabWare).

Pourquoi créer un service de tickets ?

Aujourd'hui, aucun système de suivi des demandes soumises par les utilisateurs LabWare au programmeur n'existe, tout se fait par e-mail, ou oralement un peu dans le désordre. Le travail est quand même réalisé mais nous n'avons aucun historique, aucune archive facilement consultable de ce qui est fait dans LabWare.

Quelle est la finalité ?

- Supprimer les e-mails créés en désordre pour les demandes LabWare.
- Consulter facilement le statut d'une demande en cours et suivre sa résolution sans échange d'e-mail ou consulter l'historique d'une demande clôturée par recherche par mot-clé. Un n° de ticket étant difficile à retenir.
- Système centralisé accessible à tous.

1.2. Fonctionnement du projet ([voir annexes : 2](#))

Pour créer, modifier ou consulter un Ticket, la personne, utilisateur LabWare ou Admins, devra s'identifier à l'ouverture de l'application. L'administrateur aura la gestion de tous les tickets mais pourra également en créer, à contrario l'utilisateur pourra seulement consulter ses propres tickets. L'administrateur pourra attribuer les tickets à une personne définie, modifier les tickets, assurer leurs suivis (statuts, clôture, etc...).

2. Environnement de travail

L'application sera programmée sur l'éditeur de code Visual Studio Code, et réalisée avec le Framework open-source laravel 8 et php 8, l'architecture suivra donc le schéma modèle-vue-contrôler.

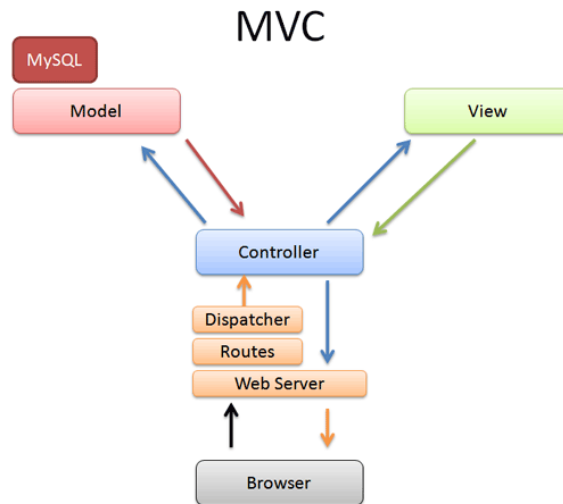


Figure 4 : Schéma modèle MVC

La gestion de la bdd se fera via SSMS (Microsoft SQL Server Management Studio).

L'application va être hébergée sur le serveur web IIS de développement d'Inovyn, gérée par le service informatique.

L'application sera déployée sur le serveur de production lorsqu'elle sera terminée.

III-Mes réalisations

1. Premiers pas

1.1. Réalisation de la base de données

La base de données est créée sous sql server et se nomme Ticketing, elle comprend 6 tables ([voir annexes : 3](#)) :

- user : gère tous les champs relatifs à l'utilisateur, administrateur compris
- ticket : gère tous les champs relatifs aux tickets
- mail : enregistre l'e-mail associé à un ticket
- suivi : enregistre les modifications apportées au ticket
- log : enregistre les logs
- type_log : gère les types de logs

L'ajout d'une table pour enregistrer les changements de mot de passe de chaque utilisateur se fera dans le futur, ainsi que l'amélioration de la table « suivi » car je n'ai pas eu le temps de commencer la fonctionnalité suivie qui suit l'historique de chaque ticket, la table n'en est qu'à sa première ébauche.

1.2. Création de la solution

Je crée ensuite le dossier laravel, et crée 3 Controllers :

- AdminController : gère les méthodes pour la partie administrateur
- ConnexionController : gère les méthodes de connexion etc... pour la partie utilisateur
- TicketController : gère les méthodes pour la création de ticket etc...

Ensuite je crée 4 modèles :

- Admin : exécute des requêtes sql spéciales pour les administrateurs
- Ticket : exécute des requêtes sql pour les tickets

- Utilisateur : exécute des requêtes sql pour les utilisateurs
- LogActivity : inscrit les différents logs dans la base de données

Je fais le lien entre ma bdd et mon site dans le fichier .env et je peux commencer à créer mes pages.

