

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE



Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique
Université AMAR TELIDJI Laghouat



**FACULTE DES SCIENCES
DEPARTEMENT DE MATHEMATIQUE ET INFORMATIQUE**

**Mémoire de fin d'études
En vue de l'obtention du Diplôme d'ingénieur d'état en
informatique**

Option : systèmes d'informations avancées

Thème

Conception et implémentation d'une application sous
Web pour la gestion des étudiants du Centre
d'Enseignement Intensif des Langues « CEIL »

Réalisé par :

- Bendahane Mohamed Nadhir.
- Benzahia Radhia.

Encadré par :

Mlle Belabbaci Amel.

Année universitaire : 2011/2012

Remerciement

Nous remercions Dieu de nous avoir accordé des connaissances de la science et de nous avoir aidé à réaliser ce travail.

Au terme de ce modeste travail nous tenons à remercier chaleureusement et respectivement tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste projet de fin d'étude, à notre encadreur M^{elle} BELABBACI Amel.

Nos vifs remerciements vont tous d'abord à M^r GUENDOZ Abdelkader et M^r MOKDAD Hamza qui nous ont orientés vers le bon sens où nous avons pu développer notre travail avec son aide précieuse.

Nous tenons à remercier ainsi les dirigeants du centre C.E.I.L pour leurs compréhension et collaborations.

Nous tenons à remercier tous les enseignants qui nous ont suivi durant notre formation ainsi pour leurs valeureux conseils.

Merci à tous

Dédicaces

Je voudrais dédie ce modeste travail A mes très chères parents qui m'ont tant soutenu et encouragé dans tout les domaines et surtout pour réaliser ce mémoire.

A mes cher grands parents qui m'ont soutenu et encourager.

A mes frères Hamza, Bachir et mes sœurs Zahia, Amel et Naçéra.

A tout la famille Benzahia à Alger et la famille Hachaichi. La famille Benzahia à Aberdine et Benbrika à Laghouat.

A mes oncles et tantes.

A mes cousins et mes cousines.

A mon binôme BENDAHANE Mohamed Nadhir.

A tous mes amies et toute la section D'Informatique 2011-2012.

A tous ceux qui m'aiment.

B. Radhia

Dédicaces

Je Dédie Ce Mémoire :

A mon cher père,

A ma chère mère,

*Pour leurs patiences et leurs encouragements, En leur souhaitant
longue vie,*

A ma Fiancée,

A mon frère Brahim,

A tous mes amis(es),

A toute ma famille,

A mon binôme BENZAHIA Radhia.

*A tous ceux qui m'ont aidé à la réalisation de ce travail,
J'exprime mes sentiments les plus profonds.*

B.Nadhir

LISTE DES FIGURES

Figure 1:	L'évolution du CEIL.....	01
Figure 2:	Représentation de l'évolution du CEIL.....	03
Figure 3:	Diagramme de contexte.....	12
Figure 4:	Diagramme générale de cas d'utilisation.....	13
Figure 5:	Diagramme générale de séquence.....	15
Figure 6:	Diagramme générale de séquence pour le cas de consultation.....	16
Figure 7:	Quelle méthode pour passer de l'expression des besoins au code de l'application?	17
Figure 8:	Schéma montrant le cycle de vie de la méthode agile XP	18
Figure 9:	Diagramme de cas d'utilisation de l'administrateur.....	23
Figure 10:	Diagramme de cas d'utilisation de l'étudiant.....	24
Figure 11:	Diagramme de cas d'utilisation de l'enseignant.....	24
Figure 12:	Diagramme de cas d'utilisation de système.....	25
Figure 13:	Diagramme de cas d'utilisation générale.....	26
Figure 14:	Diagramme de séquence d'authentification	27
Figure 15:	Diagramme de séquence de saisie d'annonce.....	28
Figure 16:	Diagramme de séquence de saisie de note.....	29
Figure 17:	Diagramme de séquence de saisie d'absence.....	30
Figure 18:	Diagramme de MAJ enseignant « Modification ».....	31
Figure 19:	Diagramme de MAJ enseignant « Ajout »	32
Figure 20:	Diagramme de MAJ enseignant « Suppression ».....	33
Figure 21:	Diagramme de MAJ étudiant « Modification ».....	34
Figure 22:	Diagramme de MAJ étudiant « Ajout ».....	35
Figure 23:	Diagramme de MAJ étudiant « Suppression ».....	36
Figure 24:	Diagramme de séquence de gestion de l'emploi du temps.....	37
Figure 25:	Diagramme de séquence de test.....	38
Figure 26:	Diagramme de séquence de consultation.....	39
Figure 27:	Diagramme de séquence de changement d'état.....	40
Figure 28:	Diagramme d'activité de saisie d'une annonce.....	41
Figure 29:	Diagramme d'activité de saisie d'une note.....	42
Figure 30:	Diagramme d'activité de saisie d'une absence.....	43

Figure 31:	Diagramme d'activité de saisie de l'emploi du temps.....	44
Figure 32:	Diagramme d'activité de MAJ étudiant « Modification ».....	45
Figure 33:	Diagramme d'activité de MAJ étudiant « Ajout ».....	46
Figure 34:	Diagramme d'activité de MAJ étudiant « Suppression ».....	47
Figure 35:	Diagramme d'activité de MAJ enseignant « Modification »	48
Figure 36:	Diagramme d'activité de MAJ enseignant « Ajout »	49
Figure 37:	Diagramme d'activité de MAJ enseignant « Suppression »	50
Figure 38:	Diagramme d'activité du test	51
Figure 39:	Diagramme d'activité de consultation.....	52
Figure 40:	Diagramme d'activité de l'IHM de l'administrateur	53
Figure 41:	Diagramme d'états transition.....	54
Figure 42:	Diagramme de déploiement.....	55
Figure 43:	Diagramme de classe	57
Figure 44:	Le modèle MVC	60
Figure 45:	L'architecture de l'application par MVC	61
Figure 46:	L'interface d'EdrawMax version 6.5.3.....	65
Figure 47:	Capture d'écran montrant la page d'accueil.....	66
Figure 48:	Capture d'écran montrant la page où consulter les annonces du centre.....	67
Figure 49:	Capture d'écran montrant une annonce.....	68
Figure 50:	Capture d'écran montrant comment l'administrateur accède à son page.....	69
Figure 51:	Capture d'écran montrant la page d'authentification de l'administrateur.....	69
Figure 52:	Capture d'écran montrant la page de l'administrateur.....	70
Figure 53:	Capture d'écran montrant la page de suivi pédagogique.....	70
Figure 54:	Capture d'écran montrant la page de MAJ d'étudiant.....	71
Figure 55:	Capture d'écran montrant la page d'ajout d'étudiant	71
Figure 56:	Capture d'écran montrant la page de modification d'étudiant.....	72
Figure 57:	Capture d'écran montrant la page de suppression d'étudiant	72
Figure 58:	Capture d'écran montrant comment accéder à la page d'annonce.....	73
Figure 59:	Capture d'écran montrant la page où créer une annonce.....	73
Figure 60:	Capture d'écran montrant la page d'ajout des groupes.....	74
Figure 61:	Capture d'écran montrant la page de modification des groupes.....	74
Figure 62:	Capture d'écran montrant la page de création d'un emploi du temps.....	75

Figure 63: Capture d'écran montrant la déconnexion de l'administrateur.....	75
Figure 64: Capture d'écran montrant la page d'accueil après la déconnexion de l'admin.....	76

LISTE DES TABLES

Tableau 1: Représentation de l'évolution du CEIL	03
---	----

SOMMAIRE

Sommaire

Chapitre 1 : Introduction générale

1	Présentation du centre.....	01
2	Evolution du centre.....	03
3	Problématique et objectifs du projet.....	04
4	Organisation du mémoire.....	05

Chapitre 2 : Conception préliminaire

1	Introduction.....	06
2	Méthodes et techniques de travail.....	06
3	Cycle de vie.....	07
4	La méthode agile XP	17
5	Conclusion.....	22

Chapitre 3: Conception et modélisation UML

1	Introduction.....	23
2	Le diagramme des cas d'utilisation.....	23
3	Les diagrammes de séquence.....	27
4	Les diagrammes d'activité	41
5	Le diagramme d'états transition	53
6	Le diagramme de déploiement.....	54
7	Le diagramme de classe.....	56
8	Le modèle relationnel.....	58
9	Conclusion.....	59

Chapitre 4: L'implémentation du système

1	Introduction.....	60
2	Environnement de développement.....	60
3	Visualisation des pages principales de l'application	66
4	Conclusion.....	77

Chapitre 5 : Conclusion générale

1	Conclusion générale et perspectives.....	78
---	--	----

Bibliographie & Webgraphie

Glossaire

Annexe

CHAPITRE I

Introduction générale

Dans ce chapitre :

- Présentation du centre
- Evolution du centre
- Problématique et objectifs du projet
- Organisation du mémoire

CHAPITRE II

Conception préliminaire

Dans ce chapitre :

- Introduction
- Méthodes et techniques de travail
- Cycle de vie
- La méthode agile XP
- Conclusion

CHAPITRE III

Conception et modélisation UML

Dans ce chapitre :

- ✦ Introduction
- ✦ Diagramme de cas d'utilisation
- ✦ Diagrammes de séquence
- ✦ Diagrammes d'activité
- ✦ Diagramme d'états transition
- ✦ Diagramme de déploiement
- ✦ Diagramme de classe
- ✦ Le modèle relationnel
- ✦ Conclusion

CHAPITRE IV

L'implémentation du système

Dans ce chapitre :

- ✦ Introduction
- ✦ Environnements de développement
- ✦ Visualisation des pages principales de l'application
- ✦ Conclusion

CHAPITRE V

Conclusion générale

Dans ce chapitre :

- ✦ Conclusion
- ✦ Perspectives

1- Présentation du centre :

Face au taux croissant d'échec des étudiants de graduation, plusieurs enquêtes ont été conduites par les autorités algériennes et menées par des experts sur le terrain dans diverses universités algériennes. Ce recul accuse en majeure partie la régression du niveau en langues étrangères et plus particulièrement en langue française, pour cette raison le Centre d'Enseignement Intensif des Langues est apparu.

Le Ministère de l'Enseignement Supérieur et de la Recherche Scientifique a sollicité la Coopération Française en Algérie pour le renforcement du dispositif des Centres d'Enseignement Intensif des Langues dans toute l'Algérie.

Le CEIL est sous la tutelle du MESRS avec une participation active des partenaires français (FSP) mais son autorité directe reste son université d'origine en collaboration avec le RACEIL (Réseau Algérien des Centres d'Enseignement Intensif des Langues).

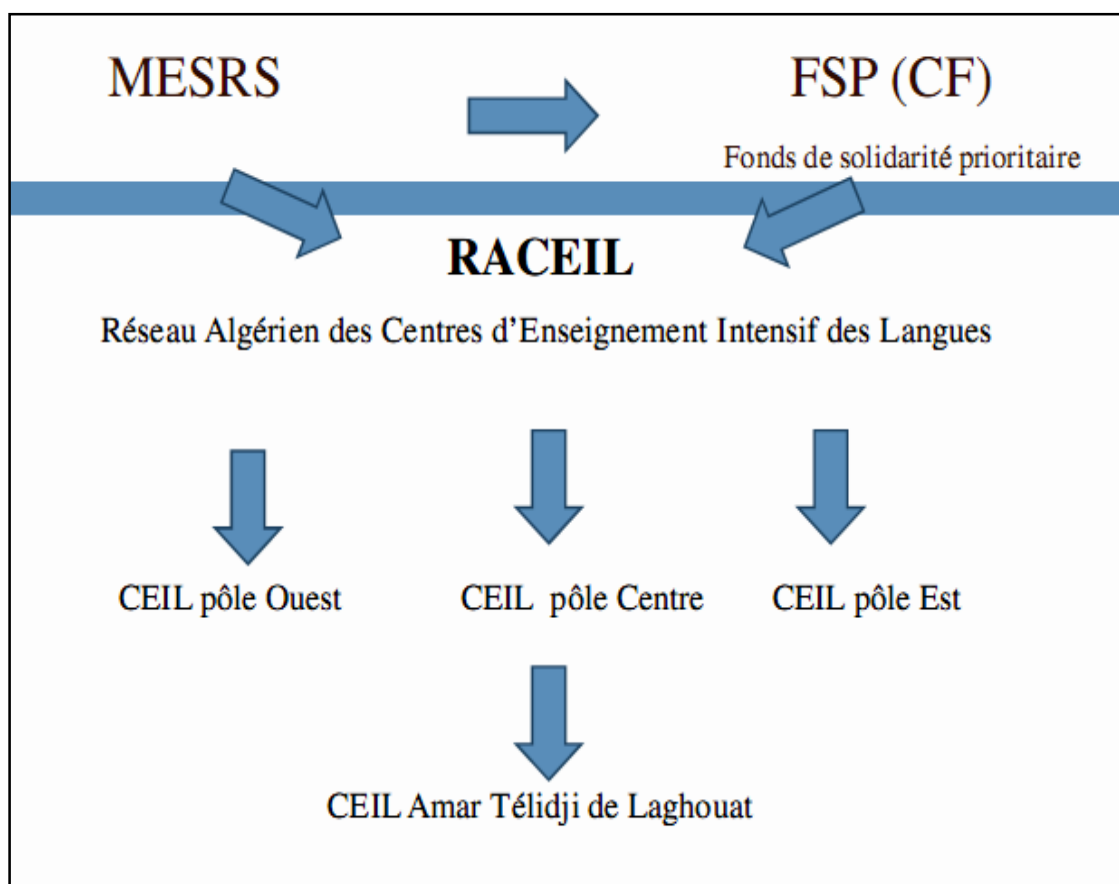


Figure 1: L'évolution du CEIL.

Pour certains centres, il s'agit d'une redynamisation, pour d'autres c'est l'émergence.

Le partenaire C.I.R.E.F.E (Le Centre International Rennais d'Étude de Français pour Étrangers), a pour mission la correction de déficit constaté dans les recours aux langues étrangères, surtout pour le sud algérien. Le Centre tend vers une amélioration de la qualité du service dans l'apprentissage des langues et est devenu très important dans la vie sociale et économique.

Le CEIL de Laghouat est le fruit d'une convention signée par le Ministère de l'Enseignement Supérieur et de la Recherche Scientifique pour la partie algérienne et l'Ambassade de France. Une proposition de jumelage entre le CEIL de Laghouat et le CIREFE de Rennes 2 a été formulée.

Son objectif est d'acquérir la maîtrise des structures fondamentales de la langue orale et écrite pouvoir s'exprimer et prendre confiance en soi à l'oral et à l'écrite.

Cadre Européen Commun de Référence pour les Langues (CECR):

Les méthodes utilisées par les enseignants du Centre s'inspirent du CECR. Un effort supplémentaire est fourni par l'équipe pédagogique du Centre afin de rendre plus accessible aux apprenants, les enseignements proposés dans les livres conçus dans un premier temps pour un public occidental.

Le centre offre la formation aux :

- Etudiants, enseignants, agents administratifs et bibliothécaires.

Langues proposées à l'enseignement sont:

- La langue française, anglaise, italienne et la langue espagnole.

Le Centre compte aujourd'hui onze enseignants : Deux (02) enseignants titulaires à l'université, sept (07) vacataires dont l'organisme de rattachement d'origine est le MEN (Ministère de l'Education Nationale) : Six (6) enseignants pour la langue française et cinq (5) enseignants pour la langue anglaise.

Perspectives:

Laghouat est une ville qui compte dans son sillage Hassi R'mel, une zone industrielle où sont implantées des entreprises étrangères, la demande d'encadrement du personnel est importante.

Même si le Centre n'a proposé aucune offre de formation à ce public, cela n'est pas exclu pour les années à venir.

2- Evolution du centre :

Depuis son ouverture au sein de l'université de laghouat au cour de l'année universitaire (2008-2009), le centre a suit une évolution importante selon les statistiques que montre le tableau suivant :

Année	Français	Anglais	Italien	Espagnol
2008-2009	87	95	-	-
2009-2010	284	161	-	-
2010-2011	130	79	40	40
2011-2012	101	225	-	48

Tableau 1: Représentation de l'évolution du CEIL.

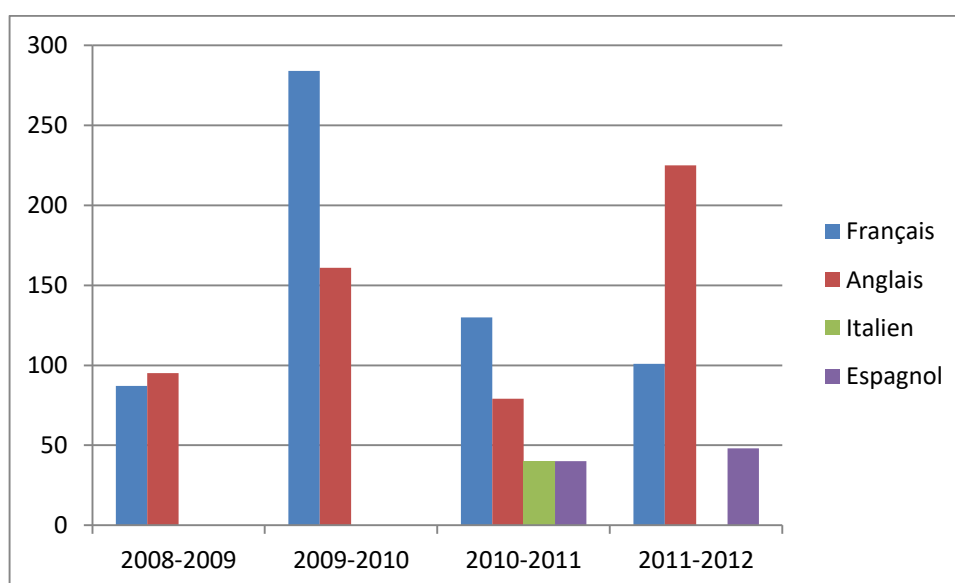


Figure 2 : Représentation graphique de l'évolution du CEIL.

On peut voir que le nombre d'inscrit a diminué en 2010 parce que l'administration du centre a limité les places à fin d'assurer une meilleure formation pour les étudiants. Si on veut faire une étude comparative, on peut dire que le nombre d'étudiants était de 182 en 2008-2009, et après 2 ans, il a atteint 374 étudiants inscrits, dans l'année en cours nous remarquons toujours que le nombre des étudiants augmente.

Difficultés rencontrées:

Pour différentes raisons et à cause de, l'incompatibilité des emplois du temps, certains étudiants n'ont pas pu rejoindre leurs groupes. Le nombre des étudiants a donc été réduit. Même si nous comptons plus d'enseignants par rapport à l'année précédente, l'effectif reste insuffisant pour couvrir toutes les demandes [1].

3- Problématique et objectifs du projet :

Depuis l'ouverture du centre au sein de l'université de Laghouat au cours de l'année (2008_2009), il suit une évolution importante, alors il a pris en charge le besoin d'ajouter des langues et vu d'améliorer la gestion des groupes en séparant les étudiants des enseignants et profiter des moments des vacances pour accélérer les cours pour les enseignants et prendre en charge l'horaire de chaque catégorie (Enseignants, Etudiants).

Donc le nombre d'inscriptions augmente ainsi que le nombre des langues ce qui nécessite une gestion plus appropriée et plus performante pour faciliter le travail et minimiser le temps de chaque tâche.

Cette évolution a présenté des problèmes de gestion du personnel et des étudiants, dans ce cadre, le centre a fait appel à son besoin de développer une application pour :

- Rendre la tâche de gestion des étudiants plus pratique et facile.
- Rendre l'accès aux informations sur les étudiants et les enseignants plus rapides selon le critère voulu.
- Permettre à l'administrateur d'être à jour et d'avoir une vision globale et générale sur le personnel attaché au centre.

Enfin, notre application a été conçue dynamiquement. On rappelle qu'il existe deux types de Site Web informatique selon le besoin : statique et dynamique.

1. Les applications statiques : ce sont des applications réalisés uniquement à l'aide des langages HTML (Hyper Text Markup Language) et CSS (Cascading Style Sheets). Elles fonctionnent très bien mais leur contenu ne peut pas être mis à jour automatiquement : il faut que le propriétaire de l'application modifie le code source pour y ajouter des nouveautés.

2. Les applications dynamiques : plus complexes, elles utilisent d'autres langages en plus de HTML (Hyper Text Markup Language) et CSS (Cascading Style Sheets), tels que PHP et MySQL. Le contenu de ces applications web est dit "dynamique" parce qu'il peut changer sans l'intervention du développeur [18].

4. Organisation du mémoire :

Notre travail s'inscrit dans le cadre de l'informatique de gestion. La préoccupation majeure qui nous a propulsés est de combattre la lourdeur sur la gestion des étudiants en y introduisant les avantages d'une gestion informatisée. Il nous est nécessaire de subdiviser l'étude en trois chapitres hormis l'introduction et la conclusion générale suivi d'Annexe :

- Le premier chapitre : on donne une conception préliminaire de notre système, nous montrons les méthodes et les techniques utilisées ainsi que le cycle de vie du logiciel.
- Le deuxième chapitre : on donne la conception et la modélisation de notre système, à l'aide du langage UML.
- Le troisième chapitre : la mise en œuvre de l'application, les outils utilisés dans l'élaboration du logiciel et des interfaces de l'application.

L'annexe présente le questionnaire proposé pour la collecte des besoins.

1- Introduction :

La difficulté de la conception des systèmes d'information nécessite une démarche de modélisation qui consiste par la suite à apporter des solutions techniques aux descriptions définies lors de l'analyse. Cette démarche n'est en fait qu'un processus de définition de la future application informatique désirée, afin qu'elle soit cohérente et qu'elle répond aux mieux aux exigences et aux besoins de l'utilisateur.

Et c'est la raison pour laquelle on a opté pour une démarche agile basée sur UML.

2- Méthodes et techniques de travail :

Pour l'élaboration de tout travail qui se veut être scientifique, on doit avoir une méthode et des techniques.

2-1. Méthode :

Selon le Disco Encarta, le mot « Méthode » signifie un ensemble des principes théoriques et pratiques sur lesquels se fondent l'application ou l'enseignement d'un art ou d'une science [2].

Pour notre travail on a choisi la méthode XP, qui appartient au paradigme des méthodes agiles.

2-2. Techniques :

Une technique est un rassemblement des procédés propres à un art [3].

Elle est définie comme étant comme un processus concret qui permet à collecter les informations nécessaires, les analyser afin de tirer une conclusion [4].

La méthode seule ne suffit pas pour atteindre le but, il faut toujours l'adjoindre aux techniques. C'est pourquoi dans le cadre de notre travail nous avons les techniques suivantes :

a. L'interview libre :

L'interview libre est un procédé au cours du quelle le premier (inter viveur), pose des nombreuses questions, non structurée à l'avance; c'est à dire une interrogation orale d'une personne à une autre.

Elle nous aide de bien avoir les informations au sein du centre.

L'interview consistant en un jeu de question - réponse, cette technique nous a permis de poser des questions auprès du personnel du centre. Afin d'avoir les réponses concernant notre étude pour une vue claire et nette sur notre domaine d'étude, et pour cela nous avons crée un questionnaire pour nous aider à collecter les informations nécessaires à notre travail.

b. La technique documentaire :

Elle nous a permis à l'assemblage des notes relatives au sujet et de documents ainsi que des ouvrages nécessaires afin de mieux cerner le contour de notre travail.

Elle nous a facilité la collecte et la consultation des différents documents et ouvrages relatifs à notre étude.

c. La technique d'observation :

Elle nous a permis de s'approcher du terrain de travail pour appréhender ce qui se fait et comment cela se fait [5].

3- Le cycle de vie :

Pour notre projet nous avons choisi le cycle de vie en cascade car c'est le mieux adapté à notre projet, il est :

- Simple et facile à utiliser et à comprendre.
- Un procédé structuré pour une équipe inexpérimentée.
- Adapté aux projets dont les besoins sont parfaitement connus...ou qu'il est possible de figer.
- Adapté aux projets très courts.
- Idéal pour la gestion et le suivi de projets.

- Fonctionne très bien quand la qualité est plus importante que les coûts et les délais [6], [7].

Nous structurons le cycle de vie du développement en 4 phases essentielles :

- Analyse.
- Conception.
- Réalisation (implémentation) et déploiement.
- Test et installation.

Nous utilisons le langage unifié de modélisation (**UML : Unified Modeling Language**) que nous présentons ci-dessous.

Ce projet comprend les fonctionnalités d'un site web non-agrégateur. Il se charge des fonctionnalités suivantes :

- **La gestion d'utilisateurs** : leurs droits d'accès, inscription, authentification, ...etc.
- **La gestion d'annonces et de l'emploi du temps**: conception de la base de données, création, suppression et modification des annonces et de l'emploi, etc.
- **La gestion des relations utilisateurs / annonces.**
- **D'autres fonctionnalités que nous détaillerons plus loin.**

3-1. Analyse :

Dans le cycle de vie d'un système, la phase d'analyse a pour objectif d'identifier les acteurs qui interagissent avec le système et de spécifier les besoins fonctionnels du système. Cette phase sert aussi à déterminer, pour chaque acteur, ses droits d'accès aux fonctionnalités du système.

Tout d'abord, nous commençons par identifier les acteurs interagissant avec notre site web. Puis, un diagramme de contexte est déduit de ces acteurs. Ensuite, nous déterminons les cas d'utilisations (fonctionnalités) pour chacun de ces acteurs. Une description textuelle de ces cas d'utilisation, ainsi que des diagrammes de cas d'utilisation sont présentées. Après, des diagrammes de séquence sont créés pour détailler chacun des cas d'utilisation. Enfin, le modèle du domaine du système est construit à l'aide de diagramme de classe [8].

3-1.1. Les besoins des utilisateurs :

Dans le cadre de la préparation de notre projet de fin d'études la collecte des besoins s'est effectuée par des interviews au niveau du centre des langues CEIL.

A travers la collecte des besoins nous avons présenté le système d'une manière générale par une description organisationnelle du système.

1) Exigences fonctionnelles :

Le site web devra regrouper toutes les fonctionnalités nécessaires de recherche, de consultation, d'inscription et de test ...etc.

1- Consultation :

- **Administrateur:** Permet à l'administrateur de consulter l'archive, l'annonce...etc.
- **Enseignant :** Permet à l'enseignant de consulter : l'annonce et l'emploi du temps.
- **Etudiant :** Permet à l'étudiant de consulter : son compte, l'annonce, l'emploi du temps et la note.

2- Afficher annonce :

Permet à l'administrateur d'afficher les annonces du centre dans la page d'accueil.

3- Gestion du groupe :

Permet à l'administrateur d'ajouter une nouvelle langue et/ou un niveau et/ou un groupe.

4- Inscription :

L'administrateur fait l'inscription des étudiants et des enseignants.

5- Test :

- **L'étudiant :** fait le test.
- **L'enseignant :** corrige le test.
- **L'administrateur :** affiche les résultats du test.

6- Suivi pédagogique :

L'administrateur fait :

- MAJ d'étudiant.
- MAJ d'enseignant.
- Saisie de :
 - ✓ La note.
 - ✓ L'annonce.
 - ✓ L'absence.

7- MAJ d'emploi du temps :

L'administrateur fait cette tâche.

2) Exigences non fonctionnelles :

Pour mieux sécurisé, amélioré et augmenté la satisfaction et la fidélisation des utilisateurs (soit : administrateur, étudiant ou enseignant) de notre Site Web, il est important de répondre à l'exigence de qualité suivante :

- ✓ L'authentification offre pour l'utilisateur un nombre de trois essayé (login 3 ESSAIS).

3-1.2 Les Acteurs :

Un acteur représente un rôle joué par une entité (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié. Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données [8].

Identification des acteurs :

Pour le cas de notre site web de centre d'enseignement intensif des langues, nous pouvons distinguer les acteurs (ou les rôles) suivants :

- Administrateur.
- Etudiant.
- Enseignant.

Diagramme de contexte :

Un diagramme de contexte est, souvent, créé au début de la phase d'analyse pour donner une idée globale du système à réaliser. Il ne donne pas des détails du système, mais il le représente comme une boîte noire. Il montre, essentiellement, l'environnement et l'entourage du système [8].

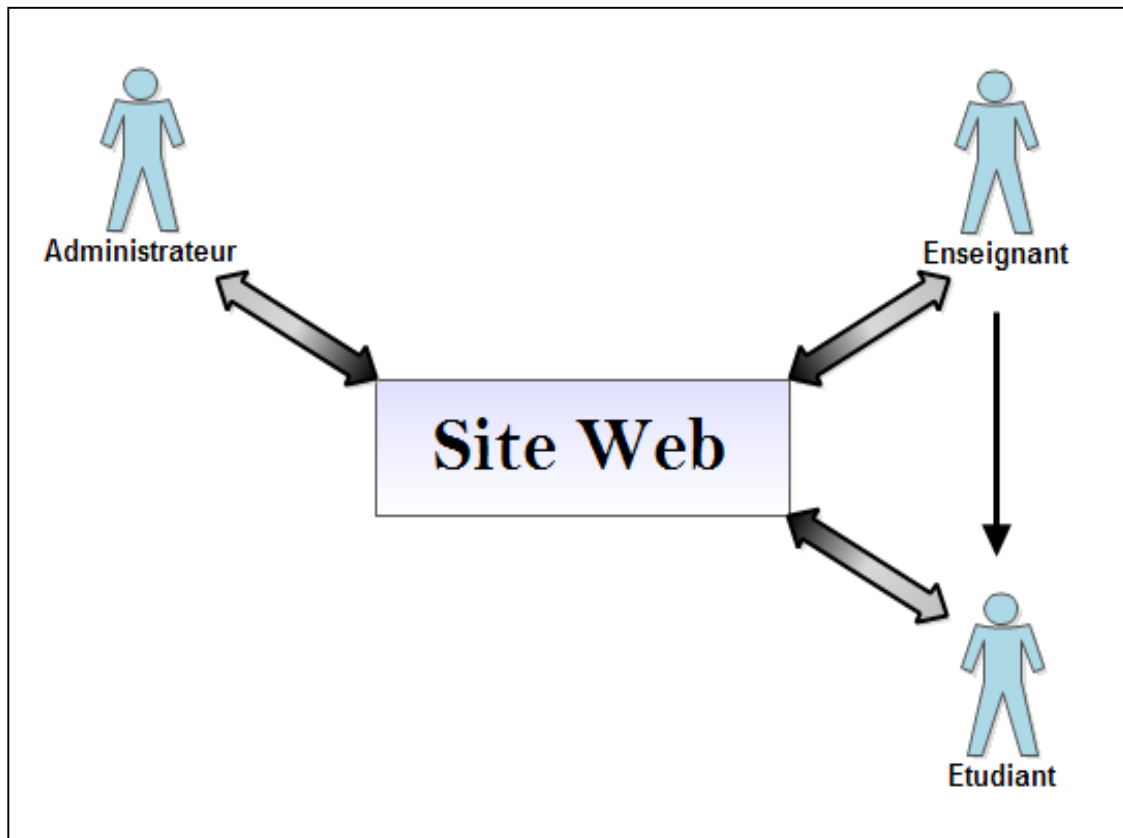


Figure 3 : Diagramme de contexte

3-1.3 Les cas d'utilisation :

Les cas d'utilisation sont les fonctionnalités (interfaces) par lesquelles, le système communique avec ses acteurs. Chaque acteur peut accéder à un sous-ensemble des cas d'utilisation du système [8].

