



HAL
open science

Dynamique et ajustement des rôles d'étudiant et d'enseignant au cours d'une formation basée sur la pédagogie par projet

Philippe Saliou, Vincent Ribaud, Françoise Le Borgne-Uguen

► To cite this version:

Philippe Saliou, Vincent Ribaud, Françoise Le Borgne-Uguen. Dynamique et ajustement des rôles d'étudiant et d'enseignant au cours d'une formation basée sur la pédagogie par projet. URAFF 2004. Unité de Recherche-Action en Formation de Formateurs (URAFF), Université de Bretagne Occidentale, Dec 2004, Brest, France. pp.1-11. hal-01448779

HAL Id: hal-01448779

<https://hal.univ-brest.fr/hal-01448779>

Submitted on 29 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Dynamique et ajustement des rôles d'étudiant et d'enseignant au cours d'une formation basée sur la pédagogie par projet

P. Saliou, V. Ribaud et F. Le Borgne-Uguen²

EA2215, Département informatique, Faculté des sciences, 29285 Brest Cedex, France

² EA3149, Atelier de recherche sociologique, Université de Brest, 29285 Brest Cedex, France

E-mail: {philippe.saliou, vincent.ribaud, francoise.leborgne-uguen} @univ-brest.fr

1. Introduction

1.1. Le contexte : un parcours de formation en DESS Ingénierie Logiciel par Immersion

Depuis Septembre 2002, l'Université de Bretagne Occidentale propose un parcours du DESS d'Informatique adapté aux étudiant(e)s diplômés d'un IUP Informatique ou provenant du secteur technologique. Ce parcours a pour objectif la maîtrise de l'ingénierie du logiciel et porte sur l'ensemble des connaissances, des procédés et des acquis scientifiques et techniques mis en application pour la conception et la fabrication d'un logiciel depuis sa commande par le maître d'ouvrage jusqu'à sa mise en exploitation.

L'idée principale de cette formation expérimentale, baptisée « **Ingénierie Logiciel par Immersion** », est de faire en sorte que l'apprentissage du monde professionnel soit importé dans les murs de l'université. Les étudiant(e)s travaillent en équipe pour concevoir et réaliser un projet logiciel, basé sur des principes solides d'ingénierie du logiciel. Le dispositif pédagogique reproduit autant que possible les phénomènes que les futurs professionnels seront amenés à rencontrer : un environnement de travail professionnel, la relation client-fournisseur, le respect d'un référentiel de développement, l'emploi de méthodes et des outils associés, la coopération entre membres d'une équipe, etc.

Cette nouvelle formation est une réponse à la difficulté d'enseigner l'ingénierie du logiciel dans le paradigme d'enseignement classique [CC2001], [PGL00], [Mey01]. Ce paradigme aboutit très souvent à un enseignement morcelé des seules activités techniques de l'ingénierie logiciel. D'où l'idée d'aborder toutes les activités d'ingénierie logiciel dans un paradigme d'apprentissage [Tar98] :

- en les mettant dans la perspective globale d'un projet réel,
- et en mettant davantage l'accent sur le processus de conception et de fabrication du logiciel dont les activités principales ont été réparties dans trois domaines principaux : gestion de projet logiciel, ingénierie du développement logiciel, support au développement logiciel.

Le dispositif global d'apprentissage de l'année est guidé par le processus de conception et de fabrication qui définit notamment le rôle et l'enchaînement des étapes d'un projet. Une première itération du processus (de 4 mois) permet l'acquisition des connaissances nécessaires à chaque étape, une deuxième itération (de 2 mois) doit les transformer en compétences et enfin une dernière itération (de 4 à 6 mois) doit permettre de les appliquer en entreprise.

1.2. Dispositif de recherche-action par participation-observante

Dès le début de la formation, un dispositif de recherche a été mis en place, dans le cadre de l'Unité de Recherche-Action initiée par le Service de Formation Continue et d'Education Permanente de l'UBO. L'objet du groupe de recherche est le recueil d'observations sur les apprentissages effectués tels qu'ils sont décrits par les acteurs, étudiant(e)s et enseignants, et mis en lien avec leur point de vue sur les méthodes et moyens en œuvre. Ce groupe était constitué des 10 étudiant(e)s engagé(e)s dans l'expérience et de 2 représentants des autres étudiant(e)s du DESS, des 2 enseignants-chercheurs du département informatique à l'origine de ce dispositif (Vincent Ribaud et Philippe Saliou) et de F. Le Borgne-Uguen, enseignant-chercheur en sociologie et chargée de mission-URAFF.

Les données recueillies et analysées sont les suivantes :

- Observation des équipes en situation de travail et de fourniture de réalisation ;
- 4 Réunions- discussions, d'une durée de 2 à 3 heures tous les mois et demi,
- 10 Récits écrits prenant la forme d'un bilan intermédiaire de la formation au terme de la première itération de 4 mois (10 pages / étudiant(e)).

L'analyse de ces données a permis de satisfaire à l'objectif d'identifier les moments-clefs et les pratiques mises en œuvre par les étudiant(e)s lors des différentes phases du déroulement de la formation. Le résultat correspondant met l'accent sur le passage d'un métier d'étudiant « standardisé » à l'initiation aux différentes facettes du métier d'étudiant engagé dans l'appropriation de savoirs qu'il va bientôt mettre en œuvre dans une situation professionnelle : celle d'ingénieur du logiciel.(cf. § 3.2).

