

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOUD MAMMARI DE TIZI-OUZOU



FACULTE DU GENIE ELECTRIQUE ET D'INFORMATIQUE  
DEPARTEMENT D'INFORMATIQUE

## Mémoire de Fin d'Etudes de MASTER PROFESSIONNEL

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : **Ingénierie des systèmes d'information**

*Présenté par*

**Imad DJOUAL  
Brahim HOCINE**

### Thème

**Guidage et Etayage dans l'Approche par  
Compétences : Application à  
l'Algorithmique**

*Mémoire soutenu publiquement le 20 / 09 / 2016 devant le jury composé de :*

**Président : Me Farida BOUARAB-DAHMANI**

**Encadreur : Me Tassadiit BERKANE**

**Examinatrice : Me Yasmine YESLI**

**Examineur : Mr Mohand-Said HABET**

**Promotion : 2016/2015**

## *Remerciements*

Nous remercions tout d'abord le bon Dieu de nous avoir donné la force, la volonté, et le courage pour réaliser ce travail.

Nous tenons à remercier vivement notre promotrice Me **BERKANE TASSADIT** pour son orientation et sa disponibilité constante tout au long de notre travail.

Nos remerciements vont également aux membres du jury qui ont accepté d'évaluer notre travail.

Enfin nous remercions tous ceux qui ont contribué de près ou de loin à la réalisation de ce projet.

## *Dédicaces*

Je dédie ce travail à :

A mes chers parents qui m'ont aidé et soutenu tout au long de mon parcours scolaire et universitaire.

A mes frères , mes sœurs et ma femme

A toute la famille « **djoual** »

A tous mes amis surtout «**abdelbaki, aissa, salim, mourad, riad** » ainsi mes collègues de travail..

A mon binôme «**hocine**» et sa famille.

**Imad**

## *Dédicaces*

Je dédie ce travail à :

A toute ma famille, en premier lieu ma chère LHADJA maman.  
A mon petit poussin ISMAIL.

A tous mes amis et mes collègues de travail.

A mon binôme «**imad**» et sa famille.

**Brahim**

# Sommaire

<b>Introduction générale</b> .....	01
<b>Chapitre 1 : Les fondements de l'approche Par Compétences</b>	
Introduction.....	03
1. Pédagogie Par Objectif.....	03
1.1 Notion d'Objectif Pédagogique .....	03
1.1.1 Définition de l'objectif pédagogique.....	04
1.1.2 Importance des objectifs pédagogiques .....	04
1.1.3. Classification des objectifs pédagogiques .....	05
1.1.4 Des objectifs pédagogiques à la pédagogie par objectifs.....	05
1.2. Limites de la Pédagogie Par Objectifs.....	06
2. Approche Par Compétences .....	07
2.1 Définition de la Compétence.....	07
2.2 Types de compétences.....	08
2.2.1 Les compétences disciplinaires .....	08
2.2.2 les compétences transversales .....	09
2.3 Notion de situation-Problèmes.....	10
2.4 Notion d'étayage.....	11
2.4.1 Fonctions d'étayage .....	12
2.4.2 Les Types d'étayage .....	13
2.5 Comparaison entre la PPO et l'APC.....	14
Conclusion .....	14
<b>Chapitre 2 : Présentation de notre Application</b>	
Introduction.....	15
1.Guidage dans l'Acquisition de Compétences Disciplinaires.....	15
2.Etayage dans l'Acquisition de Compétences disciplinaires.....	23
Conclusion.....	23
<b>Chapitre 3 : Conception de l'application guidage et étayage</b>	
Introduction.....	24
1 Présentation du langage UML.....	24
2 Etape d'Analyse.....	25
2.1 Identification des besoins.....	25
2.2 Identification des Acteurs .....	26
2.3 Diagramme de contexte.....	27
2.4 Identification des cas d'utilisation.....	27
2.5 Description textuelle des cas d'utilisation.....	30
3.Etape de Conception.....	34
3.1 Diagrammes de séquences.....	34
3.2 Diagramme des activités.....	36

3.3 Diagramme de classes .....	38
3.4 Modèle logique des données .....	38
3.5 Codification des données.....	39
Conclusion.....	40

### **Chapitre 4 implémentation de l'application**

Introduction.....	41
1 Environnement Technique de développement et d'implémentation.....	41
1.1 Architecture logicielle de notre système.....	41
1.2 Architecture logicielle .....	42
2 Outils et Langages utilisés.....	43
2.1 Outils de développement .....	43
3. Langages utilisés.....	44
4 .Scenario d'utilisation du système proposé.....	44
4.1 Scénario de l'enseignant .....	44
4.2. Scénario de l'apprenant.....	45
5. Quelques interfaces du système.....	46
Conclusion.....	51
<b>Conclusion générale</b> .....	<b>52</b>
Bibliographie .....	53

## Listes des figures :

Figure 2.1 : Relation de prérequis entre compétences.....	15
Figure 2.2 : Relation de composition entre compétences.....	16
Figure 2.3 : Association compétence-situations problèmes.....	16
Figure 3.1 diagramme de démarche .....	25
Figure 3.2 formalisme des acteurs .....	26
Figure 3.3 diagramme de contexte .....	27
Figure 3.4 diagramme de cas d'utilisation « démarche d'acquisition de compétences disciplinaires »	29
Figure 3.5 diagramme de cas d'utilisation « administrateur » .....	30
Figure 3.6 diagramme de cas d'utilisation « visiteur » .....	30
Figure 3.8 diagramme de séquence « cas d'inscription d'un apprenant ».....	35
Figure 3.9 diagramme de séquence « cas d'authentification d'un apprenant ».....	35
Figure 3.10 diagramme de séquence « traiter une compétence » .....	36
Figure 3.11 diagramme des activités .....	37
Figure 3.12 diagramme de classe.....	38
Figure 4.1 architecture a 3 niveau .....	42
Figure 4.2 architecture logicielle de notre système.....	42
Figure 4.3: Interface de PhpMyAdmin les tables de notre base de données.....	43
Figure 4.4 : Scénario d'utilisation de l'enseignant.....	45
Figure 4.5 : Scénario d'utilisation de l'apprenant.....	46
Figure 4.6 : page d'accueil.....	47
Figure 4.7 : page d'inscription.....	48
Figure 4.8 : espace d'apprentissage.....	48
Figure 4.9 : espace de travail.....	49
Figure 4.10 : espace enseignant.....	50
Figure 4.11 : espace administrateur.....	51

## Liste des Tableaux

Tableau 1.1 : comparaison entre la PPO et L'APC.....	14
Tableau 3.1 les cas d'utilisations .....	28
Tableau 3.2 description textuelle de cas d'utilisations « inscription ».....	31
Tableau 3.3 description textuelle de cas d'utilisations « authentification » .....	31
Tableau 3.4 description textuelle de cas d'utilisations « modifier profil ».....	32
Tableau 3.5 description textuelle de cas d'utilisations « traiter une CP ».....	33
Tableau 3.6 description textuelle de cas d'utilisations « gestion de SPs et CPs ».....	33
Tableau 3.7 description textuelle de cas d'utilisations « gestion des profils ».....	33
Tableau 3.8 description textuelle de cas d'utilisations « évaluer SP ou CP».....	34
Tableau 3.9 codification des données.....	34

# Introduction Générale

## Introduction Générale

Depuis plus d'une décennie, nous assistons à un bouleversement radical du système éducatif au niveau mondial. Du primaire au supérieur les réformes subies ont pour but d'enseigner et d'apprendre suivant une logique de développement de compétences.

En effet pendant longtemps, le système éducatif a connu le règne de la pédagogie par objectifs. Pour les tenants de cette approche, les programmes d'études étaient fondés sur des savoirs disciplinaires à faire acquérir par les élèves. Le contenu des matières à enseigner primait sur tout sans se soucier si cela faisait sens pour les élèves. Or, il a été constaté que beaucoup de jeunes décrochaient en raison du caractère abstrait et décontextualisé de ce qu'ils apprenaient. L'approche par compétences est apparue alors comme une alternative pouvant ramener la vie à l'école. Dans cette approche, l'enseignant cherche à créer des situations pour que les élèves apprennent de manière active et soient capables d'utiliser des ressources intellectuelles et matérielles en vue de résoudre des problèmes. Pour ce faire, les programmes d'études sont élaborés selon une logique des compétences. Dans ces programmes, on exige non seulement d'acquérir des savoirs dans les disciplines enseignées mais surtout de développer des compétences notamment disciplinaires chez les élèves. Rappelons que dans sa forme la plus simple et la plus connue, les élèves apprenants doivent être présents physiquement dans un lieu commun (les salles de classe) sous la supervision d'un enseignant. Dans cet enseignement/apprentissage dit classique, l'enseignant détient le savoir qu'il transmet à des élèves passifs et à qui on demande d'appliquer des savoirs dans des exercices d'application. Dans la logique d'apprentissage par compétences, l'élève doit non seulement acquérir des savoirs mais surtout de montrer ses compétences dans la discipline considérée. L'élève est actif, il construit de nouveaux savoirs avec l'aide de son enseignant et de ses camarades dans des situations problèmes qui lui sont proposées. L'enseignant devient un accompagnateur, un orienteur durant l'apprentissage. On parle dans ce cas de l'étayage par l'enseignant. Les camarades deviennent aussi des collaborateurs avec qui il peut partager ses expériences, ses solutions aux problèmes posées par les situations.

Notre travail s'insère dans le cadre des travaux de recherche initiés dans [Berkane, 2015]. Ces travaux rentrent dans le cadre du développement de compétences disciplinaires avec un guidage et un étayage. Notre contribution consiste à expérimenter ces travaux avec des disciplines informatiques notamment l'algorithmique et les bases de données relationnelles. L'application dans ces disciplines consiste à guider et à étayer l'apprenant durant son apprentissage. Le guidage consiste à ordonnancer les compétences à développer selon un ordre de maîtrise imposé. L'étayage est un processus qui consiste à accompagner l'élève apprenant durant son apprentissage en instaurant un dialogue entre lui et l'enseignant. Notons que durant l'apprentissage l'élève peut partager son expérience et ses solutions en collaborant avec d'autres camarades.

Ainsi, hormis l'introduction générale et la conclusion générale, nous trouvons utile de structurer notre mémoire de la manière suivante :

- Le premier chapitre donne un aperçu sur les fondements de l'approche par compétences. Dans ce chapitre nous montrons que l'approche par compétences est une version améliorée de l'approche par objectifs.
- Le deuxième chapitre présente notre contribution, à savoir notre application
- Le chapitre trois explique la conception de notre application à l'aide du langage UML,
- Le quatrième et dernier chapitre est consacré à la partie implémentation de notre application.

# Chapitre 1

## Les fondements de l'approche Par Compétences

### Chapitre 1 : Les fondements de l'approche Par Compétences

#### Introduction

L'approche par compétences (désormais APC) est une méthodologie éducative implantée actuellement dans plusieurs pays au niveau mondial. Le courant de réflexion pédagogique exploitant la notion de compétence pour structurer et orienter les dispositifs d'enseignement supérieur s'est développé à partir de 1980 aux Etats-Unis. Il a graduellement pris de l'ampleur et s'est rapidement étendu aux milieux européens, notamment anglo-saxons ces dernières années. Son application dans le système éducatif algérien date de 2002 suite à la réforme du système éducatif initié par les différents ministères (de l'éducation et de l'enseignement supérieur). Cette approche est une version améliorée de la pédagogie par objectifs (désormais PPO) et qui a pour le but de régler ses limites et inconvénients.

#### 1. Pédagogie Par Objectif

La PPO est une technologie éducative prônée par Tyler (1949) [Tyler, 1949]. Apparue aux États-Unis au cours des années 1950 d'abord dans un contexte socio-économique, celui de l'industrie automobile, elle s'est ensuite diffusée dans le domaine éducatif à travers les travaux de Bloom. Puis elle s'est développée au Canada avant d'arriver en Europe, lors du mouvement de rationalisation des systèmes éducatifs. La PPO a été appliquée dans la formation professionnelle et technique avant d'être étendue à l'enseignement général au cours des années 1980 et au français enseigné comme une langue étrangère (désormais FLE) lors du passage au communicatif avec l'appui du Conseil de l'Europe, par le biais du projet « Langues vivantes » de 1971. Depuis et jusqu'à présent cette pédagogie a apporté ses fruits dans le monde éducatif.

##### 1.1 Notion d'Objectif Pédagogique

La notion d'objectif pédagogique a été introduite pour répondre à la nécessité d'introduire davantage de rigueur dans les dispositifs de formation. Elle visait à rendre explicites les finalités d'une formation, en formalisant le contrat didactique entre les enseignants et les apprenants et énonçant clairement ce que ces derniers devaient apprendre.

### **1.1.1 Définition de l'objectif pédagogique**

La notion d'objectif pédagogique a reçu plusieurs définitions. Pour Mathilde BOURDAT un objectif pédagogique est « ce que l'apprenant sera capable de faire à l'issue de la formation ». Il est incontournable de la formation de formateurs, un élément essentiel du contrat entre le formateur et ses apprenants [Bourdat, 2009].

Omar BENKHADRA, pense que les objectifs pédagogiques désignent les comportements que l'élève doit être en état d'accomplir à l'issue d'un apprentissage. Ce dernier sert à développer une compétence qui peut s'exprimer en termes de capacité [Benkhadra, 2004].

Nous comprenons par objectifs pédagogiques, l'ensemble de comportements observables (actions mesurables) attendu de l'élève à la fin d'un processus d'apprentissage, les objectifs pédagogiques sont vus comme un ensemble de comportements (performance) qu'on désire voir l'apprenant capable de manifester pendant ou à l'issue d'un apprentissage.

Cette définition précise ce que l'apprenant doit être capable de réaliser pour donner la preuve qu'il a atteint l'objectif. Donc, définir un objectif pédagogique, c'est préciser ce qu'on se propose d'atteindre à la fin d'une action éducative.

### **1.1.2 Importance des objectifs pédagogiques :**

Les objectifs pédagogiques permettent de déterminer avec précision les résultats attendus en termes de connaissances, d'attitudes et de comportement de l'apprenant à la fin des activités d'apprentissage. Ils guident le choix d'enseignement, c'est-à-dire obligent les enseignants à penser et à préparer leurs activités de façon précise, spécifique et détaillée.

Ils permettent aussi aux enseignants d'opérer un choix judicieux des méthodes et techniques d'enseignement-apprentissage, du matériel didactique adéquat, de l'organisation des classes et des moyens d'évaluation. Les objectifs pédagogiques favorisent la transmission de savoir chez l'enseignant et l'acquisition de connaissance chez l'apprenant.

## Chapitre 1 : Les fondements de l'approche Par Compétences

---

### 1.1.3. Classification des objectifs pédagogiques :

Les objectifs pédagogiques sont classés en 5 niveaux à savoir [Mager, 1971]:

- 1. Les finalités** : ce sont des énoncés très généraux définis par le pouvoir politique et qui donnent des orientations sur la politique éducative ;
- 2. Les buts** : ce sont des énoncés définissant de façon générale les intentions poursuivies par une ou plusieurs personnes, à travers un programme ou une notion de formation ;
- 3. Les objectifs généraux** : ce sont des objectifs dont la formulation reste générale et vague, c'est-à-dire non opérationnel (non évaluable directement avec précision) ;
- 4. Les objectifs spécifiques** : ce sont des objectifs définis pour chaque branche ou sous-branches et évalués à chaque séance d'apprentissage par l'enseignant ;
- 5. Les objectifs opérationnels** : ce sont des objectifs définis en termes de comportement observable par l'enseignant et qui indiquent clairement les actions concrètes que l'élève doit exécuter pour prouver la maîtrise de l'apprentissage. Ils sont du ressort des enseignants qui sont des praticiens.

### 1.1.4 Des objectifs pédagogiques à la pédagogie par objectifs

Nous comprenons que le développement du concept d'objectif et de ses applications opérationnelles est étroitement associé à l'idée de planification méthodique des activités de formation, selon la détermination des objectifs d'apprentissage, le choix des méthodes d'enseignement et d'évaluation. C'est l'utilisation même de ces objectifs qui nous a conduits à une pédagogie par objectifs.

