

République Algérienne Démocratique et Populaire
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique
UNIVERSITE MOULOUD MAMMERRI DE TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT INFORMATIQUE

Mémoire de fin d'études
De **MASTER PROFESSIONNEL ET**
ACADEMIQUE

Spécialité : Ingénierie des Systèmes d'Information.
Système d'Information.

Thème :
Analyse, Conception et Réalisation d'une
Application Java EE.
Cas : E-Learning.

Réalisé par :

Mr BOUGHIAS Ahcene.

Melle MENGUELTI Amel.

Encadré par :

Mr: KERBICHE M'hand.

Promotion : 2019 – 2020.

Remerciement

En premier lieu, nous remercions Dieu, le Tout-Puissant pour ses faveurs et ses grâces, de nous avoir donné le courage et la patience de mener ce travail durant toute cette année.

*Nous tenons à remercier D'abord nos enseignants qui nous ont formés, vivement notre promoteur Mr **KERBICHE M'hand** qui nous a formé, orienté et encadré pour la réalisation de ce projet.*

Ensuite, nous remercions chaleureusement les membres des jurys qui ont accepté de juger notre mémoire de fin d'études.

Nous remercions tous ceux, qui de près ou de loin, ont contribué à la réalisation de ce modeste travail.

Dédicaces

Je dédie ce modeste travail à :

*Mes parents, mes frères et sœurs et à leurs enfants garçons et
filles qui m'ont toujours soutenu et encouragé.*

A mes chers camarades et amis (e)

. Fouzia, Assia, Razika, Karima, Assia, Mouloud.

Une spéciale dédicace à mes collègues et amis Ferhat

Tidjedam et Djamel pour leurs encouragements.

A mon binôme Ahcene Boughias

*Enfin à tous ceux qui me sont chers et qui m'ont aidé de près
ou de loin.*

Amel.

Dédicaces

Je dédie ce modeste travail à :

Mes parents, Ma femme et ma fille Anais.

mes frères et sœurs et à leurs enfants garçons et filles.

A mes chers camarades et amies . Rachid, Si moh, Nacer,

somio, hamida, karima .

Une spéciale dédicace à mes collègues et amis pour leurs

encouragements.

A ma binôme Amel Menguelti

Enfin a tous ceux qui me sont chers et qui m'ont aidé de près

ou de loin.

Ahcene

Table des matières

Liste des figures.....	9
Liste des tableaux.....	11
Introduction générale.....	13

Chapitre I

Introduction au E-Learning

Introduction:	16
1. E_LEARNING	16
1.1 Histoire et évolution :	16
1.2 Définition.....	17
1.2.1 Définition 1 : [1].....	17
1.2.2 Définition 2 : [2].....	17
1.3 Type de formations :	18
1.3.1 La formation à distance (FAD):.....	18
1.3.2 La formation ouverte et à distance (FOAD):.....	18
1.3.3 MOOC (Massive Open Online Course) : « cours en ligne ouvert et massif ». [6].....	19
1.4 Principe du E_Learning.....	20
1.5 Techniques de communication dans E-Learning [7] :	21
1.5.1 Sychrone :	21
1.5.2 Asynchrone :	21
1.5.3 Mixte.....	22
1.6 Acteurs du E-Learning : [8].....	22
1.6.1 L'apprenant (l'étudiant):	22
1.6.2 L'enseignant:	22
1.6.2.1 Auteur (concepteur) de cours :	23
1.6.2.2 Tuteur:	23
1.6.2.3 Évaluateur :	23
1.6.3 L'administrateur:	23
1.7 Les systèmes de gestion de cours :	23
1.7.1 LMS : Learning Management System [9]	23
1.7.2 LCMS: Learning Content Management System [9] :	24
1.8 Quelques plate formes E-Learning:	24

1.8.1 Moodle: [10].....	24
1.8.2 Ganesha: [11].....	24
1.8.3 Claroline: [12]	24
1.9 Evolution des pratiques du e-learning: Social Learning, M-Learning	25
1.9.1 Social Learning [13]:	25
1.9.2 M-Learning [14]:	25
1.10 E-Learning en Algérie	25
1.11 Les avantages et les inconvénients du e-Learning :.....	29
1.11.1 Les avantages du e-Learning:	29
1.11.2 Les inconvénients du e-Learning:.....	29
Conclusion :	30

Chapitre II

Analyse et conception

Introduction.....	32
1.Objectif :	32
2. Analyse des besoins :	32
2.1 Acteurs et cas d'utilisation :	33
2.1.1 Identification des acteurs :.....	33
2.1.2 Cas d'utilisations :	33
2.2 Spécifications des besoins : [1].	35
2.2.1 Les besoins fonctionnels :	35
2.2.2. Les besoins non fonctionnels :	36
2.3 Diagramme de contexte :.....	36
3. Conception de l'application :.....	37
3.1 Diagramme de cas d'utilisation :	37
3.2 Diagramme de Séquences :	38
3.2.1 Authentification :.....	38
3.2.2 Inscription utilisateur :.....	39
3.2.3 Ajouter Contenu (Cours/TD/ TP) :.....	40
3.2.4 Préparation examen (Création QCM) :.....	41
3.2.5 Passer examen :	42
3.2.6 Envoie Message :.....	43
3.3 Diagramme d'activité :	44

3.3.1 Diagramme d'activité Gestion utilisateur :	44
3.3.2 Diagramme d'activité Gestion des cours :	45
3.3.3 Diagramme d'activité Consultation messagerie :	46
3.4 Diagramme de classes :	47
3.4.1 Description des classes :	48

Chapitre III

Réalisation

Introduction.....	52
1. Outils de développement et Langages de programmation :	52
1.1. Outils de développement :	52
1.1.1. Caractéristiques de l'ordinateur :	52
1.1.2. Le serveur :	53
1.1.2.1. Apache Tomcat :	53
1.1.2.2. WampServer :	55
1.1.3. PhpMyAdmin :	56
1.1.4. Eclipse IDE (IntegratedDevelopment Environnement) :	57
1.2. Langages de programmation :	58
1.2.1. JAVA :	58
1.2.2. HTML 5 :	58
1.2.3. CSS3 et CSS4 :	58
1.2.4. JavaScript :	59
2. Modèle Relationnel :	60
2.1. Les règles pour le système relationnel de base des données :	60
2.2. Le modèle relationnel du projet :	62
2.3. Le modèle physique de données :	63
2.3.1. Table «Utilisateur»:	63
2.3.2. Table «Authentification»:	64
2.3.3. Table «Etudiant»:	64
2.3.4. Table «Enseignant»:	64
2.3.5. Table «Administrateur»:	65
2.3.6. Table «Livre» :	65
2.3.7. Table «Message» :	65
2.3.8. Table «Formation» :	66

2.3.9. Table «Module» :	66
2.3.10. Table «Examen» :	66
2.3.11. Table «Contenu» :	67
2.3.12. Table «Question» :	67
2.3.13. Table «Choix» :	67
2.3.14. Table «Réponse» :	68
3. Présentation de quelques interfaces :	69
3.1. Page d'authentification:	69
3.2. Page AjouterContenu:	69
3.3. Page Compte Enseignant:	70
3.4. Page liste des Enseignants:	70
conclusion.....	71

LISTE DES FIGURES

Chapitre I :

Figure I.1 : <i>Principe du E-Learning</i>	20
Figure I.2 : <i>Technique de communication E-Learning</i>	22
Figure I.3: <i>Accueil de plate-forme AVUnet</i>	26
Figure I.4 : <i>Accueil de plate-forme imadrassa</i>	27
Figure I.5: <i>Accueil de plate-forme likoul</i>	27
Figure I.6: <i>Accueil de plate-forme ONEFD</i>	28
Figure I.7: <i>Accueil de plate-forme elearning-mesrs.cerist</i>	28

Chapitre II :

Figure II.1: <i>Diagramme de contexte</i>	36
Figure II.2 : <i>Diagramme de cas d'utilisation général</i>	37
Figure II.3 : <i>Diagramme de séquence «Authentification»</i>	38
Figure II.4 : <i>Diagramme de séquence «Inscription utilisateur»</i>	39
Figure II.5 : <i>Diagramme de séquence «Ajouter Contenu»</i>	40
Figure II.6. <i>Diagramme de séquence «préparation examen (création QCM)»</i>	41
Figure II.7 : <i>Diagramme de séquence de «Passer examen»</i>	42
Figure II.8 : <i>Diagramme de séquence «Envoi Message»</i>	43
Figure II.9 : <i>Diagramme d'activité «Gestion utilisateur»</i>	44

Figure II.10 : <i>Diagramme d'activité «Gestion des cours»</i>	45
Figure II.11 : <i>Diagramme d'activité «Consultation messagerie»</i>	46
Figure II.12 : <i>Diagramme de classes général</i>	47

Chapitre III :

Figure III.1 : <i>Caractéristiques de l'ordinateur</i>	52
Figure III.2 : <i>Page d'accueil serveur «Apache Tomcat 8.0»</i>	54
Figure III.3 : <i>Page d'accueil «WampServer»</i>	55
Figure III.4 : <i>Page d'accueil de «PhpMyAdmin»</i>	56
Figure III.5 : <i>page d'accueil «Eclipse IDE»</i>	57
Figure III.6 : <i>Page «Authentification utilisateur»</i>	69
Figure III.7 : <i>Page «Ajouter Contenu»</i>	69
Figure III.8 : <i>Page «Compte Enseignant»</i>	70
Figure III.9 : <i>Page «Liste des Enseignants»</i>	70

LISTE DES TABLEAUX

Chapitre II :

Tableau II.1 : *Cas et sous cas d'utilisation des acteurs*34

Tableau II.2 : *Les besoins fonctionnels*.....35

Chapitre III :

Tableau III.1 : *Table «Utilisateur»*.....63

Tableau III.2 : *Table «Authentification»*.64

Tableau III.3 : *Table «Etudiant»*.....64

Tableau III.4 : *Table «Enseignant»*.....64

Tableau III.5 : *Table «Administrateur»*.....65

Tableau III.6: *Table «Livre»*.65

Tableau III.7 : *Table «Message»*.....65

Tableau III.8 : *Table «Formation»*.....66

Tableau III.9 : *Table «Module»*.....66

Tableau III.10 : *Table «Examen»*.66

Tableau III.11 : *Table «Contenu»*.67

Tableau III.12 : *Table «Question»*.67

Tableau III.13 : *Table «Choix»*.67

Tableau III.14 : *Table «Réponse»*.68

Introduction Générale

INTRODUCTION GENERALE

Avec l'essor de l'Internet, le télé-enseignement s'est radicalement transformé. Autrefois, il relevait essentiellement d'une expérience solitaire: l'étudiant recevait des cours magistraux ainsi que des travaux dirigés et pratiques; les moyens mis en œuvres se résumaient au tableau noir pour écrire et du papier comme supports. Par ailleurs, ce système qui est plus ou moins traditionnel imposait aux étudiants une unité de temps, de lieu, d'action, d'énergie et de rythme ce qui impliquait forcément une difficulté pour les étudiants d'assister.

Par comparaison, l'Internet accueille des salles de classe virtuelles dans lesquelles l'interactivité et le partage des ressources et des informations sont absolument essentiels. Cela ne signifie pas qu'il n'existait pas de classes virtuelles avant l'essor de l'Internet. Pendant des années, plusieurs établissements d'enseignement se sont efforcés de mettre en œuvre des programmes d'enseignement à distance conçus pour des systèmes de téléconférence, mais le coût extrêmement élevé de ce service en a empêché la généralisation. Cette technologie était hors de portée de la plupart des pays en développement, dont seuls quelques-uns ont pu la mettre en œuvre de façon limitée à l'intention de quelques privilégiés. En outre, la nécessité d'assurer une présence en temps réel rendait le système très rigide et mal adapté aux exigences d'un horaire d'enseignement souple.

En effet, il existe une solution de rechange à l'enseignement traditionnel cité ci-dessus qui est représentée par la formation à distance également appelé l'E-Learning.

Dans tous les cas, la méthode de formation/d'éducation permet théoriquement de s'affranchir de la présence physique d'un enseignant à proximité mais aussi bien d'autres avantages comme l'accès à un large panel d'approche formative et de stratégies d'apprentissage adaptées à toute sorte d'utilisateur mais aussi l'économie du temps .

L'objectif de notre projet est la réalisation d'une application E-Learning au profit des étudiants et enseignants tout en mettant à leur disposition des outils qui leur permettent d'assurer les différents rôles pédagogiques.

Globalement, notre mémoire se compose de trois chapitres :

Chapitre 1 : E-Learning.

Chapitre 2 : Analyse et conception.

Chapitre 3 : Réalisation de l'application.

Enfin, nous clôturons ce mémoire par une conclusion générale

Introduction au E-Learning

Introduction:

Les systèmes d'enseignement traditionnels imposent à tous les apprenants un lieu, du temps, des actions ainsi qu'un rythme à suivre. Les auditeurs sont passifs, souvent intellectuellement absents du fait des conditions difficiles imposées par la surcharge horaire, ce qui implique une rigidité des mécanismes et une difficulté d'adéquation avec la réalité quotidienne.

Le besoin sans cesse croissant de la formation dans les sociétés modernes d'une part et l'évolution de la technologie de l'information et de la communication d'autre part ont vu la naissance du E-Learning comme une nouvelle forme d'enseignement.

Dans ce chapitre, nous allons présenter le E-Learning et ses notions.

1. E_LEARNING

1.1 Histoire et évolution :

Une longue histoire a précédé ce qu'on regroupe aujourd'hui sous la notion du « E-Learning ». Il s'agit de la dernière forme de l'Enseignement à Distance (EAD). L'enseignement à distance ou l'apprentissage à distance se compose de techniques et méthodes permettant l'accès aux programmes éducatifs pour les étudiants qui sont séparés par le temps et l'espace. Les systèmes de e-Learning souffrent du manque de la relation élève-enseignant (une à une).

Il existe plusieurs moyens pour assurer l'enseignement à distance: la correspondance sur papier, des cassettes vidéo éducatives, éducation par ordinateur (enseignement multimédia, utilisation d'Internet pour l'éducation sur le Web), etc. Où trois phases de développement sont distinguées (qui sont tous des formes de développement de l'enseignement traditionnel):

1) Première phase : Connue sous le nom d'enseignement par correspondance; l'enseignement dans cette phase est basé sur la diffusion du contenu de cours élaboré par l'enseignant suivant sa propre logique qui s'impose à tous les apprenants. Le rythme de l'enseignement et lui aussi imposé par l'organisation de la formation.

2) Seconde phase : Caractérisé par le développement de l'enseignement assisté par ordinateur (EAO), basée sur l'approche « behavioriste » ; qui a cherché à se dégager de cette programmation linéaire et uniforme en proposant des parcours différents pour les élèves en fonction des résultats de tests.

3) Troisième phase : Fondé sur l'approche constructiviste (le savoir ou la connaissance est construite par l'apprenant). Cette phase se caractérise par l'introduction d'une plus grande variété de technologies dont notamment le développement d'Internet. C'est une mixture entre l'enseignement à distance et l'enseignement présentiel (traditionnelle) connu sous le nom « blended learning ». Il essaye de cumuler les avantages des deux formules. Avec le e-Learning c'est différent, il y'a une personnalisation des parcours pédagogiques en fonction des résultats de tests et des conseils du tuteur.

1.2 Définition

Le e_learning est un acronyme anglais, se compose du Préfixe « e » qui se prononce "i". « e » Abréviation de « électronique » et actuellement de « en ligne » Préfixe indiquant qu'un dispositif électronique est utilisé avec le terme qui suit. Avec le développement de l'Internet et des réseaux, l'aspect électronique correspond surtout à celui de «en ligne». Tel que le : e- gouvernement, e-commerce ou e-administration.

1.2.1 Définition 1 : [1]

Le e_Learning formation en ligne ou apprentissage en ligne est définit par **le conseil de l'union européenne** comme étant : «l'utilisation des nouvelles technologies multimédias de l'Internet pour améliorer la qualité de l'apprentissage en facilitant d'une part l'accès à des ressources et à des services, d'autre part les échanges et la collaboration à distance ».

1.2.2 Définition 2 : [2]

Le e_Learning ou l'apprentissage virtuel se définit comme « l'acquisition de connaissances et de compétences au moyen des technologies de l'information et des communications (TIC) dans le but de favoriser les interactions en matière d'apprentissage, tant sur le plan du contenu, des activités et des outils d'apprentissage, qu'avec d'autres utilisateurs ».

L'apprentissage virtuel ne recourt pas nécessairement à l'Internet. Il peut également s'acquérir au moyen de diverses applications indépendantes et de réseaux locaux, en tout temps et en tous lieux, tant dans un contexte éducatif, qu'en milieu de travail ou à la maison.

1.3 Type de formations :

1.3.1 La formation à distance (FAD):

La formation à distance consiste principalement en « la séparation de l'étudiant et du Professeur dans l'espace et dans le temps ». Dans sa forme la plus simple, la formation à distance vise principalement à fournir une formation pour les personnes isolées géographiquement. La communication entre les étudiants et les instructeurs se fait de deux façons, soit synchrone (en temps réel), soit asynchrone (différée dans le temps), notamment à l'aide des Techniques d'Information et de Communication (TIC). [3].

La FAD est incluse dans le concept plus général de la FOAD «Formation Ouverte et à Distance ». Souvent confondues, la FOAD n'est pas la FAD. Les caractéristiques essentielles de la FAD Résident dans le fait que les apprenants sont physiquement éloignés des enseignants; la FAD recouvre à la fois l'enseignement à distance et l'apprentissage à distance au cours duquel l'apprenant agit sans professeur en dehors du centre d'enseignement.

Les établissements sont divisés en deux types, soit unimodal et bimodal.

Unimodal: l'établissement offre uniquement de la formation à distance.

Bimodal: l'établissement offre à la fois de la formation à distance et de l'enseignement en mode présentiel.

1.3.2 La formation ouverte et à distance (FOAD):

La FOAD, comme l'indique son acronyme « Formation Ouverte et A Distance », possède deux caractéristiques [4] :

- **Ouverte** : grande souplesse et beaucoup de degrés de liberté dans l'accès à la formation. Aucune inscription préalable n'est par exemple nécessaire pour accéder aux contenus pédagogiques.
- **À distance** : la distance impliquant la dispersion géographique et/ou temporelle des acteurs impliqués dans le processus de formation.

La **Commission européenne** [5] définit ainsi la formation ouverte et à distance [Anglais: ODL (Open and distance learning)] : "La FOAD un enseignement à distance qui permet à chacun de travailler de façon autonome, à son propre rythme, quel que soit le lieu où il se trouve, notamment grâce au E-Learning."

L'UNESCO[5] quant à elle caractérise cette dernière par : « une liberté d'accès aux ressources pédagogiques mises à disposition de l'apprenant, sans aucune restriction, à savoir : absence de condition d'admission, itinéraire et rythme de formation choisis par l'apprenant selon sa disponibilité et conclusion d'un contrat entre l'apprenant et l'institution ».

1.3.3 MOOC (Massive Open Online Course) : Elle est traduite par « cours en ligne ouvert et massif ». [6]

Le MOOC est une formation interactive en ligne, ouverte à tous. C'est une nouvelle expérience d'apprentissage qui séduit chaque jour plus d'étudiants, de salariés et de responsables de formation.

Le MOOC est cadencé dans le temps et se déroule sur quelques semaines, permettant ainsi de créer des promotions d'apprenants, comme dans une formation en présentiel.

Il est composé d'activités pédagogiques (**Course**) variées :

- Visionnage de courtes vidéos animées de 5 à 10 minutes.
- Réalisation de quiz afin de vérifier l'acquisition de connaissances.
- Exercices pratiques et travaux collaboratifs pour développer son réseau et ses compétences.

Ils sont ouverts à tous et l'inscription est gratuite (**Open**). Ils sont destinés aux étudiants. Ils offrent des certificats de réussite après la réussite des épreuves ; ce qui permet au salarié de développer ses compétences. Donc ils sont utilisés dans les entreprises pour former les salariés.