Ce groupe de recherche-action a eu des retombées sur l'ajustement du parcours de formation lui-même. Une partie de cet ajustement a abouti entre autres, à la définition des différents rôles qu'un enseignant est amené à assumer dans ce type de formation (cf. § 3.1).

2. Présentation de la formation

2.1. Description générale

L'année de formation est structurée en trois périodes : une période d'apprentissage tutoré de 4 mois, une période de mise en pratique accompagnée de deux mois, une période de stage en entreprise de 4 à 6 mois (cf. figure 1). Les apprentissages sont relatifs à la discipline enseignée (l'ingénierie du logiciel) et complétés par des enseignements de langues et de communication.

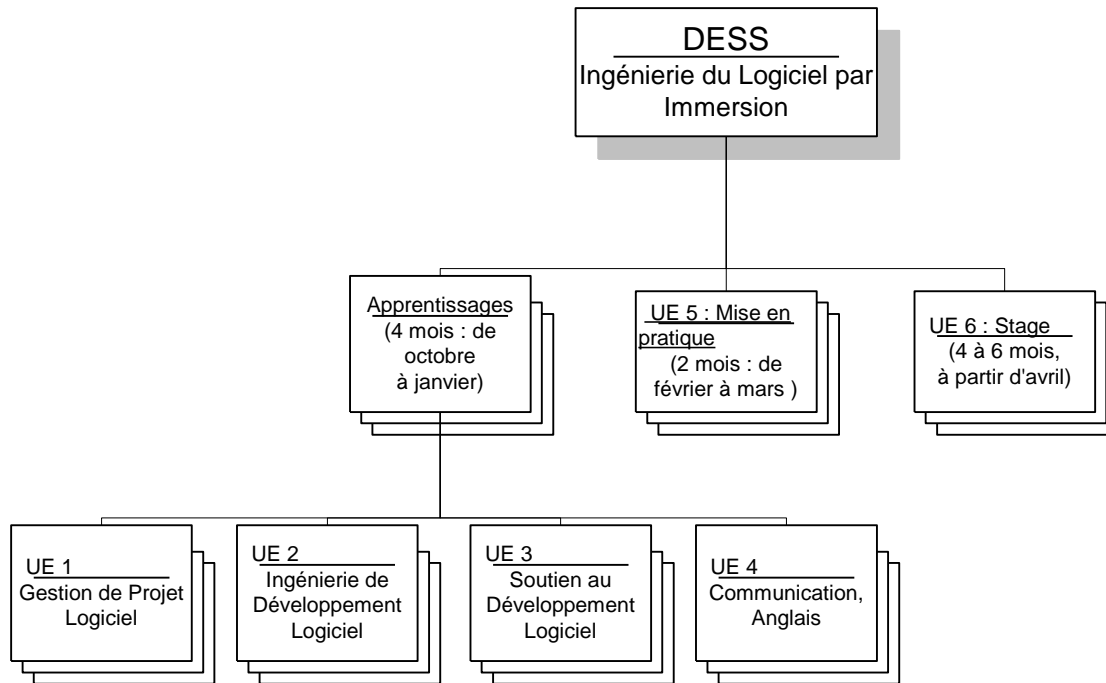


Figure 1 : Structuration de l'année

Les étudiant(e)s sont immergés dans un véritable projet qui se déroule exactement comme en entreprise, avec les spécificités suivantes :

- le projet se déroule au sein de l'université, dans une salle dédiée;
- le chef de projet qui pilote et tutore le projet est un enseignant-chercheur de l'université, appelé tuteur de compagnie;
- la finalité, à savoir le produit logiciel, n'est qu'un prétexte à l'apprentissage des activités d'ingénierie logiciel.

Les étudiant(e)s sont répartis en équipes projet indépendantes, qu'on appelle des compagnies, et ils/elles doivent jouer plusieurs rôles (organisationnel, qualité, technique, soutien, etc.) dans le contexte de collaboration et de coopération lié au travail en équipe. Les compagnies fabriquent un logiciel à partir de la même commande (le cahier des charges) mais avec des méthodes, des outils et des technologies différentes. L'architecture technique revêt une importance capitale dans la construction des systèmes informatiques actuels. Chaque compagnie dispose donc d'une plate-forme professionnelle (IBM, Oracle, Rational) relativement complexe, que les étudiant(e)s ne maîtrisent que partiellement en début de projet.

Pour simuler l'environnement standard des entreprises de développement logiciel, chaque étudiant(e) dispose d'un poste de travail personnel dans une salle spécialement réaménagée comprenant :

- un bureau « paysagé » par compagnie,
- une salle de réunion commune et un espace « Internet ».

2.2. Déroulement de l'année

L'année se déroule en 3 périodes, que l'on appelle itérations (cf. figure 2) :

- Une première itération d'apprentissage tutoré permet l'acquisition des connaissances d'ingénierie logiciel.
- Une deuxième itération de mise en pratique accompagnée a pour objet de les transformer en compétences.
- Et enfin une troisième itération permet leur mise en application lors du stage en entreprise.

