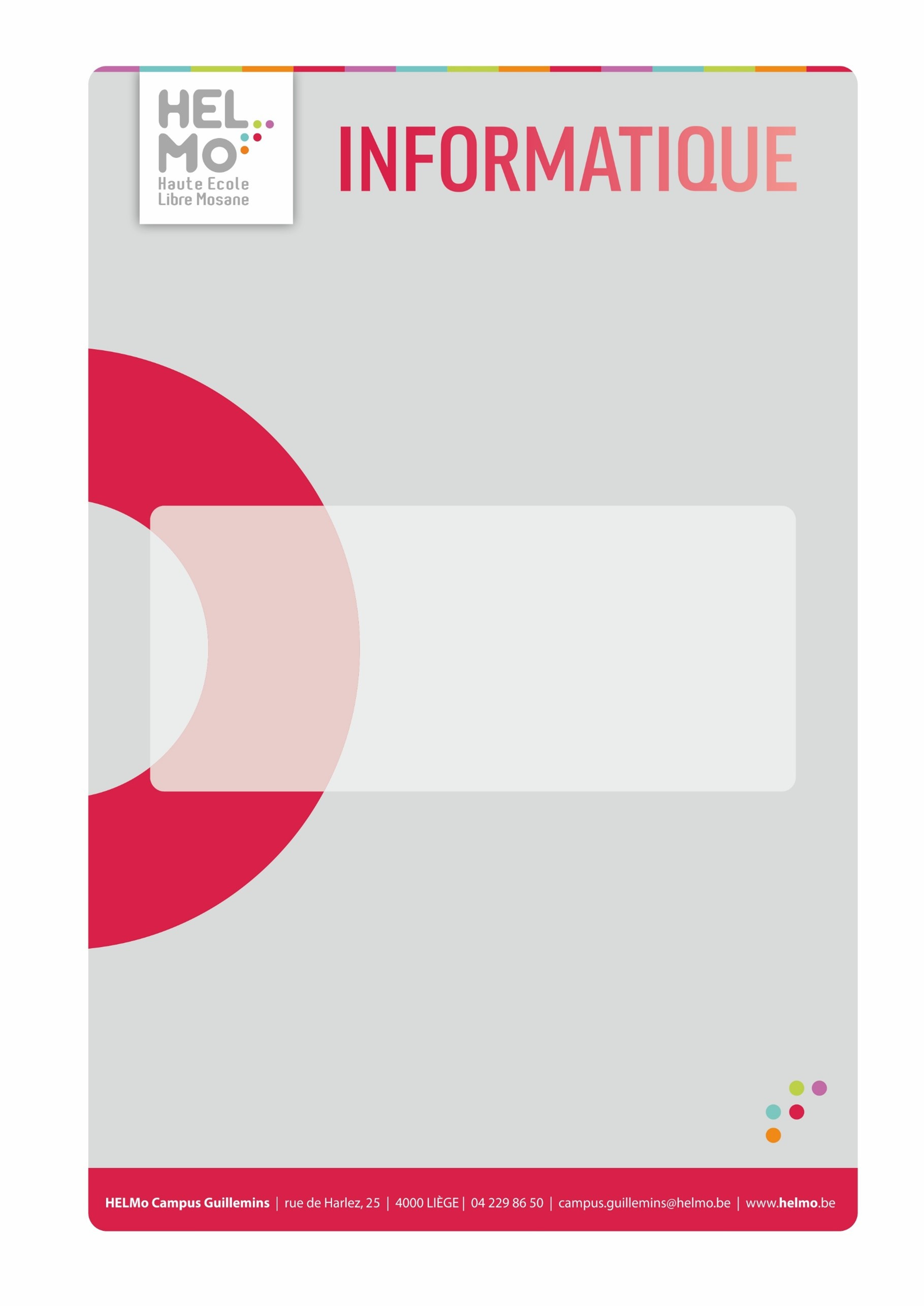
****

Mémoire de fin d’études

en vue de l'obtention du titre de

**Bachelier en Informatique   
orientation Développement d’applications**

Année académique 20 -20

**! Veuillez effacer les infos en fluo après rédaction**

**Titre centré**

**(titre exact qui paraîtra sur le diplôme)**

|  |  |
| --- | --- |
| TECHNORD Rue de la Lys, 21 7500 Tournai | Présenté par  **Zachary VANVLASSELAER** |

1. Présentation de l’entreprise

1.1 L’entreprise

Technord est une entreprise de service spécialisé dans les domaines de l’électricité, de l’automation, de l’IoT, de l’informatique industriel et de l’intelligence artificielle. Elle s’est donnée comme mission d’apporter des solutions fiables et efficaces sur le long terme à ses clients, venant principalement du monde de l’industrie.