Le MOOC est caractérisé par sa dimension massive (**Massif**). Ils peuvent être suivis par un nombre potentiellement illimité de participants. Cette dimension massive porte ainsi la promesse d'une formation initiale et continue plus ouverte et évolutive.

L'Inscription à un MOOC se fait directement en ligne, car l'intégralité du MOOC est accessible sur internet (**Online**). Les participants se connectent à une plateforme diffusant les activités et ressources pédagogiques.

1.4 Principe du E_Learning [7]

Le principe de E-Learning étant de pouvoir accéder à des cours depuis un poste distant (chez soi, depuis son entreprise), les lieux nécessaires au suivi d'une formation (établissements, classes..) n'existent plus physiquement, ils sont remplacés par le Système de Gestion des Cours ou S.G.C (plate-forme). Le SGC est le coeur du système de formation à distance, il se présente comme un lien entre les apprenants, les cursus, les tuteurs, les ressources et les contenus présents dans le système.

L'apprenant, via une plate-forme, se verra attribuer un certain nombre de modules de cours, d'exercices, d'évaluations qu'il devra effectuer selon un programme établi. Le rôle du tuteur (ou formateur) est de gérer les apprenants qu'il doit suivre, ainsi il pourra leur affecter des ressources à consulter, des cours à étudier, des évaluations afin de se rendre compte de la bonne assimilation des contenus proposés.

La communication entre tous ces acteurs se fait via internet. Dans le passé, les contraintes liées au bas débit rendraient les pages au format HTML et les documents texte les seuls contenus réellement exploitables. Aujourd'hui le haut débit permet l'accès à plusieurs formats de contenu, même les plus lourds tel que l'audio et la vidéo.

La figure suivante représente le principe de E-Learning :

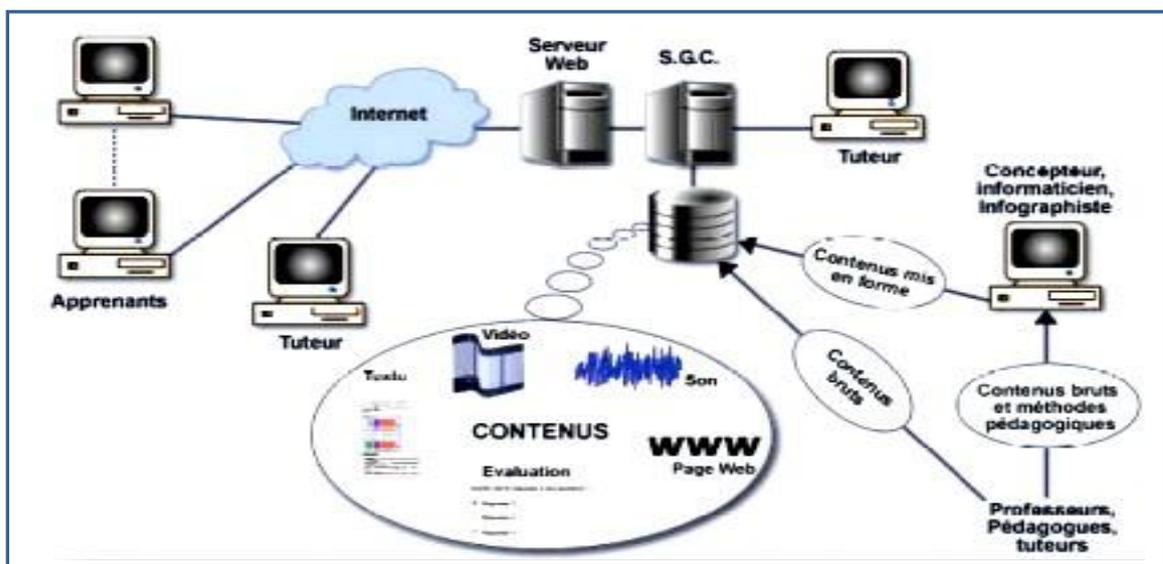


Figure I.1 : Principe du E-Learning.

1.5 Techniques de communication dans E-Learning [8] :

Il existe trois techniques de communication ou de diffusion en E-Learning.

1.5.1 Synchrones :

Méthode de communication en temps réel basé sur la parole et l'écoute l'apprenant est en contact avec son formateur ou avec les autres membres de classe virtuelle par partage d'application ou visioconférence. Les apprenants peuvent poser des questions et recevoir des réponses en temps réel quel que soit la distance à condition que la bande passante soit élevée.

Le dialogue entre les formateurs et les apprenants peut se faire par : Chat, Webcam, micro, partage d'application, fonctionnalités de prise en main à distance, tableau blanc et téléphone. Parmi les avantages de diffusion synchrone :

- Les apprenants interagissent intensivement à l'écran avec les formateurs, le langage formel et visuel est utilisé.
- Création des contenus à diffuser rapidement.

1.5.2 Asynchrone :

C'est une façon de formation qui se passe en temps différé et qui permet à l'apprenant d'accéder à la formation à sa guise et autant de fois qu'il le désire (textes, animations et multimédia...). La communication entre les apprenants et les formateurs peut se faire via e-mail, messagerie instantanée, forum de discussion, sms et partage de documents.

Les principaux avantages de formation asynchrone sont :

- Les apprenants évoluent à leurs propres rythmes.
- Les apprenants peuvent adapter l'ordre dans lequel ils appréhendent les éléments du cours.
- Ils peuvent revoir et approfondir certains aspects du cours à leur façon.

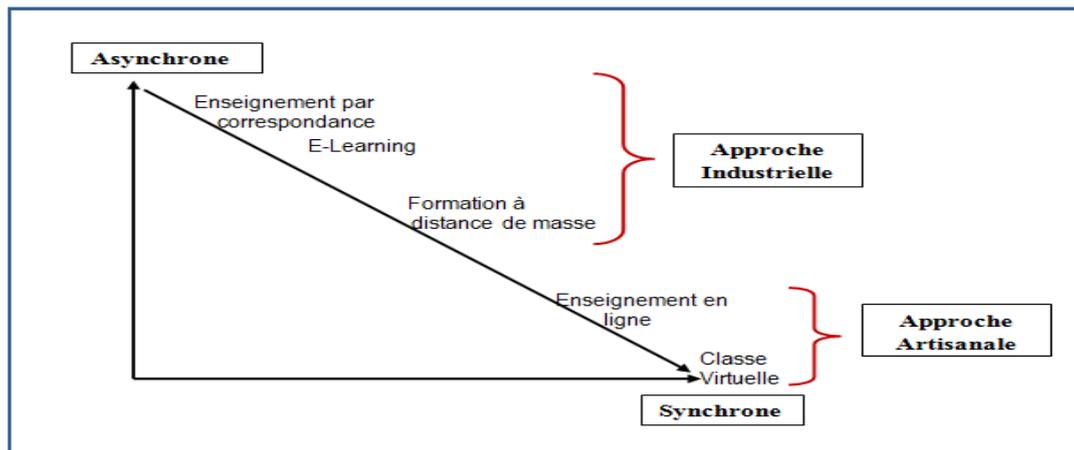


Figure I.2 : *Technique de communication E-Learning.*

1.5.3 Mixte

C'est une méthode qui combine les deux méthodes précédentes : synchrone et asynchrone, il peut utiliser pour homogénéiser les savoir avant une présentation, une conférence, une intervention ou une formation, ses principaux avantages sont :

- En milieu universitaire, des études ont démontré que l'acquisition des connaissances est meilleure avec la formation mixte.
- Lorsque on peut enregistrer le déroulement et sauvegarder les éléments échangés pendant le cours, les apprenants peuvent revoir à leur guise ce que ils souhaitent réviser.

1.6 Acteurs du E-Learning : [9]

Les différents acteurs d'une formation en ligne peuvent être classés par catégories suivant les rôles qu'ils seront amenés à jouer. Toutefois, trois catégories principales sont recensées: les apprenants, les enseignants (ou tuteur) et les administrateurs.

1.6.1 L'apprenant (l'étudiant):

L'apprenant est un individu qui s'engage à suivre les activités d'une e_formation afin d'acquérir des connaissances. Il est l'acteur central pour lequel la formation est conçue.

1.6.2 L'enseignant:

Avant le passage à la méthode d'enseignement centrée sur l'apprenant, le professeur était la pièce maîtresse dans le processus d'apprentissage, c'était lui qui détenait le savoir, mais avec la nouvelle méthode, l'accent est mis sur l'apprenant afin de le rendre plus actif. Le rôle de l'enseignant est beaucoup plus de guider et d'orienter les apprenants.

Actuellement, l'enseignant peut prendre le rôle d'un :

1.6.2.1 Auteur (concepteur) de cours :

Son rôle est la création du contenu.

1.6.2.2 Tuteur:

Il a un rôle d'accompagnement des groupes d'apprenants, surtout au cours des phases d'action à distance (suivi et motivation...) en tant qu'animateur de la formation.

1.6.2.3 Évaluateur :

Son rôle essentiel est la création des activités de validation de connaissances par la création des tests, le suivi et l'évaluation de l'apprenant.

1.6.3 L'administrateur:

L'administrateur d'un système e-Learning s'occupe de l'installation et la maintenance de la plate-forme, la gestion des droits d'accès et la suppression des acteurs et des contenus en cas d'abus et la création de liens avec les systèmes.

1.7 Les systèmes de gestion de cours :**1.7.1 LMS : Learning Management System [10]**

En français « plate-forme de gestion de la formation », désigne une famille de système axée sur l'intégration, la diffusion et la gestion de contenu pour la formation à distance.

Un LMS peut également être utilisé pour gérer des formations dites « présentielles », c'est-à-dire en présence de formateurs.

les acteurs impliqués dans les formations utilisent un LMS comme suit:

- ✓ les responsables de la formation (auteurs et formateurs) :
 - créent et mettent en ligne les formations,
 - définissent les profils d'accès aux formations,
 - gèrent et animent les formations (que ce soit en synchrone ou en asynchrone).
 - suivent les retours de Formation.
- ✓ l'administrateur du système configure le LMS et gère les statistiques sur les retours de formation.

- ✓ les apprenants suivent la formation via le LMS (s'il s'agit de e-learning) et émettent leurs retours, de façon active ou passive, via le système.

1.7.2 LCMS: Learning Content Management System [10] :

Littéralement « plate-forme de gestion de contenu de formation », Comme vu précédemment, les LMS fournissent des fonctionnalités restreintes en terme de conception de cours. C'est pourquoi les LCMS ont récemment émergé pour aider les entreprises à créer, stocker et diffuser des contenus d'apprentissage à partir d'une base de données centrale.

L'objectif est de permettre la délivrance de contenus propriétaires personnalisés, tout en réduisant les coûts de création de contenu.

1.8 Quelques plate formes E-Learning:

1.8.1 Moodle: [11]

Moodle est une plateforme d'apprentissage en ligne open source écrite en PHP. Développée à partir de principes pédagogiques, elle permet de créer des communautés s'instruisant autour de contenus et d'activités.

1.8.2 Ganesha: [12]

Ganesha est une plate-forme d'apprentissage en ligne (LMS : Learning Management System) qui met l'accent sur des parcours individualisés. Au contraire des plateformes orientées « contenu » (CMS : Content Management System), Ganesha organise les activités et les ressources pédagogiques autour de l'apprenant plutôt qu'autour d'un cours.

1.8.3 Claroline: [13]

C'est une plate-forme d'apprentissage en ligne (ou LMS) et de travail collaboratif open source . Elle permet de créer et d'administrer des formations et des espaces de collaboration en ligne.

1.9 Evolution des pratiques du e-learning: Social Learning, M-Learning

1.9.1 Social Learning [14]:

Le social Learning connu également sous la notion du E-Learning 2.0 est un nouveau phénomène du Web 2.0, c'est un mode d'apprentissage qui permet de partager, de construire, de collaborer avec d'autres à distance via des outils collaboratifs (wiki, chat, forum, blog,...) ou des réseaux sociaux d'entreprises.

1.9.2 M-Learning [15]:

Le M-Learning fait référence à l'utilisation d'appareils mobiles ou sans fil dans le but de pouvoir apprendre en se déplaçant. Comme exemples typiques des dispositifs utilisés pour le M-Learning, il y a les téléphones cellulaires, les Smartphones, les ordinateurs portables, les ordinateurs de poche et les Tablettes.

Les récentes innovations dans les applications de programmes et les logiciels sociaux en utilisant les technologies Web 2.0 (par exemple, les blogs, les wikis, Twitter, YouTube) ou les sites de réseaux sociaux (tels que Facebook et MySpace) ont fait des appareils mobiles plus dynamiques et promettent également un apprentissage plus pédagogique et plus attractif.

1.10 E-Learning en Algérie

De l'avis des spécialistes de la pédagogie, le e-Learning et l'école numérique ont fait et font leur chemin à pas de géant en Algérie. Au grand bonheur des écoliers et enseignants, voir même des administrateurs de la pédagogie, l'usage des TIC à rassembler sur le même réseau les enseignants, les apprenants et leurs parents et ce depuis l'apparition des plateformes e-Learning.

Ce pendant le e-Learning a bien mener son chemin et ce depuis le début de son apparition en Algérie qu'on peut énumérer comme suit :

Au mois de **mars 2006**, **DJAWEB**, une filiale d'Algérie Télécom, a lancé le premier portail de la formation électronique de type e-Learning en Algérie et au Maghreb, fruit d'un partenariat avec les sociétés IMT Training solutions, Human soft learning solution et Thomson Netg.

Durant la même année, a connu le lancement par le Centre National de l'Enseignement Professionnel à Distance (**CNEPD**) avec *Euro Aptitude* une nouvelle offre de formation intitulée «Certification des compétences en informatique» pour la certification

à l'International Computer Driving Licence (ICDL). D'une validité d'un an, cette formation est un standard international de mesure des compétences de base en informatique. [16].

En **Avril 2008**, une autre plateforme appelée *Tarbiatic* vit le jour, cette dernière est spécialisée en cours de soutiens aux élèves des trois paliers de l'éducation nationale. « Le réseau *Tarbiatic*, qui était utilisé par une communauté de 20.000 élèves, a même été classé par l'UNESCO parmi les cinq meilleures plateformes de télé-enseignement dans la région Afrique du Nord/Moyen-Orient » [17].

Certaines universités auront lancé au cours des années suivantes des plateformes pédagogiques (mode en ligne jumelé à l'enseignement présentiel).

De sa part, l'Université de la Formation Continue a lancé depuis **2009** deux expériences en la matière destinées à la formation professionnelle des fonctionnaires de l'État, la plateforme de e-Learning est appelée **SAATAR**. Par la suite, d'autres expériences fructueuses ont suivi.

Aujourd'hui l'e-Learning se renforce de plus en plus en Algérie grâce à plusieurs plateformes qui poussent Citons, [18] par exemple la plate-forme expérimentale appelée **AVUnet** (Université virtuelle algérienne) «<http://www.avunet.ca/>»

avunet Entretien professionnel de votre équipement informatique

info@avunet.ca
514 990-6307

english

ACCUEIL SERVICES CLIENTS CONTACTEZ-NOUS

Travail assuré sur l'équipement informatique et les guichets

DEMANDE D'ÉVALUATION

Nom*

Prénom*

Compagnie*

Téléphone

Courriel*

Commentaires:

ENVOYER >>

* Champs obligatoires

Nous vous contacterons d'ici 24 heures

Depuis plus de 13 ans, Avunet, spécialiste en nettoyage, redonne à tous les équipements leurs éclats de propreté d'origine. Avunet s'efforce d'offrir un service de qualité adapté aux besoins de ses 500 clients et plus dans toutes les régions du Québec, de l'Ontario et du Nouveau-Brunswick. La compétence et le professionnalisme de nos employés garantissent d'ailleurs la satisfaction de nos clients. Nous souhaitons établir avec eux une relation solide qui permettra une collaboration à long terme. Il en résulte une grande tranquillité d'esprit et un personnel des plus reconnaissant, d'un côté comme de l'autre.

SAVIEZ-VOUS QUE ?

- Le magazine de consommation Computing a demandé à un biologiste d'examiner au microscope plus de 30 claviers et 30 guichets automatiques de plusieurs bureaux typiques. Son examen a révélé que certains d'entre eux abritaient des bactéries qui présentaient un risque élevé de maladie pour l'utilisateur.

Figure I.3: Accueil de plate-forme AVUnet.

Et la plate-forme expérimentale appelée 'imadrassa' <https://www.imadrassa.com/> lancé avec la collaboration de l'opérateur de téléphonie mobile Djezzy et de Amana assurance créée en 2015. [19].



Figure I.4 : Accueil de plate-forme imadrassa.

Une plateforme de soutien scolaire, dénommée *Likoul*, est lancée, **mercredi 1 février 2017**, au bénéfice des cycles d'enseignement primaire, moyen et secondaire, y introduisant les plus récentes technologies de la communication. www.likoul.dz. [20]



Figure I.5: Accueil de plate-forme likoul.

La crise que traverse le pays à l'instar du monde entier en raison de la propagation du virus Covid-19 a engendré la nécessité de recourir aux e_Learning et aux TIC à travers tous les moyens offerts afin de veiller au confort du citoyen de manière générale et de permettre aux élèves et aux étudiants, en particulier, de poursuivre leur cursus scolaire.

Deux plateformes numériques à accès libre ont été mises en place par les ministères de l'Education et de l'Enseignement supérieur, à savoir :

<http://soutien-scolaire.onefd.edu.dz> et <http://elearning-mesrs.cerist.dz>

afin de suivre leurs cours de manière équitable et juste où qu'ils se trouvent à travers le territoire national. [21].



Figure I.6: Accueil de plate-forme ONEFD.



Figure I.7: Accueil de plate-forme « e-Learning-mesrs.cerist. »

1.11 Les avantages et les inconvénients du e-Learning :**1.11.1 Les avantages du e-Learning:**

- ✓ Ouvert à tous ceux qui veulent apprendre, un large public. quels qu'en soient leurs âge, leurs niveau d'instruction, leurs catégorie socioprofessionnelle
- ✓ Accès gratuit et facile à de nouvelles informations et compétences, en phase avec les progrès technologiques.
- ✓ Permet à l'apprenant de communiquer avec les autres et de connaître les différentes races et traditions.
- ✓ Aide pour accéder aux informations correctes et au participant et à la consultation pour accéder aux informations.

1.11.2 Les inconvénients du e-Learning:

- ✓ Absence de réseau ou de problèmes matériels (ordinateurs, câble. etc.)
- ✓ L'absence physique de l'enseignant.
- ✓ Nécessite une maîtrise suffisante des outils informatiques et d'Internet.

Conclusion :

Dans ce chapitre, nous avons présenté une vue globale sur le E-Learning, nous avons commencé par définir le concept E-Learning, ensuite, décrire quelques concepts de base. Enfin nous avons énuméré quelques avantages et inconvénients.

Dans le prochain chapitre nous allons entamer l'étape de l'analyse et Conception de notre application E-Learning.

Analyse et conception

Introduction:

L'étape de la conception est l'étape la plus importante pour l'élaboration de notre application, elle permet en effet, de détailler les spécifications des fonctions ainsi que la structure des données.

Au niveau De ce deuxième chapitre nous allons d'abord définir le sujet du projet, ensuite les acteurs avec leurs différentes tâches et leurs besoins, enfin nous allons détailler la conception de notre application.

Dans le but d'avoir une meilleure analyse et de rendre la conception de notre projet plus complète, nous avons adopté le langage UML (Unified Modeling Language) qui permet de bien représenter l'aspect statique et dynamique d'une application par une série de diagrammes qu'il offre.

1.Objectif :

Dans notre projet nous allons réaliser une application E-Learning qui permettra aux étudiants inscrit de suivre leur formation à leur propre rythme, à distance et complètement en ligne.