Diagramme général de cas d'utilisation :

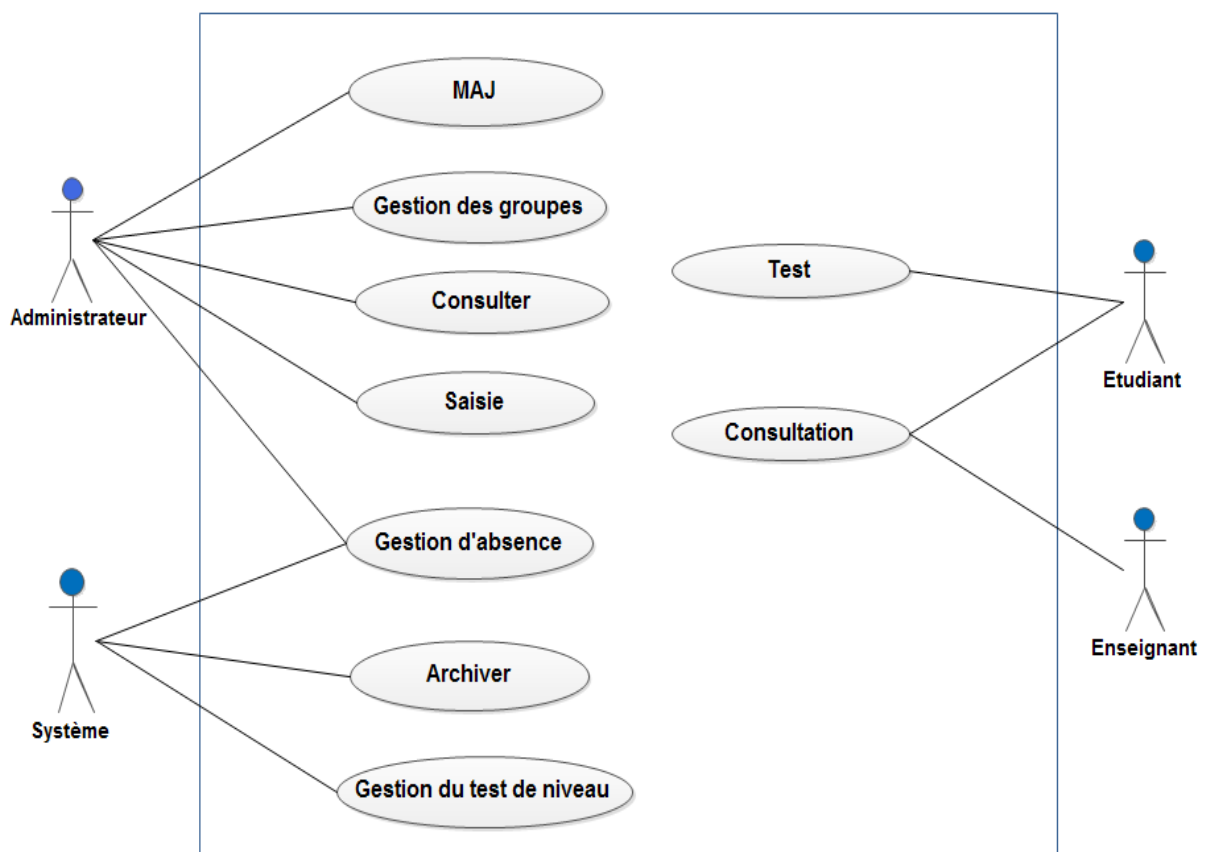


Figure 4: Diagramme générale de cas d'utilisation.

Pour choisir et spécifier les cas d'utilisation de notre site web, nous nous sommes basés sur les éléments suivants :

- Notre vision personnelle sur les besoins exprimés par l'administrateur du centre pour bien présenté les fonctionnalités du Site.

- Analyse et consultation des sites web des autres centres des langues.
exemple :
 - **Formation continue en langues** Université Rennes 2 - Haute Bretagne.
 - **Centre de formation des langues Tunisien CELA.**
- Les besoins exprimés par la directrice du centre.

3-1.4 Diagrammes de séquence :

Les diagrammes de séquence servent à décrire les scénarios nominaux des cas d'utilisation d'un système. Ils fournissent des détails que les diagrammes de cas d'utilisation ne puissent pas les fournir. Ils montrent les interactions entre le système et ses acteurs pour chaque cas d'utilisation [8].

Diagrammes générales de séquence :

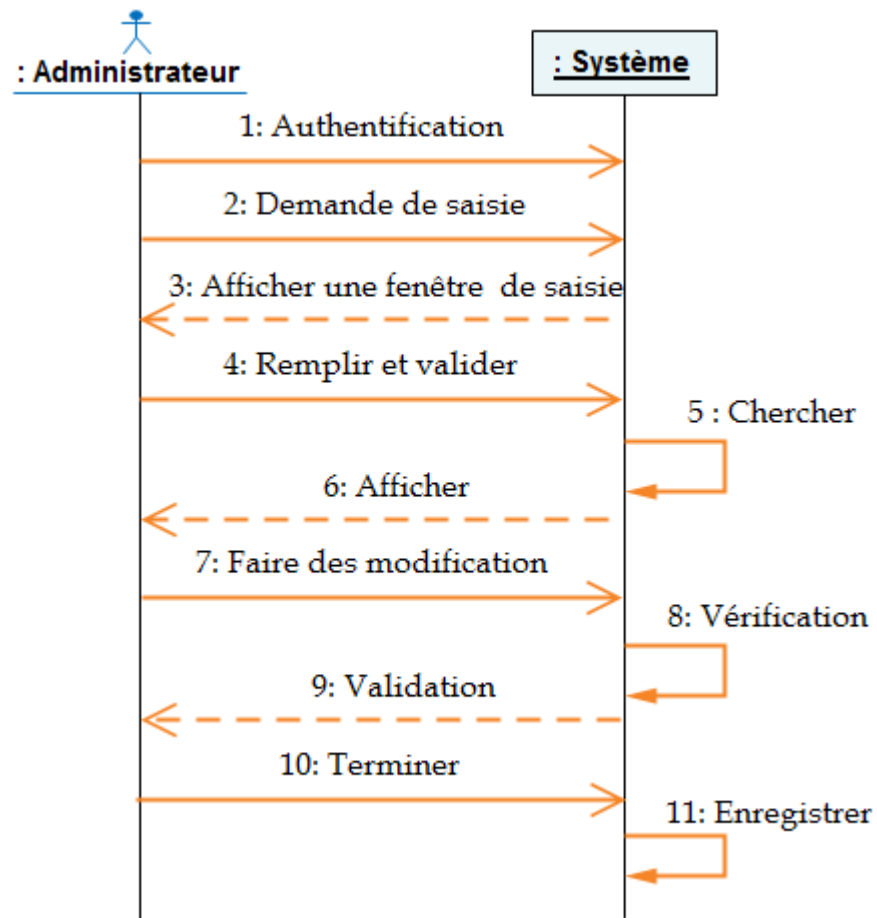


Figure 5: Diagramme générale de séquence pour l'Administrateur.

L'utilisateur est soit : un étudiant, un enseignant ou l'administrateur.

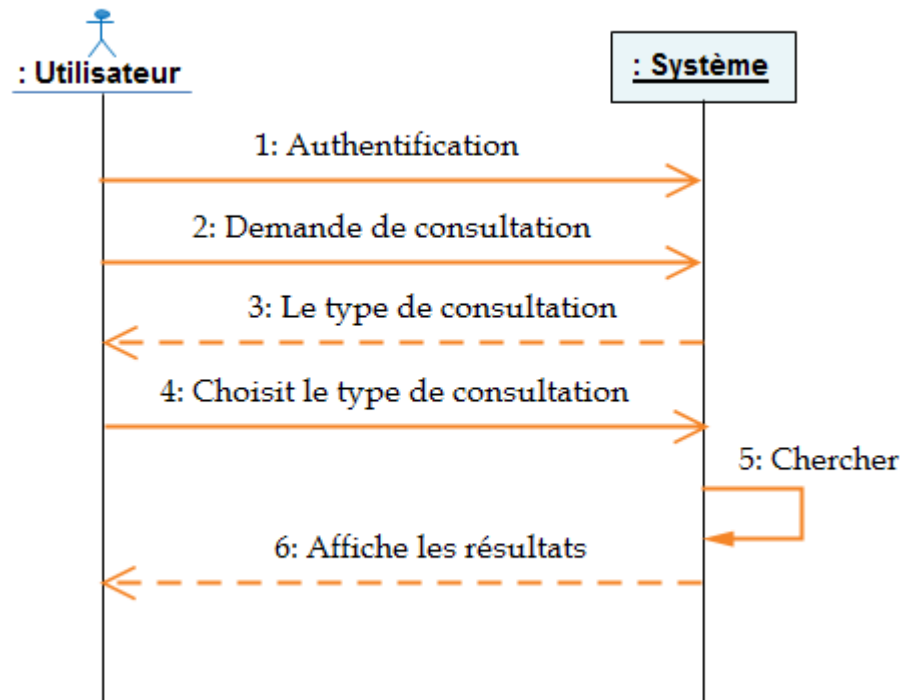


Figure 6: Diagramme générale de séquence pour le cas de consultation.

4- La méthode agile XP :

La problématique que pose la mise en œuvre d'UML est simple : comment passer de l'expression des besoins des utilisateurs au code de l'application ? Autrement dit: « nous avons une bonne idée de ce que notre application doit faire, des fonctionnalités attendues par les futurs utilisateurs. Comment obtenir le plus efficacement possible un code informatique opérationnel, complet, testé, et qui répond parfaitement au besoin » [9].

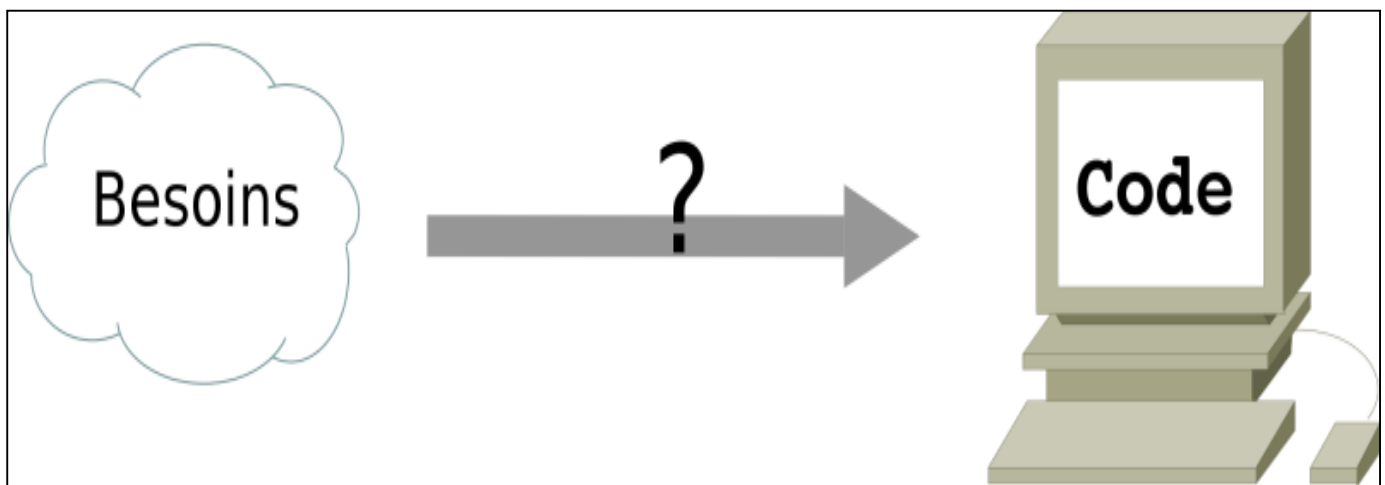


Figure 7: Quelle méthode pour passer de l'expression des besoins au code de l'application? [9]

UML ne propose pas une démarche de modélisation explicitant et encadrant toutes les étapes d'un projet, de la compréhension des besoins à la production du code de l'application. Une méthode se doit de définir une séquence d'étapes, partiellement ordonnées, dont l'objectif est de produire un logiciel de qualité qui répond aux besoins des utilisateurs dans des temps et des coûts prévisibles [9].

Nous allons utiliser une démarche de modélisation Agile.

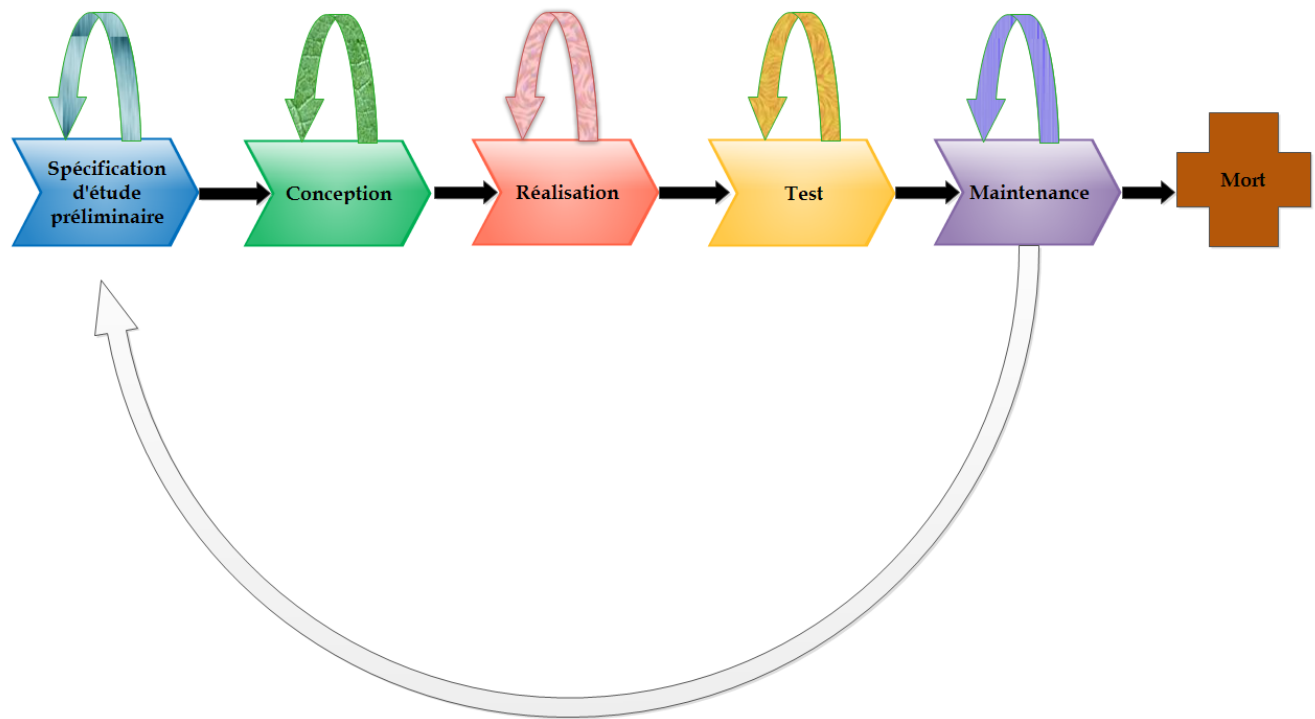


Figure 8: Schéma montrant le cycle de vie de la méthode agile XP.

Pourquoi une méthodologie Agile ?

Pour les caractéristiques suivantes :

- ✓ C'est un modèle incrémental et itératif.
- ✓ Livraison petite et fréquente.
- ✓ Accent sur le code et moins sur la documentation.
- ✓ Convient aux projets de petites et moyennes tailles [6].

Présentation de la démarche informatique XP :

Le processus que nous avons opté de suivre pour le développement d'applications web se situe à mi-chemin entre **UP** (Unified **P**rocess), un cadre général très complet de processus de développement, et les méthodes agiles en vogue actuellement, telles que **XP** (eXtrême Programming). Il s'inspire également des bonnes pratiques prônées par les tenants de la modélisation agile (Agile Modeling).

Notre choix est orienté vers la méthode **XP**, du fait de son approche nouvelle et originale. Notre projet est basé sur un processus de développement bien défini qui prend sa source de la détermination de besoins fonctionnels attendus du système jusqu'à la conception et le codage final.

Le processus XP :

L'**UP** est une trame de meilleures pratiques de développement, il doit être utilisé comme un guide pour réaliser un projet et non comme l'arme ultime et universelle de développement. Ainsi nous optons pour **XP** dans le cadre de ce travail vu son agilité et ses souples principes [10].

L'eXtreme Programming (**XP**) est un ensemble de pratiques qui couvre une grande partie des activités de la réalisation d'un logiciel, de la programmation proprement dite à la planification du projet, en passant par l'organisation de l'équipe de développement et les échanges avec le client [11].

Le **XP** a été mis en œuvre pour la première fois en 1996 sur le projet C3, Chrysler Comprehensive Compensation System. Les pères de la méthode, Ward Cunningham et Kent Beck définissent eXtrême Programming comme « *une méthode basée sur des pratiques qui sont autant des boutons de contrôle poussés.* »

L'eXtrême Programming (**XP**) est une méthodologie légère qui met l'accent sur l'activité de programmation et qui s'appuie sur les valeurs suivantes : **communication, simplicité, feedback et le courage**. Elle est bien adaptée pour des projets de taille moyenne où le contexte (besoins des utilisateurs, technologies informatiques) évolue en permanence.

1. *Communication* :

L'absence de la communication est certainement l'un de défauts les plus graves qui mettent en péril un projet. Les pratiques de **XP** tendent à rendre la communication omniprésente entre tous les intervenants.

Toutes ces pratiques ont pour but de permettre à chacun de se poser de bonnes questions et de partager l'information.

2. *Simplicité* :

XP encourage toujours de développer un système simple qu'on aura engagé de nouveaux frais plus tard pour ajouter de nouvelles fonctionnalités supplémentaires donc de s'orienter vers la solution la plus simple qui puisse satisfaire les besoins du client ; plutôt que de concevoir dès le départ un système très compliqué dont on risque de n'avoir plus besoin dans un avenir proche.

3. *Feedback* :

Le retour est immédiat pour les développeurs grâce aux tests unitaires. Pour les clients le retour se fait à l'échelle de quelques jours grâce aux tests fonctionnels qui leur permettent d'avoir une vision globale du système. Le feedback est indispensable pour que le projet puisse accueillir le changement.

4. *Courage* :

Pour mener à bien un projet **XP**, le client doit avoir du courage de donner un ordre de priorité à ses exigences, de reconnaître que certains de ses besoins ne sont pas toujours bien clairs. De son côté, le développeur doit avoir le courage de modifier l'architecture même si le développement a suffisamment avancé, de jeter du code existant et d'accepter qu'il est parfois plus rapide et efficace de réécrire une portion de code à partir du zéro plutôt que de bricoler un code existant.

Un processus de modélisation avec UML :

Le processus **XP** s'appuie sur UML tout au long du cycle de développement, car les différents diagrammes de ce dernier permettent de part leur facilité et clarté, de bien modéliser le système à chaque étape.

UML s'articule autour de treize types de diagrammes, chacun d'eux étant dédiée à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont repartis en deux groupes : structurels et les diagrammes comportementaux. UML offre beaucoup de diagrammes qui servent à la modélisation des systèmes, mais nous n'utilisons qu'un sous ensemble (suffisant) de ces diagrammes. Nous utilisons les diagrammes suivants :

- **Diagrammes de cas d'utilisation:** qui servent à représenter les fonctionnalités offertes par le système ainsi que les acteurs et les relations existant entre eux.
- **Diagrammes de séquence:** qui servent à montrer les interactions entre objets selon un point de vue temporel.
- **Diagrammes de classe:** qui servent à exprimer, de manière générale, la structure statique du système.
- **Diagrammes d'activité :** qui servent à montrer l'écoulement des actions pendant un processus dans le système.
- **Diagrammes de déploiement :** qui servent à représenter l'environnement d'implémentation du système.
- **Diagramme d'états:** Il montre les différents états et transitions possibles des objets d'une classe [10].

5- Conclusion :

A l'issu de ce chapitre, on a proposé une conception préliminaire de notre système.

Pour chaque projet on a besoin d'un cycle de vie, d'une méthode et des démarches à suivre pour ordonner et faciliter le travail.

C'est pour quoi on a choisi le cycle de vie en cascade avec la méthode agile XP afin de structurer et organiser notre travail.

1-Introduction :

Dans ce chapitre, on va présenter la conception de notre application à l'aide d'UML. Tout d'abord, nous commençons par un diagramme de cas d'utilisation, ensuite on détermine les diagrammes de séquences et les diagrammes d'activité, ainsi que le diagramme d'états transition et de déploiement. . Enfin, le diagramme de classe.

2- Diagramme de cas d'utilisation :

Les cas d'utilisation sont les fonctionnalités par lesquelles, le système communique avec ses acteurs. Chaque acteur peut accéder à un sous-ensemble des cas d'utilisation du système [12].

1) Administrateur :

Notre application va être utilisée par la directrice du CEIL et son secrétaire et comme ils ont les mêmes droits d'accès ainsi que les mêmes tâches ils sont représentés en un seul acteur principal de cette application que nous nommons « Administrateur ».

L'Administrateur : a le droit de mettre à jour et de consulter les listes des étudiants et des enseignants du centre ainsi que la gestion des emplois du temps des salles et des groupes et consulter l'archive.

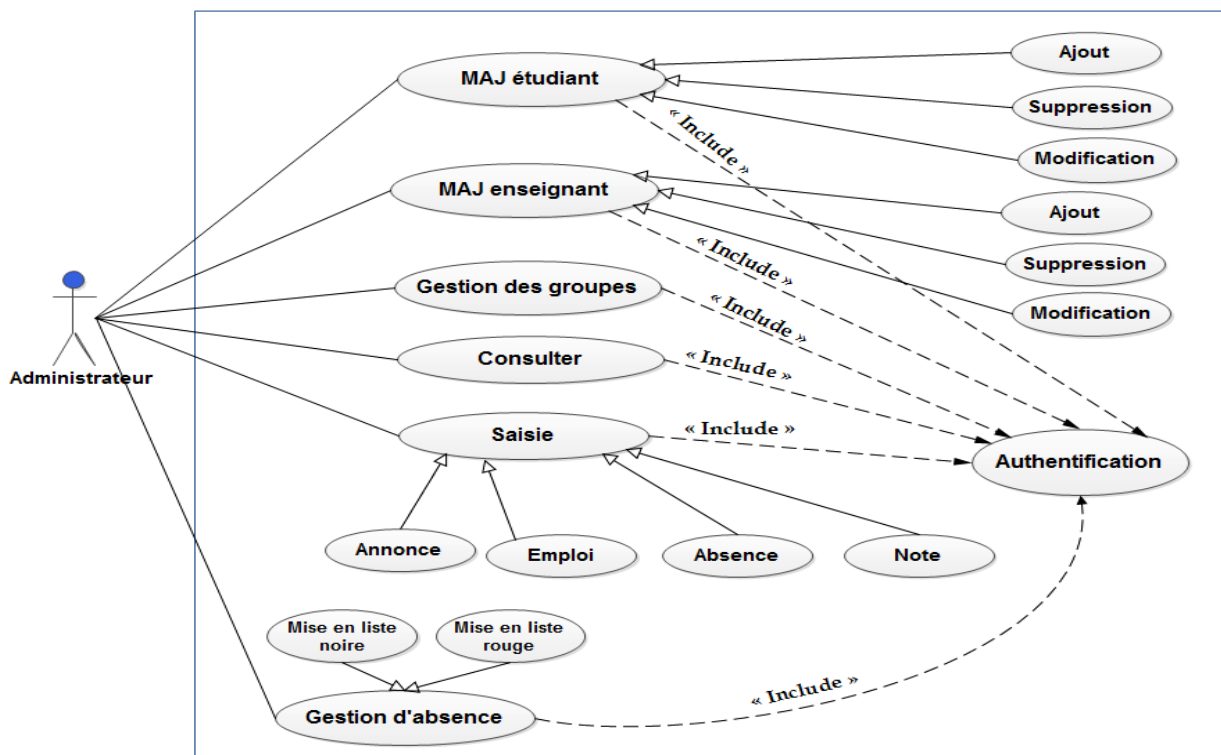


Figure 9: Diagramme de cas d'utilisation de l'administrateur.

2) Etudiant :

A le droit de consulter sa note, le compte, l'emploi du temps et les annonces ainsi de faire le test.

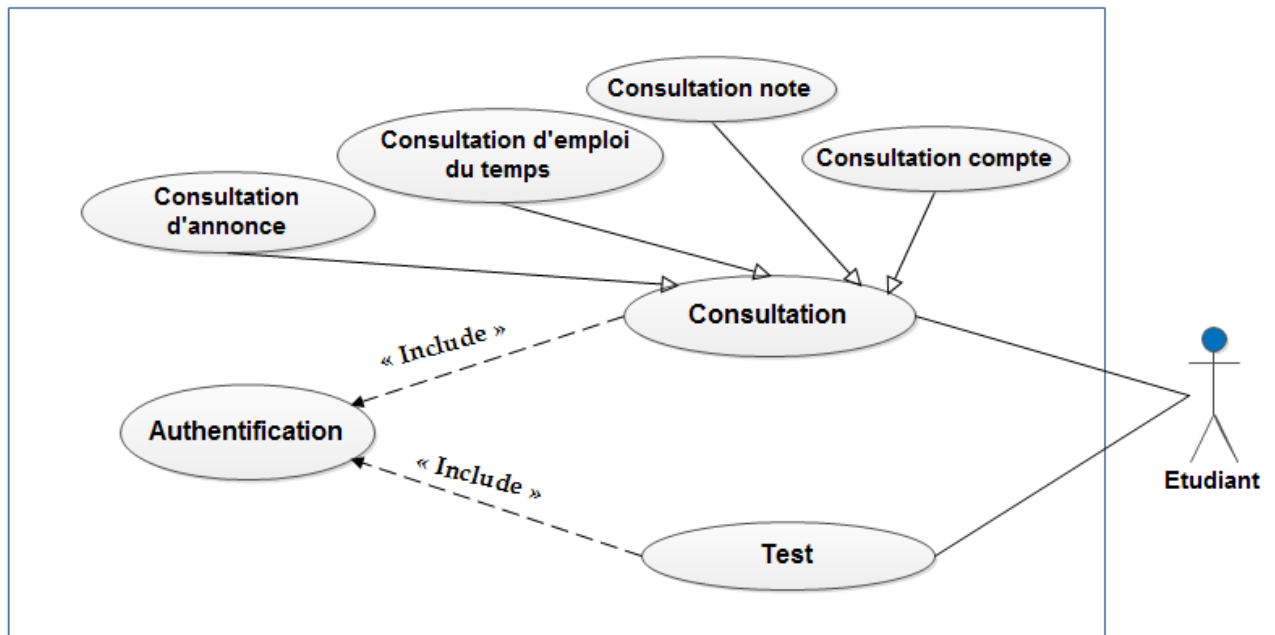


Figure 10: Diagramme de cas d'utilisation d'étudiant.

3) Enseignant :

A le droit de consulter l'emploi du temps et les annonces.

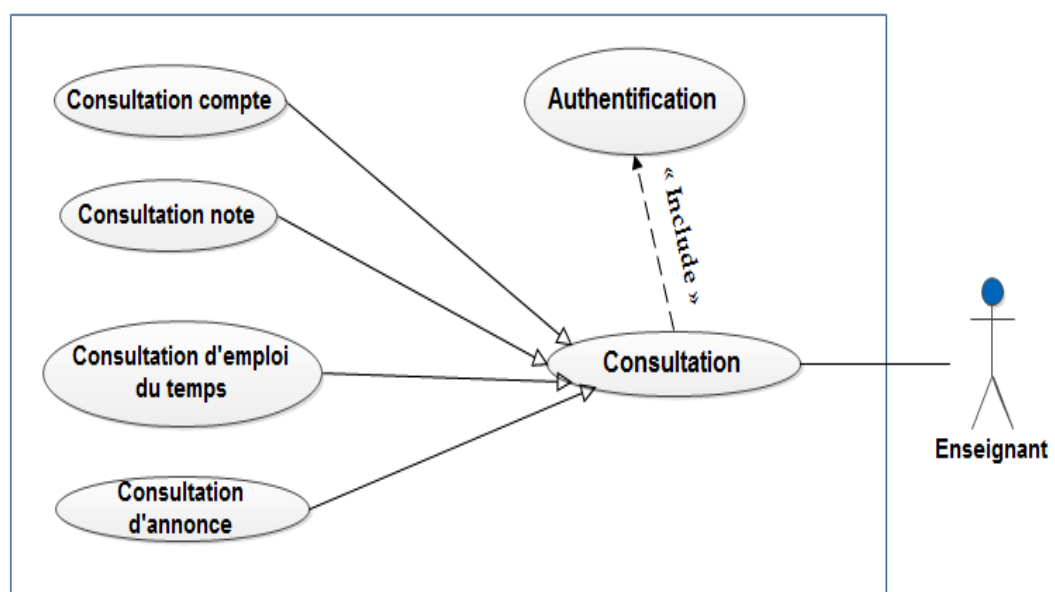


Figure 11: Diagramme de cas d'utilisation d'enseignant.

4) Système :

Le système archive les étudiants de chaque session après le test finale et la gestion des absences en mettant après chaque vérification les étudiants soit en LR ou LN, ainsi qu'il fait la gestion du premier test de niveau des étudiants.

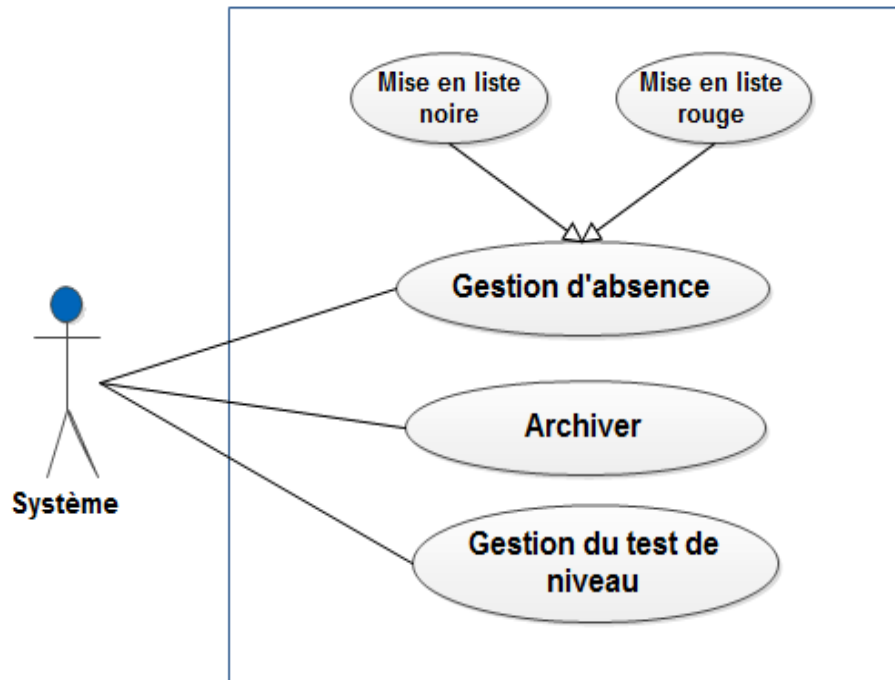


Figure 12: Diagramme de cas d'utilisation de système.