L’une des valeurs fondamentales de Technord est son amour et sa confiance du client. En effet, l’entreprise met beaucoup d’importance sur sa relation avec celui-ci. D’autres lignes directrices sont sa vision, son agilité ainsi que son expertise, sans bien évidemment oublier de garder une touche d’humour et de fun.

Elle est composée de plusieurs centaines d’employés à travers différents pays d’Europe. Son siège social se situe à Tournai, en Belgique.

Son chiffre d’affaire dépasse les 80 millions d’euros, amassant plus d’1 millions d’euros de bénéfices par an.

1.2 L’équipe

Pour ma part, je vais évoluer sur le site de Seraing*.* C’est un lieu important car il est proche des industries avec lesquelles Technord interagit au quotidien.Le bâtiment est divisé en 2 étages, représentant les 2 équipes, les 2 spécialités de l’endroit : l’automation et l’informatique industrielle MoM[[1]](#footnote-1) / MES[[2]](#footnote-2). J’ai intégré cette dernière avec 2 autres stagiaires, formant une team de 11 personnes travaillant main dans la main.

L’équipe est dirigée par Kevin Putzeys, responsable projet, qui s’assure que la réalisation des applications se fait en adéquation avec les attentes des clients. Ce sont lui et Alessandro Missoul, un IPS[[3]](#footnote-3), qui vont servir de principal intermédiaire entre les clients et les développeurs, bien que certains interagissent directement avec les concernés en cas de besoin.

Les 2 équipes sont entièrement indépendantes et s’occupent d’objectifs et de clients différents. Malgré cela, ils leur arrivent de collaborer sur certains projets communs.

Les interactions avec les autres départements sont majoritairement limitées à l’équipe d’informatique industrielle principale, située à Tournai. Celle-ci comprend des experts dans des domaines spécifiques que le site de Liège ne possède pas. C’est donc vers eux que l’on se dirige en cas de problème plus important.

2. Présentation du TFE

2.1. Avant

Tout commence lorsqu’un client de Technord introduit une demande pour la réalisation d’une application aidant à l’ouverture de chantiers. En effet, celles-ci peuvent devenir facilement compliquées à mettre en place proprement au vu du grand nombre de variables à prendre en compte. Il faut faire attention à chaque membre du personnel participant à l’ouverture de chantier, aux habilitations de ceux-ci, aux emplacements sur lesquels elle se déroule, aux risques liés à ces derniers et ainsi de suite.

Pour répondre à ce besoin, l’équipe MES de chez Technord a développé une application web en ASP.Net avec l’aide du framework Telerik qui s’occupe de faciliter tout ce processus.

Durant les années qui vont suivre, d’autres clients vont faire des demandes similaires, ce qui va conduire Technord à, dès lors, proposer l’application déjà existante à tous ceux le voulant sous forme de « package », offrant en plus de la personnalisation et une initialisation des données. Ces clients vont aussi demander de nouvelles fonctionnalités, qui vont être petit à petit ajoutées au projet initial.

2.2. La problématique

L’application a été développée sur plusieurs années, par différentes personnes et avec des technologies déjà relativement anciennes pour l’époque. De plus, en essayant de répondre à certains besoins trop vite, elle fut réalisée sans forcément porter attention à la bonne réalisation des choses, la rendant pas toujours fiable et peu attirante visuellement. Par ailleurs, le processus de création s’étant fait de manière dispersée, le code de l’application est désordonné, rendant l’ajout de nouvelles fonctionnalités plus difficile que ce que ça ne devrait l’être. Pour toutes ces raisons, l’équipe de Technord a décidé de tout recommencer de zéro, avec de nouvelles technologies et une interface grandement améliorée.

2.3. Le projet

Ce qui nous a été demandé, à mon coéquipier et moi, est de migrer l’entièreté de l’ancienne solution monolithique vers une application composé de deux services comme l’est maintenant courant : un front-end et un back-end.

Le front-end permettra à l’utilisateur d’interagir avec les fonctionnalités de l’application et sera réalisé en Angular à l’aide du framework Kendo.

Le back-end servira d’intermédiaire entre l’interface visuelle et la base de données et s’occupera du traitement des requêtes. Il sera réalisé en .Net Framework avec le langage de programmation C#.

La base de données restera majoritairement inchangée pour permettre une installation simple de la nouvelle solution chez le client.

