

ISFCE
Rue Joseph Buedts 14,
1040 ETTERBEEK

RAPPORT

Examen Spring boot



Professeur :
M. VAN OUDENHOVE

Année scolaire :
2019-2020

Nom et Prénom :
Chairi .A
El Hadouchi .Y

TABLE DES MATIERES

I) Introduction à Spring Boot et installations des différentes librairies (dépendances)_2	
II) Thème et consignes à respecter concernant la réalisation du projet :_6	
III) Méthodologie de travail et installation d'autres dépendances_____6	
IV) Présentation des différentes couches de l'application_____7	
V) Présentation des différentes vues (interfaces) créées et des fonctionnalités proposés par l'application :_____8	
VI) Conclusion_____13	



I) Introduction à Spring et installations des différentes librairies (dépendances)

Au Cours du Projet d'intégration et de Développement. Nous avons appris à gérer le langage Java et plus précisément Spring, qui est un Framework open source pour construire et définir l'infrastructure d'une application Java, dont il facilite grandement le développement ainsi que les tests. Dans le cadre du cours, nous avons utilisé ce Framework pour créer une application web permettant la gestion des stages, pour mener à bien la réalisation d'un projet en Spring différents Logiciels et plugin sont à installés au préalable.

- 1) Installer un IDE Environnement de développement (Nous utilisons l'IDE Spring tool suite 4)
- 2) Installer un JDK pour notre IDE, en l'occurrence le dernier. Nous avons installé le JDK 11 compatible pour Java 11 (Version 11)
- 3) Installer le projet Lombok, permettant essentiellement de définir pour le développeur des constructeurs, getters, setters dans nos classes modèles et ainsi synthétiser nos classes (gain de temps).
- 4) Installation du Plugin Gradle, qui est un moteur de production fonctionnant sur la plateforme Java. Il permet de construire des projets en Java, de rajouter des dépendances à notre projet et également de définir des tests unitaires.
- 5) Installations des SGBDR(s) (Système de Gestion de Base de données Relationnels). Nous avons installé le SGBDR H2.
- 6) Installation du plugin Thymeleaf qui est un moteur de template, écrit en Java pouvant générer ainsi du XML/XHTML/HTML5. Thymeleaf peut être utilisé dans un environnement web (utilisant l'API Servlet). Son but principal est d'être utilisé dans un environnement web pour la génération de vue pour les applications web, il se base sur le modèle MVC (Modèle, Vue, Controller).

II) Thème et consignes à respecter concernant la réalisation du projet

Tâches à développer obligatoirement

Une répartition des tâches sera établie en classe en fonction que vous travaillez individuellement ou en groupe

1. Réaliser toute la couche métier ainsi que les DAO associés
2. prévoir les interfaces pour les processus suivants :
 - (a) une vue pour créer et modifier un maître de stage
 - (b) une vue pour encoder/modifier un stage
 - (c) une vue pour évaluer les compétences d'un stage (par le maître de stage)
 - (d) une vue pour évaluer les degrés de maîtrise d'un stage (par le maître de stage)
3. Utiliser des tests unitaires pour vérifier le bon fonctionnement de l'application



III) Méthodologie de travail et installation d'autres librairies

A présent nous allons vous expliquer la méthodologie que nous avons adoptées afin de réaliser le Projet ainsi que notre conception notre compréhension de ce Dernier. Pour Commencer, la méthodologie adoptée durant toute la conception du projet repose sur une méthodologie « du monde professionnel », ainsi nous avons voulu garder à l'esprit que nous devons réaliser un projet tout en étant confronté aux aléas du monde professionnel et des discordances qu'il peut y'avoir. Pour ce faire ont à utiliser le dépôt Gitlab (gestionnaire permettant d'évoluer en équipe, et de conserver un suivit des fonctionnalité implémentées). Ainsi, toute la conception du projet repose sur un processus itératif et incrémental dans la mesure où nous avons dû répéter une série d'opérations afin d'aboutir à un résultat désiré (produire une version améliorée) et qu'on a dû produire davantage en produisant quelque chose de nouveau tout en modifiant l'ancienne.

IV) Présentation des différentes couches de l'application

Pour respecter une bonne organisation du projet, nous avons appris à réaliser un projet considérant une architecture MVC qui consiste à séparer un projet en différentes couche afin de mieux gérer les éventuels bugs qui peuvent survenir et ne pas adoptée « une programmation spaghetti ». Les différentes couches de l'architecture MVC sont :

La couche Modèle : Les classes « métier » qui sont nos Entités (deviendrons des tables dans la Base de données). C'est à travers cette couche que le Project Lombok prend tout son sens, car c'est là que nous devons définir des constructeurs, getters, setters.

La couche DAO (Data Acces Object) : Assurant l'interactions avec nos entités (tables), ainsi on aura autant de classe DAO qu'il n'y a d'entités. C'est également à travers cette couche que nous définirons l'ensemble des requêtes pour interroger nos entités. Fort heureusement, nous avons également utiliser La Java Persistance API (abrégiée en JPA), qui est une interface de programmation Java permettant aux développeurs d'organiser des données relationnelles. Ainsi certains requêtes, opérations de types CRUD (Create, Read, Update, Delete) essentiellement, sont déjà définit dans l'interface JPARepository, et qui sont prête à l'emploi.

