

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

UNIVERSITE MOULOU D MAMMERI DE TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT INFORMATIQUE

Mémoire de fin d'études
De MASTER PROFESSIONNEL

Spécialité : Ingénierie des Systèmes d'Information.

Thème :

Réalisation d'une application Java EE
pour les systèmes E-Learning

Réalisé par :

AKOUCHE Rebiha.

KECHICHE Lamia.

Encadré par :

M^r: KERBICHE M'hand.

Promotion : 2018 - 2019.



Remerciement

*Ce présent travail est le fruit des longues années d'études. A cet effet nous tenons à remercier D'abord nos enseignants qui nous ont formés, vivement notre promoteur Mr **KERBICHE M'hand** qui nous a formés, 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.

*Nos vifs remerciement au directeur de L'**INSIM** de Tizi-Ouzou Mr **ABDELMALEK** qui nous a ouvert les portes, accueilli et a mis tout les moyens à notre disposition afin d'effectuer notre stage pratique.*

*Mme **ACHIR**, chef département d'informatique à l'**INSIM** qui nous a accueillis, orienté, conseillé et aidé.*

*Mr **ALIANE**, nous lui témoignons notre gratitude pour ses encouragements, ses orientations et sa précieuse aide qui nous ont été précieux pour mener à bien notre projet.*

Enfin, nous remercions tous ceux, de près ou de loin, contribué à la réalisation de ce modeste travail.

Rebiha & Lamia

Dédicaces

Je dédie ce modeste travail à :

Ma chère grand-mère qui m'a toujours soutenu et encouragé.

*Ma mère et mes sœurs Sonia, Maya, pour leur aide et encouragements
ainsi mon frère Adam.*

*Mes tantes, oncles, cousins et cousines pour m'avoir accompagné, en
particulier Mahmoud et Eliana.*

*Mon meilleur ami Amine qui m'a toujours soutenu et encouragé et ainsi
à sa famille.*

Tous mes amis pour leur compagnie.

*Ma binôme Lamia et sa famille pour leur accueil chaleureux et leurs
bonté.*

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

Rebiha

Dédicaces

Je dédie ce modeste travail à :

Mes chers parents qui m'ont toujours soutenu et encouragé.

*Mes sœurs Naima, Nacima, Karima, Lila et Yasmine pour leurs
aide, conseils et encouragements.*

Mon futur mari et ma belle famille pour m'avoir accompagné.

Tous mes amis pour leur compagnie.

Ma binôme Rebiha et sa famille.

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

Lamia

Table des matières

Liste des figures.....	9
Liste des tableaux	11
Introduction générale.....	12
Chapitre I	E-Learning
Introduction:.....	14
I.1. Définitions.....	16
I.1.1. E-learning.....	16
I.1.2. Technologies de l'information et des communications (TIC).....	16
I.1.3. La formation à distance (FAD).....	15
I.1.4. La formation ouverte et à distance (FOAD).....	16
I.1.5. Massive Open Online Course (MOOC).....	16
I.1.6. Le tracking.....	17
I.2. Principes du E-Learning.....	18
I.2.1. Pourquoi le E-learning.....	19
I. 3. Les profils dans les systèmes e-learning.....	20
I.3.1. L'apprenant (l'étudiant).....	20
I.3.2 L'enseignant.....	20
I.3.2.1. Auteur (concepteur) de cours.....	20
I.3.2.2. Tuteur.....	20
I.3.2.3. Evaluator.....	20
I.3.3. L'administrateur.....	20
I.4. Types du E-Learning.....	20
I.4.1. Un cours magistral synchrone	21
I.4.2. L'assistance	21
I.4.3. L'auto-formation connectée.....	21
I.4.4. Blended learning (formation mixte).....	21
I.4.5. Le social Learning (l'apprentissage social).....	22
I.5. Techniques de communication dans E-Learning.....	22
I.5.1. Synchrone.....	22
I.5.2. Asynchrone	22
I.5.3. Mixte.....	23
I.6. Critères de qualité du E-learning.....	23
I.6.1. Contenu centré sur l'apprenant	23
I.6.2. Contenu stimulant.....	23
I.6.3. Interactivité.....	23
I.6.4. Personnalisation	23
I.7. Les systèmes de gestion de cours.....	24
I.7.1. Learning Management System (LMS).....	24
I.7.2. Learning Content Management System (LCMS).....	24
I.7.3. Virtual Classroom System (VCS).....	24
I.7.4. Intelligent Tutoring Systems (ITS).....	24
I.7.5. Le social learning, Un nouvel air du E-Learning.....	25
I.8. Quelques plateformes E-Learning.....	25
I.8.1. Claroline.....	25
I.8.2. Ganesha.....	25
I.8.3. Moodle (Modular Object-Oriented Dynamic Learning Environment).....	26
I.9. Les outils de communication	27
I.9.1. La messagerie	27
I.9.2. Le forum	27

I.9.3. Le Chat	27
I.9.5. Les documents pédagogiques partagés.....	27
I.9.6. Les news	27
I.9.6. Les sondages	27
I.9.7. Le Bloc-notes	28
I.9.8. Les Glossaires	28
I.9.9. Les foires aux questions(FAQ)	28
I.9.10. Outils complémentaires	28
I.10. E-learning en Algérie.....	29
I.11. Les enjeux du E-Learning.....	29
I.12. Avantages et inconvénients	29
I.12.1. Coté Apprenant.....	30
I.12.2. coté Formateur.....	30
I.12.3. Coté Entreprise (Ecole).....	31
Conclusion :	32

Chapitre II

Organisme D'accueil

Introduction	34
II.1. Présentation de l'organisme d'accueil	34
II.1.1. Historique de l'INSIM	34
II.1.2. Situation géographique	35
II.1.2.1. Adresse de l'institut	35
II.1.2.2. Emplacement de l'institut INSIM par satellite	35
II.1.3. Activité principale et mission	36
II.1.3.1. Filiales du groupe INSIM	36
II.1.3.2. Partenaires du groupe INSIM	37
II.1.3.3. Cadre de travail	39
II.1.4. L'organigramme de l'INSIM	40
II.1.5. Description des services	41
II.1.5.1. Direction générale	41
II.1.5.1.1. Secrétariat de direction	41
II.1.5.1.2. Département des relations extérieures «RELEX»	41
II.1.5.1.3. Département administratif et financier	41
II.1.5.1.4. Chargé des moyens généraux	42
II.1.5.1.5. Service scolarité / admission	42
II.1.5.1.6. Département chargé des études	42
II.1.5.1.7. Chargé des formations principales	43
II.1.5.1.8. Chargé des stages	43
II.1.6. Aspect matériel et logiciel	44
II.1.6.1. Aspect matériel	44
II.1.6.2. Aspect logiciel	46
II.1.7. Objectif de l'étude	47
Conclusion :	48

Chapitre III:

Analyse et conception

Introduction:.....	50
III.1. Problématique.....	50
III.2. Objectif	50
III.3. Analyse des besoins	51
III.3.1. Sujet du projet	51
III.3.2. Acteurs et cas d'utilisation	51

III.3.2.1. Identification des acteurs	51
III.3.2.2. Cas d'utilisations	52
III.3.3. Spécifications des besoins	53
III.3.3.1. Les besoins fonctionnels	53
III.3.3.2. Les besoins non-fonctionnels	54
III.3.4. Diagramme de contexte	54
III.4. Conception de l'application	55
III.4.1. Diagramme de cas d'utilisation	55
III.4.2. Diagramme de Séquences	56
III.4.2.1. Authentification	56
III.4.2.2. Inscription utilisateur	57
III.4.2.3. Ajouter Contenu (Cours/TD/ TP)	58
III.4.2.4. Préparation examen (Création QCM)	59
III.4.2.5. Passer examen	60
III.4.2.6. Envoie Message	61
III.4.3. Diagramme d'activité	62
III.4.3.2. Gestion des cours	63
III.4.3.3. Consultation messagerie	64
III.4.4. Diagramme de classes	65
III.4.4.1. Description des classes	66
Conclusion :	68

Chapitre IV

Réalisation

Introduction	72
IV.1. Outils de développement et Langages de programmation	72
IV.1.1. Outils de développement	72
IV.1.1.1. Caractéristiques de l'ordinateur	72
IV.1.1.2. Le serveur	73
IV.1.1.2.1. Apache Tomcat	73
IV.1.1.2.2. WampServer	75
IV.1.1.2.3. Jboss	76
IV.1.1.2.4. GlassFish	77
IV.1.1.2.5. WebLogic	78
IV.1.1.3. PhpMyAdmin	79
IV.1.1.4. Eclipse IDE (Integrated Development Environnement)	80
IV.1.2.1. JAVA	81
IV.1.2.2. HTML 5	81
IV.1.2.3. CSS3 et CSS4.....	81
IV.1.2.4. JavaScript	82
IV.1.2.5. JQuery	82
IV.2. Modèle Relationnel	83
IV.2.1. Les règles pour le système relationnel de base des données	83
IV.2.2. Le modèle relationnel du projet	85
IV.2.3. Le modèle physique de données	86
IV.2.3.1. Table «Utilisateur».....	86
IV.2.3.2. Table «Authentification».....	87
IV.2.3.3. Table «Etudiant».....	87
IV.2.3.4. Table «Enseignant».....	87
IV.2.3.5. Table «Administrateur».....	87
IV.2.3.6. Table «Livre»	88
IV.2.3.7. Table «Message»	88
IV.2.3.8. Table «Formation»	88
IV.2.3.9. Table «Module»	89

IV.2.3.10. Table «Examen»	89
IV.2.3.11. Table «Contenu»	89
IV.2.3.12. Table «Question»	89
IV.2.3.13. Table «Choix»	90
IV.3. Présentation de quelques interfaces	91
IV.3.1. Interface page d'accueil.....	91
IV.3.2. Interface d'authentification de l'utilisateur.....	91
IV.3.3. Interface d'ajout enseignant.....	92
IV.3.4. Interface d'ajout contenu.....	92
IV.3.5. Interface de création du QCM.....	93
IV.3.6. Interface page d'accueil de l'administrateur.....	93
Conclusion	94
Conclusion générale.....	96
<i>Annexe I</i>	JavaEE
<i>Annexe II</i>	UML
Références	117

LISTE DES FIGURES

Chapitre I :

Figure I.1 : *Topologie de la formation à distance (FAD).*

Figure I.2 : *Principe du E-Learning.*

Figure I.3 : *Technique de communication E-Learning.*

Chapitre II :

Figure II.1 : *Adresse de «l'INSIM».*

Figure II.2 : *Emplacement de l'institut «INSIM» par satellite.*

Figure II.2 : *Emplacement de l'institut «INSIM».*

Figure II.3 : *Organigramme de l'institut «INSIM».*

Chapitre III :