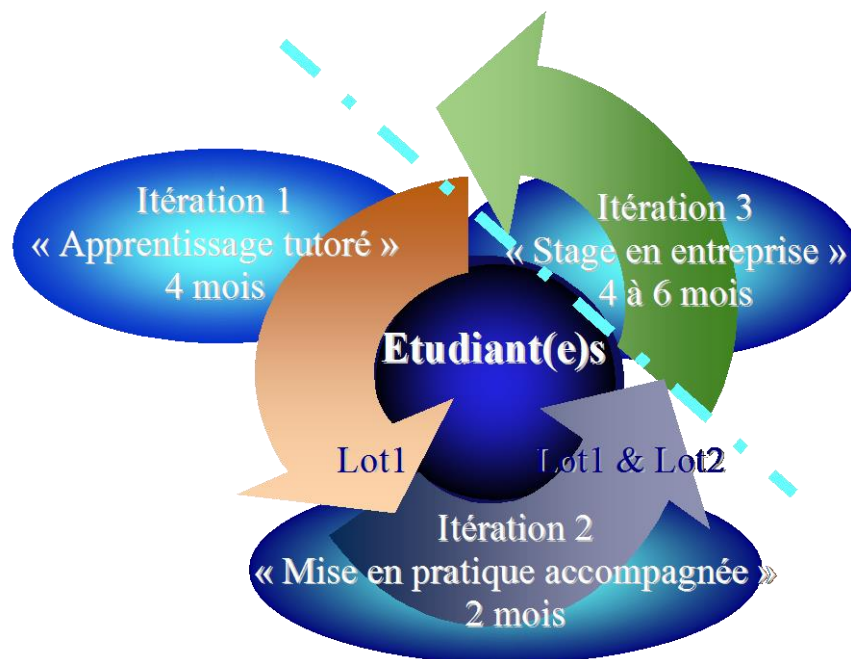


Figure 2 : 3 itérations d'apprentissage

Lors de la première itération d'apprentissage tutoré (4 mois), une version incomplète du logiciel est réalisée dans un cadre totalement piloté et tutoré par le tuteur de la compagnie. Toutes les activités d'ingénierie sont alors mises en œuvre dans un cycle de développement complet.

Lors de la deuxième itération de mise en pratique accompagnée (2 mois), chaque compagnie est rendue relativement autonome afin de mettre en pratique les connaissances, compétences et savoir-faire acquis lors de la première itération. Le produit logiciel attendu à l'issue de cette itération correspond à une version complète du logiciel tel que demandé dans le cahier des charges. Lors de cette itération l'équipe reste toujours accompagnée et encadrée par le tuteur de compagnie.

Enfin la troisième itération, le stage en entreprise, s'inscrit dans la continuité de la formation. L'étudiant(e) se voit alors confier au sein de l'entreprise, la réalisation d'un petit projet de « A à Z » selon le processus de développement acquis lors de sa formation. Une autre alternative consiste à obtenir un stage permettant d'approfondir une (voire plusieurs) des activités essentielles d'ingénierie logiciel : mise en place d'une gestion de configuration du logiciel, capture des besoins, modélisation en analyse et/ou conception, etc..

2.3. Dispositif d'apprentissage

L'apprentissage par immersion reproduit l'environnement réel d'un projet logiciel et fournit un dispositif d'apprentissage des compétences visées.

L'environnement d'apprentissage se compose d'au moins quatre éléments essentiels :

- Un ensemble d'activités (ou de tâches) liées au métier d'ingénieur logiciel.
- Un processus de conception et de fabrication logiciel qui organise l'ensemble des activités permettant de transformer en produit logiciel les besoins exprimés par les utilisateurs.
- Un référentiel métier qui définit les pratiques et capitalise le savoir-faire du métier d'ingénieur logiciel [TIS02].
- Une plate-forme de travail comprenant des infrastructures communes et des suites d'outils destinées à la conception, à la fabrication et à la documentation des produits logiciels.

Le dispositif d'apprentissage se matérialise par un ensemble de fiches d'apprentissage. Chaque fiche d'apprentissage décrit le travail à réaliser, les produits attendus et fournit les ressources pédagogiques liées à l'activité demandée :

- Guides et procédures du référentiel métier ISO9001 mis à disposition par la filiale informatique du groupe Thales [TIS02].
- Exemples réels.
- Supports d'utilisation des plates-formes techniques.
- Ouvrages et supports d'autoformation.

2.4. Processus d'apprentissage

Le processus d'apprentissage est guidé par le processus de développement logiciel retenu, le 2TUP (2 Track Unified Process) [RV02], qui fait partie de la famille des processus unifiés, utilisant la notation UML [JBR99].

- Il se matérialise sous la forme d'un enchaînement d'étapes avec des objectifs clairs et contrôlés à chaque étape.
- A chaque étape correspond un certain nombre de travaux décrits par des fiches d'apprentissage définissant précisément le travail à réaliser, les apprentissages visés et/ou les compétences à mobiliser.
- Chaque fiche d'apprentissage est affectée à un ou plusieurs étudiant(e)s qui doivent alors assumer le métier (ou rôle) inhérent à l'activité concernée.
- Chaque étape met à disposition des fournitures pédagogiques que les étudiant(e)s doivent s'approprier.
- Pour chaque fiche d'apprentissage, les étudiant(e)s peuvent s'appuyer au quotidien sur l'assistance de tuteurs.
- Chaque fiche d'apprentissage décrit la forme et le contenu des produits attendus. Elle constitue le support principal de l'évaluation.