La couche Service : permet entre autres de faire le pont entre la couche Controller et la couche DAO. Elle assure donc l'implémentations de chacune de nos méthodes définit préalablement dans notre couche DAO, ainsi c'est à travers la couche Contrôleurs qu'on pourra accéder aux méthodes implémentées dans la couche Service et donc les utilisés.

La couche Contrôleurs : permet de faire le pont (l'intermédiaire) entre les couches vues et service. Elle contient la « Logique » concernant les actions effectuées par un utilisateur. C'est à travers cette couche qu'on enverra à la « vue » les données (méthodes implémentées) issus de la couche service. L'interaction « indirect » entre la couche vue et la couche service passe par le contrôleur. C'est à l'aide du Model (pas la couche mais l'objet de type Map) qu'on ajoutera un attribut grâce à la méthode (addAttribute) qu'on définira une clef à laquelle on associera une valeur (exemple : liste des stages), ainsi et pourra accéder à cette clef dans la vue et donc afficher la vue.

La couche vue : contiendra la présentation des interfaces graphique des vus l'aide du Template Thymeleaf. C'est également à travers cette couche qu'on définira l'héritage des « vues », il ne sera donc pas nécessaire de re définir des vue « communes » à chaque fois. Ce principe d'héritage de vue est assuré par thymeleaf dans lequel il existe « une balise »



V) Présentation des différentes vues (interfaces) créées et des fonctionnalités proposées par l'application

Présentation de la vue du lors du démarrage de l'application

Au lancement de l'application, La vue principale représentant « le site de l'école » apparaît, dans lequel il y a un menu de navigation, des informations relatifs au cours et un pied de page. Au départ, le menu de navigation nous permet seulement de nous connectés, d'accéder à la page principal (dans lequel se trouvent les informations relatives expliquez précédemment) ou de changer la locale (langue de l'application). Le pied de page quant à lui affiche des informations ou permet d'accéder aux réseaux sociaux relatives à l'école. Lorsqu'on souhaite se connecter à l'application, l'utilisateur est redirigé vers une nouvelle vue dans lequel il doit rentrer son nom d'utilisateur et son password, n'importe qui ne peut pas se connecter à l'application ! rappelez-vous c'est un site dédié à des membres ayant un lien direct (ou pas) avec l'école. Par conséquent des comptes doivent être créés au préalable par un « Super Utilisateur » qu'on appelle admin, c'est donc ce dernier qui peut se connecter en premier à l'application afin de créer les comptes associés aux membres. Vous trouverez ci-dessous, une explication détaillée des fonctionnalités proposées et des droits d'accès selon le rôle de l'utilisateur connectés.

Rôle Administrateur : Lorsque l'administrateur se connecte, plusieurs onglets apparaissent dont l'onglet « Gestion des comptes ». Comme son nom l'indique cet onglet permet notamment de créer les comptes (nécessaires aux membres pour se connecter), ou encore d'accéder à la liste des membres selon leurs rôles (Ex : Liste des maîtres de stages).

L'administrateur est donc en mesure de créer des comptes selon les rôles des utilisateurs suivants :

- Les Professeurs de l'établissement.
- Les Etudiants de l'établissement, comme c'est souvent le cas dans les écoles des comptes créés préalablement et sont attribués aux étudiants.
- Les Maître de Stage, responsable des étudiants effectuant un stage dans leurs entreprises.

Lorsque l'admin crée un compte il devra remplir un formulaire dans lequel, des champs sont à définir (informations relatives à un utilisateur), Les informations « primaire » pour chaque utilisateur sont : un username, un password, un nom, un prénom et un email. Lorsque l'admin souhaite créer un compte qui a pour rôle (étudiant ou maître de stage), il devra en plus définir un champ (numéro de téléphone), ce champ là n'existe pas pour le professeur. Bien entendu Tous les champs se doivent d'être mentionnés

par le responsable de la création d'un compte ! Sans quoi, une erreur se produira si jamais l'un des champs est manquant ou que l'un des champs ne respect pas une certaine « norme » d'écritures (tel que L'email par exemple).

Comment l'utilisateur sera tenu au courant de la création de son compte ?

Car oui c'est là que se trouve tout l'enjeu : Prévenir le principal concerné pour lequel son compte est créé. Pour assurer ce besoin, au moment de la création d'un compte, nous avons rajouté une fonctionnalité qui « avertira » l'utilisateur concerné que son compte est créé. Oui mais Comment ?! Tout simplement par envoi de mail ! Rappelez-vous, lors de la création du compte de l'utilisateur il faut également spécifier son mail. De cette façon, un mail reprenant les informations nécessaires à la connexion (username et password) sera automatiquement envoyé à l'utilisateur.



Ainsi il pourra savoir aisément lorsqu'il pourra accéder à l'application. Par la même occasion, lorsque le compte d'un utilisateur créé son mot de passe défini sera crypté (dans la base de données), ce qui assure une sécurité supplémentaire, dans le cas où une personne mal intentionnée s'amuserait à essayer de pirater un compte. L'administrateur peut également consulter plusieurs

Liste tel que :

- La liste des maitres de stages
- La liste des cours proposé par l'établissement
- La liste des stages en cours
- La liste des étudiants

Rôle étudiant :

Lorsqu'un étudiant se connecte, il pourra avoir accès à partir du menu de navigation à un lien qui le redirigera vers la liste de ces stages clôturée et il pourra voir le résultat final. Il aura accès aussi à l'onglet autre qui lui permettra de d'avoir accès au lien profil qui le redigera vers le menu pour modifier son mot de passe s'il le souhaite.