2. Page de Connexion

2.1 Création de la page

La page de connexion n'est pas une page très compliquée, il faut que l'utilisateur puisse saisir son fr et un mdp, et appuie sur le bouton de connexion.



Figure 5 : Design page de connexion

Pour la partie traitement, un formulaire est envoyé en méthode post et une méthode du controller ConnexionController récupère le fr et le mdp rentré.

Une requête est lancée pour trouver le mdp dans la base de données en fonction du fr. Ensuite je compare le mot de passe récupéré depuis la base de données avec le mot de passe rentré par l'utilisateur.

S'ils sont identiques l'utilisateur a accès à la suite du site sinon il obtient un message d'erreur et doit ressaisir son mot de passe et son fr, la même situation se produit si la requête ne trouve pas de mot de passe avec le fr rentré.

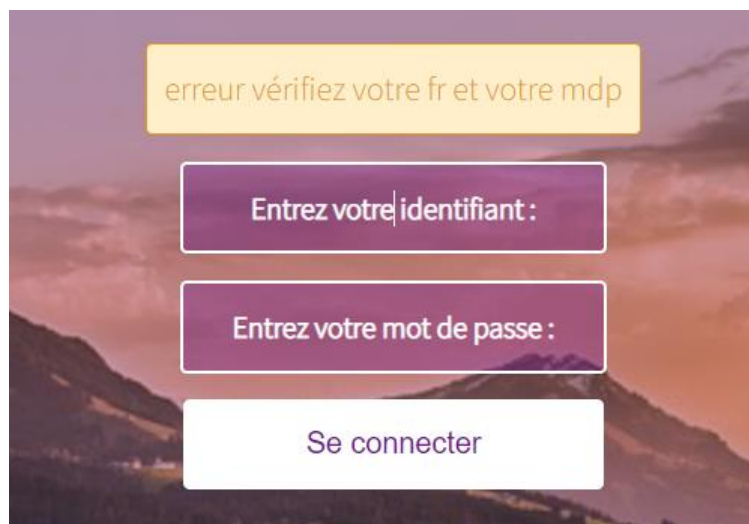


Figure 6 : Message d'erreur page de connexion

Bien sûr le message d'erreur ne dit pas si c'est le mdp ou le fr qui est faux, cela rajoute de la sécurité.

2.2 Sécurité du mot de passe

Pour éviter d'écrire les mots de passe en dur dans la base de données et rajouter de la sécurité il faut chiffrer les mots de passe.

Pour cela j'ai utilisé une fonction de php `sha1()` qui permet de hacher le mot de passe, lorsqu'elle reçoit une chaîne de caractères, elle la retourne sous la forme d'un nombre hexadécimal d'une taille de 40 caractères.

hello → aaf4c61ddcc5e8a2dabede0f3b482cd9aea9434d

Mais cette fonction peut facilement être contournée par des rainbow tables qui sont des bases de données de mots avec tous les hash pré-calculés, avec ça on peut retrouver le mot de passe.

Pour contourner ce problème il faut ajouter un grain de sel, c'est-à-dire une variable connue au début du mot de passe et à la fin et le hasher avec la fonction, comme cela même si le mot de passe est déchiffré grâce à une rainbow tables il n'est pas utilisable.