En effet, cette pédagogie dite par objectifs trouve son origine aux États-Unis dans un contexte socio-économique de rationalisation des processus de production industrielle à partir des années soixante. Ralph TYLER en est le fondateur patenté. Sa pédagogie par objectif propose une organisation scientifique et rationnelle de l'éducation. Il a été suivi par Benjamin BLOOM qui de son côté, développait aussi cette notion dans les années 1970.

La pédagogie par objectifs trouve également son origine dans le contexte théorique du Béhaviorisme. Ces deux origines nous amènent à l'apport du Béhaviorisme et celui de Benjamin BLOOM.

### 1.2. Limites de la Pédagogie Par Objectifs

Bien que la PPO ait apporté ses fruits pendant des décennies, elle présente quelques limites auxquelles l'approche par compétences apporte des solutions. Ces limites sont :

- ✓ Cette approche n'est pas adaptée à tous les niveaux, enseignement, matière ;
- ✓ L'apprenant a une place d'exécutant, il ne prend pas part à la découverte progressive des contenus en suivant sa propre progression ;
- ✓ Le risque de morcellement des apprentissages ;
- ✓ Le comportement observable ne nous dit pas l'essentiel ;
- ✓ L'apprenant étant au centre des préoccupations, il apprend des objectifs sans en comprendre le sens et sans savoir quel lien a son apprentissage avec la vie de tous les jours.
- ✓ La PPO s'est renfermée, selon Pelpel [Pelpel, 2002], « dans un opérationnalisme comportemental, ce qui l'a énormément éloignée de l'acte pédagogique et l'a transformée en un acte constitué de réflexes conditionnés faisant abstraction de toute pensée créative chez l'apprenant. » Ce qui implique qu'avec la PPO nous sommes donc dans le conditionnement, le montage de réflexes et non dans la construction des savoirs par l'apprenant, dans l'appel à son potentiel cognitif.
- ✓ Étant soumis aux objectifs de l'enseignant, l'apprenant n'est pas toujours au centre du processus d'apprentissage, surtout si la PPO se résume à fixer d'une manière technocratique les objectifs.
- ✓ On reproche à la PPO de fractionner les savoirs à apprendre au point que l'élève perd la finalité des apprentissages, ce que souligne Deronne [Déronne, 2012] en accusant la PPO « de trop compartimenter les savoirs en décomposant les contenus en de multiples objectifs opérationnels. cette accumulation de connaissances cloisonnées engendrait une perte de sens des apprentissages et une incapacité des élèves à mobiliser les savoirs spontanément dans des situations pour lesquelles ils seraient pertinents. ».

A travers ces différents défauts, les chercheurs en Sciences de l'Éducation ont mis en exergue de nouvelles pistes de réflexions et d'action pour sortir de ces impasses : l'approche par compétences revient en force et devient de plus en plus courante dans la littérature en éducation (enseignement)

### 2. Approche Par Compétences :

L'émergence du concept de compétence et le développement progressif de l'approche par compétences en formation, constituent une réponse à la préoccupation d'apporter des solutions aux problèmes et aux limites identifiés ci-haut dans l'approche précédente. Une approche pédagogique par compétences utilise les compétences nécessaires dans un domaine ou une pratique comme point de départ dans la conception et le développement d'une activité.

En effet, cette approche par compétence trouve ses origines dans les mouvements sociopolitiques plutôt que dans une théorie d'apprentissage. Son développement dans l'enseignement-apprentissage menait les apprenants à répondre aux critères nécessaires pour s'intégrer dans un domaine ou une pratique. Pour bien comprendre les perspectives de cette approche, nous abordons les points qui suivent.

#### 2.1 Définition de la Compétence

Des nombreux chercheurs ayant creusé le concept de compétence s'accordent à dire que toute compétence résulte d'une mobilisation de ressources, elle se met en œuvre dans une situation bien précise et elle est faite pour résoudre des situations-problèmes.

Philippe PERRENOUD [Perrenoud, 2000] définit la compétence comme étant une capacité d'action efficace face à une famille de situation, qu'on arrive à maîtriser parce qu'on dispose à la fois des connaissances nécessaires et de la capacité de les mobiliser à bon escient, en temps opportun, pour identifier et résoudre des vrais problèmes.

Selon Jacques Tardif [Tardif, 2006], une compétence est un « Un savoir agir complexe prenant appui sur la mobilisation et la combinaison efficaces d'une variété de ressources internes et externes à l'intérieur d'une famille de situations »

De KETELE J.M [De Ketele, 2006] soutient que la compétence est un ensemble ordonné de capacités, activités qui s'exercent sur des contenus dans une catégorie donnée de situations pour résoudre des problèmes posés par celles-ci. Ainsi, il distingue trois éléments structurels de la compétence : le contenu, la capacité et la situation.

Pour Richer [Richer, 2014], le monde du travail « a emprunté à la linguistique chomskyenne la notion de compétence pour son sens d'adaptation créative aux exigences de la complexité. Il s'en est servi afin de désigner chez les employés et dans les entreprises une capacité, devenue indispensable, à s'adapter à des situations professionnelles de plus en plus

## **Chapitre 1 : Les fondements de l'approche Par Compétences**

---

complexes, instables et événementielles ». Le Boterf désigne cette capacité de sortir du prescrit pour faire face à l'imprévu par l'expression de « savoir-agir ».

Selon Zarifian [Zarifian, 2003] une compétence est « le pouvoir d'action, la capacité à donner du sens et l'engagement de la subjectivité de ceux qui s'affrontent, au quotidien, à des situations professionnelles. ». La compétence dans le monde de travail est profondément ancrée dans l'action singulière, fait que traduit la définition désormais classique de la compétence donnée par Le Boterf.

Malgré les différentes définitions apportées au concept de compétence, nous relevons des termes communs: savoirs, savoirs- faire, savoirs- être, capacité, attitude, connaissance, activité, démarche, situation, tâche, action, complexe, comportement, performance et habilité. Cela montre la souplesse de ce terme et sa complexité, mais nous pouvons dire que la compétence est l'action de l'apprenant en face d'une tâche, action ou un problème dans une situation complexe lorsque l'apprenant doit analyser la tâche et la situation, qu'il doit ensuite remobiliser ses connaissances « théoriques et pratiques » pour sélectionner ce qu'il croit être efficace dans cette situation. Cette situation impose à l'élève de combler le manque concernant ses connaissances afin d'élaborer un plan de travail et de fixer les objectifs finaux sans oublier de choisir ou de tenir compte des paramètres qui serviront à l'évaluation, positive ou négative, de la mise en œuvre du plan et des objectif.

### **2.2 Types de compétences**

Dans le cadre de l'approche par compétences, les compétences ci-après peuvent être développées lors des diverses activités scolaires : les compétences disciplinaires et les compétences transversales.

#### **2.2.1 Les compétences disciplinaires**

Elles constituent un ensemble de compétences propres à atteindre dans une discipline scolaire. Par discipline, ces compétences sont rassemblées dans un référentiel qui présente de façon structurée toutes les compétences à acquérir au cours d'un cycle scolaire. Donc, une compétence disciplinaire est la possibilité pour un élève, de mobiliser un ensemble de ressources en vue de résoudre un problème. On distingue :

## Chapitre 1 : Les fondements de l'approche Par Compétences

---

### *a. Les compétences d'imitation*

Elles permettent de reproduire à l'identique des actions. Ce sont des compétences qui rendent possible la réalisation d'activités d'exécution répétitives, automatisées selon des procédures spécifiques prédéfinies.

### *b. Les compétences de transposition*

Elles permettent, partant d'une situation donnée, de faire face à des situations imprévues mais proches, en raisonnant par analogie. Dans ce cas, il s'agit d'adopter, d'ajuster à des situations différentes des démarches bien maîtrisées car déjà mises par ailleurs.

### *c. Les compétences d'innovation*

Elles permettent de faire face à un problème nouveau, donc avec une solution inconnue, en puisant dans un patrimoine de connaissances et en recomposant à partir d'elles les éléments nécessaires à la solution. Ici, l'élève ne dispose plus de modèle à appliquer, auquel se référer, il doit rechercher, concevoir, créer. C'est une situation de résolution de problème, d'élaboration de stratégies visant à mettre en œuvre une solution non préalablement identifiée.

## 2.2.2 LES COMPETENCES TRANSVERSALES

Philippe JONNAERT [Joannert, 2009] trouve qu'une compétence transversale permet l'utilisation par le sujet d'un réseau opératoire stabilisé de capacités et d'habiletés dans de nombreuses situations qui n'appartiennent pas nécessairement à la pédagogie, classe de situations.

D'après Mario Richard & Steve BISSONNETTE [Richard, 2001], les compétences transversales sont communes à plusieurs tâches, activités, fonctions, métiers et réutilisables dans un grand nombre de situations. Une compétence est transversale lorsqu'elle s'exerce dans un large spectre de situations : problématiques des différentes disciplines, problématiques multidisciplinaires et problématique de la vie courante.

Tout comme la compétence disciplinaire, la compétence transversale est un savoir-agir faisant appel à des connaissances, habiletés et capacités qui sont intégrées et accessibles en mémoire, mobilisables dans l'action, parce qu'elles ont été exercées régulièrement avec succès

## Chapitre 1 : Les fondements de l'approche Par Compétences

---

dans une grande variété de contextes de discipline et cet autant à l'école qu'à l'extérieur du milieu scolaire. Ainsi, les compétences transversales se développent de la même manière et en utilisant les mêmes schèmes et niveaux d'intégration mémoriels que les compétences disciplinaires, sauf qu'elles s'en distinguent par la diversification de leurs contextes d'utilisation.

### 2.3 Notion de situation-Problèmes

Le terme «situation-problème» est souvent utilisé à mauvais escient et regroupe plusieurs définitions, sans doute à cause d'une dénomination qui prête à confusion, liée au terme « problème », mais aussi d'un manque de références aux sources originelles du concept ou encore d'une interprétation erronée de la démarche proposée, donc c'est une activité pédagogique consistant en l'aménagement d'une tâche de travail destiné à faire découvrir, par l'élève lui-même, des solutions à un problème. La résolution de ce problème doit permettre à l'élève l'acquisition de nouvelles connaissances (savoir, savoir-faire...).

Selon MEIRIEU [MEIRIEU,1987] une situation problème est « Un sujet, en effectuant une tâche, s'affronte à un obstacle. »

La situation-problème doit être :

- **Réaliste**, réelle ou vraisemblable, ce qui montrera à l'étudiant la valeur concrète des savoirs et le rapport des activités avec le monde extérieur ;
- **Nouvelle**, pour empêcher le recours à des recettes et amener l'étudiant à mobiliser ses ressources ;
- **Globale**, comportant des données qui précisent le contexte et le but à atteindre. Ce but est un savoir construit qui apporte une explication raisonnée à un phénomène où le dépassement d'une difficulté concrète.

Il s'agit donc de concevoir une situation qui mette les étudiants devant un obstacle à surmonter, un défi à réussir ou une énigme à résoudre. Les étudiants doivent se sentir aptes à le faire et s'y impliquer. Les situations-problèmes peuvent être utilisées pour l'enseignement, l'apprentissage et l'évaluation. Elles permettent :

- ✓ De consolider l'apprentissage en amenant l'étudiant à réemployer ses savoirs et ses savoir-faire et à les mettre en relation,

## Chapitre 1 : Les fondements de l'approche Par Compétences

---

- ✓ D'engager l'étudiant dans une réflexion sur ses démarches d'apprentissage (comment il s'y prend, son degré de maîtrise des résultats d'apprentissage attendus par la formation,) et sur sa capacité à les exploiter,
- ✓ De développer l'identité professionnelle chez l'étudiant en l'exposant à des situations contextualisées.

Une situation problème permet à l'élève d'utiliser ses acquis antérieurs mais néanmoins insuffisants, pour résoudre le problème proposé. La situation-problème est lancée généralement en début de séance de cours. La construction de la réponse se fait tout au long de la séance.

### 2.4 Notion d'étayage

Le concept d'étayage a été défini pour la première fois par le psycholinguiste Jérôme Bruner dans un article publié dans *Savoir- faire, savoir- dire*. A partir d'une situation dyadique expérimentale au cours de laquelle un tuteur (expert) adulte aide des enfants d'âge préscolaire à la résolution d'un puzzle tridimensionnel, Bruner analyse les composantes des activités d'étayage déployées par l'expert. Il s'agit d'un soutien pédagogique initial que fournit l'enseignant à ses élèves dans le but de leur permettre de développer une compétence ou d'apprendre un contenu il s'appuie sur le concept d'étayage et qu'il désigne comme un ensemble d'outils permettant de faciliter l'apprentissage de l'apprenant sans nuire au développement de son autonomie

L'étayage des apprentissages renvoie à une définition du processus de tutelle : "Il s'agit des moyens grâce auxquels un adulte (ou un spécialiste) vient en aide à une personne moins adulte ou moins spécialiste que lui". Donc, le processus d'étayage consiste à rendre l'apprenti capable de résoudre un problème, de mener à bien une tâche, d'atteindre un but qui aurait été, sans assistance, au-delà de ses possibilités. Ce qui signifie que le soutien de l'adulte consiste à prendre en main les éléments de la tâche qui excèdent initialement les capacités du débutant, lui permettant de se concentrer sur les éléments qui demeurent dans son domaine de compétences et de les mener à terme.

Le processus d'étayage guide l'apprenant par la forme du contexte, de la situation problème proposées, à s'autonomiser vers une conduite de résolution, qui sans cette aide n'aurait pu être réalisés, car au-dessus de ces capacités. Différentes fonctions et types d'étayage ont été proposées.

### 2.4.1 Fonctions d'étayage

Bruner propose un processus d'étayage en six fonctions [Bruner, 1997]. Ces fonctions sont censées aider l'apprenant à surmonter les difficultés qu'il peut ou pourra rencontrer lors de la résolution de situations problèmes. Donc, en faisant comprendre ses difficultés à l'apprenant, nous lui venant en aide. Par- là, l'enseignant s'en tient à un rôle de validation jusqu'à ce que l'apprenant devienne plus ou moins indépendant et puisse voler de ses propres ailes. Les six fonctions sont :

- **L'enrôlement** : cette fonction consiste à deux choses, d'une part, « susciter l'intérêt et l'adhésion de l'apprenant envers les exigences de la tâche » et d'autre part également accepter cette situation de l'apprenant.
- **La réduction des degrés de liberté** : c'est « une simplification de la tâche par réduction du nombre des actes constitutifs requis pour atteindre la solution ». Le tuteur va décomposer la tâche en une multitude de tâches que l'apprenant sait faire seul, le cadrage des différentes tâches peut permettre au final à l'apprenant de résoudre seul ce qu'il n'aurait pu faire seul si la tâche lui avait été proposée dans sa globalité.
- **Le maintien de l'orientation** : cette fonction est importante, car elle permet à l'apprenant de ne pas s'éloigner de son objectif, « le tuteur a pour charge de les maintenir à la poursuite d'un objectif défini ». Cela passe donc par la valorisation et l'encouragement dans la poursuite des tâches et de sa formation.
- **La signalisation des caractéristiques déterminantes** : les tuteurs doivent veiller à souligner les caractéristiques pertinentes de la tâche pour son exécution, ce qui permet à l'apprenant d'évaluer ce qu'il a produit et « ce que lui-même aurait considéré comme une production correcte. ».
- **Le contrôle de la frustration** : l'échec dans la résolution d'une tâche, d'une situation problème doit être accepté, l'erreur ne doit pas être vécue tel un jugement, l'apprenant doit se servir de celle-ci et le tuteur l'y encourager. « La résolution de problème devrait être moins périlleuse ou éprouvante avec un tuteur que sans lui ». Toutefois, il ne faut pas générer une trop grande dépendance vis-à-vis du tuteur.
- **La démonstration ou la représentation de modèle** : « la démonstration ou présentation de modèles solution pour une tâche, si l'on observe attentivement, exige

considérablement plus que la simple exécution en présence de l'élève. ». Bruner souligne l'intérêt pour un apprenant de voir le tuteur tester par lui-même sa solution (celle de l'apprenant), par imitation, cela permet à l'apprenant de comprendre ce qui a permis de réussir ou d'échouer dans la réalisation de la tâche, puis permettre à l'apprenant de refaire cette tâche suite à la démonstration.