Rôle Professeur : Lorsqu'un professeur se connecte, il pourra bénéficier principalement des mêmes fonctionnalités que l'admin. Il pourra donc avoir accès à l'onglet « Gestion des comptes » pour lequel il pourra créer des comptes étudiants et maitres de stages mais pas admin (soyons logique). Il pourra en plus avoir accès à l'onglet « Stage » dans lequel il sera en mesure de créer un stage ou de consulter la liste des stages en cours. Pour assurer l'établissement d'un Stage (la création), le professeur devra dans un premier temps définir des informations élémentaires (les informations supplémentaire seront par la suite défini pas le maitre de stage) , il devra donc sélectionner la liste des modules représentant le stage, dans le Bachelier Informatique il en existe 2 : IAPF et ISIP où chacun des modules est sous la responsabilité d'un professeur en particulier, sélectionner l'étudiant et pour finir sélectionner le maitre responsable du stage de l'étudiant.

Oui, mais plusieurs problématiques se pose tel que :

Comment sont gérer les créations des stages sachant que les modules sont sous la responsabilité de différents professeurs ?

Il est vrai qu'intuitivement on aurait tendance à dire que le professeur connecter sera en mesure de créer un stage pour le module dont il est responsable. Mais dans le cadre de cette application nous avons laissé « un libre arbitre » aux professeurs en ce qui concerne la création d'un stage, cela signifie qu'un professeur peut très bien crée un stage pour un module qui ne le concerne pas.

Cependant cela fait-il de lui le responsable du stage pour autant ?

Absolument pas, la logique défendue veut simplement qu'après concertation des professeurs, ils puissent s'entraider en ce qui concerne les créations des stages (qui sont pour souvent des taches assez « fastidieuses »), cela permettra donc d'alléger grandement la masse de travail à faire. De la même façon, cette « entraide » entre professeurs engendra un gain de temps non négligeable.

Comment connaitre les étudiants qui peuvent passer le stage pour l'un des modules (IAPF/ISIP) ?

En effet pour pouvoir passer un stage, l'étudiant doit être inscrit pour l'un des modules du stage. Il est donc évident qu'on ne doit pas récupérer l'ensemble des étudiants du Bachelier mais seulement ceux qui sont inscrit aux modules pour le stage.



D'accord mais en fonction du module sélectionner, Comment savoir si l'étudiant est inscrit pour ce module-là du stage et pas pour l'autre ?

Encore une fois une problématique assez pertinente auquel on doit faire face, il est vrai que récupérer l'ensemble des étudiants pour l'un des 2 modules ne suffit pas. Vous ne voyez toujours pas le problème qui risque d'arriver ? Prenons un exemple concret.

Tableau reprenant les étudiants inscrits au module IAPF ou ISIP :

Modules	Etudiants Inscrits			
IAPF	ET1	ET2	ET3	ET5
ISIP	ET1	ET2	ET4	

Comme on peut s'en apercevoir nous sommes confrontés à deux cas de figures :

Le 1^{er} cas : un même étudiant peut être inscrits aux deux modules (Ex : ET1,ET2)

Le 2^{ème} cas : un étudiant peut être inscrits pour l'un des deux modules (Ex : ET3, ET4, ET5)

A Présent, revenons à notre problématique de base. Il est clair que pour les étudiants inscrits aux 2 modules (1^{er} cas) cela ne pose aucun souci.

Quand est-il pour celui qui est inscrit pour l'un des deux (2ème cas) ?

C'est là qu'intervient la deuxième problématique énoncé précédemment, Supposons que nous sélectionnons le module ISIP, seuls les étudiants inscrits pour ce module sont censé apparaître (ET1, ET2, ET4), et pour le module IAPF même principe seulement les étudiants (ET1,ET2,ET3,ET5) sont censé apparaître. En d'autres termes, en plus d'afficher l'ensemble des étudiants inscrits aux modules il faudra les récupérer « dynamiquement » selon ceux inscrits pour le module sélectionné. Pour ce faire nous avons créé une méthode en Ajax (asynchronous JavaScript and XML), dans laquelle nous exécutons la requête qui récupère les étudiants en fonction du module sélectionné. Enfin pour la création du stage, il ne faudra pas oublier d'également sélectionner le maître du stage, dans le cas où aucun n'est encore existant, un lien apparaîtra permettant ainsi de rediriger le professeur vers le formulaire permettant de créer le maître de stage. Il faut également savoir qu'un stage passe par différent « état » marquant ainsi son évolution, par conséquent lors de sa création (du stage), ce dernier passera par un premier état qui est « INIT » signifiant l'état « d'initialisation ». Le professeur responsable du module associé au stage pourra à tout moment arrêter (supprimer) un stage qui se déroulera entre l'état « init-started ». Supposons qu'après concertation avec un étudiant et pour x raisons ce dernier ne pourra pas assumer le stage.

D'accord jusque-là, mais une fois l'établissement du stage créé, comment les intervenants (l'étudiant et le maître de stage) sont tenus informé ?

