

Juin 2015

Rapport de projet intégré

GEORGES Nicolas, HAINAUX Jordan, JACQUES Loïc, LONCIN Benjamin et TACCHI Pierre

2ème informatique de gestion



Table des matières

1. Introduction
2. Réel perçu
3. But du logiciel
4. Besoins fonctionnels et non fonctionnels
5. Phase d'analyse
 - 5.1. Diagramme des cas d'utilisation
 - 5.2. Diagramme des classes métiers
 - 5.3. Diagrammes d'activités
 - 5.4. Diagramme entité-association
6. Phase de conception
 - 6.1. Diagramme des classes raffiné
 - 6.2. Schéma relationnel
7. Choix technologiques
8. Difficultés rencontrées
9. Perspectives
10. Conclusion

Introduction

Ce présent rapport décrit les différents points développés par notre groupe de 5 étudiants en 2ème informatique de gestion au cours de cette année scolaire 2014-2015. Ainsi, nous avons rédigé un texte expliquant le réel perçu du projet à développer ainsi que des explications concernant les buts de l'outil que nous avons mis en place. Nous avons réalisé les diagrammes des cas d'utilisations par rapport à la liste des fonctionnalités demandées, modifié quelques points du diagramme des classes et du schéma entité-association proposés par les enseignants et réalisé les diagrammes d'activités que nous expliquerons dans la suite de ce rapport (analyse et conception). Les choix technologiques (en plus de ceux imposés au départ) et des pistes d'améliorations pour le projet final sont défendus en fin de rapport. Nous concluons ce rapport par une démonstration via des prises d'écrans du logiciel implémenté.

Réel perçu

Lorsqu'un étudiant en informatique arrive en deuxième année, il doit choisir un sujet pour son travail de fin d'études (TFE) qui se fera sur un total de deux années. Pour ce travail, l'étudiant doit proposer un ou plusieurs sujets. Les quatre professeurs d'informatique décideront ensuite pour chaque étudiant les propositions qu'ils acceptent et celles qu'ils refusent. L'étudiant peut alors décider quelle proposition acceptée il veut concrétiser en TFE. Il peut aussi refaire d'autres propositions. Il est souhaitable que le travail de l'étudiant soit pour une autre personne que lui. En d'autres mots, il serait préférable qu'il ait un client.

Une fois son TFE choisi, l'étudiant se voit attribuer un promoteur dont le rôle sera de suivre l'étudiant le long de son parcours. L'étudiant sera ensuite évalué par ce promoteur durant le restant de ses études sur des travaux à rendre à des dates précises via un site web appelé iCampus. C'est sur ce même site que ces travaux seront corrigés. Il est également demandé à l'étudiant de communiquer avec son client (s'il en a un) et son promoteur.

En fin de troisième année, l'étudiant doit faire une défense orale de son TFE devant les professeurs et un président de jury.

Au début de leur 3e année, les étudiants doivent choisir un stage qu'ils exécuteront pendant tout le deuxième quadrimestre de cette année. Pour cela, il doivent, de manière similaire aux propositions de TFE, proposer leurs stages aux professeurs.

Les propositions de stage ainsi concrétisées seront supervisées et évaluées par un maître de stage côté entreprise et par un promoteur côté école.

But du logiciel

Le but du logiciel est de faciliter les démarches des étudiants avant et pendant leur stage et le long de la réalisation de leur TFE. Le logiciel permettra de faciliter la communication entre les membres (étudiants, professeurs, maitres de stages et président du jury) en leur donnant à diverses informations dont les TFE, les stages et les propositions de stages des étudiants.

Le logiciel proposé a en conséquence comme autre utilité d'aider les étudiants des différentes sections à choisir leur futur stage en connaissant l'entreprise grâce aux informations collectées par les professeurs et étudiants des années précédentes. Il vise également à tenir à jour le déroulement du stage/TFE en proposant d'y insérer des suivis qui seront consultées et notés par les professeurs. De plus, les maitres de stages et le président de jury des TFE auront la possibilité de noter les étudiants directement sur les suivis postés, mais également de mettre des points pour chaque partie de l'évaluation (parties écrite, orale, suivi, démonstration éventuelle, ...).

Enfin, outre ses fonctionnalités, un projet comme celui-ci permettra de faciliter la collaboration entre les professeurs et les étudiants, et proposera une manière structurée et efficace de gérer les données sur les stages et les TFE pour chacune des parties participant tout au long du cursus de chaque étudiant.

Besoins fonctionnels

1. Gérer (ajouter/modifier/supprimer) prof (pour *professeur*)
2. Gérer (ajouter/modifier/supprimer) étudiant (pour *professeur*)
3. Gérer maître de stage (pour *professeur*)
4. Gérer président de jury (pour *professeur*)
5. Poster proposition de stage (pour *professeur* et *étudiant*) pour un lieu de stage existant ou pour un lieu de stage inexistant (*extends* ajout lieu de stage)
6. Gérer entreprise (pour *professeur*)
7. Valider une proposition de stage (*extends* ajout maître stage) (pour *professeur*)
8. Gérer TFE (pour *professeur*)
9. Gérer défense (pour *professeur*)

10. Poster/modifier évaluation (extends ajout critère) (pour *professeur*, *président de jury*, *maître de stage*).
11. Gérer critères (pour *professeur*)
12. Rechercher stages et propositions de stage (pour *étudiant* et prof)
13. Poster un commentaire sur un stage (pour *étudiant* et *professeur*)
14. Rechercher TFE (selon technologies, selon année académique) (pour *étudiant* et *professeur*)
15. Afficher défenses (pour *étudiant* et *professeur*)
16. Pouvoir visualiser l'état de validation de sa proposition de stage (pour *étudiant*).
17. Poster un suivi pour une échéance (pour *étudiant*)
18. Corriger un suivi (pour *professeur*)
19. Valider un suivi (pour *professeur*)