Grainhellosel → 303c4f5fc1991a09d76d711595aaf6a769a25e37

3. Affichage Utilisateur

3.1 Les tickets

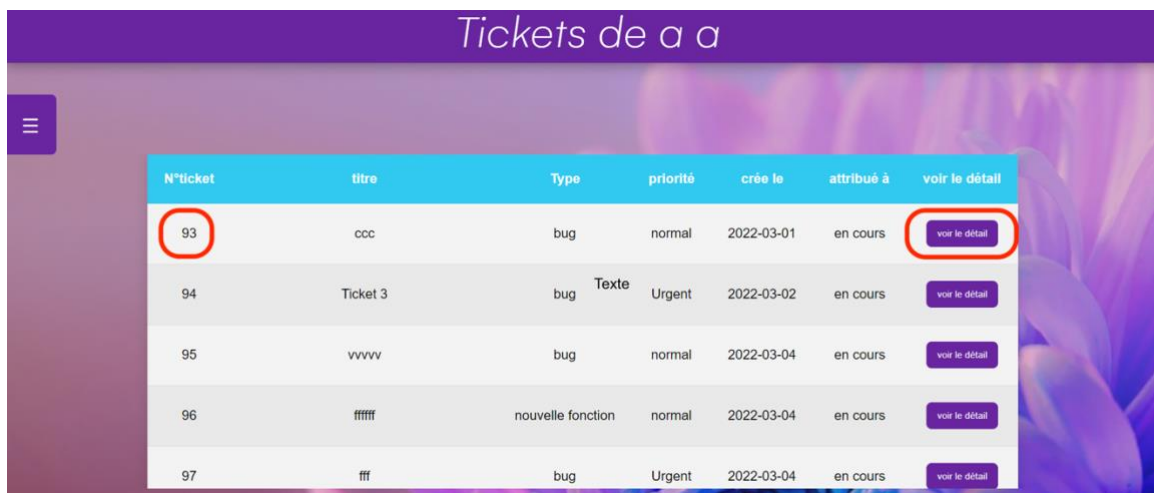
L'affichage de l'utilisateur comprend dans un premier temps, l'affichage des tickets de l'utilisateur, pour cela une requête va chercher dans la base de données tous les tickets ou l'id du user est celui de l'utilisateur connecté à la session. Pour cela il faut mettre l'id de l'utilisateur dans la session lors de sa connexion et réutiliser ce même id pour lancer la requête

```
//j'enregistre le fr l'id le nom complet et le statut dans la session pour pouvoir les réutiliser après
$request->session()->put('fr',$fr);
$request->session()->put('UserNom_Complet',$nom_complet);
$request->session()->put('statut',$statut);
$request->session()->put('UserId',$id);
```

Figure 7 : Enregistrement des infos de l'utilisateur connecté dans la session

Grâce au statut, j'oriente mon utilisateur soit vers l'affichage des utilisateurs, soit vers celui des administrateurs.

Pour voir le détail d'un ticket on peut cliquer sur le bouton « voir le détail » qui récupère l'id du ticket et l'envoie à une méthode du controller qui exécute une requête à la base de données pour récupérer toutes les informations d'un ticket suivant son id.



N°ticket	titre	Type	priorité	crée le	attribué à	voir le détail
93	ccc	bug	normal	2022-03-01	en cours	voir le détail
94	Ticket 3	bug Texte	Urgent	2022-03-02	en cours	voir le détail
95	vvvvv	bug	normal	2022-03-04	en cours	voir le détail
96	ffffff	nouvelle fonction	normal	2022-03-04	en cours	voir le détail
97	fff	bug	Urgent	2022-03-04	en cours	voir le détail

Figure 8 : Affichage des tickets de l'utilisateur connecté



N°ticket	titre	description	priorité	Type	progression	crée le	date souhaité	attribué à	date de clôture
93	ccc	cccc	normal	bug	0 %	2022-03-01	2022-03-02	en cours	

Figure 9 : Affichage du détail d'un ticket sélectionné

3.2 Création d'un nouveau ticket

Pour créer un nouveau ticket l'utilisateur doit fournir : un titre, une description, l'urgence ou pas, le type (déjà défini à l'avance), une date de réalisation souhaitée, il peut fournir une pièce jointe ou non.

Un récapitulatif du ticket sera obligatoirement envoyé à l'administrateur par e-mail. L'utilisateur peut choisir de l'envoyer à une personne de la société ou non.

Figure 10 : Formulaire de création de ticket

L'envoi de l'e-mail se fait grâce à une messagerie créée pour le projet, une view sert de patron et récupère toutes les informations rentrées grâce à la session.