Effectivement, il faut pouvoir avertir les intervenants de toutes activités les concernant. C'est là qu'intervient le système d'envoi de Notifications, vous savez ces petites bulles qui spam vos réseaux sociaux et qui pour la plupart du temps sont rouge avec un nombre qui vous indique combien de notifications vous avez eu. Oui, C'est exactement à l'aide de ce système que les intervenant seront informé de l'évolution du stage de sa création jusqu'à son évaluation. Pour assurer ce système-là, il nous a également fallu créer une méthode Ajax dans laquelle nous exécutons la requête affichant la liste des notifications selon l'utilisateur connecté (il est évident que les autres utilisateurs n'ont aucun intérêt d'être averti d'un stage qui ne les concerne pas, voyons chacun balayer devant sa porte), par la même occasion l'utilisateur verra également le nombre de notifications reçu. Le professeur aura aussi la possibilité de consulter la liste de tous les stages en cours et de voir ainsi l'évolution de chacun.



Mais c'est en contradiction avec ce que tu as expliqué précédemment.

Comment le professeur peut voir la liste de tous les stages, sachant qu'il se peut que certains d'entre eux ne le concerne pas ?

Tu as le sens du détail mais consulter ne signifie pas influencer (modifier) ! Le professeur authentifié, pourra certes consulter La liste des stages en cours, mais il ne pourra avoir une influence (modifier) que les stages pour lesquels il est responsable, là est toute la nuance ! Rappelez-vous la responsabilité existante entre le stage et le professeur est le module associé (IAPF ou ISIP). Un professeur sera donc en mesure de Signer les conventions de stage pour le module dont il est responsable (Oui mais que lorsque les informations supplémentaires de ce dernier seront remplies par le maître de stage). Ainsi une fois les conventions de stage signé, son état évoluera à « SIGNED ». Comme vous vous en doutez bien-entendu l'étudiant et le maître de stage pour lequel le stage est signé sauront tenu au courant grâce à l'envoi d'une notification. Pour ce qui est du commencement du stage (STARTED) et de la fin du stage (ENDED) ces états seront détaillés d'avantage dans le rôle concernant le maître du stage. A Présent, projetons-nous plus loin dans les états, lorsque le stage d'un étudiant prend fin (ENDED) ce dernier doit être évalué par les intervenants du stage qui sont : Le professeur responsable du module du stage et le maître de stage responsable de l'étudiant durant son passage dans l'entreprise.

Les intervenants devront évaluer l'étudiant de 2 façons, la première consiste à évaluer les compétences et la seconde consiste à évaluer les degrés de maîtrises vous trouverez ci-dessous une explication détaillée de ces évaluations :

- **Evaluation compétence** : Comme son nom l'indique, Un formulaire permettra aux parties concernées (professeur et maître de stage) d'évaluer des compétences.

Mais encore ?

- Mettons fin au suspense, chaque module (cours) possèdent une liste de compétences qui lui est propre, par conséquent cette évaluation consiste juste à déterminer si l'étudiant a acquis (ou pas) les compétences repris en fonction du module. Ainsi lorsqu'il faudra évaluer le stage, il faudra non seulement récupérer l'id (numéro) du stage mais en plus de sa la liste des compétences associé au module (ici IAPF ou ISIP)

Ça fait beaucoup de « il faudra là », Tu veux nous dire que 2 modules (cours) non pas forcément les mêmes compétences ?

C'est exactement ça, et ce n'est là que le début de nouveaux problèmes (une fois n'est pas coutume). Car en plus de récupérer la liste des compétences en fonction du module, il faut savoir que certaines compétences sont à évaluer pas le professeur et certaines autres compétences sont à évaluer par le maître de stage. Donc c'est la combinaison des 2 parties de compétences (Professeur et maître Stage) qui forme complètement l'évaluation de compétence d'un stage. Autant vous dire que ça fait beaucoup de chose à traiter.



Donc si je suis bien tes dires, il faudra non seulement récupérer le numéro du stage, récupérer la liste des compétences en fonction du module du stage et enfin selon le rôle de l'utilisateur connecté (professeur ou maitre de stage) récupérer les compétences qu'il doit évaluer ?

Ça en a mis du temps mais on y est arrivé, voilà c'est à l'aide d'une requête qu'il a fallu gérer ce cas-là. Pour ce faire on a défini dans notre formulaire (vue) des champs qu'on va cacher dans lesquels on va récupérer la valeur du numéro du stage, ainsi que le numéro de chaque compétence qui sera évaluer et un booléen (vrai ou faux), nous dirons quels sont les compétences qui ont été acquis (ou pas). En plus de ces champs cachés nous avons défini 2 autres champs cachés. Le premier récupérant le numéro de la première compétence et le second récupérant le numéro de la dernière compétence.

D'accord tu m'as perdu, autant je comprenais le fait de définir un champ pour récupérer le numéro du stage, le numéro de la compétence à évaluer ainsi que le champ booléen permettant de savoir ce qui a été acquis ou pas, autant j'ai du mal à comprendre l'intérêt des 2 champs supplémentaire ajouté.