Besoins non-fonctionnels

1. Langue : français ;
2. Pas de limite au nombre d'utilisateurs ;

Phase d'analyse

Diagramme des cas d'utilisation

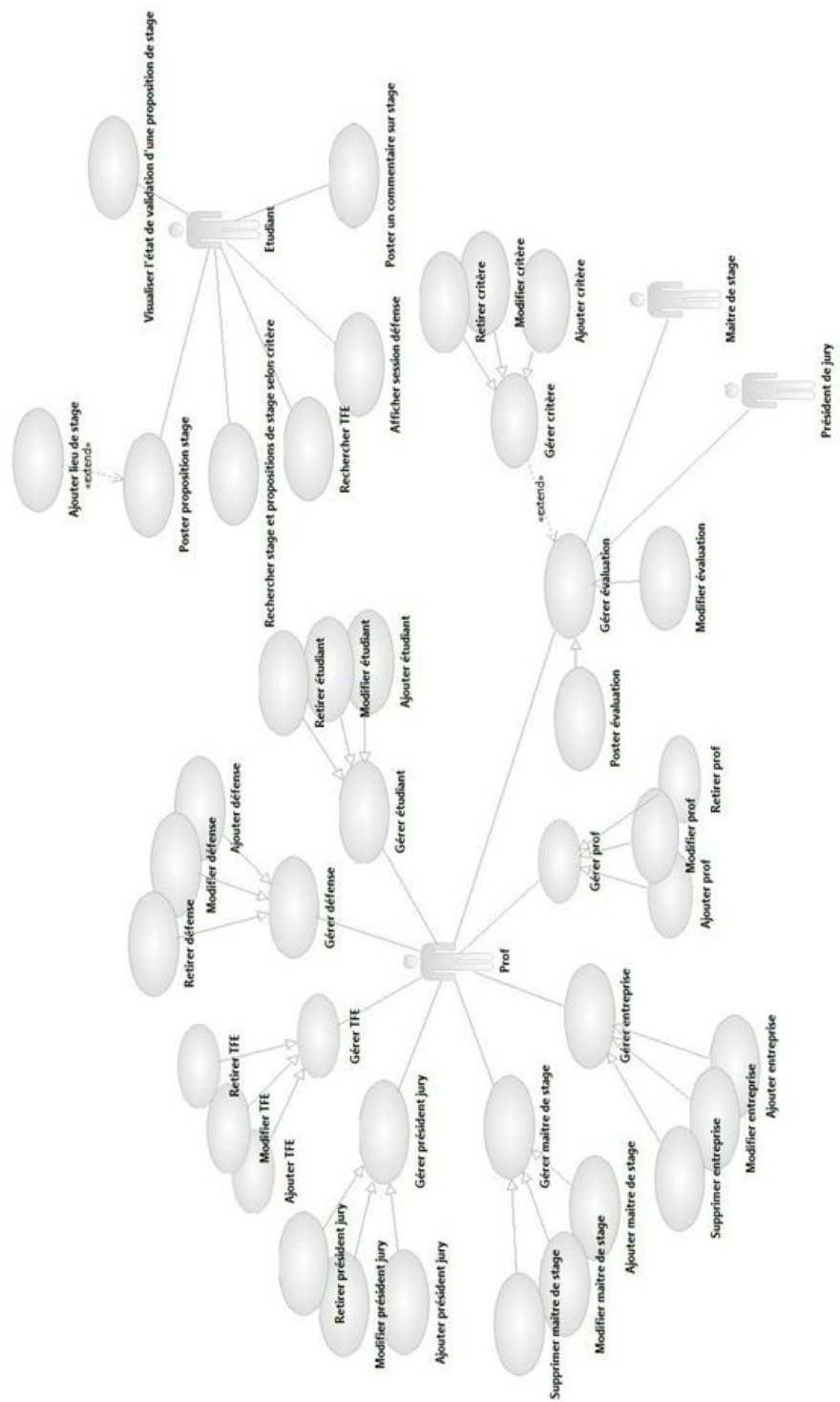
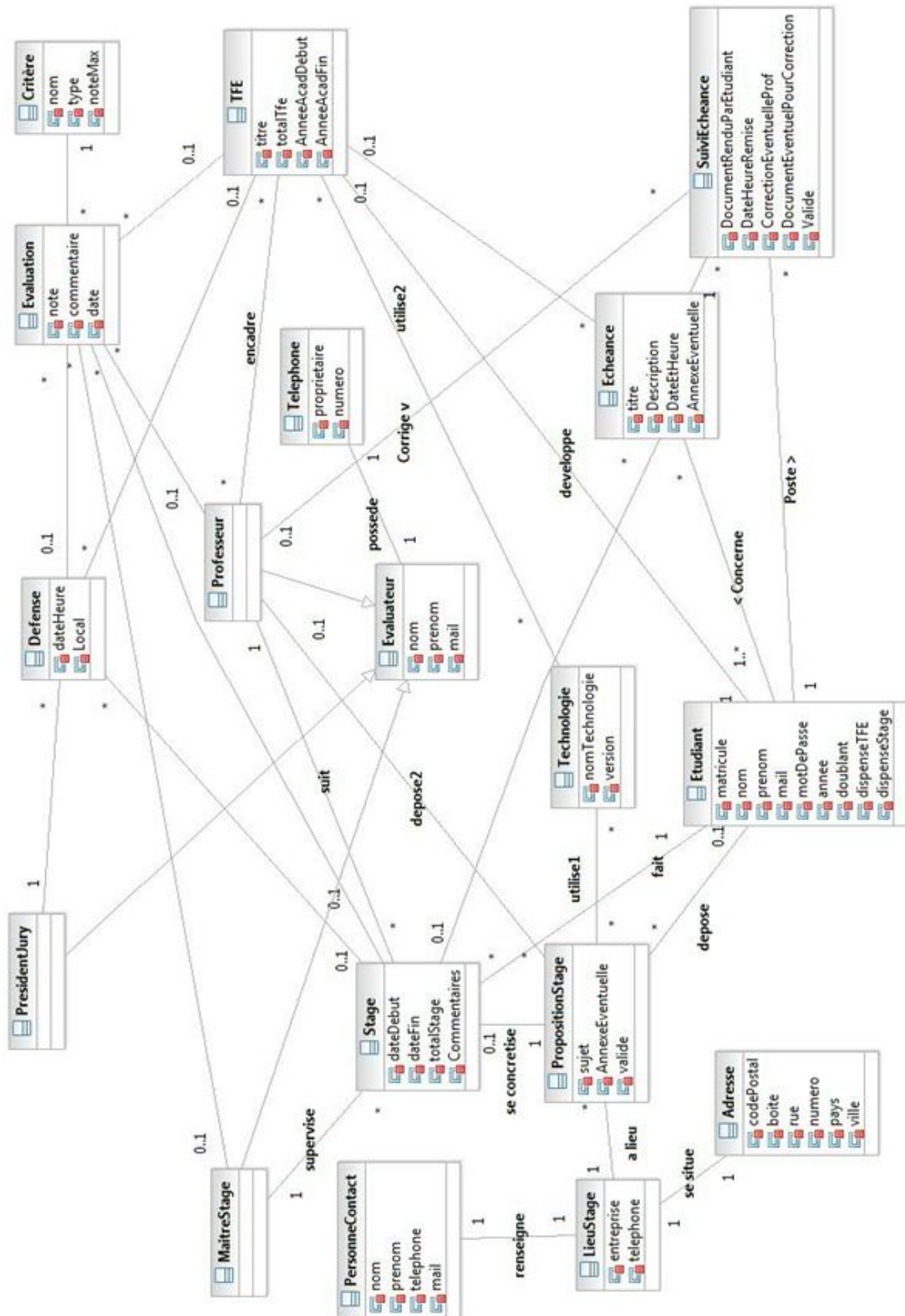


Diagramme des classes métiers



Explications :

Les étudiants ont un matricule, un nom, un prénom, une adresse mail, un mot de passe, une année et des variables indiquant s'il sont doublants ou non, s'ils sont dispensés de leur TFE et s'ils sont dispensés de leur stage. Ils peuvent déposer plusieurs propositions de stage mais ils ne peuvent faire qu'un stage maximum. Il développe également un stage et un TFE. Pour ceux-ci, il est évalué sur plusieurs remises de travaux (objet SuiviEcheance) pour chacune des échéances.

Un TFE est caractérisé par un titre, un total (points sur lesquels seront calculés la note), son année académique de début et celle de fin. Chaque TFE est promu par un professeur appelé promoteur et a plusieurs défenses (car l'étudiant peut rater une défense) et plusieurs évaluations. Il utilise plusieurs technologies ; les technologies utilisées par cette proposition ont un nom et une version (exemple : nom=java, version=8). Il est fait par un étudiant et possède plusieurs échéances.