5) Cas d'utilisation globale :

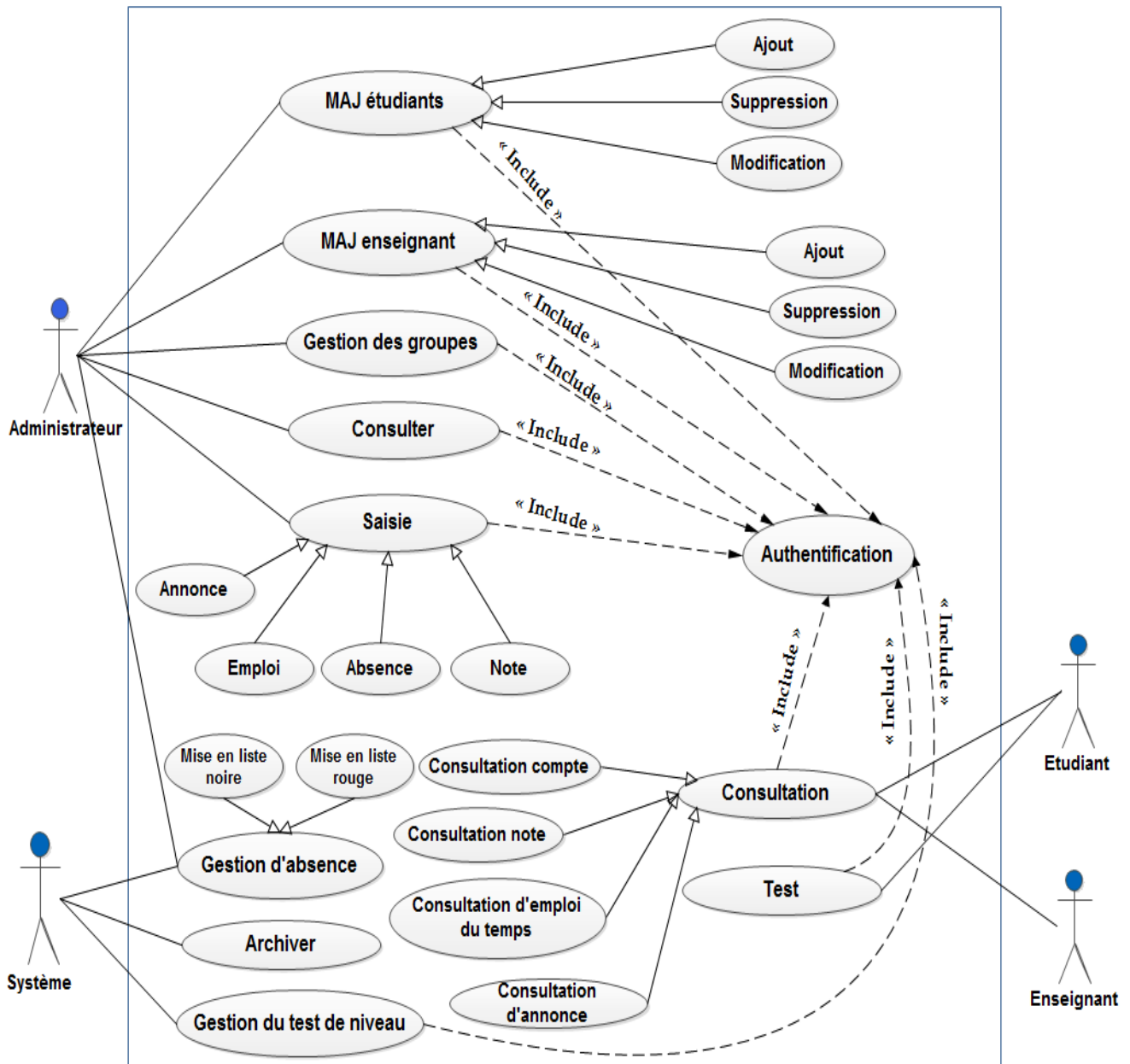


Figure 13: Diagramme de cas d'utilisation générale.

3- Diagrammes de séquences :

Les diagrammes de séquence servent à décrire les scénarios nominaux des cas d'utilisation d'un système. Ils fournissent des détails que les diagrammes de cas d'utilisation ne puissent pas les fournir. Ils montrent les interactions entre le système et ses acteurs pour chaque cas d'utilisation [12].

1) Authentification :

Scénario :

- L'utilisateur accède à l'application.
- Le système affiche une fenêtre pour s'authentifier.
- L'utilisateur saisie le mot de passe et le nom d'utilisateur.
- Le système vérifie le mot de passe et le nom d'utilisateur.
- Si les informations saisies sont correctes l'accès au système est donc autorisé.
- Si les informations sont fausses, le système affiche un message d'erreur.
- L'utilisateur essaye pour la deuxième fois, si les informations sont fausses alors il essaye pour la troisième fois.
- Le système à un nombre d'essai de 3 fois si l'utilisateur n'accède pas à la troisième fois le système interrompt l'accès.

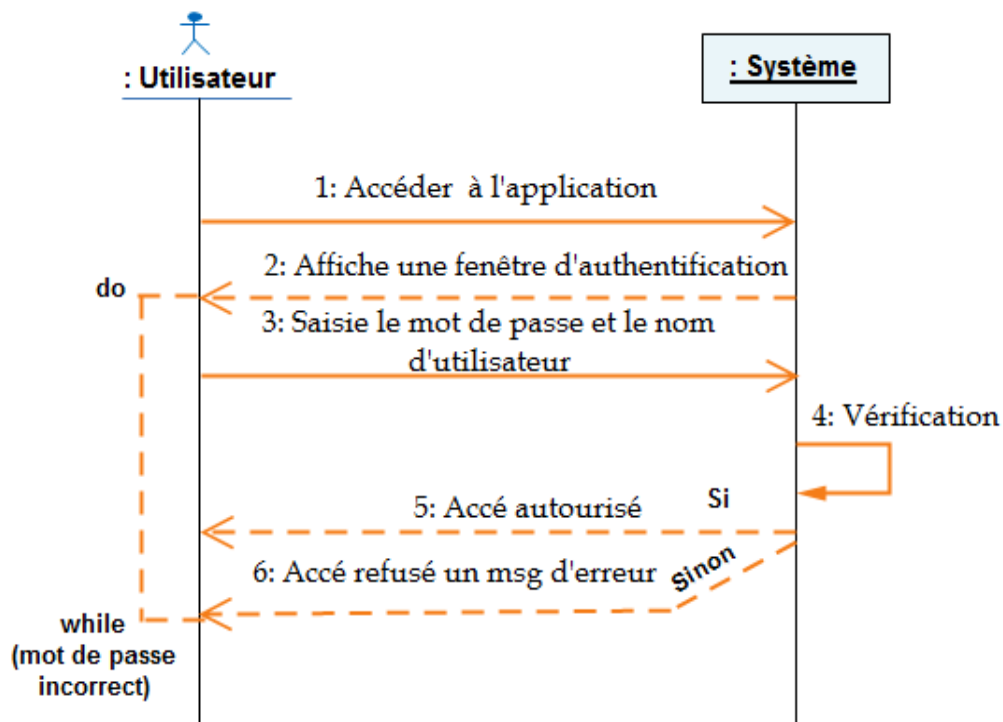


Figure 14: Diagramme de séquence d'authentification.

2) Saisie d'une annonce :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur demande au système l'accé pour saisir une annonce.
- Le système affiche un formulaire pour créer l'annonce.
- L'administrateur crée l'annonce et la valide.
- Le système enregistre l'annonce dans la BDD.
- Le système publie l'annonce sur le site.

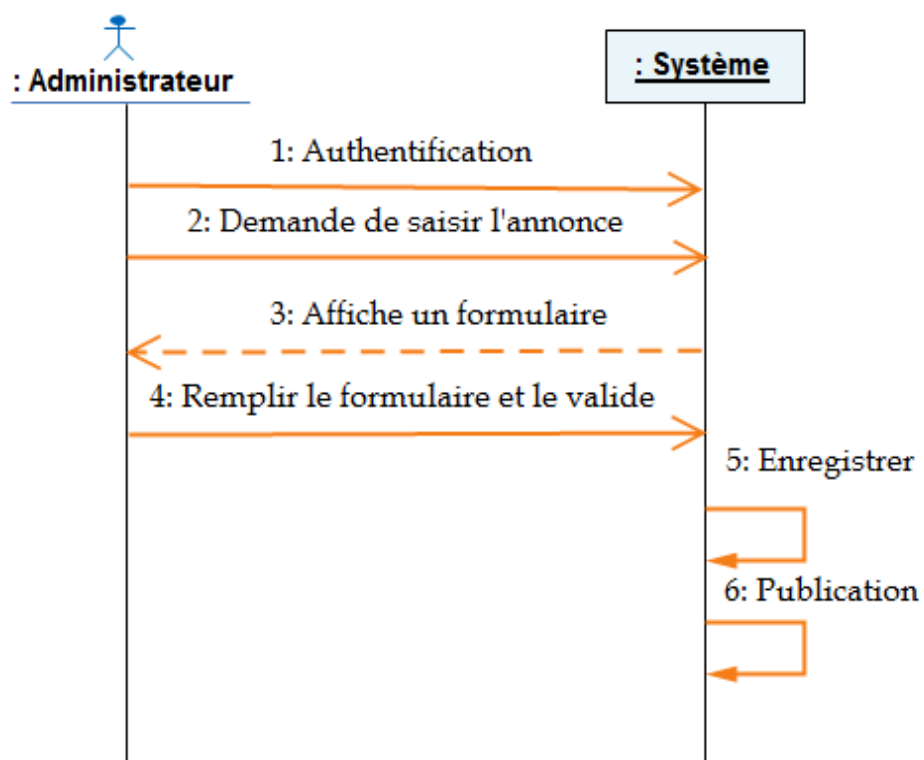


Figure 15: Diagramme de séquence de saisie d'annonce.

3) Saisie d'une note :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur demande de saisir des notes.
- Le système affiche un formulaire à l'administrateur pour choisir un groupe.
- L'administrateur saisie les informations nécessaires.
- Le système cherche le groupe et affiche la liste des étudiants voulu.
- L'administrateur saisie les notes des étudiants et les valide.
- Le système enregistre la liste.

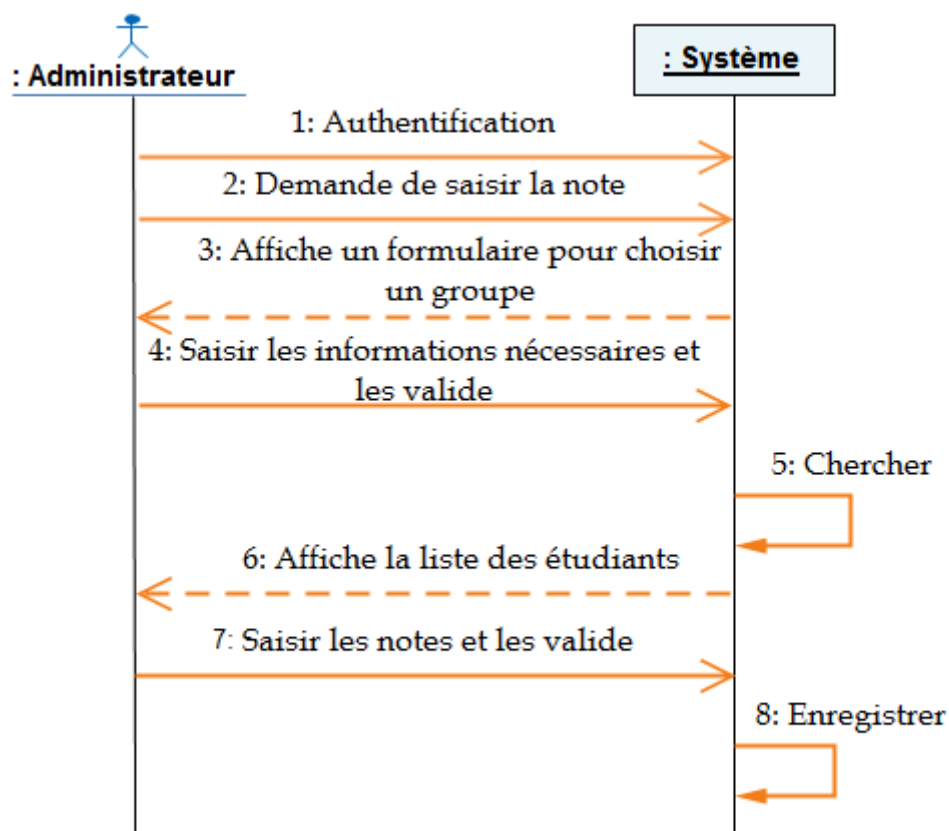
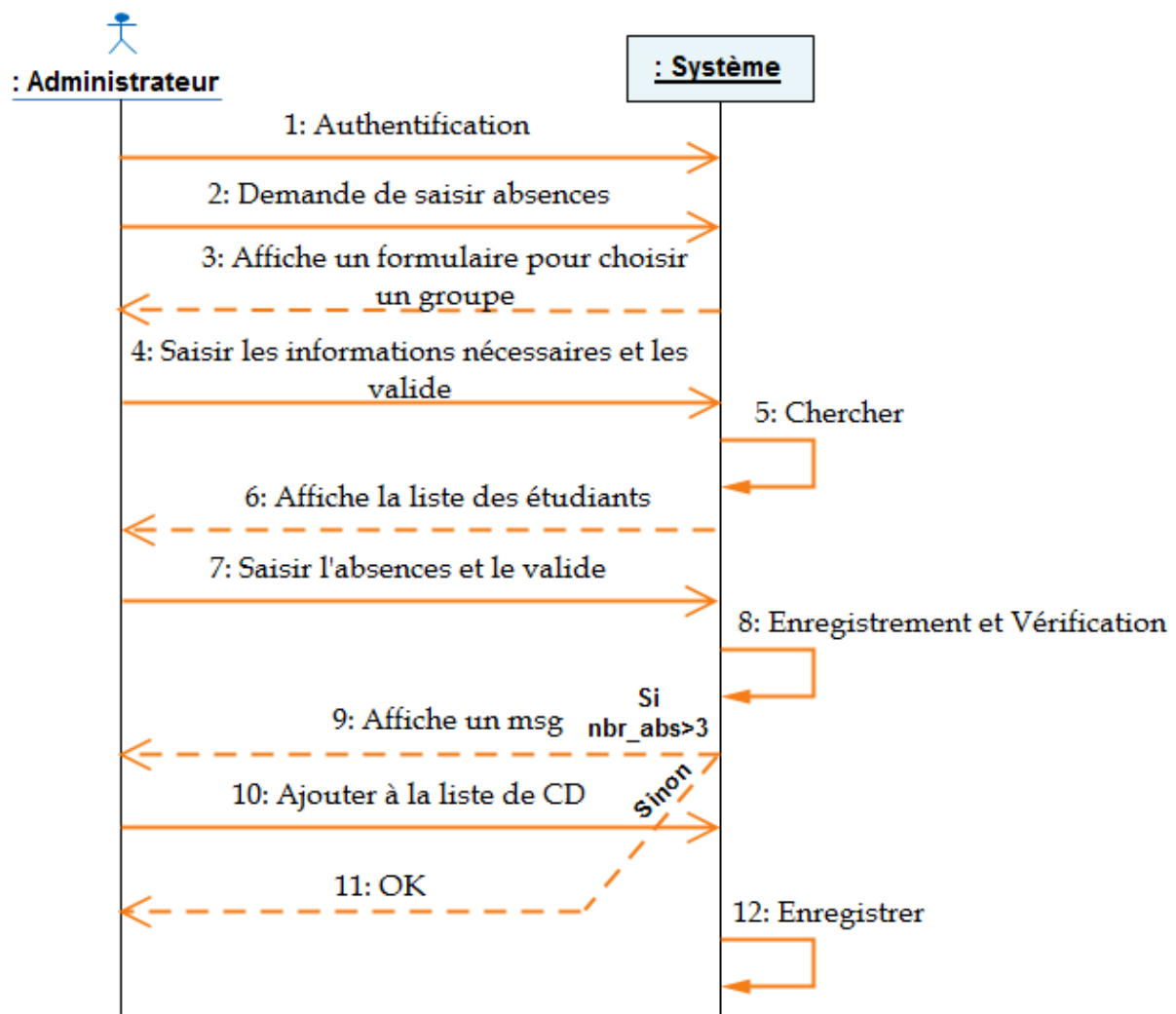


Figure 16: Diagramme de séquence de saisie note.

4) Saisie d'une absence :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur demande de saisir les absences.
- Le système affiche un formulaire à l'administrateur pour choisir le groupe.
- L'administrateur saisie les informations nécessaires.
- Le système cherche le groupe et affiche la liste des étudiants.
- L'administrateur saisie l'absence de chaque étudiant pour chaque séance et la valide.
- Le système enregistre la liste dans la BDD.
- Si le système trouve pour chaque étudiant un nombre d'absence supérieure à 3, il affiche un message pour l'administrateur.
- L'administrateur ajoute l'étudiant à la liste de conseil discipline "CD".
- Le système sauvegarde les renouvellements.

**Figure 17:** Diagramme de séquence de saisie d'absence.

5) Mise à jour d'enseignant :**1. Modification :**

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur envoi une requête de demande de MAJ enseignant.
- Le système demande à l'administrateur le type de MAJ.
- L'administrateur choisit le type de modification.
- Le système affiche la liste des enseignants.
- L'administrateur choisi l'enseignant à modifier.
- Le système affiche un formulaire à l'administrateur.
- L'administrateur modifie le formulaire et l'envoi au système.
- Le système vérifie les modifications et les valide.
- Le système enregistre les modifications dans la BDD.

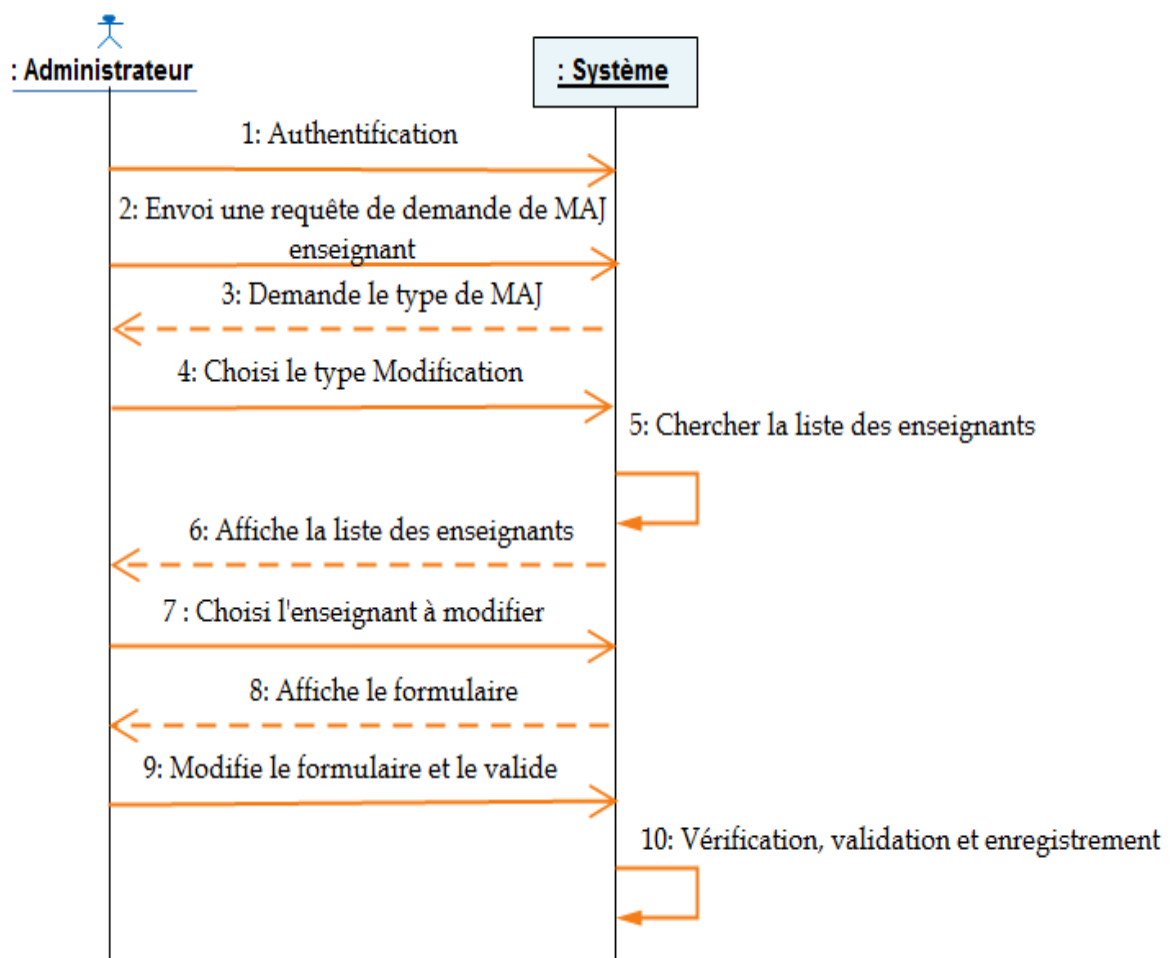


Figure 18: Diagramme de séquence de MAJ enseignant « Modification ».

2. Ajout :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur envoi une requête de demande de MAJ enseignant.
- Le système demande à l'administrateur le type de MAJ.
- L'administrateur choisit le type d'ajout.
- Le système affiche un formulaire à l'administrateur.
- L'administrateur remplit le formulaire et le valide.
- Le système vérifie les informations et les valide.
- Le système enregistre les modifications dans la BDD.

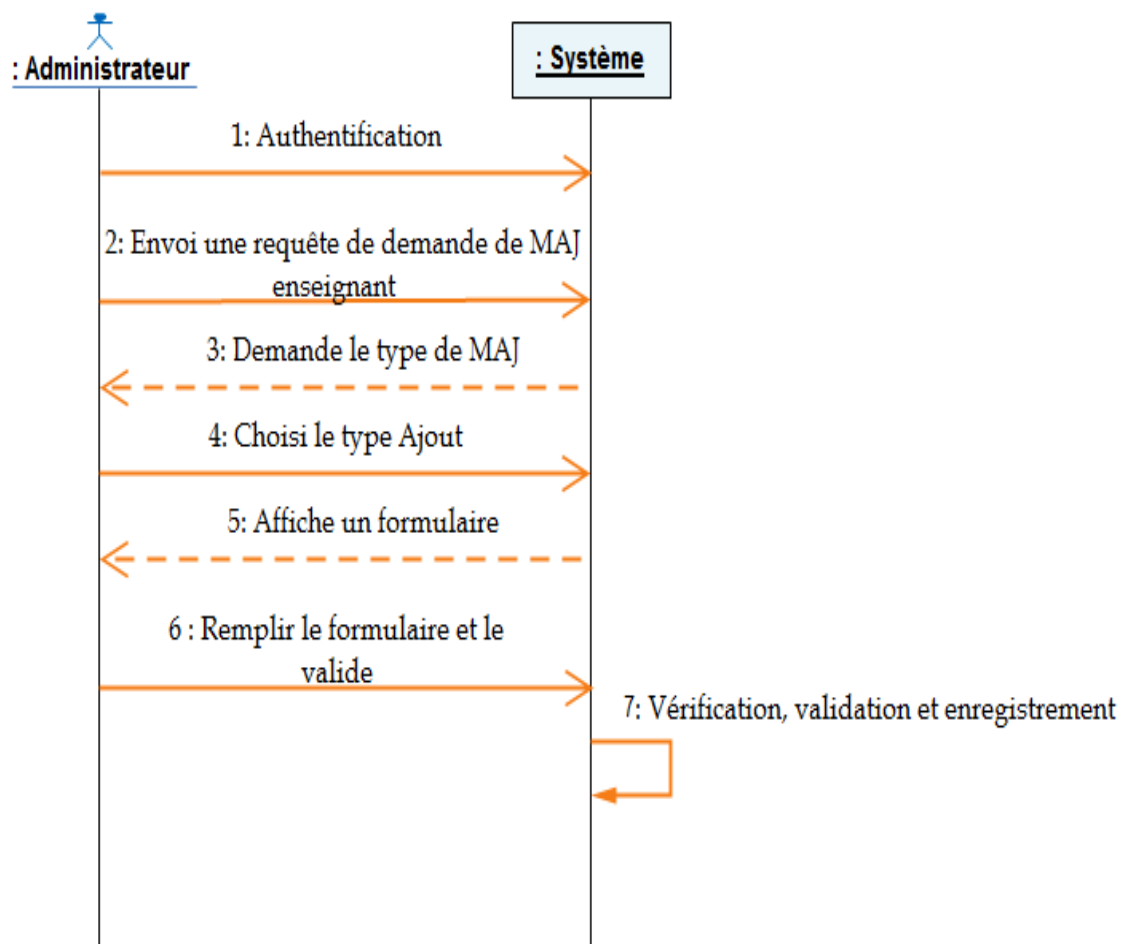


Figure 19: Diagramme de séquence de MAJ enseignant « Ajout ».

3. Suppression :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur envoi une requête de demande de MAJ enseignant.
- Le système demande à l'administrateur le type de MAJ.
- L'administrateur choisit le type de suppression.
- Le système affiche la liste des enseignants.
- L'administrateur choisi l'enseignant à supprimer.
- Le système supprime l'enseignant de la BDD.

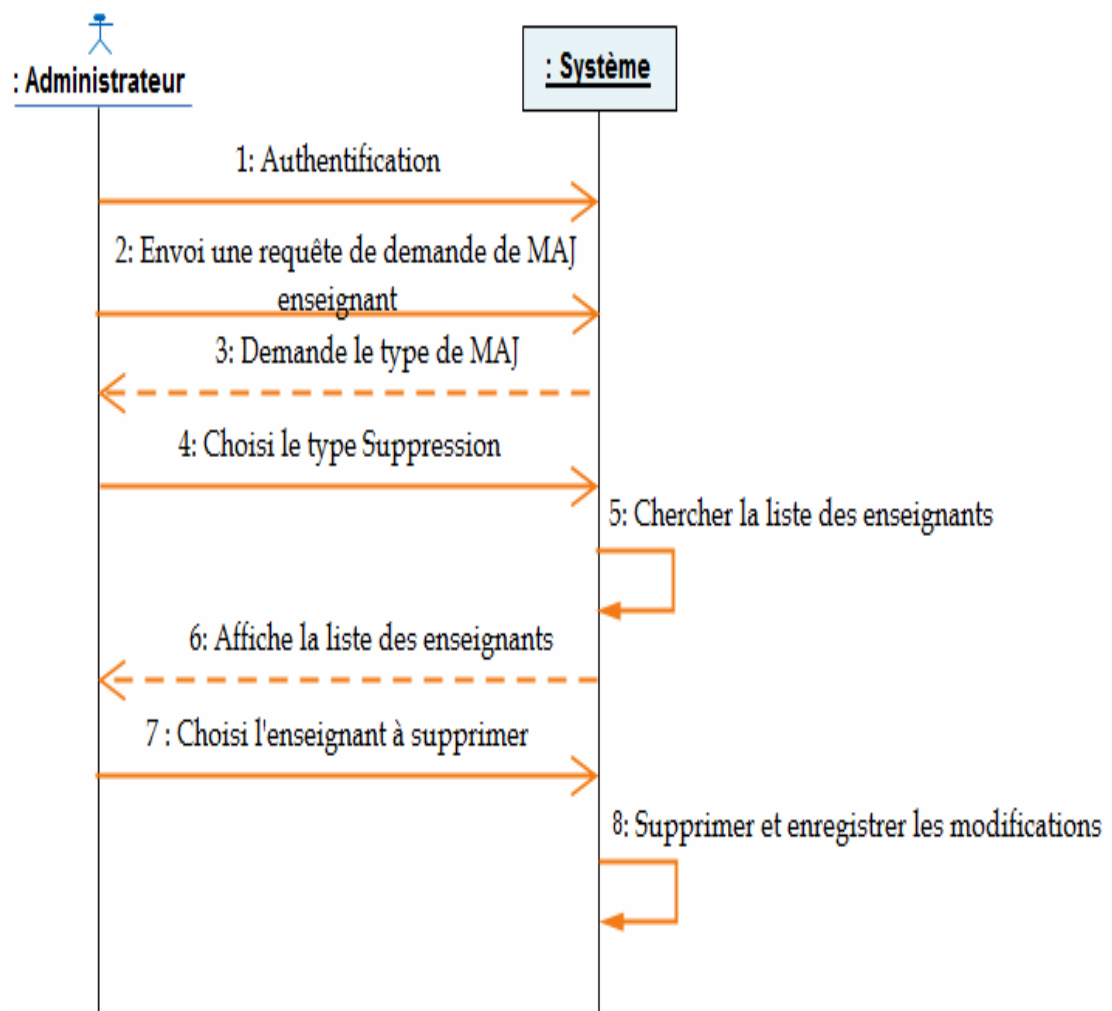


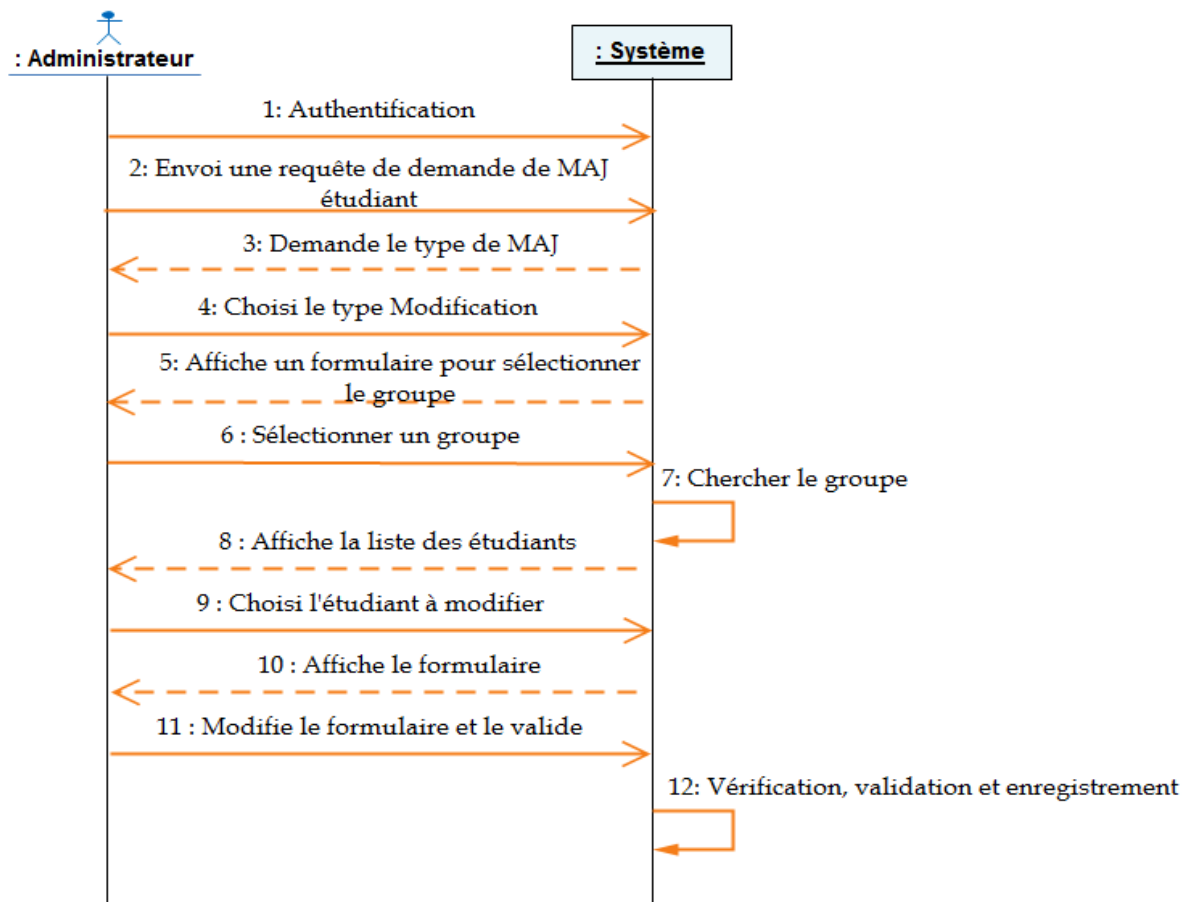
Figure 20: Diagramme de séquence de MAJ enseignant « Sppression ».

6) Mise à jour d'étudiant :

1. Modification :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur envoi une requête de demande de MAJ étudiant.
- Le système demande à l'administrateur le type de MAJ.
- L'administrateur choisit le type de modification.
- Le système affiche le formulaire de sélection.
- L'administrateur choisi la langue, le niveau et le groupe à modifier.
- Le système cherche dans la BDD les informations et affiche la liste des étudiants.
- L'administrateur sélectionne l'étudiant.
- Le système affiche le formulaire de l'étudiant.
- L'administrateur envoi les modifications.
- Le système vérifie les modifications et les valide.
- Le système enregistre les modifications dans la BDD.