D'accord revenons sur l'une des problématiques évoqués précédemment, rappelez-vous le maitre de stage et le professeur ne doivent pas évaluer les mêmes compétences (ce qui est logique) mais c'est l'ensemble des compétences évalués par les 2 parties qui définit l'évaluation des compétences pour le stage. Autrement dit comme les 2 parties n'évalue pas les mêmes compétences, forcément les numéros de ces derniers seront différents il faut donc pouvoir récupérer « dynamiquement » les numéros des compétences selon le rôle de l'utilisateur connecter. Ne me regarder pas avec ces grands yeux, Un exemple concret ferai le plus grand bien

Tableau reprenant les compétences à évaluer pour IAPF (à titre d'exemple)

Module Concerné IAPF							
Compétence	C1	C2	C3	C4	C5	C6	C7
Eval Prof	prof	prof	prof	prof	Non prof	Non prof	Non prof

Comme on peut s'en apercevoir, les données du tableau reprennent des compétences (à titre d'exemple) pour le module IAPF, chacune des compétences spécifie si elle doit être évaluée par la partie professeur ou la partie maitre de stage. Donc les numéros des compétences allant de (1 à 4) sont évaluer par la partie professeur et les compétences allant de (5 à 7) sont évaluer pas la partie maitre de Stage.

D'accord mais pourquoi ne pas parcourir directement les compétences de 1 à 4 si le rôle de l'utilisateur connecter est professeur et inversement de 5 à 7 si c'est un maitre de stage ?

Ç'est envisageable Oui mais seulement si le module est IAPF et que les compétences ne changent pas entre temps ! Tu imagines si tu devais évaluer un stage ayant pour module ISIP ? Rappel toi tous les modules n'ont pas les mêmes compétences. D'où l'intérêt de rechercher dynamiquement les numéros de compétence. Ainsi selon le rôle de l'utilisateur authentifié, le premier champ récupéra le numéro de la première compétence trouver et le deuxième récupéra le numéro de la dernière compétence, mais de façon dynamique sans forcément avoir besoin de savoir quel le premier et le dernier numéro !



Remarque c'est assez pratique dans le cas où plusieurs modules n'ont pas les mêmes compétences et surtout dans le cas où on ne connaît pas au préalable le nombre de compétence pour un module. C'est donc en procédant ainsi que nous ajouterons les évaluations compétences pour un stage.

D'accord mais y'a-t-il un ordre en ce qui concerne les évaluations des compétences ?

Parfaite Transition, J'allais y venir justement dans le cadre du projet il est jugé préférable que ce soit le maître de stage qui évalue en premier les compétences, autrement dit le professeur ne pourra pas évaluer les compétences d'un stage tant que celle du maître de stage associé ne sont pas faites. De ce fait lorsque le stage prend fin « ENDED » seul le maître de stage sera en mesure d'évaluer les compétences, lorsque ce sera fait, l'état du stage passera à « EVALUATED_MS » signifiant que le maître de stage a déjà évalué les compétences du stage et comme vous devez sûrement vous en doutez cela signifie par la même occasion que le professeur peut à son tour évaluer les compétences du stage associé, lorsque ce sera fait l'état du stage passera à « EVALUATED_PROF ».

Il faut savoir que lorsque l'ensemble des compétences sont évaluées (par le professeur et le maître de stage), 2 cas de figure se présentent :

- 1) **L'étudiant a acquis toutes les compétences cela implique qu'il aura au minimum 50 % .**
- 2) **L'étudiant a raté au moins une compétence, et là c'est l'échec assuré du stage. Impliquant que ce dernier passera à l'état « CLOSED » signifiant que le stage est clôturé (évaluer complètement).**

Le cas n°2 étant réglé (échec du stage) que se passe après le cas n°1 ?

Ça a été sans doute l'une des plus grandes problématiques du projet tant les points de discordances furent nombreux (sans rires). Commençons dans un premier temps par détailler simplement les étapes suivant les évaluations des compétences. Lorsque les compétences d'un stage sont évaluées complètement et pour lesquels la totalité est réussie, cela donne place aux fameuses évaluations qui sont les degrés de maîtrise, Rappelez-vous c'est la deuxième façon d'évaluer un stage.

- **Evaluation de degré de maîtrise :** Pour faire court et simple, les évaluations de degré de maîtrises sont assez similaires aux évaluations compétences (à peu de chose près), la grande distinction réside dans le fait que cette fois le module du cours aura en plus d'une liste de compétence, une liste de degré de maîtrise qui regroupe un ensemble de critères pour lesquels une note sera attribuée à l'étudiant.

Donc encore une fois, les liste de degrés de maîtrises diffère selon le module et par la même occasion selon le rôle de l'utilisateur connecter (professeur ou maître de stage), il ne pourra évaluer que certains degrés de maîtrises ?

C'est ça, « ouf » quel soulagement j'ai eu pour un moment de devoir expliquer tout le processus qu'il a fallu expliquer pour les évaluations de compétences d'autant plus que c'est pareil (rire), si ce n'est qu'on attribuera cette fois une note pour chacune des degrés de maîtrises.



Jusque-là je te suis mais un détail semble m'échapper, tu as dit tout à l'heure que lorsque toutes les compétences évaluées sont acquises, l'étudiant ayant effectué le stage aura au minimum 50%, mais du coup qu'est-ce qui concerne les 50 % restant ?

Heureusement que tes questions nous mettent dans la bonne voie, les 50% restant concerne justement l'évaluation des degrés de maîtrises, autrement dit la somme des degrés de maîtrise devant être évalué par le professeur et le maître de stage forme la note de 50 %. Par ailleurs des exceptions surviendront également si prise de sympathie pour un étudiant l'une des 2 parties tente de lui attribuer à une note supérieure à la note maximale pour un degré de maîtrise (Exemple 12/10, 8/5), soyons raisonnable quand même.