Une proposition de stage a un sujet (exemple : développement mobile chez Habaguanex), une annexe éventuelle (PDF informatif) et une valeur indiquant si elle est validée ou pas. Le lieu comprend un nom d'entreprise, adresse (CP, boîte, rue, numéro de rue, pays et ville), une personne de contact (nom, prénom, téléphone et adresse mail) et un numéro de téléphone.

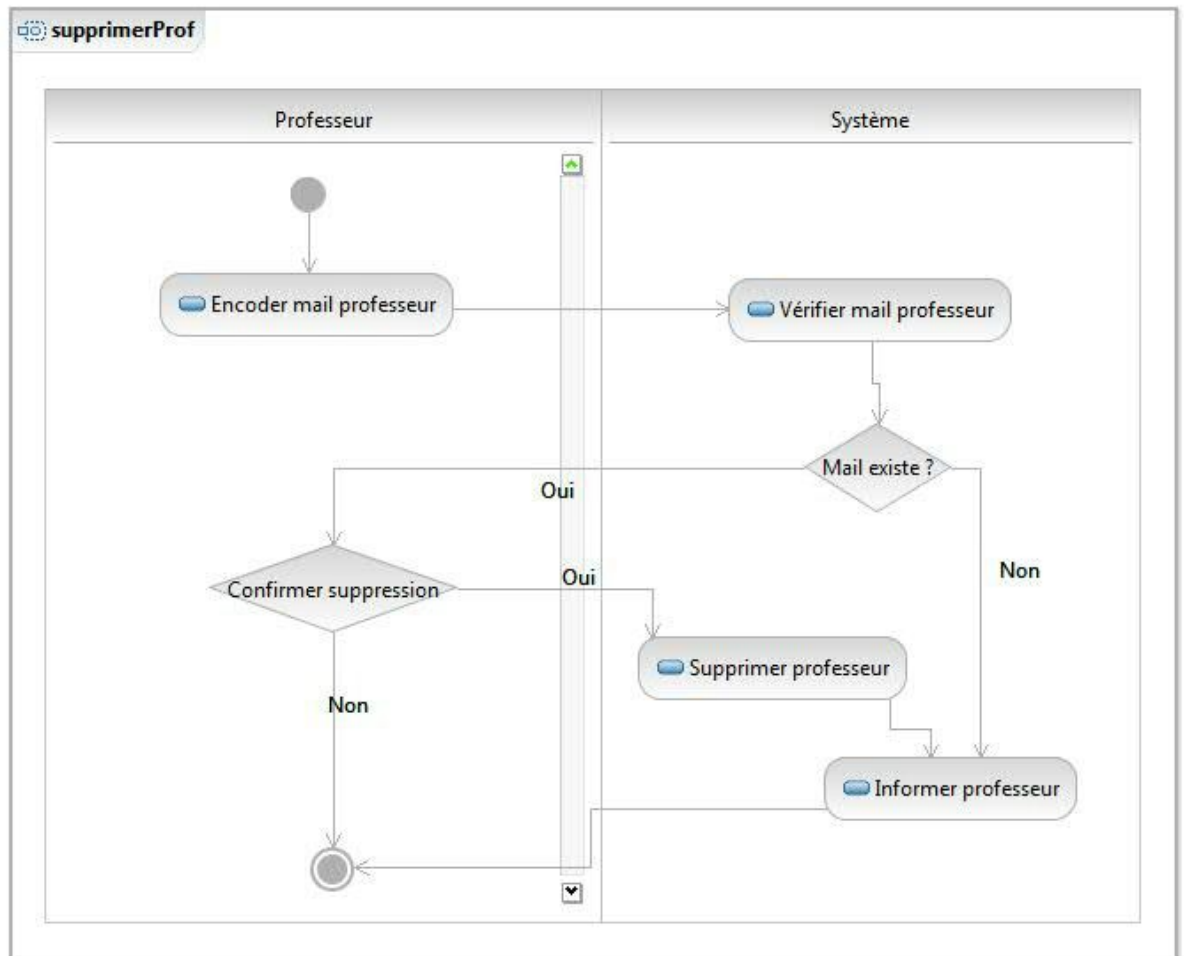
Chaque proposition peut être concrétisée en stage. Dans ce cas, un objet Stage associé à la proposition est créé. Cet objet contient les informations supplémentaires suivantes : date de début, date de fin, total (le nombre de points sur lequel on note les évaluations) et commentaire. Il est également précisé quelles technologies il utilise. Un stage est obligatoirement supervisé par un maître de stage et un professeur (promoteur) peut avoir plusieurs défenses et plusieurs évaluations. Il est fait par un étudiant et possède plusieurs échéances.

Chaque échéance est caractérisée par un titre, une description, une heure et une annexe éventuelle. Chaque suivi de cette échéance contient un document rendu par l'étudiant, une date de remise (incluant l'heure), une correction du professeur (texte) avec une annexe éventuelle et une valeur indiquant si le suivi est valide ou pas (il ne faut pas que la date de remise dépasse la date de l'échéance).

Une défense se fait à une certaine date, à une certaine heure, dans un certain local et concerne soit un TFE, soit un stage. Chaque évaluation a une note, un commentaire et la date à laquelle elle a été faite. Chaque défense est catégorisée dans un critère.