Figure III.1 : *Diagramme de contexte.*

Figure III.2 : *Diagramme de cas d'utilisation général.*

Figure III.3 : *Diagramme de séquence «Authentification».*

Figure III.4 : *Diagramme de séquence «Inscription Personne».*

Figure III.5 : *Diagramme de séquence «Ajouter Contenu».*

Figure III.6 : *Diagramme de séquence «préparation examen (création QCM)».*

Figure III.7 : *Diagramme de séquence «consulter contenu (téléchargement)».*

Figure III.8 : *Diagramme de séquence de «Passer examen».*

Figure III.9 : *Diagramme de séquence «Envoi Message».*

Figure III.10 : *Diagramme d'activité «Gestion utilisateur».*

Figure III.11 : *Diagramme d'activité «Gestion des cours».*

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

Figure III.12 : *Diagramme de classes général.*

Chapitre IV :

Figure IV.1 : *Caractéristiques de l'ordinateur.*

Figure IV.2 : *Page d'accueil serveur «Apache Tomcat 9.0.16».*

Figure IV.3 : *Page d'accueil «WampServer».*

Figure IV.4 : *Page d'accueil «Jboss».*

Figure IV.5 : *Page d'accueil «GlassFish».*

Figure IV.6 : *Page d'accueil «WebLogic».*

Figure IV.7 : *Page d'accueil de «PhpMyAdmin».*

Figure IV.8 : *page d'accueil «Eclipse IDE».*

Figure IV.9 : *page d'accueil.*

Figure IV.10 : *Page «Authentification utilisateur».*

Figure IV.11 : *Page «Ajouter Enseignant».*

Figure IV.12 : *Page «Ajouter Contenu».*

Figure IV.13 : *Page «Créer QCM».*

Figure IV.14 : *Page «Accueil administrateur».*

LISTE DES TABLEAUX

Chapitre I :

Tableau I.1 : *Les avantages et inconvénients du coté de l'apprenant.*

Tableau I.2 : *Les avantages et inconvénients du coté du formateur.*

Tableau I.3 : *Les avantages et inconvénients du coté de l'entreprise (Université, école...).*

Chapitre II :

Tableau II.1 : *Filiales du groupe «INSIM».*

Tableau II.2 : *Partenaires du groupe INSIM.*

Tableau II.3 : *Aspect Logiciel.*

Chapitre III :

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

Tableau III.2: *Les besoins fonctionnels.*

Chapitre IV :

Tableau IV.1 : *Table «Personne».*

Tableau IV.2 : *Table «Authentification».*

Tableau IV.3 : *Table «Etudiant».*

Tableau IV.4 : *Table «Enseignant».*

Tableau IV.5 : *Table «Administrateur».*

Tableau IV.6: *Table«Livre».*

Tableau IV.7 : *Table «Message».*

Tableau IV.8 : *Table «Formation».*

Tableau IV.9 : *Table «Module».*

Tableau IV.10 : *Table «Examen».*

Tableau IV.11 : *Table «Contenu».*

Tableau IV.12 : *Table «Question».*

Tableau IV.13 : *Table «Choix».*

Tableau IV.14 : *Table «Réponse».*

Introduction générale

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 de l'institut de formation professionnelle INSIM tout en mettant à la disposition des formateurs des outils leurs permettant d'assurer les différents rôles.

Globalement, notre mémoire se compose de quatre chapitres :

Chapitre 1 : E-learning.

Chapitre 2 : Présentation de l'organisme d'accueil.

Chapitre 3 : Analyse et conception.

Chapitre 4 : Réalisation de l'application.

Chapitre I

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.

I.1. Définitions:

I.1.1. E-learning :

E-learning, ou également «apprentissage électronique», anciennement nommé «apprentissage assistée par ordinateur». Désigne le processus par lequel un apprenant apprend en utilisant les fonctionnalités d'un réseau électronique (web, internet ou intranet), en utilisant des supports numériques. [1].

Le E-Learning est un mode d'apprentissage avec l'usage des technologies de l'information et de la communication à tous les niveaux de l'activité de formation. Il s'agit plus particulièrement d'un dispositif de formation dont les principaux objectifs peuvent être définis comme l'autonomie d'apprentissage, la formation à distance, l'individualisation des parcours de formation et le développement des relations pédagogiques en ligne. [2].

L'apprenant peut ainsi facilement accéder à un large panel de connaissances avec seulement une connexion Internet.

I.1.2. Technologies de l'information et des communications (TIC) :

Les technologies de l'information et des communications (TIC) regroupent une panoplie de moyens de diffusion analogiques, numériques ou informatiques qui, jumelés au mode traditionnel (papier, crayon), permettent d'offrir un enseignement souple et diversifié aux étudiants. Dans leur essence, le rôle des TIC est de soutenir les individus dans leur tâche manuelle ou intellectuelle, car elles augmentent la rapidité de communication entre eux, offrant un accès instantané à l'information et diversifiant les formats de distribution de contenu pédagogique. [3].

I.1.3. 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].

FAD est incluse dans le concept plus général de «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.

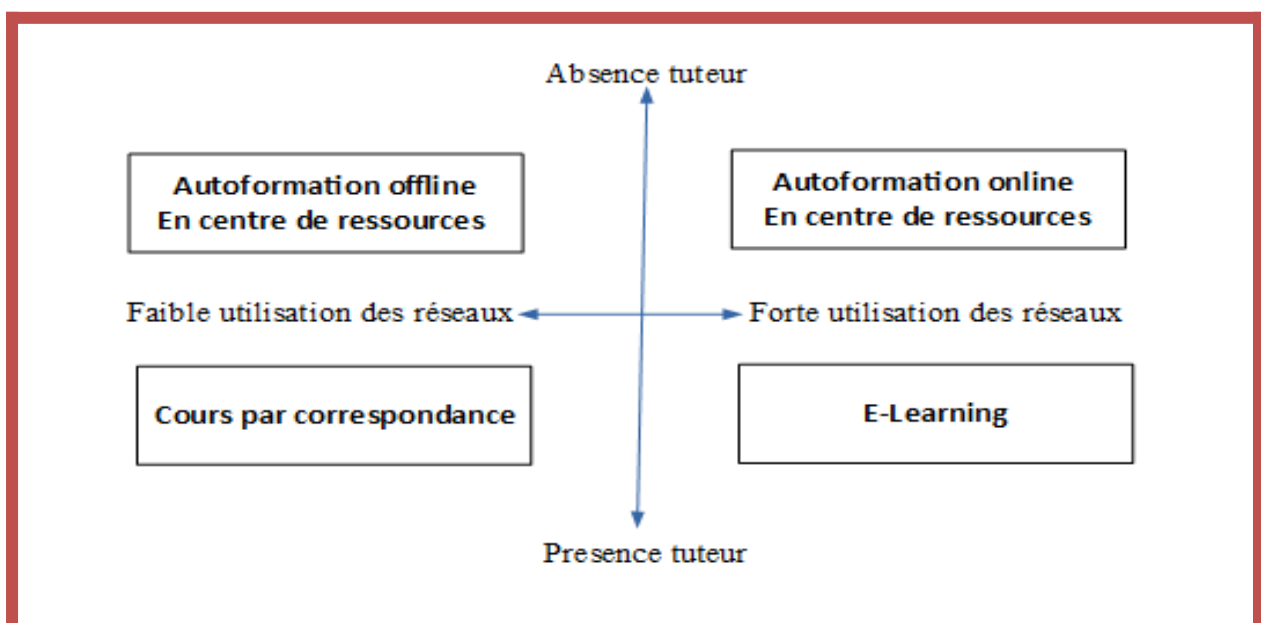


Figure I.1 : Topologie de la formation à distance (FAD).

I.1.4. La formation ouverte et à distance (FOAD):

Dispositif de formation permettant à la fois d'offrir les avantages de la souplesse d'utilisation pour tout apprenant ayant la possibilité d'avoir accès à des espaces d'apprentissage ouverts (c'est l'apprenant qui a la possibilité de choisir parmi un ensemble de propositions: le moment de son apprentissage, le module (thème, niveau...) et l'approche cours, exercice, démonstration, découverte, illustration, témoignage, débat...) et permettant de réaliser des apprentissage à distance: c'est-à-dire à distance physique des enseignants et des autres apprenants. [1].

Pour une même formation un ensemble d'apprenant peuvent donc réaliser des apprentissages sur des thèmes différents, à des moments différents et dans des lieux différents).

❖ La différence entre la FAD et la FOAD :

La notion de formation ouverte et de formation à distance peut être différenciée, ainsi :

- S'il y a rupture de l'unité de lieu, il s'agit de formation à distance.
- S'il y a rupture de l'unité de temps et d'action, même dans un même lieu commun, il s'agit de formation ouverte.
- S'il y a rupture des deux, il s'agit de formation ouverte et à distance.

I.1.5. Massive Open Online Course (MOOC):

Un MOOC ou Formation en Ligne Ouverte à Tous (FLOT), aussi appelée Cours En Ligne Ouvert et Massif (CLOM), est un type ouvert de formation à distance capable d'accueillir un grand nombre de participants. L'appellation MOOC est passée dans le langage courant en France; elle est désormais reconnue par les principaux dictionnaires.

Les participants aux cours, enseignants et élèves, sont dispersés géographiquement et communiquent uniquement par Internet. Des ressources éducatives libres sont souvent utilisées. Dans le monde anglophone, il peut arriver que plus de 100000 personnes soient réunies pour un cours. [4].

I.1.6. Le tracking :

Le tracking Internet consiste à observer et analyser les comportements d'un utilisateur sur internet. [5].

Les fonctions de tracking en le e-learning permettent le suivi pédagogique des apprenants. Un grand nombre d'information concernant le parcours individuel de chaque stagiaire peut être stocké dans le système. Il s'agit couramment des résultats obtenus dans les modules d'évaluation et de test, le temps passé sur un module de cours, le temps passé à réaliser un exercice, etc.

Ces données sont ensuite interprétées par le tuteur, qui peut ainsi se rendre compte de l'assiduité des stagiaires, des problèmes rencontrés sur certains chapitres ou sur la résolution des problèmes.

Pour que les fonctions de tracking soient pleinement opérationnelles le système complet - Système de Gestion de Contenu (SGC et contenus) devra répondre à un standard normalisé.

1.2. Principes du E-learning :

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 cœur 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.