2.4. Les objectifs

D’abord, il est très important que la totalité des fonctionnalités de l’ancienne solution soit disponibles sur la nouvelle. Il est question fournir une version améliorée aux clients. L’application sera donc testée rigoureusement afin de ne pas créer une perte de productions chez ces derniers.

Ensuite, comme déjà mentionné, la nouvelle application sera un renouvellement visuel. La barre sera donc grandement relevée en ce qui concerne le design de l’interface.

Enfin, il devra être possible de facilement ajouter de nouvelles fonctionnalités au projet, que ça soit dans un future lointain ou non. Certains ajouts vont d’ailleurs être proposés comme tâches bonus aux stagiaires en fonction de l’avancement du stage.

3. Méthodologie

Technord utilise la méthode SCRUM afin de mener à bien ses projets. Ceux-ci sont divisés en plusieurs petites tâches réalisables par une seule personne qui composeront ce que l’on appelle le backlog. La réalisation d’un projet se fait durant des sprints, qui sont des unités de temps arbitraires choisies au début du projet. Dans notre cas, chaque sprint dure 2 semaines, divisant notre stage en 7 sprints. La première semaine n’est pas prise en compte, ayant majoritairement servi à se mettre en place, à analyser la solution existante et à se familiariser avec l’équipe.

C’est durant une réunion appelée sprint planning, faites à chaque début de sprint, que l’on attribue les tâches à chacun. Les 2 maîtres de stage proposent des tâches intéressantes à faire pour le prochain sprint et les 2 stagiaires donnent leur avis sur la faisabilité de celles-ci. Il est important d’attribuer à chaque tâche un certain nombre de story points, qui est une unité arbitraire représentant le temps estimé que prendra la réalisation de cette dernière.

A la fin d’un sprint, les membres de l’équipe organise une autre réunion, nommée sprint review, dans laquelle chacun pourra exprimer son ressenti par rapport au travail réalisé. Les points difficiles seront passés en revue ainsi que les raisons expliquant ces difficultés. Les story points seront aussi comparé à la véritable durée de chaque tâche. Tout ce processus permet d’améliorer le prochain sprint planning, afin d’estimer de manière plus précise le travail qui pourra être accompli.

Daily reviews

Jira

4. Analyse

4.1 Fonctionnalités

Au vu de la taille relativement conséquente qu’était cette application d’ouverture de chantier, Technord a décidé de remettre ce travail à 2 personnes. L’HELMo ne voulant pas que 2 étudiants aient exactement le même stage, les maîtres de stages se sont concertés afin de diviser les fonctionnalités développées par Noah et moi de la manière la plus équitable possible. Il a donc été décidé que mon collègue s’occuperait plutôt de la partie centrale de l’application, l’ouverture de chantier en elle-même, tandis que je serai taché de gérer tous les éléments gravitant autour de cette dernière.



Ces tâches peuvent être divisées en plusieurs sections : la gestion des facteurs, la gestion des sous-traitants et l’encodage et réalisation des questionnaires. Je vais expliquer chacune de ces sections en détails à l’aide de « User Stories » et de commentaires.

4.1.1 Gestion des facteurs

Comme mentionné plus tôt une ouverture de chantier dépend de beaucoup de facteurs différents qui changent en fonction de l’ouverture de chantier. Il est donc très important de laisser la possibilité aux utilisateurs de gérer ceux-ci. Chaque facteur a donc une page désignée permettant au minimum les opérations CRUD[[4]](#footnote-4), voire plus si nécessaire. Les « User Stories » se présentent de la manière suivante :

Pour les paramètres, les types de certifications, les types d’habilitations, les risques, les moyens de préventions, les emplacements, les permis, les rôles intervenants, les types d’évaluations, les catégories d’évaluations et les EPI :

* En tant qu’utilisateur, je veux pouvoir ajouter un facteur afin de pouvoir l’utiliser dans une ouverture de chantier.
* En tant qu’utilisateur, je veux pouvoir voir tous les facteurs à ma disposition afin de pouvoir apporter des changements à ceux-ci si nécessaire.
* En tant qu’utilisateur, je veux pouvoir modifier un facteur afin que ses changements se reflètent dans son utilisation dans une ouverture de chantier.
* En tant qu’utilisateur, je veux pouvoir supprimer un facteur afin de ne plus pouvoir l’utiliser.
* En tant qu’utilisateur, je veux pouvoir voir les facteurs supprimés afin de pouvoir potentiellement en restaurer un.
* En tant qu’utilisateur, je veux pouvoir restaurer un facteur afin de pouvoir l’utiliser de nouveau.