L'envoi à une boîte e-mail se fait comme cela :


```
// envoie du mail
Mail::to('paul.bertelli@inovyn.com')->send(new MessageGoogle());
```

Figure 11 : Envoi à une adresse e-mail depuis le controller

```
<p>Nouveau Ticket n° {{ $idTicket=Session::get('idTicket')}} </p>
<p>Titre : {{ $idTicket=Session::get('titre')}} </p>
<p>type : {{ Session::get('type')}} </p>
<p>priorité : {{ Session::get('priorite')}} </p>
<p>description : {{ $idTicket=Session::get('description')}} </p>
<p>date de réalisation souhaitée : {{ Session::get('date_release_souhaitee')}} </p>
<p>fichier join : {{ Session::get('originalName')}} </p>
<p>email en copie : {{ Session::get('email')}} </p>
```

Figure 12 : View qui sert de patron pour la structure du mail

Il faut aussi créer une classe pour pouvoir envoyer l'e-mail en SMTP, dans cette classe on spécifiera la structure globale de l'e-mail : le sujet, le destinataire, etc...

```
$idTicket ="[GTL # $idTicket] Nouveau Ticket $TitreTicket.";
return $this->from("ticketing.LabWare@inovyn.com") // L'expéditeur
->subject($idTicket) // Le sujet
->view('emails.contactMail'); // La vue demise en forme de l'email
```

Figure 13 : Fonction de la classe MessageGoogle.php

Bien sûr toutes les données remplies dans le formulaire sont enregistrées dans la base de données, et un log est créé à chaque fois qu'un utilisateur crée un ticket. Pour cela une requête d'insertion dans la table log est déclenchée à chaque création de ticket.

3.3 Modifier le mot de passe

Pour que l'utilisateur puisse modifier son mot de passe, j'ai créé un formulaire classique avec demande d'ancien mot de passe et confirmation du nouveau. Pour gérer les inputs j'ai utilisé les règles de validations de laravel tels que : required, min, confirmed et des regex pour la conformité et la sécurité maximale du nouveau mot de passe. Donc laravel compare automatiquement les 2 mots de passe et vérifie qu'ils sont bien identiques, avant de les chiffrer avec la méthode du grain de sel vu précédemment et de les remplacer dans la bdd.

Nouveau mot de passe

Mot de passe actuel :

votre mdp actuel:

Nouveau mot de passe:
8 caractères, une majuscule, une minuscule et un chiffre au minimum.

Confirmation du mot de passe :

Confirmer

*vous allez être redirigé sur la page de connexion

Figure 14 : Formulaire pour modifier le mot de passe

Mot de passe actuel inexact

Nouveau mot de passe

Mot de passe actuel :

Figure 15 : Erreur du formulaire pour modifier le mot de passe

Les erreurs liées aux règles sont en anglais, j'ai donc créé un nouveau fichier de langue en français et j'ai seulement traduit les erreurs qui étaient utiles à mon cas.

3.4 Le menu et la déconnexion

Le menu est représenté par une side bar qui contient tous les liens vers les différentes fonctions de l'utilisateur vu précédemment.