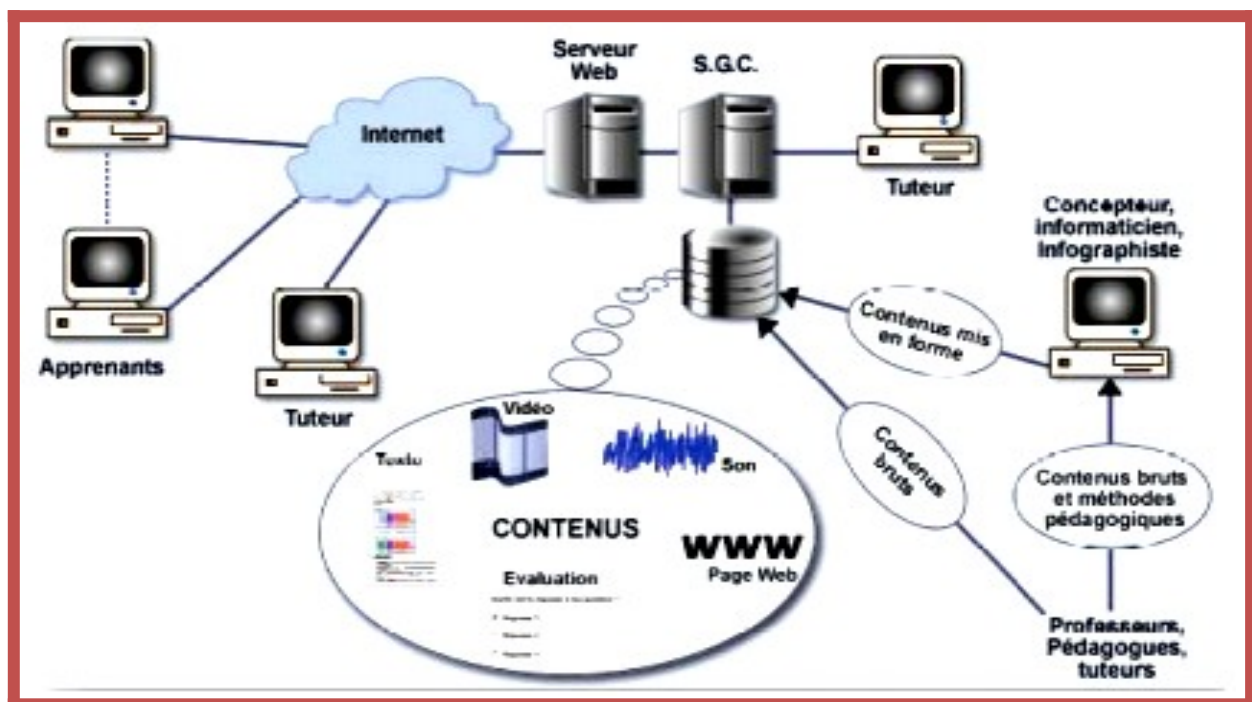


Figure I.2 : Principe du E-Learning.

I.2.1. Pourquoi le E-learning ?

Le E-learning est une bonne solution lorsque :

- Il existe une quantité significative de contenu à délivrer à un grand nombre d'apprenants.
- Les apprenants proviennent d'endroits géographiquement dispersés.
- Les apprenants ont une mobilité réduite.
- Le temps que les apprenants peuvent consacrer chaque jour à l'apprentissage est limité.
- Les apprenants ne possèdent pas de compétences de communication suffisantes.
- Il est nécessaire de développer des connaissances de base homogènes sur le sujet.
- Les apprenants sont fortement motivés et apprécient d'apprendre à leur propre rythme.
- Le contenu doit être réutilisé pour d'autres groupes d'apprenants dans le futur.
- Le cours répond à des besoins de formation à long terme plutôt qu'à court terme.
- Il est nécessaire de recueillir et d'assurer le suivi des données.

L'apprentissage numérique n'est pas idéal dans toutes les situations, il est ainsi peu probable qu'il remplacera complètement la formation en classe au sein d'une organisation. Le e-learning peut représenter une solution plus économique pour compléter la formation conventionnelle, et atteindre le plus d'apprenants possibles.

I.3. Les profils dans les systèmes e-learning :

Un profil utilisateur est une représentation explicite de ses caractéristiques. Les principaux profils (acteurs) d'inscrits sur une plateforme d'apprentissage en ligne (E-Learning) sont les suivants :

I.3.1. 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.

I.3.2 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 :

I.3.2.1. Auteur (concepteur) de cours :

Son rôle est la création du contenu.

I.3.2.2. 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.

I.3.2.3. 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.

I.3.3. 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.

I.4. Types du E-Learning:

On distingue plusieurs types d'E-learning :

I.4.1. Un cours magistral synchrone :

Un formateur est relié en mode audio-visuel à plusieurs apprenants (en moyenne quatre à huit) situés dans des lieux géographiques qui peuvent être très différents. L'avantage de l'E-learning est alors double : offre la possibilité de se former depuis leur domicile. C'est confortable et permet de gagner du temps sans surplus des coûts liés au déplacement des apprenants.

I.4.2. L'assistance :

L'apprenant travaille individuellement, mais dispose d'une assistance audio-visuelle en direct. Le formateur reste à sa disposition et peut, à distance, prendre le contrôle de son poste de travail. Il peut ainsi partager les logiciels ou les documents et, le cas échéant, converser avec lui.

L'assistance peut également se manifester de manière asynchrone, au moyen de messageries ou de forums informatiques.

I.4.3. L'auto-formation connectée:

Les apprenants en formation peuvent, à partir des ressources pédagogiques ou des parcours de formation disponibles sur le serveur Internet ou Intranet, suivre un cours magistral enregistré, en audio et en vidéo, sous forme par exemple, de séances de rattrapage.

I.4.4. Blended learning (formation mixte):

Le mot « Blended learning » vient de l'anglais « blend » qui signifie mélange, et « learning » signifie apprentissage.

S'agit d'un dispositif alliant différents modes de formation afin d'offrir à l'apprenant le moyen de formation le plus complet du marché... Le Blended learning permet donc d'associer par exemple des cours d'anglais par téléphone, des cours d'anglais en ligne, des mails leçons, du mobile learning, des cours d'anglais particuliers en face-à-face et encore des cours d'anglais collectifs.

I.4.5. Le social Learning (l'apprentissage social) :

Le social Learning 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.

I.5. Techniques de communication dans E-Learning:

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

I.5.1. Synchrones :

Les événements synchrones se déroulent en temps réel. Pour qu'une communication entre deux personnes soit synchrone, elles doivent être toutes les deux présentes à un moment donné. Les chats et les conférences audio/vidéo sont des exemples d'activités synchrones.

I.5.2. Asynchrone :

Les événements asynchrones sont indépendants des questions de temps. Un cours en auto-apprentissage est un exemple d'apprentissage en ligne asynchrone puisque l'apprentissage peut se dérouler à n'importe quel moment. Les forums de discussion ou les courriels sont des exemples d'outils de communication asynchrones.

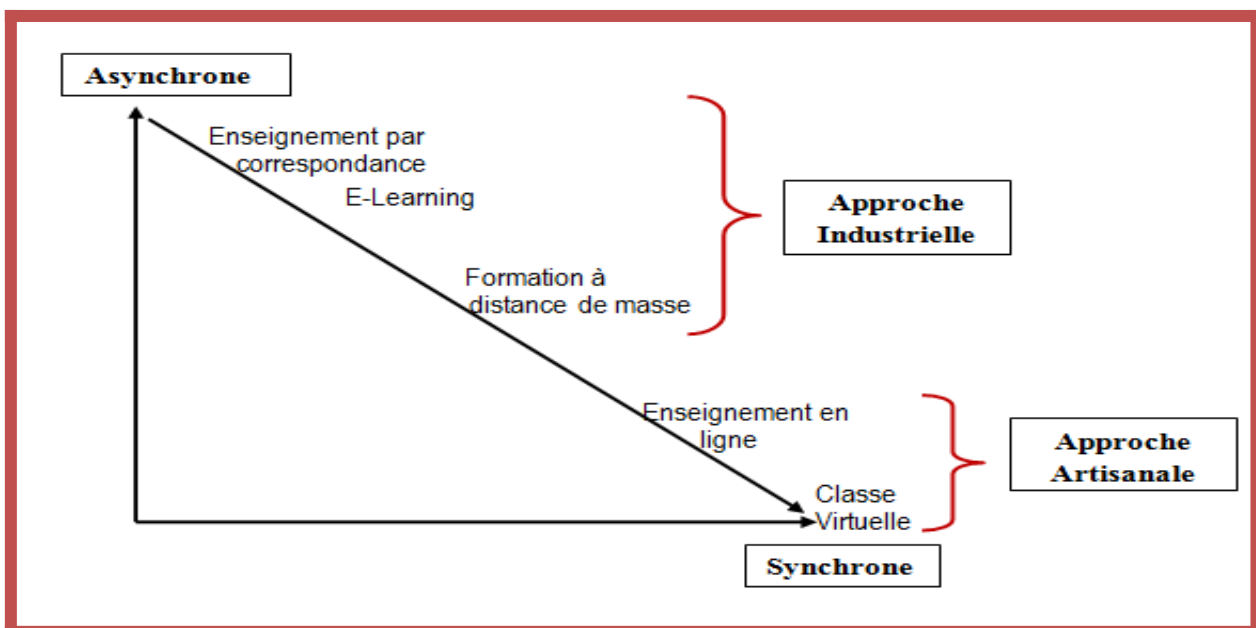


Figure I.3 : Technique de communication E-Learning.

I.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.

I.6. Critères de qualité du E-learning :

La qualité d'un cours e-learning réside dans les éléments qui suivent :

I.6.1. Contenu centré sur l'apprenant :

Le curriculum du cours numérique doit être précis et adapté aux besoins de l'apprenant ainsi qu'à ses fonctions et responsabilités professionnelles. Les compétences, les connaissances et les informations contenues dans le cours doivent viser cet objectif.

I.6.2. Contenu stimulant :

Les techniques et les méthodes d'enseignement doivent être utilisées de manière créative afin d'élaborer un cours stimulant et motivant pour l'apprenant.

I.6.3. Interactivité :

De fréquentes interactions avec l'apprenant sont nécessaires pour maintenir l'attention et encourager l'apprentissage.

I.6.4. Personnalisation :

Les cours en auto-apprentissage doivent être personnalisables afin de refléter les besoins et les intérêts des apprenants ; dans les cours dirigés, le formateur, le tuteur et/ou le facilitateur doivent être capables de suivre les progrès et les performances des apprenants de manière individuelle.

I.7. Les systèmes de gestion de cours :

Les systèmes de gestion de cours appartiennent aux catégories suivantes :

I.7.1. Learning Management System (LMS):

LMS est une famille de systèmes, qui permet la gestion de toutes les activités de la formation. Les systèmes LMS catégorisent les utilisateurs, leur donnent certaines autorisations à des modules de formation et affectent les utilisateurs à des groupes spécifiques de formation.

Exemples : MOODLE, Dokeos et Sakai.