**Figure 21:** Diagramme de séquence de MAJ étudiant « Modification ».

2. Ajout :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur envoi une requête de demande de MAJ étudiant.
- Le système demande à l'administrateur le type de MAJ.
- L'administrateur choisit le type d'ajout.
- Le système affiche un formulaire à l'administrateur.
- L'administrateur rempli le formulaire et le valide.
- Le système vérifie si l'étudiant existe déjà dans la LN ou LR.
- Si l'étudiant existe dans l'un des deux listes alors l'ajout est refusé.
- Sinon l'ajout est effectué.
- L'administrateur les valide.
- Le système enregistre l'étudiant dans la BDD.

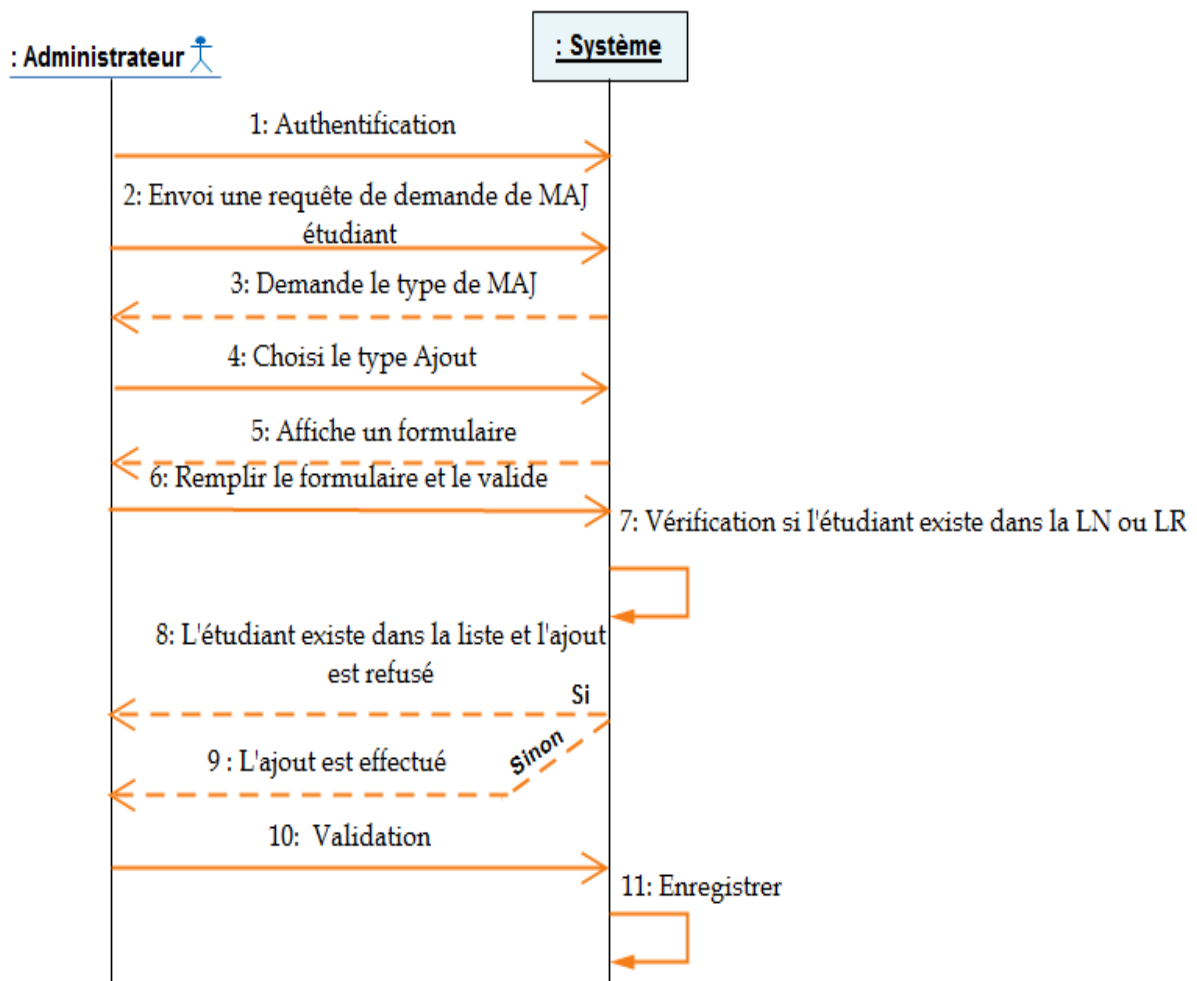


Figure 22: Diagramme de séquence de MAJ étudiant « Ajout ».

3. Suppression :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur envoi une requête de demande de MAJ étudiant.
- Le système demande à l'administrateur le type de MAJ.
- L'administrateur choisit le type suppression.
- Le système affiche la liste des étudiants.
- L'administrateur choisi l'étudiant à supprimer.
- Le système supprime l'étudiant.

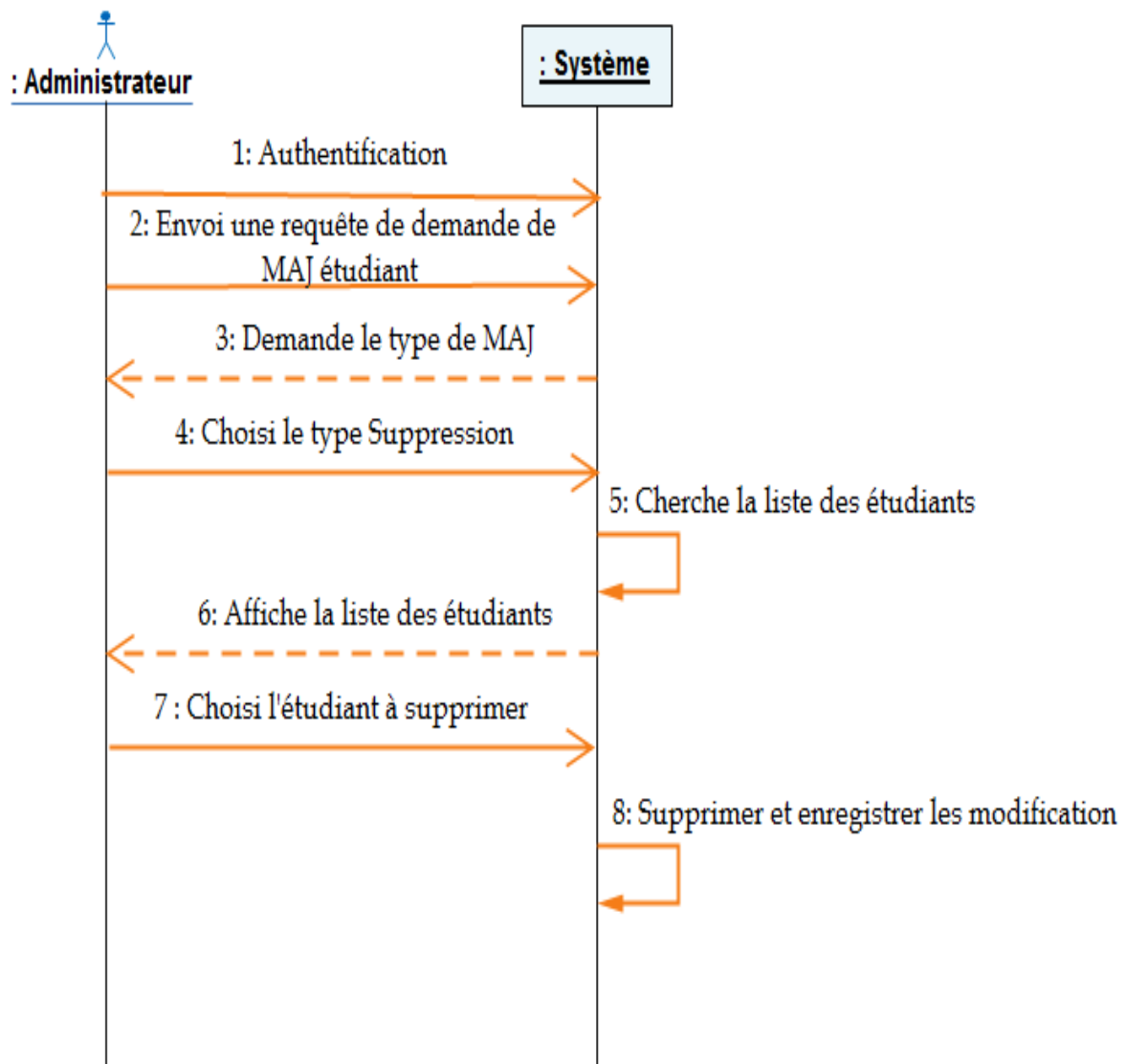


Figure 23: Diagramme de séquence de MAJ étudiant « Suppression ».

7) Gestion de l'emploi du temps :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur affiche les salles libres avec heure et date.
- Le système envoie un message aux enseignants et étudiants inscrits dans cette session contient les salles libres avec heure et date.
- Les enseignants et les étudiants choisissent les dates et l'heure qui conviennent à leur horaire et l'envoient au système.
- Le système vérifie les messages arrivées et les rassemble pour envoyer les résultats à l'administrateur.
- L'administrateur demande au système de saisir l'emploi du temps, après le sauvegarder et d'organiser les groupes selon l'emploi.
- Le système affiche à l'administrateur les formulaires de l'emploi du temps et de groupe.
- L'administrateur remplit les formulaires et les valide.
- Le système enregistre l'emploi du temps et les groupes dans la BDD et le publie.

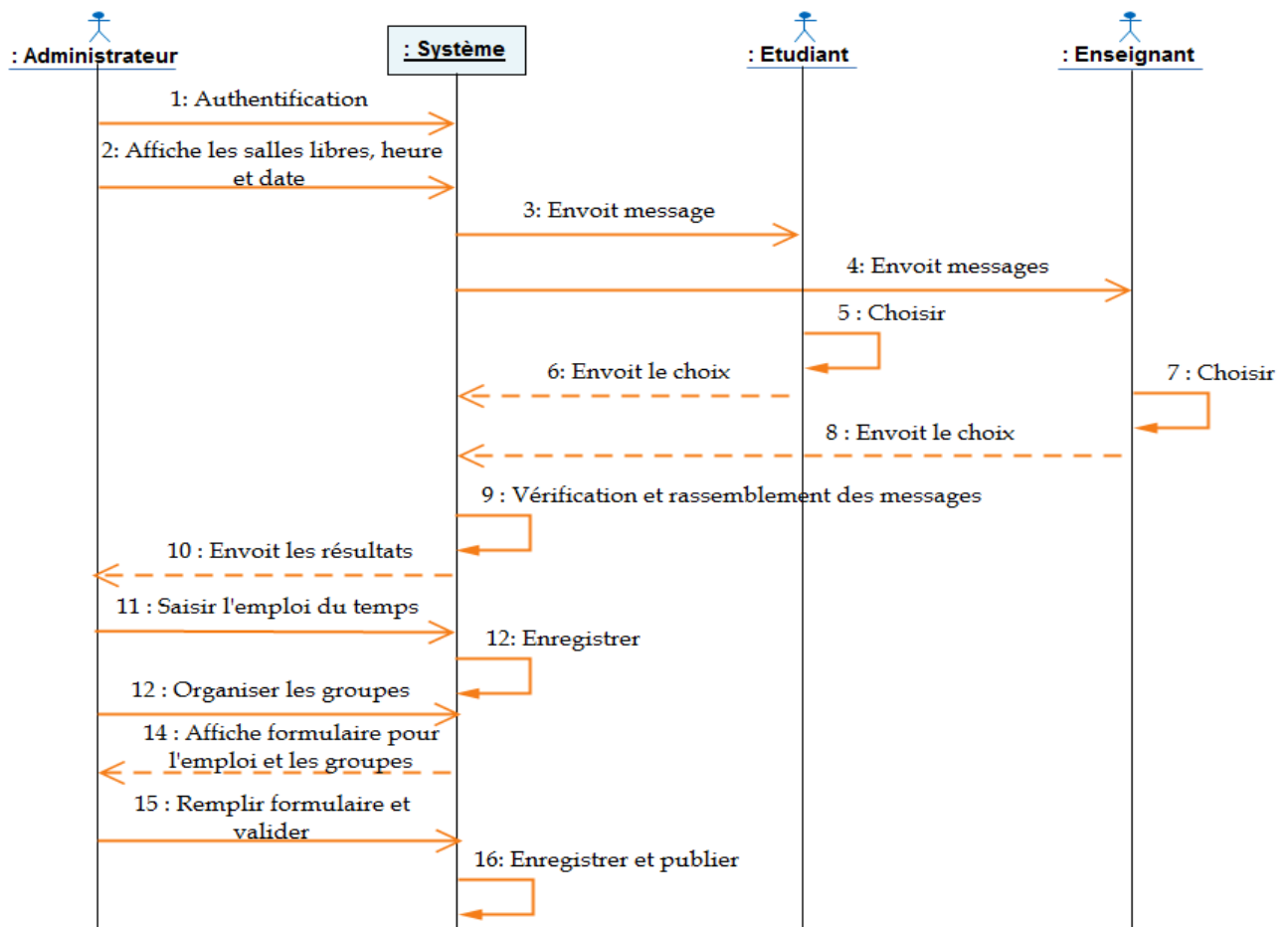
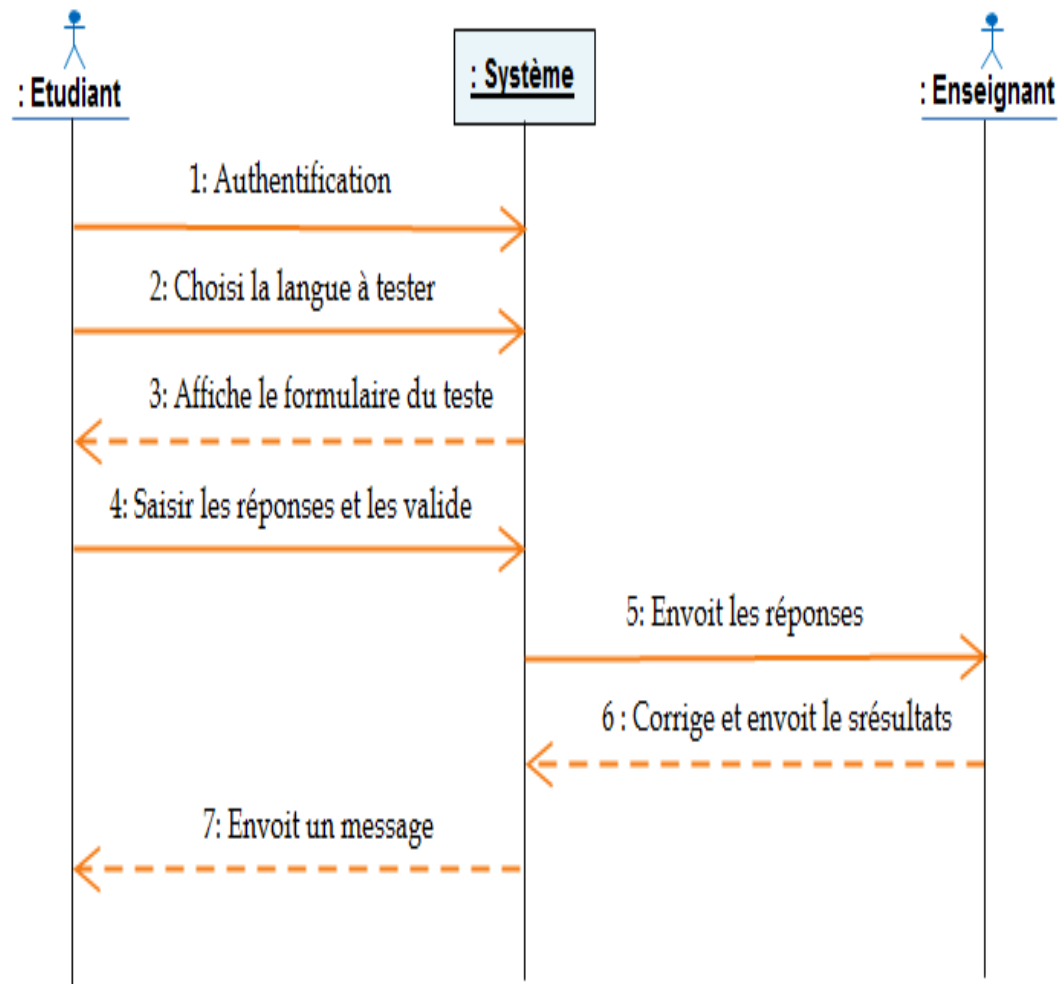


Figure 24: Diagramme de séquence de gestion de l'emploi du temps.

8) Test :

Scénario :

- L'étudiants s'authentifier d'abord.
- L'étudiants choisi la langue à tester.
- Le système affiche le formulaire du test.
- L'étudiant saisi ces réponses et les valide.
- Le système envoi les réponses des étudiants à l'enseignant pour corriger.
- L'enseignant corrige le test et envoi les résultats au système.
- Le système envoi un message de réussite ou d'échec à l'étudiant.

**Figure 25:** Diagramme de séquence du test.

9) Consultation : « Administrateur / Enseignant/ Etudiant »

Scénario :

- L'utilisateur s'authentifie d'abord.
- L'utilisateur envoie une requête de demande de consultation.
- Le système demande le type de consultation.
- L'utilisateur choisit le type de consultation.
- Le système cherche dans la BDD et affiche les résultats à l'utilisateur.

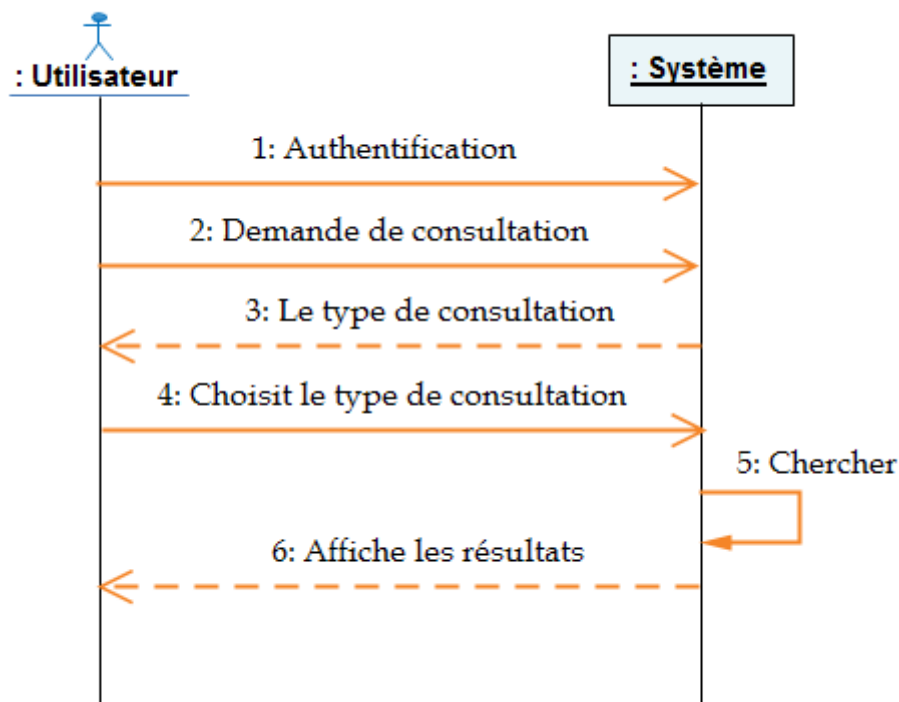


Figure 26: Diagramme de séquence de consultation.

10) Changé d'état de l'étudiant :

Scénario :

- L'administrateur s'authentifier d'abord.
- L'administrateur envoi une requête de demande de changement d'états d'étudiant.
- Le système demande à l'administrateur le groupe à modifier.
- L'administrateur choisit le groupe.
- L'administrateur accède à l'étudiant à modifier son état et il valide.
- Le système enregistre les modifications dans la BDD.

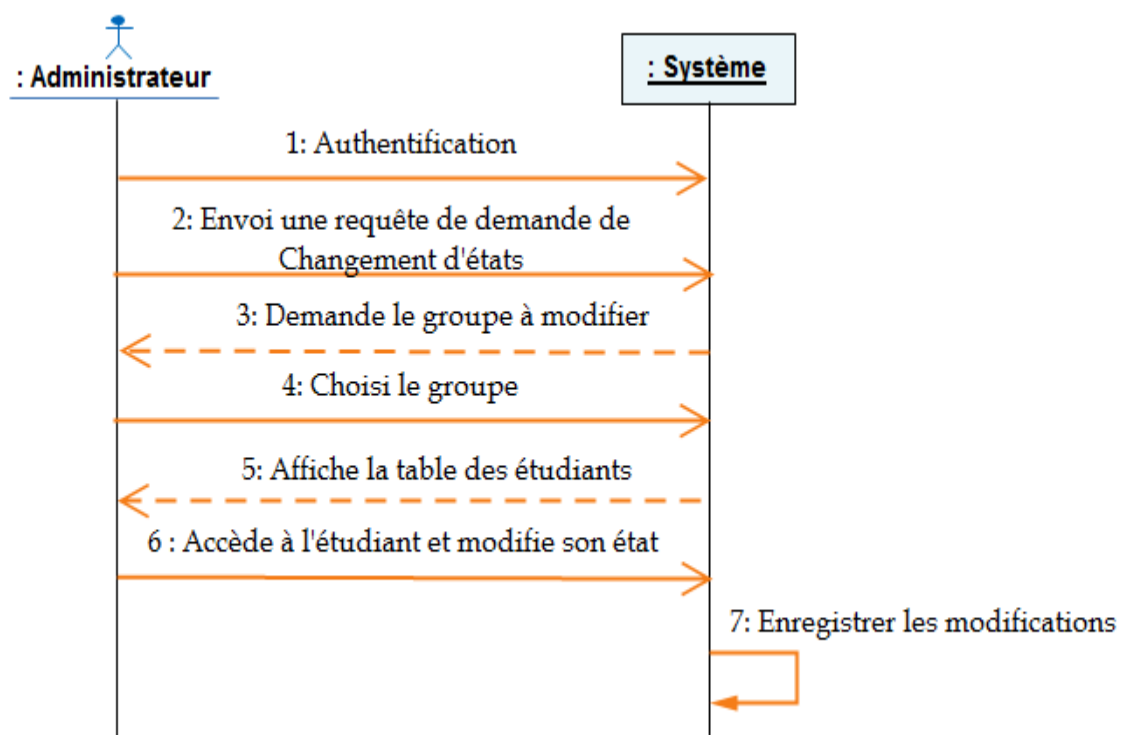


Figure 27: Diagramme de séquence de changement d'état de l'étudiant.

4- Diagrammes d'activité :

Ce diagramme permet de représenter graphiquement le comportement de notre application [12].

➤ **Saisie d'une annonce :**

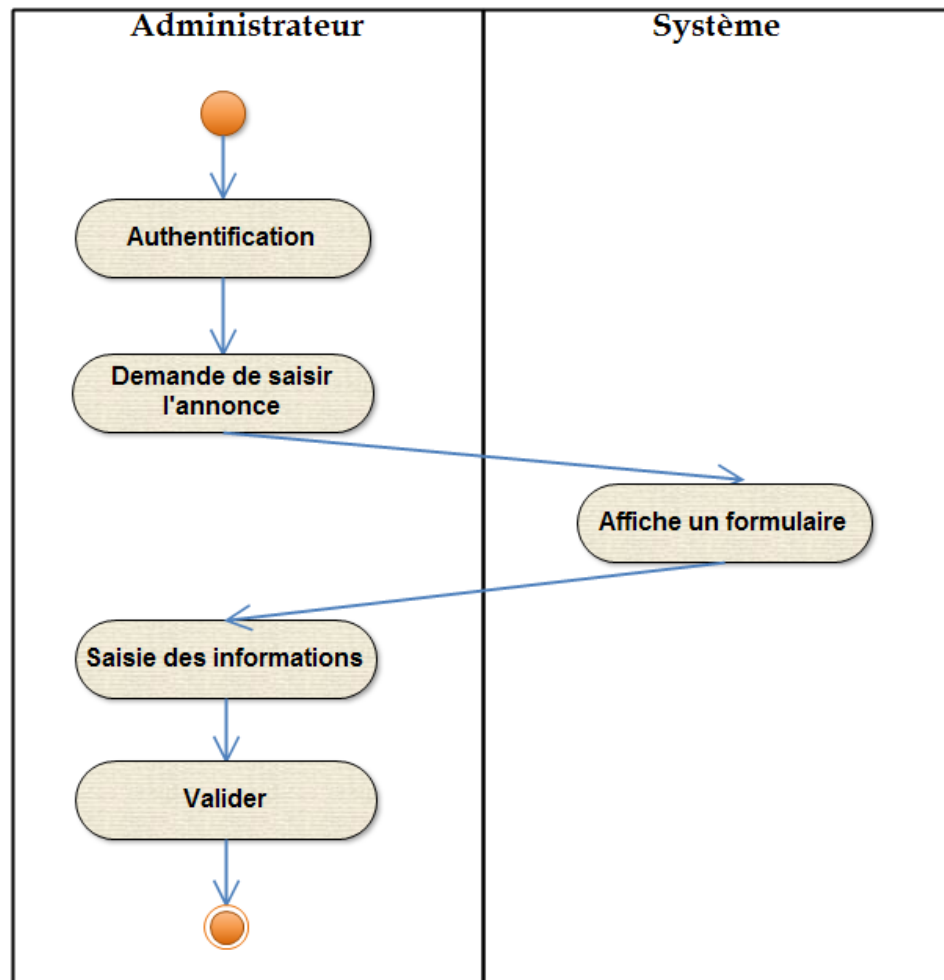


Figure 28: Diagramme d'activité de saisie annonce.

➤ Saisie d'une note :

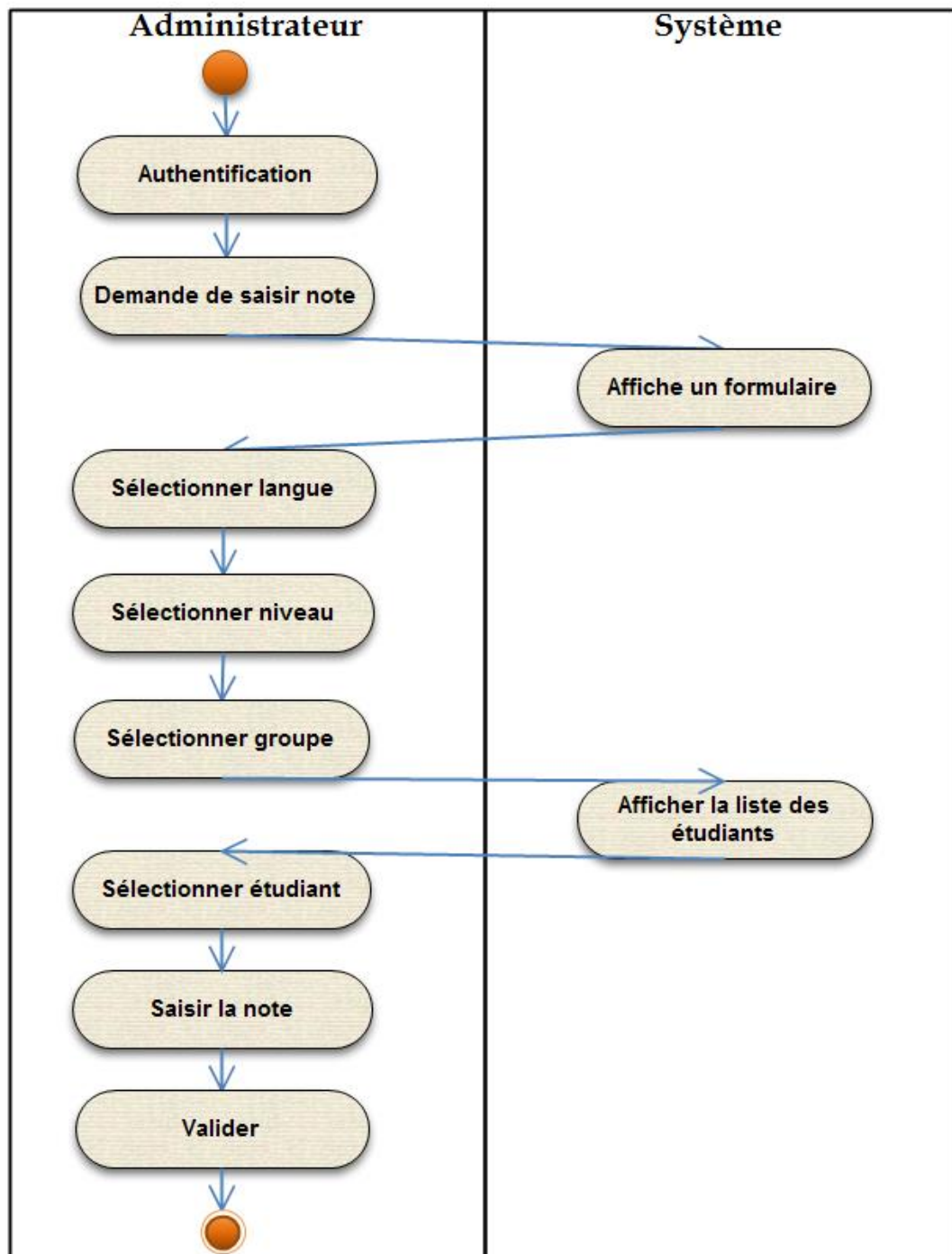
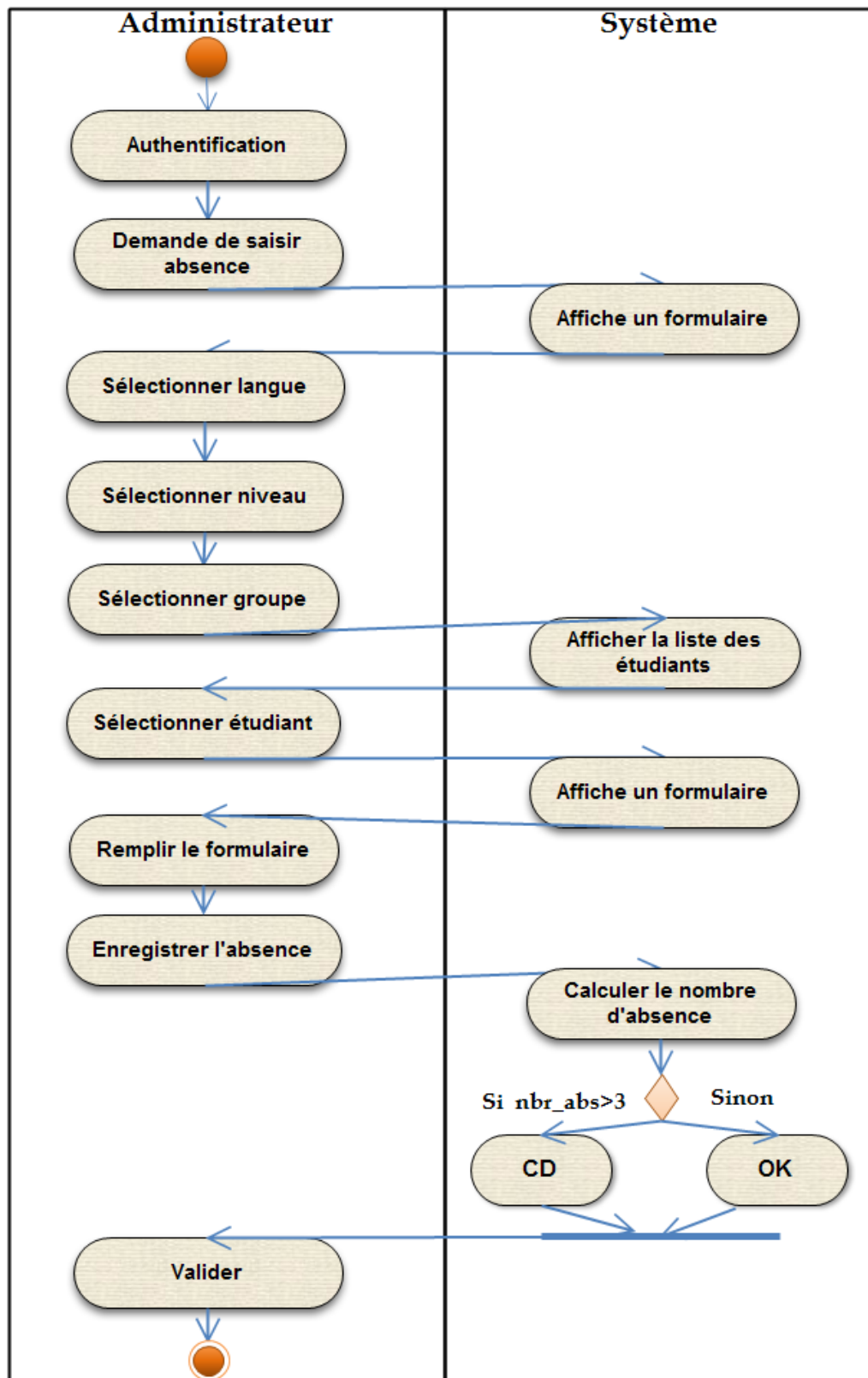
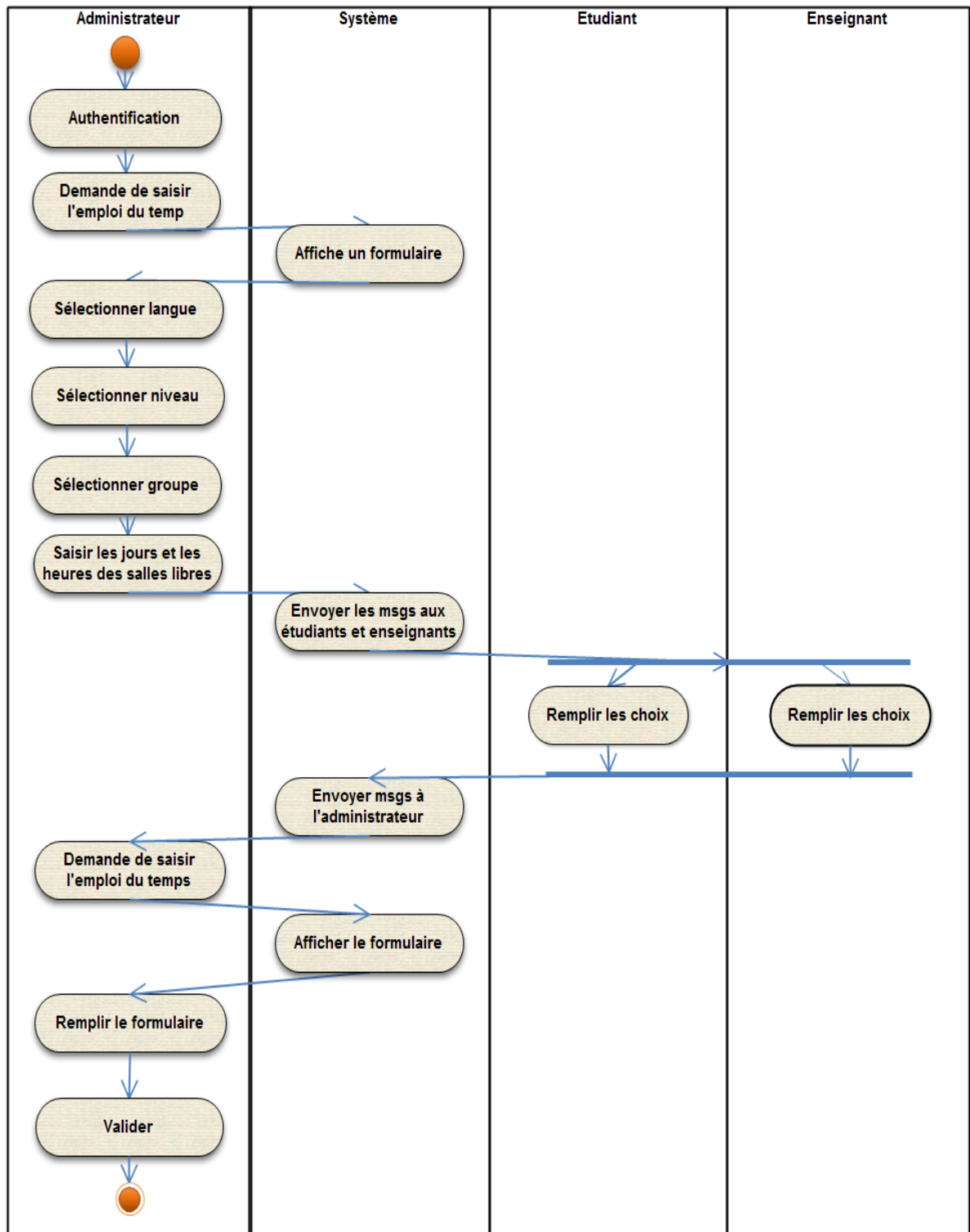