On peut représenter ce processus dans un tableau à deux dimensions (cf. figure 3).

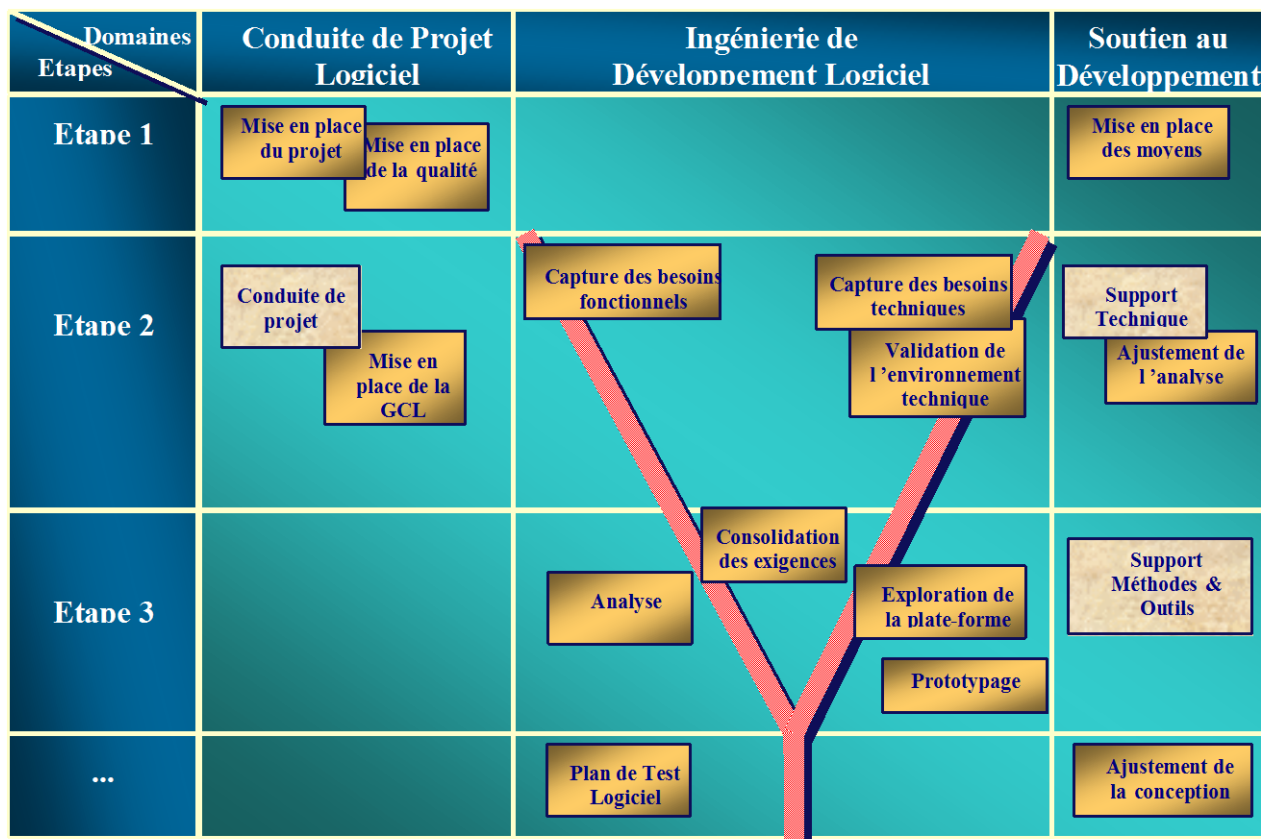


Figure 3 : Vue partielle du processus d'apprentissage

Les 3 colonnes (que l'on peut assimiler à des couloirs) représentent les 3 domaines principaux d'activités couverts par la formation. Ces 3 couloirs peuvent également être perçus comme 3 sous-processus complémentaires et fortement imbriqués. L'axe vertical de ce processus représente le temps. Il permet de matérialiser l'enchaînement des étapes et jalons essentiels à l'apprentissage de l'ingénierie logiciel. A chaque étape correspond des objectifs clairs et contrôlés. La durée d'une étape peut varier de 2 à 4 (voire 5) semaines en fonction des travaux mis en œuvre dans l'étape.

Le parcours de formation comprend deux cycles (appelés itérations) du processus.

Lors de la 1^{ère} itération, à chaque changement d'étape, l'étudiant(e) se voit confier de nouveaux travaux qui correspondent à des changements sur plusieurs axes :

- Le temps qui permet d'aborder une nouvelle activité d'ingénierie logiciel qui n'a de sens qu'à ce moment car s'appuyant sur les activités et travaux réalisés antérieurement.
- Les apprentissages que nécessite la nouvelle activité et qui ne sont pas forcément dans la continuité de ses activités précédentes.
- Le domaine d'activité de rattachement change également : un(e) étudiant(e) peut passer de la conduite de projet à une action de soutien au développement, etc.
- La spécificité « métier » de la nouvelle tâche : l'étudiant(e) change de rôle, de chef de projet à responsable de la gestion de configuration du logiciel, d'analyste à ingénieur système, ...