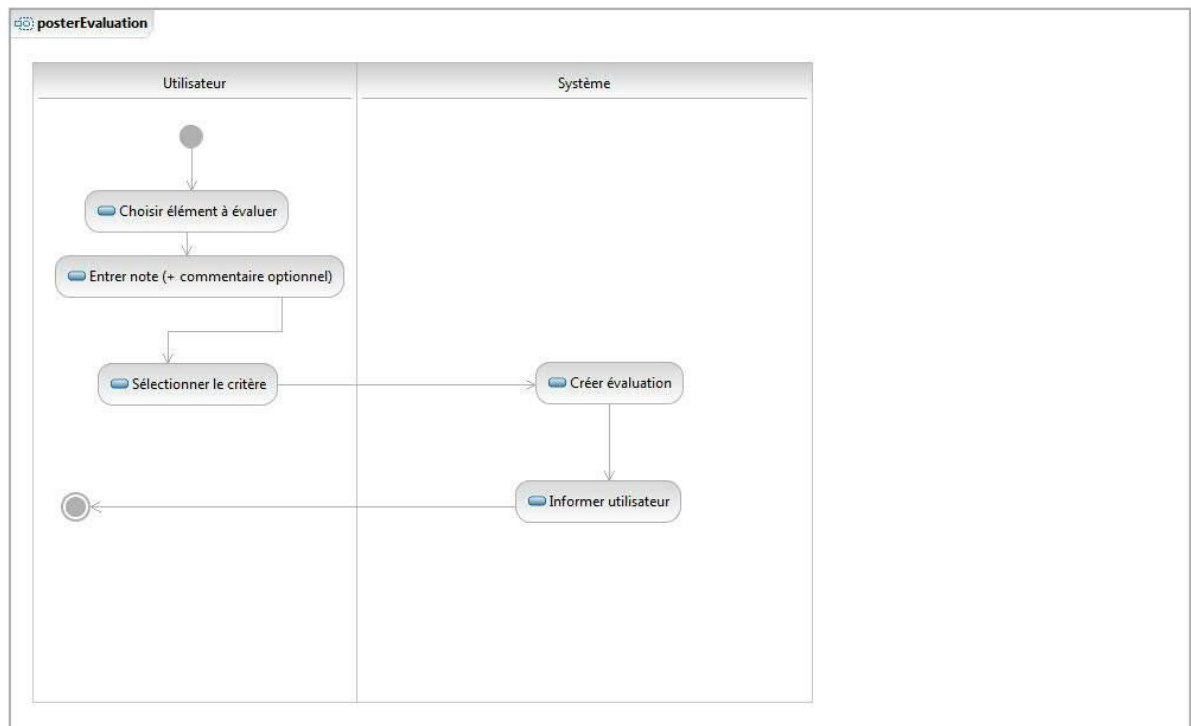
Diagrammes d'activités

Supprimer un professeur



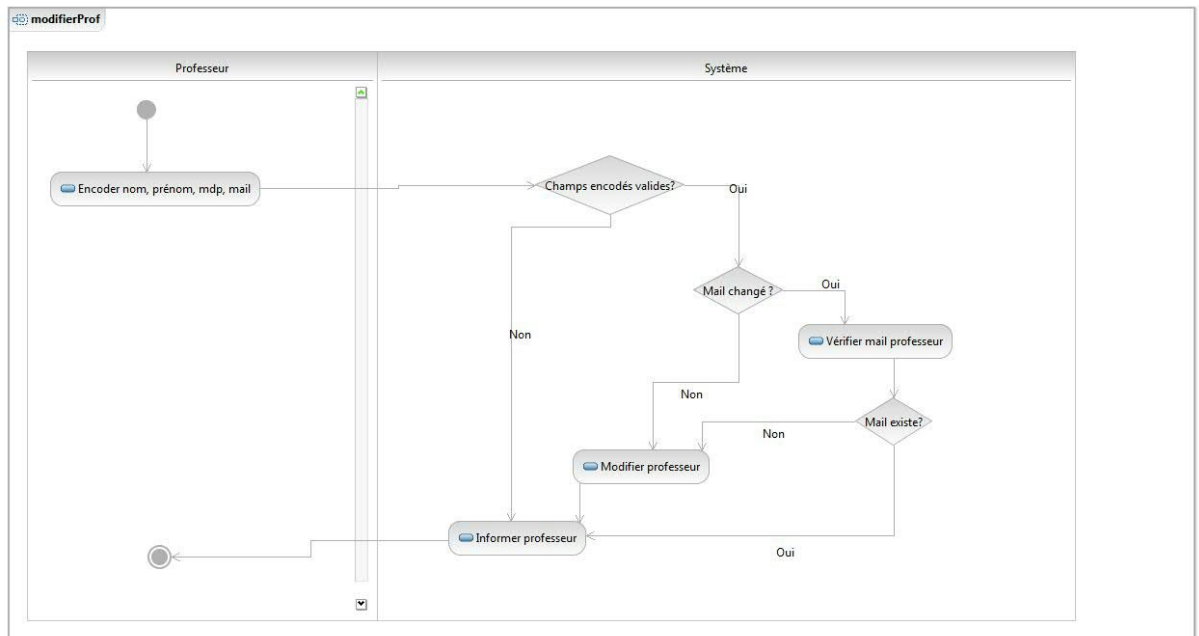
Pour supprimer un professeur, il faut sélectionner le professeur dans la liste, ce qui fait que le programme utilise le mail pour vérifier qu'il existe bien. S'il existe, on demande confirmation de la suppression. Si l'utilisateur confirme, le professeur référant à l'adresse mail du professeur sélectionné est supprimé.

Poster une évaluation



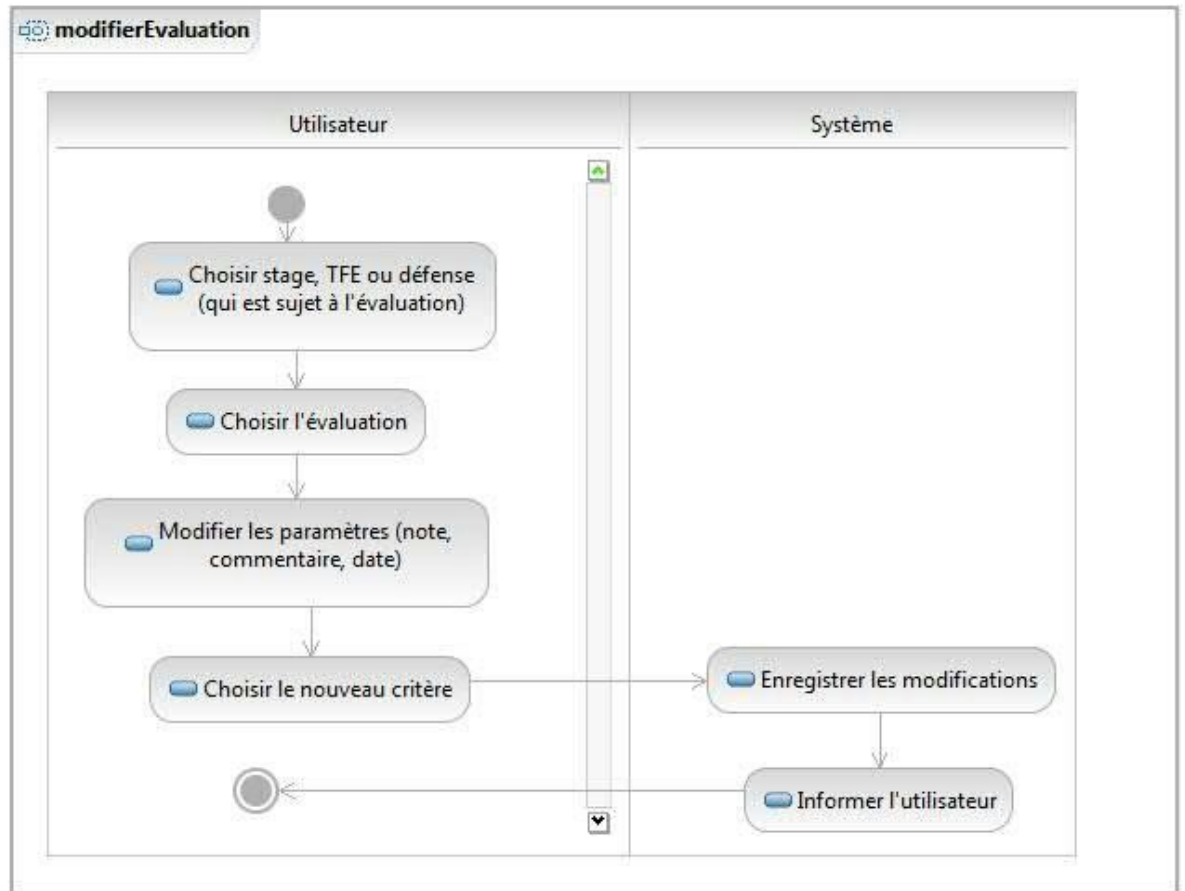
Lorsqu'un évaluateur souhaite créer une évaluation sur une échéance (suivi), il choisit l'étudiant et l'échéance. Il arrive alors sur une page, où il peut entrer sa note ainsi qu'un commentaire s'il le souhaite. Ensuite, il peut sélectionner le critère pour lequel il entre l'évaluation.