Figure 29: Diagramme d'activité de saisie d'une note.

➤ Saisie d'une absence :

**Figure 30:** Diagramme d'activité de saisie d'une absence.

➤ Saisie de l'emploi du temps :

**Figure 31:** Diagramme d'activité de saisie de l'emploi du temps.

➤ MAJ étudiant : « Modification »

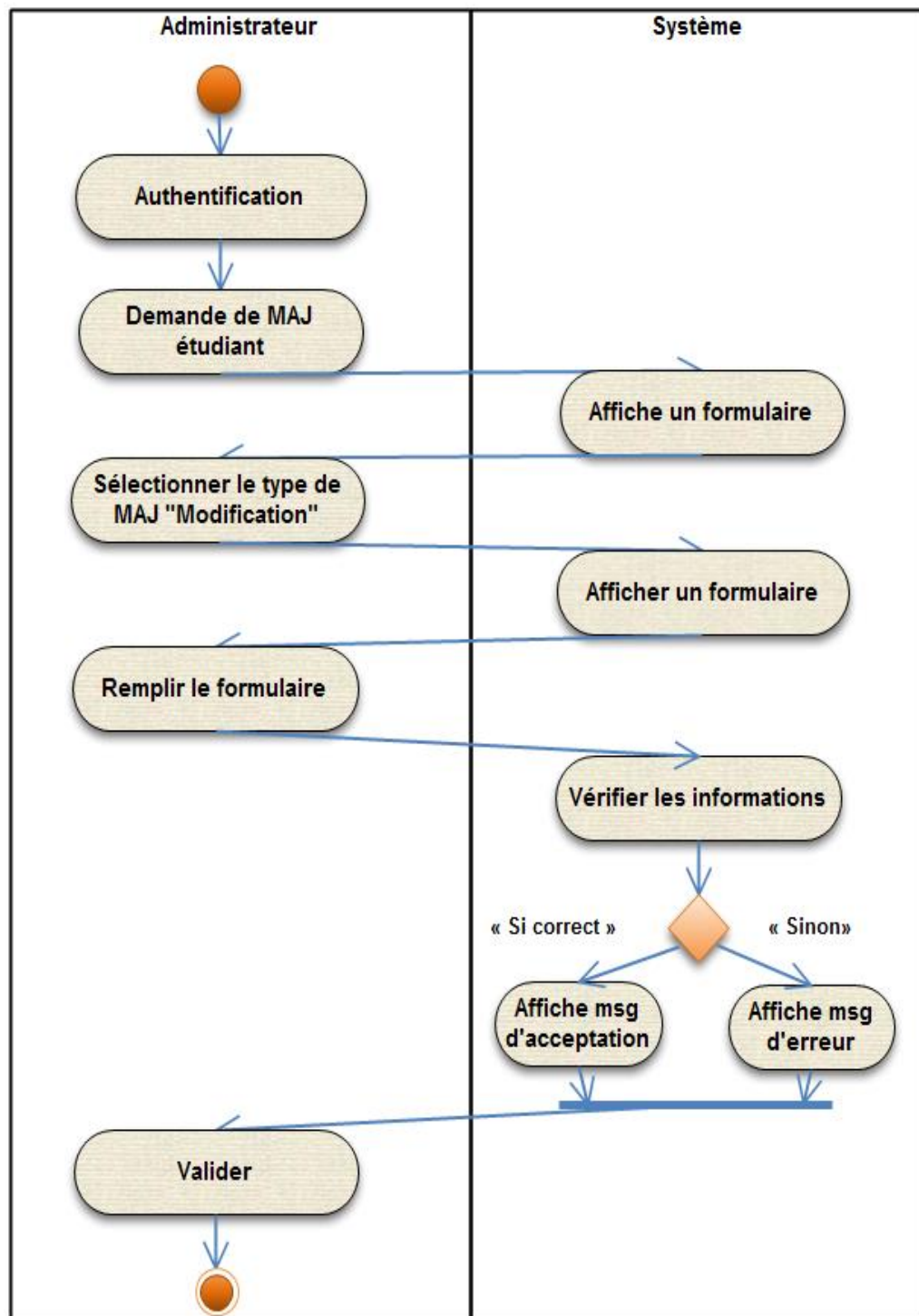


Figure 32: Diagramme d'activité de MAJ étudiant « Modification ».

➤ MAJ étudiant : « Ajout »

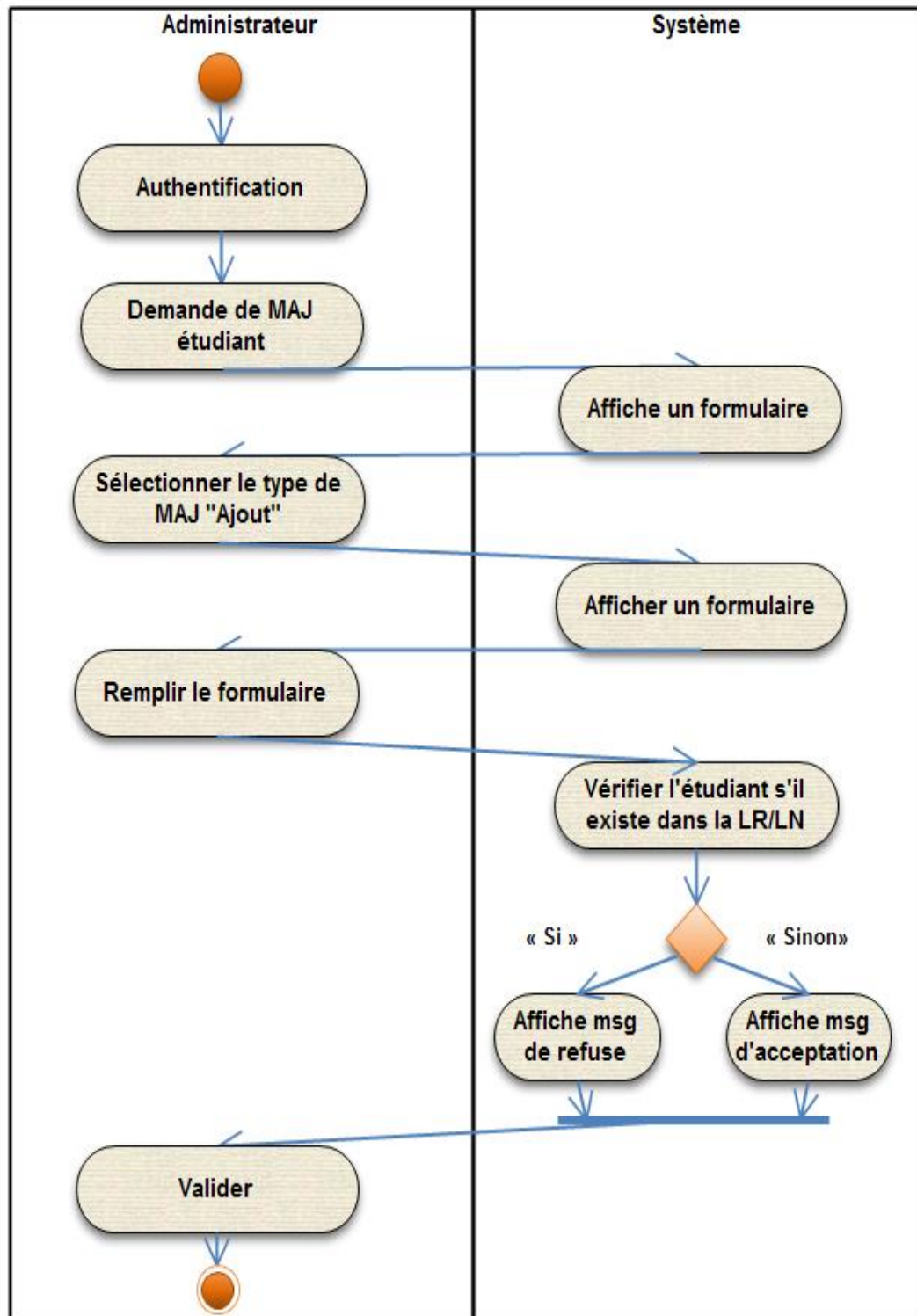


Figure 33: Diagramme d'activité de MAJ étudiant « Ajout ».

➤ MAJ étudiant : « Suppression »

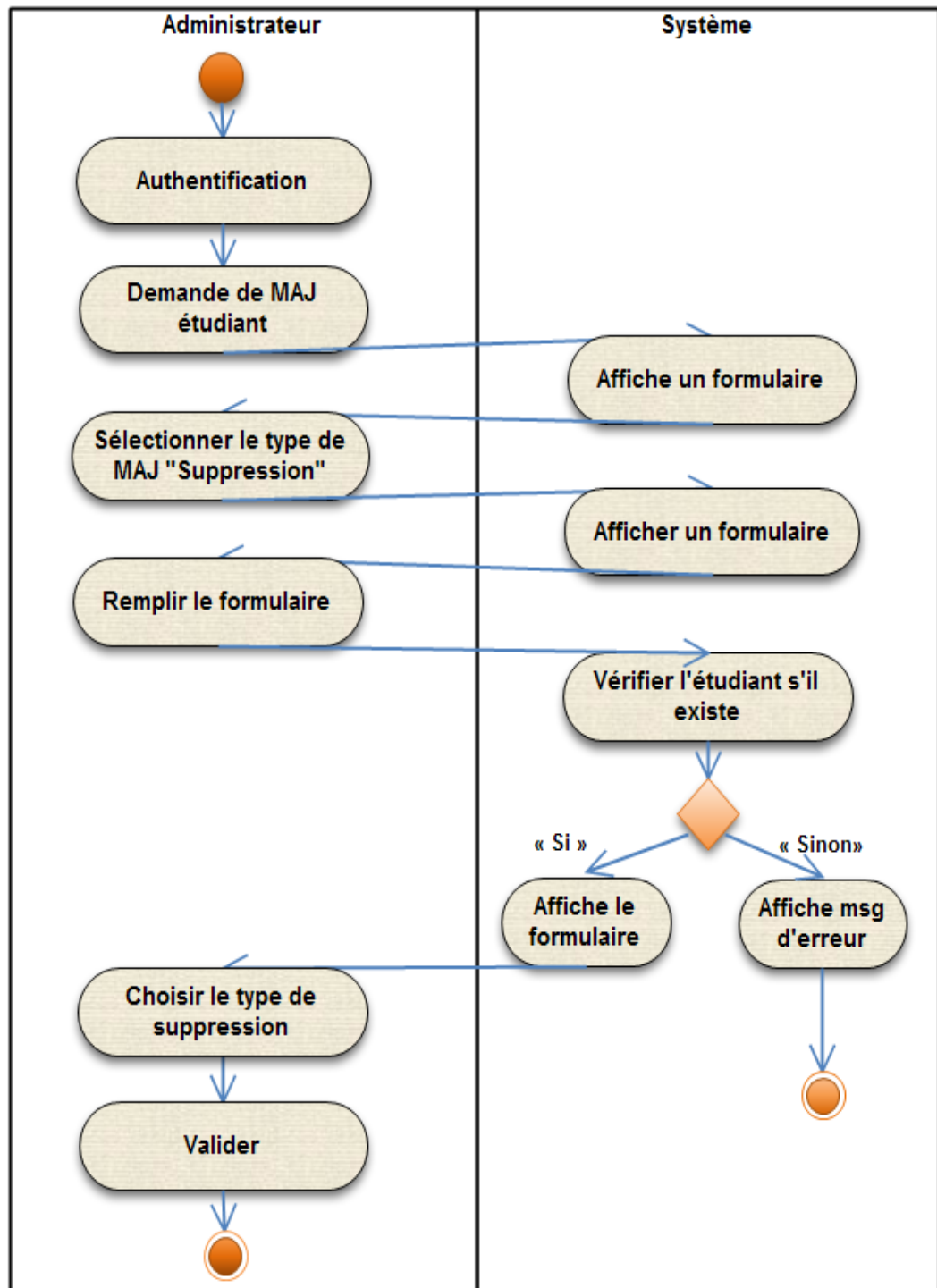


Figure 34: Diagramme d'activité de MAJ étudiant « Suppression ».

➤ MAJ enseignant : « Modification »

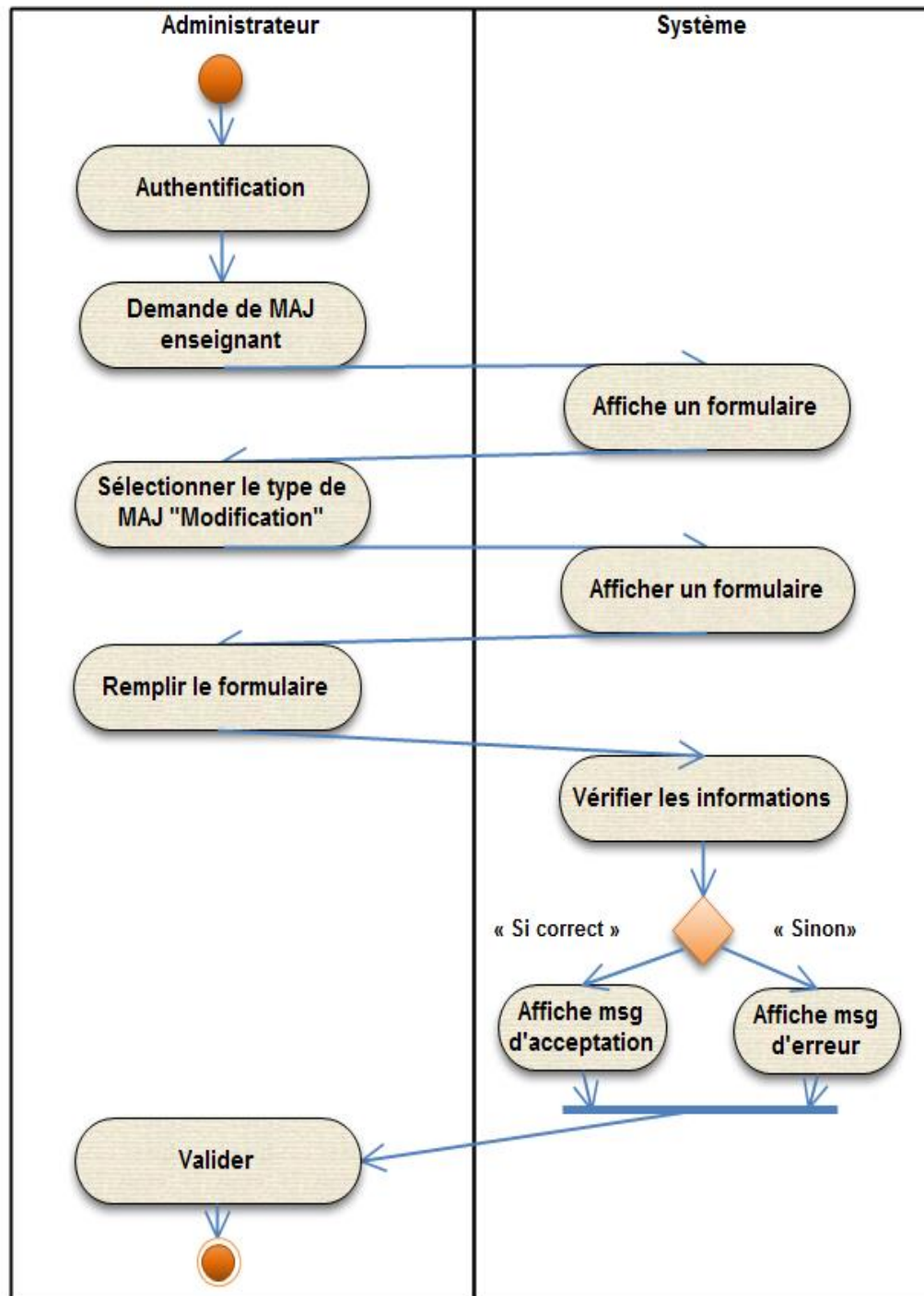


Figure 35: Diagramme d'activité de MAJ enseignant « Modification ».

➤ MAJ enseignant : « Ajout »

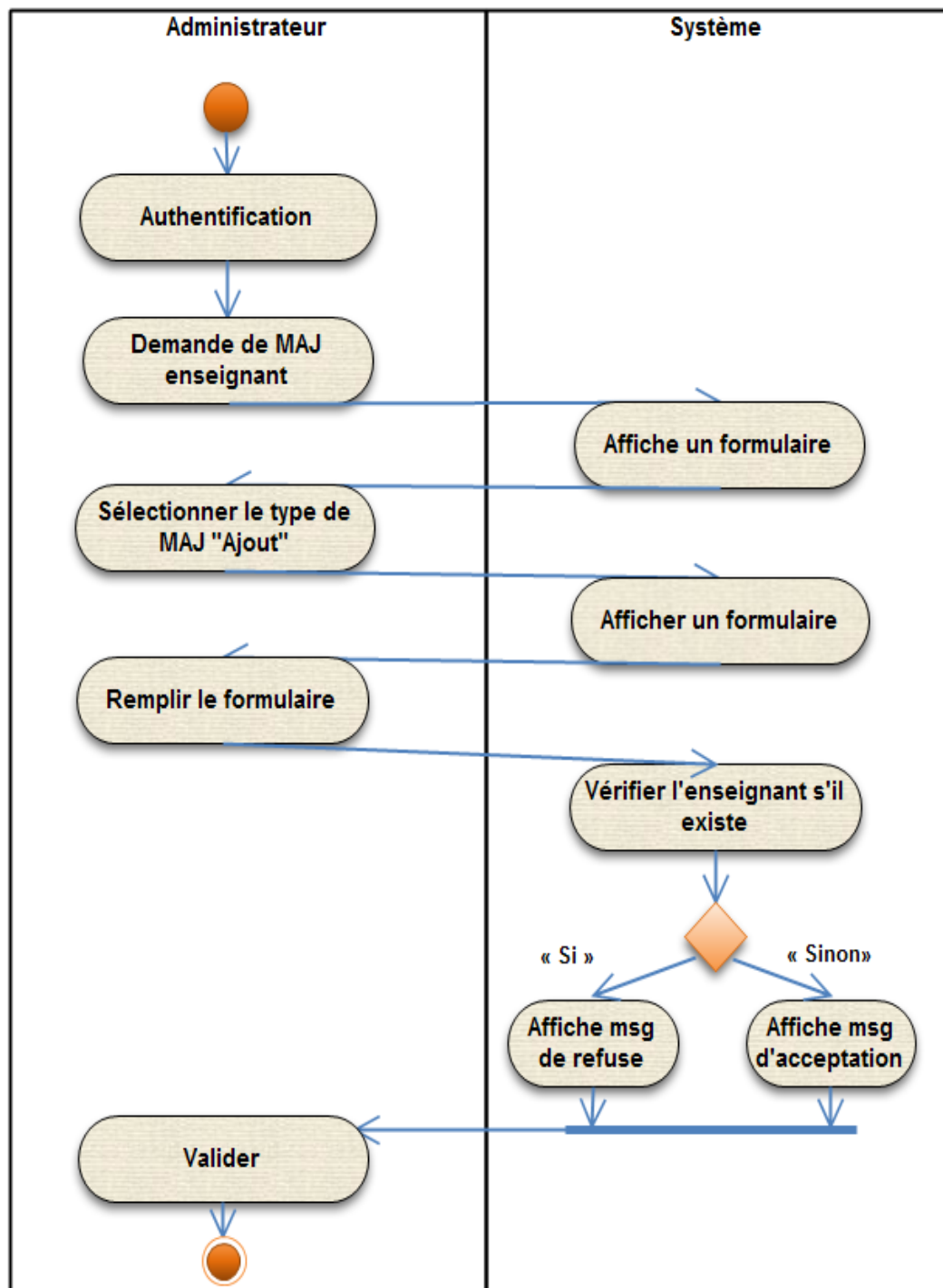


Figure 36: Diagramme d'activité de MAJ enseignant « Ajout ».

➤ MAJ enseignant : « Suppression »

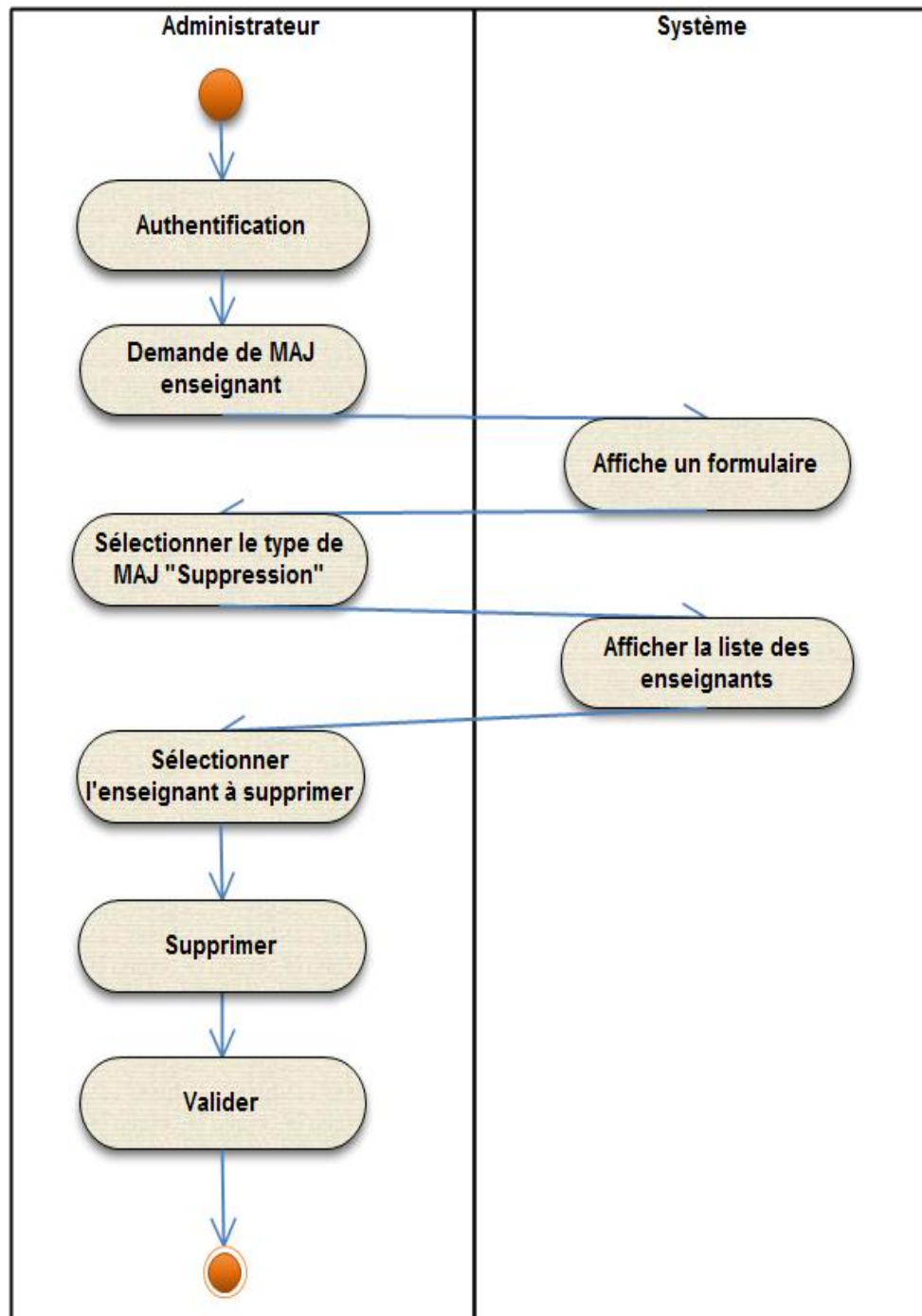


Figure 37: Diagramme d'activité de MAJ enseignant « Suppression ».

➤ Test :

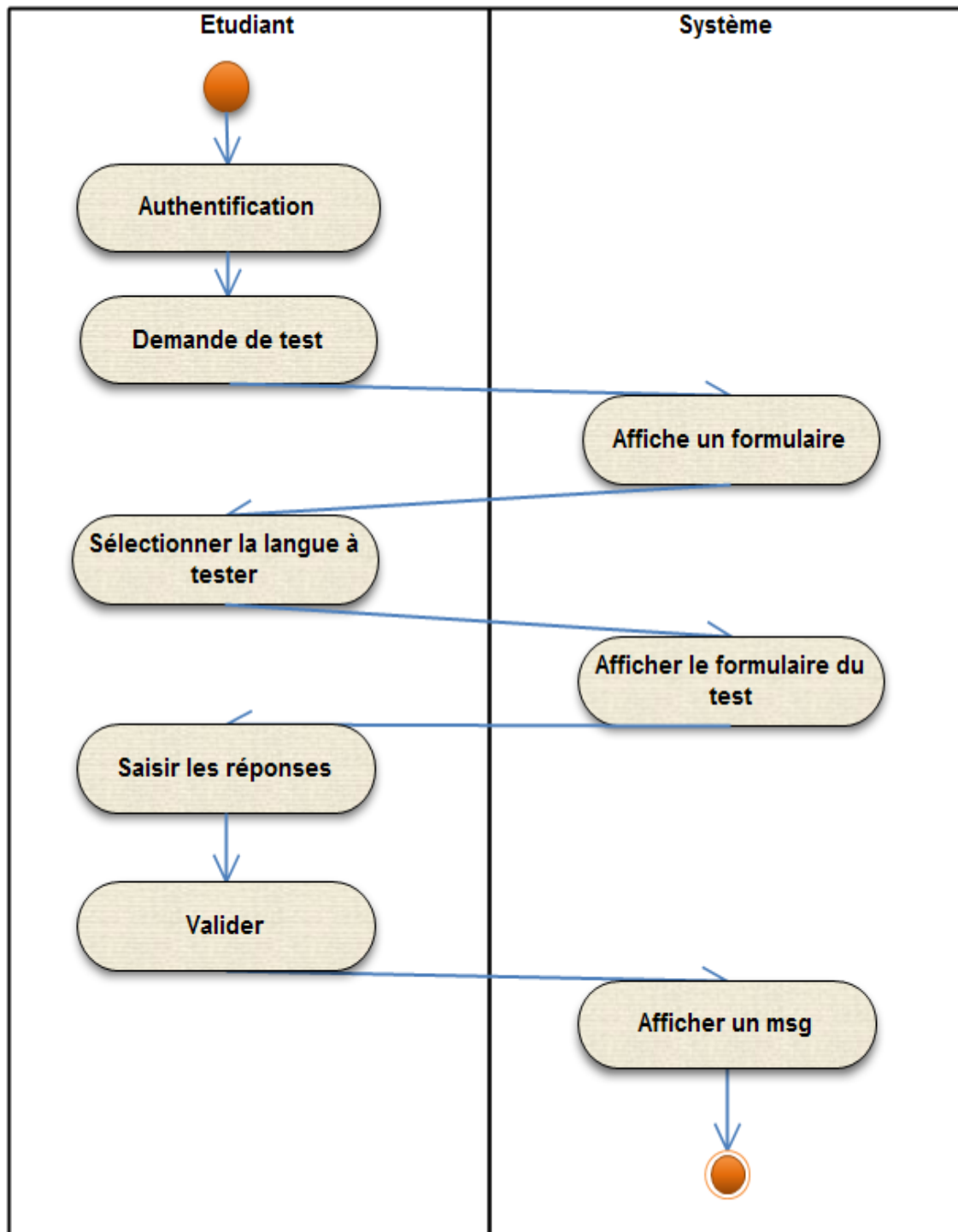


Figure 38: Diagramme d'activité du test.