Lors de la 2^{ème} itération, une organisation fixe du travail est mise en place. Un(e) étudiant(e) assume la fonction de chef de projet, les autres assurant les autres métiers inhérents à la conception et à la fabrication d'un logiciel : analyste, concepteur, développeur, etc.

3. Transformation des métiers

3.1. Du métier d'enseignant aux rôles d'enseignant-tuteur

3.1.1. Avant le début des opérations

Pour que la formation puisse se dérouler dans de bonnes conditions, un travail de préparation est nécessaire. Ce travail est du ressort du « **Tuteur-Coordinateur** » du dispositif d'apprentissage que l'on retrouve toute l'année. Ce dernier :

- **définit le nouveau projet** dans lequel les étudiant(e)s seront immergés. Cela consiste essentiellement à rédiger un cahier des charges et une proposition technique et commerciale.
- **prépare la logistique** nécessaire au bon fonctionnement des opérations. Il s'agit d'inventorier et faire le tri dans les ressources de chacune des compagnies, effectuer une remise à zéro des plates-formes de développement, anticiper les nouveaux besoins en matériel et logiciel, suivre les contrats avec les éditeurs de logiciels impliqués dans la formation, approvisionner et réceptionner les nouveaux matériels, logiciels, ouvrages, supports de formation, etc.
- **consolide et adapte le référentiel d'apprentissage** pour en faciliter son usage par les enseignants-tuteurs et les étudiant(e)s. Il peut s'agir de la prise en compte de nouveaux ouvrages, de nouveaux supports de formation comme de l'enrichissement ou de la mise à jour du référentiel métier de Thales-IS, des fiches d'apprentissage, etc.

3.1.2. Première itération

Le bon fonctionnement de la 1^{ère} itération repose en partie sur le « Tuteur-Coordinateur » pour ce qui est du dispositif d'apprentissage, mais surtout sur le tuteur de compagnie qui assure la double fonction :

- « Tuteur-Chef de Projet »
- « Tuteur d'Apprentissage »

Ce dernier peut également faire intervenir des tuteurs spécialisés comme « Tuteur de Travaux » spécifique à une activité : rédaction des cas d'utilisation, développement EJB, tenue de réunion, etc.

Le rôle du « **Tuteur-Coordinateur** » s'inscrit dans la continuité de ses activités amonts (cf. paragraphe 3.1.1). Il est le garant du référentiel d'apprentissage et l'interlocuteur privilégié des tuteurs de compagnie. Il gère l'accès aux ressources, il partage les moyens, il ajuste le référentiel à la réalité terrain des compagnies, il anticipe l'année suivante, etc. Une partie de ce travail se reporte sur la 2^{ème} itération mais de manière moindre.

Le « **Tuteur-Chef de Projet** » est l'authentique chef de projet de la compagnie : il est chargé d'animer l'équipe projet étudiant(e)s, il est responsable au quotidien de l'avancement des travaux, il est l'interlocuteur privilégié du « Tuteur-Coordinateur », etc.

L'intervention du « Tuteur-Chef de Projet » commence dès la première étape d'apprentissage. Contrairement à la réalité d'un projet en entreprise, il va déléguer à son équipe d'étudiant(e)s l'initialisation des actions d'organisation et de conduite de projet qui se dérouleront pendant toute la durée du projet. A cette fin, la rédaction du Plan de Développement Logiciel est confiée à 3 étudiant(e)s qui doivent définir l'organisation des travaux à réaliser, les échéanciers et jalons de chacune des activités du projet, les ressources humaines et matériels, les méthodes et techniques, les outils, etc.

Une fois le projet mis en route, c'est néanmoins le « Tuteur-Chef de Projet » qui coordonne le travail des différents étudiant(e)s, adapte le planning en fonction de l'avancement.

Au quotidien (au minimum 1 heure par jour), il est l'interlocuteur principal des étudiante(s) mais aussi de toutes personnes impliquées dans le dispositif : le client, le responsable système & réseau de l'université, le « Tuteur-Coordinateur », etc.

C'est également lui qui prépare et pilote la réunion d'avancement hebdomadaire (au moins 2 heures) qui est un moment privilégié d'échange entre tous membres de l'équipe. Il confie à tour de rôle aux étudiant(e)s la rédaction du compte-rendu d'avancement. Chaque compte-rendu donne lieu ensuite à une évaluation formative (puis sommative) suivie d'un débriefing individualisé.

En plus de solides connaissances techniques et d'une expérience significative dans les différents métiers couverts par la formation, le rôle de « Tuteur-Chef de Projet » demande de la rigueur, de la méthode et une excellente organisation. Pour le jeu de rôle, il doit avoir d'excellentes qualités relationnelles, un sens de la communication (voire du théâtre) développé, la capacité à encadrer et à motiver une équipe d'étudiant(e)s, ainsi qu'une capacité certaine à l'improvisation, etc.

Il ne faut pas perdre de vue que l'objectif principal de l'itération 1 est l'apprentissage : le « Tuteur-Chef de Projet » est avant tout un pédagogue, soucieux de transmettre ses compétences aux membres de son équipe.

Le « **Tuteur d'apprentissage** » coordonne et régule les apprentissages collectifs et individuels, tutore et assiste les étudiant(e)s dans leurs apprentissages, et enfin évalue les fournitures produites. Il s'appuie pour cela sur le processus d'apprentissage et sur les fiches d'apprentissage du référentiel.