2. Analyse des besoins :

Notre application permet aux étudiants de suivre leurs formations, auxquelles ils se sont inscrits, à distance totalement en ligne. En effet, cette plateforme permet le partage entre étudiants et enseignants. En ce qui suit, nous présentons le fonctionnement de l'application:

- Une fois l'administrateur inscrit les étudiants et les enseignants, ces derniers peuvent accéder à leurs espaces personnels en ligne.
- La plateforme permet aux enseignants d'insérer les contenus (cours, TD, TP ...), de préparer des examens (sous forme de QCM), enfin remettre les notes aux étudiants et à l'administration.
- Les étudiants pourront consulter les différents contenus envoyés par l'enseignant sur leurs propres espaces, remettre leurs différents devoirs et passer les examens en un temps prédéfini.
-

- De plus, Les différents utilisateurs (administrateur, enseignants, étudiants) de cette application pourront communiquer via un système de messagerie.

2.1 Acteurs et cas d'utilisation :

2.1.1 Identification des acteurs :

Un acteur est une entité extérieure au système modélisé, il représente un rôle que peut jouer l'utilisateur avec le système. Un acteur interagit directement avec celui-ci.

Les principaux acteurs sont les utilisateurs du système, des logiciels déjà disponibles à intégrer dans le projet, ou des systèmes informatiques externes.

Après études du sujet, nous avons identifié trois (03) acteurs principaux qui sont:

- **L'administrateur :**

Comme dans tout site internet, il est indispensable d'avoir un responsable qui gère techniquement la plateforme, et contrôler la gestion de la base de données, ainsi la gestion entière des matières et formation.

- **L'enseignant :**

Représente toute personne qui peut créer des cours et préparer des évaluations (examens, test QCM sur plateforme), il est identifié par un nom d'utilisateur et un mot de passe.

- **L'étudiant :**

Il s'agit du bénéficiaire de la formation, également appelé stagiaire .pouvant suivre ses cours et passer des examens. Il a renseigné ses informations lors de son inscription papier, il est identifié par un nom d'utilisateur et un mot de passe.

2.1.2 Cas d'utilisations :

Un cas d'utilisation est un service rendu à un acteur, c'est une fonctionnalité. Il décrit une fonction qu'un système exécute pour atteindre l'objectif de l'utilisateur.

Un cas d'utilisation doit renvoyer un résultat observable, utile pour l'utilisateur du système.

Le tableau ci-dessous donne les tâches et sous-tâches spécifiques pour chaque acteur :

Acteur	Tâches	Sous-tâches
Administrateur	❖ Se connecter/ Se déconnecter	/
	❖ Gérer compte.	➤ Créer compte. ➤ Modifier compte. ➤ Supprimer compte.
	❖ Messagerie.	➤ Envoyer messages. ➤ Supprimer messages.
	❖ Gérer médiathèque.	/
Enseignant	❖ Se connecter / se déconnecter	
	❖ Gérer contenus.	➤ Ajouter contenu (cours, TD, TP). ➤ Modifier contenu. ➤ Supprimer contenu. ➤ Envoyer contenu.
	❖ Créer examen.	➤ Ajouter question. ➤ Modifier question. ➤ Supprimer question. ➤ Ajouter barème.
	❖ Gérer examen.	➤ Supprimer examen. ➤ Envoyer examen.
	❖ Gérer notes.	➤ Modifier note. ➤ Envoyer note.
	❖ Messagerie.	➤ Envoyer messages. ➤ Supprimer messages.
	❖ Consulter médiathèque.	/
Etudiant	❖ Se connecter / se déconnecter.	/
	❖ Consulter contenus.	➤ Télécharger contenus. ➤ Remettre travail.
	❖ Passer examen.	/
	❖ Messagerie.	➤ Envoyer messages. ➤ Supprimer messages.
	❖ Consulter médiathèque.	

Tableau II.1 : Cas et sous cas d'utilisation des acteurs.

2.2 Spécifications des besoins : [1].

On distingue deux types de besoins : les besoins fonctionnels et les besoins non fonctionnels.

2.2.1 Les besoins fonctionnels :

Il s'agit des fonctionnalités du système. Ce sont les besoins spécifiant un comportement d'entrées / sorties du Système.

De ce point de vue fonctionnel, notre application devra comporter toutes les fonctionnalités nécessaires pour répondre et satisfaire les besoins de l'utilisateur.

Les grandes fonctionnalités de notre système sont données dans le tableau ci-dessous:

Acteurs	Besoins fonctionnels
Administrateur	<ul style="list-style-type: none"> ➤ Création de comptes (enseignants, étudiants). ➤ Calcul des moyennes des étudiants. ➤ Messagerie. ➤ Gestion médiathèque.
Enseignant	<ul style="list-style-type: none"> ➤ Dépôt de cours, TD et TP. ➤ Création des examens. ➤ Remise les notes à l'administration et étudiants. ➤ Messagerie. ➤ Accès à la médiathèque.
Étudiant	<ul style="list-style-type: none"> ➤ Consultation et téléchargement de contenus. ➤ Remise des travaux. ➤ Passation des examens. ➤ Messagerie. ➤ Accès à la médiathèque.

Tableau II.2 : *Les besoins fonctionnels*

2.2.2. Les besoins non fonctionnels :

Il s'agit des besoins qui caractérisent le système. Ce sont des besoins en matière de performance, de type de matériel ou le type de conception. Ces besoins peuvent concerner les contraintes d'implémentation (langage de programmation, type SGBD, de système d'Exploitation...).

De ce point de vue non fonctionnel, l'ergonomie, la sécurité et la performance jouent un rôle très important dans notre application :

➤ **L'ergonomie :**

S'agit de mettre à la disposition du client une application facile à utiliser et qui ne demande pas une grande maîtrise de l'informatique.

➤ **La sécurité :**

L'application doit respecter la confidentialité des données.

➤ **La performance :**

Toute fonctionnalité ne doit pas nécessiter beaucoup de temps pour l'utilisateur, et la mise en page de notre application facilitera les tâches à l'aide d'une représentation claire et intuitive.

2.3 Diagramme de contexte :

Le diagramme de contexte met en évidence les acteurs concernés par l'application. Ci-dessous nous présentons le schéma lié à notre application.

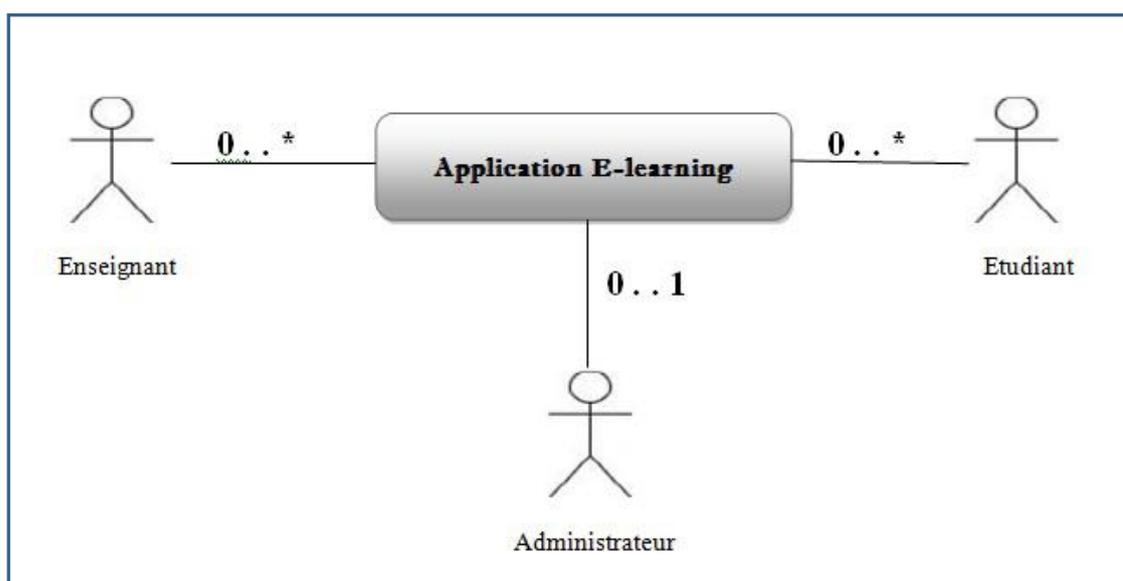


Figure II.1: Diagramme de contexte.

3. Conception de l'application :

3.1 Diagramme de cas d'utilisation :

Le cas d'utilisation est une description des interactions qui vont permettre à l'acteur d'atteindre son objectif en utilisant le système. Les *use case* (cas d'utilisation) sont représentés par une ellipse sous-titrée par le nom du cas d'utilisation. Un acteur et un cas d'utilisation sont mis en relation par une association représentée par une ligne. [2].

Ci-après nous donnons le diagramme des cas d'utilisation par acteur :

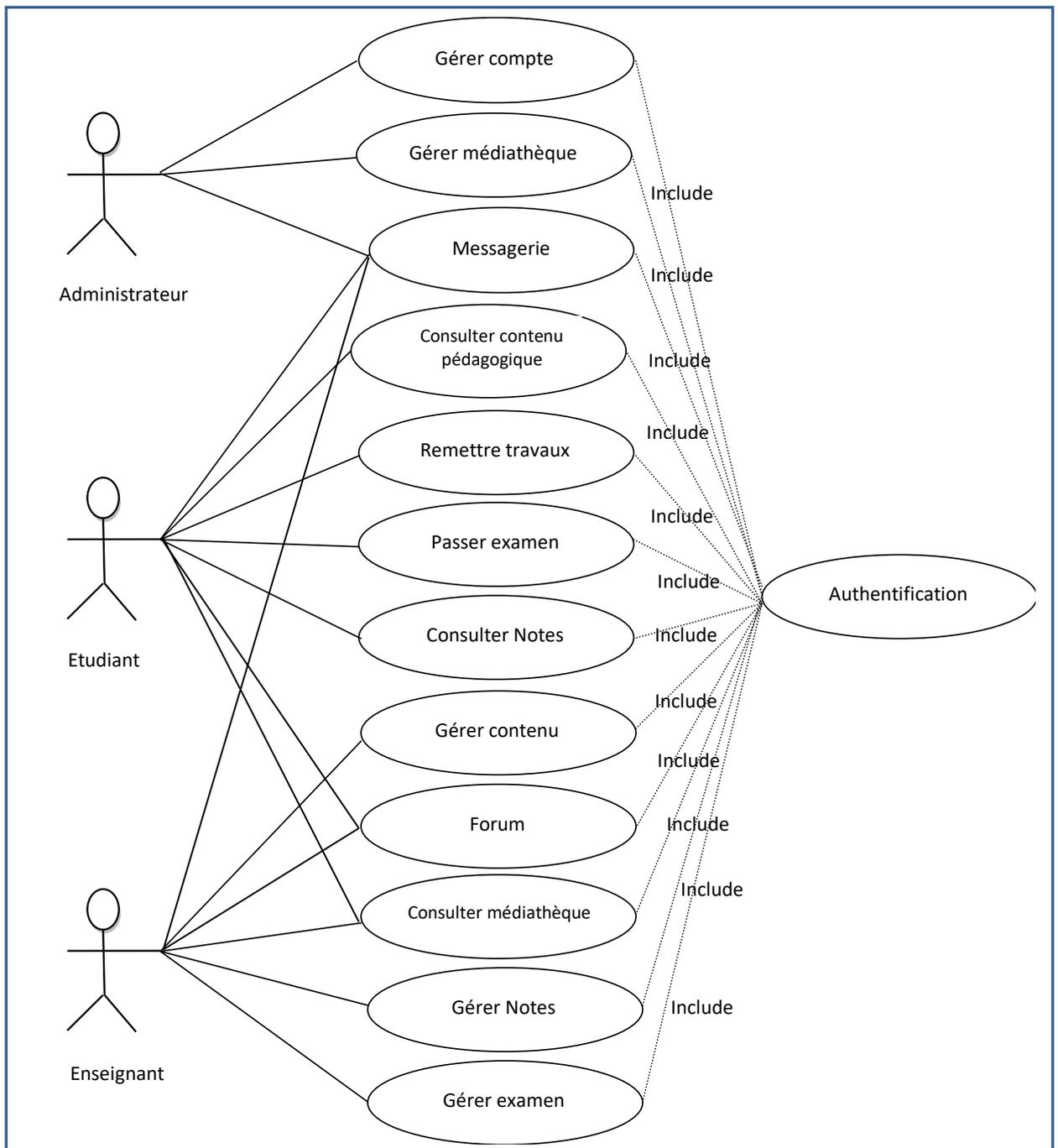


Figure II.2 : Diagramme de cas d'utilisation général.

3.2 Diagramme de Séquences :

Le diagramme de séquence est utilisé pour une représentation claire et précise des fonctionnalités fournies. Ce diagramme offre une représentation dynamique du système.

Il montre pas à pas le déroulement des actions constituant le processus de publication par ordre chronologique.

3.2.1 Authentification :

Quand l'utilisateur souhaite se connecter, il remplit les champs nécessaires. Lors de validation avec la méthode Submit(), les données seront transmises au contrôleur par la méthode Post() qui seront elles-mêmes envoyées par le contrôleur avec la méthode seConnecter() au modèle (classe Personne) qui va vérifier la validité des champs en interrogeant la base de donnée. De cette façon la page associée à l'utilisateur est retournée.

Le diagramme suivant illustre cet enchaînement.

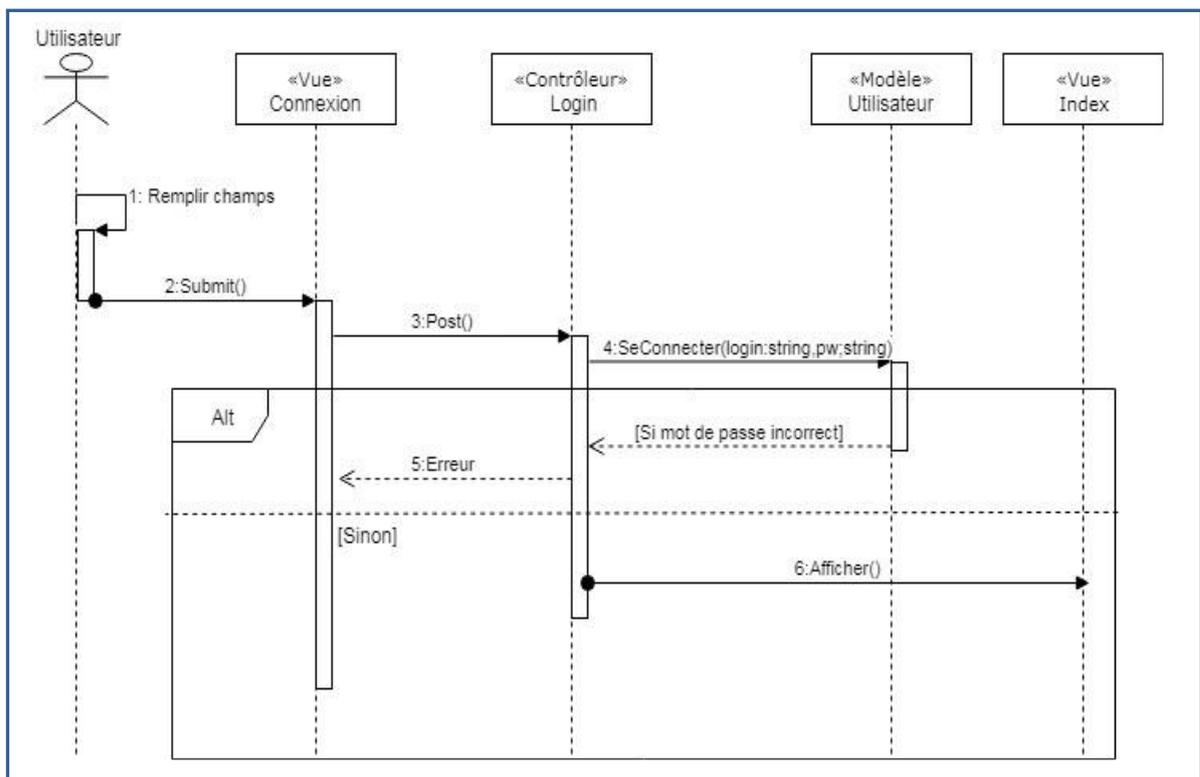


Figure II.3 : Diagramme de séquence «Authentification».

3.2.2 Inscription utilisateur :

Quand l'administrateur procède à l'inscription d'une nouvelle personne que ça soit un enseignant ou un étudiant, il remplit le formulaire lors de la validation, les données seront envoyé au contrôleur par la méthode `Post()`, puis elles seront passé en paramètre au constructeur de la classe `Personne` afin d'insérer la nouvelle personne (enseignant/ étudiant) dans la base de données.

Le diagramme suivant illustre cet enchaînement.

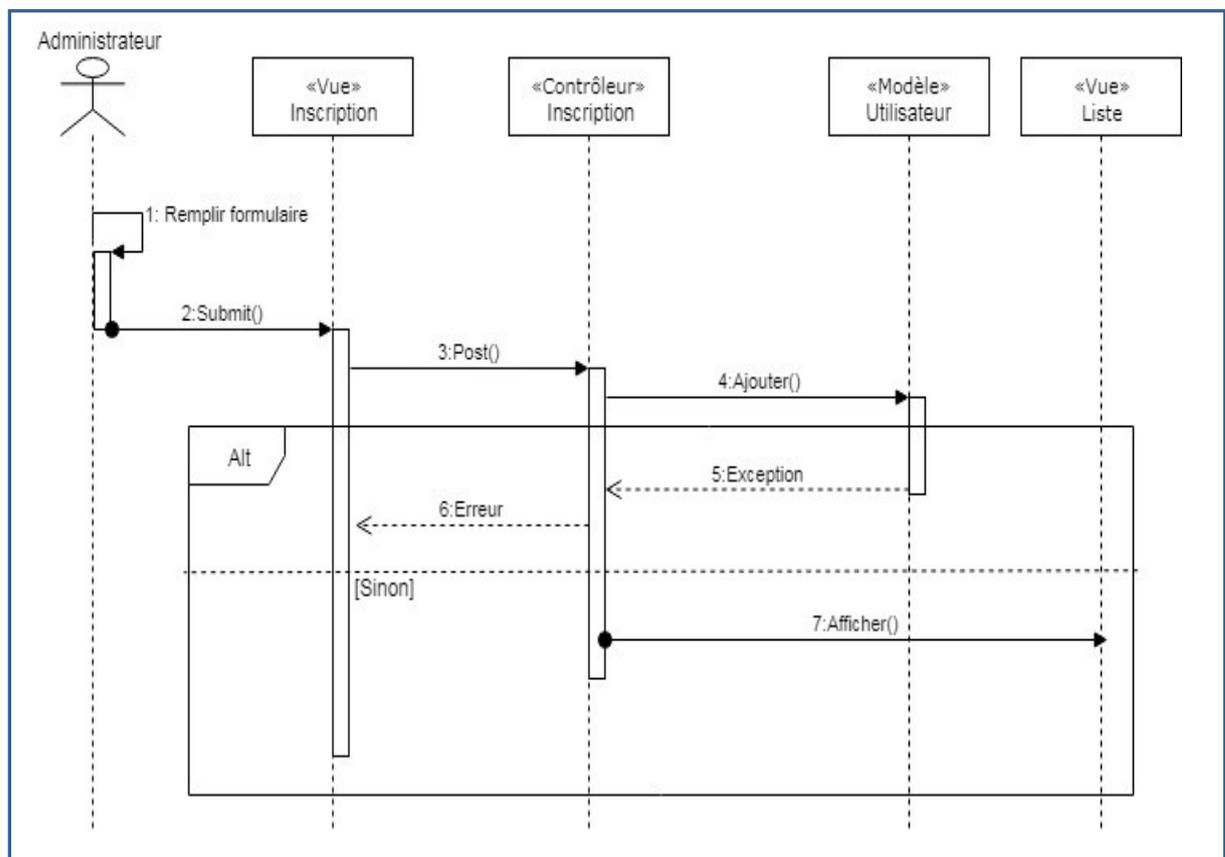


Figure II.4 : Diagramme de séquence «Inscription utilisateur».

3.2.3 Ajouter Contenu (Cours/TD/ TP) :

Quand l'enseignant veut insérer un document (cours, TD, TP ou ressource pédagogique) dans l'espace personnel des étudiants d'un ou plusieurs groupe, il se rend dans son espace, ajoute (upload) le document souhaité et valide l'envoi. Les données seront transmises via la méthode Post () au contrôleur qui va ensuite appeler la méthode fileUpload () qui construira l'objet contenu et mettre à jour la base de données. A ce moment là, les étudiants pourront télécharger le fichier.