Modifier un professeur



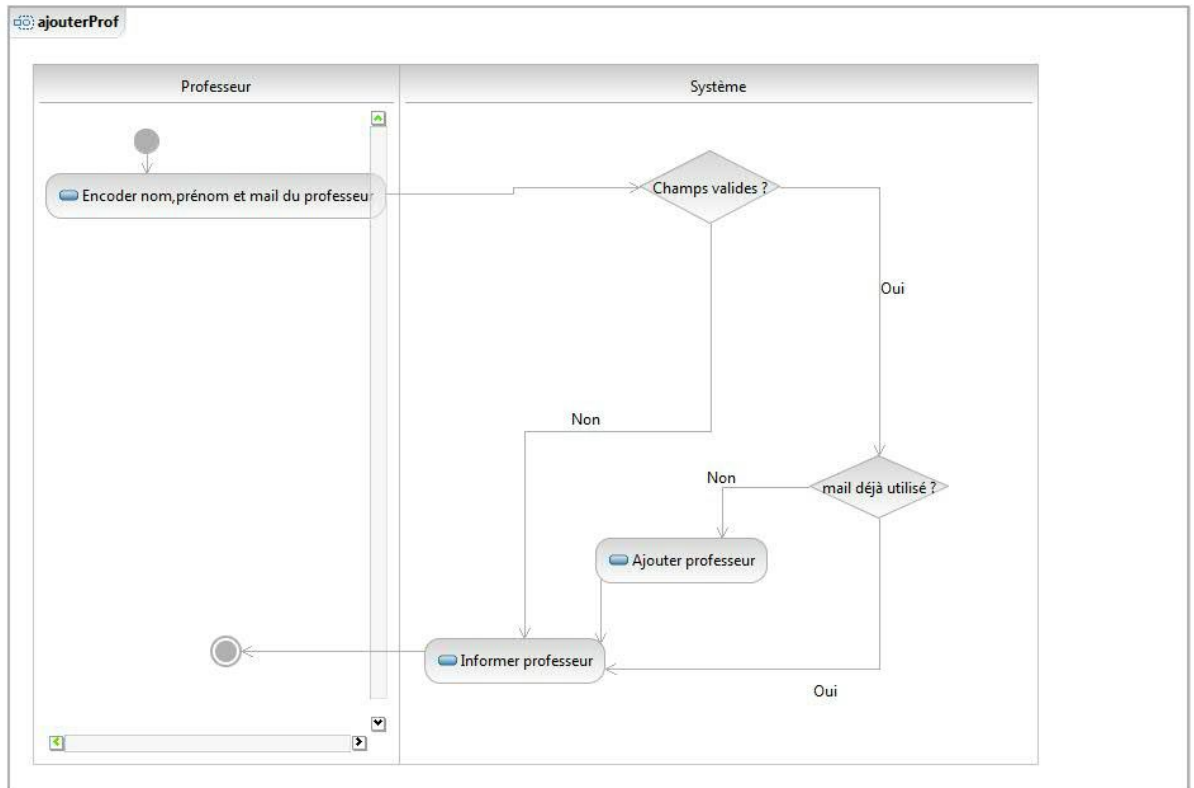
Lorsqu'un professeur veut modifier les informations d'un autre professeur ou ses propres informations, il encode les données. Les champs sont ensuite validés et si le mail a changé on vérifie qu'il n'est pas déjà utilisé par un autre, s'il l'est, le professeur n'est pas modifié, sinon, on modifie les informations du professeur. On informe alors l'utilisateur du résultat de la demande.

Modifier une évaluation



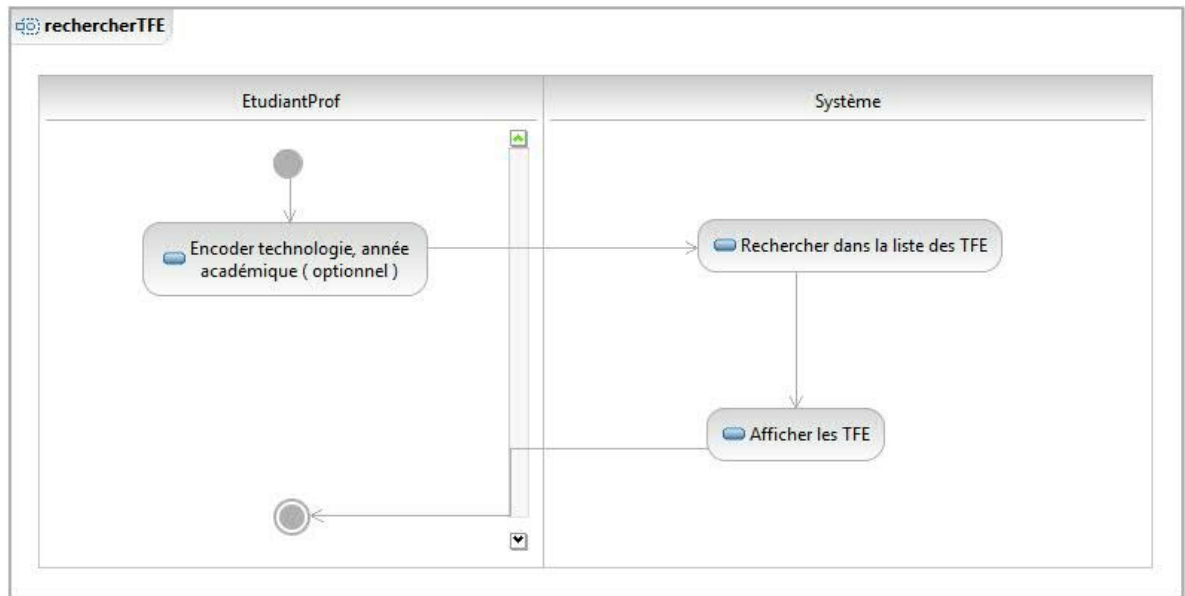
Pour modifier une évaluation, l'utilisateur doit choisir le stage, le tfe ou la défense ensuite choisir l'évaluation sur ce sujet. Il arrive alors sur une autre page où il lui est demandé d'entrer les différents paramètres de l'évaluation. Il peut alors choisir un nouveau critère ou garder le précédent. Les modifications sont ensuite enregistrées et l'utilisateur est informé du résultat.