### 2.4.2 Les Types d'étayage

On distingue les types suivants :

- **L'Étayage Dialogique** : L'enseignant, en mettant du sens sur la situation à un moment donné, peut le faire par le biais d'un étayage dialogique. Il s'agit d'une mise en mots qui peut permettre à l'élève de transformer la situation. Selon Bruner, pour qui le langage de l'adulte (en particulier la mère) est très important pour le tout-petit, un facteur primordial du développement de la pensée enfantine est que l'expérience de l'enfant n'est jamais purement sensori-motrice. Elle est mise en œuvre, aussitôt, par le langage de l'adulte.
- **L'Étayage En Aval** : Dans ce type d'étayage, c'est l'enseignant qui sollicite, les élèves parlent en premier. Autrement dit, la personne qui étaye parle en second, elle approuve, corrige, ajoute, se sert d'éléments du discours de la personne étayée « reprises »
- **L'Étayage En Amont** : Contrairement à l'étayage en aval, le sujet qui étaye (l'enseignant) parle en premier (ébauche, introduction d'un thème).
- **L'Étayage Latéral** : Ce type d'étayage consiste à aider la personne (dans notre cas c'est l'élève) à trouver elle-même comment faire.
- **L'Étayage Complémentaire** : Pour ce type d'étayage, l'enseignante essaye d'introduire un discours en amont ou en aval qui est le complément de ce que la personne (l'élève) doit faire, par exemple la question « et pour quoi faire ? ».
- **L'Étayage Parallèle** : Cet étayage consiste à établir une ébauche, dire ce que l'autre n'a pas pu dire, donner un modèle qui a la particularité ici de renvoyer l'élève sur ce qu'il a déjà dit une première fois (implicite)
- **L'Étayage Métalinguistique** : C'est un étayage en aval, c'est-à-dire faire redire autrement, faire expliciter.

## Chapitre 1 : Les fondements de l'approche Par Compétences

- **L'Étayage Global** : C'est le fait d'être attentif au bien-être du locuteur et ne pas oublier qu'il y ait des effets de contre-étayages par exemple quand celui qui étaye utilise trop l'étayage parallèle.

### 2.5 Comparaison entre la PPO et l'APC

La PPO et l'APC reposent chacune sur une démarche qui lui est propre. Le tableau ci-dessous donne la comparaison entre ces deux approches en se basant sur un ensemble de critères pertinents comme le fondement et le profil de sortie.

Critère de comparaison	PPO	APC
Entrée	Les contenus	Les situations
Fondement	La transmission de l'enseignement	Les actions de l'apprenant
Contextualisation	Hors contexte	Dans un contexte
Centrée sur	L'enseignement d'un contenu	L'apprentissage
Ressources	Ressources cognitives sur les contenus	Pluralité de ressources
Profil de sortie	Contenus disciplinaires à reproduire	Situations à traiter au terme de la formation
Paradigme	Le comportementalisme	Comportementalisme et Socio-constructivisme

Tableau I.1 : comparaison entre la PPO et L'APC

### Conclusion :

Dans ce chapitre, nous avons donné un aperçu global sur l'approche par compétences, courant pédagogique nouveau. L'Approche par compétences est une manière de concevoir, de penser et de mettre en œuvre l'enseignement et l'apprentissage qui vient combler les insuffisances d'une approche par objectifs d'avantage centrée sur l'acquisition de savoirs et savoir-faire (apprendre quoi), négligeant l'acquisition des processus intellectuels (comment apprendre ?). Elle se propose de faire participer activement l'élève à sa formation par l'action. Dans le chapitre qui suit nous donnons un aperçu sur l'étayage dans l'apprentissage disciplinaire, comme nouvelle méthode pour soutenir les apprenants dans la résolution de situations problèmes.

# Chapitre 2

## **Présentation de notre Application**

### Chapitre 2 : Présentation de notre Application

#### Introduction

Dans le chapitre précédent, nous avons donné un aperçu sur l'approche par compétences. On rappelle que dans la logique de cette approche, les élèves ont pour but de développer des compétences disciplinaires et/ou transversales tout en étant guidés et aidés (étayés) par leur enseignant. Les élèves peuvent aussi collaborer entre eux lors de la résolution de situations problèmes liées aux compétences à développer. Nous présentons dans le cadre de ce chapitre notre application concernant l'expérimentation des travaux de recherche menés dans [Berkane, 2015]. Ces travaux consistent en un guidage et un étayage de l'élève-apprenant dans l'apprentissage de compétences liées à toute discipline. Notre expérimentation concerne la discipline « Algorithmique ».

#### 1. Guidage dans l'Acquisition de Compétences Disciplinaires

Quel que soit l'approche d'enseignement/apprentissage utilisée dans une discipline donnée, l'enseignant se base sur les programmes scolaires pour organiser son enseignement/apprentissage. De manière générale il ordonnance les concepts à apprendre selon une organisation qui lui est propre, la plupart du temps en se basant sur l'ordre de maîtrise de ces concepts. L'ordre le plus utilisé est celui des prérequis. Par exemple dans l'approche par compétences qui nous intéresse, l'élève doit maîtriser certaines compétences avant d'autres car nécessaires à ces dernières (Cf. figure 2.1).

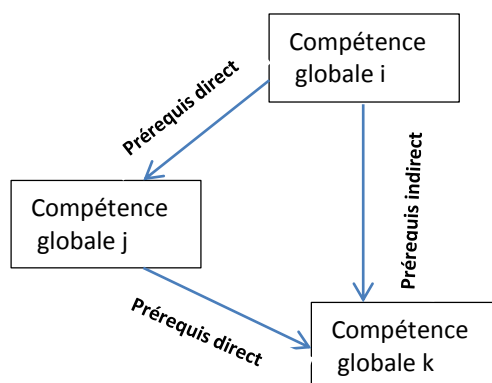


Figure 2.1 : Relation de prérequis entre compétences

## Chapitre 2 : Présentation de notre Application

L'enseignant prévoit aussi un autre ordre, celui d'aller du général au spécifique et vice versa. Pour ce dernier cas, on parle de relation de composition entre concepts (Cf. figure 2.2).

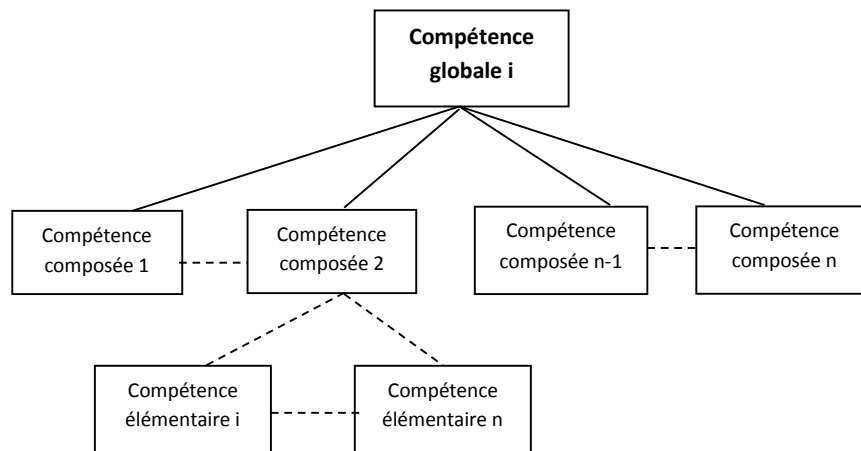


Figure 2.2 : Relation de composition entre compétences

Il existe trois types de compétence (CP) :

**-Compétence globale (CG):** chaque discipline se compose d'un ensemble de compétences globale (CG) à acquérir. Chaque CG est composée d'autres compétences d'un niveau inférieur appelées compétences intermédiaires(CI).

**-Compétence intermédiaire (CI) :** est un nœud intermédiaire dans notre arbre de compétences, qui sera composée de plusieurs compétences de bas niveau appelées compétences élémentaires (CE).

**-Compétence élémentaire (CE):** représente une compétence granulaire ou basique donc indivisible.

A chaque compétence (globale, intermédiaire ou élémentaire), l'enseignant associe des situations problèmes dans lesquelles ces compétences se développent (Cf. Figure 2.3). Chaque situation problème étant bâtie autour d'obstacles identifiés par l'enseignant concernant la discipline enseignée.

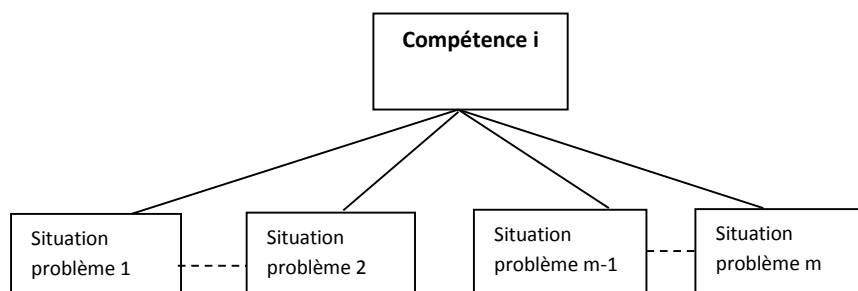


Figure 2.3 : Association compétence-situations problèmes

## Chapitre 2 : Présentation de notre Application

---

C'est sur la base des deux ordres (prérequis et composition) entre compétences qu'un guidage est imposé aux élèves lors de leur acquisition de compétences disciplinaires. L'algorithme ci-dessous donne ce guidage. Le principe de cet algorithme est que pour chaque utilisateur (user, apprenant), un plan guide est affiché. Chaque élément du plan [Berkane, 2015] est une cartographie d'éléments dont la racine n'est rien d'autre qu'une compétence globale choisie parmi d'autres grâce à la relation de prérequis. Les nœuds intermédiaires et terminaux de cet arbre sont des compétences respectivement intermédiaires et élémentaires qui composent la compétence globale racine. Le parcours d'un arbre se fait de bas en haut traduisant un parcours allant du spécifique vers le général. Mais afin de montrer les compétences accessibles de celles qui ne le sont pas au cours d'un apprentissage dans un arbre donné, dans [Berkane, 2015] on a utilisé la technique d'annotation des compétences à l'aide de couleurs dont l'explication est donnée ci-dessous.

L'algorithme afficher-plan- guide ci-dessous, permet d'afficher un plan guide initial pour un apprenant « user » donné. Un plan guide est composé à un instant donné de l'apprentissage d'un ensemble d'éléments qui sont des arbres. Un arbre a pour racine une CGs, des nœuds intermédiaires sont des CIs et les nœuds terminaux des Ces. Le rôle de l'algorithme est :

-Pour chaque utilisateur on va consulter la table t\_cg contenant les CGs à développer chez l'apprenant et récupérer la liste des CGs non acquises et qui n'ont pas des prérequis ou dont les prérequis sont acquis, pour chaque compétence on va ajouter un menu de CGs récupérées (url\_cg) ;

- pour chaque CG récupérées on va chercher la liste de ses CIs non acquises qui n'ont pas des prérequis ou dont les prérequis sont acquis, pour chaque compétence on va ajouter un sous menu des CIs récupérées (url\_ci) ;

- pour chaque CI récupérées on va chercher la liste de ses CEs et pour chaque compétence on va ajouter un sous menu de Ces récupérées (url\_ce) ;

On va annoter les CPs existantes dans les menus comme suit :

- La couleur noire pour les CPs non entamées avec des CPs de bas niveaux non acquises ;
- La couleur bleue pour les CPs entamées avec au moins une CP de bas niveaux non acquises ;
- La couleur verte pour les Cps entamées avec les CPs de bas niveaux acquises ;
- La couleur magenta pour les CPs entamées qui possède au moins une sp non traitée ;

## Chapitre 2 : Présentation de notre Application

---

Donc pour les CPs avec la couleur noire ou bleue on ne peut pas les traiter, il faut d'abord traiter ses compétences de bas niveaux. Par contre Les CPs de couleur magenta ou verte sont prêtes à être traitées. Le menu plan guide sera mis à jour après chaque traitement d'une compétence donnée.

### Algorithme général d'apprentissage () ;

T\_app : table apprenants ;

Id\_app : identifiant de l'apprenant ;

Fcg (id\_cg, url\_cg): fichier des compétences globales ;

Fci (id\_ci,url\_ci): Fichier des compétences intermédiaires ;

Fce (id\_ce,url\_ce) : Fichier des compétences élémentaires ;

T\_cg (id\_app,id\_cg,etat\_cg): ce fichier contient l'historique des CGs traitées par un apprenant ;

T\_ci (id\_app, id\_ci,etat\_ci, id\_cg) : fichier contient l'historique des CIs traitées par un apprenant ;

T\_ce (id\_app, id\_ce, etat\_ce, id\_ci) : fichier contient l'historique des CGs traitées par un apprenant ;

/\* etat\_cp peut prendre trois valeur : 'A' (acquis), 'E'(entamé), 'N' (non touché)\*/

Réponse : variable booléenne ;

Début

    /\* vérifier l'inscription de l'apprenant\*/

    Ecrire ('êtes-vous déjà inscrit ?') ;

    Lire (réponse) ;

    Si réponse = 'non' alors

        /\*utilisateur non inscrit\*/

        Inscription (id\_app) ;

        Afficher \_plan\_guide (id\_app, fcg, fci,fce, t\_cg, t\_ci, t\_ce) ;

    Sinon

        /\*utilisateur déjà inscrit\*/

        Authentification ;

        Restaurer plan guide ;

    Fin si

Fin

## Chapitre 2 : Présentation de notre Application

---

Algorithme **afficher plan guide** (id\_app, fcg, fci, fce, t\_cg, t\_ci, t\_ce) ;

Trouvé : variable booléenne ;

t,z,b,d : chaîne de caractère ;

Début

t ← id\_app ;

Ouvrir t\_cg ;

Tant que (1) (non fin t\_cg) faire

Récupérer t\_cg[ id\_cg ] ; /\*l'identifiant de cg \*/ z ← id\_cg ;

Récupérer t\_cg[ id\_app ] ; /\*l'identifiant de l'apprenant\*/

Récupérer t\_cg[etat\_cg] ; /\*l'état de la cg\*/

Si (1) (t\_cg[ id\_app ] = t et t\_cg[etat\_cg] <> 'A') alors

pre\_requis (t, z, fcgp ; trouvé) ;

Si (2) (trouvé=vrai) alors

ouvrir fcg ;

Tant que (non fin fcg) faire

Récupérer fcg[ id\_cg ] ; /\*l'identifiant de la cg\*/

Récupérer fcg[ url\_cg ] ; /\* l'url de la cg\*/

Si (fcg[ id\_cg ] = z) alors

Afficher fcg[ url\_cg ] ; /\* afficher l'url de la cg\*/

Créer un menu\_G (url\_cg) ;

Fin si

Fin tant que

Ouvrir t\_ci ;

Tant que(2) (non fin t\_ci) faire

Récupérer t\_ci [ id\_ci ] ; /\*l'identifiant de ci \*/

b ← id\_ci ;