Le diagramme suivant illustre cet enchaînement.

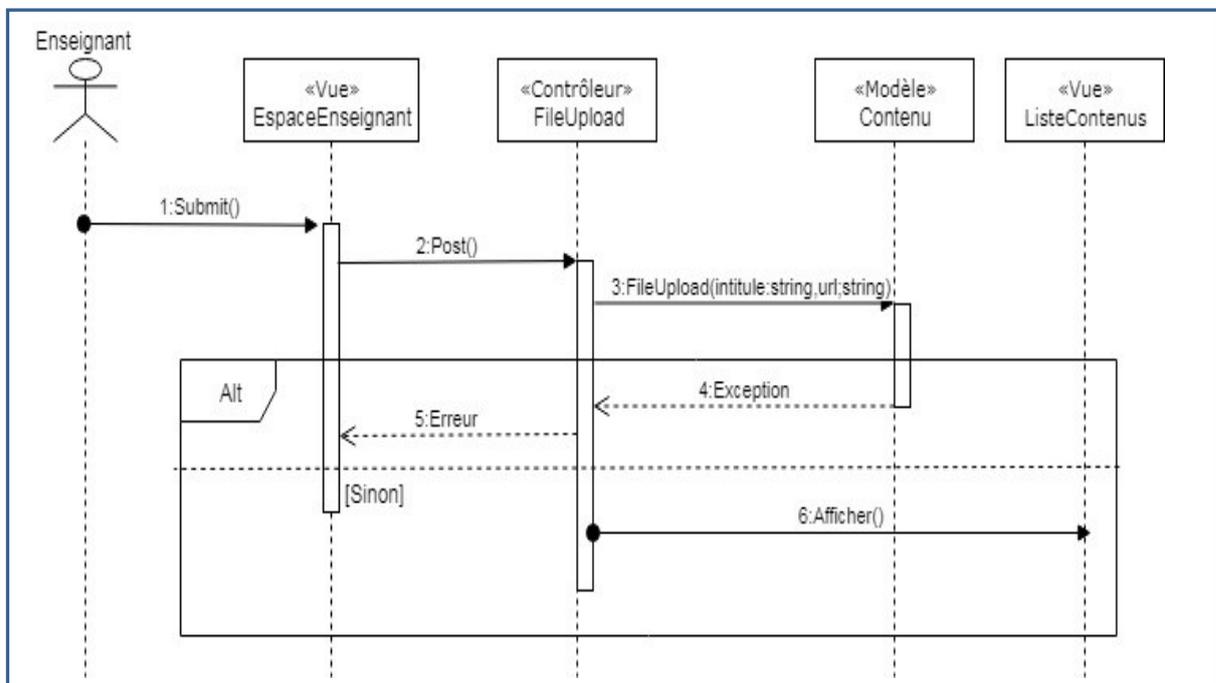


Figure II.5 : Diagramme de séquence «Ajouter Contenu».

3.2.4 Préparation examen (Création QCM) :

Pour la création du QCM l'enseignant ouvre l'interface AjoutQCM et clique sur le bouton de création. Le système crée un espace mémoire pour ce nouvel examen via la méthode Post (), puis il saisit une nouvelle question qui via le contrôleur se charge dans le modèle Question en appelant la méthode AjouterQuestion ().

Ensuite, l'enseignant saisit les propositions possible et choisit la réponse juste parmi celles-ci, ces dernières à leurs tours seront transmises au modèle Proposition.

L'étudiant ne peut voir le contenu de l'examen que lors de son lancement par l'enseignant c'est-à-dire au moment de l'examen.

Le diagramme suivant illustre cet enchaînement.

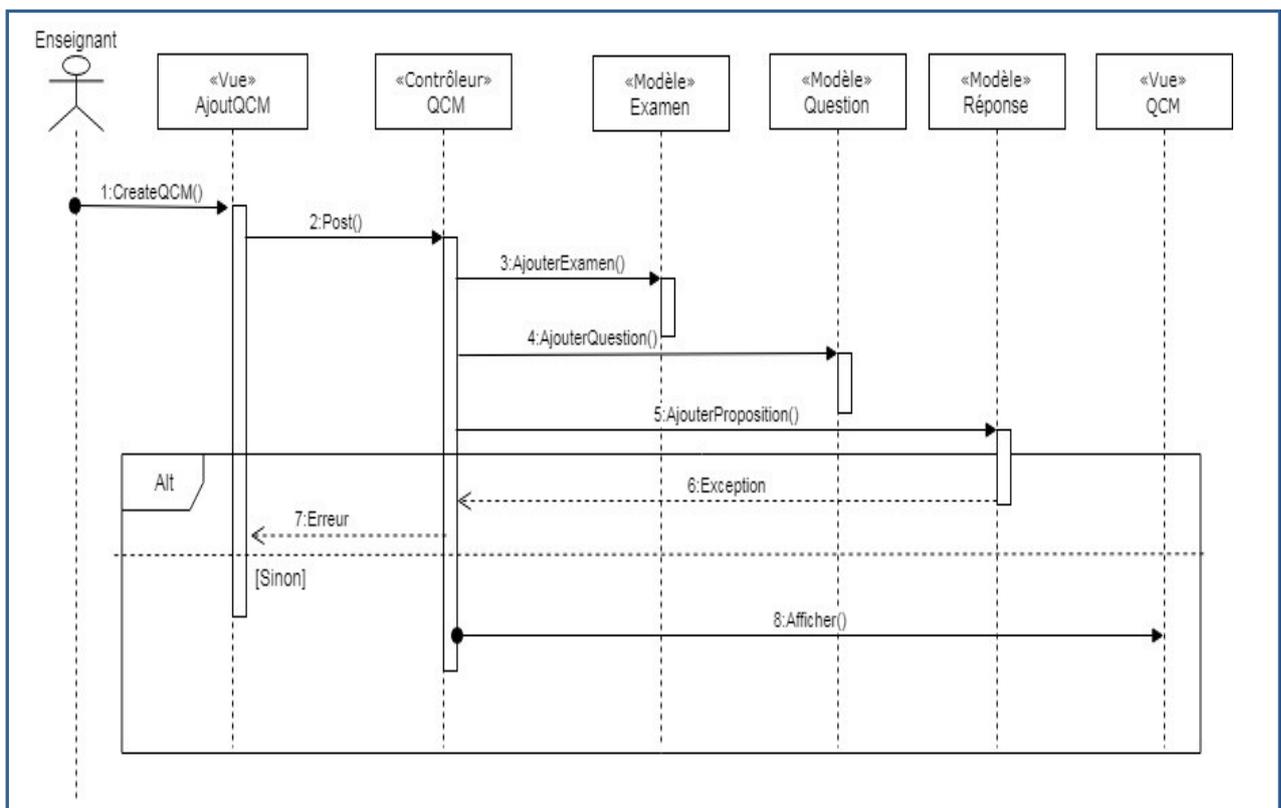


Figure II.6. Diagramme de séquence «préparation examen (création QCM)».

3.2.5 Passer examen :

Une fois l'examen lancé par l'enseignant, l'étudiant se rend sur son espace personnel dans la rubrique de l'examen, clique sur le bouton 'Commencer'. Le système interroge le modèle question et proposition une à une. L'étudiant coche la réponse qu'il juge juste et valide, ces dernières seront transmises par la méthode Post () au contrôleur qui va à son tour les insérer dans le modèle Choix, ainsi sa note est calculé automatiquement. L'examen est tenu par la contrainte de temps, une fois écoulé, l'interface se bloque.

Le diagramme suivant illustre cet enchaînement.

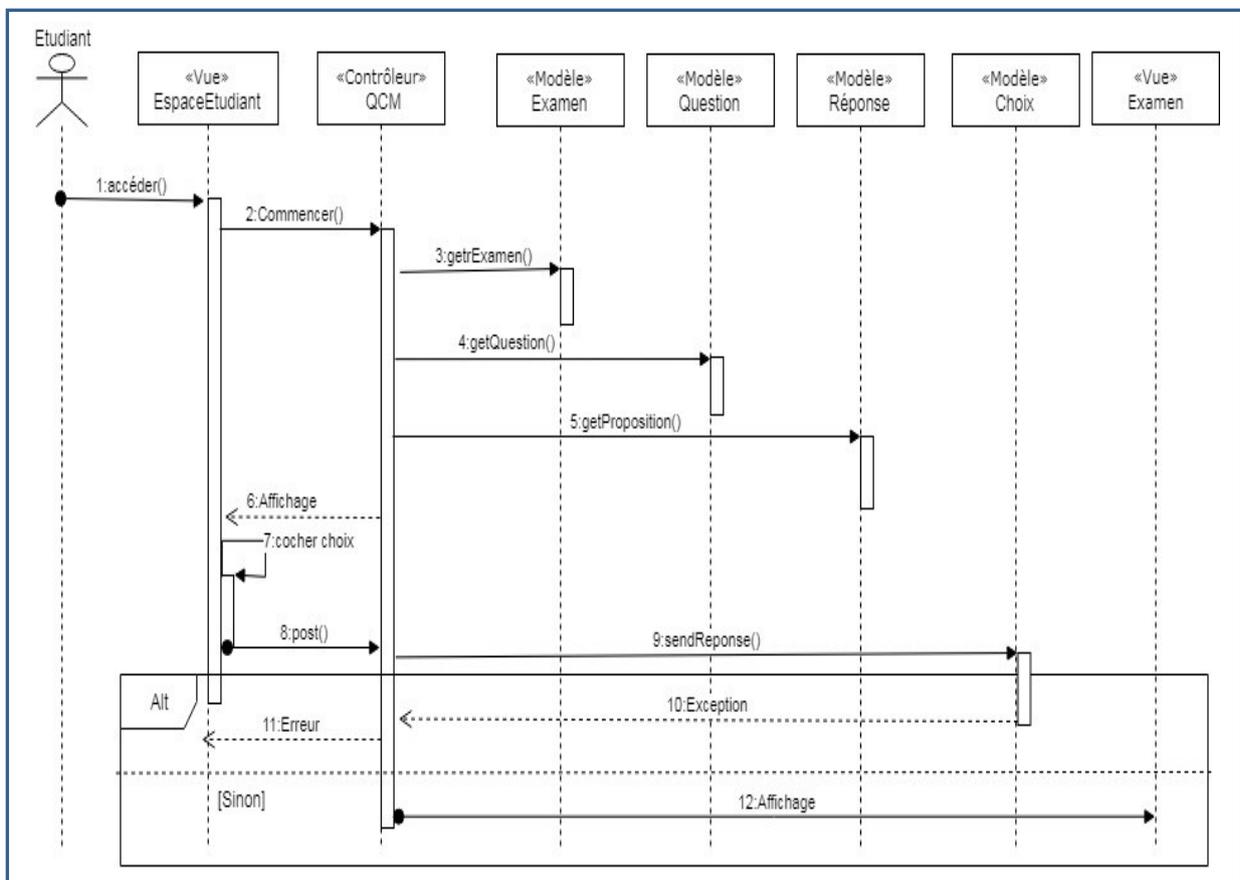


Figure II.7 : Diagramme de séquence de «Passer examen».

3.2.6 Envoie Message :

Pour envoyer un message l'utilisateur se rend sur la fiche de l'utilisateur avec lequel il veut communiquer, il saisit le message qu'il veut envoyer clique sur le bouton "envoyer", les données saisi seront envoyé au contrôleur via la méthode Post(), le contrôleur l'envoi au modèle Message via la méthode sendMSG () et crée un objet de type Message grâce à son constructeur.

Le diagramme suivant illustre cet enchaînement.

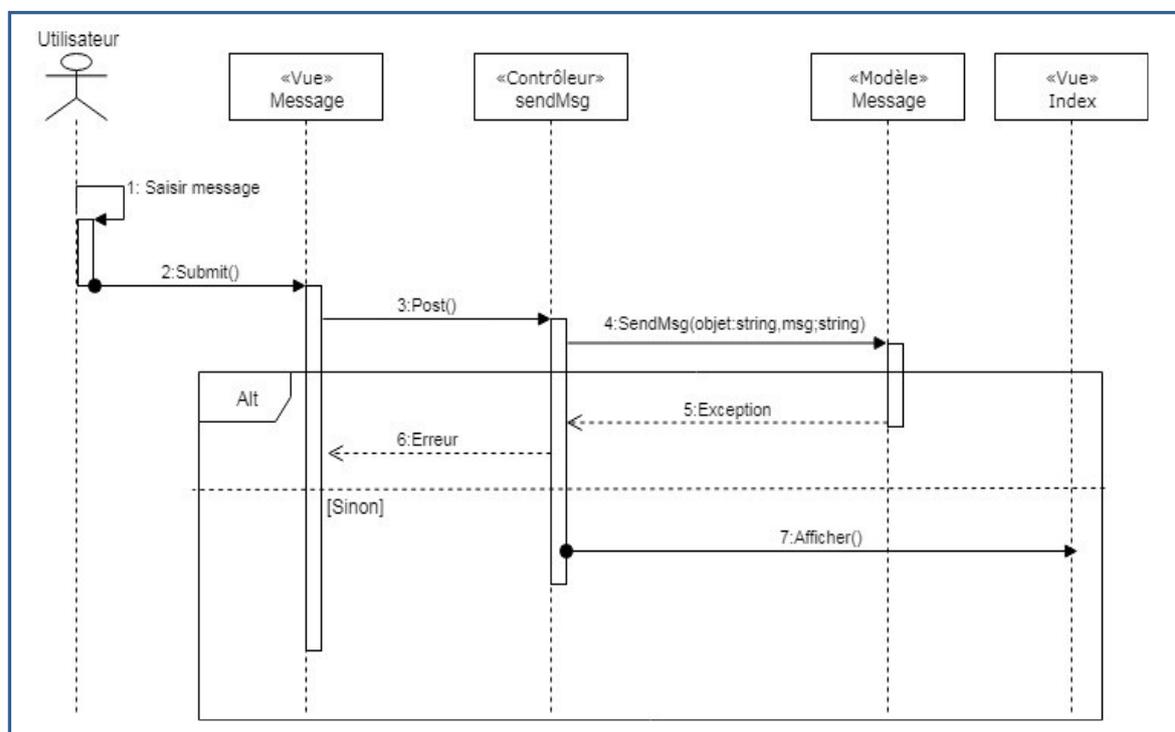


Figure II.8 : Diagramme de séquence «Envoi Message».

3.3 Diagramme d'activité :

Un diagramme d'activité permet de modéliser un processus interactif, global ou partiel pour un système donné (logiciel, système d'information). Il est recommandable pour exprimer une dimension temporelle sur une partie du modèle, à partir de diagrammes de classes ou de cas d'utilisation.

En ce qui suit nous illustrons quelques diagrammes d'activités relatifs à notre projet.

3.3.1 Diagramme d'activité Gestion utilisateur :

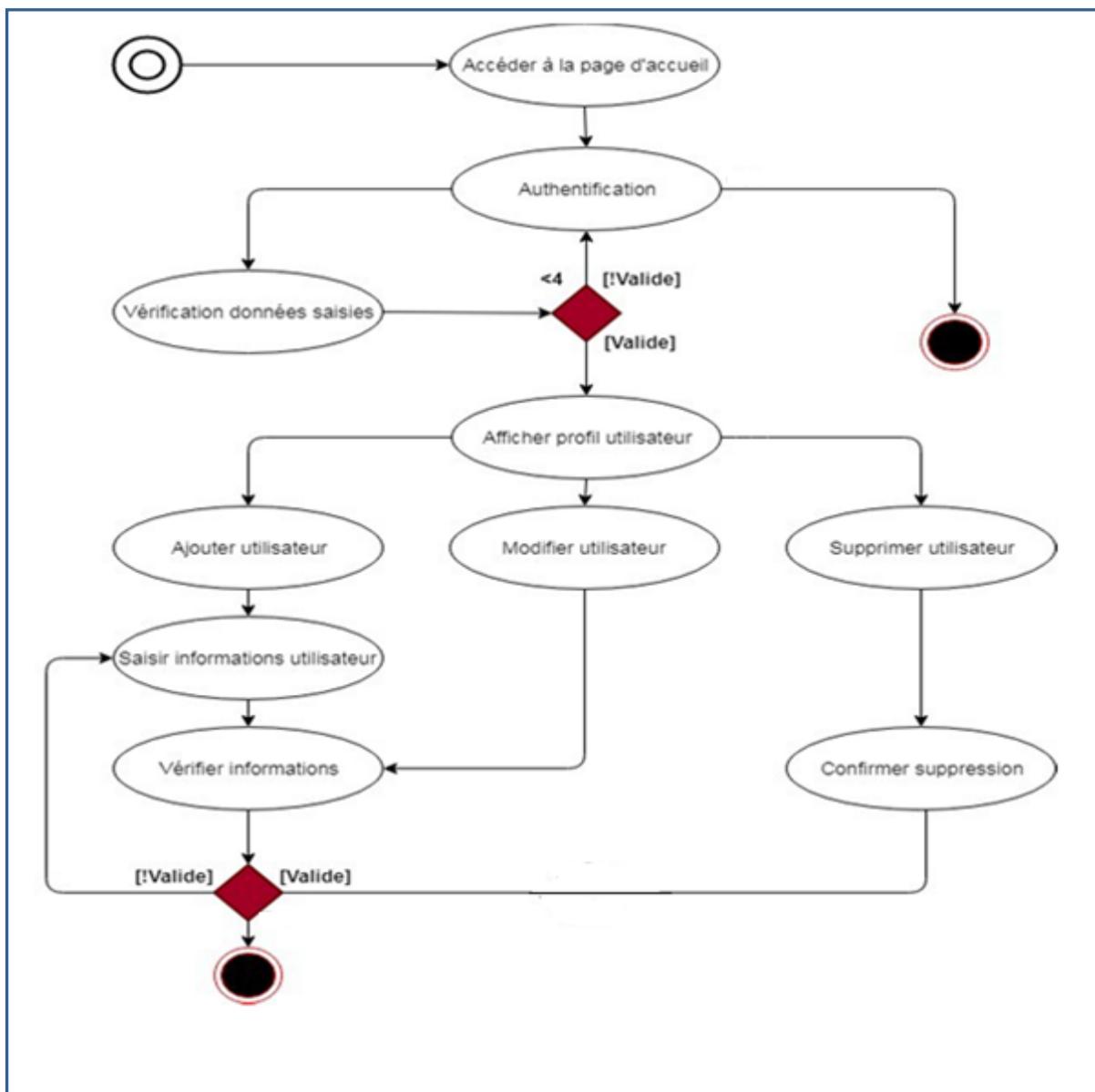


Figure II.9 : *Diagramme d'activité «Gestion utilisateur».*

3.3.2 Diagramme d'activité Gestion des cours :

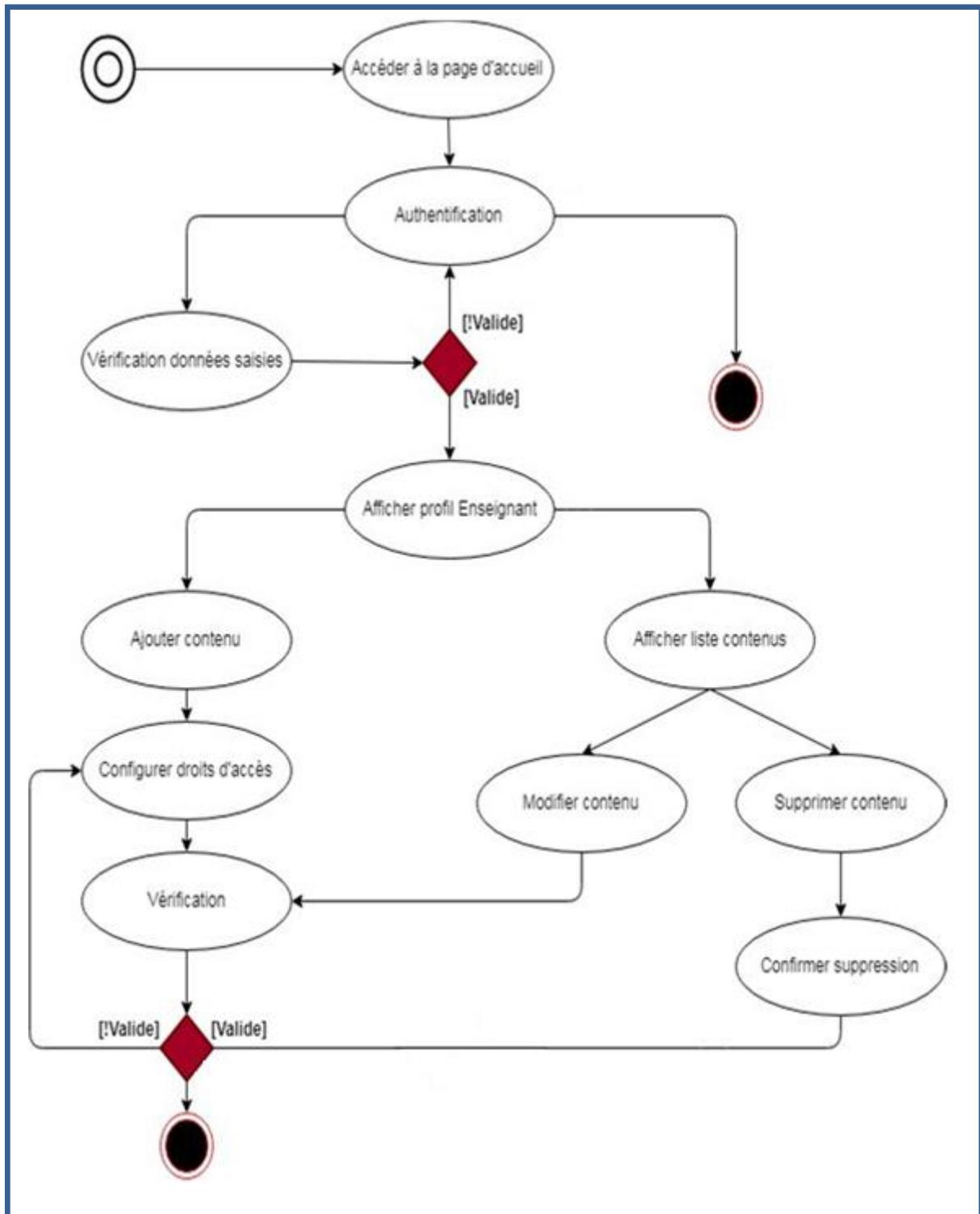


Figure II.10 : Diagramme d'activité «Gestion des cours».