Je crois voir plus claire à présent donc la somme des 50% obtenus pour la réussite des compétences avec le résultat (sur 50) obtenus pour les évaluations de degrés de maîtrises constitue la note globale pour le stage ?

On avance lentement mais sûrement, C'est effectivement ainsi que les stages sont évalués. Cependant les points de discordances dont nous faisons allusions plus haut n'ont toujours pas été abordés. En effet lorsqu'un stage se termine la partie devant évaluer les compétences en premier est le maître de Stage.

Quand est-il des degrés de maîtrise ? A Suivre ton raisonnement ça serait également le maître de stage qui serait amené à évaluer les degrés de maîtrises en premier Non ?

Eh bien c'est là que les choses se corsent. Si on analyse de plus près, il est clair que certaines actions menées par les parties (professeurs ou maître de stage), dépendent l'une de l'autre.

Un Bref récapitulatif :

1. Pour que le maître de stage puisse remplir les informations supplémentaires pour un stage, ce dernier doit être créé « INIT » au préalable par le professeur responsable.
2. Pour qu'un professeur puisse signer « SIGNED » un stage, les informations supplémentaires se doivent d'être mentionnées par le maître de stage.
3. Pour qu'un professeur puisse évaluer (sa partie) des compétences pour un stage, les (autres) compétences de ce dernier doivent au préalable d'être évaluées par le maître de stage « EVALUATED_MS »

C'est donc là que toute notre réflexion se repose, Comment faire en sorte qu'il n'y ait pas à chaque fois cette dépendance en ce qui concerne les actions menées entre les parties (Professeurs et Maître de Stage). Notre conception de l'application était basée sur des fonctionnalités ergonomiques, et simple d'utilisations, l'objectif étant de ne pas inciter, ni précipiter l'une des 2 parties à vite effectuer une action sans quoi l'une restera « bloquée ». De ce fait pour les évaluations des degrés de maîtrises, nous avons jugé préférables que les parties ne dépendent pas l'une de l'autre, autrement dit un maître de stage peut très bien évaluer sa partie de degré de maîtrise sans pour autant attendre que le professeur le fasse en premier et vice-versa.



Et encore une fois plusieurs cas de figure sont à prévoir :

- Le maître de stage se connecte, évalue les compétences et les degrés de maîtrise d'un stage en une fois puis se déconnecte.
- Le maître de stage (étant occupé par exemple) se connecte, décide d'évaluer seulement les compétences, se déconnecte ensuite, entre temps le professeur se connecte il évalue ensuite sa partie des compétences et se déconnecte et pour finir le maître de stage se reconnecte et décide d'évaluer les degrés de maîtrise.
- Le maître de stage (étant occupé par exemple) se connecte, décide d'évaluer seulement les compétences, se déconnecte ensuite, entre temps le professeur se connecte il évalue ensuite en une fois sa partie des
- compétences et des degrés de maîtrise pour le stage et se déconnecte et pour finir le maître de stage se reconnecte et décide d'évaluer les degrés de maîtrises.

Connecte, reconnecte, déconnecte ça va finir par nous donner le tournis tout ça, Mais du coup,

Comment vous avez procédé pour casser « cette dépendance » qui existe entre les parties (professeur et maître de stage) ?

C'est là tout l'enjeu car lorsque le maître de stage évalue seulement les compétences d'un stage, ce dernier passe à l'état « EVALUATED_MS » et lorsque le professeur évalue les compétences d'un stage, ce dernier passe à l'état « EVALUATED_PROF », par conséquent n'ayant pas d'autres état après si ce n'est « CLOSED » signifiant que le stage est clôturé, il est compliqué de savoir lorsque le professeur ou le maître de stages évaluent les degrés de maîtrises. Ainsi il faut être vigilant à ce qu'un stage ne soit pas clôturé « CLOSED » avant que les 2 évaluations soit faites par les 2 parties (professeur et maître de stage). Encore une fois c'est là que réside tout la problématique, Supposons que le maître de stage évalue en premier les degré de maîtrise, Il faudra s'assurer que l'état du stage ne passe pas à « CLOSED », la raison est simple : Le professeur n'a pas encore évaluer sa partie de degré de maîtrise ! et inversement si c'est le professeur qui évalue en premier les degré de maîtrise il faudra s'assurer que le stage ne passe pas à « CLOSED » sachant que le maître de stage n'a pas encore évaluer sa partie des degrés de maîtrises.

Pour pallier ce problème, il nous a fallu crée une requête où on récupérera l'ensemble de stage pour lesquels toutes les compétences sont acquises, ainsi que les stages pour lesquels l'ensemble des degrés de maîtrises non pas été évalués complètement.

Oui, Oui c'est bien ingénieux tout ça mais encore une fois un détail illogique me turlupine, Tout à l'heure tu as dit que si l'une des compétences n'a pas été acquises par l'étudiant, l'état du stage passera à « CLOSED » clôturer, signifiant par ailleurs que le professeur ou le maître de stage ne sera pas en mesure d'évaluer les degrés de maîtrise, ce qui marque une incohérence, prenons les cas de figures suivant (Tu vas voir, moi aussi je sais en faire) :

Le maître de stage se connecte il évalue toutes les compétences, pour lesquels l'ensemble est réussi ensuite il évalue les degrés de maîtrise et il se déconnecte, là le professeur se connecte il évalue ses compétences mais il s'avère que l'une n'a pas été acquise → problème ! le maître de stage à évaluer les degrés de maîtrise alors que l'une des compétences pour le stage n'a pas été acquise.