Récupérer t\_ci [ id\_cg ] ; /\*l'identifiant de cg \*/

Récupérer t\_ci[ id\_app ] ; /\*l'identifiant de l'apprenant\*/

Récupérer t\_ci[etat\_ci] ; /\*l'état de la ci\*/

## Chapitre 2 : Présentation de notre Application

```
Si (3) (t_ci[id_app]=t et t_ci[etat_ci] <> 'A' et t_ci[id_cg]=z) alors
pre_requis (t, b, fcip ; trouvé) ;

Si(4) (trouvé=vrai) faire
    ouvrir fci ;

    Tant que (non fin fci) faire
        Récupérer fci[ id_ci ]; /*l'identifiant de la ci*/
        Récupérer fci[ url_ci ]; /* l'url de la ci*/

        Si (fci[ id_ci ]= b) alors
            Afficher fci[ url_ci ]; /* afficher l'url de la ci*/
            Créer un sous menu_ci (url_ci) ;
        Fin si
    Fin tant que

Ouvrir t_ce ;

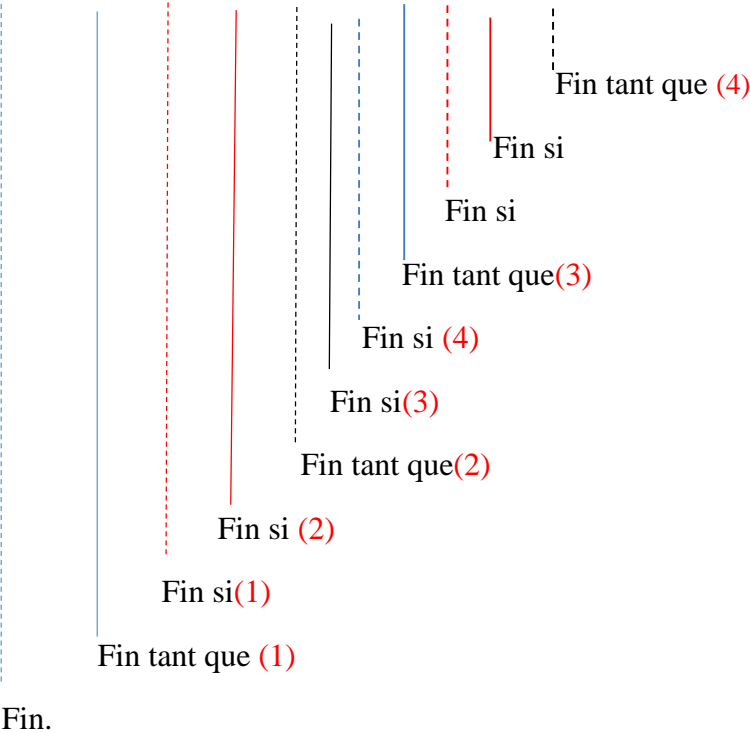
Tant que(3) (non fin t_ce) faire
    Récupérer t_ce [ id_ce ]; /*l'identifiant de ce */d←id_ce ;
    Récupérer t_ce [ id_ci ]; /*l'identifiant de ci */
    Récupérer t_ce[ id_app ]; /*l'identifiant de l'apprenant*/
    Récupérer t_ce[etat_ce] ; /*l'état de la ce*/

    Si (t_ce[ id_app ]=t et t_ce[etat_ce] <> 'A' et t_ce[id_ci]=b) alors
        pre_requis (t, d, fcep ; trouvé) ;
        si ( trouvé= vrai )
            ouvrir fce ;

            Tant que (4) (non fin fce) faire
                Récupérer fce[ id_ce ]; /*l'identifiant de la ce*/
                Récupérer fce[ url_ce ]; /* l'url de la ce*/

                Si (fce[ id_ce ]= d) alors
                    Afficher fce[ url_ce ]; /* afficher l'url de la ce*/
                    Créer un sous menu_ce (url_ce) ;
                Fin si
            Fin tant que
        Fin si
    Fin tant que
```

# Chapitre 2 : Présentation de notre Application



## Chapitre 2 : Présentation de notre Application

Procédure **pre\_requis** (t, z, fcpp ; trouvé ) ;

Fcpp (id\_cp, id\_cp\_gp): fichier des compétences prérequis; /\* il peut être fcp, fcip ou fcep\*/

/\*Trouvé est un variable booléen qui prend la valeur « vrai » pour les cps non acquises qui non pas des prérequis ou dont les prérequis sont acquis, et une valeur « faux » pour les cps qui sont acquis ou qui ont au moins un prérequis non acquis \*/

i : real ; /\* un compteur qui compte le nombre de cps prérequis non acquis\*/

t,z,y : chaîne de caractère ; /\* k représente l'id de l'apprenant, z représente l'id\_cp\*/

Début

    Trouvé ← vrai ; i ← 0 ;

    ouvrir fcpp ;

    Tant que (non fin de Fcpp) faire

        Récupérer fcpp[ id\_cp ] ; /\*l'identifiant de cp \*/

        Récupérer fcpp[ id\_cp\_gp ] ; /\*l'identifiant de cp prérequis \*/

        Y ← fcpp[ id\_cp ] ;

        Si (fcpp[ id\_cp\_gp ] = z) alors

            Ouvrir t\_cp ;

            Tant que (non fin t\_cp) faire

                Si ( t\_cp[id\_cp] = y et t\_cp[id\_app] = t ) alors

                    Récupérer t\_cp[ état\_cp ] ;

                    Si (état ≠ A ) alors

                        i ← i + 1 ;

                    Fin si

                Fin si

            Fin tant que

        Fin si

    Fin tant que

    Si (i > 0) alors

        Trouvé ← faux ;

    Fin si

Fin

### 2. Etayage dans l'Acquisition de Compétences disciplinaires

Dans chaque plan guide affiché, l'utilisateur apprenant à la possibilité de développer des compétences affichées dans l'ordre de leur accessibilité. Pour chaque compétence choisie, un ensemble de situations problèmes est affichées. L'utilisateur a le choix de choisir une situation problème à traiter. Deux espaces sont alors affichés à l'utilisateur : un espace où il peut traiter la situation problème et un espace où il peut dialoguer avec un enseignant qui le suit à distance. Il peut poser des questions à cet enseignant et répondre aux questions de cet enseignant. L'étayage proposé se réduit ici à un questionnement entre utilisateurs et enseignants comme proposé dans [Berkane, 2015].

L'étayage peut se faire à la demande de l'apprenant ou à l'initiative de l'enseignant qui constate des difficultés chez l'apprenant. L'étayage de l'enseignant qui intervient de son propre gré a pour but de cadrer l'apprenant pour qu'il ne s'éloigne pas de l'objectif visé par la situation problème qu'il est en train de résoudre. L'étayage dans notre cas se fait à l'aide d'un dialogue (questions-réponses) entre l'enseignant et l'apprenant.

La résolution d'une situation problème (SP) passe par trois (3) étapes principales. Dans chaque étape il peut contacter ses camarades, consulter des ressources et dialoguer avec un enseignant qui le suit. Les étapes sont :

- ✓ **Décodage** : dans cette étape l'apprenant cherche les mots pertinents dans l'énoncé de la SP et réécrit la sp avec ses propres phrases et envoie sa réponse à l'enseignant, ce dernier évalue la réponse, s'il trouve que la réponse est bonne, il autorise l'apprenant de passer à l'étape suivante, en cas de difficulté l'apprenant peut envoyer une question à l'enseignant comme il peut contacter ses camarades et consulter les ressources.
- ✓ **Mobilisation et Explication de la solution** : dans cette étape l'apprenant va expliquer la démarche suivie pour résoudre la sp et représenter cette démarche avec des schémas et des mots, il peut envoyer une question à l'enseignant comme il peut contacter ses camarades et consulter les ressources, à la fin il envoie sa réponse à l'enseignant, ce dernier évalue la réponse, s'il trouve que la réponse est bonne, il autorise l'apprenant de passer à l'étape suivante.
- ✓ **Validation de la solution** : après la vérification des résultats obtenues l'apprenant peut valider sa solution et l'envoyer à l'enseignant, après l'évaluation de l'enseignant il peut passer à d'autres situations à résoudre.

### Conclusion

Dans ce chapitre, nous avons présenté notre application. Dans le chapitre suivant nous donnons la conception de cette application à l'aide du langage UML.

# Chapitre 3

## Conception de l'application guidage et étayage

### Chapitre 3 : Conception de l'application guidage et étayage

#### Introduction

Dans les deux premiers chapitres nous avons présenté respectivement les fondements de l'approche par compétences et l'application de cette approche dans des disciplines informatiques notamment l'algorithmique et les bases de données. Dans ce chapitre nous présentons la conception de notre application à l'aide du langage UML.

#### 1. Présentation du langage UML[UML, 97]

UML (Unified Modeling Language) est un langage de spécification, de représentation graphique et de documentation d'un système orienté objets. Il devient aujourd'hui un standard dans le domaine d'analyse et de conception orienté objet. UML est une notation graphique simple, précise et homogène. Il propose plusieurs modèles (de cas d'utilisation, d'interactions, d'états, de déploiement) qui sont des descriptions abstraites du système étudié. Ces différents modèles sont élaborés par les utilisateurs au moyen de diagrammes. Un diagramme spécifie un aspect précis du modèle. UML comporte neuf (09) principaux diagrammes regroupés dans deux vues différentes :

- ✓ **La vue statique**, permettant de représenter le système physiquement :
  - Diagrammes de classes: représentent des collections d'éléments de modélisation statiques (classes, paquetages...), qui montrent la structure d'un modèle.
  - Diagrammes d'objets: ces diagrammes montrent des objets (instances de classes dans un état particulier) et des liens (relations sémantiques) entre objets.
  - Diagrammes de cas d'utilisation: identifient les utilisateurs du système (acteurs) et leurs interactions avec le système.
  - Diagrammes de composants: permettent de décrire l'architecture physique statique d'une application en termes de modules : fichiers sources, librairie exécutables, etc.
  - Diagrammes de déploiement: montrent la disposition physique du matériel qui compose le système et la répartition des composants sur ce matériel.
- ✓ **La vue dynamique**, montrant le fonctionnement du système :
  - Diagrammes de collaboration: montrent des interactions entre objets (instances de classes et acteurs).
  - Diagrammes de séquences: permettent de représenter des collaborations entre objets selon un point de vue temporel, on y met l'accent sur la chronologie (envoi de messages).
  - Diagrammes d'états-transitions: permettent de décrire les changements d'états d'un objet ou d'un composant, en réponse aux interactions avec d'autres objets/composants ou avec des acteurs.
  - Diagrammes des activités: sont une variante des diagrammes d'états-transitions, ils servent à représenter graphiquement le comportement d'une méthode ou déroulement d'un cas d'utilisation.

## Chapitre 3 : Conception de l'application guidage et étayage

Un diagramme est la représentation graphique d'un ensemble d'éléments que l'on représente en général par un graphe relié de sommets (éléments) et d'arcs (relations). Un diagramme est une projection dans un système. Dans notre conception, nous utilisons cinq diagrammes : le diagramme du contexte, le diagramme des cas d'utilisation, le diagramme de séquences, le diagramme des activités et le diagramme de classes. La conception va se dérouler en deux étapes : l'étape d'analyse et l'étape de conception proprement dite.

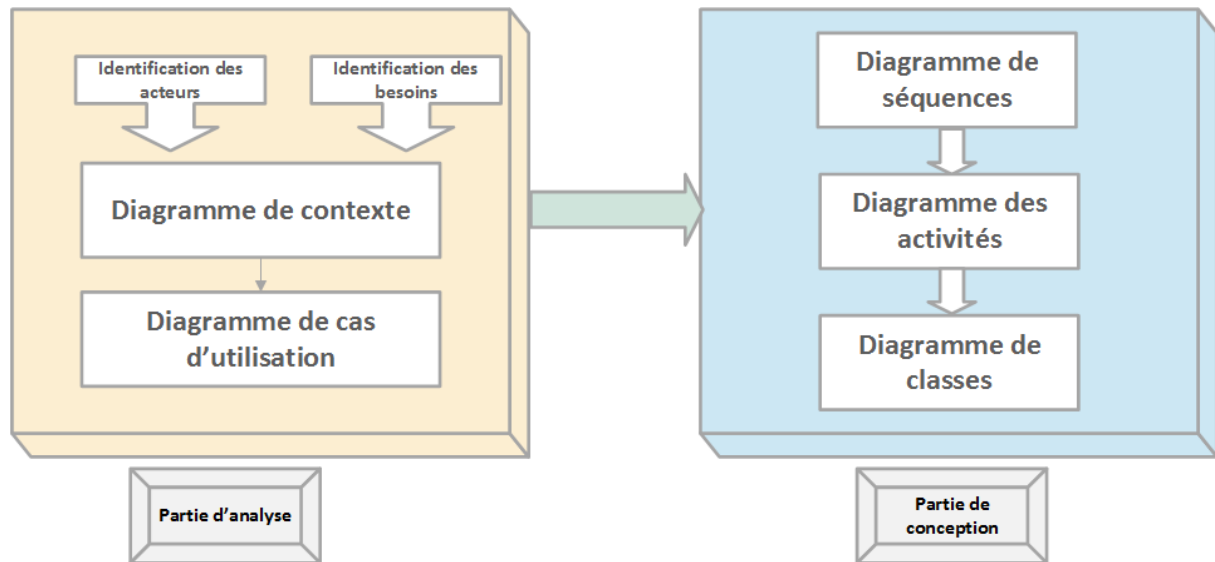


Figure 3.1 diagramme de démarche

### 2. Etape d'Analyse

L'analyse essaie de comprendre, d'expliquer et de représenter la nature profonde du système qu'elle observe. Elle ne se préoccupe pas de solutions, mais de questions, elle identifie le quoi faire et l'environnement d'un système, sans décrire le comment qui est le propre de la conception [mul, 97].

Dans cette partie, nous identifions en premier lieu les besoins et les acteurs, puis nous développons un premier modèle UML de niveau contexte, pour pouvoir établir précisément les frontières du système. Ensuite, nous identifions et décrivons en détail les cas d'utilisation.

#### 2.1 Identification des besoins

L'objectif principal de notre travail est de concevoir un système intelligent qui offre d'une part à l'apprenant l'opportunité d'acquérir des nouvelles compétences en suivant un parcours adapté, et d'autre part permet à l'enseignant d'aider et orienter l'apprenant tout au long de son parcours.

### 2.2 Identification des Acteurs

Un acteur représente un rôle joué par une personne ou une chose qui interagit avec un système. Il est représenté sous la forme d'un personnage et déclenche des cas d'utilisation. Il existe deux catégories d'acteurs :

- Les acteurs humains : ce sont les personnes qui utilisent les fonctions du système. Dans notre cas on a l'apprenant, l'enseignant, l'administrateur et le visiteur.
  - L'apprenant : toute personne désirant tester ses compétences disciplinaires dans des disciplines données. Si l'apprenant est nouveau il s'inscrit, s'il est ancien il s'authentifie. Après connexion, il accède à des plans guides affichés au fur et à mesure par le système pour choisir les compétences à développer. Pour chaque compétence à développer il doit résoudre les situations problèmes correspondantes. L'apprenant peut en cas de difficulté durant son parcours, dialoguer avec son enseignant en lui posant des questions et en répondant à ses questions. Il peut aussi partager son expérience avec des camarades, consulter son profil, modifier son profil et se déconnecter.
  - L'enseignant : toute personne qui étaye (aide ou soutient) l'apprenant implicitement (à la demande de l'apprenant) ou explicitement (à l'initiative de l'enseignant). Cet étayage se fait à l'aide d'un dialogue entre l'apprenant et l'enseignant : l'enseignant pose des questions et répond aux questions des apprenants, évalue les situations problèmes et les compétences.
  - L'administrateur est la personne chargée de gérer la base de données relative aux compétences et situations problèmes, aux utilisateurs (apprenants, enseignants).
  - Le visiteur est toute personne désirant visiter le site, de s'inscrire éventuellement pour tester ses compétences dans une discipline donnée.
- Les acteurs matériel ou système : dispositifs matériels nécessaires pour être utilisés ou systèmes avec lesquels le système doit agir. Dans notre cas nous avons le système ou outil d'apprentissage proposé.
  - le système : il est chargé d'afficher aux apprenants des plans guides de navigation dans des compétences ordonnancées (triées selon l'ordre de prérequis et de composition), de mettre à jour durant l'apprentissage de l'apprenant les plans guides en supprimant, masquant et rajoutant des compétences, d'annoter des compétences, de consulter et modifier le profil cognitif de l'apprenant.