➤ Consultation :

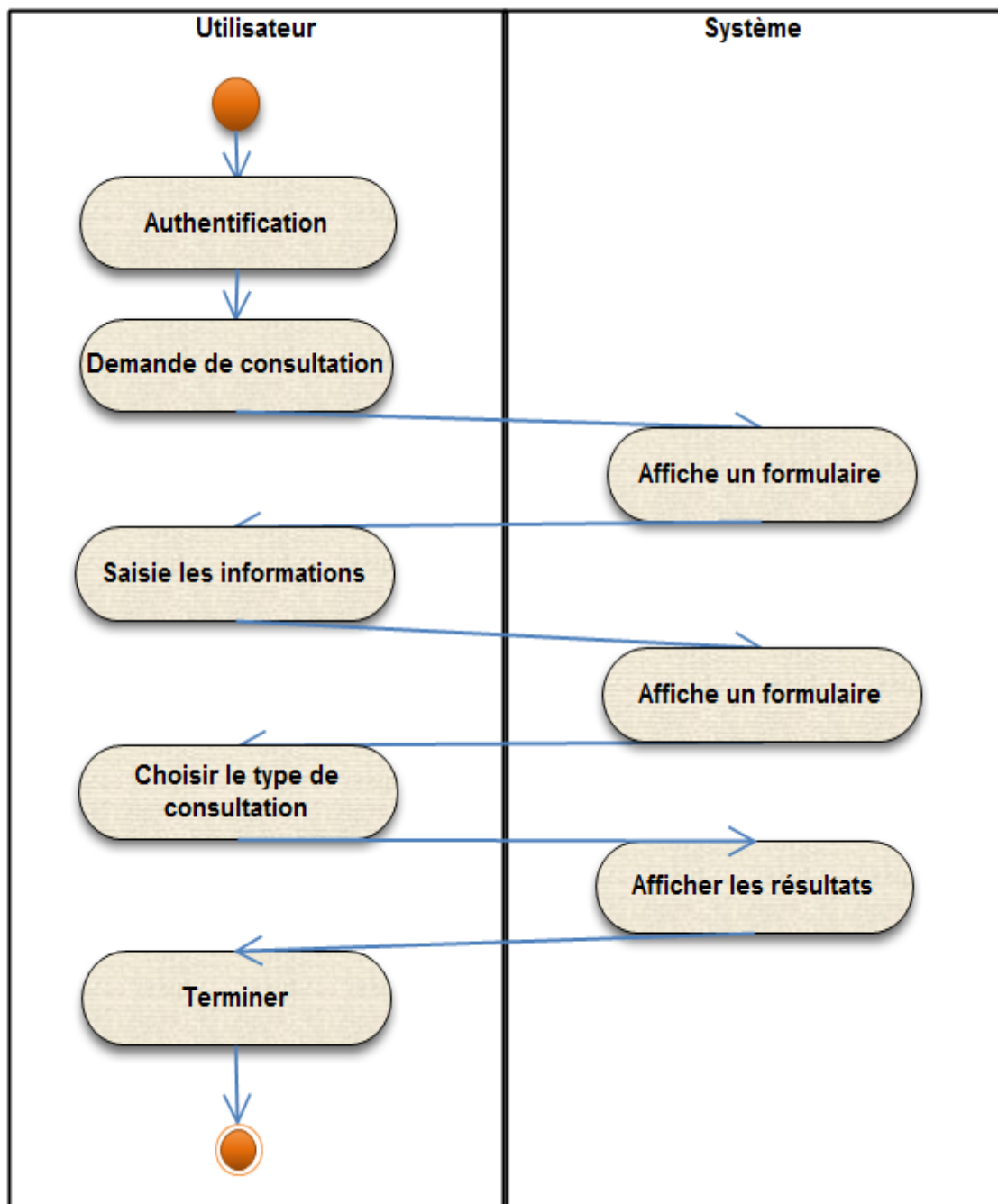


Figure 39: Diagramme d'activité de consultation.

- L'interface Homme Machine que se présente à l'administrateur :

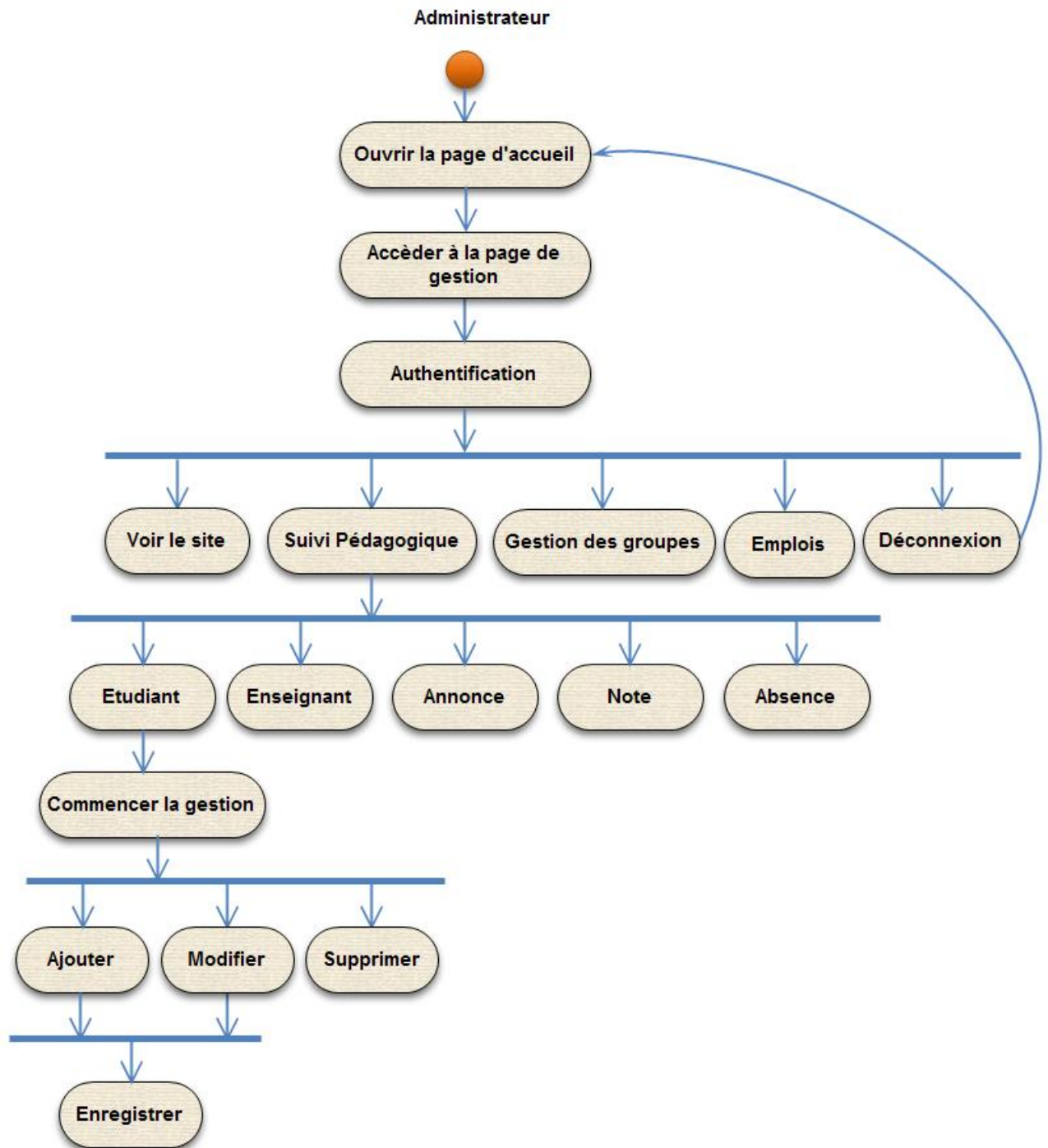


Figure 40: Diagramme d'activité de l'IHM de l'administrateur.

5- Diagramme d'états transition :

Ces diagrammes permettent de décrire les changements des états des objets [12].

Dans notre système, le seul objet qui passe par plusieurs états est l'objet étudiant.

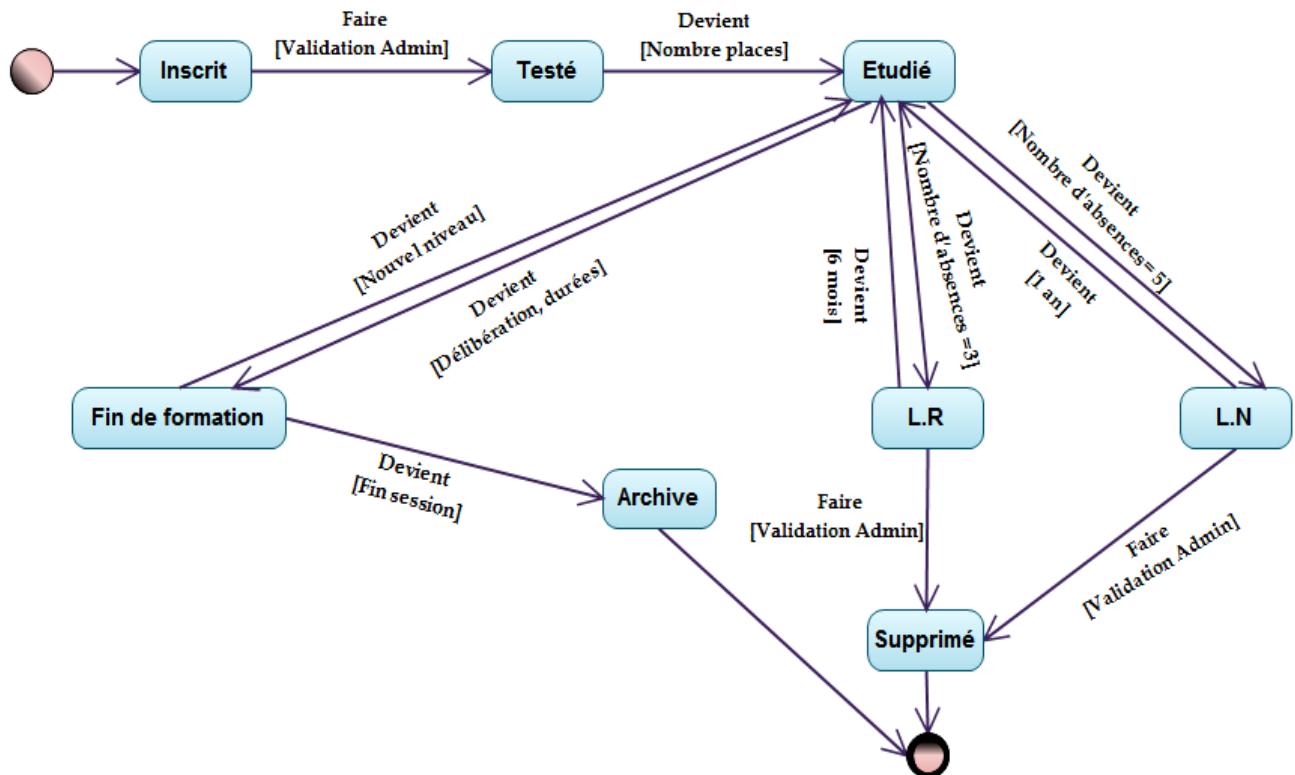


Figure 41: Diagramme d'états transition.

6-5. Diagramme de déploiement :

Le diagramme de déploiement suivant montre la dispositions physique des matériels qui composent le site web et la répartition des composants sur ces matériels [12].

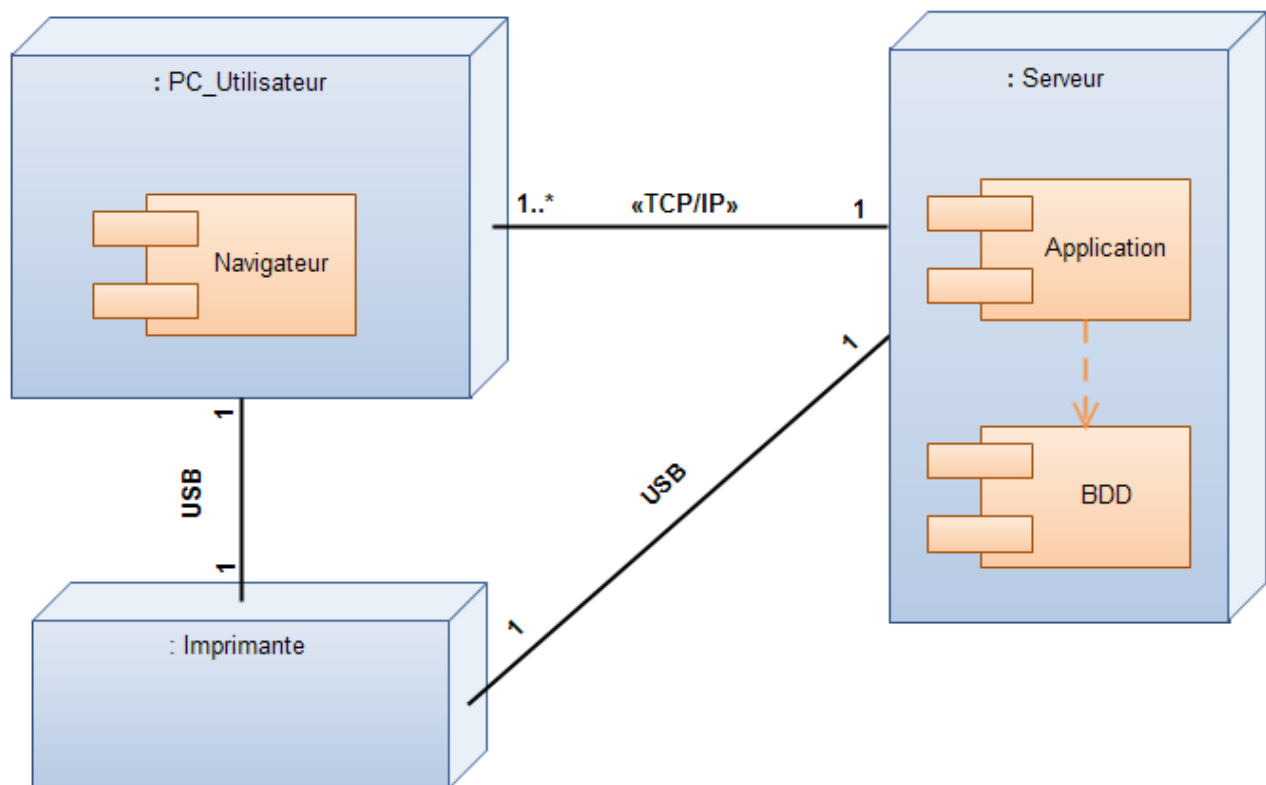


Figure 42: Diagramme de déploiement.

7- Diagramme de classe :

Un diagramme de classes est une collection d'éléments de modélisation statiques qui exprime de manière générale la structure statique d'un système, en termes de classes et de relations entre ces classes. Il montre les aspects statiques du modèle et fait l'abstraction des aspects dynamiques ou temporels [12].

Spécification des quelques algorithmes utilisés :

➤ Algorithme 1 :

Algorithme Ajoute_Etd ;

Ajouter (Nom, Prénom, Date_naiss, Adresse, @) : nouvel étudiant dans la BD ;

Tester **si** étudiant ∟ (LR ou LN) **alors**

Ajouter l'étudiant

Sinon

Echec d'ajout ;

Envoi un message d'erreur à l'administrateur ;

Finsi ;

Fin.

➤ Algorithme 2 :

Algorithme Saisie_Absence ;

Saisir_abs (Absence) : Liste d'absence ;

Chercher groupe ;

Afficher la liste des étudiants ;

Saisir l'absence **si** nbr_abs>3 **alors**

Afficher message à l'administrateur pour changer l'état de l'étudiant

Sinon

Enregistrer l'absence ;

Finsi ;

Fin.

Le diagramme de classes pour notre application est représenté sur la figure (Figure 41).

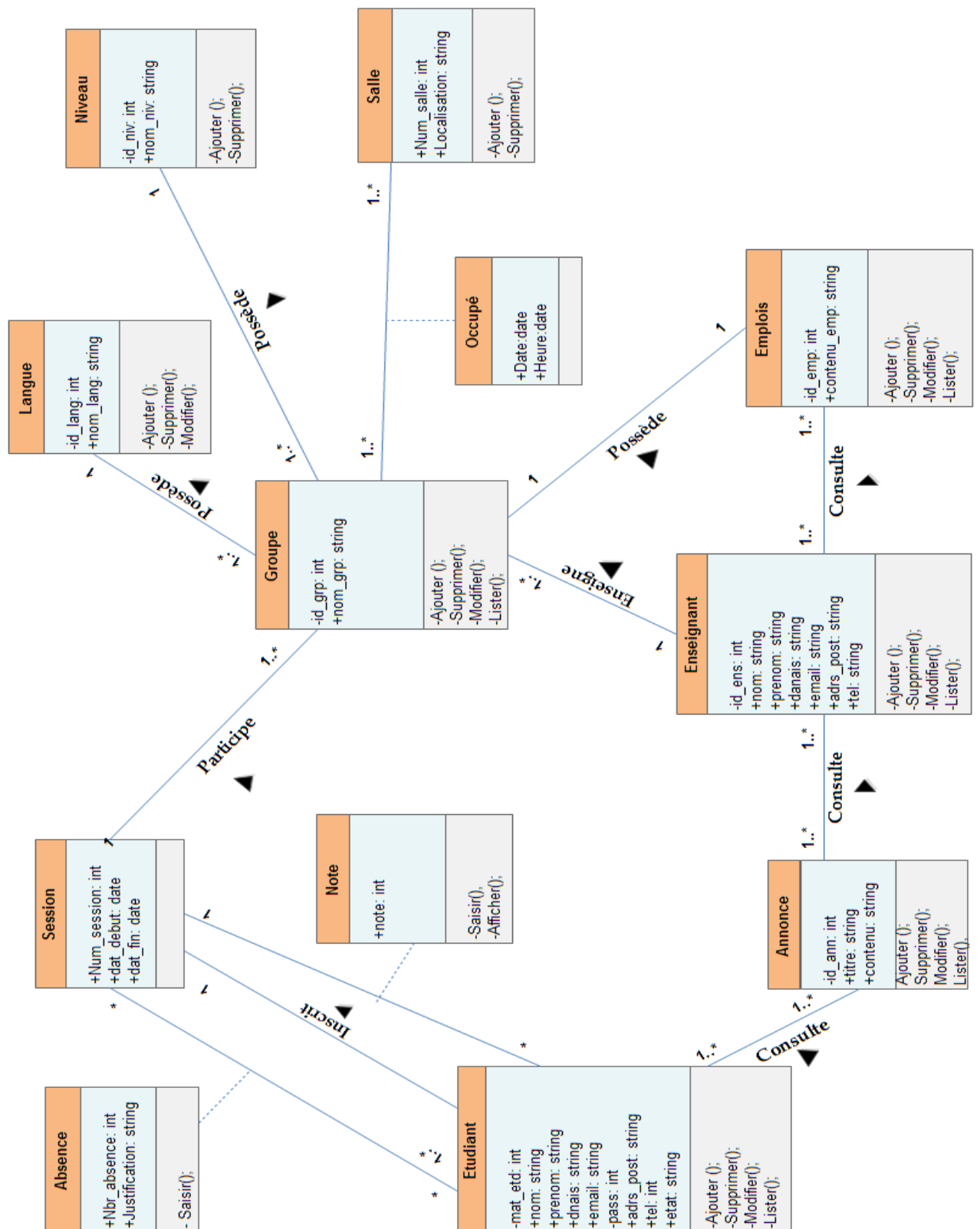


Figure 43: Diagramme de classe.

8- Le modèle relationnel :

Etudiant (Mat_etd, nom, prenom, dnais, email, pass, adrs, tel, etat).

Enseignant (id_ens, nom, prenom, danais, email, adrs_post, tel).

Groupe (id_grp, nom_grp).

Annonce (id_ann, titre, contenu).

Emploi (id_emp, contenu_emp).

Note (Mat_etd, Num_session, note).

Absence (Mat_etd, Num_session, Nbr_absence, Justification).

Session (Num_session, dat_debut, dat_fin).

Langue (id_lang, nom_lang).

Niveau (id_niv, nom_niv).

Occupé (id_grp, num_salle, date, heure).

Salle (num_salle, Localisation).

9- Conclusion :

Dans ce chapitre nous avons réalisé la conception de l'application en utilisant la démarche Agile qui est basée sur les concepts d'UML.

Nous avons utilisé quelques diagrammes de la démarche proposée pour obtenir une spécification formelle d'un système d'information. Seules les propriétés fonctionnelles du système sont effectivement prises en compte par cette méthode.

Nous avons utilisé les diagrammes : de cas d'utilisation, de séquence, d'activité, de déploiement, d'états transition et de classe pour bien représenter le système.

Finalement nous avons traduit le diagramme de classe en modèle relationnel pour passer à la partie implémentation.

1-Introduction :

Nous consacrons ce chapitre à la description de la phase de réalisation de notre Site Web. Nous présentons l'environnement de développement, ainsi que les outils et les technologies utilisées. Ensuite, nous discutons quelques points sur l'implémentation.

Ce dernier chapitre met en valeur l'environnement de développement, l'architecture de déploiement, l'aspect sécuritaire et enfin les interfaces utilisateurs. Ceci a pour but de montrer le résultat final afin de mieux évaluer le travail effectué.

2-Environnement de développement :

2-1 L'architecture MVC :

Principe:

Le modèle **MVC** (Modèle-Vue-Contrôleur) cherche à séparer nettement les couches présentation, traitement et accès aux données. Une application web respectant ce modèle sera architecturée de la façon suivante :

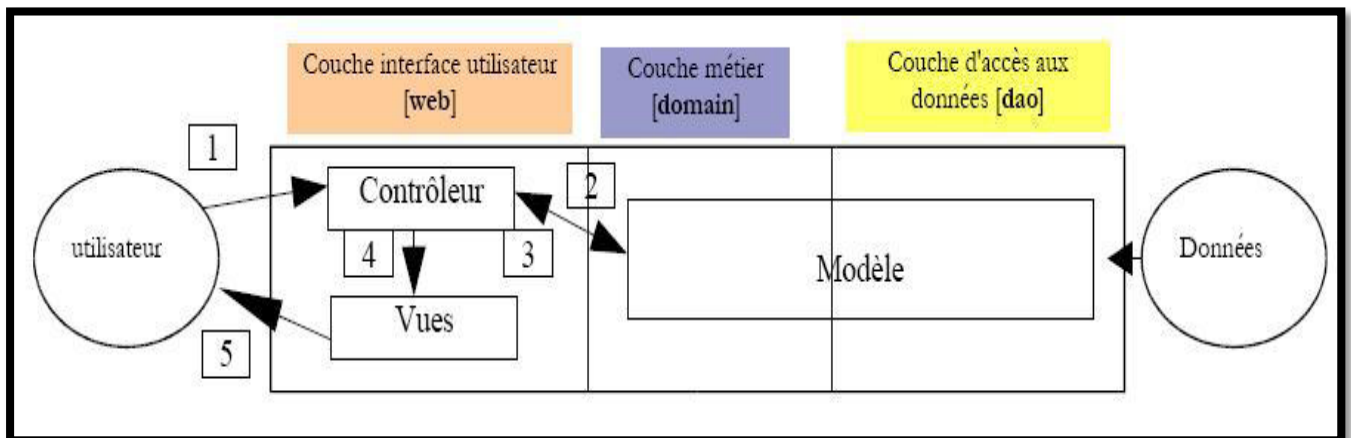


Figure 44: Le modèle MVC.

Le traitement d'une demande d'un client se déroule selon les étapes suivantes :

1. Le client fait une demande au contrôleur. Ce contrôleur voit passer toutes les demandes des clients. C'est la porte d'entrée de l'application.
2. Le contrôleur traite cette demande. Pour ce faire, il peut avoir besoin de l'aide de la couche métier.
3. Le contrôleur reçoit une réponse de la couche métier. La demande du client a été traitée. Celle-ci peut appeler plusieurs réponses possibles. Un exemple classique est :

Une page d'erreurs si la demande n'a pu être traitée correctement.

Une page de confirmation sinon.

4. Le contrôleur choisit la réponse (= vue) à envoyer au client. Celle-ci est le plus souvent une page contenant des éléments dynamiques. Le contrôleur fournit ceux-ci à la vue.
5. La vue est envoyée au client.

Une telle architecture est souvent appelée "architecture 3-tier" ou à 3 niveaux [13].

Notre application aura l'architecture MVC suivante pour l'étudiant:

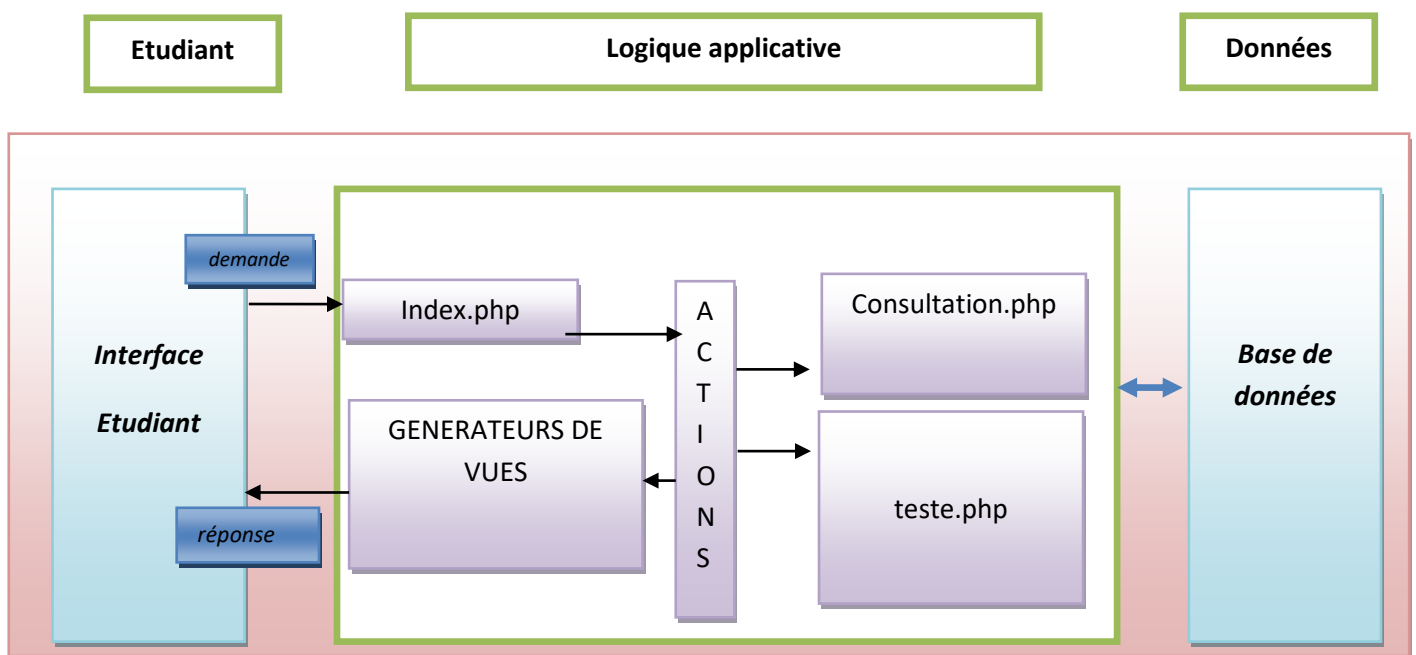


Figure 45: L'architecture de l'application par MVC.

2.2 Langages de programmation :

Pour le développement des applications web, deux catégories de langages de programmation existent. Les langages de programmation du côté serveur, et les langages de programmation du côté client.

Nous avons choisi **PHP** (**PHP Hypertext Preprocessor**) comme langage de programmation du côté serveur. Le choix s'est porté sur PHP pour les raisons suivantes [14], [15] :

1. Open-source, gratuit et multiplateforme.
2. Spécialement conçu pour le développement des applications web.
3. Facilité d'utilisation et d'intégration avec les pages HTML.
4. Supporte de nombreuses bases de données (MySQL, Sqlite, PostgreSQL, etc.)
5. La plupart des serveurs web actuels supportent PHP (Apache, Microsoft® IIS, etc.)
6. Son importante communauté.



Et pour la programmation du côté client, JavaScript est évidemment le choix. Les codes écrits en JavaScript sont interprétés par le navigateur web (web browser). Il est généralement utilisé pour :

1. Contrôler la fenêtre du navigateur.
2. Réagir aux actions de l'utilisateur.
3. Valider les données des Forms HTML.
4. Manipuler le contenu de la page HTML.

Nous n'avons pas utilisé le JavaScript pure, mais une bibliothèque appelée **jQuery**. Cette bibliothèque offre les avantages suivants [16] :

1. Cross-Browser (les mêmes lignes de code peuvent être exécutées sur tous les navigateurs sans aucun problème).



2. Supporte Ajax et rend très facile son utilisation.
3. Simplifie l'accès aux éléments d'une page HTML (par rapport au JavaScript pure).
4. Permet la manipulation des éléments d'une page (création, suppression et modification).
5. Des animations et des effets visuels.

2.3 Système de gestion de bases de données :

Le choix le plus naturel d'un SGBD avec PHP est MySQL. PHP/MySQL est le couple le plus utilisé pour le développement web [17].

1. Portabilité (UNIX, Windows, MacOS X).
2. Haute vitesse d'exécution.
3. Scalabilité (peut supporter des dizaines de millions d'enregistrement).
4. Facilité d'installation et d'administration (par rapport aux autres SGBD).
5. Multi-threading (peut gérer plusieurs transactions simultanément).



2.4 Serveur web :

Avec PHP/MySQL, le serveur web le plus adapté et le plus utilisé est Apache. Ce triplet (PHP, MySQL, Apache) est rassemblé dans le paquet WAMP (**W**indows **A**pache **M**ySQL **P**HP). WAMP est un projet open source gratuitement disponible sur <http://www.wampserver.com>. Le serveur web Apache offre les avantages suivants :

1. Open source et gratuit.
2. Très bonnes performances.
3. Portabilité (sur Windows et la plupart des systèmes UNIX).
4. Extensible, modulaire et configurable.



2.5 Outils supplémentaires utilisés :

phpMyAdmin

L'outil **phpMyAdmin** offre une interface graphique très simple pour l'administration complète des bases de données MySQL. Nous l'avons utilisé pour l'élaboration, la mise à jour et la maintenance de notre base de données. Nous mentionnons quelques fonctionnalités qu'il offre :

1. Gestion (création, modification et suppression) des bases de données.
Importation et exportation des bases de données.
2. Gestion des tables et des colonnes.
3. Gestion des views et des indexes.
4. Gestion des utilisateurs de la base de données et leurs droits d'accès.



5. L'exécution des requêtes SQL (écrits par les utilisateurs).

L'insertion, la mise à jour et la suppression des enregistrements de la base de données.

Notepad++

C'est un éditeur de texte open source et gratuit. Il est distribué sous les termes de la licence GPL (General Public License) de GNU. Il est gratuitement disponible sur <http://notepad-plus.sourceforge.net>. Nous l'avons utilisé pour le développement du code PHP, JavaScript, HTML et CSS. Il reconnaît les syntaxes de tous ces langages et les affiche de manière compréhensible et lisible.



EdrawMax

Le langage UML est supporté par différents outils, qui permettent:

- D'effectuer la transition des différentes vues vers le code et du code vers les Différentes vues. Dans le premier cas, les outils sont capables de générer des squelettes de code à partir des vues UML. Dans le second cas, il s'agit d'engendrer des vues UML à partir du code.
- La détection automatique de l'incohérence entre les différentes vues [11], [12].