I.7.2. Learning Content Management System (LCMS) :

LCMS est une famille de systèmes d'E-Learning les plus avancés technologiquement. Les systèmes LCMS intègrent des fonctionnalités des LMS et des CMS, assurant à la fois la création et la gestion des contenus éducatifs. Ils offrent la possibilité d'évaluer les connaissances assimilées par les utilisateurs.

Exemples: TopClass, SimplyLearn.

I.7.3. Virtual Classroom System (VCS):

Les VCS sont des systèmes qui incluent les fonctionnalités suivantes:

- Transmission de la voix et de la vidéo en temps réel entre tous les participants.
- Tableau blanc partagé (shared whiteboard).
- Espace intégré pour la projection de diapositives ou autres supports visuels.
- Capacité d'interaction textuelle, y compris les conversations ou « note-passing».
- Des moyens permettant aux apprenants d'indiquer qu'ils ont des questions.
- Outils pour évaluer l'humeur actuelle, les opinions et la compréhension ainsi que pour solliciter des questions ou des commentaires.

I.7.4. Intelligent Tutoring Systems (ITS):

Les ITS sont similaires aux LMS. Ils peuvent donner des feedback intelligents à l'utilisateur. Les ITS emploient les techniques de l'intelligence artificielle, pour pouvoir comprendre, informer et diriger l'apprenant quand il termine ses exercices ou ses tests. Ils visent à répliquer le rôle du tuteur qui guide et forme les apprenants d'une manière efficace.

Le tuteur humain est souvent remplacé par des entités intelligentes, appelées «agents» capables de suivre et guider l'apprenant durant son apprentissage.

I.7.5. Le social learning, Un nouvel air du E-Learning :

Bien que le terme social Learning fût utilisé bien avant l'apparition du e-Learning, le terme social learning représente un nouveau phénomène du Web 2.04. Actuellement, l'expression de «social learning» s'est répandue sur le web comme un concept qui caractérise l'utilisation des médias électroniques synchrones ou asynchrones pour le développement des savoirs, par le biais de connexion avec des collègues, des mentors ou des experts dans une optique collaborative.

I.8. Quelques plateformes E-Learning :

I.8.1. Claroline :

C'est une plate-forme d'apprentissage en ligne (ou LMS) et de travail collaboratif open source (sous licence GPL). Elle permet de créer et d'administrer des formations et des espaces de collaboration en ligne. [6].

Fonctionnalités de la plateforme :

- Gestionnaire de cours.
- Agendas.
- Annonces.
- Gestion des documents et des liens.
- Création d'exercices (QCM, Correspondance, texte à trous).
- Tracking, Gestion des parcours pédagogiques.
- Gère les contenus SCORM*.
- Gestion des groupes (partage de fichiers, forum).
- Personnalisation de la plateforme (HomePage, style CSS).

I.8.2. Ganesha :

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. [7].

Fonctionnalités de la plate-forme :

- Forum.
- Messagerie intégrée.
- Chat.
- Zone de dépôt de documents.
- Gère les contenus standardisés SCORM.
- Tracking et évaluation.
- Suivi pédagogique.
- Administration Web.
- Multilingue.

I.8.3. Moodle (Modular Object-Oriented Dynamic Learning Environment):

Moodle est une plateforme d'apprentissage en ligne LMS libre distribuée sous la Licence publique générale GNU écrite en PHP. Développée à partir de principes pédagogiques, elle permet de créer des communautés s'instruisant autour de contenus et d'activités. [8].

Fonctionnalités de la plateforme :

- Zone de dépôt de documents.
- Multilingue.
- Administration Web.
- Suivi pédagogique.
- Tracking et évaluation.
- Messagerie.
- Générateur HTML Wysiwyg.
- Gère les contenus standardisés SCORM.
- Sondage.
- Forum.
- Chat.
- Modules additionnels (Plugins).

I.9. Les outils de communication :

Généralement les plates-formes SGC intègrent un certain nombre d'outils de communication, la liste des outils disponibles dépend de la richesse du SGC. Ces outils sont souvent très usités sur le Web et ne sont en rien spécifique au e-learning, ils font simplement partie intégrante du SGC et ne nécessitent pas de configuration spécifique sur les postes des apprenants (par exemple, il n'est pas nécessaire de configurer un client messagerie pour profiter de ce service). On peut citer :

I.9.1. La messagerie :

C'est un espace de communication entre les membres du groupe (stagiaires et tuteurs). Ce système permet l'envoi et la réception de mails avec ou sans fichier attaché. Il peut être interne à la plate-forme et ne nécessite pas d'avoir un e-mail personnel.

I.9.2. Le forum :

Le forum, pouvant être public ou réservé à un groupe, permet aux stagiaires et aux tuteurs : de poster des messages qui seront accessibles à l'ensemble des membres du groupe de formation, de répondre aux messages déjà postés et ainsi engager une discussion sur un sujet donné.

I.9.3. Le Chat :

Il permet à l'ensemble des membres du groupe de discuter en temps réel.

I.9.4. Les documents pédagogiques partagés :

Une zone commune peut être utilisée pour mettre à disposition des étudiants différents documents sous forme numérique (Word, Excel, Pdf, etc.) pour l'ensemble du groupe.

I.9.5. Les news :

La possibilité de créer des news permettra, par exemple, d'informer ses apprenants de nouveaux rendez-vous, ou de leur communiquer diverses informations.

I.9.6. Les sondages :

Le sondage permettra de poser des questions ouvertes ou de proposer des choix de réponses. Puis, les réponses obtenues sur un cours en particulier pourront être analysées.

I.9.7. Le Bloc-notes :

C'est un espace privé à chaque utilisateur. Il peut y noter toutes les informations qu'il souhaite: ce qu'il a retenu de sa formation, des actions à mener...etc. Ces notes sont archivées par date et il est possible de les compiler dans une même note.

I.9.8. Les Glossaires :

Des glossaires peuvent être créés, destinés à l'ensemble des utilisateurs ou un groupe en particulier. Les différents termes avec leur signification sont enregistrés, un moteur de recherche permet aux utilisateurs de faciliter la recherche d'un terme dans les glossaires.

I.9.9. Les foires aux questions(FAQ) :

Cet espace permet aux tuteurs et aux administrateurs d'industrialiser la réponse à des questions récurrentes.

I.9.10. Outils complémentaires :

Certains SGC intègrent un outil dédié à la conception de contenus, permettant notamment la création aisée de modules de tests et d'évaluations de différentes formes. Ils ne permettent pas de développer des modules de manière aussi souple et riche qu'un véritable logiciel auteur pourrait le faire, mais constituent une réponse intéressante pour la mise en place de contenus simples, à la portée de n'importe quel tuteur, sans nécessiter aucune connaissance informatique.

I.10. E-learning en Algérie:

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

Mahieddine Djoudi estime que : « la plate-forme e-Learning permet de répondre à plusieurs problématiques liées notamment, à la croissance du nombre des étudiants et à l'insuffisance de l'encadrement pédagogique. La solution E-Learning vient mettre trois composantes en réseau à savoir l'enseignant/chercheur, l'étudiant et le monde extérieur. » Avec le développement des technologies l'E-Learning se renforce de plus en plus en Algérie grâce à plusieurs plateformes qui poussent. Citons par exemple la plateforme expérimentale appelée AVUnet (Université virtuelle algérienne) et le portail des TIC, qui fournissent aux apprenants des cours structurés et des interfaces pour y accéder et communiquer entre eux ou avec un enseignant, l'utilisation des technologies dans l'action pédagogique conduit à améliorer la qualité de l'enseignement-apprentissage et à accroître son efficacité.

I.11. Les enjeux du E-Learning :

Les enjeux du e-learning sont notamment de :

- Rendre plus efficaces, plus solides, plus adaptés les processus d'apprentissage et l'accès à la connaissance.
- Bénéficier des avantages des technologies éducatives (interactivité, simulation...).
- Bénéficier des avantages de la formation à distance (plus grande autonomie, élimination de contraintes...) tout en éliminant les inconvénients de déshumanisation.

I.12. Avantages et inconvénients :

Les tableaux ci-dessous tendent à citer les avantages et les inconvénients attribués au E-Learning, du point de vue des 3 acteurs de la formation E-Learning : l'apprenant, le formateur et l'entreprise.

I.12.1. Coté Apprenant :

Avantages	Inconvénients
<ul style="list-style-type: none"> ➤ Apprenant acteur de sa formation E-learning ➤ Interactivité et attractivité du contenu E-learning ➤ Flexibilité et adaptabilité selon ses disponibilités (heurs, lieu) ➤ Formation à son propre rythme indépendamment des autres apprenants ➤ Auto-évaluation en cours et en fin de cursus ➤ Suivi personnel de son avancement dans la formation e-learning et bilan des résultats obtenus grâce au tracking 	<ul style="list-style-type: none"> ➤ Appréhension de l'outil informatique, réticence face aux nouvelles technologies ➤ Gestion de son autonomie car pas de cadre de travail imposé ➤ Gestion de sa motivation et de son implication dans sa formation e-learning ➤ effort de concentration ➤ Pas de contacts directs avec le formateur (sauf en cas de formation blended-learning)

Tableau I.1: *Les avantages et inconvénients du coté de l'apprenant.*

I.12.2. Coté Formateur :

Avantages	Inconvénients
<ul style="list-style-type: none"> ➤ Pré-requis pour évaluer le niveau des apprenants ➤ Formation sur des outils interactifs et attractifs pour lui et les apprenants ➤ Flexibilité et adaptabilité selon les disponibilités (heurs, lieu) ➤ Suivi des apprenants depuis la plateforme (tracking) 	<ul style="list-style-type: none"> ➤ Pas de contacts directs avec l'apprenant (sauf en cas de formation blended learning) ➤ Transformation du rôle de « sachant » en rôle de « chef d'orchestre » ➤ Passage d'une communication majoritairement orale à écrite

Tableau I.2: *Les avantages et inconvénients du coté du formateur.*

I.12.3. Coté Entreprise (École) :

Avantages	Inconvénients
<ul style="list-style-type: none"> ➤ Formation « de masse » (nombres d'apprenant illimité) ➤ Economie sur les couts indirects liés aux formations en présentiel (déplacement, logement etc.) ➤ Flexibilité et adaptabilités selon les disponibilités des apprenants (heure, lieu) ➤ Individualisation et ajustement des parcours de formation suivant les compétences et les objectifs pédagogiques prédéfinis ➤ Contraintes logistique réduites (pas de location de salle, de déplacement des salariés, de logement etc.) ➤ Reporting précis des formations et analyse des résultats automatisée grâce au tracking ➤ Supports de formation pérennes et actualisables 	<ul style="list-style-type: none"> ➤ Entreprises parfois mal informées et frileuses face aux nouvelles technologies ➤ Pas de contrôle sur la motivation, ➤ l'implication et la gestion de leur formation par les apprenants ➤ Investissement en matériel informatique et en logiciels ➤ Conduite du changement à mettre en œuvre dans le service formation

Tableau I.3: *Les avantages et inconvénients du coté de l'entreprise (Université, école ...).*

Conclusion :

Dans ce chapitre, nous avons présenté une vue globale sur le E-Learning, nous avons commencé par définir le mot E-Learning, ensuite, décrire quelques concepts de base. Enfin nous avons énuméré quelques avantages et inconvénients du côté des acteurs de ce dernier. Dans le prochain chapitre nous allons présenter l'organisme d'accueil pour lequel on a réalisé cette application.