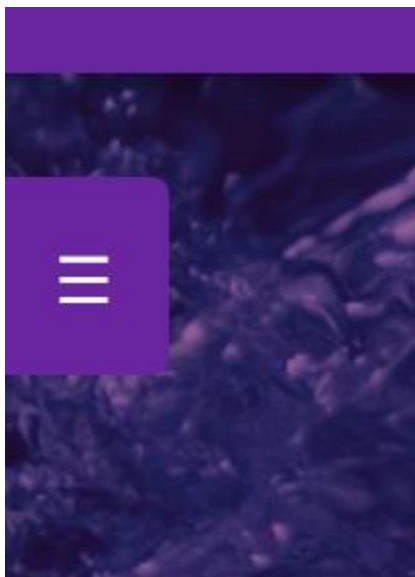


Figure 16 : Bouton ouverture side bar

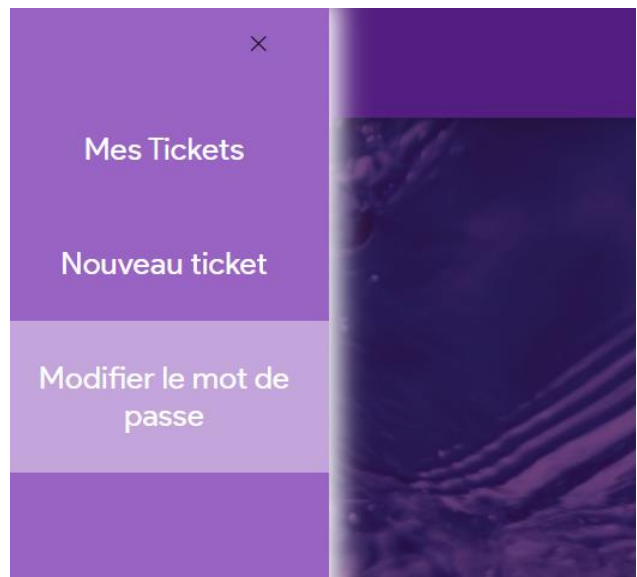


Figure 17 : Side bar fonctionnalités

Il y a aussi la fonction de déconnexion qui est directement en bas de la side bar et qui permet à l'utilisateur de se déconnecter du site et de revenir sur la page d'accueil.

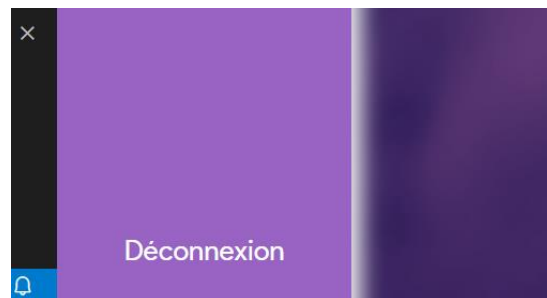


Figure 18 : Bouton de déconnexion

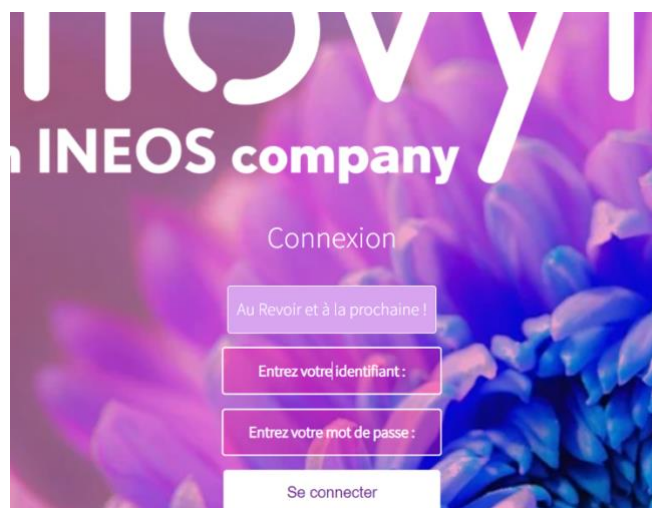


Figure 19 : Page d'accueil avec message de déconnexion

4. Affichage administrateur

Pour l'affichage administrateur, quelques fonctions sont gardées et d'autres partiellement modifiées. Pour l'affichage des tickets plus besoin d'id pour afficher le ticket de l'utilisateur car l'administrateur a accès à tous les tickets qui ont été créés. Il peut créer un ticket de la même manière que l'utilisateur, cette fonction ne change pas du tout et est identique pour les deux. Il peut modifier le mot de passe d'un utilisateur en entrant son fr et un nouveau mot de passe. Comme l'administrateur a plus de droits qu'un simple utilisateur il n'y a pas de vérification de mot de passe identique.

Les fonctions qui n'ont pas pu être réalisées sont : la recherche de ticket par mots clés et l'édition de ticket.

IV-Bilan

1. Conclusion professionnelle

La mission qui m'a été confiée pendant ce stage n'a pas pu être totalement terminée, faute de temps. Il reste à finir la recherche de ticket par mots clés et l'édition de ticket.

Bien sûr le site-web n'en est qu'à son premier jet, il y aura d'autres versions pour l'améliorer. Cela mis de côté le site est parfaitement fonctionnel pour le moment.

2. Conclusion personnelle

Ce stage a été une expérience très enrichissante, tant sur les différentes technologies informatiques utilisées pour le projet que sur les technologies servant à faire fonctionner une telle usine. J'ai apprécié de participer à ce projet utile, qui permettra à l'administrateur de mieux encadrer les demandes des employés. Ce contexte de travail a été stimulant pour moi et m'a poussé à donner le meilleur de moi-même.