Je te l'accorde tu m'as bien eu sur ce coup à force tu vas finir par trouver des incohérences partout, Mais parfois dans le cadre de la création d'une application il faut savoir dissocier « la logique absolu » de la réalité de la vie.



C'est ce sur quoi repose toute notre conception de l'application, tachant de respecter une certaine logique mais en essayant d'y intégrer une part de ce qui est susceptible de se faire dans la vie réelle, le but étant aussi de ne pas « déboussolé » l'utilisateur afin qu'il puisse effectuer certaines actions à travers l'application de tel sorte qu'il serait amené à les faire dans la vie réel. Et Dans ce cadre, lorsqu'un étudiant finit son stage il est très courant que le maître de stage évalue en une fois les compétences et les degrés de maîtrise, ainsi l'étudiant sait à l'avance les degrés de maîtrise qu'il a eu pour la partie maître de stage (bien qu'on ne sait pas à l'avance si le professeur validera toutes les compétences de l'étudiant). Et Personnellement je ne trouve pas ça plus mal, ça évitera de mettre l'étudiant dans un « état » de stress dû à la longue attente des résultats, ainsi on abrègera rapidement l'ascenseur émotionnels par lequel il est susceptible de passer.

Bon après ce long périple sur « les évaluations » (je vous rassure c'était sans doute la partie la plus compliqué à gérer), passons sur une fonctionnalité plus « soft ». Lorsque les stages seront évalués (complètement ou pas), le professeur sera en mesure de voir la liste « résultats stage » qui se trouve dans l'onglet « Stage » (évidemment). Ainsi il pourra voir précisément les stages évalués ou en cours d'évaluations. En cliquant sur un bouton, pour un stage voulu, une fenêtre « Modal » apparaîtra (vous savez ce sont ces fenêtres « pop-up » qui apparaissent sans-cesse lorsque vous cliquez sur un lien à risque par exemple), ainsi le professeur pourra voir précisément les compétences et/ou les degrés de maîtrises (si certaines sont évalués) pour un étudiant en particulier. Pour traiter ce cas nous avons rajouté une classe supplémentaire qui se nomme « ClassComposite » dans laquelle nous avons 2 listes la première étant la liste des évaluations compétences et la seconde étant la liste des degrés de maîtrises. Ensuite nous exécutons une requête qui récupérera dans un premier temps la liste des évaluations compétences pour le stage et dans un second temps la liste des évaluations de degré de maîtrise. Evidemment la deuxième liste (évaluation degré de maîtrises) ne s'exécutera que si la liste des évaluations compétence ne possède pas de compétence non acquise. Et enfin nous renvoyant notre objet ClassComposite contenant nos 2 listes. Ensuite ces données seront traitées dans notre fichier JS, de façon à ce que ça soit plus « attractif » et lisible pour l'utilisateur. Lorsque la fenêtre Modal apparaîtra, On aura donc un compte rendu complet des évaluations avec le numéro du stage, le module du stage, l'étudiant et le résultat obtenus.

Pour faciliter « l'interprétations » des évaluations, des codes couleur apparaîtront ayant une signification particulière :

Pour la liste des évaluations compétences : un badge vert contenant le numéro de la compétence, indiquera que cette dernière a été validée et un badge rouge indiquera à fortiori le contraire. Au survole de chaque badge on y affichera son texte relatif à la compétence.

Pour la liste des évaluations de degrés de maîtrises : Même principe, mais cette fois-ci on y retrouvera dans chaque badge, la note obtenus par l'étudiant pour une dégrée de maîtrise. L'idée est la suivante, si pour une degré de maîtrise la note obtenu est supérieur à la moitié de la valeur de la dégrée de maîtrise, le badge sera en vert (Ex : 4/5), Si par contre la note obtenu est égal à la moitié de la valeur de la dégrée de maîtrise le badge sera en orange (Ex : 2.5/5), Sinon cela signifie que la dégrée de maîtrise est raté et dans ce cas le badge sera en rouge (Ex : 2/5). Evidement à chaque survole du degré de maîtrise apparaîtra le texte du degré de maîtrise associé.

Rôle Maître de stage : Lorsqu'un maître de stage se connecte, il pourra avoir accès à l'onglet « Gestion des comptes » pour lequel il pourra ajouter les information manquante que le professeur ou l'admin n'a pas pu communiquer car il n'a pas les informations suivante (Nom Entreprise,rue,code postale,ville) .Il pourra en plus avoir accès à l'onglet « Stage » ou il pourra consulter la liste des stages qui le concerne.



Que peux faire le maitre de stage en consultant cette liste qui se trouve dans l'onglet stage ?

Tout d'abord quand le professeur ou l'admin crée le stage il s'ajoute à liste de stage à l'état INIT (comme expliqué précédemment dans la partie professeur) à ce moment-là, le maitre de stage aura un lien sur le stage qu'il le redirigera vers un formulaire qui vas lui permettre de mettre à jour le reste des données du stage (date de début, date de fin, nombre heure, intitulé du stage) une fois les données rentre il sera redirigé vers la liste des stages où l'état passera dan une état transitoire « en attente signer »,ou il devra attendre que le professeur signe le stage pour qu'il puissent passé à l'état « Signed ».