3.3.3 Diagramme d'activité Consultation messagerie :

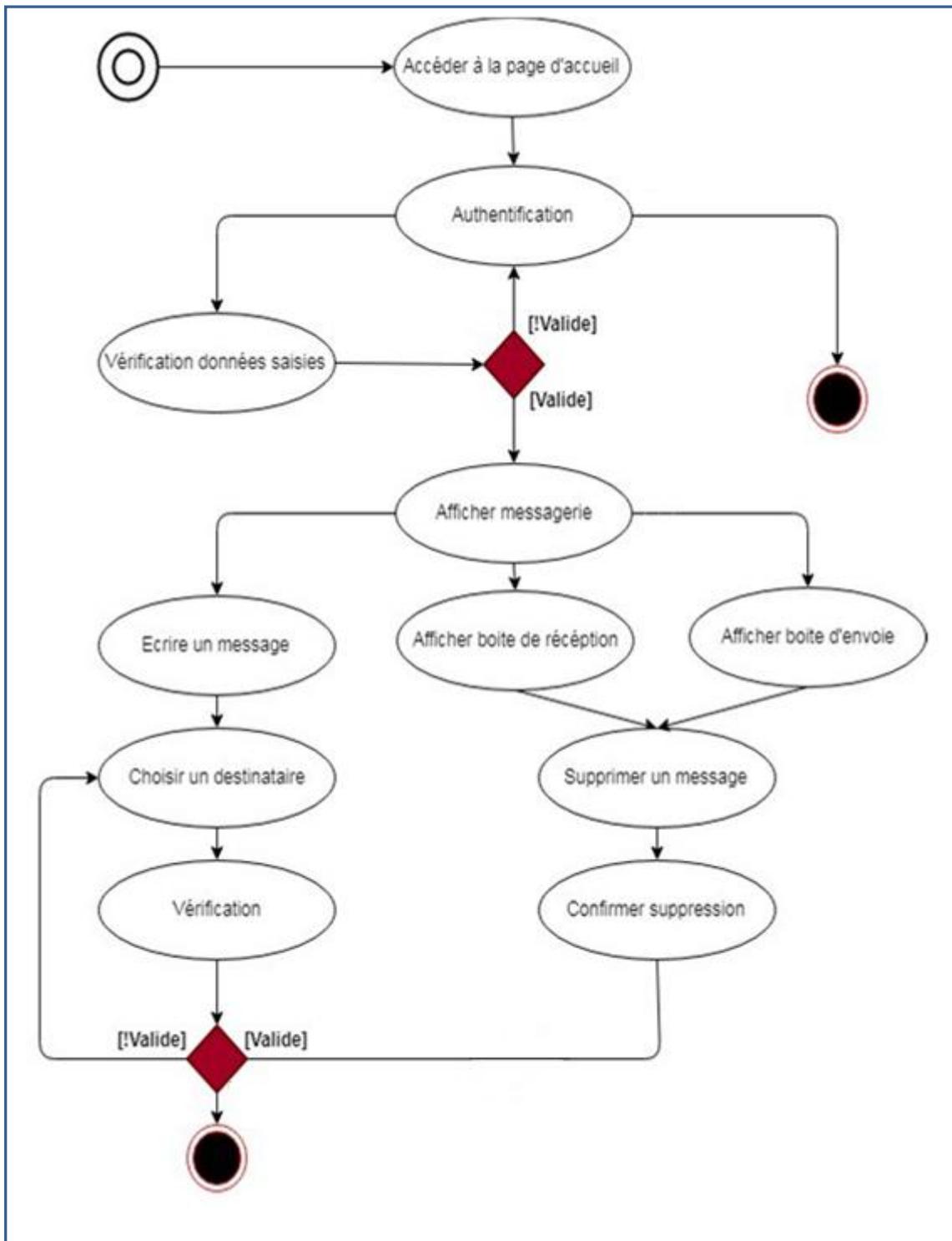


Figure II.11 : Diagramme d'activité «Consultation messagerie».

3.4 Diagramme de classes :

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.

Le diagramme de classe de notre projet est représenté comme suit :

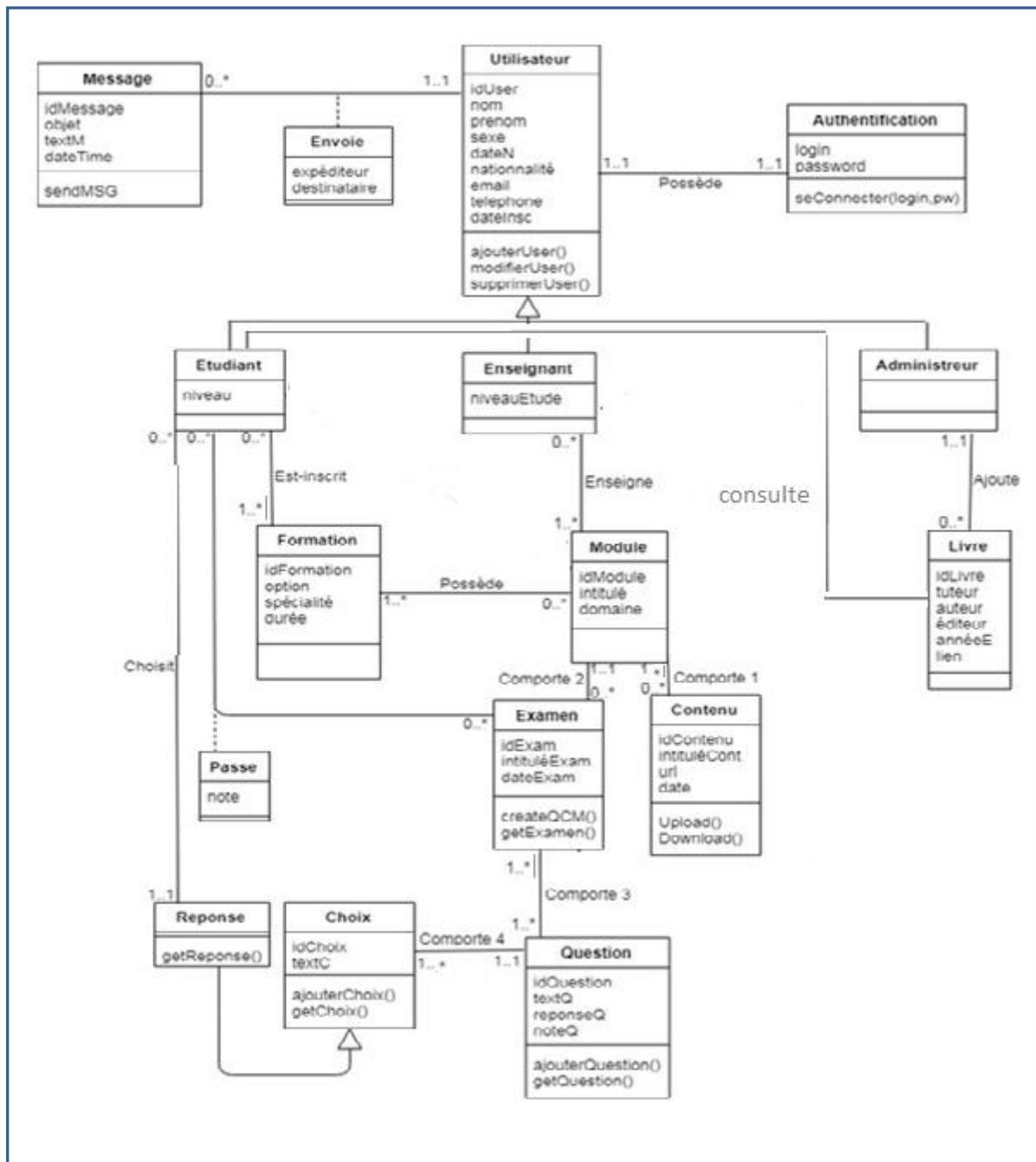


Figure II.12 : Diagramme de classes général.

3.4.1 Description des classes :

Nous décrivons en détail toutes les classes utilisés dans le diagramme de classes précédant.

Classe utilisateur:

La classe utilisateur représente trois acteurs principaux qui sont : l'enseignant, l'étudiant et l'administrateur. Ces derniers peuvent s'échanger des messages, chaque personne possède un nombre d'attributs qui permettent d'identifier un utilisateur donné.

Classe Étudiant:

La classe étudiant représente les étudiants inscrits, elle hérite de la classe utilisateur. Chaque étudiant est inscrit dans une formation ainsi il peut consulter et télécharger les différents contenu de chaque module. L'étudiant passe les examens sous forme de QCM à l'aide de la méthode faireQCM ().

Classe Enseignant:

Cette classe représente les enseignants, elle hérite de la classe utilisateur. Chaque enseignant est caractérisé par son domaine.

L'enseignant insère ses différents contenus (cours, TD, TP) à l'aide de la méthode upload(). Il crée les Examens sous forme de QCM.

Classe Administrateur :

La classe administrateur représente le responsable de la plateforme, elle hérite de la classe utilisateur. L'administrateur a la possibilité d'inscrire les enseignants et les étudiants et de modifier leurs propres informations, ainsi il crée les parcours de formations en ajoutant les différents modules.

Classe Module :

Un module peut appartenir à plusieurs formations, et est suivi par plusieurs étudiants. Chaque module a ses différents contenus dont les cours, TD et TP, et comporte aussi plusieurs examens.

Classe Examen :

Chaque examen est caractérisé par une date et une durée, il est concerné par un et un seul module. L'examen comporte plusieurs questions et chaque étudiant passant l'examen obtient une note.

Classe Question :

La classe question est concernée par un seul examen et chaque question comporte plusieurs propositions de réponses.

Classe Choix :

La classe choix représente les réponses proposées pour la question, parmi elles se retrouve la réponse correcte.

Classe Réponse :

C'est le choix de l'étudiant comme réponse pour la question. L'étudiant va choisir une seule parmi les propositions qui va être comparée à la réponse juste déclarée par l'enseignant.

Classe Contenu :

La classe contenue est caractérisé par l'attribut genre qui peut être un cours, TD ou TP. Ces derniers étant chargés par l'enseignant à partir de l'URL qui est l'adresse du fichier sur le serveur, pour que les étudiants puissent les consulter et les télécharger. Le contenu appartient à un et un seul module.

Classe Message :

Il s'agit de la classe contenant les messages des personnes utilisant cette application. Chaque message est caractérisé par l'heure et date, l'objet et le contenu.

Classe Formation :

La classe formation représente les différentes formations proposées, chaque formation appartient à une spécialité (BTS, Licence, Master ...) et est caractérisé par une durée qui peut être généralement plusieurs mois. La formation donc à plusieurs modules.

Conclusion :

Dans ce chapitre, nous avons présenté l'analyse des besoins et la conception de notre application en utilisant le langage de modélisation UML ainsi que l'architecture MVC (Model View Controller).

On a commencé par le diagramme de cas d'utilisation général de notre projet, suivi des diagrammes de séquences détaillé de chaque cas, ensuite on a illustré quelques cas d'utilisation avec les diagrammes d'activité, enfin nous avons clôturé avec le diagramme de classes général, ainsi qu'une description de chaque classe abordé en ce dernier.

Notamment nous pouvons entamer la phase suivante qui est la phase de réalisation de la solution qui sera détaillé dans le chapitre suivant.

Réalisation

Introduction :

Dans le chapitre précédant nous avons présenté les différentes étapes d'analyse et conception de notre application.

Nous allons entamer en ce dernier chapitre les différentes phases de la réalisation, commençant d'abord par présenter l'environnement de développement, ensuite les outils de qui ont servi à la réalisation, enfin nous allons illustrer quelques fonctionnalités de l'application avec les pages les plus importantes du système.

1. Outils de développement et Langages de programmation :

Pour pouvoir bien mener un projet informatique, il est nécessaire de choisir des technologies permettant de simplifier sa réalisation.

Pour notre cas nous avons opté pour les outils suivant :

1.1. Outils de développement :

Pour la réalisation de notre travail, nous avons utilisé un micro-ordinateur de bureau avec les caractéristiques illustré ci-dessous :

1.1.1. Caractéristiques de l'ordinateur :



Figure III.1 : Caractéristiques de l'ordinateur.

1.1.2. Le serveur :

Un serveur Web est un logiciel permettant à des clients d'accéder à des pages Web, c'est-à-dire des fichiers au format HTML à partir d'un navigateur (aussi appelé browser) installé sur leur ordinateur distant.

Un serveur Web est donc un logiciel capable d'interpréter les requêtes HTTP arrivant sur le port associé au protocole HTTP (par défaut le port 80), et de fournir une réponse avec ce même protocole.

Les principaux serveurs Web sont: Apache, Microsoft IIS (Internet Information Server), Microsoft PWS (PersonalWeb Server). [1]

Afin d'exécuter et de tester notre application durant la partie réalisation, nous avons opté pour l'utilisation d'Apache Tomcat Server.

1.1.2.1. Apache Tomcat :

Tomcat est un serveur HTTP à part entière. De plus, il gère les servlets et les JSP (par un compilateur Jasper compilant les pages JSP pour en faire des servlets). Tomcat a été écrit en langage Java. Il peut donc s'exécuter via la machine virtuelle Java sur n'importe quel système d'exploitation la supportant.

Catalina est le conteneur de servlets utilisé par Tomcat. Il est conforme aux spécifications servlet de Oracle Corporation et les JavaServer Pages (JSP).

Coyote est le connecteur HTTP de Tomcat, compatible avec le protocole HTTP 1.1 pour le serveur web ou conteneur d'application.

Jasper est le moteur JSP d'Apache Tomcat. Jasper parse les fichiers JSP afin de les compiler en code Java en tant que servlets (gérés par Catalina). Pendant son exécution, Jasper est capable de détecter et recompiler automatiquement les fichiers JSP modifiés. [2]

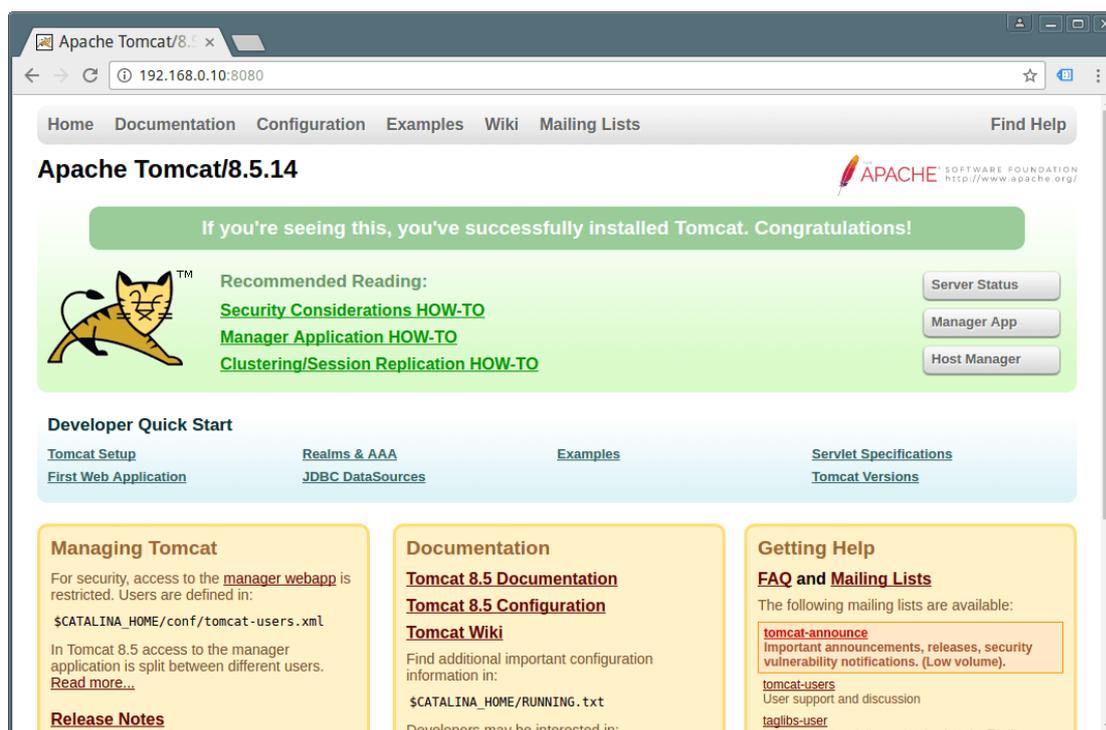


Figure III.2 : Page d'accueil serveur «Apache Tomcat 8.5.14».

Notre choix s'est porté sur le serveur Apache pour les raisons suivantes :

- Apache est aujourd'hui le serveur le plus répandu sur Internet.
- Il s'agit à la base d'une application fonctionnant sous les systèmes d'exploitation de la famille Unix, mais il a désormais été porté sur les différents autres systèmes dont Windows.
- C'est un logiciel libre.
- Il est extensible, modulaire et configurable.
- Un niveau élevé de performance des exigences matérielles modestes.
- C'est un serveur gratuit (peut être téléchargé à partir du site du groupe Apache à l'adresse 'http://www.apache.org').
- Robuste et sécurisé.

1.1.2.2. WampServer :

WampServer (anciennement WAMP5) est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans avoir à se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant trois serveurs (Apache, MySQL et MariaDB), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL.

Il dispose d'une interface d'administration permettant de gérer et d'administrer ses serveurs au travers d'un trayicon (icône près de l'horloge de Windows).