Chapitre II

Introduction :

L'INSIM est un établissement privé renommé nationale et internationale. Il dispense un enseignement de qualité inégalable. Les cours sont dispensés par une équipe d'auteurs chercheurs, en majorité des docteurs d'état en sciences de gestion, qui ont une grande expérience d'enseignement et de conseil, aux USA, en Europe et en Algérie.

II.1. Présentation de l'organisme d'accueil :**II.1.1. Historique de l'INSIM :**

INSIM, initiales École de formation professionnelle, est un institut agréé par le ministère de la formation et de l'enseignement professionnel. Il a été créé en 1994 à Hydra-Alger par un groupe d'enseignants chercheurs.

INSIM Tizi-Ouzou, est l'une des filiale d'INSIM Alger, créée en 2004 par Mr. ZEROUROU Mohammed et Mr. LAMIRI Abdelhak, agréé par l'état sous le N° 72/DFEP/RAA du 21Fevrier 2004. L'établissement dispense des formations dans diverses spécialités des sciences de gestion et des Nouvelles Technologies de l'Information et de la Communication (NTIC) dans différents paliers, allant du Brevet Technicien Supérieur «BTS» au Master «MBA», de même qu'il assure des missions de conseil et d'expertise auprès des entreprises.

L'INSIM Tizi-Ouzou rayonne désormais sur l'ensemble du territoire national et accueille plus de 8000 étudiants. Et pour accroître son rayonnement culturel et s'inscrire dans une perspective de modernité et se rapprocher des standards d'enseignements universels, l'INSIM s'est ouvert à l'international en développant des partenariats avec des institutions de renommé.

II.1.2. Situation géographique :

II.1.2.1. Adresse de l'institut :

Le siège de l'INSIM se situe dans la nouvelle ville de Tizi-Ouzou, plus précisément au «Centre d'affaires Glaxy Zhun Sud - Quartier B - N°03. Nouvelle ville- Tizi-Ouzou».

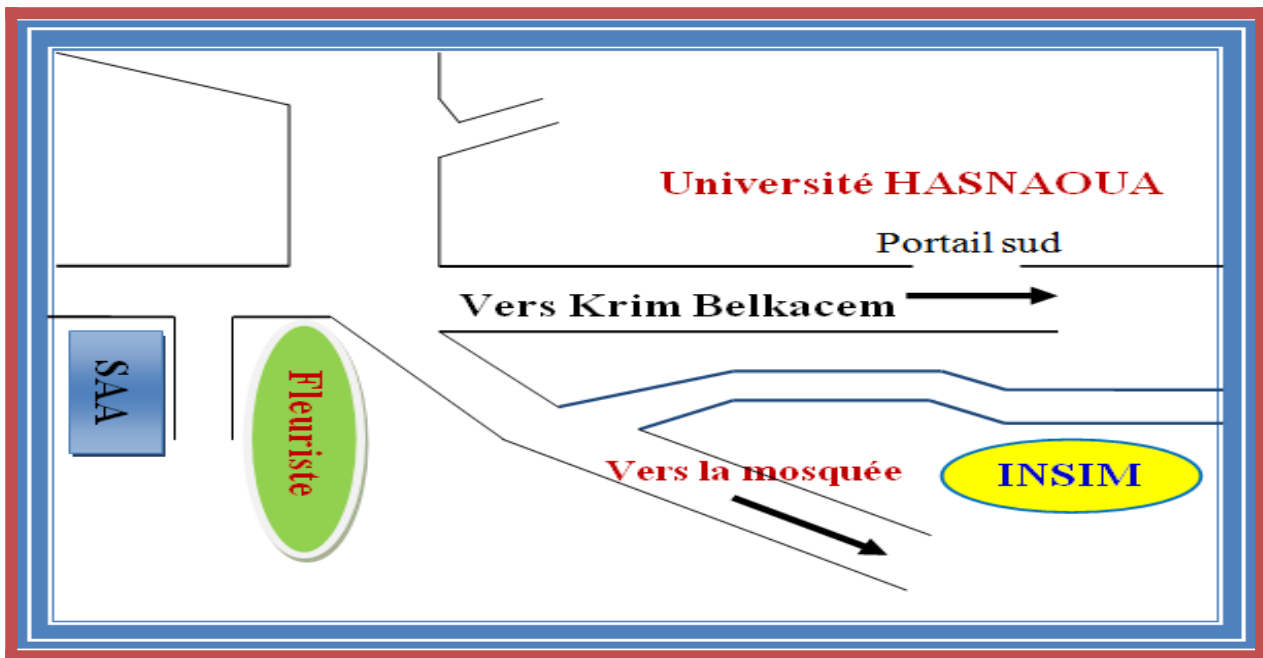


Figure II.1 : Adresse de «l'INSIM».

II.1.2.2. Emplacement de l'institut INSIM par satellite :



Figure II.2. Emplacement de l'institut «INSIM».

II.1.3. Activité principale et mission :

L'INSIM, en tant que projet sociétal, a pour mission de former les cadres et les managers de demain, de contribuer à l'éclosion de jeunes entrepreneurs par des formations adaptées et d'accompagner les entreprises à relever les nombreux défis qui les attendent dans un environnement économique de plus en plus concurrentiel et fortement instable.

Son ouverture à l'international, ses nombreux partenariats et la diversification de ses programmes font de l'INSIM une institution de référence dans le domaine de la formation, en témoigne les nombreuses entreprises (publiques, privées et multinationales) qui recrutent des étudiants de l'INSIM, avant même la fin de leur formation pour certains d'entre eux.

II.1.3.1. Filiales du groupe INSIM :

Filiales	Opérationnelle depuis
Tizi-Ouzou	Février 2004
Bejaia	Septembre 2004
Annaba	Septembre 2004
Hassi Messaoud	Septembre 2004
Oran	Septembre 2005
Constantine	Septembre 2007
Bouira	Février 2009
Sétif	Septembre 2009
Boumerdes	Septembre 2009
Ouargla	Septembre 2010
Blida	Septembre 2010

Tableau II.1 : Filiales du groupe «INSIM».

II.1.3.2. Partenaires du groupe INSIM :

Partenaires	Formations développées
École Supérieur de Gestion de Paris (ESG)	➤ Master in Business Administration «MBA» - Gestion des entreprises. - Gestion des ressources humaines. - Marketing et publicité. - Audit et contrôle de gestion. - Finance.
École Supérieur de Gestion et de Commerce International de Paris (ESGCI)	➤ Bachelor of Business Administration «BBA» - Ingénieur commercial. - Ingénieur d'affaires.
École Supérieur de Génie Informatique de Paris (ESGI)	➤ Bachelor en Informatique - Informatique de Gestion - Gestion des Réseaux - Développeur Web ➤ Master en Informatique - Architecture logicielle - Système, réseaux et sécurité - Réseaux de télécommunication et mobilité
Institut National des Techniques Economiques et Comptables Paris (INTEC)	➤ Diplôme en gestion et comptabilité (DGC)
Ecole Française des Attachés de Presse et des Métiers de la Communication (EFAP)	x - Master en Communication x - Licence en Communication x - Responsable de la Communication x - Attaché de presse x - Chef de publicité x - Chargé de la Communication interne x - Directeur d'agence de Communication
Collège de Sherbrook Québec-Canada	➤ Technicien Supérieur (AEC) - Gestion des entreprises - Informatique de gestion - Gestion des Ressources Humaines - Gestion de supervision dans les organisations - Commerce international - Vente et Marketing - Comptabilité et Finance - Logistique et Transport
Collège Lasalle Québec-Canada	➤ Technicien Supérieur (AEC) - Informatique - Hôtellerie et Tourisme

CISCO Networking Academy (Boston, USA)	<ul style="list-style-type: none"> ➤ L'INSIM bénéficie du statut d'académie locale CISCO, qui permet de développer des formations en Informatique. - Instructeur. - Académique : CCNA1, CCNA2, CCNA3 et CCNA4.
Certification ICDL (International Computer Driving Licence)	<ul style="list-style-type: none"> ➤ Passeport de Compétence Informatique Européen (PCIE)
MICROSOFT MSDN academic alliance	<ul style="list-style-type: none"> ➤ INSIM utilise les produits de plateforme, de serveur et de développement MICROSOFT.
Brevet de Technicien Supérieur (Diplôme d'État)	<ul style="list-style-type: none"> ➤ Commerce international. ➤ Marketing. ➤ Gestion des ressources humaines. ➤ Banque. ➤ Assurance. ➤ Comptabilité et gestion. ➤ Informatique.