En dehors des tâches de préparation des apprentissages, de planification et d'affectation, le « Tuteur d'apprentissage » est au quotidien l'interlocuteur de tous les membres de l'équipe pour faciliter leurs apprentissages et les assister à la réalisation de leurs tâches.

A l'issue d'un apprentissage, il évalue le(s) document(s) et produit(s) réalisés : il annote, corrige, propose, réoriente. Il rédige une fiche d'évaluation qui comprend un bilan général de l'évaluation ainsi que la liste exhaustive des points à améliorer ou à refaire. Même si le travail évalué a été réalisé seul ou en binôme, le « Tuteur d'apprentissage » effectue un débriefing en présence de toute l'équipe. Cela permet d'approfondir, de discuter, de remettre en question collégalement, les remarques apportées par le tuteur. A l'issue de ce débriefing, les étudiants concernés doivent mettre à jour et même parfois refondre entièrement leur production. Il s'en suivra une 2^{ème} évaluation pour vérifier la prise en compte des remarques initiales.

En général deux évaluations sont suffisantes pour obtenir un résultat satisfaisant en terme d'apprentissage (même si dans la réalité de l'entreprise, il resterait encore beaucoup à dire). Si malgré tout le résultat n'est pas bon, on n'insiste pas toujours pour ne pas compromettre les échéances suivantes (... nous ne sommes qu'à la 1^{ère} itération ... après maturation, le même document produit dans la 2^{ème} itération sera bien meilleur ...).

Une difficulté récurrente est un premier résultat d'évaluation si peu satisfaisant qu'il n'est pas possible de le rendre en l'état, sous peine de casser la dynamique de groupe. Le « Tuteur d'apprentissage » peut solliciter le « Tuteur-Coordinateur » ou le tuteur d'une autre compagnie pour avis. Si les opinions convergent, il se doit de refaire une passe complète sur le document en modérant ces remarques.

3.1.3. Deuxième itération

Contrairement à ce qui avait été imaginé dans un 1^{er} temps, le travail du tuteur de compagnie est tout aussi important dans la 2^{ème} itération que ce soit en termes de travail, d'investissement personnel ou que ce soit vis-à-vis du dispositif d'apprentissage.

Il assure une à une triple fonction :

- « Tuteur-Modérateur »
- « Tuteur-Facilitateur »
- « Tuteur-Responsable d'Affaires »

Même si dans la réalité ce n'est pas aussi cloisonné, on peut considérer qu'à chacun de ces rôles correspond une responsabilité principale, essentielle au bon déroulement du projet sur le théâtre des opérations ILI.

Le « **Tuteur-Modérateur** » peut accompagner de manière individualisée chaque étudiant(e).

Si l'on reprend la terminologie du référentiel TEMPO de Thales-IS : « *Le modérateur est un expert extérieur à l'affaire dont l'expérience est en adéquation avec le projet courant pour apporter un regard impartial sur les aspects fonctionnels et techniques du projet. Il intervient essentiellement comme un pair sur lequel le chef de projet ou les membres de l'équipe peuvent se reposer pour conforter tel ou tel choix.* »

Il s'agit essentiellement d'une activité de consolidation qui porte sur les documents et fournitures produits à l'issue de chaque fiche de travail. Cela peut être assimilé à une activité de « revue de pair » du point de vue du projet mais c'est aussi une activité « d'évaluation formative » et dans un second temps « d'évaluation sommative » du point de vue de la formation. La consolidation est primordiale sur les documents ou produits dont la qualité moyenne (voire médiocre) pourrait poser problème pour le bon déroulement des étapes suivantes du projet.

Le « **Tuteur-Facilitateur** » s'occupe de la logistique (mise à disposition de nouvelles ressources matérielles ou logiciel, préparation des environnements de réception, etc.) mais surtout assiste et soutient l'étudiant(e) qui a la lourde responsabilité de chef de projet. Cela se matérialise par une rencontre journalière plus ou moins longue (de 10 minutes à 1 heure) où tous les aspects du projet peuvent être abordés. C'est aussi une assistance à la préparation des réunions hebdomadaires d'avancement interne, à la gestion des ressources, à la rédaction des fiches de travail, à la planification des tâches, etc. Il s'agit également de soutenir le chef de projet lors des réunions d'avancement lorsque celui-ci est en difficulté soit vis-à-vis d'une impasse technique ou fonctionnelle soit vis-à-vis d'une situation conflictuelle qu'il faut désamorcer le plus rapidement possible.

Le « **Tuteur-Responsable d'Affaires** » arbitre, régule (voire impose) en cas de conflit ou de blocage de la production.

Si l'on tient à la définition ci-dessus, le « Tuteur-Modérateur » n'a pas de pouvoir de décision sur l'équipe projet. Or il s'avère que pour la bonne marche de cette 2^{ème} itération et notamment pour résoudre les situations de blocage, le tuteur de compagnie se doit d'avoir un pouvoir hiérarchique sur toute l'équipe projet. Il a une partie de ce pouvoir en tant qu'enseignant mais celui-ci ne porte que sur le dispositif de formation. Cette autorité n'est cependant pas suffisante car elle ne s'applique pas dans le jeu de rôle qui est l'imitation d'un projet réel, ce qui confère à chaque participant des prérogatives spécifiques.