Ajouter un professeur



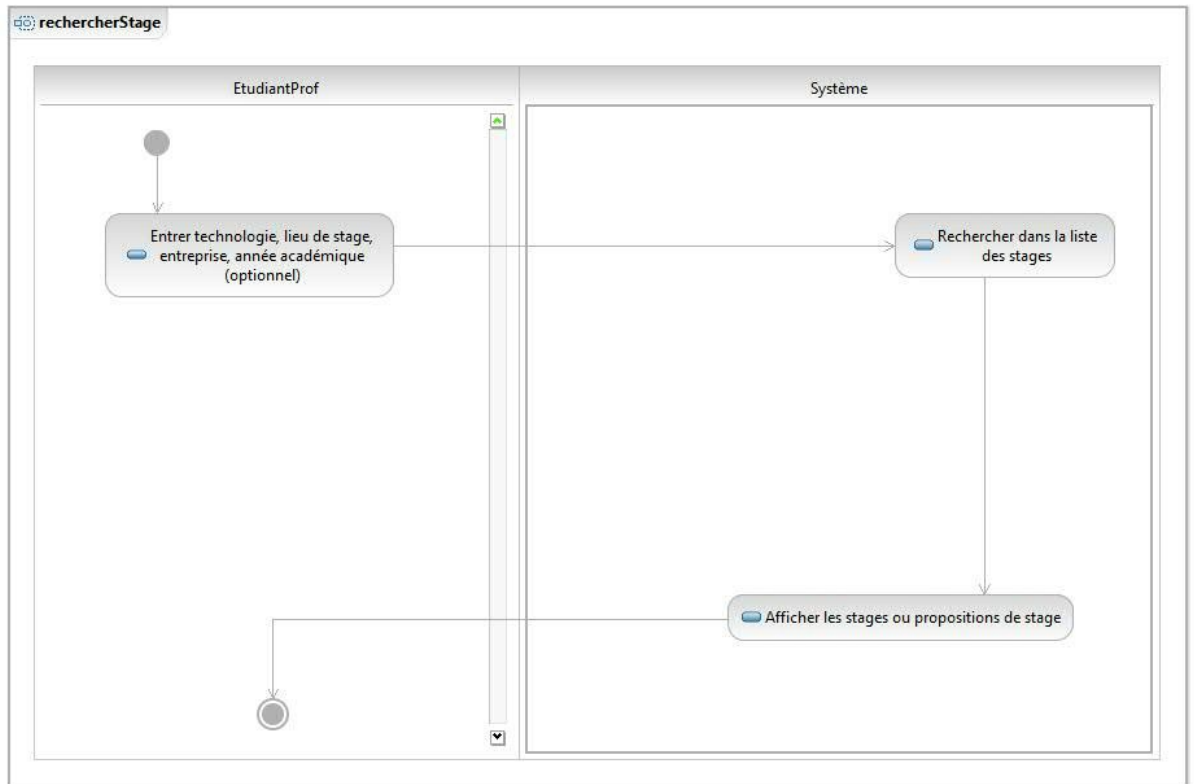
Pour ajouter un professeur, l'utilisateur (autre professeur, admin) entre ses informations. Les champs sont ensuite vérifiés. Si le mail est déjà utilisé, le professeur n'est pas ajouté. Le résultat est alors communiqué à l'utilisateur.

Rechercher un TFE

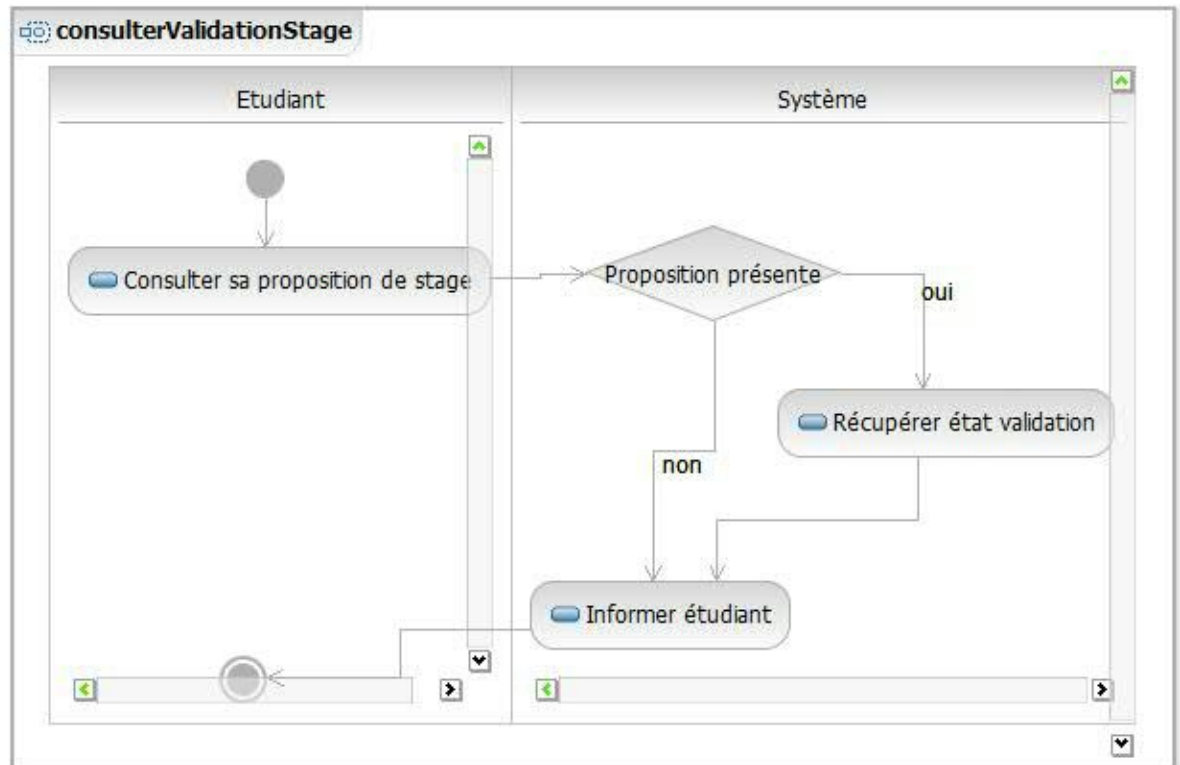


Pour rechercher un TFE, l'utilisateur encode les informations nécessaires à sa recherche, ensuite le système recherche dans la liste des TFE