La grande nouveauté de WampServer 3 réside dans la possibilité d'y installer et d'utiliser n'importe quelle version de PHP, Apache, MySQL ou MariaDB en un clic. Ainsi, chaque développeur peut reproduire fidèlement son serveur de production sur sa machine locale. [3].

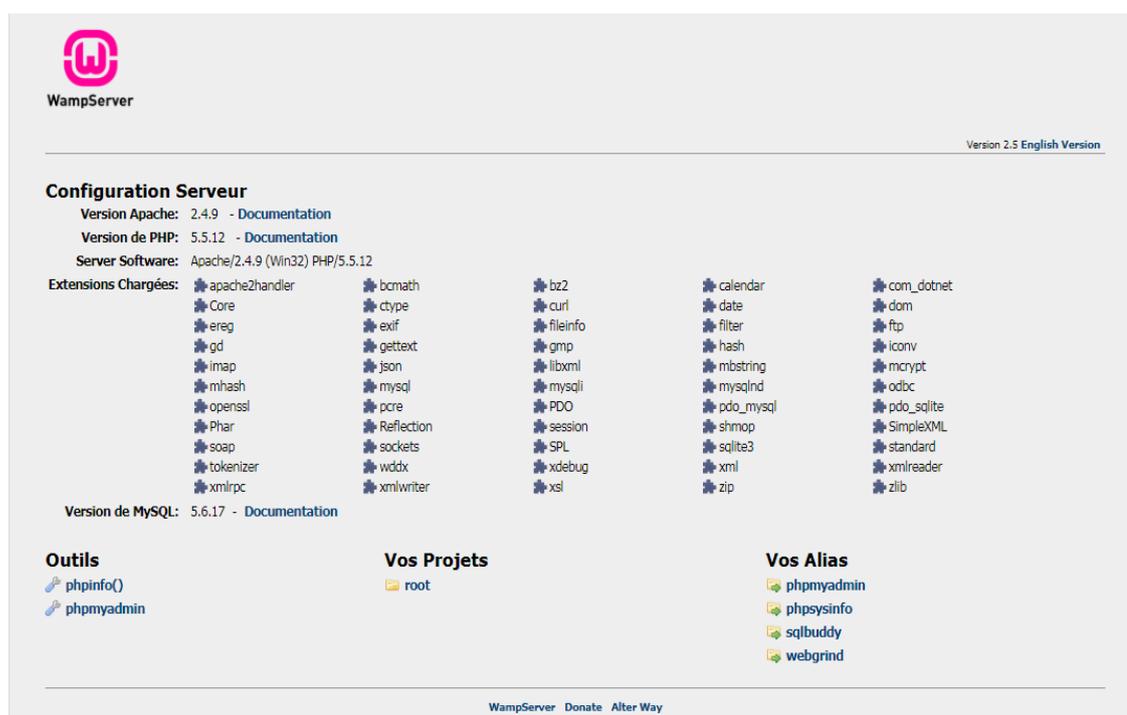


Figure III.3 : Page d'accueil «WampServer».

1.1.3. PhpMyAdmin :

C'est une interface de gestion de base de données MySQL écrite en PHP qui supporte une large gamme d'opérations pour les bases MySQL et MariaDB. C'est une application fort répandue dans le milieu de l'hébergement web mutualisé, car elle permet au fournisseur de service de donner un accès à ses utilisateurs à leur propre base de données, afin d'y effectuer des recherches, des opérations de maintenance (sauvegarde, optimisation), de créer des tables, ... [5].

Fonctionnalité de PhpMyAdmin:

- Création et suppression de base de données.
- Création, modification, copie et suppression de table.
- Edition, ajout et suppression de champ.
- Exécution des commandes SQL.
- Création d'index.
- Chargement de fichier dans des tables.

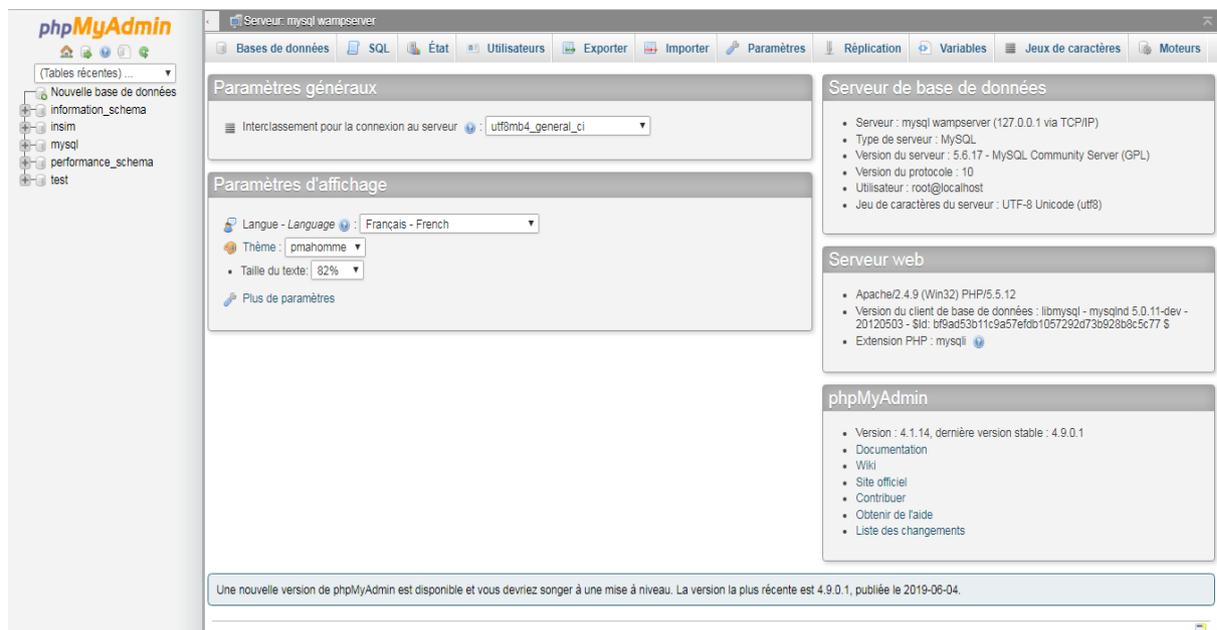


Figure III.4 : Page d'accueil de «PhpMyAdmin».

1.1.4. Eclipse IDE (IntegratedDevelopment Environnement) :

Eclipse IDE est un environnement de développement intégré libre (le terme *Eclipse* désigne également le projet correspondant, lancé par IBM) extensible, universel et polyvalent, permettant potentiellement de créer des projets de développement mettant en oeuvre n'importe quel langage de programmation. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions.[6].

De nombreux langages sont d'ores et déjà supportés par Eclipse IDE parmi lesquels :Java, RPG pour system I, C#, C++, C, Objective Caml, Python, Perl, Ruby, COBOL, Pascal, PHP, JavaScript, XML, HTML, XUL, SQL, ActionScript, Coldfusion.

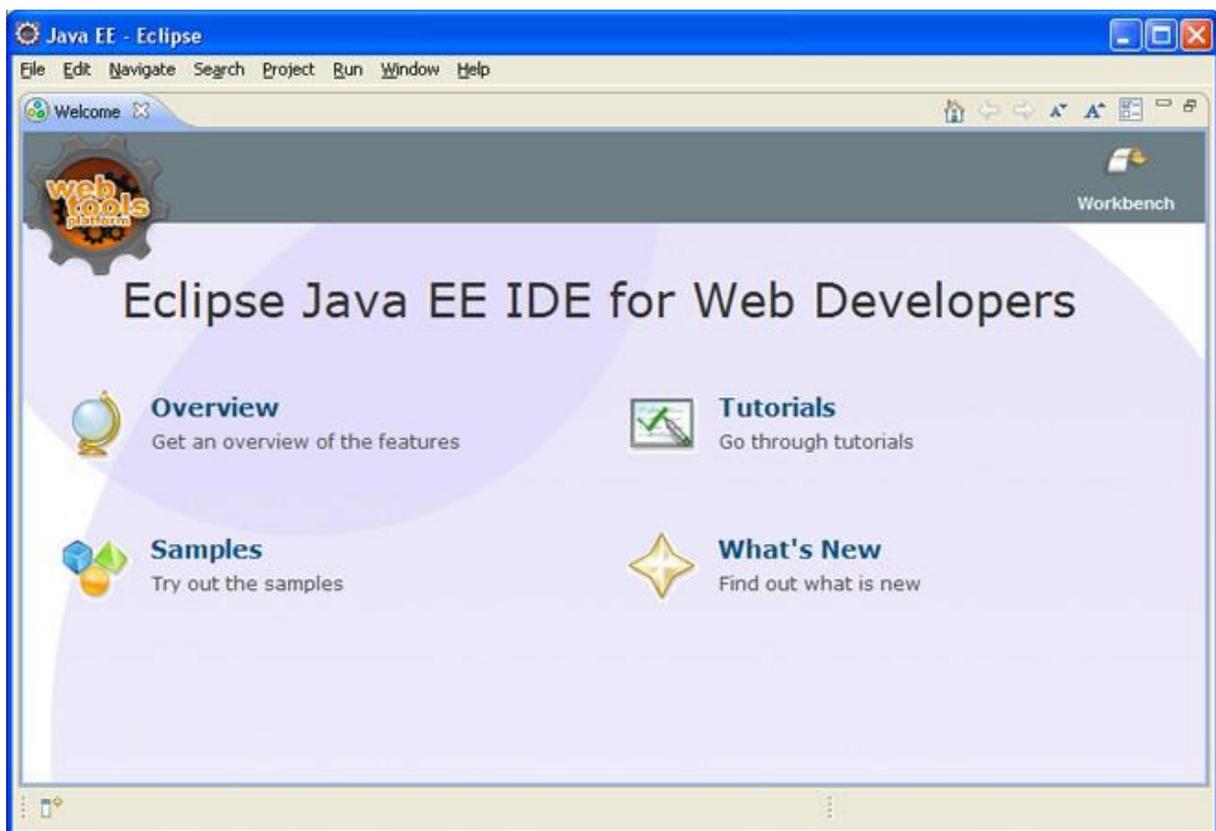


Figure III.5 : page d'accueil «Eclipse IDE».

1.2. Langages de programmation :

1.2.1. JAVA :

Java est un langage de programmation informatique orientée objet créé par "Sun Micro Systems" en 1995. Java est rapide, sécurisé et fiable. La particularité et l'objectif central de Java sont que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX (Uniplexed Information and Computing Service), Windows, Mac OS (Operating System) ou GNU (GNU's Not UNIX)/Linux, avec peu ou pas de modifications. Pour cela, diverses plates-formes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java. [7].

1.2.2. HTML 5 :

L'HyperText Markup Language, généralement abrégé HTML, est le langage de balisage conçu pour représenter les pages web. C'est un langage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et logiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie et des programmes informatiques. Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. [8].

1.2.3. CSS3:

Les feuilles de styles (en anglais "Cascading Style Sheets", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML ou XML. Les styles permettent de définir des règles appliquées à un ou plusieurs documents HTML. Ces règles portent sur le positionnement des éléments, l'alignement, les polices de caractères, les couleurs, les marges et espacements, les bordures, les images de fond, etc.

Le but de CSS est séparer la structure d'un document HTML et sa présentation. En effet, avec HTML, on peut définir à la fois la structure (le contenu et la hiérarchie entre les différentes parties d'un document) et la présentation. [9].

1.2.4. JavaScript :

Le JavaScript est un langage informatique utilisé sur les pages web. Ce langage a la particularité de s'activer sur le poste client, en d'autres mots c'est votre ordinateur qui va recevoir le code et qui devra l'exécuter. C'est en opposition à d'autres langages qui sont activés côté serveur. L'exécution du code est effectuée par votre navigateur internet tel que Chrome Firefox ou Internet Explorer. L'une des choses primordiales à savoir est de bien se rendre compte que le JavaScript n'a aucun rapport avec le Java qui est un autre langage informatique. La particularité du JavaScript consiste à créer des petits scripts sur une page HTML

Dans le but d'ajouter une petite animation ou un effet particulier sur la page. Cela permet en général d'améliorer l'ergonomie ou l'interface utilisateur, mais certains scripts sont peu utiles et servent surtout à ajouter un effet esthétique à la page. L'intérêt du JavaScript est d'exécuter un code sans avoir à recharger une nouvelle fois la page. La technique AJAX (Asynchrone JavaScript And XML) utilise grandement le JavaScript dans le but d'interagir sur la page de manière dynamique. [10].

2. Modèle Relationnel :

Le modèle relationnel est une manière de modéliser les informations contenues dans une base de données qui repose sur des principes mathématiques mis en avant par **Edgar Frank Codd**(l'inventeur du model relationnel des SGBDR). [11].

2.1. Les règles pour le système relationnel de base des données :

Le modèle relationnel repose sur les 12 règles suivantes :

1. La règle informationnelle : Toutes les informations dans la base des données sont explicites et portent la même structure logique et sont représentées par valeurs dans les tableaux.

2. La règle de certitude :

Toutes les données sont dans le modèle relationnel accessible par la combinaison de nom de tableau, la valeur de clé principale et le nom de la colonne.

3. Le traitement des valeurs nulles systématique : Les valeurs nulles sont bien adoptées par le modèle relationnel et pour représentation de l'information qui n'est pas définie et ne sont pas dépendantes du type de données.

4. Le catalogue en ligne basé sur le modèle relationnel :

La description de la base des données est exprimée sur le même niveau logique et de même manière que les dates de client. Alors un utilisateur autorisé peut appliquer la même langue relationnelle pour appliquer sa demande comme utilisateur qui travaille avec les données.

5. La langue de définition des données :

Le système relationnel peut soutenir plusieurs langues et modes pendant l'utilisation de terminal. Mais il faut qu'il y ait au moins une langue vaste qui puisse définir la structure des données, la manipulation interactive, restrictions intégrales, autorisation pour accéder à la base des données, les commandes transactionnelles etc. (SQL)

6. La règle de création de vue :

Toutes les vues sont théoriquement possibles et sont aussi créées par système.

7. Le pouvoir de créer, insérer et effacer :

Il est possible de préserver les règles relationnelles sur les relations de base et aussi relations dérivées non seulement en regardant les données, mais aussi pendant les opérations d'intersection, l'union et effacement des données.

8. L'indépendance physique des données :

Les programmes d'application sont indépendants de la structure physique des données.

9. L'indépendance de la logique des données :

Les programmes d'application sont indépendants de la structure logique de base des données.

10. L'indépendance intégrale :

Les restrictions intégrales doivent être définies par les moyens de la base des données relationnelle ou son langage et doivent être situées dans le catalogue et non dans le programme d'application.

11. L'indépendance de la distribution :

Il est nécessaire que le système de gestion de base des données relationnel soit capable d'être implémenté sur les autres architectures informatiques.

12. La règle d'accès sur la base des données :

Si le système relationnel a la langue de niveau bas, on ne peut pas utiliser ce niveau pour exprimer les restrictions intégrales et il est nécessaire de les exprimer en langue relationnelle de niveau supérieur.

2.2. Le modèle relationnel du projet :

Le passage vers le modèle relationnel donne lieu à un modèle logique de données qu'on écrit sous une forme relationnelle comme suit :

1. Utilisateur (**idUser**, nom, prénom, sexe, dateN, nationalité, email, telephone, dateIns, #login);
2. Authentification (**login**, password, #idUser);
3. Etudiant (**#idUser**, niveau, #idFormation);
4. Enseignant (**#idUser**, niveauEtude);
5. Administrateur (**#idUser**);
6. Livre (**idLivre**, tuteur, auteur, éditeur, annéeE, lien, #idUser);
7. Message (**idMessage**, objet, textM, DateTime);
8. Formation (**idFormation**, option, spécialité, durée, filiale);
9. Module (**idModule**, intitulé, domaine);
10. Examen (**idExam**, intituléExam, dateExam, #idModule);
11. Contenu (**idContenu**, intituléCont, url, date, #idModule);
12. Question (**idQuestion**, textQ, reponseQ, noteQ, #idExam);
13. Choix (**idChoix**, textC, #idQuestion);
14. Reponse (**#idChoix**, #idQuestion, #idUser);

2.3. Le modèle physique de données :

Dans la méthode Merise, le modèle physique des données (MPD) consiste à implanter une base de données dans un SGBDR.

Le langage utilisé pour ce type d'opération est le SQL. [12].

Ce modèle nous donne la représentation physique de l'ensemble des tables de la base de données de notre système réalisé comme suit:

2.3.1. Table «Utilisateur»:

Nom du champ	Désignation	Type de données	Clé(s).
idUser	Identifiant de l'utilisateur.	Int (10)	Primaire
nom	Nom de l'utilisateur.	Varchar (100)	
prénom	Prénom de l'utilisateur.	Varchar (100)	
Sexe	Sexe de l'utilisateur		
dateN	Date de naissance de l'utilisateur	Date	
nationalité	Nationalité de l'utilisateur	Varchar (100)	
email	Adresse Email de l'utilisateur.	Varchar (100)	
téléphone	Numéro de téléphone de l'utilisateur.	Number (10)	
dateInsc	Date d'inscription de l'enseignant/étudiant.	DateTime	
login	Login de l'utilisateur.	Varchar(100)	Étrangère

Tableau III.1 : Table «Utilisateur».