Cette autorité hiérarchique nécessaire, se concrétise par le rôle de responsable d'affaires attribué au tuteur de compagnie. Cela lui permet non seulement d'assurer de veiller au bon respect des coûts et délais mais aussi de suppléer « hiérarchiquement » au chef de projet étudiant(e) sur tel ou tel aspect fonctionnel, technique ou organisationnel.

Les situations où cette autorité hiérarchique est nécessaire sont multiples :

- Certains membres de l'équipe ne prennent pas en compte les demandes d'évolution ou de modification émanant du chef de projet étudiant(e).
- Il n'est pas toujours aisé de résoudre un problème en équipe surtout lorsqu'il y a plusieurs solutions.
- Certains n'hésitent pas à verrouiller les connaissances pour se rendre indispensable, faisant ainsi passer leurs intérêts personnels avant les intérêts collectifs de l'équipe.
- Les décisions du chef de projet étudiant(e) sont parfois contestées et remises en cause. Très souvent, elles ne sont hélas justifiables que par l'expérience du chef de projet qui dans notre cas est quasiment nulle.
- Etc.

3.2. Etudiant(e) dans ce parcours de formation : de nouvelles pratiques ?

Ce dispositif de recherche-action peut permettre une étude sur deux directions, en lien avec les observations effectuées.

Soit elle s'oriente sur les processus de l'apprentissage et la maîtrise des différentes fonctions liées au métier d'ingénieur-logiciel. Il peut s'agir de repérer les enjeux cognitifs de ces processus pédagogiques par rapport à celle d'une pédagogie expositive ou démonstrative avec, par exemple, une présentation conceptualisée des phases du développement d'un logiciel avec application et étude critique.

A ce stade, nous faisons plutôt le choix de privilégier le **repérage des transformations et des enjeux identitaires** liés au fonctionnement du groupe d'apprenant(e)s (travail en équipe) en termes de professionnalisation et de passage du métier d'étudiant(e) à celui d'ingénieur en situation de quasi-production et de réponse aux différentes dimensions des demandes des partenaires (chefs de projets, clients...). Une attention particulière est portée aux formes de coopération entre étudiant(e)s occupant des fonctions spécifiques dans les deux itérations d'apprentissage tutorées par les enseignants-chercheurs. Les étudiant(e)s dans ce parcours sont amenés à pratiquer, dans le cadre universitaire, les différentes facettes de l'activité du métier d'ingénieur-logiciel. Dans ce contexte, les capacités validées, au-delà des expertises en termes de savoirs conceptuels renvoient à des savoirs procéduraux, il s'agit d'apprendre à exercer la partie du métier qui a été confiée, en tenant compte du regard des autres partenaires de la production. Ce processus d'apprentissage permet une appropriation de l'univers des métiers par la réalisation d'un projet commun mené à son terme, par la fourniture d'un produit. Cela s'appuie sur la participation coordonnée aux diverses phases d'un processus dans un temps contraint

Cette perspective, que mobilise l'ensemble des étudiant(e)s dans les récits qu'ils font de leur parcours au cours de la seconde itération, est attestée par l'ensemble des acteurs. Ils l'associent à la présence des aspects suivants dans le DESS :

- **La capacité à passer d'une coopération communautaire à une coopération complémentaire** dans le travail en équipe de conception. Alors qu'au début de la première itération, les membres des compagnies insistent sur leur homogénéité, leur ressemblance, et que cela génère de la coopération par l'identification mutuelle, cette coopération se modifie au cours du déroulement de la formation. Sans que cette coopération « communautaire » ne disparaisse, elle devient compatible ou laisse place à une autre forme de coopération « complémentaire » dans laquelle l'action coopérative passe par la négociation à partir de réseaux de confiance qui prennent appui sur les intérêts individuels, les spécificités, la division du travail et les engagements internes [Dam02]. Plus qu'une socialisation collective, source d'identité sociale, qui est compatible avec un mode de management peu délégitif, c'est la coopération complémentaire qui est attendue et validée, en particulier lors de la deuxième itération. Au delà des attendus et des normes des entreprises avec la valorisation de la performance et de la mobilisation individuelle, cette coopération complémentaire est source de performance et de pérennité du collectif. C'est ce mode d'engagement en tant qu'étudiant avancé qui est attendu et valorisé par les enseignants-tuteurs, car il leur semble garantir d'une optimale capacité d'adaptation professionnelle à court mais surtout à moyen terme.
- **La mise en place d'une compétence transférable dans les organisations du futur travail** : démarche qui consiste à apprendre à apprendre. Le parcours de formation met l'accent, non sur la réalisation de tâches cloisonnées, identifiées et maîtrisées pour elles-mêmes mais sur un ensemble d'activités connectées et visant à une certaine cohérence entre elles. Les capacités de clarification des problèmes, des attentes des utilisateurs finaux comme de ses partenaires, conduisent à expérimenter l'interdépendance des acteurs en temps réel. Ils obligent aussi à se familiariser avec une certaine incertitude avec laquelle il leur faut composer, incertitude qui conduit à ne jamais reproduire une application automatique d'un modèle, d'un manuel ou d'un guide de procédures mais à en interroger la pertinence et les usages au fil de l'eau.
- **Une dynamique relationnelle couplée à un processus d'acquisition de compétences** et garante d'une capacité de socialisation et d'intégration dans le travail. Les étudiant(e)s en formation sur ce métier intègrent progressivement, de manière presque prototypique, les dimensions principales de leurs futures activités. Il s'identifient aux acteurs professionnels, qualité qu'ils reconnaissent aux tuteurs. Cette identification ouvre une familiarisation possible au métier et les conduit à se détacher progressivement d'un statut d'étudiant. Ils entretiennent avec ce statut une forte familiarité et parfois une certaine ambivalence, car l'exercice de ce rôle d'étudiant est plus neutre que celui de futur ingénieur du logiciel. Le rôle d'étudiant peut être exercée de manière plus ou moins active tandis que le métier mobilise une activité conforme de travailleur engagé et créatif.