Chaque acteur est schématisé par une figure, dans notre cas la figure suivante :



Figure 3.2 formalisme des acteurs

### 2.3 Diagramme de contexte

Notre application est un système multi-utilisateurs (acteurs): à tout instant on peut avoir plusieurs instances de chaque acteur connectées au système

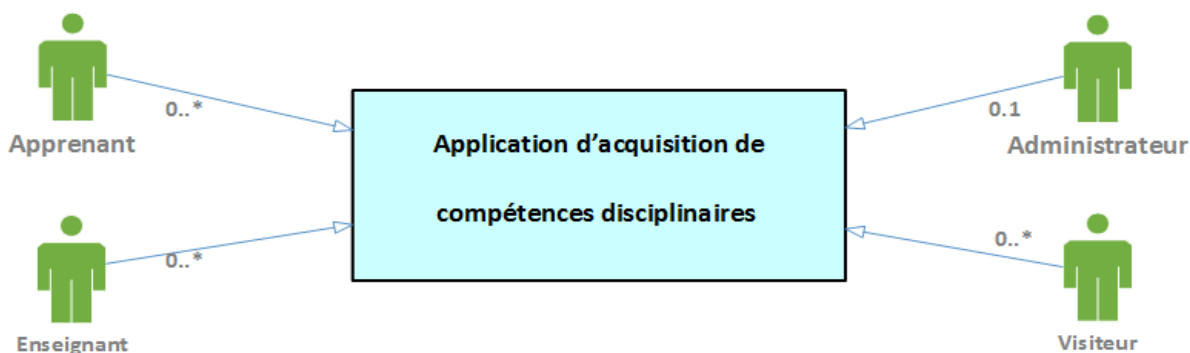


Figure 3.3 diagramme de contexte

### 2.4 Identification des cas d'utilisation

Un cas d'utilisation (CU) désigne une fonctionnalité visible de l'extérieur du système dont on désire décrire le fonctionnement, Un cas d'utilisation regroupe une famille de scénario d'utilisation du système, Les cas d'utilisation concernant notre application sont donnés dans le tableau ci-dessous :

Les Acteurs	Désignation CU	Description CU	Sous CU	
Apprenant	Inscription	Renseigner son nom, son prénom et son login		
	Authentification	Saisir son login et mot de passe pour pouvoir accéder à son espace de travail		
	Accès au plan guide	Accéder au plan guide pour suivre un parcours d'apprentissage.	Traiter une CP	
	Traiter une CP	Choisir une CP à développer dans le plan guide affiché	Résoudre une SP	
	Résoudre une SP	Fournir une solution en suivant les étapes de résolution	Consulter des ressources	
	Accès au profil		Consulter le profil cognitif (les CPS déjà traitées et les SPS résolues) ou modifier son mot de passe	Consulter Profil cognitif
				Modifier Profil
	Dialoguer avec Enseignant		Poser et répondre aux questions de l'enseignant qui l'accompagne	Poser des questions à l'enseignant en cas de difficulté
				Répondre aux questions posées par l'enseignant
				Consulter des questions déjà posées avant de poser ses questions
Consulter des questions				
Partager sa solution	Partager son expérience avec des camarades			
Déconnexion	Se déconnecter à tout moment de l'application			

## Chapitre 3 : Conception de l'application guidage et étayage

Enseignant	Poser une question	Poser des questions aux apprenants en situation de résolution de SPs	
	Répondre aux questions	répondre aux questions posées par les apprenants en situation de résolution de SPs	
	Evaluer CP	Evaluer chaque CP traitée après évaluation de toutes les SPs rattachées à cette CP.	
	Evaluer SP	Evaluer chaque SP traitée après validation de toutes les étapes de résolution de la SP.	
	Déconnexion	Se déconnecter à tout moment de l'application	
Système	Afficher plan guide	Affiche le plan guide aux apprenants connectés	Afficher CP Afficher SP d'une CP Choisie
	Modifier plan guide	Modifier le plan guide et le profil durant le parcours d'apprentissage de chaque apprenant	Ajouter CP et ses SPs
			Supprimer SP ou CP
			Annoter CP Masquer CP
Restaurer/sauvegarder Document	A chaque déconnexion le document courant est sauvegardé		
Administrateur	Gérer les comptes des apprenants et des enseignants	Ajouter, modifier ou supprimer des comptes d'apprenants et d'enseignants	Ajouter apprenant ou enseignant
			Modifier apprenant ou enseignant
			Supprimer apprenant ou enseignant
	Gérer la BDD des CPs, SPs et Ressources liées aux CPs et SPs	Ajouter, modifier et supprimer les informations liées aux CPs, SPs et Ressources	Ajouter CP ou SP ou ressource
Modifier CP ou SP ou ressource Supprimer CP, SP ou ressource			
Répondre aux messages	Répondre aux messages envoyés par des visiteurs		
Visiteur	Accéder au plan guide	Accéder au plan guide pour suivre un parcours d'apprentissage	Traiter une CP
			Résoudre une SP
	Envoyer Message	Envoyer un message à l'administrateur	
Lire Message	Lire Message de l'administrateur		

Tableau 3.1 les cas d'utilisations

Un diagramme des cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes, les cas d'utilisation, ayant un sens pour les acteurs. Ainsi ces cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin au contraire d'une vision informatique.

Il représente les cas d'utilisation, les acteurs et les relations entre eux. UML utilise trois types de relations pour le diagramme des cas d'utilisation : relation de communication, d'utilisation et d'extension. Ci-dessous nous donnons trois diagrammes des cas d'utilisation :

### Chapitre 3 : Conception de l'application guidage et étayage

le diagramme relatif aux trois acteurs apprenant, enseignant et système, il représente une démarche de développement de compétences disciplinaires par l'acteur apprenant :

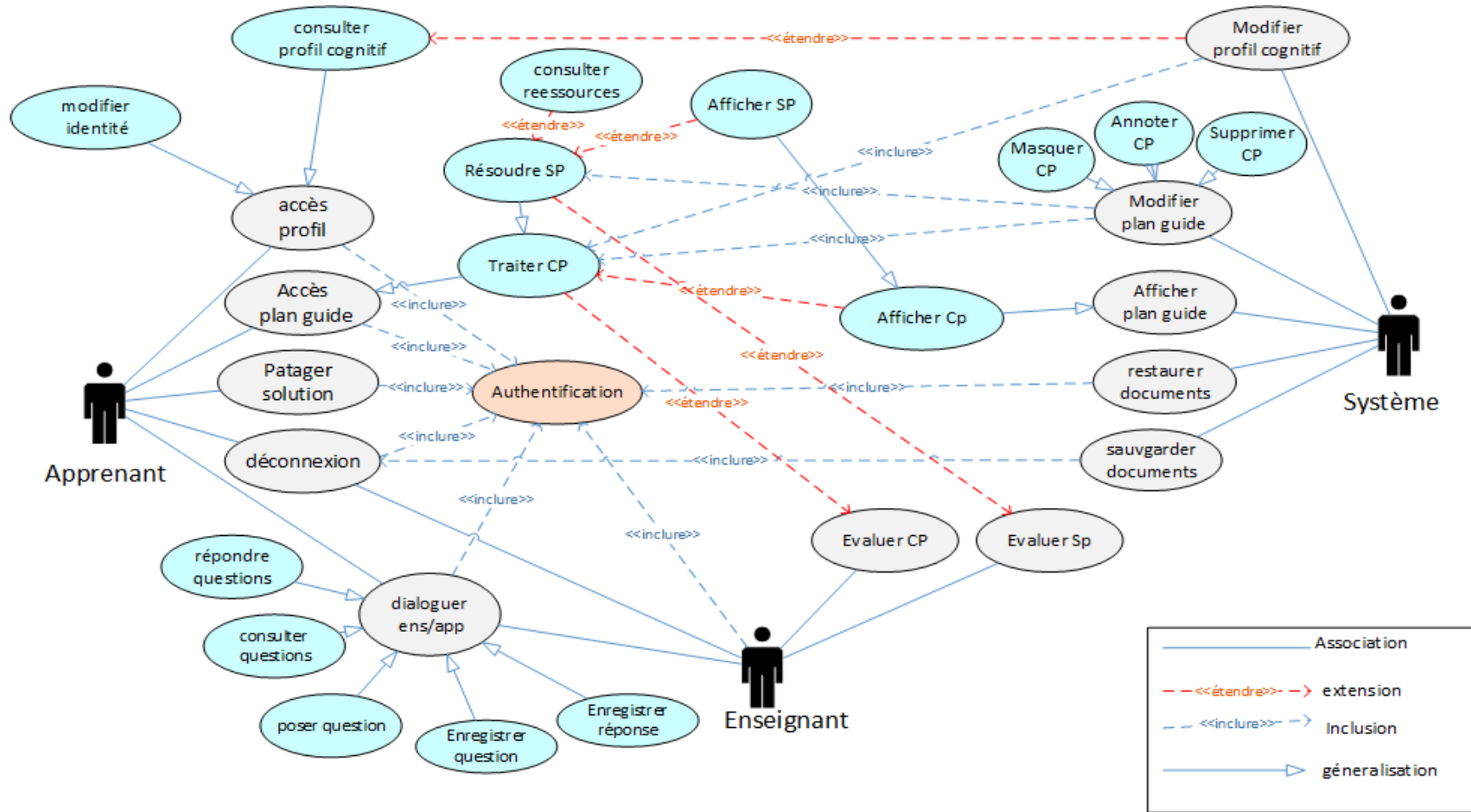


Figure 3.4 diagramme de cas d'utilisation « démarche d'acquisition de compétences disciplinaires »



## Chapitre 3 : Conception de l'application guidage et étayage

Description de cas « Inscription »
<b>Identification :</b> <b>Nom de cas :</b> inscrire un nouvel apprenant. <b>But :</b> pour suivre un parcours d'apprentissage. <b>Acteur principal :</b> Apprenant.
<b>Pré condition :</b> L'apprenant ne possède pas de compte
<b>Enchaînement nominal :</b> <ol style="list-style-type: none"><li>1. Demande d'inscription par un nouvel apprenant,</li><li>2. Le système affiche le formulaire « demande inscription »</li><li>3. L'apprenant remplit le formulaire</li><li>4. Le système vérifie dans la base des apprenants si ces informations ne sont pas déjà utilisées par un apprenant déjà inscrit. Si c'est le cas, il génère un message d'erreur sinon enregistre l'apprenant dans la base de données et lui crée un plan guide initial</li></ol>
<b>Enchaînements Alternatifs :</b> L'apprenant existe déjà dans la base de données. L'enchaînement alternatif démarre au point 3 du scénario nominal, <ol style="list-style-type: none"><li>5. Le système indique que l'apprenant existe déjà .</li></ol>

Tableau 3.2 description textuelle de cas d'utilisations « inscription »

Description de cas « Authentification »
<b>Identification :</b> <b>Nom de cas :</b> authentification <b>But :</b> permet d'accéder à son espace privé. <b>Acteur principal :</b> Apprenant, enseignant.
<b>Pré condition :</b> L'acteur doit être créé dans la base de données.
<b>Enchaînement nominal :</b> <ol style="list-style-type: none"><li>1. L'acteur se connecte au système ;</li><li>2. Le système affiche la page d'accueil ;</li><li>3. L'acteur remplit le formulaire d'authentification</li><li>4. Le système vérifie si l'authentification est correcte ;</li><li>5. Le système restaure et affiche un document interrompu par l'acteur (apprenant ou enseignant) lors d'une déconnexion ;</li></ol>
<b>Enchaînements Alternatifs :</b> Le login ou le mot de passe incorrect. L'enchaînement alternatif démarre au point 3 du scénario nominal, <ol style="list-style-type: none"><li>6. Le système indique que le login ou le mot de passe est incorrect.</li></ol>

Tableau 3.3 description textuelle de cas d'utilisations « authentification »

## Chapitre 3 : Conception de l'application guidage et étayage

Description de cas « modifier le profil »
<b>Identification :</b> <b>Nom de cas :</b> modifier le profil <b>But :</b> permet de modifier un mot de passe. <b>Acteur principal :</b> Apprenant, enseignant.
<b>Pré condition :</b> L'acteur doit être créé dans la base de données.
<b>Enchaînement nominal :</b> <ol style="list-style-type: none"><li>1. L'acteur se connecte;</li><li>2. Le système lui affiche la page d'accueil ;</li><li>3. L'acteur remplit le formulaire d'authentification ;</li><li>4. Le système vérifie si l'authentification est correcte ;</li><li>5. Le système restaure et affiche l'espace de travail de l'acteur ;</li><li>6. L'acteur clique sur le bouton « modifier profil » ;</li><li>7. Le système affiche la fenêtre « modifier profil » ;</li><li>8. L'acteur saisit son nouveau mot de passe ;</li><li>9. Le système enregistre les nouvelles informations dans la base de données des acteurs (apprenants et enseignants).</li></ol>
<b>Enchaînements Alternatifs :</b> Le login ou le mot de passe incorrect. L'enchaînement alternatif démarre au point 3 du scénario nominal, <ol style="list-style-type: none"><li>10. Le système indique que le login ou le mot de passe est incorrect.</li></ol>

Tableau 3.4 description textuelle de cas d'utilisations « modifier profil »

Description de cas « traiter une compétence »
<b>Identification :</b> <b>Nom de cas :</b> traiter une compétence <b>But :</b> permet de traiter une compétence disciplinaire. <b>Acteur principal :</b> Apprenant.
<b>Pré condition :</b> L'apprenant doit être créé dans la base de données.
<b>Enchaînement nominal :</b> <ol style="list-style-type: none"><li>1. Un apprenant se connecte au système;</li><li>2. Le système affiche la page d'accueil ;</li><li>3. L'apprenant remplit le formulaire d'authentification ;</li><li>4. Le système vérifie si l'authentification est correcte ;</li><li>5. Le système restaure et affiche le plan guide ;</li><li>6. L'apprenant choisir une compétence à traiter ;</li><li>7. Le système afficher la liste des SPS de cette CP ;</li><li>8. L'apprenant choisir une SP à résoudre et saisir son réponse ;</li><li>9. Le système enregistre la réponse et mettre à jour le profil apprenant et le plan guide.</li></ol>
<b>Enchaînements Alternatifs :</b> Le login ou le mot de passe incorrect. L'enchaînement alternatif démarre au point 3 du scénario nominal,

## Chapitre 3 : Conception de l'application guidage et étayage

10. Le système indique que le login ou le mot de passe est incorrect.  
Le login ou le mot de passe incorrect.

Tableau 3.5 description textuelle de cas d'utilisations « traiter une CP »

Description de cas « gestions des CPs et des SPs »
<b>Identification :</b> <b>Nom de cas</b> gestions des CPs et des SPs <b>But :</b> Ce cas d'utilisation permet à l'administrateur de gérer (modifier, ajouter et supprimer) les CPs et les SPs. <b>Acteur principal :</b> Administrateur.
<b>Pré condition :</b> une (des) nouvelle(s) changement (s) à effectuer à la base de données.
<b>Enchaînement nominal :</b> <ol style="list-style-type: none"><li>1. L'administrateur accède à son espace privé (idem que s'authentifier) ;</li><li>2. L'administrateur choisir une opération ;</li><li>3. Le système afficher la fenêtre concernée ;</li><li>4. L'administrateur effectuer des changements ;</li><li>5. Le système mettre à jour la base de donnée CP et SP.</li></ol>

Tableau 3.6 description textuelle de cas d'utilisations « gestion de SPs et CPs »

Description de cas « gestions des profils enseignants et apprenants »
<b>Identification :</b> <b>Nom de cas</b> gestions des profils enseignants et apprenants <b>But :</b> Ce cas d'utilisation permet à l'administrateur de gérer (modifier, ajouter et supprimer) les listes des apprenant et enseignant. <b>Acteur principal :</b> Administrateur.
<b>Pré condition :</b> une (des) nouvelle(s) changement (s) à effectuer à la base de données.
<b>Enchaînement nominal :</b> <ol style="list-style-type: none"><li>1. L'administrateur accède à son espace privé (idem que s'authentifier) ;</li><li>2. L'administrateur choisir une liste à modifier (apprenant ou enseignant) ;</li><li>3. Le système afficher la liste concernée ;</li><li>4. L'administrateur choisir un utilisateur à modifier ;</li><li>5. Le système mettre à jour la base de donnée.</li></ol>

Tableau 3.7 description textuelle de cas d'utilisations « gestion des profils »

## Chapitre 3 : Conception de l'application guidage et étayage

Description de cas « Evaluer »
<b>Identification :</b> <b>Nom de cas :</b> Evaluation <b>But :</b> Ce cas d'utilisation permet à l'enseignant d'évaluer une SP ou une CP. <b>Acteur principal :</b> Apprenant.
<b>Pré condition :</b> L'enseignant doit être créé dans la base de données et connaître ses Identifiants.
<b>Enchaînement nominal :</b> <ol style="list-style-type: none"><li>1. L'enseignant accéder à son espace privé (idem que s'authentifier) ;</li><li>2. L'enseignant choisir une SP ou une CP à évaluer ;</li><li>3. Le système affiche les réponses de l'apprenant ;</li><li>4. L'enseignant évaluer les réponses de l'apprenant ;</li><li>5. Le système mettre à jour le profil et le plan guide de l'apprenant.</li></ol>

Tableau 3.8 description textuelle de cas d'utilisations « évaluer SP ou CP »

### 3. Etape de Conception

#### 3.1 Diagrammes de séquences

Avec le texte seul il est difficile de montrer comment les enchaînements se succèdent. Il est alors recommandé de compléter la description textuelle par un ou plusieurs diagrammes UML. Pour notre cas, nous utilisons les diagrammes de séquences pour décrire les scénarios nominaux des cas d'utilisation. Un diagramme de séquences montre les interactions entre objets selon un point de vue temporel. La représentation du contexte des objets se centre sur l'expression des interactions.