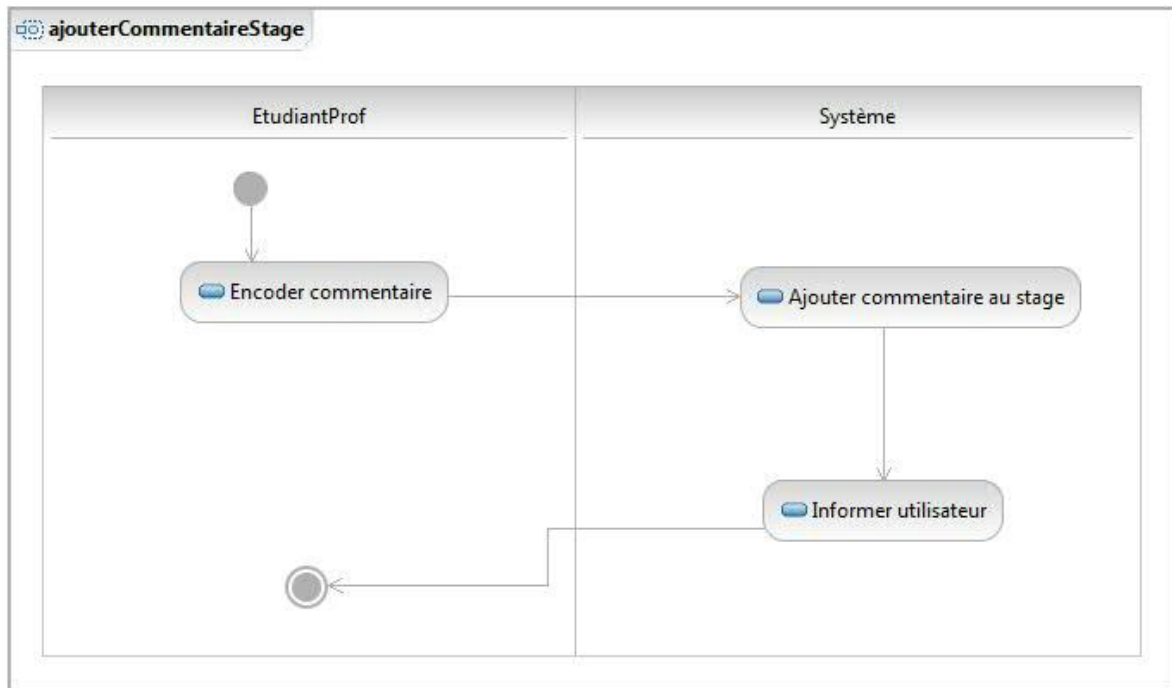
Rechercher un stage



Consulter l'état de validation d'un stage



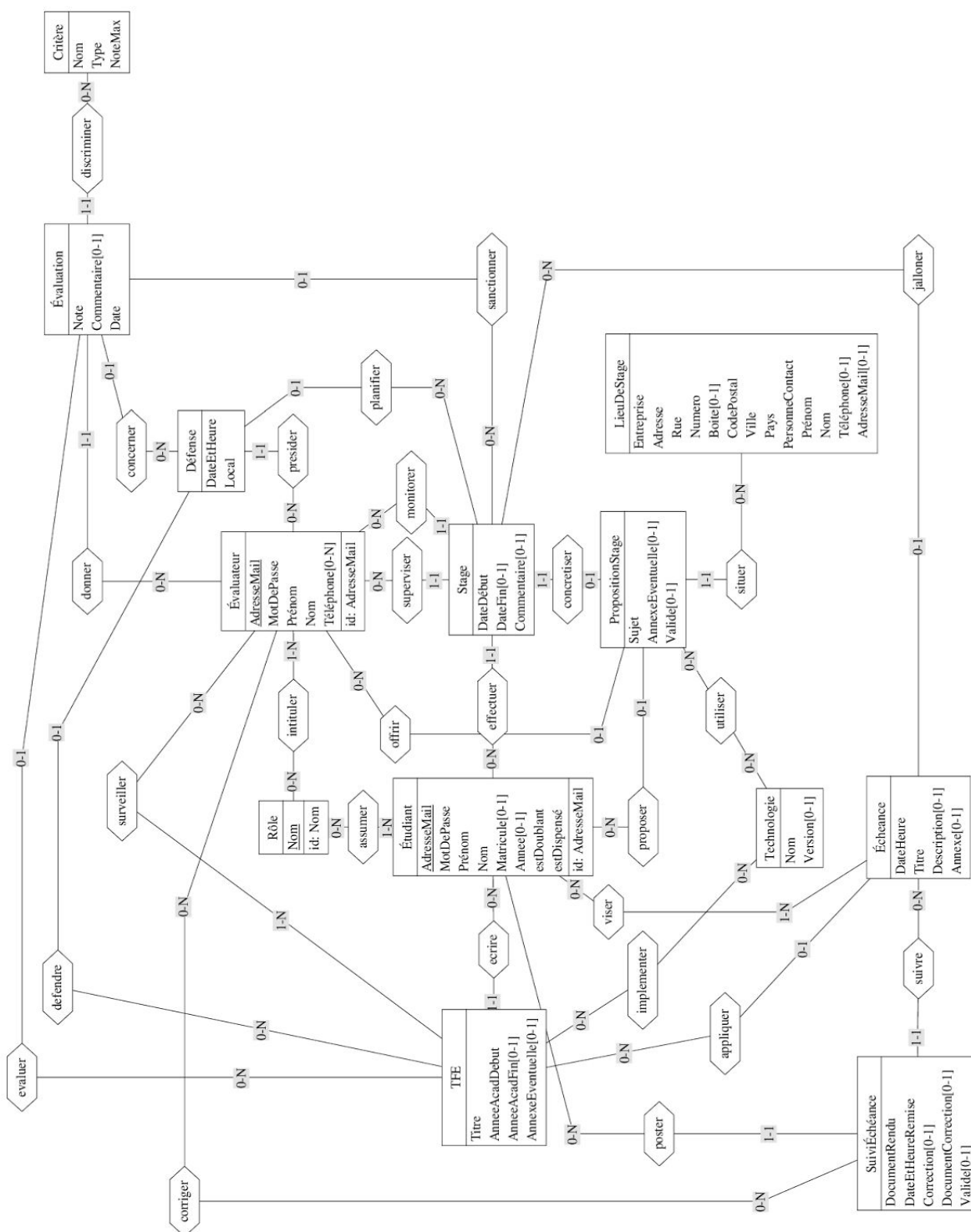
Ajouter un commentaire à un stage



L'utilisateur entre le commentaire, puis le système enregistre la modification. Il informe l'utilisateur du bon déroulement de la tâche

Phase de conception

Diagramme entité-association



page 19 sur 23

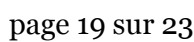
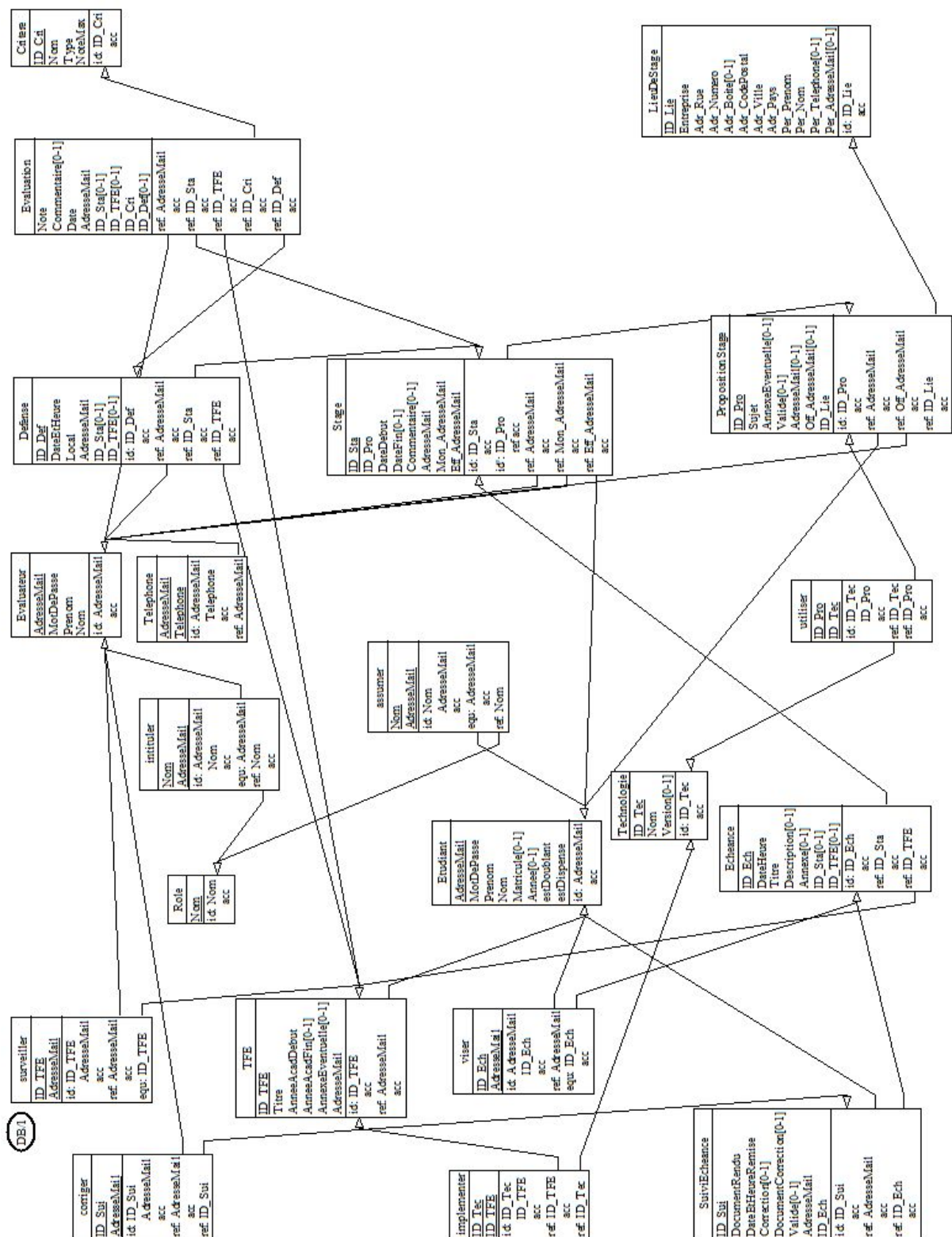