2.3.2. Table «Authentification»:

Nom du champ	Désignation	Type de données	Clé(s).
login	Login de l'utilisateur.	Varchar (100)	Primaire
password	Mot de passe de l'utilisateur.	Varchar (100)	
idUser	L'identifiant de l'utilisateur.	Int (10)	Étrangère

Tableau III.2 : *Table «Authentification».***2.3.3. Table «Etudiant»:**

Nom du champ	Désignation	Type de données	Clé(s).
idUser	Identifiant de l'étudiant	Int(10)	Primaire Étrangère
niveau	Niveau scolaire de l'étudiant	Varchar(100)	
idFormation	Identifiant de la formation	Int(10)	Etrangère

Tableau III.3 : *Table «Etudiant».***2.3.4. Table «Enseignant»:**

Nom du champ	Désignation	Type de données	Clé(s).
idUser	Identifiant de l'enseignant	Int(10)	Primaire Étrangère
niveauEtude	Niveau d'étude de l'enseignant	Varchar(100)	

Tableau III.4 : *Table «Enseignant».*

2.3.5. Table «Administrateur»:

Nom du champ	Désignation	Type de données	Clé(s).
idUser	Identifiant de l'administrateur	Int(10)	Primaire Étrangère

Tableau III.5 : *Table «Administrateur».***2.3.6. Table «Livre» :**

Nom du champ	Désignation	Type de données	Clé(s).
idLivre	Identifiant du livre.	Int(10)	Primaire
tuteur	Le tuteur du livre.	Varchar(100)	
auteur	L'auteur du livre.	Varchar(100)	
editeur	L'éditeur du livre.	Varchar(100)	
année	L'année d'édition du livre.	Year	
lien	L'url du livre.	Varchar(100)	

Tableau III.6: *Table «Livre».***2.3.7. Table «Message» :**

Nom du champ	Désignation	Type de données	Clé(s).
idMessage	Identifiant de message.	Int(10)	Primaire
objet	Objet du message.	Varchar(100)	
textM	Le contenu du message.	Varchar(1000)	
dateTime	Date d'envoi du message.	Date	

Tableau III.7 : *Table «Message».*

2.3.8. Table «Formation» :

Nom du champ	Désignation	Type de données	Clé(s).
idFormation	Identifiant de la formation.	Int(10)	Primaire
option	Catégorie de la formation.	Varchar(100)	
spécialité	Intitulé de la formation.	Varchar(100)	
durée	Durée de la formation.	Varchar(100)	
filiale	La filiale de la formation.	Varchar(100)	

Tableau III.8 : *Table «Formation».***2.3.9. Table «Module» :**

Nom du champ	Désignation	Type de données	Clé(s).
idModule	Identifiant du module	Int(10)	Primaire
intitulé	Intitulé du module	Varchar(100)	
domaine	Domaine d'études du module.	Int(10)	

Tableau III.9 : *Table «Module».***2.3.10. Table «Examen» :**

Nom du champ	Désignation	Type de données	Clé(s).
idExam	Identifiant de l'examen	Int(10)	Primaire
intituléExam	Désignation de l'examen	Varchar(20)	
dateExam	Date de l'examen	Date	
idModule	Identifiant du module	Int(10)	Etrangère

Tableau III.10 : *Table «Examen».*

2.3.11. Table «Contenu» :

Nom du champ	Désignation	Type de données	Clé(s).
idContenu	Identifiant de contenu	Int(10)	Primaire
intituléCont	L'intitulé du contenu.	Varchar(100)	
url	L'url du contenu.	Varchar(100)	
date	Date de partage du contenu.	Date	
idModule	Identifiant du module.	Int(10)	Etrangère

Tableau III.11 : *Table «Contenu».***2.3.12. Table «Question» :**

Nom du champ	Désignation	Type de données	Clé(s).
idQuestion	Identifiant de la question	Int(10)	Primaire
textQ	Intitulé de la question	Varchar(100)	
réponseQ	Réponse correcte de la question	Varchar(10)	
noteQ	Note associé à la question	Decimal(1,2)	
idExam	Identifiant de l'examen	Int(10)	Etrangère

Tableau III.12 : *Table «Question».***2.3.13. Table «Choix» :**

Nom du champ	Désignation	Type de données	Clé(s).
idChoix	Identifiant de la proposition.	Int(10)	Primaire
textC	Contenu de la proposition.	Varchar(50)	
idQuestion	Identifiant de la question.	Int(10)	Étrangère

Tableau III.13 : *Table «Choix».*

2.3.14. Table «Réponse» :

Nom du champ	Désignation	Type de données	Clé(s).
idChoix	Identifiant de la réponse choisie.	Int(10)	Primaire
idQuestion	Identifiant de la question.	Int(10)	Étrangère
idUser	Identifiant de l'étudiant.	Int(10)	Étrangère

Tableau III.14 : *Table «Réponse».*

3. Présentation de quelques interfaces :

En ce qui suit nous allons présenter quelques interfaces de notre application

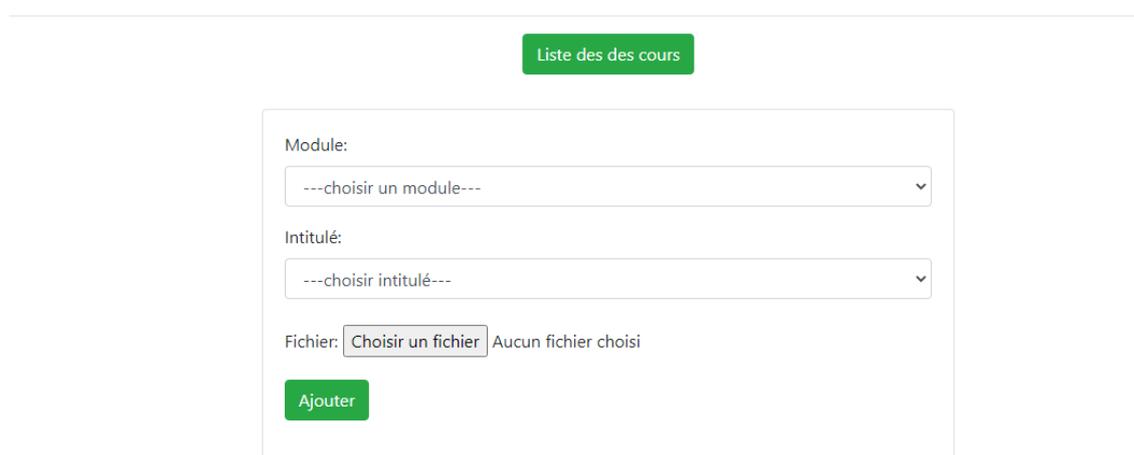
3.1. Page d'authentification:



The screenshot shows the user authentication interface. At the top, there is a red navigation bar with the text "E-LEARNING" on the left and a menu of links: "Accueil", "Enseignant", "Etudiant", "Formation", "Messagerie", "Mediatheque", and "Contact". Below the navigation bar, the main heading reads "Bienvenue dans notre Plateforme E-learning". Underneath, there is a sub-heading "Saisissez votre Login et votre Mot de passe". The form contains two input fields: "Login" and "Mot de passe", both with placeholder text. A green "Connexion" button is positioned below the password field. At the bottom of the page, there is a black navigation bar with the same menu of links as the top bar.

Figure III.6 : Page «Authentification utilisateur».

3.2. Page AjouterContenu:



The screenshot shows the "Ajouter Contenu" page. At the top, there is a green button labeled "Liste des des cours". Below this, there is a form with three fields: "Module:" with a dropdown menu showing "--choisir un module--", "Intitulé:" with a dropdown menu showing "--choisir intitulé--", and "Fichier:" with a text input field containing "Choisir un fichier" and the text "Aucun fichier choisi" to its right. A green "Ajouter" button is located at the bottom of the form.

Figure III.7 : Page «AjouterContenu».

3.3. Page Compte Enseignant:

E-LEARNING Accueil Enseignant Etudiant Formation Messagerie Mediatheque Contact

Ajouter un nouveau enseignant

Liste des enseignants

Nom

Prénom

Date de naissance

Waya de naissance

E-mail

Figure III.8 : Page «Compte Enseignant».

3.4. Page liste des Enseignants:

E-LEARNING Accueil Enseignant Etudiant Formation Messagerie Mediatheque Deconnexion

Liste des enseignants

Ajouter un nouveau enseignant

ID	Nom	prenom	daten	nationalite	email	téléphone	dateinsc	login	Sexe	password	niveauEtude	
6	aa	aa	2020-01-01	aa	aa	111	2020-01-01	aaa	f	aaa	aaaa	Edit Delete
7	aa	aa	2020-01-01	aaa	aaa	111	2020-01-01	a	a	aaaaa	aaaa	Edit Delete
8	aa	aa	2020-01-01	aa	aa	111	2020-01-01	ddd	d	dddd	dddd	Edit Delete
9	aaa	aaa	2020-01-01	sss	sss	111	2020-01-01	aaa	f	cccc	cccc	Edit Delete

Figure III.9 : Page «liste des Enseignants».

Conclusion :

Dans ce chapitre nous avons présenté les outils et l'environnement développement de l'application. Nous avons en outre les composants de la base de données, puis pour terminer nous avons présenté quelques interfaces de l'application.

Conclusion Générale

Conclusion Générale

L'objectif de notre travail est la réalisation d'une application E-Learning ,et cela grâce à la technologie JavaEE et la plateforme WAMP qui intègre MySQL pour gérer la base de données.

La réalisation de cette application nous a permis d'exploiter toutes les connaissances que nous avons acquies durant notre cursus, d'approfondir nos connaissances théoriques et pratiques, de nous familiariser avec HTML, JavaScript, MySQL et JavaEE, et de maîtriser les concepts et les outils de la méthode de conception UML.

Le travail que nous avons réalisé peut être amélioré et enrichi afin d'en faire un système plus performant.

Parmi les perspectives à prendre en compte pour améliorer le fonctionnement du système, nous citons :

- Intégration d'un moteur de recherche par mots clés.
- Mettre en place un espace qui permet aux étudiants de débattre et poser des questions sur le cours.
- Notifier les étudiants par Email sur les nouveaux contenus partagés.
- Téléconférence.

Enfin, nous espérons que notre modeste projet puisse être une référence et un outil facilitant le développement d'applications E-learning pour les promotions à venir.

Annexe

Introduction :

Ce chapitre à pour but de traduire les modèles d'analyse et de conception dans un langage de programmation cela nécessite un bon choix de solution technique, l'architecture adapté et les outils de développement nécessaire qui permettent la réalisation de l'application.

Dans ce chapitre on a donné une description de l'architecture et la modélisation de celle-ci sous forme de diagramme de déploiement, la plateforme choisit ainsi les outils et langages de programmation utilisés et à la fin quelques écrans de l'application réalisée.

Dans ce chapitre on va étudier une technologie d'architecture logicielle (Java Entreprise Edition) qui est nécessaire pour la réalisation des applications des entreprises.

1. Architecture Logicielle :

1.1. Définition :

L'architecture logicielle décrit d'une manière symbolique et schématique les différents éléments d'un ou de plusieurs systèmes informatiques, leurs interrelations et leurs interactions.

Contrairement aux spécifications produites par l'analyse fonctionnelle, le modèle d'architecture, produit lors de la phase de conception, ne décrit pas ce que doit réaliser un système informatique mais plutôt comment il doit être conçu de manière à répondre aux spécifications. L'analyse décrit le « quoi faire » alors que l'architecture décrit le « comment le faire ».

2. Les modèles d'architecture :

2.1. Le modèle des 4+1 vues :

Le modèle de Kruchten dit modèle des 4 + 1 vues est celui adopté dans le processus unifié (UnifiedProcess). Ici encore, le modèle d'analyse, baptisé vue des cas d'utilisation, constitue le lien et motive la création de tous les diagrammes d'architecture.

2.1.1. La vue des cas d'utilisation :

La vue des cas d'utilisation est un modèle d'analyse formalisé par Ivar Jacobson. Un cas d'utilisation est défini comme un ensemble de scénarios d'utilisation, chaque scénario représentant une séquence d'interaction des utilisateurs (acteurs) avec le système. L'intérêt des cas d'utilisation est de piloter l'analyse par les exigences des utilisateurs. L'ensemble des cas d'utilisation du logiciel en cours de spécification est représenté par un diagramme de cas d'utilisation, chacun des scénarios de celui-ci étant décrit par un ou plusieurs diagrammes dynamiques : diagrammes d'activités, de séquence ou d'états-transitions.

2.1.2. La vue logique :

La vue logique constitue la principale description architecturale d'un système informatique. Cette vue décrit, de façon statique et dynamique, le système en termes d'objets et de classes.

La vue logique permet d'identifier les différents éléments et mécanismes du système à réaliser. Elle permet de décomposer le système en abstractions et constitue le coeur de la réutilisation. La vue logique est représentée, principalement, par des diagrammes statiques de classes et d'objets enrichie de descriptions dynamiques : diagrammes d'activités, de séquence ou d'états-transitions.

2.1.3. La vue des processus :

La vue des processus décrit les interactions entre les différents processus, threads (fils d'exécution) ou tâches, elle permet également d'exprimer la synchronisation et l'allocation des objets. Cette vue permet avant tout de vérifier le respect des contraintes de fiabilité, d'efficacité et de performances des systèmes multitâches. Les diagrammes utilisés dans la vue des processus sont exclusivement dynamiques : diagrammes d'activités, de séquence, diagrammes de communication ou d'états transitions.

2.1.4. La vue de réalisation :

La vue de réalisation permet de visualiser l'organisation des composants (bibliothèque dynamique et statique, code source...) dans l'environnement de développement. Elle permet aux développeurs de se retrouver dans le capharnaüm que peut être un projet de développement informatique. Cette vue permet également de gérer la configuration (auteurs, versions...). Les seuls diagrammes de cette vue sont les diagrammes de composants.

2.1.5. La vue de déploiement :

La vue de déploiement représente le système dans son environnement d'exécution. Elle traite des contraintes géographiques (distribution des processeurs dans l'espace), des contraintes de bandes passantes, du temps de réponse et des performances du système ainsi que de la tolérance aux fautes et aux pannes. Cette vue est fort utile pour l'installation et la maintenance régulière du système. Les diagrammes de cette vue sont les diagrammes de composants et les diagrammes de déploiement.

3. Types d'architectures :

Dans les applications Web, la communication entre le client et le serveur est réalisée selon le protocole TCP/IP qui est chargé du routage des données. Le transit des informations s'effectue selon le protocole HTTP pour le Web, les données sont alors transmises entre le client et le serveur via TCP/IP. On distingue alors deux types de clients :

➤ **3.1 Le client léger :**

Il est aussi appelé client Web car le module d'exécution est alors un navigateur. Les applications clientes sont composées de pages HTML/XHTML voir DHTML avec l'utilisation du langage client JavaScript.

➤ **3.2 Le client lourd :**

Il s'agit d'une application composée d'une interface graphique évoluée ou en mode console. Dans l'idéal, les clients lourds communicants ne contiennent que la logique présentation (affichage des données). Tous les traitements sont délégués à des composants métier distants.

Il existe deux types d'architecture pour le web

3.1. L'architecture Client/serveur :

L'architecture client/serveur est un mode de communication entre différents nœuds d'un réseau dans lequel un processus appelé le serveur agit comme fournisseur de ressources pour d'autres processus qui demandent ces ressources, ces derniers sont des clients. Le processus client et le processus serveur s'exécutent le plus souvent sur des machines différentes reliées au même réseau.

Le client est un logiciel exécuté sur de petites machines, par contre le serveur s'exécute sur de grosses machines ayant une grande capacité de traitement, la communication se fait via le réseau.

3.2 L'architecture MVC (Model View Controller) :

L'architecture MVC proposée par Sun est la solution de développement Web côté serveur qui permet de séparer la partie logique/métier de la partie présentation dans une application Web. C'est un point essentiel du développement de projets car cela permet à toute l'équipe de travailler séparément (chacun possède ses fichiers, ses logiciels de développement et ses composants). Cette architecture trouve son origine dans le langage Small Talk au début des années 1980, ce n'est donc pas un modèle (design pattern) nouveau

uniquement lié à JavaEE. L'objectif principal est de diviser l'application en trois parties distinctes : le modèle, la vue et le contrôleur. Dans l'architecture MVC, nous retrouvons :

➤ **3.2.1 La couche Vue :**

C'est la partie qui s'occupera de la présentation des données à l'utilisateur, elle retourne une vue des données venant du modèle, en d'autres termes c'est elle qui est responsable de produire les interfaces de présentation de l'application à partir des informations qu'elle dispose (page HTML par exemple).

➤ **3.2.2 La couche Contrôleur :**

C'est la couche chargée de router les informations, elle va décider qui va récupérer l'information et la traiter. Elle gère les requêtes des utilisateurs et retourne une réponse avec l'aide de la couche Modèle et Vue.

➤ **3.2.3 La couche Modèle :**

C'est la partie du code qui exécute le logique métier de l'application. Cette couche est responsable de récupérer les données, de les convertir selon les concepts de la logique de votre application tels que le traitement, la validation, l'association et toute autre tâche concernant la manipulation des données. Elle est également responsable de l'interaction avec la base de données, elle sait en quelque sorte comment se connecter à une base de données et d'exécuter les requêtes (CREATE, READ, UPDATE, DELETE) sur une base de données.

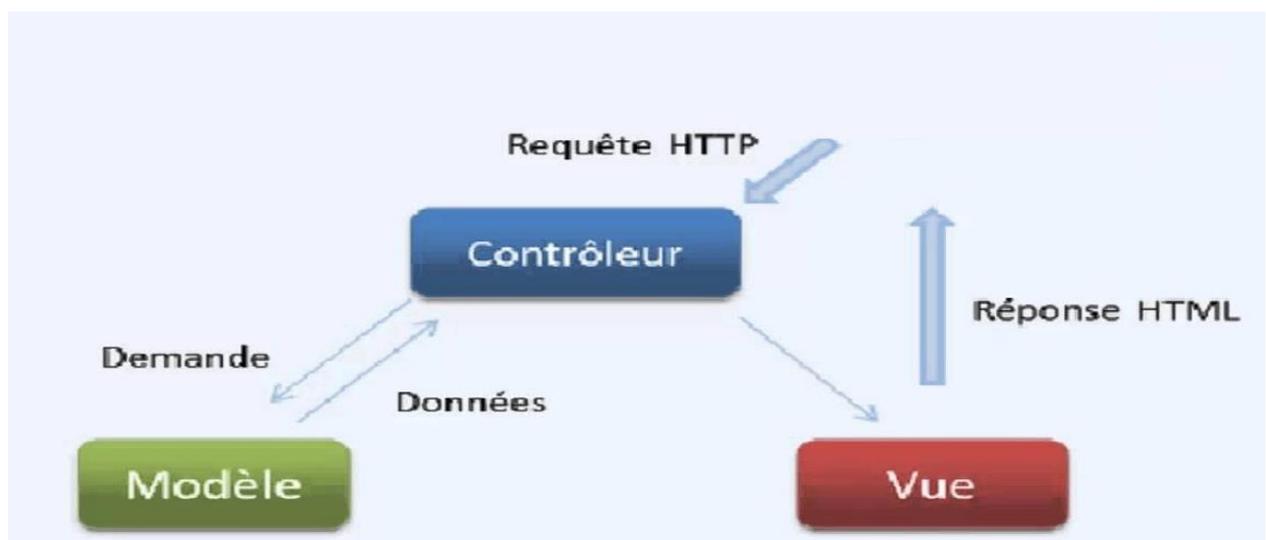


Figure 04 : *Fonctionnement de l'architecture MVC.*

4. Java Entreprise Edition (Java EE):

Le terme « Java » fait bien évidemment référence à un langage, mais également à une plate-forme : son nom complet est « Java SE » pour *Java Standard Edition*, et était anciennement raccourci « J2SE ». Celle-ci est constituée de nombreuses bibliothèques, ou API (Application Programming Interface).

Le terme « Java EE » signifie Java Enterprise Edition, et était anciennement raccourci en « J2EE ». Il fait quant à lui référence à une extension de la plate-forme standard. Autrement dit, la plate-forme Java EE est construite sur le langage Java et la plate-forme Java SE, et elle y ajoute un grand nombre de bibliothèques remplissant tout un tas de fonctionnalités que la plate-forme standard ne remplit pas d'origine. L'objectif majeur de Java EE est de faciliter le développement d'applications web robustes et distribuées, déployées et exécutées sur un serveur d'applications.

4.1 Comment JAVA EE implémente-t-il le modèle MVC :

Avec la plateforme JEE, chaque élément du modèle MVC porte un nom.

Le **Contrôleur** porte le nom de **Servlet**. Le **modèle** est en général géré par des objets Java ou des **JavaBeans**. Il peut être amené aussi à communiquer avec les bases de données pour stocker les informations. La **Vue** quant à elle est gérée par les pages **JSP** (Java Server Pages) qui sont des pages qui vont utiliser du code HTML et du code spécifique en général en JAVA.

4.1.1. La Servlet :

La servlet est une classe Java qui permet le traitement des requêtes (recevoir les requêtes HTTP envoyées depuis le navigateur (Client web) et de renvoyer une réponse http à l'utilisateur. Une servlet http doit toujours hériter de la classe `HttpServlet` qui est en effet une classe abstraite qui fournit des méthodes qui doivent être redéfinir dans la classe héritière. Parmi ces méthodes on y retrouve entre autres :

➤ **doGet() :**

Méthode qui permet de récupérer une ressource du serveur coté client.

➤ **doPost() :**

Méthode qui permet de soumettre au serveur des données de tailles variables.

4.1.2. JSP (Java Server Pages) :

Les JSPs est une technologie qui permet d'écrire très facilement des pages dynamiques en insérant des portions de code java dans une page html, destiné à l'origine à la présentation des données. Une page JSP permet de définir le design d'une page (en HTML) et d'y intégrer les données (grâce aux portions de code java).

4.1.3. EJB (Entreprise Java Beans) :

Les Javabeans sont un modèle de composants du langage Java. Ce sont des 'pièces logicielles' qui peuvent être réutilisées pour créer des programmes dans un environnement visuel de développement d'applications. Elles représentent généralement les données du monde réel. Les principaux concepts mis en jeu pour un Bean sont les suivants :

➤ **Propriétés :**

Un JavaBean doit pouvoir être paramétrable. Les propriétés permettent de paramétrer le Bean, en y stockant des données.

➤ **La sérialisation :**

La sérialisation est un processus qui permet de sauvegarder l'état d'un Bean et donne ainsi la possibilité de le restaurer par la suite.

➤ **La réutilisation :**

Un Bean est un composant n'a pas de lien direct avec la couche de présentation, et peut également être distant de la couche d'accès aux données. C'est cette indépendance qui lui donne ce caractère réutilisable.

➤ **L'introspection :**

C'est un processus qui permet de connaître le contenu d'un composant (attributs, méthodes et événements) de manière dynamique, sans disposer de son code source.