Ainsi, la formation est construite autour d'étapes d'apprentissage et de réalisation, mobilisées dans les deux itérations. Elle permet :

- de prendre en compte le rôle structurant du temps et son intensité dans le monde du travail. L'apprentissage de la capacité à hiérarchiser les séquences, à maîtriser un processus global de travail sera mobilisée au-delà de cette activité précise.
- d'établir des relations coopératives entre apprenants ce qui conduit à alterner entre des temps individualisés de travail en équipe et des temps de véritable travail d'équipe.

Au terme de la formation les étudiant(e)s ont pris la mesure de la complexité du métier, complexité sur laquelle ils peuvent raisonner. Ils ont développé leur capacité à clarifier des choix, à construire des alternatives, à mobiliser des ressources spécialisées et parfois individualisées. Ils s'estiment apte à s'engager dans un travail de mise en cohérence de moyens humains et techniques en faveur de la faisabilité et de la cohésion du produit et d'un service fourni.

4. Perspectives et conclusion

Le dispositif est reconduit pour l'année prochaine (2003-2004) avec un grand chantier : l'évaluation individualisée et collective, à la fois de l'apprentissage, des compétences et de la performance.

Le conseil d'administration de l'université de Brest vient d'approuver le cadre général du futur contrôle de connaissances. Il s'agit pour nous maintenant de le mettre en application.

A l'issue de l'année prochaine, nous serons en mesure de proposer ce dispositif expérimental comme un parcours pour la dernière année du Master Professionnel en Informatique, lors du passage au système Licence-Master-Doctorat prévu à la rentrée universitaire 2004-2005.

Arrivée en 5ème année, il devient inutile de surcharger les connaissances acquises lors de la formation initiale de savoirs disciplinaires et didactiques, si l'on ne prend pas le temps d'apprendre à les intégrer, à les mobiliser dans l'action.

L'important est de viser des compétences et de travailler les savoirs comme des ressources, non comme des fins en soi. Il ne s'agit pas de tourner le dos aux savoirs disciplinaires ou méthodologiques, mais de les mettre au service de l'action. Cette approche appliquée à l'ingénierie logiciel a pour conséquence de remanier sensiblement les dispositifs pédagogiques.

L'enseignement pratiqué dans la formation ILI s'est appuyé sur les principes suivants :

- Se centrer sur les compétences à développer et les situations d'apprentissage les plus fécondes, plutôt que de se contenter d'enseigner des savoirs discipline par discipline.
- Développer une pédagogie active et coopérative fondée sur le projet et le jeu de rôle : immersion des étudiant(e)s dans un projet de 6 mois simulant au mieux un projet en entreprise.
- Travailler en équipe, raconter ce qu'on fait, coopérer avec ses collègues.
- Elaborer un référentiel d'apprentissage par immersion, fondations du dispositif d'apprentissage présenté dans cet article.

Il y a eu un grand enthousiasme des étudiant(e)s et des deux tuteurs lors de cette année passée. Cette dynamique a permis de résoudre la plupart des problèmes rencontrés. Les étudiant(e)s sont très satisfait(e)s de la formation et commencent une publicité de bouche à oreille qui est un gage de succès.

Le point faible reste l'évaluation individuelle des apprentissages, des compétences comme des performances. Un des enjeux de la prochaine année académique est de mettre au point des outils de suivi et de mesure individualisés.

5. Références

- [CC2001] Computing Curricula, Computer Science Volume, IEEE and ACM, 2001
- [Dam02] Dameron S., 2002, La dynamique relationnelle au sein d'équipe de conception , *Le travail humain*, Vol. 65, n°4, Oct-Déc, pp 339-361
- [JBR99] Ivar Jacobson, Grady Booch, James Rumbaugh, *The Unified Software Development Process*, Addison-Wesley Longman, 1999
- [Mey01] Bertrand Meyer, *Software Engineering in the Academy*, IEEE Computer, May 2001
- [PGL00] Gilda Pour, Martin L. Griss, Michael Lutz, *The Push to Make Software Engineering Respectable*, IEEE Computer, May 2000
- [RV02] Pascal Roques, Franck Vallée, *UML en action*, Eyrolles, 2002
- [Tar98] Jacques Tardif, *Intégrer les nouvelles technologies de l'information – Quel cadre pédagogique ?*, ESF, 1998
- [TIS02] TEMPO, *la maîtrise du développement de systèmes informatiques*, Thales Information System, 2002