Schéma relationnel



Choix technologiques

Il n'y avait pas grands choix à faire pour implémenter ce logiciel : les langages sont imposés par les enseignants pour ce qui est du plus gros du projet. Il est vrai qu'étant donné notre cursus, ce projet intégré comptant pour une grosse partie des points de l'année des cours d'analyse, de PAOO, de web et de laboratoire BD, il fallait utiliser les langages appris et censés connus. Ainsi, nous avons implémenté les classes métier et les DAO en Java, puis avons réalisé les différentes vues des utilisateurs avec Vaadin, qui est un framework pour application web en open-source s'interfaçant très facilement avec Java. Si choix il y a eu, ce fut donc de la part du corps enseignant, mis à part l'utilisation du langage Mysql pour ce qui est de la base de données. Étant le langage le plus courant lorsqu'il s'agit d'applications web (qui comprend une application web de gestion pour la base de données : phpMyAdmin), nous avons pensé qu'il serait plus judicieux de l'utiliser dans le cas de notre projet plutôt que d'utiliser une base de données Oracle. Nous préférons d'ailleurs ce langage à Oracle. De plus, il y a des auto-incréments et des booléens. Au final, ces choix nous ont permis d'apprendre deux nouveaux langages intéressants : Vaadin, qui pourrait devenir intéressant bien qu'il montre quelques problèmes (nous en avons rencontré plus d'un lors de la phase de conception), et Mysql, bien que ce langage ne diffère pas beaucoup du langage Oracle sql.

Difficultés rencontrées

Du point de vue relationnel, on pourrait penser qu'un groupe de 5 étudiants a plus de chances de réaliser un projet de cette ampleur qu'un groupe de 2 ou 3. Pourtant, le nombre n'est pas toujours la donnée qui compte le plus dans un travail de groupe : notre groupe en est la parfaite illustration. En effet, l'entente, l'ambiance, les mentalités et la façon de travailler sont d'avantage des données qui définissent un groupe qui fonctionne. Cela dit, il ne faut pas penser que notre groupe n'ait pas fonctionné lorsque l'on relate ce genre de réflexions. Il est évident que notre groupe rassemblait 5 étudiants complètement différents, bien que certains se soient plus facilement intégrés que d'autres dans ce projet. Il est vrai aussi que nous avons eut des tensions dans le groupe, des coups de gueules... bref, l'ambiance laissait parfois à désirer. Cependant, nous avons réussi à passer outre ces difficultés car notre but n'était ni plus ni moins que de réussir ce projet tous ensemble, avec les bons et les moins bons côtés de chacun. Du point de vue technique, au début du projet, nous avons eut quelques difficultés avec le cahier des charges et l'analyse.

Il n'est pas à démontrer que l'énoncé de base de ce projet était très vague... il a donc fallu réfléchir, poser des questions, et réaliser des diagrammes pour bien se mettre d'accord sur la marche à suivre. Souvent en désaccord entre eux, les professeurs ne nous ont pas facilité la tâche au début pour savoir exactement ce qu'il fallait réaliser (était-ce voulu ou pas, cela nous aura donné un bon exercice de remise en question continue).

Du point de vue implémentation, la découverte de Vaadin a été l'une des parties les plus difficiles. En effet, nous avons eut peu de cours sur ce *framework* et nous partions plus ou moins en terrain inconnu. Heureusement nous avons trouvé comment l'utiliser et avons implémenté les vues de l'application. L'overlay a été difficile à mettre en place. En effet, nous n'étions pas habitué à devoir injecter du css dynamiquement dans une page afin qu'elle corresponde à nos attentes via Vaadin, et il fallait chercher la meilleure façon de le faire pour pouvoir réaliser la vue du menu utilisateur (profil).

Perspectives

Afin de faciliter au mieux la gestion des TFE, il pourrait y avoir certaines modifications quant à la manière de réaliser ce TFE à son début. En effet, dans notre implémentation, un sujet de TFE correspond à une échéance. Or, une proposition de stage n'en est pas. Qui des deux styles de propositions a raison? Les deux manières sont bonnes, bien que l'une d'elle (sujet de TFE) implique que le TFE ne possède pas de document explicatif au départ, il aurait pu être intéressant d'en avoir un afin de présenter au mieux les idées de départ de l'étudiant et ses raisons qui l'ont poussé à soumettre ce sujet.

Notre application pourrait, à terme, rassembler la gestion des stages et TFEs de toutes les sections de la HERS, pour le suivi des échéances et la communication entre les étudiants et les enseignants. Il est évident que plusieurs parties et différentes exigences seraient à rajouter, mais la base que ce projet offre une grande marge d'améliorations et d'opportunités, toutes dans le but de faciliter l'organisation, le respect des échéances, et l'implication des étudiants.

Conclusion

Ce projet nous aura appris beaucoup de choses, non seulement du point de vue technique, mais aussi du point de vue relationnel et travail de groupe. Nous avons pu apprendre un nouveau langage (Vaadin) et expérimenter le fait de développer un projet en ayant des contraintes sur les langages à utiliser, ainsi que de respecter les différentes exigences demandées par un (des) client(s) (le corps enseignant). Nous avons pu réfléchir ensemble sur la manière d'implémenter mais aussi d'analyser ce projet tous ensemble. Certaines réflexions ont conduit à prendre des initiatives telles que d'utiliser Mysql plutôt qu'Oracle sql,... Cette mise en situation aura donc été globalement une belle expérience pour tous, il a en effet fallu penser en groupe avant de penser aux attentes personnelles.