Dans la suite, nous présentons juste quelques diagrammes de séquences, vu leur nombre considérable, en supposant que l'accès à l'application se fait à partir de la page d'accueil.

##### 3.1.1 Acteur apprenant :

##### Cas 1 1 demander d'inscription :

- ✓ Un apprenant se connecter au système et demander une inscription,
- ✓ Le système affiche le formulaire « demande inscription »
- ✓ L'apprenant remplir ce formulaire
- ✓ Le système consulte la base des apprenants pour vérifier si ces informations sont déjà utilisées par un autre apprenant déjà inscrit, si c'est le cas, il génère un message d'erreur sinon cet apprenant sera bien inscrit dans la base de donnée.
- ✓ Le système affiche un test initial pour déterminer le profil de l'apprenant.
- ✓ L'apprenant répond aux questions du test initial.
- ✓ le système enregistrer les informations sur l'apprenant, crée un profil propre à lui et afficher un plan guide initial

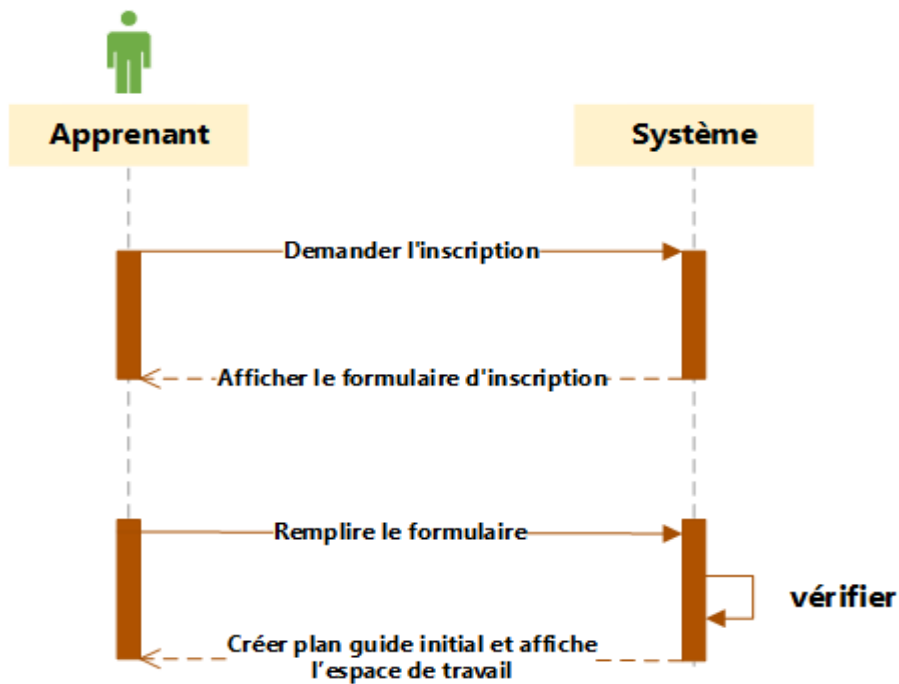


Figure 3.8 diagramme de séquence « cas d'inscription d'un apprenant »

### Cas 1 2 authentification :

- ✓ Un apprenant se connecte au système et demande une authentification,
- ✓ Le système affiche le formulaire « authentification »
- ✓ L'apprenant saisit son login et son mot de passe
- ✓ Il clique sur le bouton **Se connecte**
- ✓ Le système vérifie la combinaison nom utilisateur et mot de passe.
- ✓ Si cette Combinaison correcte, Le système restaure le dernier état de son profile.
- ✓ Si le nom d'utilisateur ou le mot de passe est erroné, le système affiche un message d'erreur

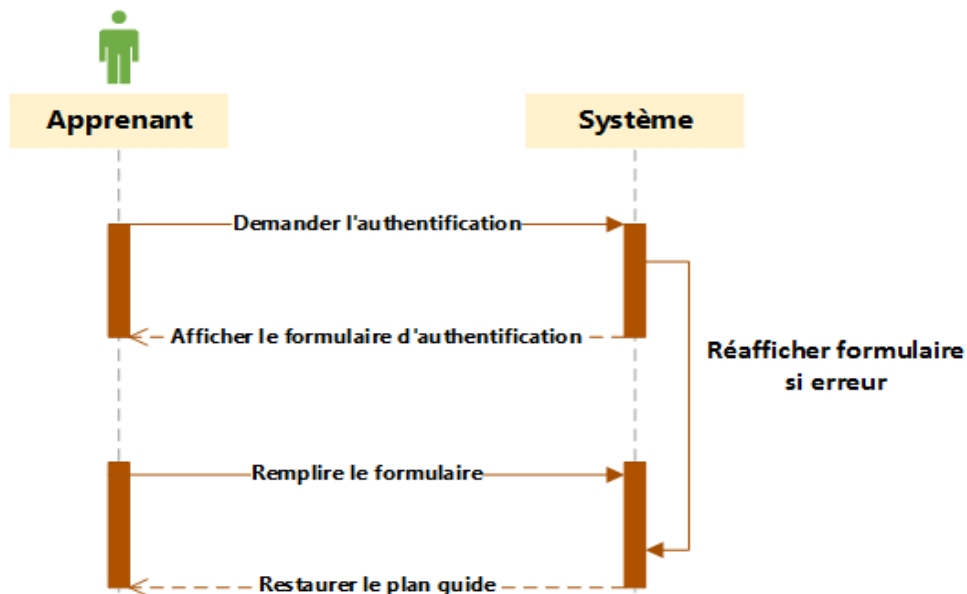


Figure 3.9 diagramme de séquence « cas d'authentification d'un apprenant »

## Chapitre 3 : Conception de l'application guidage et étayage

### Cas 1 3 traiter une compétence :

- ✓ L'apprenant accède à son espace de travail
- ✓ Le système lui restaure sa session et affiche la dernière activité qui est une compétence ou bien une SP non achevée,
- ✓ L'apprenant reprend le traitement de la CP/ SP inachevée avant de passer à la suivante,
- ✓ Dans le cas d'une compétence générale, Le système affiche la liste des compétences intermédiaires et élémentaires qui y associées,
- ✓ L'apprenant doit d'abord commencer par les compétences élémentaires puis les intermédiaires et enfin les générales,
- ✓ A chaque fois l'apprenant choisit une cp, Le système affiche les situations problèmes y associées.
- ✓ L'apprenant traite les situations problèmes,
- ✓ Le système met à jour le profil apprenant,
- ✓ L'apprenant passe à la compétence suivante,

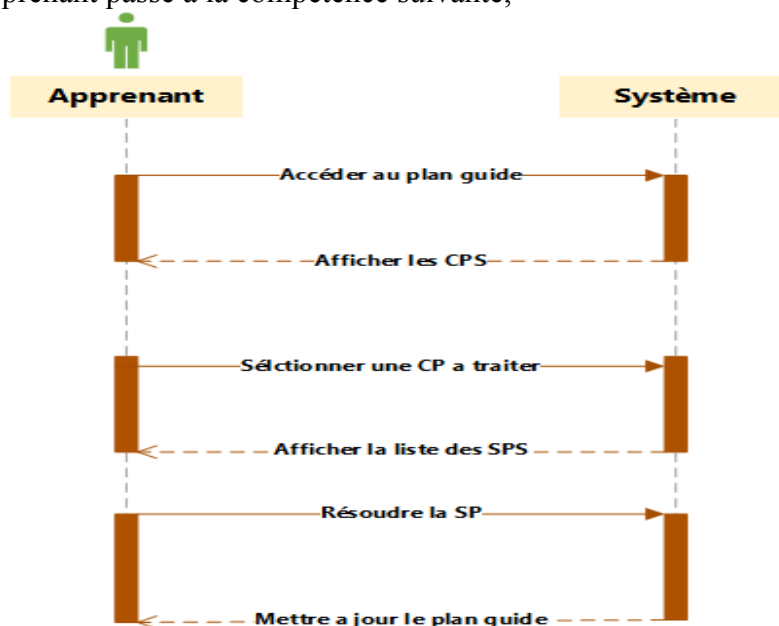


Figure 3.10 diagramme de séquence « traiter une compétence »

### 3.2 Diagramme des activités

Un diagramme d'activités représente l'état d'exécution d'un mécanisme sous la forme d'un déroulement d'étapes regroupées séquentiellement dans les branches parallèles de flots de contrôle.

## Chapitre 3 : Conception de l'application guidage et étayage

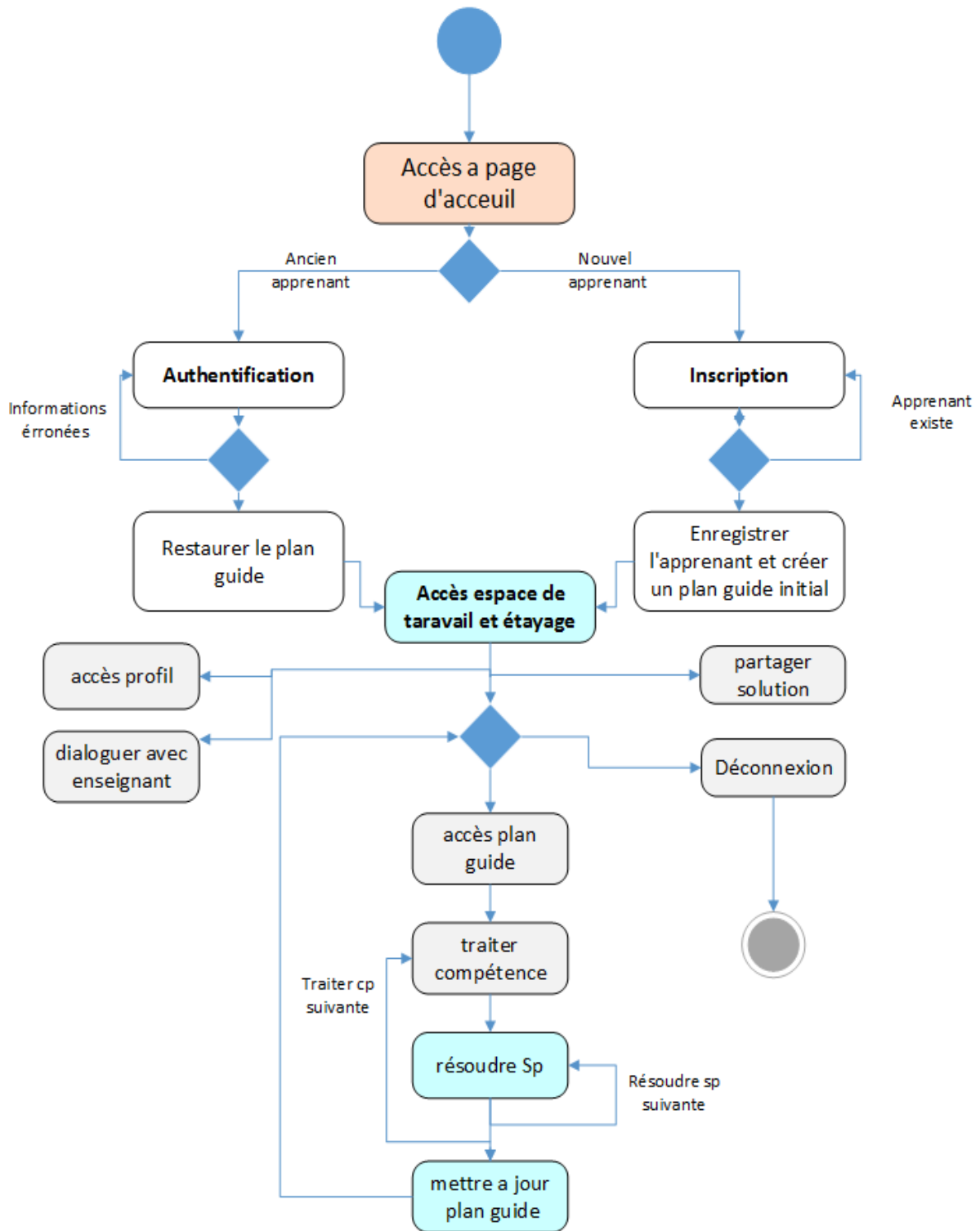


Figure 3.11 diagramme des activités



### Chapitre 3 : Conception de l'application guidage et étayage

---

3- Chaque association de type 1-n est prise en compte en incluant comme clé étrangère dans la relation .....

L'application des règles de passage citées précédemment conduit au modèle relationnel suivant :

Compétence\_cc (id\_cc, url\_cc, dm\_cc)

Compétence\_ce (id\_ce, id\_cc\*, url\_ce, dm\_ce)

Discipline (id\_dis, intitulé)

Situation-problème (id\_sp, url\_sp, dd\_sp, dm\_sp)

Solution (id\_sol, id\_sp\*, url\_sol)

Apprenant (id\_app, nom, prénom, login, mot \_passe)

Enseignant (id\_ens, nom, prénom, login, mot \_passe)

Pre\_requis\_ce (id\_ce\*, id\_cep\*)

Pre\_requis\_cc (id\_cc\*, id\_ccp\*)

Evaluer\_sp (id\_app\*, id\_sp\*, seuil \_perf, solution, état, corriger, note)

Evaluer\_cp (id\_app\*, id\_cp\*, seuil \_perf, solution, état, corriger, note)

Critère (id\_cr, url\_cr)

Question (id\_quest, destinataire, expéditeur, url\_quest)

Traiter\_cc (id\_app\*, id\_cg\*, état, score).

Traiter\_ce (id\_app\*, id\_ce\*, id\_ci\*, état, score).

Résoudre\_sp\_cc (id\_app\*, id\_sp\*, id\_cg\*, etat, corriger, score).

Résoudre\_sp\_ce (id\_etu\*, id\_sp\*, id\_ce\*, etat, corriger, score).