Donc à ce moment-là, le maitre de stage devra se reconnecter pour passer le stage de l'état signed vers started ?

NON ! Pas du tout, imaginé la plaie que ça doit faire pour un maitre de stage d'aller vérifier/se rappeler pour chacun de ses stagiaire la date du début du stage (surtout s'il a plusieurs stagiaire).

Comme je m'en doute bien, vous n'avez pas refiler cette tache au professeur ? (Vue la quantité d'élève que possède un professeur)

Bien vue Sherlock Holmes ! Non, ce n'est pas non plus le professeur qui gère ce changement d'état pour les mêmes raisons.

Alors, qui peux bien gère ce changement d'état (Signed → Started) ?

Comme vous pouvez le constatée chaque acteur est préoccupé par autre chose est n'a pas que sa à faire de sa vie à vérifier si un stagiaire/élève débute bel est bien son stage au jour le jour. Donc nous avons fait en sorte que cette problématique soit gérée automatique. Pour cela nous avons juste vérifier la date de début stage (inséré par le maitre de stage) et égale à la date actuelle et que l'état se trouve bien à Signed.

A super ! Mais imaginé qu'un maitre de stage rentre une date de début antérieur celle actuelle ?

Pour ce cas-là nous avons simplement empêché lors de l'insertion qu'il ne rentre pas une date inferieur a la date actuelle.

Et je suppose que vous avez dû faire pareil pour passer de l'état Started vers Ended ?

Oui, Tout à fait, mise par le fait que lors de l'insertion des données de la date de fin on a fait en sorte que

la date de fin ne soit pas antérieure ou égale à la date actuelle.

Au point de vue, je suppose que la suite de l'encodage pour clôturer un stage se passe de la même manière que pour le professeur ?

Oui, comme dit précédemment le processus est le même donc le maitre devra commencer à évaluer les compétences en premier pour permettre au professeur d'encodé le reste des points.

Pourquoi avoir créé deux listes pour les stages ?

Le professeur/maitre de stage possède dans la menu navigation deux onglet :

1 : Stage : qui contient un lien qui le redirige vers la liste des stages à tout état confondu (init→closed).



Liste des Stages



code	Module	Date	Nombre d'Heures	Etudiant	Maitre De Stage	Etat Du Stage	Actions
1	IAPF-1-A	2019-10-12 au 2020-03-04	350	et1	maitretf	ENDED	
2	IAPF-1-A	2019-09-10 au 2020-03-04	300	et2	maitretf	ENDED	
3	IAPF-1-A	2019-12-15 au 2020-03-04	250	et3	maitretf	ENDED	
4	ISIP-1-A	2019-12-15 au 2020-03-04	250	et3	maitretf	ENDED	
6	IAPF-1-A	2020-05-16 au 2020-05-17	200	et1	maitretf	Attente SIGNED	
7	IAPF-1-A	a completer	0	et3	maitretf	INIT	

ISFCE
Home
Gestion Comptes
Stage
Evaluations Stages

Autre
Français

Liste des St
Liste

2 : Evaluation stage : qui contient un lien qui le redirige vers la liste de stage qui sont à l'état ended à started.

Liste des stages



code	Module	Date	Nombre d'Heures	Etudiant	Maitre De Stage	Etat Du Stage	Actions
1	IAPF-1-A	2019-10-12 au 2020-03-04	350	et1	maitretf	EVALUATION_DEGREE_MS	
2	IAPF-1-A	2019-09-10 au 2020-03-04	300	et2	maitretf	ENDED	
3	IAPF-1-A	2019-12-15 au 2020-03-04	250	et3	maitretf	ENDED	
4	ISIP-1-A	2019-12-15 au 2020-03-04	250	et3	maitretf	ENDED	

ISFCE
Home
Gestion Comptes
Stage
Evaluations Stages

Autre
Français

Bienvenu au
Evaluation des stage
IPID

Nous avons créé deux listes simplement pour faciliter la recherche d'un étudiant pour son évaluation, ainsi quand un maitre de stage ou un professeur veut évaluer un stage il va sur la liste qui contient justes les stages à évaluer.



Conclusion

Pour conclure sur ce « nouveau » framework qu'est Spring Boot et Thymeleaf, nous aimerions dire qu'elle fut très enrichissante et très intéressante dans tous ses aspects. Nous appréhendons grandement ce « nouveau framework », dû entre autres à une syntaxe très différente de ce que nous avons pu connaître, notamment dû à l'utilisation des annotations ou encore l'import des dépendances. Nous ne pensions pas dire à la fin de ce cours que nous finirons autant par nous familiariser au point d'en perdre nos habitudes avec d'autres frameworks (rires). Tout ce cheminement (de la création jusqu'à la finition) de l'application, nous a permis de garder et consolider notre cohésion de groupe bien que parfois nous n'étions pas d'accord sur toutes les implémentations, nous avons toujours réussi à trouver un terrain d'entente de façon à concrétiser nos différentes idées pour concevoir un projet commun. Nous aimerions enfin remercier le responsable du cours, Monsieur Van Houdenhove pour son implication à nous transmettre son savoir, le contenu complet du syllabus fourni (qui fut par ailleurs notre premier guide dans cet apprentissage et notamment sur certains « aspects » encore nouveaux pour nous) et pour nous avoir appris davantage sur cette technologie.