Tableau II.2 : Partenaires du groupe INSIM

II.1.3.3. Cadre de travail :

Par l'étendu et la diversité de ses moyens matériels, INSIM offre un cadre idéal pour la formation, réunissant les conditions d'un apprentissage de qualité et un environnement convivial. INSIM couvre une superficie considérable agrémentée d'un espace large disposant de :

- 22 Salles de cours.
- 04 salles machines.
- 02 salles machines pour les cours Cisco.
- 01 salle d'Examen Cisco.
- Un bureau de maintenance informatique (moyens généraux).
- Une salle d'enseignants.
- Une bibliothèque.
- Une salle de lecture (médiathèque).
- Le bureau de l'assistante du directeur
- Quatre salles de cours A1, A2, A3 et A4
- Direction des études et de la pédagogie

II.1.4. L'organigramme de l'INSIM :

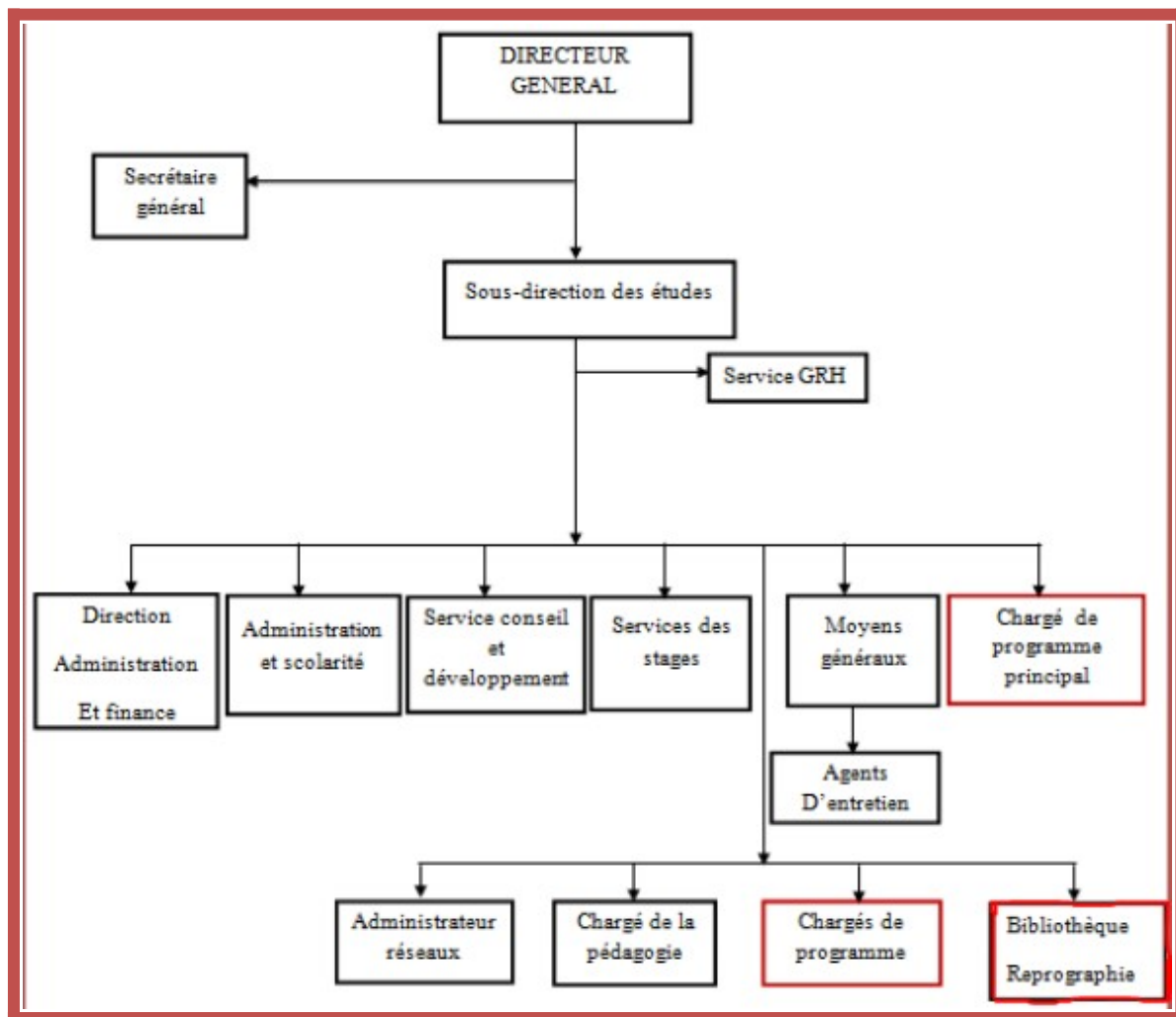


Figure II.3 : Organigramme de l'institut «INSIM».

II.1.5. Description des services :

II.1.5.1. Direction générale :

- Implémentation d'une politique qui a pour objectif de définir les grandes lignes de conduite de l'Institut, de les traduire en stratégie d'action et de prévoir les résultats escomptés.
- Recueillir en permanence toutes les informations nécessaires sur l'Institut et son environnement.

II.1.5.1.1. Secrétariat de direction :

- Assurer le secrétariat de la Direction Générale.
- Contribuer au développement de l'image du service interne et externe de l'Institut.
- Assurer le secrétariat des différents services.

II.1.5.1.2. Département des relations extérieures «RELEX» :

- Assurer le développement de nouveaux produits.
- Suivre le développement des marchés en permanence.
- Organiser la promotion des produits existants.
- Séminaires.
- Conférences.
- Rencontres.
- Journées portes ouvertes.

II.1.5.1.3. Département administratif et financier :

- Prise en charge de l'organisation du cadre de travail et de l'affectation des ressources humaines, matérielles et financières nécessaires à l'activité de l'Institut.
- Gestion administrative de l'ensemble du personnel et des étudiants.
- Application de la politique financière de l'Institut.
- Assure l'interface avec le commissaire aux comptes.
- Veille au respect de la réglementation dans tous les domaines relevant des compétences de la structure.

II.1.5.1.4. Chargé des moyens généraux :

- Oriente, coordonne et contrôle l'ensemble des activités de soutien logistique nécessaire au bon fonctionnement de l'Institut.
- Assure la gestion du patrimoine de l'entreprise.
- Gère les actes de propriété et les contrats de location.
- Assure les achats et la gestion des stocks.
- Assure les travaux d'entretien général y compris le nettoyage.
- Assure la sécurité et l'entretien des lieux.

II.1.5.1.5. Service scolarité / admission :

- Assurer l'ensemble des activités liées à la scolarité.
- Réception des étudiants et des parents d'élèves.
- Exécution de toutes les opérations liées à l'admission (préinscription, inscription, etc...)
- Constitution des dossiers administratifs des étudiants.
- Elaboration des conventions ou des contrats de formation.
- Elaboration des échéanciers de paiements.
- Transmettre une copie de la convention et de l'échéancier de paiement à la structure comptabilité.

II.1.5.1.6. Département chargé des études :

- Assurer l'activité pédagogique de l'ensemble des programmes d'enseignements.
- Préparer les programmes d'enseignements
- Assurer la coordination des programmes d'études suivants :
 - Programmes d'études DESSG, DESCF (MBA).
 - Programmes d'études PGSM (Ingéniorats).
 - Programmes d'études TSA (Diplômes d'État).
 - Programmes d'études Collège la salle.
 - Programmes d'études Collège Sherbroke.
 - Programmes d'études PRE BTS.
- Conception et mise en place du système d'évaluation.
- Assurer la gestion des dossiers pédagogiques des étudiants (Relevés des notes etc.)

II.1.5.1.7. Chargé des formations principales :

- Assurer l'activité pédagogique de la structure.
- Assurer la mise en œuvre des programmes.
- Établir les emplois de temps.
- Assurer la gestion du cours enseignant.
- Etablir les plannings des examens (semestriels, annuels et de rattrapage) et en assurer le bon déroulement.
- Organiser et gérer les activités de soutenance.
- Assurer la gestion et le suivi des étudiants et des enseignants (assiduité ponctualité, etc...).
- Réunions périodiques avec les délégués de groupe et des enseignants.

II.1.5.1.8. Chargé des stages :

- Assurer la gestion du processus des stages.
- Élaboration de la liste des étudiants partant en stage en relation avec les chargés de programmes.
- Élaboration des thèmes par spécialité en relation avec les chargés de programmes.
- Désignation des promoteurs et des entreprises.
- Élaboration des conventions de stage avec les entreprises.
- Suivi des étudiants durant le stage.
- Évaluation du rapport de stage par l'entreprise et le promoteur (suivi).
- Participer à l'élaboration du budget.
- Veiller à l'application du règlement intérieur de la structure.

II.1.6. Aspect matériel et logiciel :

II.1.6.1. Aspect matériel :

Bloc B :

x Administration et scolarité :

- 03 Micro-ordinateurs :
 - ❖ 01 micro-ordinateur : RAM= 1GO, Microprocesseur= 3 GHZ, Disque dur= 80Go.
 - ❖ 02 micro-ordinateur : RAM=3 GO, Microprocesseur =3 GHZ, Disque dur = 80GO.
- 02 Onduleurs.
- 01 Imprimante CANON.
- 01 imprimante EPSON Multifonctions.

x Finance :

- 02 micro-ordinateurs :
 - 01 micro-ordinateur : RAM= 1 GO, Microprocesseur=3 GO, Disque dur= 80Go.
 - 02 micro-ordinateur : RAM= 3 GO, Microprocesseur=3.20 GHZ, Disque dur=80GO.
- 02 Onduleurs .
- 01 Imprimantes.

x Pédagogie :

- 04 Micro-ordinateurs :
 - ❖ 04 Micro-ordinateur : RAM=4GO, Micro-processeur=3 GHZ, Disque dur= 80GO.
- 04 Onduleurs.
- 01 Imprimante Samsung.
- 01 Imprimante Canon multi fonctions.
- 01 Imprimante Canon.

x R/H :

- 01 Micro-ordinateur.
- 01 Imprimante Canon.

x Administrateur réseau :

- ❖ 01 Micro-ordinateur : RAM= 1Go, Microprocesseur = 2.33 GHZ, Disque dur = 50GO, serveur power Edge 2950.
- ❖ Capacité =1 tira ,2 disque =500 Go, RAM = 8GO ,2 Processeurs processeurs, Windows 2003.

Bloc E :**x E1 :**

- 19 Micro-ordinateurs (Windows 7, RAM=2GO, Microprocesseurs=2.7 GHZ, Disque dur=235GO, Système : logiciel Cisco, MS Project).
- 19 Onduleurs.

x Salle E2 : 18 Micro-ordinateurs (RAM= 3Go, Microprocesseur=3.20, Disque dur =80 Go).

x E3 :

- 17 Micro-ordinateurs (RAM= 4GO, Microprocesseur= 2.70 GHZ, Disque dur= 235 GO).
- 17 onduleurs.

x E4 : 17 Micro-ordinateurs.

x E5 : 22 micro-ordinateurs RAM=3GO, Microprocesseur =3GHZ, Disque dur=80GO).

x E6 : 14 micro-ordinateurs :(RAM=1GO, Microprocesseur= 200GHZ, Disque dur =160GO).

Cellule Informatique :

- 01 micro-ordinateur.
- 01 onduleur.
- 01 Imprimante Epson multifonctions.

II.1.6.2. Aspect logiciel :

<p>Le système d'exploitation</p>	<ul style="list-style-type: none"> ➤ Toute les machines du site INSIM exécutent «Microsoft Windows 07 Professionnel version 2002 Service Pack 2/3» ➤ A l'exception d'une salle machine du BLOC E qui exécute «LINUX».
<p>Applications</p>	<ul style="list-style-type: none"> ➤ Bureautique : Offices 2003 et Offices 2007 ➤ Gestion commercial: Logiciel appelé : Ciel! ➤ Web: Programmation des sites web (Illustrator, PHP).
<p>Utilitaires</p>	<ul style="list-style-type: none"> ➤ Logiciel DeepFreeze: Il congèle le système c-à-d : impossible de modifier le contenu des ordinateurs. ➤ Adobe Reader : Permet de lire les fichiers avec l'extension PDF. ➤ Utilitaire WinRAR : Pour la compression ou la décompression des fichiers.