J'ai aussi apprécié de visiter le service analyse car mon grand-père y a travaillé pendant 40 ans.

J'en ressors avec une idée plus claire du rôle de l'informatique et de son utilité dans une grande entreprise (mon premier stage ayant été fait en 2021 dans une TPE).

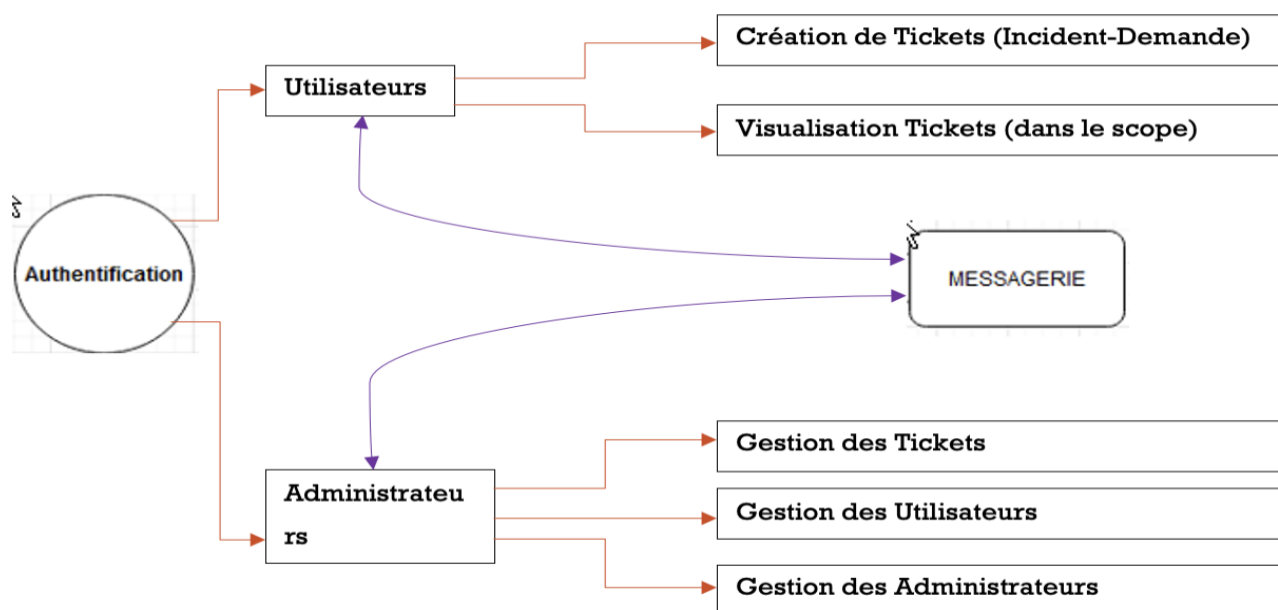
Liste des figures

- Figure 1 : Présentation des sites Ineos
- Figure 2 : Représentation des cinq groupes d'activités d'Inovyn
- Figure 3 : Utilisation des produits Inovyn
- Figure 4 : Schéma modèle MVC
- Figure 5 : Design page de connexion
- Figure 6 : Message d'erreur page de connexion
- Figure 7 : Enregistrement des infos de l'utilisateur connecté dans la session
- Figure 8 : Affichage des tickets de l'utilisateur connecté
- Figure 9 : Affichage du détail d'un ticket sélectionné
- Figure 10 : Formulaire de création de ticket
- Figure 11 : Envoi à une adresse e-mail depuis le controller
- Figure 12 : View qui sert de patron pour la structure du mail
- Figure 13 : Fonction de la classe MessageGoogle.php
- Figure 14 : Formulaire pour modifier le mot de passe
- Figure 15 : Erreur du formulaire pour modifier le mot de passe
- Figure 16 : Bouton ouverture side bar
- Figure 17 : Side bar fonctionnalités
- Figure 18 : Bouton de déconnexion
- Figure 19 : Page d'accueil avec message de déconnexion

Annexe 1 : Vue aérienne du site de Tavaux



Annexe 2 : Schéma du fonctionnement de GTL



Annexe 3 : Diagramme de la base de données Ticketing