Il est important de noter que l’application dans son entièreté repose sur un système de droit. Pas tout le monde n’a le droit de faire ce qu’il veut. Chaque page et chaque action a un droit qui lui est associé et seules les personnes ayant ce droit peuvent accéder à cette page ou réaliser cette action. Dans la majorité des cas, le fait de pouvoir voir les éléments supprimés et donc restaurer un facteur est limité aux administrateurs, qui sont les seules avec les droits nécessaires. Cependant, l’application permet aussi de gérer les droits associés à chaque type d’utilisateur. Il est donc entièrement possible qu’un utilisateur puisse restaurer un facteur si le client le décide. C’est donc pour cette raison que j’ai généralisé la personne faisant l’action à « utilisateur » dans les « User Stories ».

Bien évidemment, la gestion des facteurs ne se limitent pas qu’à ces simples opérations. La majorité des pages proposant des fonctionnalités supplémentaires. Je vais donc vous détailler celles-ci ainsi que vous donner plus de contexte pour chaque facteur.

La gestion des paramètres n’est pas vraiment ce à quoi on s’attend lorsque l’on parle de paramètres d’application. C’est plutôt une liste d’option possible que l’on peut choisir pour une famille de paramètre donnée. Cette fonctionnalité permet donc de définir par exemple tous les types d’habilitation, ou toutes les familles de risques, etc.

Schéma de comparaison

Pour cette page, on peut ajouter comme « User Story » :

* En tant qu’utilisateur, je veux pouvoir changer la famille de paramètre actuelle afin que les opérations que je fais sur les paramètres s’appliquent à celle-ci.

La gestion des types d’habilitations et des types de certifications suivent le même principe, étant seulement différent dans la chose à laquelle elles s’appliquent. Ce sont des agrégations qui permettent de s’assurer qu’un membre du personnel (pour les habilitations) ou une société (pour les certifications) soit en mesure de répondre à un potentiel risque. Ces agrégations doivent potentiellement être validées par une personne compétente avant qu’elles prennent effet. Il est donc possible de définir qu’un type d’agrégation donné demande une validation et tout changement à cette valeur doit se refléter dans les agrégations héritant de ce type. Chaque agrégation possède aussi une de fin de validité, qui peut être définie à l’aide de son type. Voici donc les « User Stories » à ajouter :

* En tant qu’utilisateur, lorsque je décide qu’un type d’agrégation ne demande plus de validation, je veux que chaque agrégation héritant de ce type soit automatiquement validée.
* En tant qu’utilisateur, lorsque je change la date de fin de validé d’un type d’agrégation, je veux que chaque agrégation héritant de ce type voit sa date de fin de validité changer en accordance avec le type.

La gestion des risques et des moyens de prévention est la seule exception à « une page par facteur » étant donné qu’ils sont intrinsèquement liés. En effet, un risque possède automatiquement des moyens de préventions, vu qu’il ne serait pas cohérent de commencer une ouverture de chantier avec un risque que l’on ne peut pas éviter. Il est aussi possible de lier des habilitations, des certifications et des permis à un moyen de prévention. Ainsi, lorsqu’un risque est présent sur une ouverture de chantier, chaque moyen de prévention associé à ce risque devra avoir ses agrégations remplies par quelqu’un ou quelque chose. Je vais donner un exemple afin de mieux illustrer mes propos. Imaginons un scénario simple avec le risque unique suivant :



Si une ouverture de chantier présente le risque « Chute d’objets », il est alors impératif qu’au moins une des personne travaillant sur cette ouverture de chantier est l’habilitation « Levage / Elingage » et que la société faisant l’ouverture de chantier possède le permis « Balisage ».

Par ailleurs, chaque risque est lié à une famille de risque, permettant de mieux organiser ceux-ci. De plus, on peut associer des fichiers à un moyen de prévention. Les « User Stories » suivante doivent donc être prise en compte :

* En tant qu’utilisateur, je veux pouvoir changer la famille de risque actuelle afin de pouvoir gérer les risques venant de cette famille.
* En tant qu’utilisateur, je veux pouvoir associer des habilitations, des certifications ou des permis à un moyen de prévention afin que ceux-ci soient pris en compte dans une ouverture de chantier présentant le risque auquel le moyen de prévention est associé.
* En tant qu’utilisateur, je pouvoir faire les opérations CRUD présenté précédemment à des fichiers associés à un moyen de prévention.

Ajout de la gestion multi-site

1. Manufacturing Operation Management [↑](#footnote-ref-1)
2. Manufacturing Execution System [↑](#footnote-ref-2)
3. Ingénieur projet senior [↑](#footnote-ref-3)
4. Create Read Update Delete : Créer, lire, mettre à jour et supprimer [↑](#footnote-ref-4)