Tableau II. 3 : *Aspect Logiciel.*

II.1.7. Objectif de l'étude :

Dans nos temps modernes, plusieurs facteurs ont un impact sur l'apprentissage efficace, que ce soit les méthodes appliquées, le design du matériel, ou tout simplement les facteurs autres comme les outils et médias employés pour améliorer l'expérience pédagogique. L'e-learning offre beaucoup d'avantages pour l'apprenant dans un contexte scolaire. En effet, Chaque personne utilisant cette méthode d'enseignement peut le faire à sa façon et l'adapter à ses besoins. De plus, l'apprenant peut suivre la formation à son rythme. S'il a plus de facilité, celui-ci peut aller plus rapidement tandis que l'apprenant qui a plus de difficultés pourra y aller plus lentement. Ensuite, il y a la liberté d'accès à toute heure. Les étudiants peuvent donc suivre le cours quand bon leur semble: le matin, l'après-midi, le soir ou la nuit. Par la suite, l'accompagnement dans son parcours personnel par un formateur permet d'avoir des rétroactions sur son cheminement. Il y a aussi la prise en considération du projet personnel. Finalement, la souplesse du temps et du lieu ainsi que la formation à la portée de tous sont d'autres avantages du *E-Learning*. Cependant, la difficulté d'offrir en enseignement à distance toutes les formations offertes en présentiel, le coût de la formation et les problèmes techniques sont quelques inconvénients que l'on peut rencontrer lors de l'apprentissage en ligne.

L'INSIM de Tizi-Ouzou envisage de mettre en place une plateforme E-learning serait un élément favorisant le taux de réussite pour toutes les raisons énumérées ci-haut. Cette nouvelle forme d'interaction enseignant-étudiant permet de pouvoir sélectionner les méthodes les plus adaptées en fonction des besoins détectés. Il permet aux étudiants de devenir autonomes tout au long du parcours.

Conclusion :

L'INSIM est un établissement privé de renommée nationale et internationale. Pour valoriser son image de marque; l'INSIM collabore en partenariat avec des organismes nationaux et des institutions d'enseignement internationales de renommée mondiale en matière de gestion, telles que l'Université de Montréal (UQAM), le Collège de Sherbrooke (Québec), le Collège de Lasalle (Canada), la Paris Graduate School of Management (PGSM, ex – Ecole Supérieure de Gestion de Paris) et EFAP (Ecole Française des Attachés de Presse et des Professionnels de la Communication).

Dans le prochain chapitre nous allons entamer l'étape de l'analyse ainsi que la conception de notre application E-Learning destinée pour l'INSIM de Tizi-Ouzou.

Chapitre III

Introduction:

L'étape de 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 du troisième 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.

III.1. Problématique

L'INSIM offre des formations aux étudiants qui souhaitent acquérir ou approfondir leurs connaissances, développer leurs champs de compétences théoriques et pratiques, ainsi d'acquérir une formation avec une base solide et adaptée aux nouvelles exigences du marché du travail et aux nouvelles technologies, enfin décrocher des diplômes de renommée internationale.

Sauf que beaucoup d'étudiants n'arrivent pas à s'inscrire vu la surcharge d'emploi de temps de leurs travail, de la distance... A cet effet nous avons opté pour une autre solution «E-Learning» qui permet de s'inscrire chez soi et suivre des cours dans le temps libre, et d'avoir toute une équipe d'enseignants et pédagogues qui l'accompagneront tout au long de sa formation, ainsi offrir la possibilité à l'enseignant de faire des heures supplémentaires en quelques clics.

III.2. Objectif :

Dans notre projet nous allons réaliser une application E-Learning pour l'INSIM de Tizi-Ouzou qui permettra aux étudiants inscrits de suivre leur formation à leur propre rythme, à distance et complètement en ligne. Bien aussi d'obtenir une attestation de réussite et ou un diplôme de renommé internationale.

III.3. Analyse des besoins :

III.3.1. Sujet du projet :

L'institut de formations professionnelles INSIM opte à mettre en place une plate forme E-Learning qui permet à ses étudiants de suivre leurs formations, auxquelles ils se sont inscrit, à distance totalement en ligne. En effet, la 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 de l'INSIM, ils peuvent accéder à leurs espaces personnels en ligne.
- La plate forme permet aux enseignants d'insérer les contenus (cours, TD, TP ...), ainsi 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.

III.3.2. Acteurs et cas d'utilisation :

III.3.2.1. 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 administrateur pour assurer la gestion, comme dans notre cas, la gestion des utilisateurs, en ajoutant, modifiant, supprimant d'autres acteurs, ainsi la gestion entière des matières et formation.
- **L'enseignant** : Un enseignant est un internaute connecté, il a été inscrit et a été validé par l'administrateur en tant que professeur pouvant donner des cours et préparer des évaluations (examens), 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, étudiant, formé, stagiaire. Un étudiant est un internaute qui est connecté inscrit, validé par l'administrateur en tant qu'étudiant 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.

III.3.2.2. 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.	/
Étudiant	❖ Se connecter / se déconnecter.	/
	❖ Consulter contenus.	➤ Télécharger contenus. ➤ Remettre travail.
	❖ Passer examen.	/
	❖ Messagerie.	➤ Envoyer messages. ➤ Supprimer messages.
	❖ Consulter médiathèque.	

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

III.3.3. Spécifications des besoins : [1].

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

III.3.3.1. 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 III.2: *Les besoins fonctionnels.*

III.3.3.2. 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.

III.3.4. 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.

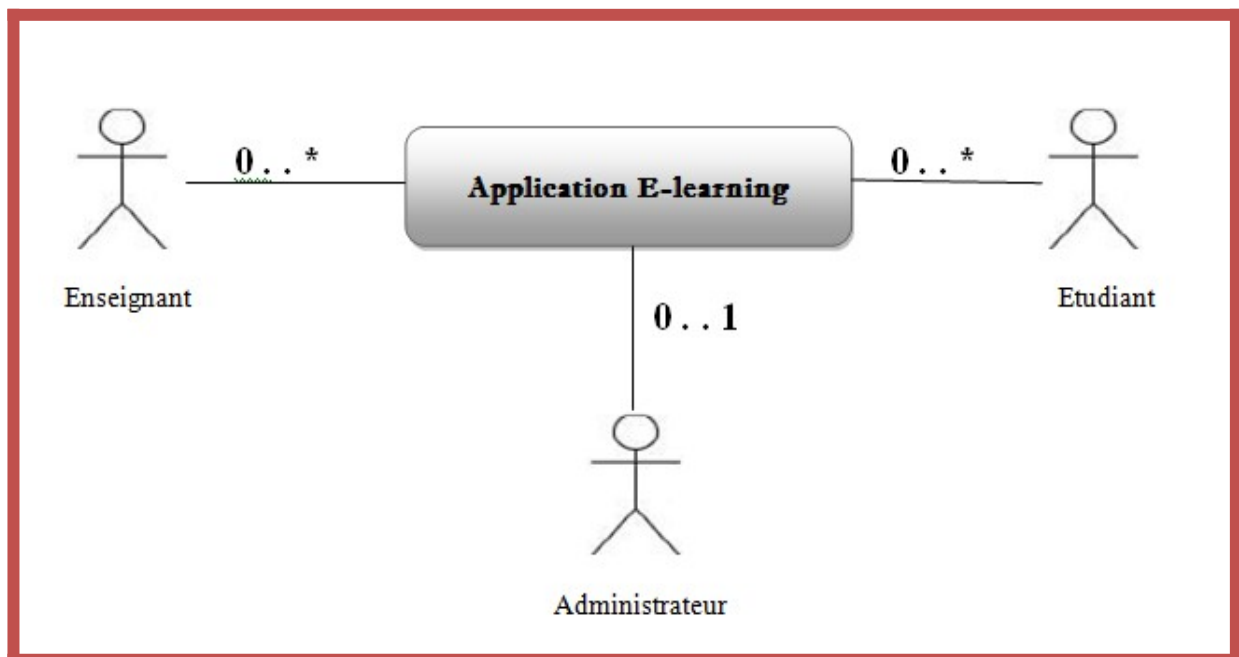


Figure III.1: Diagramme de contexte.

III.4. Conception de l'application :

III.4.1. 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 :

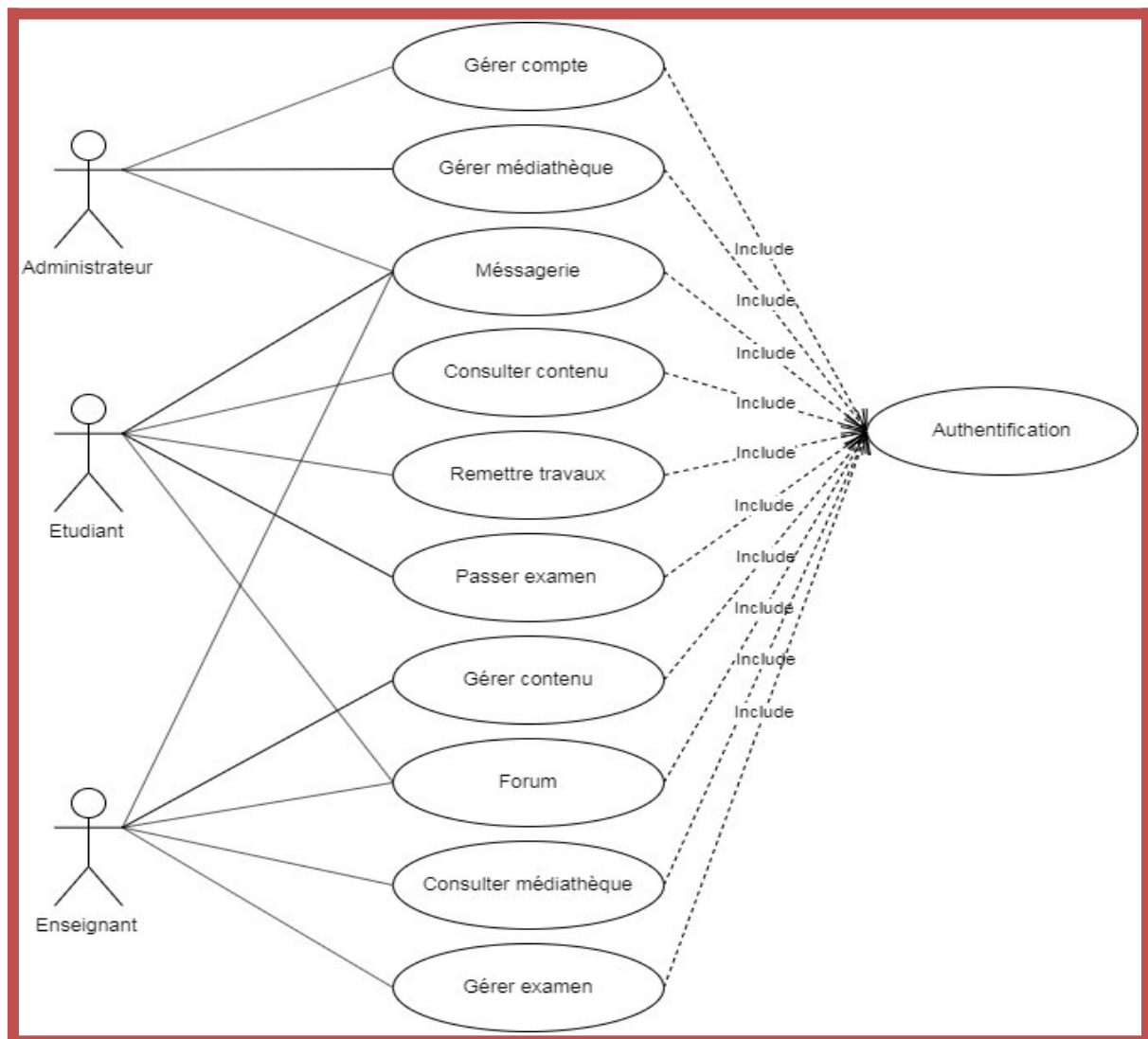


Figure III.2 : Diagramme de cas d'utilisation général.