5. Les API JAVA EE :

Java EE spécifiait les infrastructures de gestion de nos applications au travers composants sous forme d'API (Application Programming Interface). La spécification de la Plateforme JAVA EE prévoit un ensemble d'extensions Java standard que chaque plateforme JEE doit prendre en charge :

➤ **JNDI (*Java Naming and Directory Interface*) :**

Permet de localiser et d'utiliser les ressources. Permet d'accéder à divers services de noms et de répertoires. Derrière JNDI il peut avoir un appel à des services tels que : CORBA, DNS, LDAP, etc.

➤ **JDBC (*Java Database Connectivity*) :**

Est une API qui permet aux programmes JAVA de se connecter et d'interagir avec les bases de données SQL

➤ **JMS (*Java Message Service*) :**

Est une API permettant aux développeurs de construire des applications professionnelles qui se servent de messages pour transmettre des données.

➤ **JTA (*Java Transaction API*) :**

Définit des interfaces standards entre un gestionnaire de transactions et les éléments impliqués dans celle-ci : l'application, le gestionnaire de ressource et le serveur.

➤ **Authentication :**

Java EE fournit des services d'authentification en se basant sur les concepts d'utilisateur, de domaines et de groupes.

6. Les Framework JEE :

Un Framework est un ensemble de composants permettant de réaliser ou de créer l'architecture et les grandes lignes d'une application. Il se présente sous diverse forme qui peut inclure tout une partie des éléments suivants :

- Un ensemble de classes généralement regroupées sous la forme de bibliothèques pour proposer des services plus ou moins sophistiqués
- Un cadre de conception reposant sur les design patterns pour proposer tout ou partie d'un squelette d'application
- Des recommandations sur la mise en oeuvre et des exemples d'utilisation
- Des normes de développement
- Des outils facilitant la mise en oeuvre.

L'objectif d'un Framework est de faciliter la mise en oeuvre des fonctionnalités de son domaine d'activité. Il doit permettre au développeur de se concentrer sur les tâches

spécifiques de l'application à développer plutôt que sur les tâches techniques récurrentes telles que : l'architecture de base de l'application, l'accès aux données, la sécurité

(authentification et gestion des rôles), le paramétrage de l'application, la journalisation des événements (logging), etc.

En ce qui concerne JAVA EE, il existe de nombreux Framework tels qu'autres JSF (Java Server Faces), Struts, Spring, Hibernate etc.

7. Avantages de Java EE :

- Meilleure organisation du code;
- Diminution de la complexité lors de la conception;
- Conception claire et efficace grâce à la séparation des données de la vue et du contrôleur;
- Possibilité de réutilisation de code dans d'autres applications;
- Un gain de temps de maintenance et d'évolution du site;
- Une plus grande souplesse pour organiser le développement du site entre différents développeurs;
- Plus de facilité pour les tests unitaires.

8. Inconvénients de Java EE :

Augmentation de la complexité lors de l'implantation; Éventuel cloisonnement des développeurs; Architecture complexe pour des petits projets;

Le nombre important de fichiers représente une charge non négligeable dans un projet.

Conclusion :

Java EE fournit plusieurs composants (API) et conteneur qui fournissent des services de bas niveaux au développeur de telle sorte que ces derniers n'ont plus besoin de développer eux même des fonctionnalités basiques mais portent toute leur attention et les savoir-faire sur le développement des fonctions métier de l'application. Quant aux conteneurs elles constituent des environnements de d'exécution pour les composants d'application Java EE et fournissent-elles aussi des services tels que la gestion des transactions, la connexion aux bases de données, etc.

Introduction :

Pour programmer une application, il ne convient pas de se lancer tête baissée dans l'écriture du code : il faut d'abord organiser ses idées, les documenter, puis organiser la réalisation en définissant les modules et étapes de la réalisation. C'est cette démarche antérieure à l'écriture que l'on appelle modélisation ; son produit est un modèle.

L'approche objet permet en principe à la maîtrise d'ouvrage de s'exprimer de façon précise selon un vocabulaire qui, tout en transcrivant les besoins du métier, pourra être immédiatement compris par les informaticiens.

Dans le but de formaliser l'analyse et la conception de notre système nous avons opté pour la méthodologie UML, un langage graphique de modélisation s'appuyant sur des normes très structurantes.

1. Définition d'UML (Unified Modeling Language) :

Le Langage de Modélisation Unifié, en anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet. [1].

UML est utilisé pour spécifier, visualiser, modifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet. UML offre un standard de modélisation, pour représenter l'architecture logicielle. Les différents éléments représentables sont :

- ❖ Activité d'un objet/logiciel.
- ❖ Acteurs.
- ❖ Processus.
- ❖ Schéma de base de données.
- ❖ Composants logiciels.
- ❖ Réutilisation de composants.

Grâce aux outils de modélisation UML, il est également possible de générer automatiquement tout ou partie du code d'une application logicielle, par exemple en langage Java, à partir des divers documents réalisés.

Dans l'étude conceptuelle de notre Application, nous avons utilisé cinq diagrammes :

- ❖ Diagramme de contexte.
- ❖ Diagramme de cas d'utilisation.
- ❖ Diagramme de séquence.
- ❖ Diagramme de classe.

Diagramme d'activités.

2. Les diagrammes UML :

UML dans sa version 2 propose treize diagrammes qui peuvent être utilisés dans la description d'un système. Ces diagrammes sont regroupés dans deux ensembles :

➤ Les diagrammes structurels :

Représentent l'aspect statique d'un système :

- ❖ Diagramme de classe.
- ❖ Diagramme d'objet.
- ❖ Diagramme de composant.
- ❖ Diagramme de déploiement.
- ❖ Diagramme de paquetage.
- ❖ Diagramme de structure composite.

➤ Les diagrammes de comportements :

Représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs :

- ❖ Diagramme de cas d'utilisation.
- ❖ Diagramme d'état transition.
- ❖ Diagramme d'activité.
- ❖ Diagramme de séquence.
- ❖ Diagramme de communication.
- ❖ Diagramme globale d'interaction.
- ❖ Diagramme de temps.

3. Description des cas d'utilisation :

Les cas d'utilisation sont des outils formels qui permettent de consigner et d'exprimer des interactions entre les utilisateurs et le système. On peut noter que les cas d'utilisations sont utilisés durant tout le processus car ils servent à la création de l'IHM, à la spécification des tests.

3.1. L'acteur : Un acteur est un type stéréotypé représentant une abstraction qui réside juste en dehors du système à modéliser. Un acteur représente un rôle joué par une personne ou une chose qui interagit avec le système.

Un acteur n'est pas nécessairement une personne physique: il peut être un service, une société, ou un système informatique...

On distingue quatre types d'acteurs :

3.1.1. Les acteurs principaux:

Les personnes qui utilisent les fonctions principales du système.

3.1.2. Les acteurs secondaires:

Les personnes qui effectuent des tâches administratives ou de maintenance.

3.1.3. Le matériel externe:

Les dispositifs matériels incontournables qui font partie du domaine de l'application et qui doivent être utilisés.

3.1.4. Les autres systèmes:

Les systèmes avec lesquels le système doit interagir.

3.2. Le cas d'utilisation :

Le cas d'utilisation (use case) correspond à un objectif du système, motivé par un besoin d'un ou plusieurs acteurs.

L'ensemble des use cases décrit les objectifs (le but) du système.

3.2.1. La relation :

Elle exprime l'interaction existante entre un acteur et un cas d'utilisation. Il existe trois types :

- La relation d'utilisation.
- La relation d'extension.
- La relation d'inclusion.

❖ **La relation de généralisation :**

Dans une relation de généralisation entre deux cas d'utilisations, le cas d'utilisation fils est une spécialisation du cas d'utilisation père.

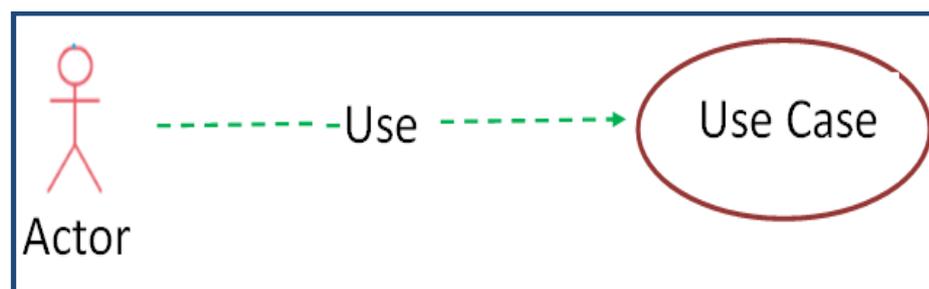


Figure 01 : Relation généralisation.

❖ La relation d'extension :

Indique que le cas d'utilisation source ajoute son comportement au cas d'utilisation destination. L'extension peut être soumise à condition. Le comportement ajouté est inséré au niveau d'un point d'extension défini dans le cas d'utilisation destination. Cette relation permet de modéliser les variantes de comportement d'un cas d'utilisation (selon les interactions des acteurs et l'environnement du système).

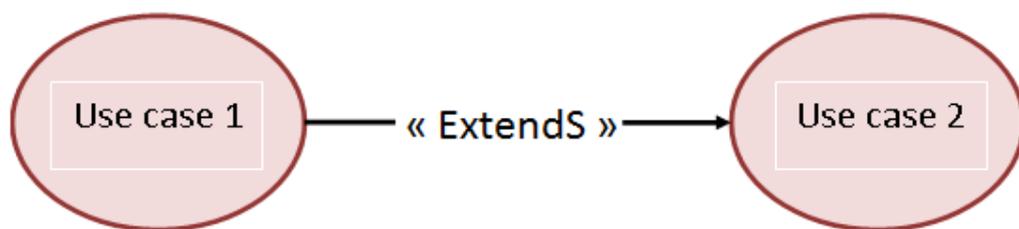


Figure 02 : *Relation d'extension.*

❖ La relation d'inclusion :

Indique que le cas d'utilisation source contient aussi le comportement décrit dans le cas d'utilisation destination. L'inclusion a un caractère obligatoire, la source spécifiant à quel endroit le cas d'utilisation cible doit être inclus. Cette relation permet ainsi de décomposer des comportements et de définir des comportements partageables entre plusieurs cas d'utilisations.

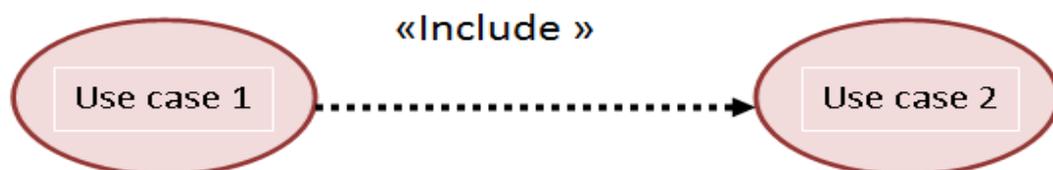


Figure 03 : *Relation d'inclusion.*

3.2. Le diagramme de classe :

Est un diagramme structurel qui exprime d'une manière générale la structure statique d'un système en termes de classes, d'interfaces et de collaborations, ainsi que leurs relations.

3.2.1. La classe :

Identifie une catégorie d'objet. Elle est représentée par un rectangle séparée en trois parties :

- La première partie contient le nom de la classe.
- La seconde contient les attributs de la classe.
- La dernière contient les méthodes de la classe.

La seconde et la dernière représentent le comportement de la classe.

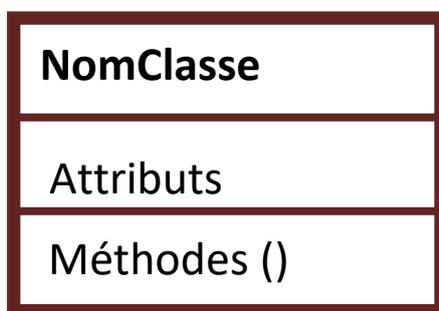


Figure 03 : représentation d'une classe UML.

3.2.2. L'association :

Est une connexion sémantique entre deux classes (relation logique). Une association peut être nommée. L'invocation d'une méthode est une association. Elle peut être binaire, dans ce cas elle est représentée par un simple trait, ou n-naire, les classes sont reliées à un losange par des traits simples. Ces relations peuvent être nommées.

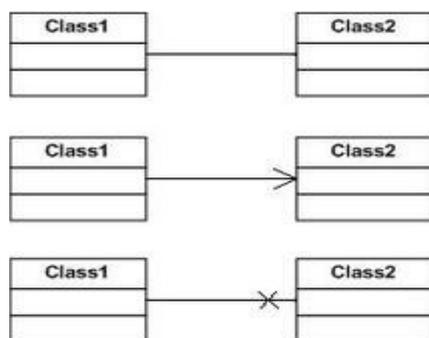


Figure 03 : représentation d'Association UML.

3.3. Le diagramme de séquence :

Montre les interactions entre objets selon un point de vue temporel. La représentation du contexte des objets se concentre sur l'expression des interactions.

Un objet est matérialisé par une barre verticale appelée ligne de vie des objets. Les objets, communiquent en échangeant des messages présentés au moyen des flèches orientés, de l'émetteur du message vers le destinataire. L'ordre des messages est donné par leur position sur l'axe vertical.

3.3.1. Message synchrone et message asynchrone :

Dans un diagramme de séquence, deux types de messages peuvent être distingués :

3.3.1.1. Message synchrone :

Dans ce cas, l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions. La flèche avec extrémité plane symbolise ce type de message. Le message retour peut ne pas être représenté car il est inclus dans la fin d'exécution de l'opération de l'objet destinataire du message.

3.3.1.2. Message asynchrone :

dans ce cas, l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations. C'est une flèche avec une extrémité non plane qui symbolise ce type de message.

3.4. Diagramme d'activité :

Dans une entreprise, les diagrammes d'activités aident les différents intervenants côté commercial et côté développement à collaborer pour comprendre un même procédé et un même comportement. Nous avons utilisé un ensemble de symboles spécialisés y compris ceux que l'on emploie pour commencer, terminer, fusionner ou recevoir des étapes dans le flux pour faire un diagramme d'activités .

3.4.1 Symboles d'un diagramme d'activité.



Symbole de début :

Représente le début d'un processus ou d'un flux de travail dans un diagramme d'activités. Il peut être utilisé seul ou avec un symbole de note qui explique le point de départ.



Symbole d'activité :

Indique les activités qui composent un processus modélisé. Ces symboles, qui comprennent de brèves descriptions dans la forme, sont les principales composantes d'un diagramme d'activités.



Symbole de raccord :

Indique le flux directionnel, ou flux de contrôle, de l'activité. Une flèche entrante marque le début d'une étape d'une activité ; une fois l'étape terminée, le flux se poursuit avec la flèche sortante.



Symbole de décision :

Représente une décision et possède toujours au moins deux embranchements avec le texte de la condition pour permettre aux utilisateurs de voir les options. Ce symbole représente la ramification ou la fusion de différents flux et sert de cadre ou de conteneur.

Références :

I. [1] : Définition de l'union européenne

I. [2]: CANARIE.S An E-learning Vision: Towards a Pan-Canadian Strategy and Action Plan, document de travail, Ottawa, 2002.

I. [3] « Avis sur la formation à distance », fédération étudiante universitaire du Québec, Année 2009.

I. [4] :H. BENMOHAMED ICTT @ Lab: « un environnement informatique pour la génération et l'exécution de scénarios de télé TP ». Thèse de doctorat en Informatique. Lyon : INSA de Lyon, 2007, 279 p.

I. [5] : <http://www.educnet.education.fr/superieur/glossaire.htm>.

I. [6] : <http://www.unow.fr/index.php/blog-unow/120-tout-savoir-sur-les-mooc> , « MOOC : minutes pour tous savoir » , article, 2015 , consulter le :12/04/2016.

I. [7] : AKOUCHE .R et KECHICHE.L Réalisation d'une application Java EE pour les systèmes E-Learning. Encadreur : KERBICHE.M. Université Mouloud Mammeri Tizi Ouzou.Mémoire de master2, promotion 2018-2019.

I. [8] : http://foad.inffolor.org/component/option,%20com_docman/task,doc_view/gid,177, « Qu'est-ce que l' E-Learning », Février 2010, consulter le :12/04/2016.

I. [9] : M.FAISA et E.IMANE : « Etude comparative des différentes plates-formes de la formation en ligne « E-Learning » » , Encadreur : TOUIL ,mémoire de master2, Juin 2015.

I. [10] : AMARA MAMMAR «concevoir une plateforme d'apprentissage basée sur l'évaluation formative »: cas d'application cours algorithmique. Mémoire de fin d'étude pour l'obtention du diplôme d'ingénieur d'état en informatique. Ecole nationale supérieure d'informatique (ESI).oued semar.

I. [11] :<https://fr.wikipedia.org/wiki/Moodle>.

I [12]: <http://sarremia.com/blog/index.php/ganesh-plate-forme-e-learning-scorm-2004-open-source/73>.

I [13] :<https://fr.wikipedia.org/wiki/Claroline>.

I [14] : http://www-igm.univ-mlv.fr/~dr/BOURGOGNON_C2011/creer_et_animer_un_blog_pedagogique/le_social_learning_en_action/lulu.com.

I [15] : [https://www.memoireonline.com/KUKULSKA-HULME A. & TRAXLER J/2005/mobile_learning_a_handbook_for_educators_and_trainers.html](https://www.memoireonline.com/KUKULSKA-HULME_A.&TRAXLER_J/2005/mobile_learning_a_handbook_for_educators_and_trainers.html).

I.[16] : AMEL KHELLAF & ABDELHAKIM NACEF :« Vers un e-Learning efficace pour la formation continue » état présent et perspectives. Colloque "Veille technologique et e-

novation pédagogique en formation continue". Université de la formation continue (UFC) Alger, le 18, 19 et 20 octobre 2014.

I.[17] : <http://algerienetwork.com/algerie/laprometteuse-et-remarquable-poussee-du-e-learning-en-algerie>. Algérie Network, Mai 2013.

I. [18] : BELBACHIR, F. (2016). Le e-Learning comme méthode d'apprentissage (Doctoral dissertation).

I.[19] : <https://www.algerie360.com/ooredoo-encourage-le-e-learning-a-travers-la-plateforme-imadrassa-com/>

I. [20] : <https://www.radioalgerie.dz/news/fr/node/101601>.

I. [21] : <https://www.radioalgerie.dz/news/fr/article/20200419/192524.html>

II. [1] : https://www.memoireonline.com/02/09/1973/m_conception-et-developpement-dune-application-de-la-gestion-dune-bibliotheque5.html.

II. [2]: https://fr.wikipedia.org/wiki/Diagramme_des_cas_d%27utilisation. Cas d'utilisation.

III. [1] : **IV. [1]** : AOUAM DJAMEL et CHABANE HOCINE : «Conception et réalisation d'une plateforme E-learning», mémoire fin d'études, promotion 2010-2011.

III. [2]: https://fr.wikipedia.org/wiki/Apache_Tomcat.

III. [3] : <https://fr.wikipedia.org/wiki/WampServer>.

III. [4] : http://www-igm.univ-mlv.fr/~dr/XPOSE2003/alexandrebole/jboss_1.html

III.[5]: https://doc.fedorafr.org/wiki/PhpMyAdmin:gestion_de_vos_bases_de_donn%C3%A9es_MySQL.

III. [6] : <https://www.techno-science.net/definition/517.html>.

III. [7] : [https://fr.wikipedia.org/wiki/Java_\(langage\)](https://fr.wikipedia.org/wiki/Java_(langage)) .

III. [8] : https://fr.wikipedia.org/wiki/Hypertext_Markup_Language .

III. [9] : <https://www.futura-sciences.com/tech/definitions/internet-css-4050/> .

III. [10] : <http://glossaire.infowebmaster.fr/javascript/> .

III. [11] : <http://dictionnaire.sensagent.leparisien.fr/Mod%C3%A8le%20relationnel/fr-fr/> .

III. [12] : https://fr.wikipedia.org/wiki/Mod%C3%A8le_physique_des_donn%C3%A9es.