Actuellement, on trouve dans le marché une vingtaine d'outils de modélisation UML parmi lesquels on a choisi d'utiliser l'outil **EdrawMax_v6.5.3** version portable (Figure 44).

Programme polyvalent de diagrammes, avec des caractéristiques qui le rendent parfait non seulement pour les diagrammes de flux, organigrammes, diagrammes d'affaire et graphiques, mais aussi des diagrammes de réseau, des plans de construction, les mind maps, des flux de travail, des conceptions de mode, des diagrammes UML, des diagrammes d'ingénierie électrique, illustration de la science [19].

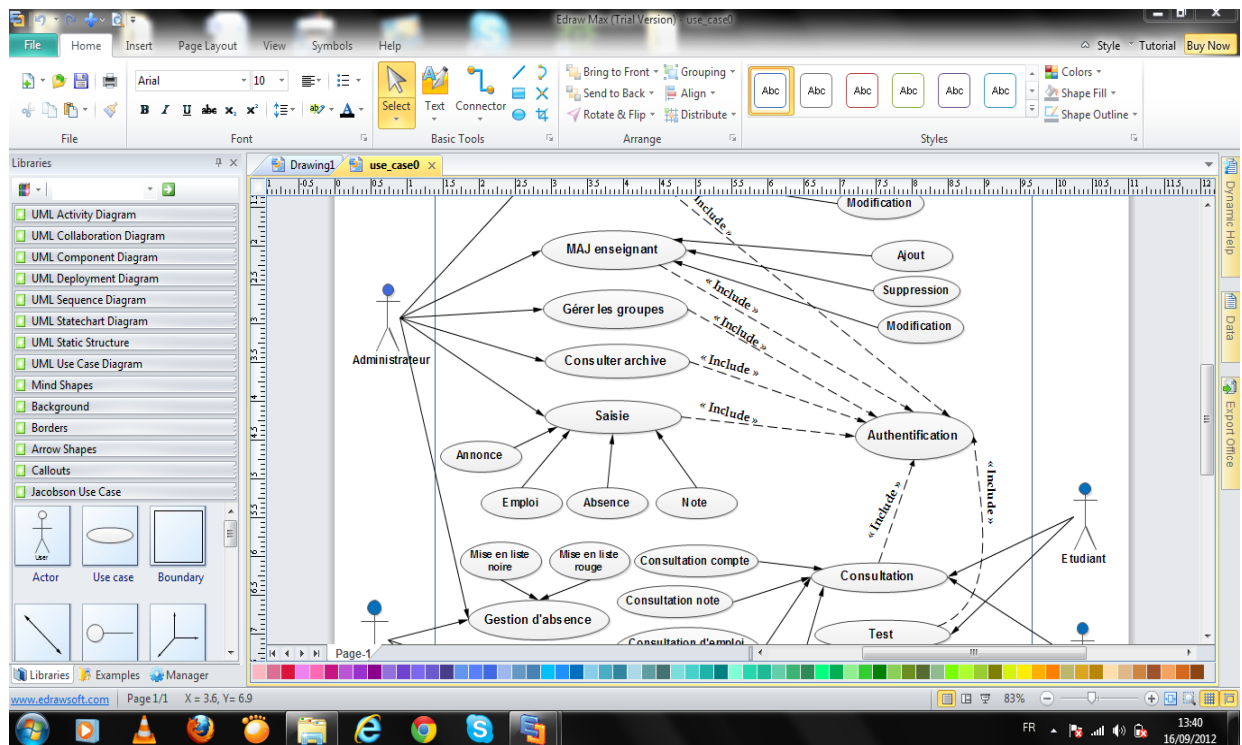


Figure 46: L'interface d'EdrawMax version 6.5.3.

Microsoft Excel

Microsoft Excel, dont le nom officiel actuel est Microsoft Office Excel,



est un logiciel de tableur de la suite bureautique Microsoft Office, développée

et distribuée par l'éditeur Microsoft ; il est destiné à fonctionner sur les plates-formes Microsoft Windows ou Mac OS X. Le logiciel Excel intègre des fonctions de calcul numérique, de représentation graphique, d'analyse de données (notamment de tableau croisé dynamique) et de programmation, laquelle utilise les macros écrites dans le langage VBA (Visual Basic for Applications) qui est commun aux autres logiciels de Microsoft Office [20].

2.6 Maquettes :

La présentation des maquettes était notre moyen concret de collecter les besoins vu que cette dernière est une représentation partielle du système à concevoir pour tester et valider certains aspects. Ainsi, la maquette permet de faire des choix, de définir plus précisément les obligations et les besoins et mieux imaginer sa future application.

Un aperçu des interfaces finales de l'application de gestion de C.E.I.L chose que nous allons montrer dans ce qui suit dans ce chapitre.

3- Visualisation des pages principales de l'application :

Nous allons décrire dans ce qui suit, le fonctionnement de système développé, en présentant quelques interfaces utilisateurs selon les modules cités en conception.

3-1. Page d'accueil :

Cette page est la page principale pour accéder à d'autres pages, elle contient la présentation du centre et l'enseignement, chaque utilisateur peut consulter cette page et voir les annonces du centre et l'emploi du temps. Pour les étudiants du centre l'accès à leur compte se fait à partir de cette page et pour l'administrateur aussi quand il veut accéder à sa page.



Figure 47: Capture d'écran montrant la page d'accueil.

3-2. Voir les annonces :

Cette page est pour consulter les nouvelles annonces du centre et on accède à partir de la page d'accueil.



Figure 48: Capture d'écran montrant la page où consulter les annonces du centre.

Exemple d'une annonce :

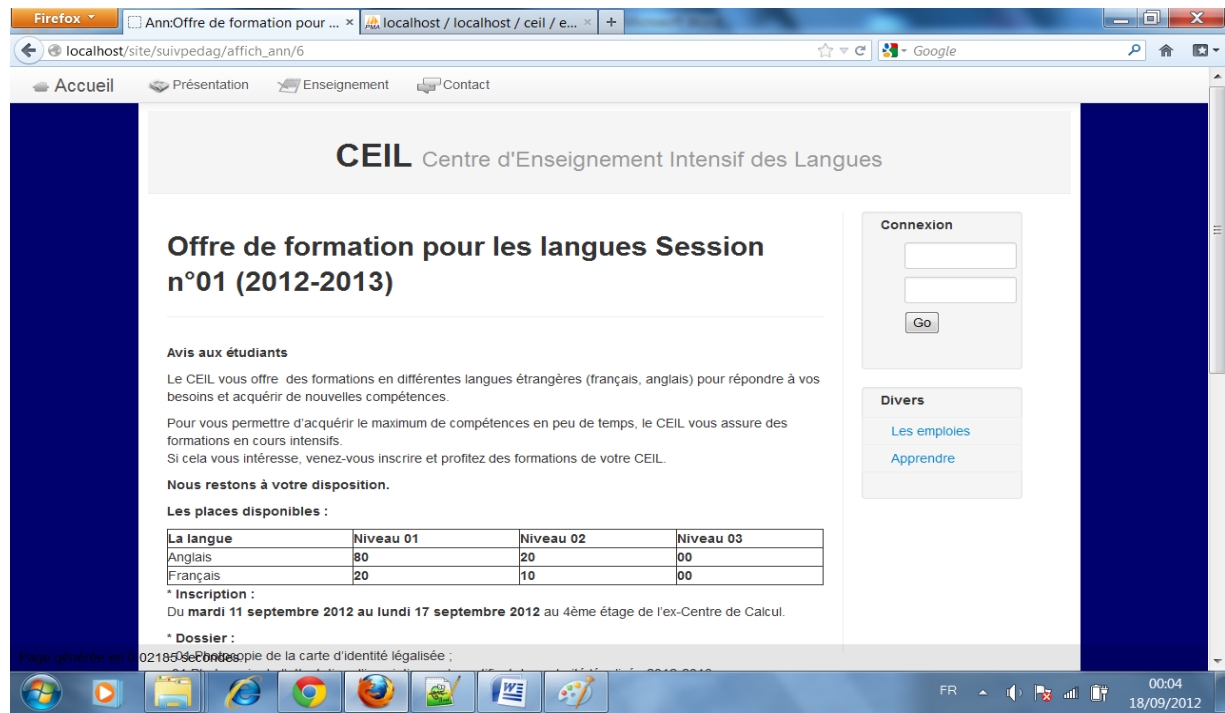


Figure 49: Capture d'écran montrant une annonces.

3-3. Page d'administrateur :

L'accès à cette page n'est pas autorisé à quiconque pour cela chaque personne désirant utiliser les fonctionnalités que présente notre système doit d'abord avoir le droit d'accès. Un nom d'utilisateur et un mot de passe doivent être fournis pour pouvoir se connecter. Par mesure de sécurité et de confidentialité des données cela présente plus un besoin pour l'utilisateur.

NB : ce n'est que l'administrateur qui peut accéder à l'application.

Pour accéder à cette page l'administrateur doit se connecter d'abord à partir de la page d'accueil (il écrit l'URL du page).

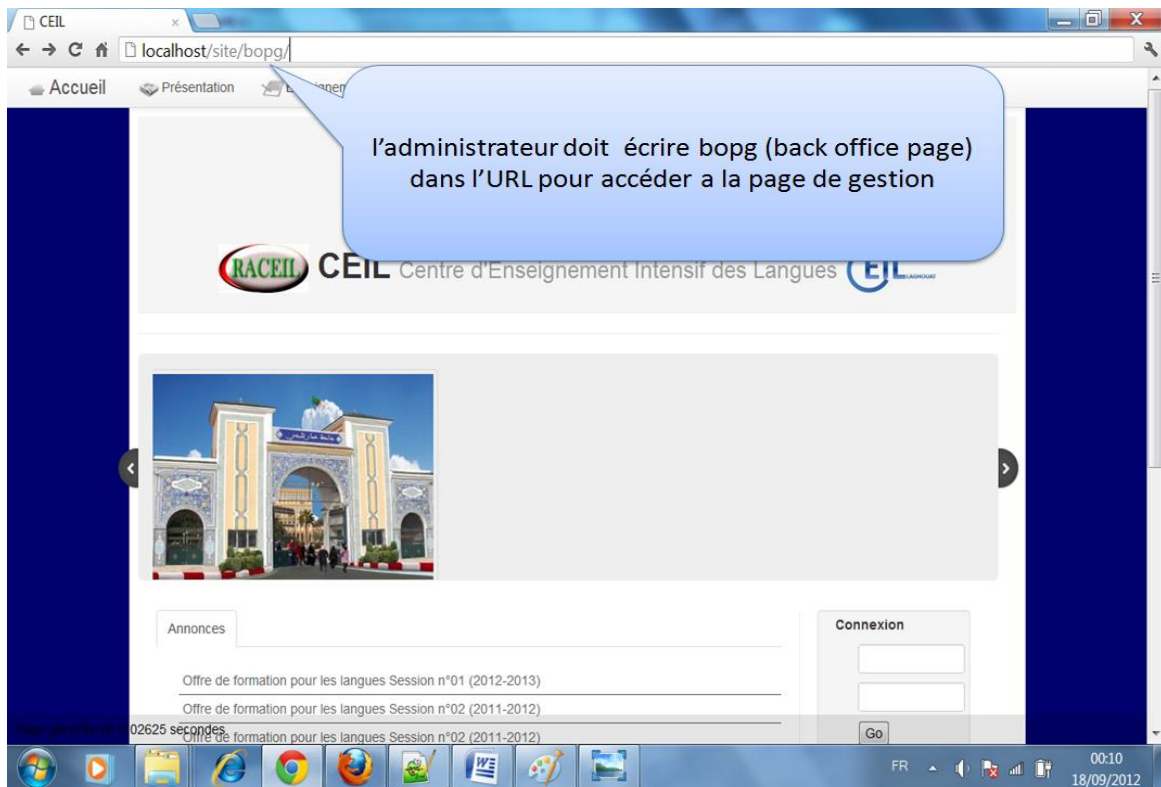


Figure 50: Capture d'écran montrant comment l'administrateur accède à son page.

Quand l'administrateur accède à la page il s'authentifie d'abord.



Figure 51 Capture d'écran montrant comment l'administrateur s'authentifie.

Maintenant on voit la page d'administrateur.



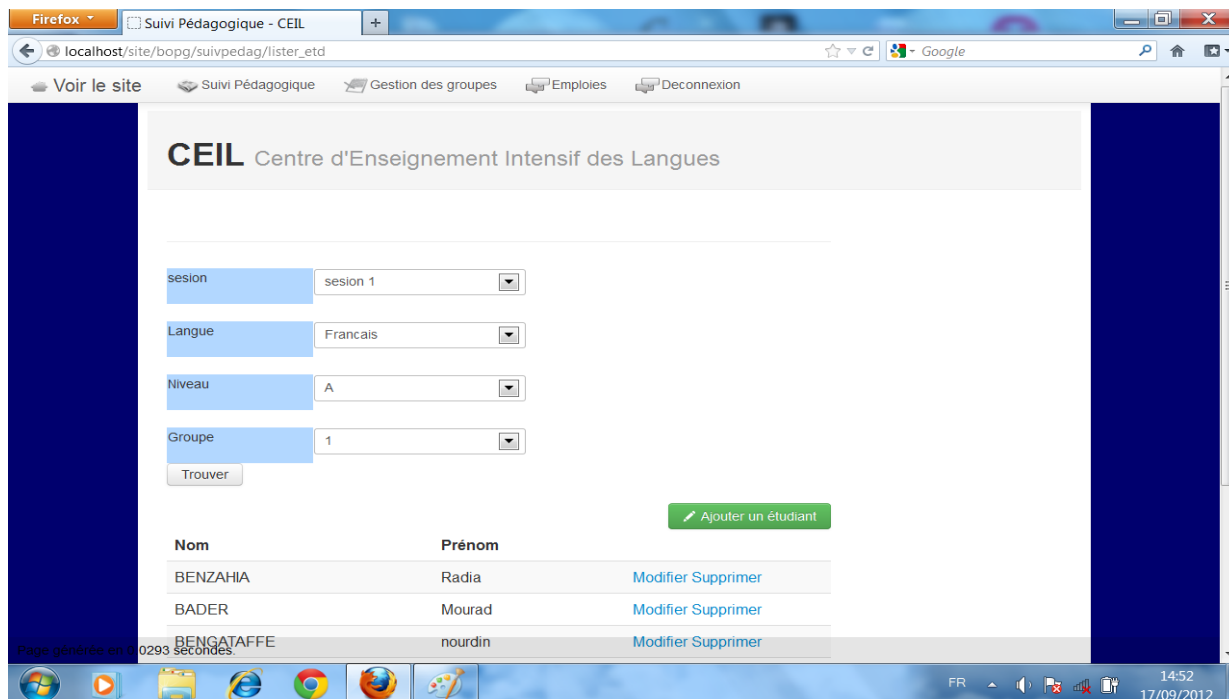
Figure 52: Capture d'écran montrant la page de l'administrateur.

Maintenant, on voit la page de "Suivi Pédagogique" qui permet à l'administrateur la MAJ de l'étudiant, l'enseignant, l'emploi du temps, l'annonce et la saisie des notes.



Figure 53: Capture d'écran montrant la page de l'administrateur.

On voit ici la page de MAJ d'étudiant.

**Figure 54:** Capture d'écran montrant la page de MAJ étudiant.

Cette figure si dessus montre la page d'ajout d'un étudiant.

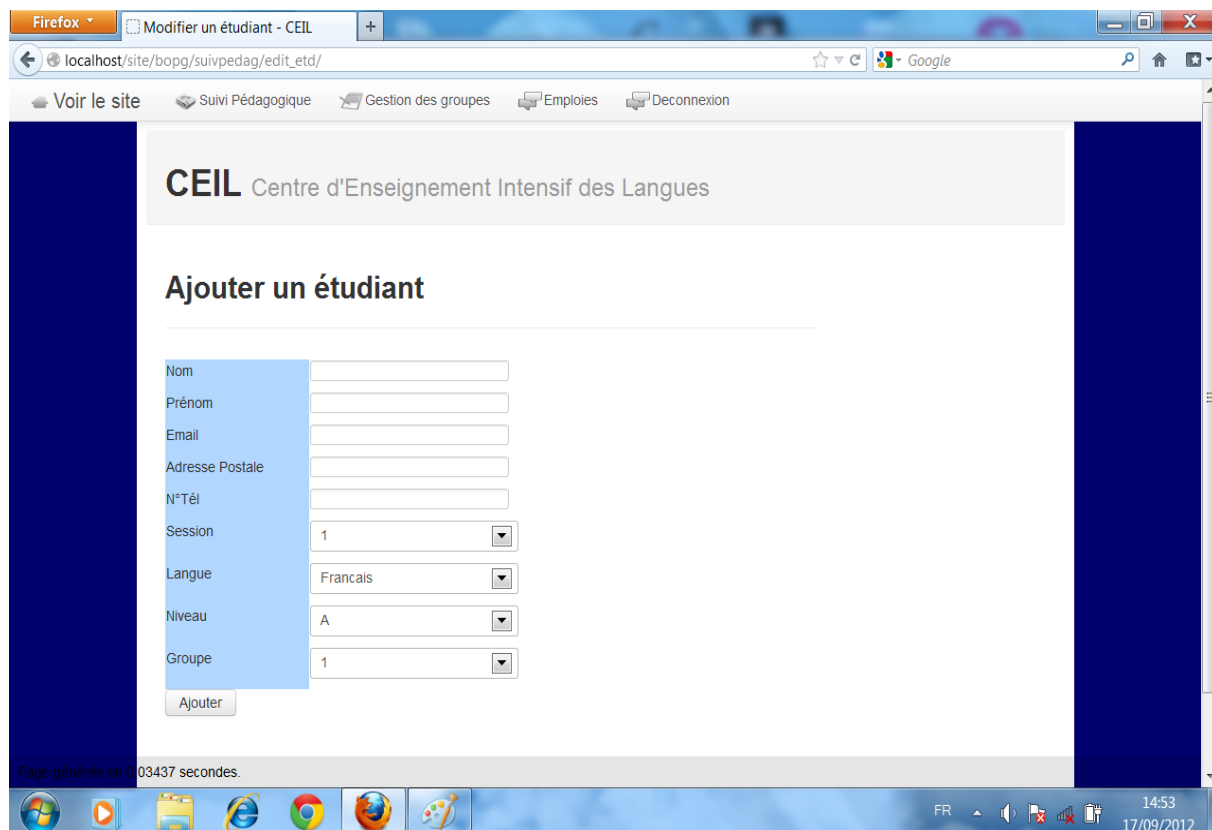
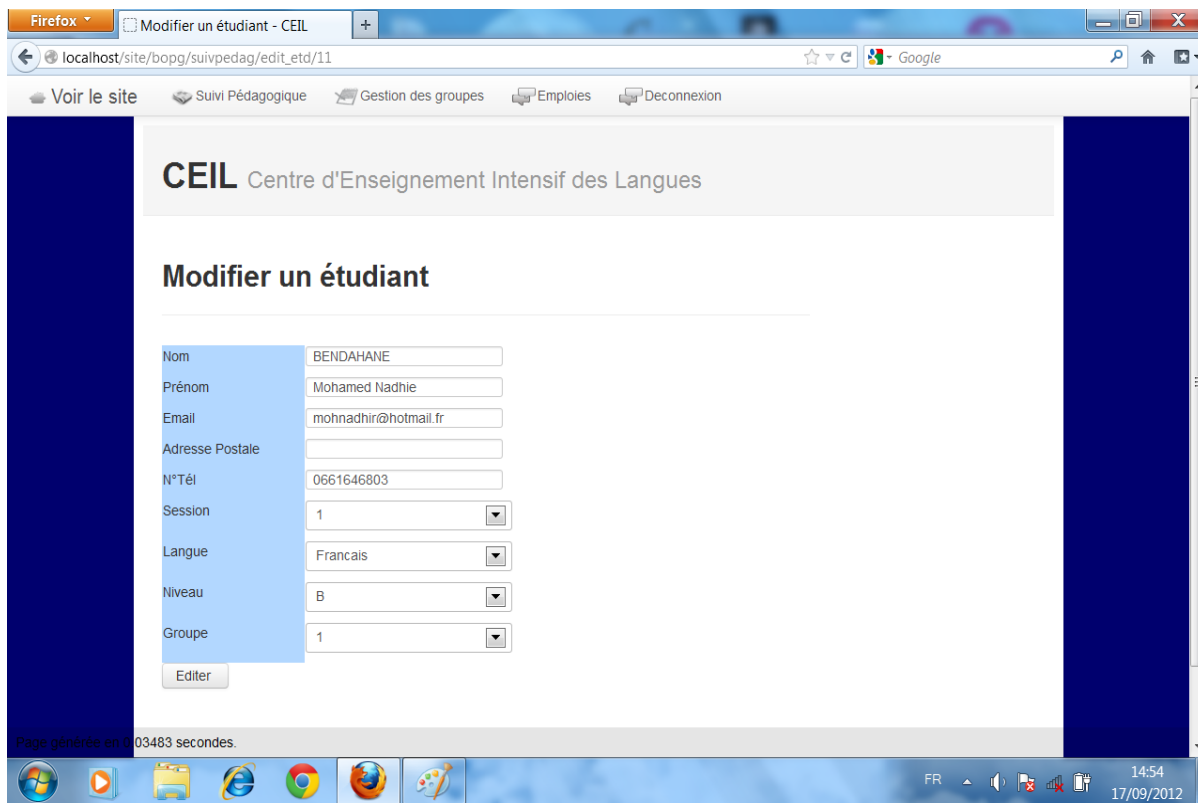


Figure 55: Capture d'écran montrant la page d'ajout d'un étudiant.

Cette figure si dessus montre la page de modification d'un étudiant.

**Figure 56:** Capture d'écran montrant de la page de modification.

Cette figure si dessus montre la suppression d'un étudiant.

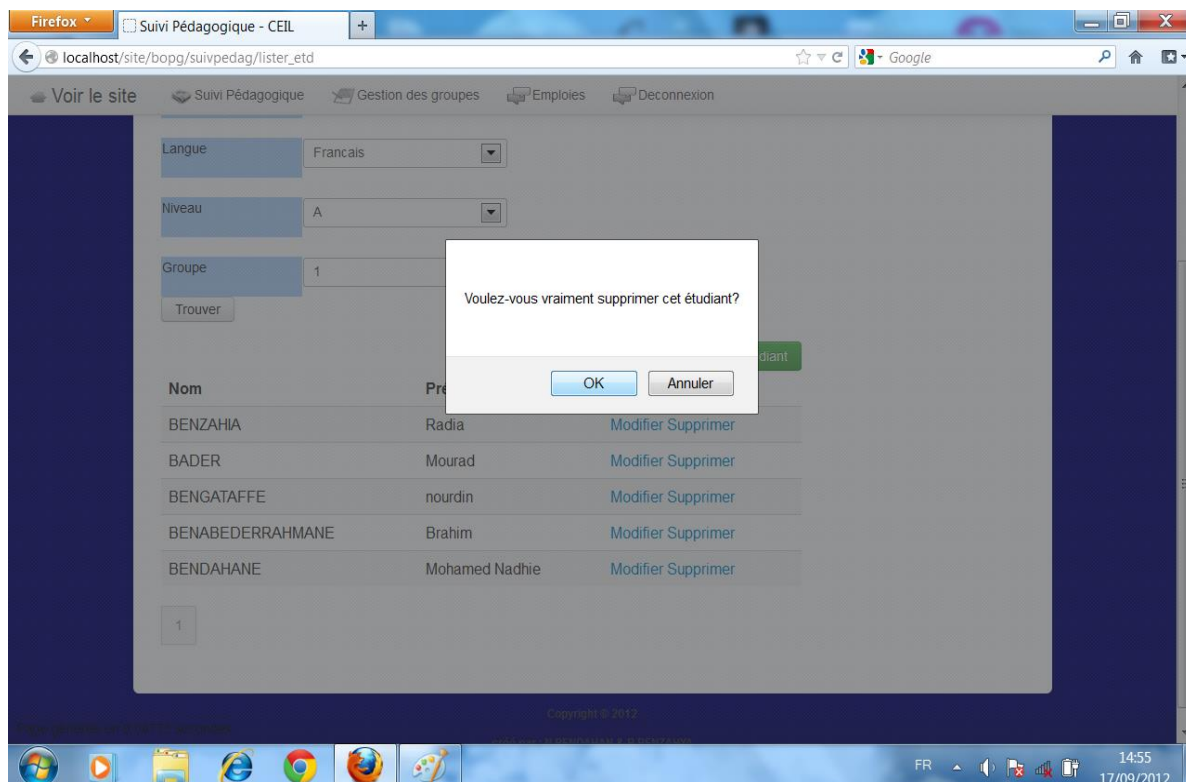
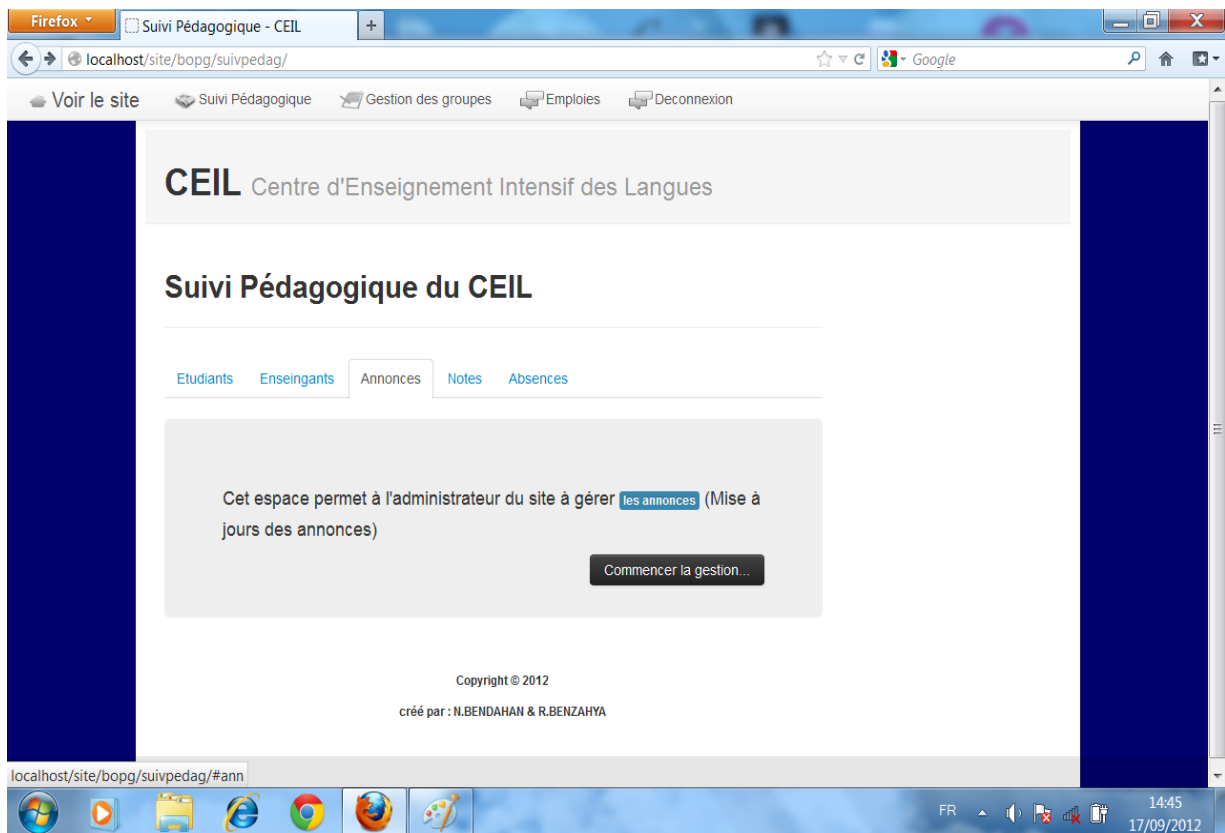


Figure 57: Capture d'écran montrant la suppression d'un étudiant.

Cette figure si dessus montre la page pour accéder à la page d'annonces.

**Figure 58:** Capture d'écran montrant comment accéder à la page d'annonce.

Cette figure si dessus montre comment créer une annonce.

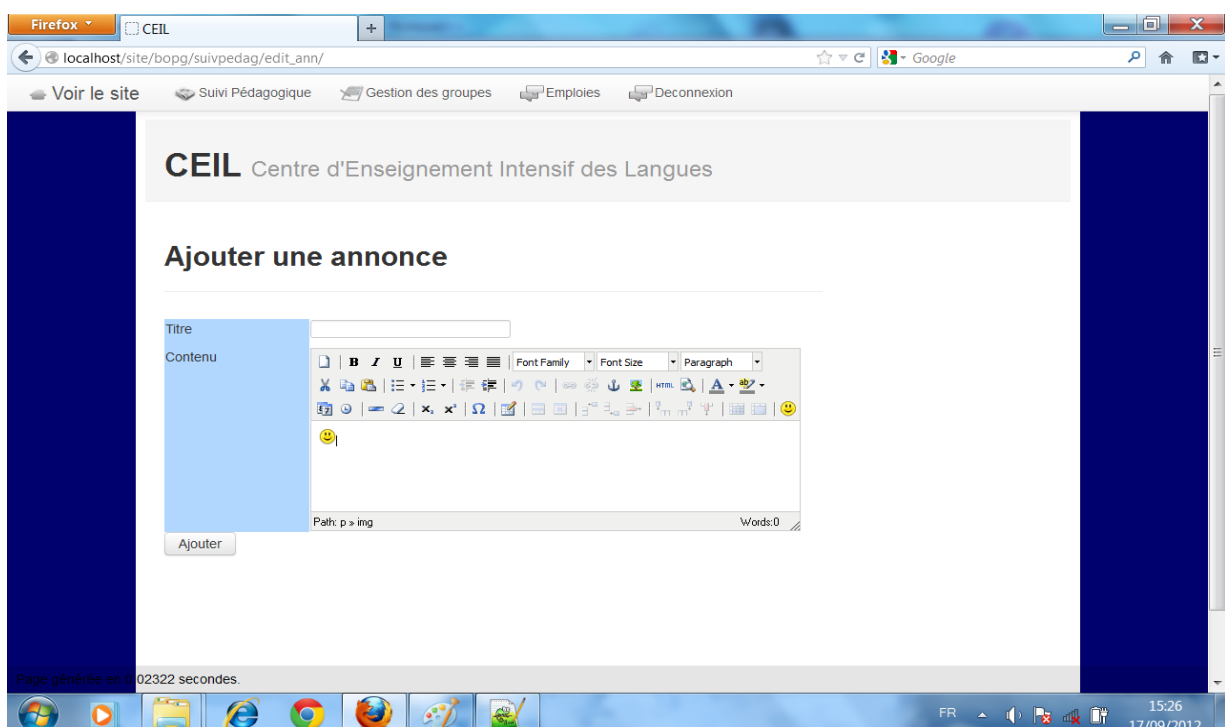


Figure 59: Capture d'écran montrant la page où créer une annonce.

Maintenant, dans les figures qui suit, on fait la gestion des groupes (figure 58, 59).