### 3.5 Codification des données

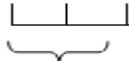

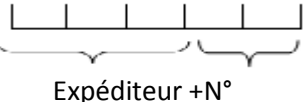
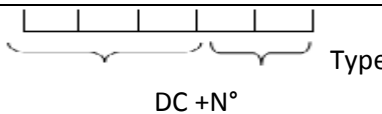
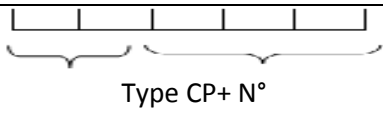
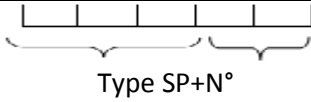
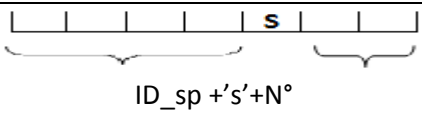
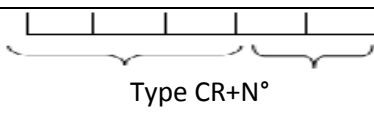
Code mnémorique attribut	Désignation	Type codification	Format codification	Observation
ID-Ens	Identifiant enseignant	Séquentiel		
ID-App	Identifiant apprenant	articulé		Login
ID-Que	Identifiant question	articulé		Exemple App20, ens02
ID-DC	Identifiant discipline	articulé		Exemple Algo1 , bdd2
ID-CP	Identifiant compétence	articulé		Exemple CG1 ou CI5 ou CE3
ID-SP	Identifiant compétence	articulé		Exemple SPG1 ou SPI3
ID-SOL	Identifiant solution	articulé		Exemple Sp1s1
ID-CRI	Identifiant critère	articulé		Exemple crg1 ou crs3

Tableau 3.9 codification des données

### Conclusion

Dans ce chapitre, le but est de donner le schéma conceptuel de données et le schéma relationnel de l'application. Dans l'analyse, nous avons spécifié les utilisateurs de notre application ainsi que leurs besoins, qui sont exprimés par les diagrammes de cas d'utilisation. Pour chaque cas d'utilisation nous avons donné une description textuelle et une présentation graphique par l'utilisation de diagrammes de cas d'utilisation. En fin, dans la conception nous avons présenté quelques diagrammes de séquences, d'activités et de classes. Dans le chapitre suivant, nous abordons le dernier maillon de la chaîne objet qui est l'implémentation.

# Chapitre 4

## **Implémentation de l'application**

### Chapitre 4 : Implémentation de l'application

#### Introduction

Dans le présent chapitre, nous décrivons l'architecture logicielle de notre système ainsi que l'environnement et les outils de développement, et nous donnons un exemple concret de notre application au travers de quelques interfaces.

#### 1 Environnement Technique de développement et d'implémentation

##### 1.1 Architecture logicielle de notre système

L'architecture de notre système est basée sur l'architecture client/serveur 3 tiers. Cette architecture est partagée entre (Cf. figure 4.1):

- **Un client** : demandeur de ressources (Formulaires d'inscription ou d'authentification, compétences à traiter, situations problèmes à résoudre, etc.) via un navigateur.
- **Un serveur d'application** : chargé de fournir les ressources aux clients en faisant appel à un autre serveur, le serveur de bases de données. A ce niveau on distingue trois modules : un module qui gère les profils des apprenants et enseignants, un module chargé d'afficher et de modifier des plans guides, un module chargé de la résolution de situations problèmes, un module fournissant un étayage à l'apprenant lors de la résolution des situations problèmes, un module pour la collaboration, un module pour l'évaluation des compétences et des situations problèmes.
- **Un serveur de données** : fournissant au serveur d'application les données dont il a besoin et qui se trouve dans des bases de données. Ces données concernent les profils apprenants, les enseignants, les compétences, les situations problèmes, les questions et réponses liées à l'étayage, etc.

## Chapitre 4 : Implémentation de l'application

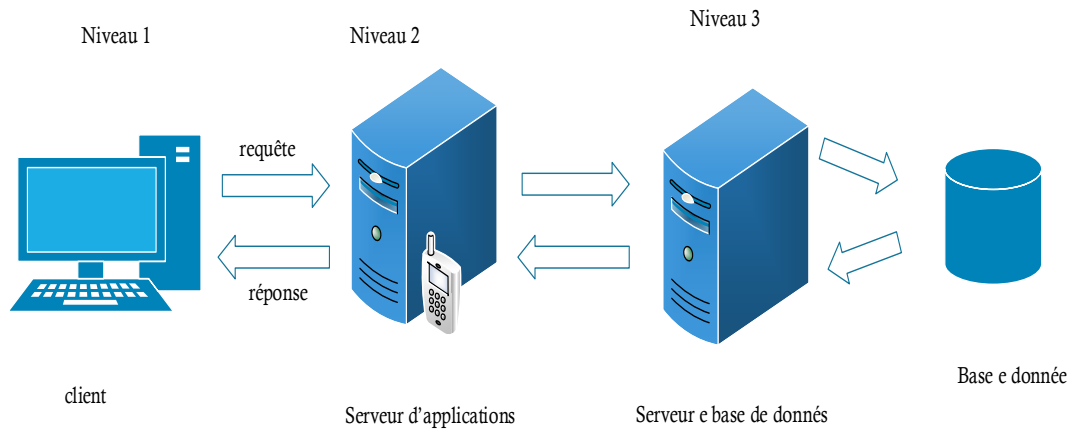


Figure 4.1 architecture a 3 niveau

### Architecture logicielle :

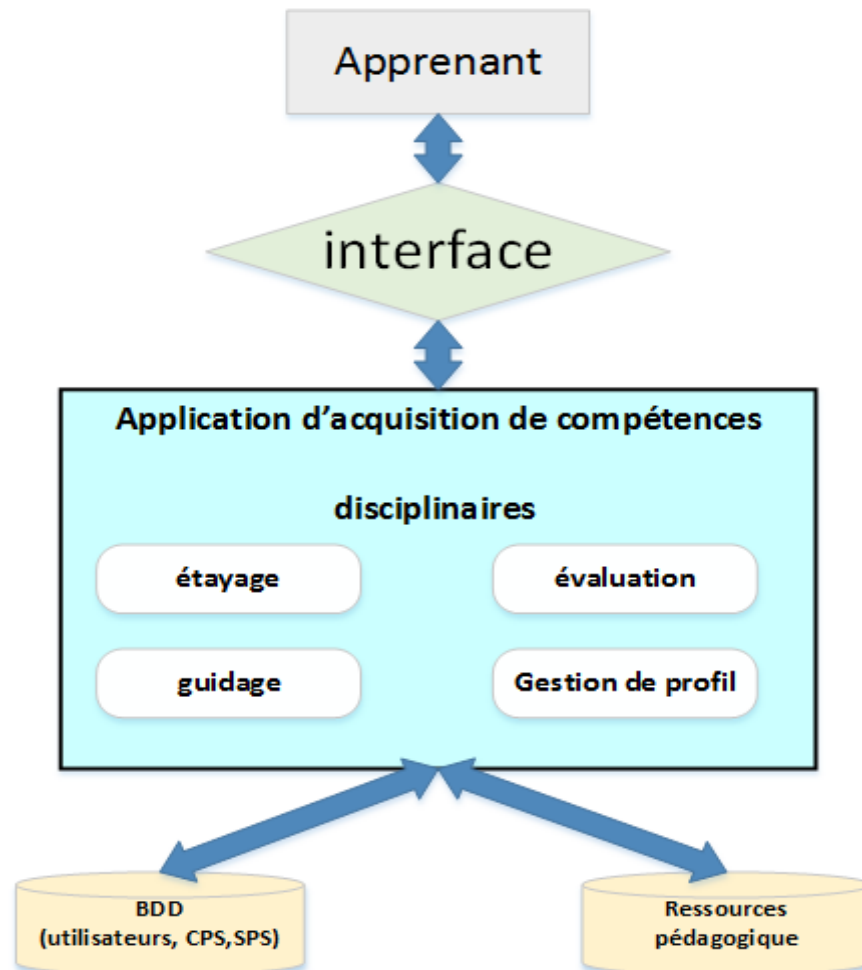


Figure 4.2 architecture logicielle de notre système

### IV.2 Outils et Langages utilisés

#### IV.2.1 Outils de développement

- a. **Serveur MySQL** : MySQL est un serveur de bases de données relationnelles, basé sur le langage de requête SQL, il est implémenté sur un mode client/serveur avec du côté serveur : le serveur MySQL, et du côté client : les différents programmes et bibliothèques. MySQL se caractérise par sa rapidité et sa facilité d'utilisation. Il est offert avec l'outil d'administration de bases de données « PhpMyAdmin ».
- b. **PhpMyAdmin** : L'outil PhpMyAdmin est développé en PHP, il offre une interface graphique pour l'administration de bases de données MySQL via un navigateur web. Les fonctions principales de PhpMyAdmin sont (Cf. figure IV.3) :
  - ❖ Création de nouvelles bases de données ;
  - ❖ Création/Suppression/Modification de tables ;
  - ❖ L'édition, l'ajout et la suppression de champs ;
  - ❖ L'exécution des commandes et des requêtes SQL.

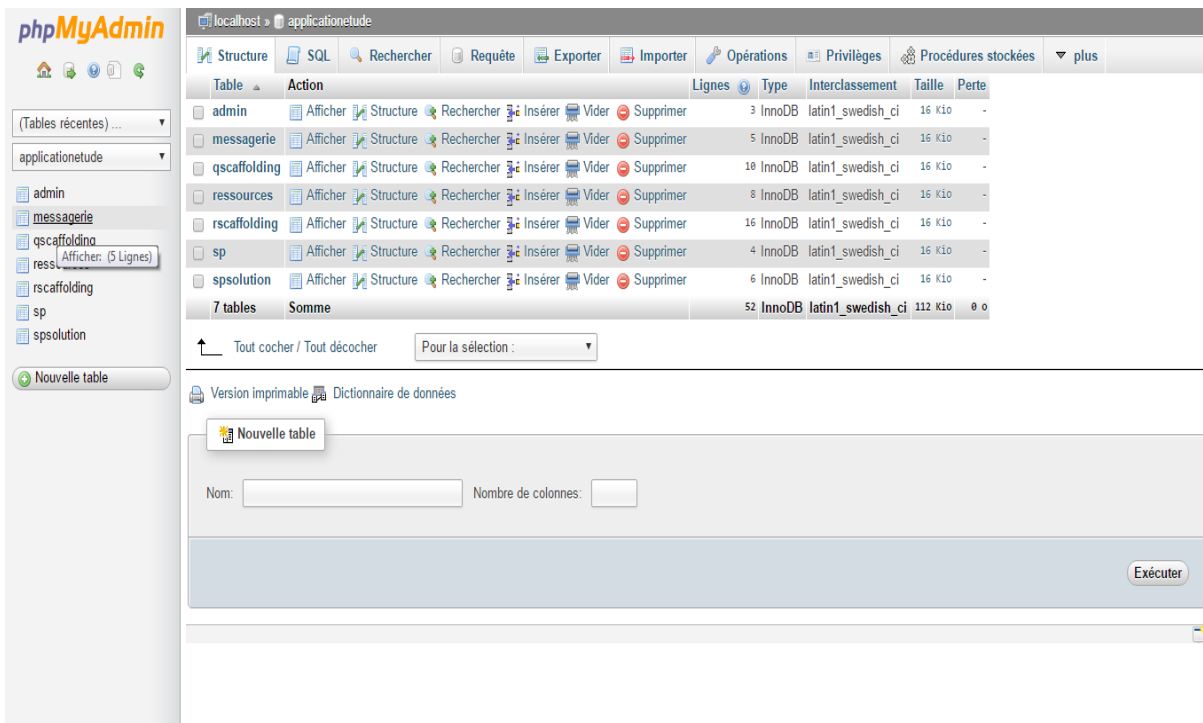


Figure 4.3: Interface de PhpMyAdmin les tables de notre base de données

### IV.3 : Langages utilisés

- a. **JAVA** : Java est un langage de programmation orienté objet développé par **Sun Microsystems** vers la fin 1995. Sa syntaxe est proche de celle du langage C. Java possède une machine virtuelle JVM qui se charge de l'exécution de ses programmes, cette plate- forme lui garantit une excellente portabilité : une fois votre programme créé, il fonctionnera automatiquement sous Windows, Mac, Linux, etc.

Java offre une allocation dynamique de la mémoire grâce au Garbage Collector (ramasse miette). Ce qui fait de lui le langage le plus utilisé par les programmeurs. Il existe plusieurs profilers open source notamment

- Netbeans profiler
- Eclipse Test & Performance Tools Platform
- JBoss Profiler
- Jprof

Pour la réalisation de Notre application nous avons utilisé Netbeans. C'est un environnement de Développement Intégré (EDI) pour Java, mais peut aussi supporter d'autres langages comme C, C++, XML, PHP et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi langage, éditeur graphique d'interfaces et de pages web).

- b. **SQL** : C'est un langage de manipulation de bases de données, il permet notamment :

- ❖ La manipulation de tables : création, suppression et modification de la structure de la table.
- ❖ La manipulation de bases de données : sélection, modification de la structure de la table.
- ❖ La gestion des droits d'accès aux tables : contrôle des données et validation de la modification.

### IV.4 Scenario d'utilisation du système proposé

Ci-dessous, nous donnons un aperçu sur l'utilisation du système à travers deux types de scénarios : scénario enseignant et scénario apprenant.

#### IV.4.1 Scénario de l'enseignant

Ce scénario se fait selon les étapes suivantes :

- a. Pour accéder à son espace (espace enseignant), l'enseignant doit saisir son login et mot de passe.
- b. Après avoir ouvert la session, le système se charge de restaurer le profil de l'enseignant.

## Chapitre 4 : Implémentation de l'application

- c. Le rôle de l'enseignant est d'étayer (aider) les apprenants d'une façon synchrone pendant la résolution des situations problème et d'évaluer les situations problèmes ainsi que les compétences disciplinaires. L'étayage peut se faire à la demande de l'apprenant ou à l'initiative de l'enseignant qui constate des difficultés chez l'apprenant. L'étayage de l'enseignant qui intervient de son propre gré a pour but de cadrer l'apprenant pour qu'il ne s'éloigne pas de l'objectif visé par la situation problème qu'il est en train de résoudre. L'étayage dans notre cas se fait à l'aide d'un dialogue (questions-réponses) entre l'enseignant et l'apprenant.

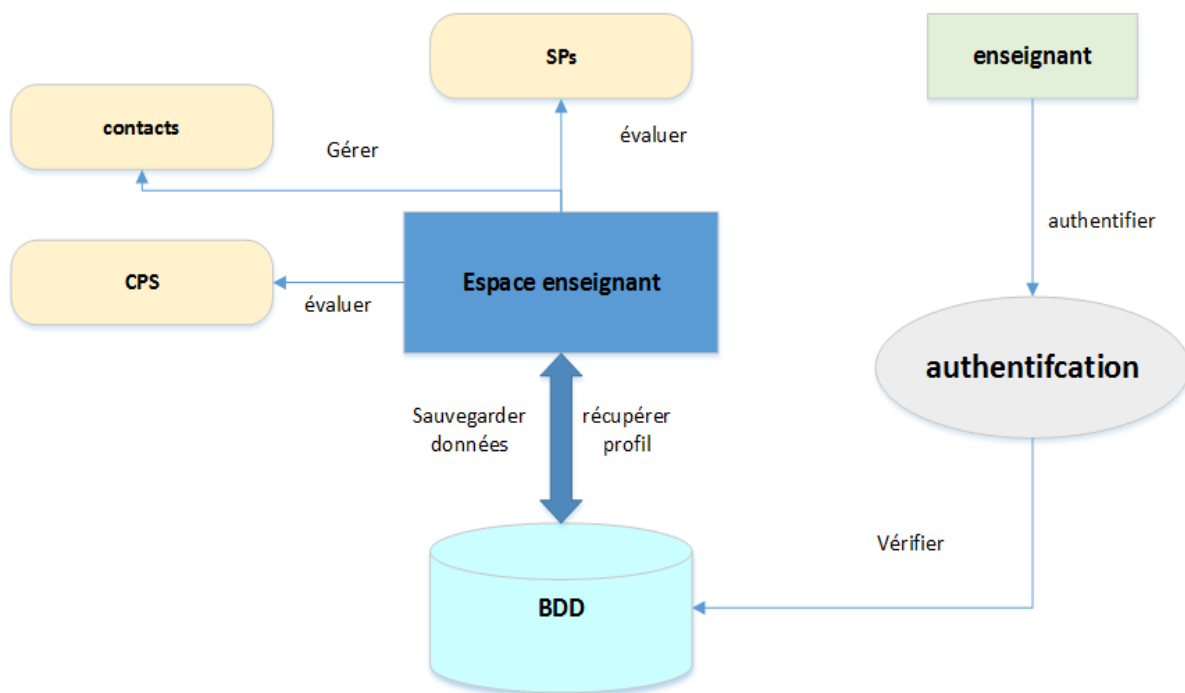


Figure 4.4 : Scénario d'utilisation de l'enseignant

### IV.4.2 : Scénario de l'apprenant

On distingue un nouvel apprenant et un ancien apprenant (Cf. Figure 4.5) :

Quand il s'agit d'un nouvel apprenant, ce dernier doit d'abord s'inscrire. Un formulaire d'inscription est alors présenté à l'apprenant. Ce dernier saisit son identité (le nom, le prénom, le login et le mot de passe,), le système se charge de sauvegarder ces informations dans une table nommée « Apprenant » et de lui afficher un espace de travail et de dialogue. Dans l'*espace de travail*, le système affiche un plan guide initial des compétences que l'apprenant pourra développer. L'apprenant pourra alors choisir une compétence à développer. Pour chaque compétence choisie, le système affiche une liste de situations problème où la compétence doit se développer. L'apprenant doit résoudre toutes ces situations une à une en suivant des étapes de résolution. C'est à travers l'*espace d'étayage* que l'apprenant aura

## Chapitre 4 : Implémentation de l'application

l'occasion de voir les étapes de résolution demandées pour chaque situation problème entamée, de dialoguer avec l'enseignant (lui poser des questions en cas de blocage, de lire les réponses de l'enseignant) et de collaborer avec des camarades.