1. Ajouter groupe :

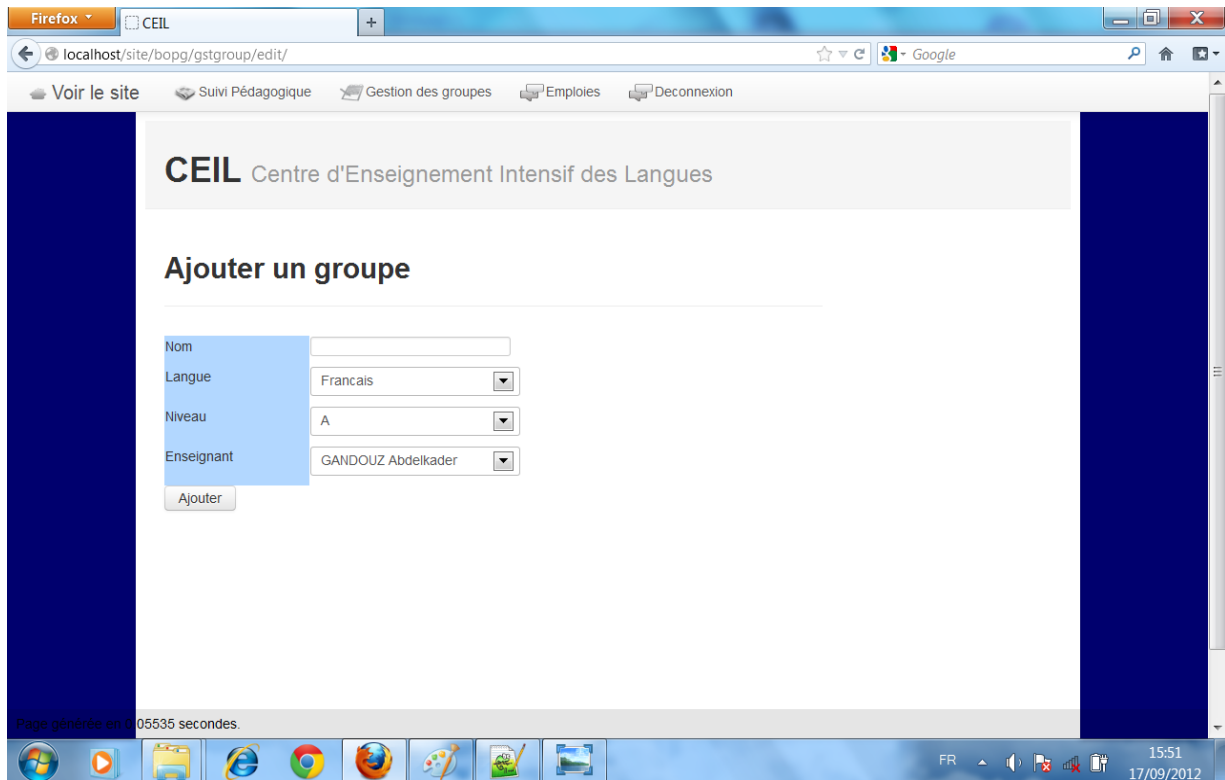


Figure 60: Capture d'écran montrant la page d'ajout des groupes.

2. Modifier groupe :

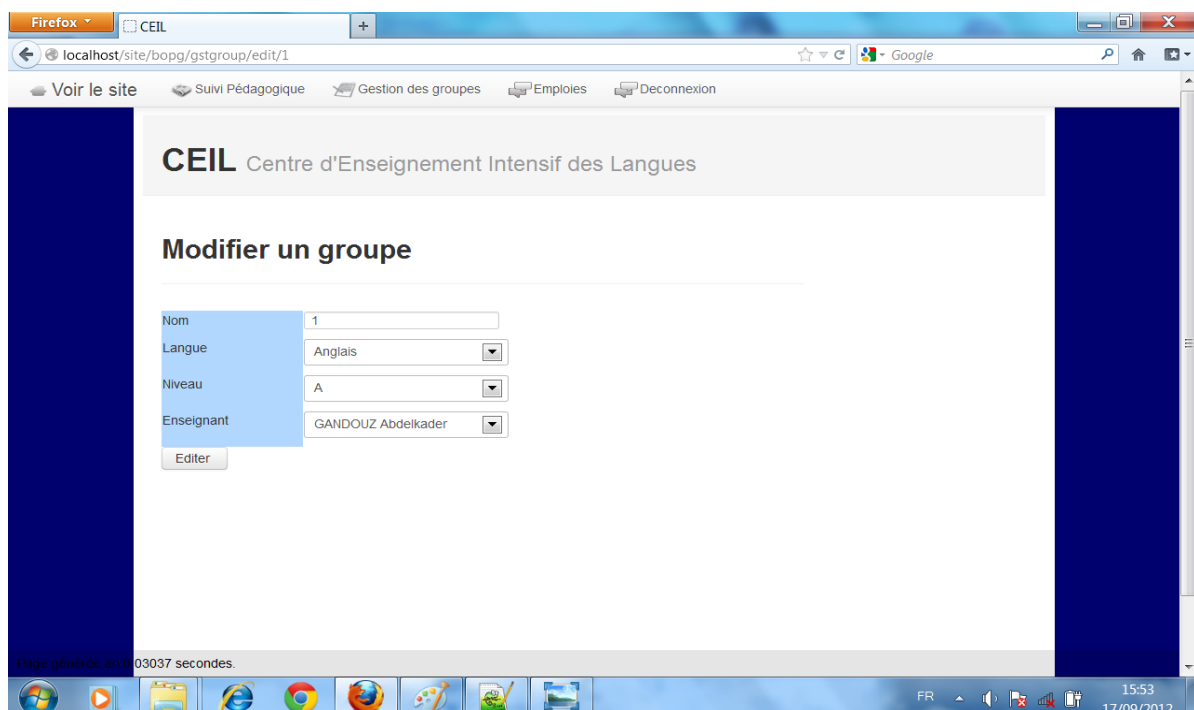
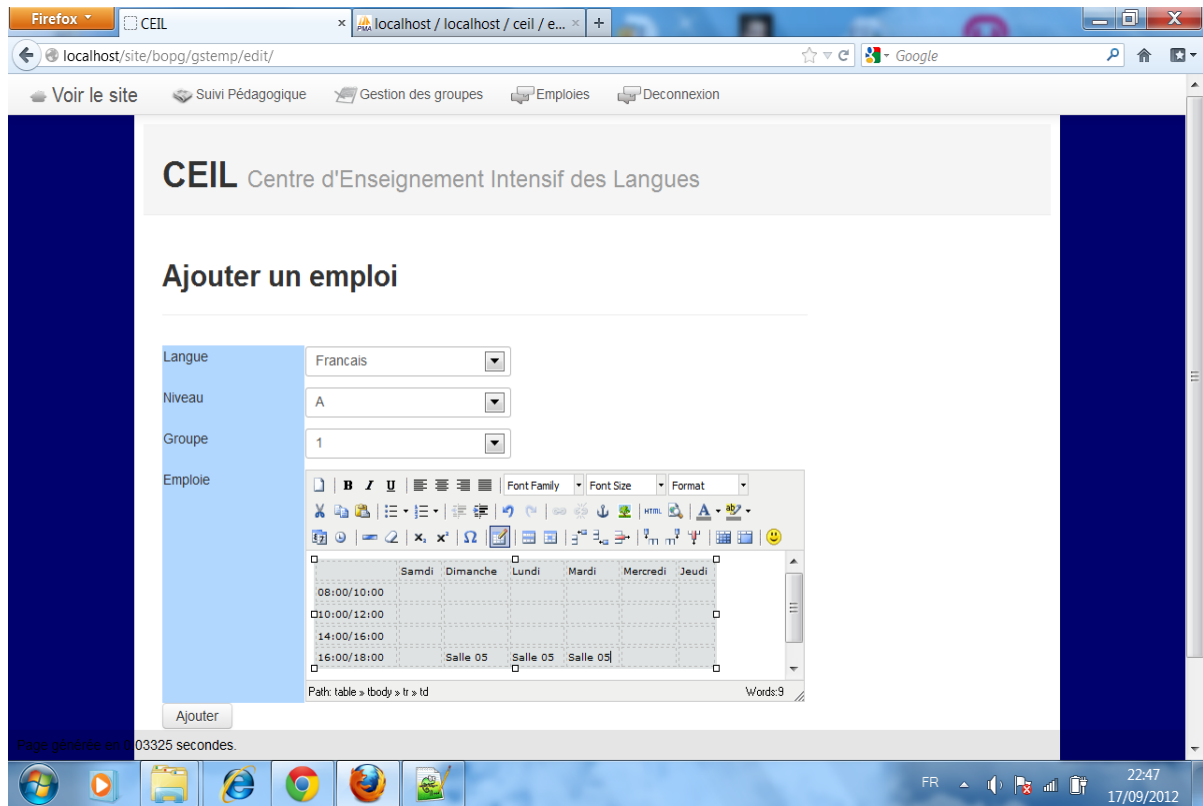


Figure 61: Capture d'écran montrant la page de modification groupe.

Maintenant, dans la figure qui suit, on ajoute un emploi du temps.

**Figure 62:** Capture d'écran montrant la page de création d'un emploi du temps.

Maintenant, on montre la déconnexion de l'administrateur.

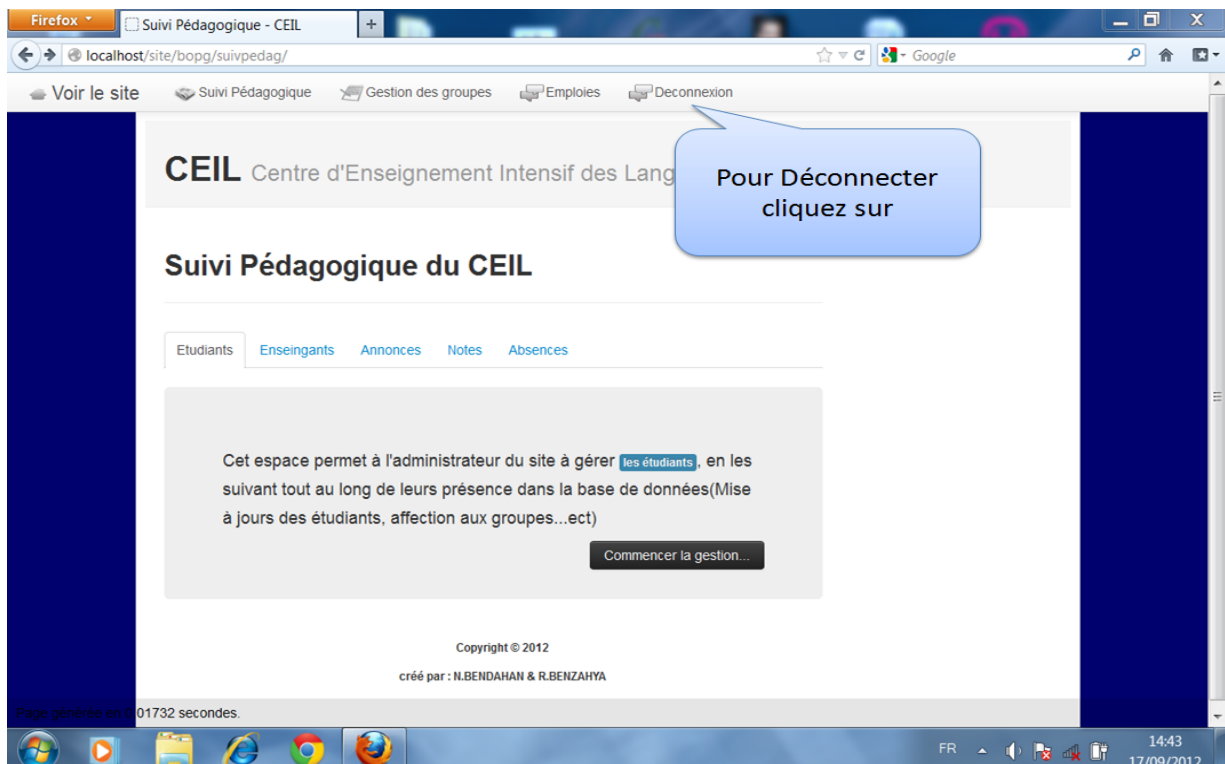
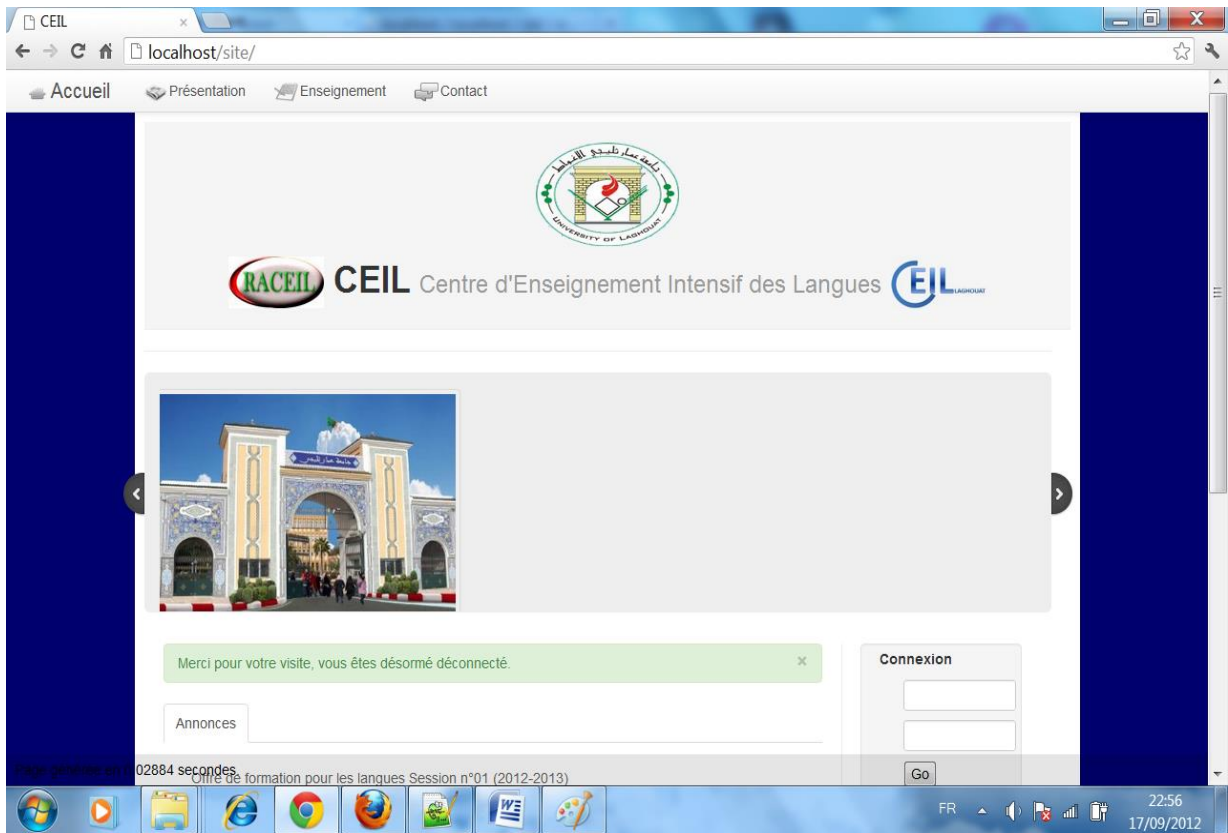


Figure 63: Capture d'écran montrant la déconnexion de l'administrateur.

Après la déconnexion de l'administrateur en revient à la page d'accueil.

**Figure 64:** Capture d'écran montrant la page d'accueil après la déconnexion de l'administrateur.

3- Conclusion :

Dans ce chapitre, on a montré quelques interfaces homme machine qui montre les fonctionnalités standard de système.

Globalement ce chapitre présente la version finale de notre application, une description des outils et de l'environnement de programmation utilisé lors de la réalisation de cette dernière.

Conclusion Générale et Perspectives

Un projet de fin d'étude est une première expérience pour un ingénieur afin qu'il puisse avoir une idée sur le genre de problèmes aux quel il sera confronté dans sa vie professionnelle. A travers ce présent travail que nous avons réalisé, nous avons eu cette idée.

Alors, notre projet consiste à réaliser une application sous web pour le centre d'enseignement intensif des langues (C.E.I.L) au sein de notre université Amar theliji à Laghouat.

Nous enveloppons notre modeste travail en faisant un bilan général sur ce que nous avons appris par ce travail, sur ce qui a été réalisé et sur ce qui reste à réaliser comme perspectives.

Ce présent travail nous a permis de cerner deux notions principales à savoir :

- Le couplage des architectures web et des bases de données pour faire naître un système d'information qui est d'une grande simplicité d'utilisation surtout du coté utilisateur puisqu'il ne nécessite pas de configuration spécifique, un simple navigateur fera l'affaire ;
- Nous avons aussi eu l'occasion d'acquérir un certain nombre de notions concernant la gestion des projets, les aspects essentiel et primitives d'un site web, la connaissance des différents outils que nous avons eu besoin dans notre travail dès le langage de programmation jusqu'à le logiciel de dessin des diagrammes et d'autres

Nous avons essayé le maximum pour atteindre nos but et réalisé le plus possible des fonctionnalités demandé par le centre ou lesquels nous avons proposée pour amélioré et facilité leur travail.

Nous avons parvenu à réaliser un Site Web qui permet une gestion de quelques fonctionnalités qui sont les suivantes :

- MAJ de l'étudiant et l'enseignant.
- La gestion du l'emploi du temps, du groupe et d'absence.
- L'identification de l'étudiant...ect
- La saisie des notes et d'annonce.

Néanmoins, tout travail peut être amélioré en ajoutant des fonctionnalités qui manquent, et des perspectives d'évolution, pour notre vision nous présentons les suivantes:

1. Les cours en ligne.
2. Une gestion automatique du test de niveau (la partie théorique a été réalisée).
3. Forum.
4. Une bibliothèque numérique avec des livres proposée par le centre pour aider les étudiants.

Tout au long de notre travail nous avons appris beaucoup de choses sur la gestion d'un centre des langues, qui est été un sujet très intéressant et aussi nous avons eu une idée bien précise sur le travail d'une équipe, la collaboration entre le travail des informaticiens et les échanges de leur idées pour toujours mieux atteindre les meilleurs résultats.

Bibliographie

Webgraphie



- [1] Directrice du Centre: S. ROUIS RAYYAH et Coordonnatrice pédagogique: S. LAHCENE, «**C.E.I.L: Centre d'Enseignement Intensif des Langues**», PDF représentant le centre de Laghouat.
- [2] « **Disco Encarta 2009** », Dictionnaire.
- [3] D. NANCI et B ESPINASSE, «**Ingénieur de Système d'Information**», *Merise*, 2^{ème} génération, P.61
- [4] NTUMBA, « **Note de cours de Méthode de Recherche Scientifique**, 2^{ème} Licence, *ESMICOM 2007 - 2008*, P. 25
- [5] Papy ZIRIMWABAGABO RUGAJO, «**Conception et développement d'un site web pour hôpital: Cas de l'hôpital général de référence de Kirotshe**» ISC Goma - Licence en Informatique de Gestion 2008, mémoire de fin d'étude.
- [6] Mohammed Amine Mostefai, «**Cours Génie Logiciel - Cours 2 - Cycles de vie**», ESI 30 Septembre 2011. P 23/29
- [7] Frédéric Bernard-Payen, «**Opportunité du Cycle de vie en cascade**».
- [8] Mouad DEBBAR, «**Conception et réalisation d'un portail web 2.0 d'annonces immobilières**», Mémoire de fin d'études, Pour l'obtention du diplôme d'Ingénieur d'Etat en Informatique, Option : Systèmes Informatiques, ESI.
- [9] Laurent AUDIBERT, « **UML 2 - de l'apprentissage à la pratique (cours et exercices)**», La version finalisée, largement enrichie et corrigée de cette première ébauche de cours est parue, dans la collection Info+, chez les éditions Ellipse.
- [10] Ruphin NYAMI, «**Conception d'une application web de gestion des équipements et matériels dans un réseau de sante**», Institut supérieur de statistiques - Licence 2010.
- [11] Pascal Rocques, «**Le Cahier du Programmeur**», 4^{ème} éd. EYrolles, 2002, 2007, 2008. P11/12
- [12] Pascal Roques, « **UML 2 - Modéliser une application web** », [4^{ème} édition], Groupe Eyrolles, année 2007.

- [13] Serge Tahé, «**Méthodologie de développement MVC d'une application PHP**», site developpez.com, Date de publication : Janvier 2004, Dernière mise à jour : 2 août 2008.
- [14] Toby Butzon, « **PHP By Example** », Edition Que, année 2002.
- [15] Matt Zandstra, «**Sam's Teach yourself PHP4 in 24 hours**», SAMS, Matt Zandstra, année 2000.
- [16] Steven Holzner, «**Visual Quick Start Guide jQuery**», Peachpit Press, année 2009.
- [17] Mark Matthews et Jim Cole et Joseph Gradecki, «**MySQL and Java Developer's Guide**», Wiley Publishing, année 2003.
- [18] <http://www.siteduzero.com/tutoriel-3-197288-introduction-a-php.html>
- [19] <http://fr.edrawsoft.com/>
- [20] http://wikipedia.org/wiki/Microsoft_Excel/
- [21] <http://fr.wikipedia.org/>
- [22] <http://www.sitedezero.com/>
- [23] <http://www.developpez.com/>
- [24] <http://www.uml.org/>
- [25] http://www.memoireonline.com/07/10/3657/m_Conception-dune-application-web-de-gestion-des-equipements-et-materiels-dans-un-reseau-de-sante0.html#toc0
- [26] <http://fr.slideshare.net/mostefaiamine/cours-gnie-logiciel-cours-2-cycles-de-vie>

Glossaire

- ❖ **Agile** : Une méthode agile est une approche **itérative** et **incrémentale**, qui est menée dans un esprit **Collaboratif**. Elle génère un produit de haute **qualité**, tout en prenant en compte l'évolution des **besoins** des **clients**.
- ❖ **Base de donnée** : Ensemble de données structurées (champs, catégories) et évolutives. Une base de données constitue une source d'information modifiable, exploitable et consultable à partir de requête pour créer et gérer des annuaires, des archives, des sites web, ...
- ❖ **Courrier électronique (e-mail)** : Echange d'informations entre deux utilisateurs distants utilisant pour ce faire un réseau informatique, Internet par exemple. Chaque utilisateur est identifié par une adresse électronique.
- ❖ **Forum de discussion** : Lieu d'échange thématique asynchrone dans lequel les utilisateurs peuvent apporter leurs contributions, soit sous forme de questions soit de réponses à des questions posées par d'autres. Certains forums sont réservés à des communautés fermées et demandent donc une identification des utilisateurs, d'autres sont modérés, d'autres non.
- ❖ **Interface** : Le terme interface désigne à la fois le dispositif matériel et logiciel qui permet à l'utilisateur d'interagir avec le système.
- ❖ **Les bases de données** : Les bases de données constituent le cœur des applications dynamiques du Web. C'est souvent des bases de données que le serveur Web obtient les informations actualisées destinées aux pages Web créées à la volée. Une base de données est un ensemble des données structuré qui sont fiables, cohérente et partagé par plusieurs utilisateurs.
- ❖ **Le protocole http** : Le but du protocole HTTP est de permettre un transfert de fichiers (essentiellement au format HTML) localisé grâce à une chaîne de caractères appelée URL entre un navigateur (le client) et un serveur Web (appelé d'ailleurs Httpd).

La communication se fait au-dessus des protocoles TCP. Le client demande une connexion avec le serveur à l'aide d'une requête en transférant l'URL du document demandé. Le serveur accepte la connexion et fournit au client le document correspondant à l'URL Demandé puis coupe la connexion.

- ❖ **Les systèmes de gestion de base de données :** Les systèmes de gestion de base de données SGBD est un logiciel permettant d'accéder aux données de la base de données. Cet accès fait appel à des demandes d'accès formulées par les utilisateurs. Le SGBD forme avec la base de données le système de base de données.
- ❖ **Le World Wide WEB :** C'est un ensemble de pages Web liées entre elles par des liens hypertextes formant une énorme toile d'araignée mondiale de l'information, qui permet quand on y accroche un fil, de tirer à soi toute la toile (Web). Un simple click de souris, nous permet de basculer d'un document Web à un autre, tout en appelant des textes, des images, des vidéos, des sons, etc. Et cela sans avoir recours aux touches de clavier ni aux commandes particulières.
- ❖ **Login :** Nom spécifique attribué à un utilisateur et couplé à un mot de passe pour garantir la sécurité des échanges sur un serveur ou sur un réseau.
- ❖ **PHP :** PHP est un langage de script « embarqués » (*embedded*) dans HTML, exécuté côté serveur. L'essentiel de sa syntaxe est emprunté aux langages C, Java et Perl, avec des améliorations spécifiques. L'objet de ce langage est de permettre aux développeurs WEB d'écrire des pages dynamiques rapidement et de les interfacer aux bases de données. Notons qu'aujourd'hui, les capacités de PHP vont bien au-delà de la génération de pages HTML. PHP génère des documents PDF, des images ou même des animations Flash à la volée.
- ❖ **Plate-forme :** système informatique qui comprend un ensemble d'outils de communication synchrone et asynchrone, un système de partage de fichiers, de mise en ligne de documents, un système de gestion des inscriptions, des évaluations des apprenants, etc.
- ❖ **Site :** Ensemble de pages HTML contenant des textes et des éléments multimédias. On y accède grâce à un navigateur par une adresse (URL).
- ❖ **UML :** « *Unified Modeling Language* » :se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier, concevoir des solutions et communiquer des points de vue.
- ❖ **Un browser (navigateur) :** C'est un logiciel de consultation (ex : Netscape et Internet Explorer) s'exécute sur tous les types de plates-formes (Macintosh, Unix, PC). Il dialogue avec un serveur selon un protocole spécifique (http, ftp...), qui va lui fournir l'information structurée en code HTML.

- ❖ **Un serveur Web** : Ce logiciel est en permanence à l'écoute des requêtes que vont former les clients (via http) Il en vérifie la validité, s'assure que la demande émane d'un client autorisé à accéder au document, et lui envoie finalement textes et images avant de fermer la connexion.
- ❖ **UP : processus unifié** : La complexité croissante des systèmes informatiques a conduit les concepteurs à s'intéresser aux méthodes de développement. Ces dernières ont toujours essayé d'apporter un contrôle continu sur un projet tout au long de son processus de vue.

Bien que des méthodes de développement existent depuis 30 ans (Merise, SADT), nous ne pouvons constater aujourd'hui l'existence d'une règle qui soit à la fois formelles et commune à toutes les cultures.

Le Processus Unifié (**PU** ou **UP** en anglais pour **Unified Process**) est une méthode de développement logiciel construite sur **UML** ; elle est *itérative* et *incrémentale*, *centrée sur l'architecture*, *conduite par les cas d'utilisation* et *pilotée par les risques*.

- **Itérative et incrémentale** : la méthode est dite itérative dans le sens où elle permet de faire des itérations lors de différentes phases, ceci garantit que le modèle construit à chaque phase ou étape soit affiné et amélioré. Chaque itération peut servir d'ajouter de nouveaux incréments.
- **Conduite par les cas d'utilisation** : elle est orientée utilisateur pour répondre aux besoins de ce dernier.
- **Centrée sur l'architecture** : tout système complexe doit être décomposé en parties modulaires afin de permettre une maintenance et une évolution facilitée c'est-à-dire les grandes mailles, l'architecture de type qui sera retenue pour le développement, l'implémentation et en suite le déploiement du système.
- **Pilotée par les risques** : en définissant les priorités pour chaque fonctionnalité, on peut minimiser les risques d'échec du projet.

UP répète un certain nombre de fois une série de cycle qui s'articulent sur 4 phases :

1. **Préétude** (Inception) ou **Analyse de besoins** : c'est à ce niveau qu'on évalue l'un petit plus à ajoutée du développement et la capacité technique à le

réaliser (étude de faisabilité). L'analyse de besoins donne une vue du projet comme un produit fini et surtout elle fait face aux questions suivantes:

- Que va faire le système ?
- Par rapport aux utilisateurs principaux, quel service va-t-il rendre ?
- Quelle va être l'architecture générale (cible) de ce système ?
- Quels vont être : les délais, les coûts, les ressources, les moyens à déployer ?

2. **Elaboration** : c'est ici que sera confirmée la concordance parfaite du système aux besoins des utilisateurs et à livrer l'architecture de base ou stable.

3. **Construction** : sert à livrer progressivement toutes les fonctions du système c'est-à-dire l'architecture de référence se métamorphose en un produit complet ayant tous les cas d'utilisations mis en place.

4. **Transition** : déployer le système sur des sites opérationnels, et le produit étant en version beta, d'autres erreurs peuvent être détectées par les utilisateurs d'où la nécessité d'une formation, la mise en place de l'assistance et correction d'erreurs.

Le résultat de chacune d'itérations de phase précédentes, est un système testé, intégré et exécutable.

L'approche itérative est fondée sur la croissance et l'affinement successifs d'un système par le biais d'itérations multiples. Le système croît avec le temps de façon incrémentale, itération par intention, et c'est pourquoi l'acronyme de méthode itérative et incrémentale. Il s'agit là d'un principe primordial et la devise même du Processus Unifié.

Toutes ces activités du processus de développement sont définies par six (6) disciplines qui décrivent la *capture des besoins*, la *modélisation métier*, l'*analyse et la conception*, l'*implémentation*, et en fin le *test de déploiement*.

Signalons aussi que ces différentes étapes peuvent se dérouler à travers plusieurs phases.

Le processus unifié doit donc être compris comme une trame commune des meilleures pratiques de développement.

Par ailleurs des méthodes séquentielles comme celles se basant sur le cycle en V, ont vite révélé leur limite dans un environnement régi par des changements réguliers, impliquant une quasi impossibilité de revenir en arrière, et de ce fait laissant une très petite marge d'erreur.

Questionnaire

Thème :

Conception et implémentation d'une application sous Web pour la gestion des étudiants du Centre d'Enseignement Intensif des Langues « CEIL ».

Au sein de l'université Ammar Telidji Laghouat

Q₁ : Depuis quand le centre est ouvert ?.....

Q₂ : Quelles sont les différentes catégories des gens qui participe à la formation ?

.....
.....
.....
.....

Q₃ : Quelles sont les langues enseignées au centre ?

.....
.....
.....
.....
.....

Q₄ : Quelles sont les langues les plus demandées ?

.....
.....
.....

Q₅ : Est-ce que le nombre des participants à la formation augmente ?

Oui ☐

Non ☐

Q₆ : Les étudiants qui suivent les cours au centre sont :

- ☐ Universitaire
- ☐ Non Universitaire
- ☐ Les deux

Q₇ : Les enseignants qui enseignent dans le centre sont :

- ☐ Enseignants de l'université
- ☐ Enseignants hors l'université
- ☐ Les deux

Q₈ : La gestion du groupe se fait selon le nombre des :

- ☐ Places
- ☐ Étudiants
- ☐ Salles

Q₉ : Le nombre des salles réservé au centre sont fixée par :

- ☐ Les départements
- ☐ Le centre

Q₁₀ : Est-ce que le nombre des salles est varié ?

- ☐ Oui
- ☐ Non Citer le nombre :

Q₁₁: Est-ce que vous avez un système d'archivage ?

- ☐ Oui
☐ Non

Q₁₂ : Combien de niveaux par langue ?

Français :....
Anglais :....
Espagnol :....
Italien :....

Q₁₃ : Comment vous classez les niveaux ?

.....
.....
.....

Q₁₄ : Quels sont les informations nécessaires pour la première inscription ?

.....
.....
.....
.....

Q₁₅ : Quels sont les informations nécessaires pour la deuxième inscription ?

.....
.....
.....
.....

Q₁₆ : Quels sont les informations nécessaires pour les enseignants ?

.....
.....
.....
.....

Q₁₇ : Combien est le nombre d'enseignants pour chaque groupe ?

Q₁₈ : Est-ce que le temps de l'inscription est fixé ?

- ☐ Oui
☐ Non

Q₁₉ : Combien de session vous avez dans l'année ?

Q₂₀ : Quelle est la période de chaque session ?

1^{ier} session : entre :.....et.....
2^{ier} session : entre :.....et.....
3^{ier} session : entre :.....et.....

Q₂₁ : Quelle est la durée de formation par niveau ?

.....

.....

.....

.....

.....

Q₂₂ : L'emploi du temps se fait par :

☐ L'administrateur

☐ L'enseignant

☐ L'enseignant et l'étudiant

☐ Autre (commenter).....

Q₂₃ : Comment vous gérez les absences ?

.....

.....

Q₂₄ : Quels sont les intervenants dans le centre ?

.....

.....

MERCI