Quand il s'agit d'un ancien apprenant, ce dernier doit s'authentifier en saisissant son login et son mot de passe pour ouvrir une session. A l'ouverture de la session, le système se charge de restaurer la session sauvegardée. Dans cette étape l'apprenant va continuer là où il s'est arrêté.

Que l'étudiant soit nouveau ou ancien, le système le soutient durant la réalisation des situations problèmes en lui fournissant des consignes sous forme de pistes de remédiation ou de régulation. L'apprenant peut également collaborer avec ses camarades en vue de corriger ses fautes. Il peut aussi consulter des ressources (supports de cours, liens de sites intéressants, etc.).

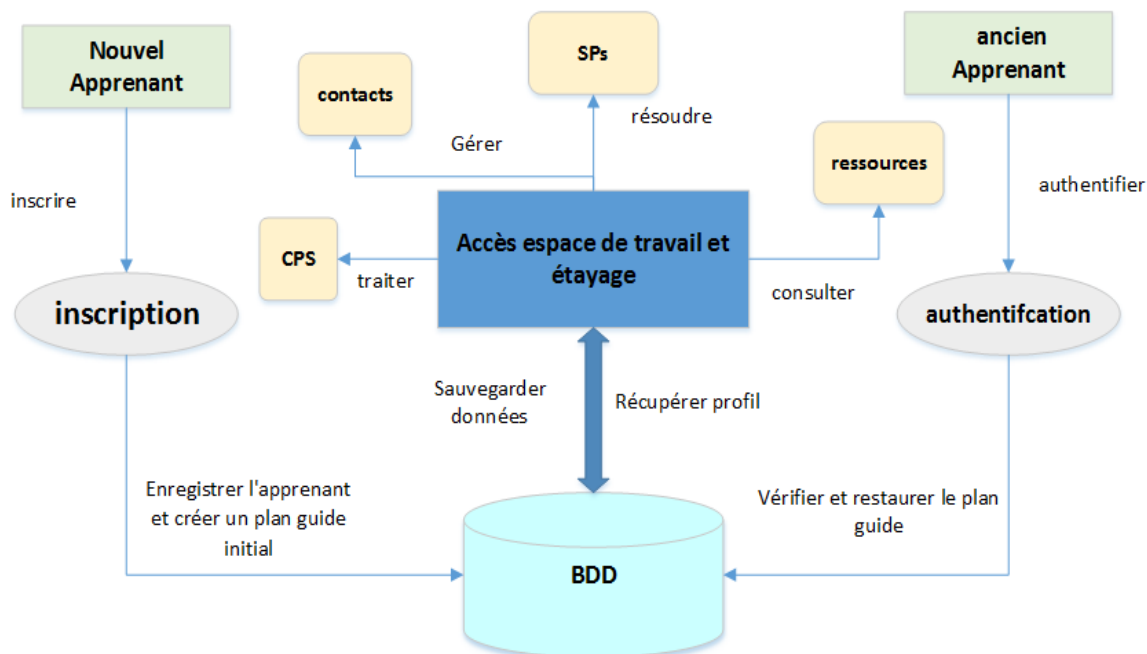


Figure 4.5 : Scénario d'utilisation de l'apprenant

### IV.5 : Quelques interfaces du système

- **Page d'accueil**

Cette page est composée des éléments suivants :

- ✓ Un bouton **A propos** qui permet de se renseigner sur l'application.
- ✓ Un bouton **Quitter** pour fermer l'application.
- ✓ Un bouton **Contacteur** qui permet aux visiteurs de contacter l'administrateur
- ✓ Un lien pour **s'inscrire** pour les nouveaux utilisateurs
- ✓ Un espace pour **s'authentifier** pour les anciens utilisateurs.



Figure 4.6 : page d'accueil

### • Page d'inscription

Elle permet à un nouvel utilisateur, de s'inscrire dans l'application de résolution de situations problèmes. Cette page contient un formulaire d'inscription où l'utilisateur doit fournir ses données personnelles : nom, prénom, login et mot de passe qu'il doit confirmer.

Après que l'apprenant ait rempli tous les champs, il n'aura qu'à cliquer sur le bouton valider, s'il ne veut pas valider, il cliquera sur le bouton annuler. Une fois l'inscription validée, un espace apprenant est affiché.

Cette page est composée des éléments suivants :

- ✓ Un bouton **accueil** qui permet d'afficher la page d'accueil.
- ✓ Un bouton **Quitter** pour fermer l'application.
- ✓ Un lien pour **s'inscrire** pour les nouveaux utilisateurs

apprentissage de compétences disciplinaires

accueil s'inscrire quitter

Bienvenus dans l'environnement d'acquisition de compétences disciplinaires

inscription

nom  
prenom  
login  
password  
confirmer password  
vous etes: apprenant  
Discipline: algorithme  
valider annuler

Figure 4.7 : page d'inscription

### • Espace d'apprentissage

Une fois l'apprenant a saisi son login et son mot de passe, un plan guide sera affiché, l'apprenant n'a qu'à choisir une (ou des) compétences à traiter.

apprentissage de compétences disciplinaires

accueil etat congétif profil contact ressources déconnexion

plan guide bienvenus imad dans votre espace d'apprentissage

permutation
tri
somme de 2 matrices
équation

veuillez choisir une compétence à traiter dans le plan guide

- les compétences affichées sont des compétences non acquises qui non pas de prérequis ou dont les pré-requis sont acquis.
- les compétences non affichées sont celles qui sont acquises ou qui ont au moins un prérequis non acquis.

ces compétences sont annotées comme suit:

- couleur noire= pour toute cp non entamée avec des CPs de bas niveaux non acquis
- couleur bleue= pour toute cp entamée avec au moins une cp de bas niveau non acquis
- couleur verte= pour toute cp entamée avec les cps de bas niveaux acquises
- couleur magenta= pour toute cp entamée qui possède au moins une sp non résolées

Figure 4.8 : espace d'apprentissage

## Chapitre 4 : Implémentation de l'application

Dans cet espace on trouve :

- ✓ Bouton profil qui contient les informations personnelles de l'apprenant connecté avec la possibilité de changer le mot de passe ;
- ✓ Bouton état cognitif qui affiche l'état cognitif de l'apprenant ;
- ✓ Bouton ressource : pour consulter des cours et des liens intéressants sur la discipline ;
- ✓ Bouton déconnexion permet aux apprenant de se déconnecter à tout moment ;
- ✓ Bouton contact : pour poser une question et consulter les messages.
- ✓ Espace plan guide : cet espace est réservé à afficher des compétences à apprendre.

- **Espace de travail**

*apprentissage de compétences disciplinaires*

retour contact ressources profil

**espace de travail**

énoncé de la SP

écrire l'algorithme qui permet la permutation de 2 nombres

votre réponse

critère

**espace scafoolding**

**Etapes à suivre:**

- 1-Décodage
- 2-explication
- 3-validation

**Explicitation**

- 2.1: Préciser ce qui devrait être fait pour atteindre cette solution
- 2.2: Organiser les étapes de la démarche de solution
- 2.3: Identifier toute l'information utile à la démarche
- 2.4: Sélectionner les données pertinentes dans la situation
- 2.5: Appliquer la démarche avec les données sélectionnées

**Evaluation des étapes de la SP**

réponse étape décodage

**Evaluation globale de la SP**

afficher l'évaluation globale

**Questions/Réponses diagnostique**

?-Questions/Réponses apprenant

?-Questions/Réponses enseignant

Figure 4.9 : espace de travail et d'étayage

Cet espace contient les boutons ressources, profil, contact, déconnexion, il est divisé en 2 partie :

## Chapitre 4 : Implémentation de l'application

- ✓ Partie espace de travail qui permet au apprenant de résoudre les situations problèmes.
  - ✓ Partie espace scffolding qui permet au apprenant d'une part de contacter l'enseignant et d'une autre part de consulter les critères, les obstacles et les consignes,
  - ✓ A chaque étape l'apprenant va répondre à la sp en passant par les étapes mentionnées dans l'orientation puis **attendre la réponse de l'enseignant** pour la compléter,
- **Espace d'enseignant**

*apprentissage de compétences disciplinaires*

contact ressources quitter

**espace enseignant**

la liste des SPs non corrigées

user	id_ce	id_SP	etat	corriger	réponse
imad	ce2	sp11	1	N	le mots p...

rafraichir

l'apprenant: imad

id\_sp: sp11

énoncé de la SP

écrire l'algorithme qui permet la permutation de 2 nombre

réponse de l'étape 1

le mots pertinent sont permutation et nombre

évaluation de l'étape

passé a l'étape suivante

refaire cette étape

valider

Figure 4.10 : espace d'enseignant

- Cet espace contient les listes des SPs non corrigées, l'enseignant évaluer les réponses des apprenants, il peut contacter les apprenants

## Chapitre 4 : Implémentation de l'application

- Espace administrateur

Cet espace permet au administrateur de modifier les listes des CPs et SPs et les listes es utilisateurs on va donner un exemple de fenêtre ajouter une CG

The screenshot shows the administrator interface for adding a global competency (CG). The interface is titled "apprentissage de compétences disciplinaires" in a blue header. It contains three main sections:

- Ajouter une compétence globale:** This section has a text input field for "URL CG" containing "somme de 2 matrice" and a dropdown menu for "Degré de maîtrise de CG" set to "faible". There is an "Ajouter CG" button.
- Ajouter des CG prérequis:** This section has a dropdown menu for "choisir les CG prérequis" set to "cg1" and a "rajouter une CG prérequis" button. There is also a "rafraichir" button.
- associer une situation problème:** This section has a dropdown menu for "Choisir au moins une sp" set to "sp1" and a "seuil" input field. There are "rafraichir" and "associer cette sp" buttons. A "ajouter une nouvelle sp" button is also present.

At the bottom of the interface, there is a "fermer" button.

Figure 4.11 : espace administrateur

## Conclusion

Dans le présent chapitre, nous avons présenté l'environnement technique de développement et d'implémentation de notre système. Pour cet effet, nous avons pris le soin d'exposer les étapes de réalisation de l'application en montrant quelques interfaces de l'application. Notre application en montrant quelques interfaces

# Conclusion Générale

## Conclusion générale

Notre travail consistait en la réalisation d'une application permettant le développement de compétences disciplinaires dans des disciplines données. Ce travail rentre dans le cadre d'un travail de recherche publié portant sur le guidage et l'étayage des apprenants dans la logique d'apprentissage par compétences. Une expérimentation dans les disciplines « algorithmique » et « bases de données relationnelles » a été tentée. Le travail réalisé nous a permis :

- D'approfondir nos connaissances théoriques et pratiques acquises durant notre cursus et de toucher à plusieurs domaines de l'informatique comme l'internet, le web, les bases de données relationnelles, UML, etc.
- De découvrir des langages de programmation (comme Java et Javascript) et de modélisation (avec UML) ainsi que des outils de développement (comme Dreamweaver, etc.).

Nous avons pu réaliser certaines fonctionnalités comme :

- Le guidage reposant sur un ordre de maîtrise dicté par les prérequis entre les compétences disciplinaires ainsi que sur la décomposition de compétences en compétences de bas niveaux. L'apprentissage des compétences est réalisé en partant des compétences de bas niveaux dites élémentaires vers les compétences de haut niveau dites globales ou générales.
- L'étayage où nous avons tenté de proposer un processus de dialogue à l'aide de questions réponses entre l'apprenant et ce, dans le but d'aider l'apprenant dans son apprentissage en cas de difficulté.

Néanmoins le travail réalisé est loin d'être opérationnel à cent pour cent. Bien que le processus de guidage fonctionne bien, le processus d'étayage demande une perfection. Comme perspectives nous proposons d'améliorer ce dernier processus ainsi que le processus de collaboration entre apprenants pour avoir une expérimentation réussie.

## Bibliographie

[Benkhadra,2004] :Benkhadra « de la pédagogie par objectifs a la pédagogie de l'intégration» avril 2004

[Berkane, 2015] : Berkane T. & Bouarab-Dahmani F., « Adaptive Guidance for Disciplinary Competencies Self-learning Using an Educational Hypermedia », proceedings of the 14th European Conference on e-Learning (ECEL), University of Hertfordshire Hatfield, UK 2015.

[Bourdat,2009] :Mathilde Bourdat « De l'intérêt des objectifs pédagogiques » 18 mars 2009

[Bruner,1997] : Bruner J. S « Les problèmes de l'école à la lumière de la psychologie culturelle », Col. Psychologie, Retz, Paris, 1997

[De KETELE,2006] :De KETELE J-M., « L'approche par compétences : ses fondements, Bruxelles »: UCL 2006

[Déronne,2012] :Deronne, M. « L'approche par les compétences dans l'enseignement des mathématiques ». Mémoire de master en sciences mathématiques. Université du Mons, France.

[JONNAERT,2009] :Jonnaert, Ph., « Compétences et socioconstructivisme »

Bruxelles : De Boeck.2009

[Mager,1971] : Mager, R.-F. « Comment définir les objectifs pédagogiques », Paris, Gautier-Villars 1971

[Muller, 1997] : Muller P. A., « Modélisation Objets avec UML», Edition Eyrolles, 1997.

[Pelpel,2002] : Pelpel, P « Se former pour enseigner », Paris : Dunod 2002

[PERRENOUD,2000] :Perrenoud, Ph., « L'école saisie par les compétences ». In Bosman, C., Gerard, F.-M. et Roegiers, X. (dir.) « Quel avenir pour les compétences »? Bruxelles : De Boeck, pp. 21-41.

[RICHARD,2001] :Bissonnette, S. et Richard M., . «Comment construire des compétences en classe», Montréal : Chênevière/McGraw-Hill 2001.

[Richer,2014] :Richer, J.J. 2014. « Condition d'une mise en œuvre de la perspective actionnelle en didactique des langues », Recherches et pratiques pédagogiques en langue de spécialité. Cahiers de L'APLIUT, vol. XXXIV n°1, février 2014.

[Tardif,2006] :Jacques Tardif, de l'Université de Sherbrooke, conférence du 27 avril 2006 dans cette université

[Tyler,1949] : Tyler, R.W. 1949, «Basic principles of curriculum and instruction ». Chicago : University of Chicago Press.

[zarifane,2003] :Zarifian, Ph. « A quoi sert le travail ? » Paris : La dispute 2003.