III.4.2. 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.

III.4.2.1. 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.

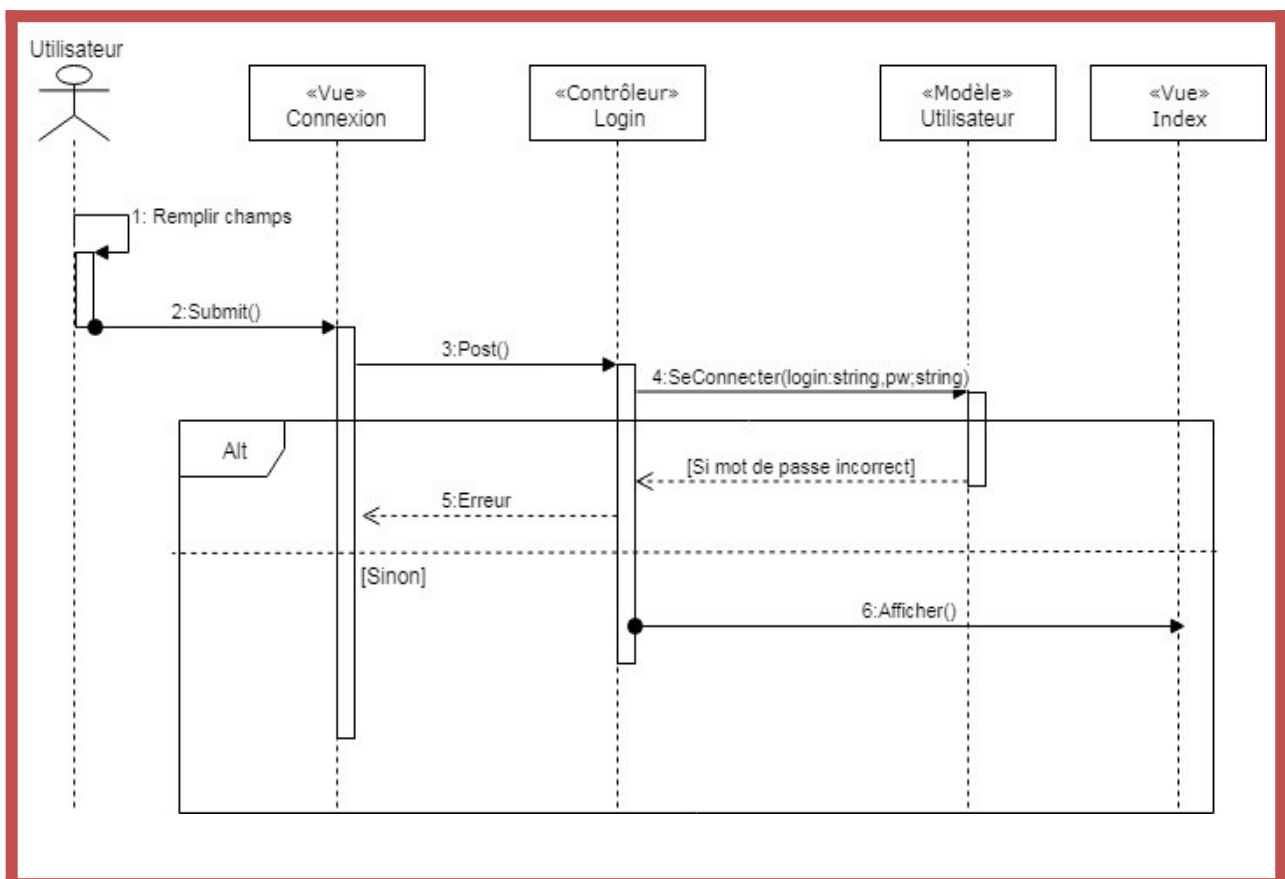


Figure III.3 : Diagramme de séquence «Authentification».

III.4.2.2. Inscription utilisateur :

Quand l'administrateur décide d'inscrire 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.

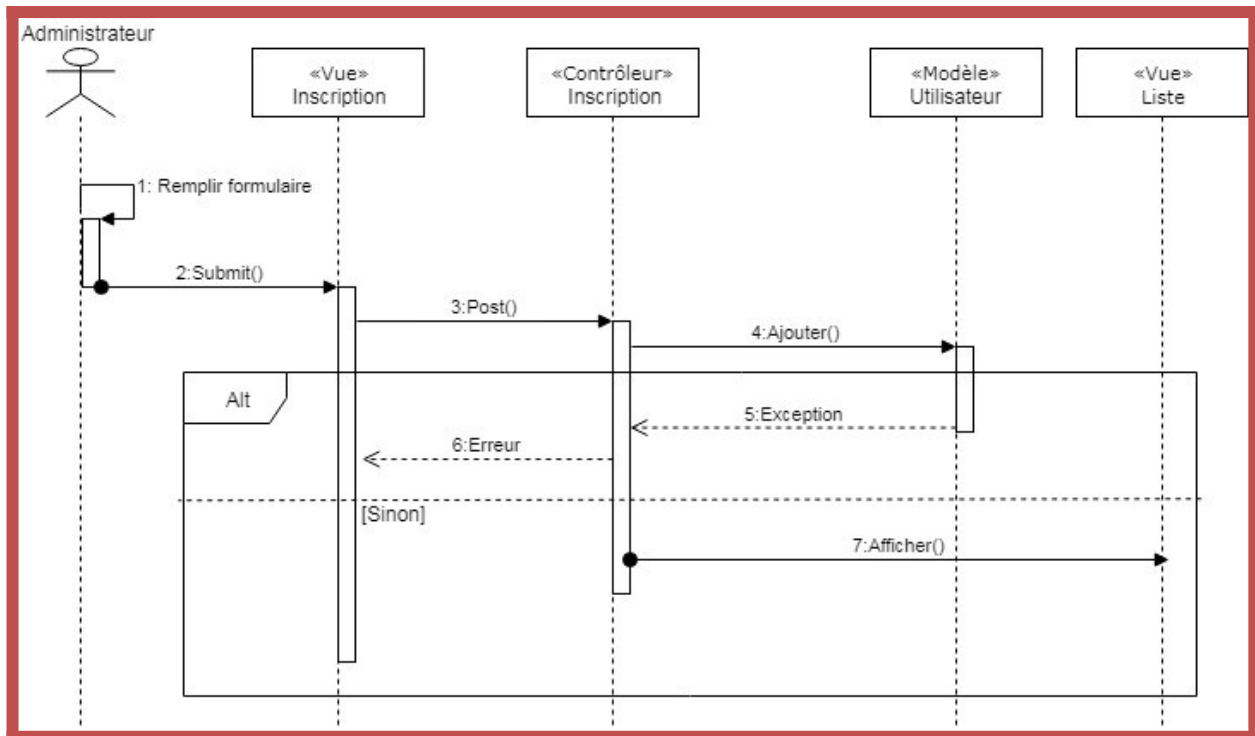


Figure III.4 : Diagramme de séquence «Inscription utilisateur».

III.4.2.3. 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.

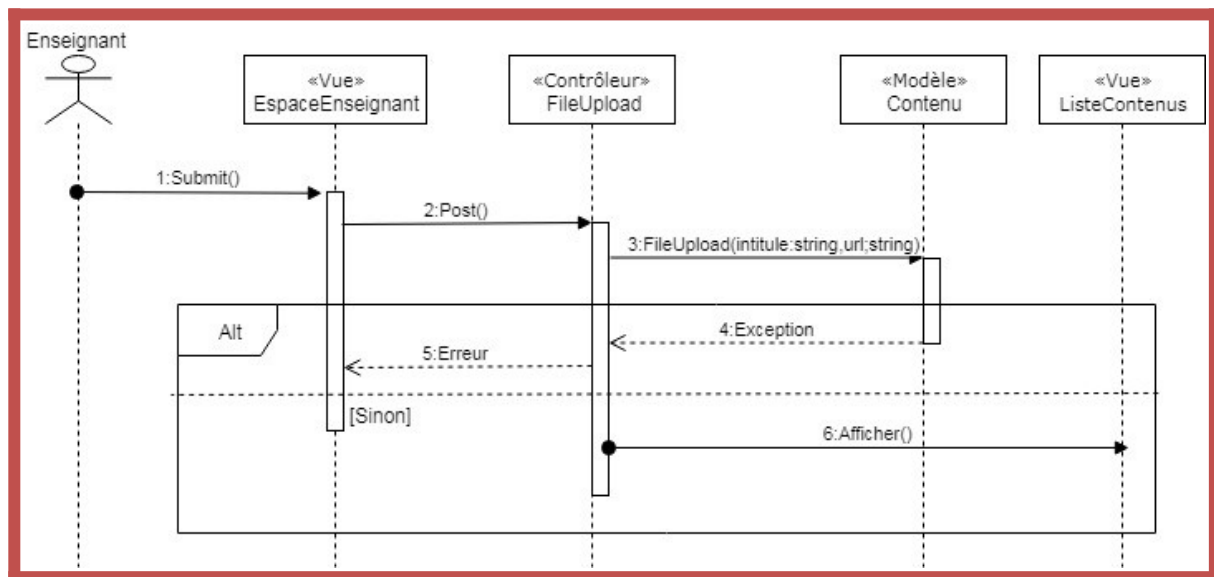


Figure III.5 : Diagramme de séquence «Ajouter Contenu».

III.4.2.4. 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 () et AjouterBareme (). 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.

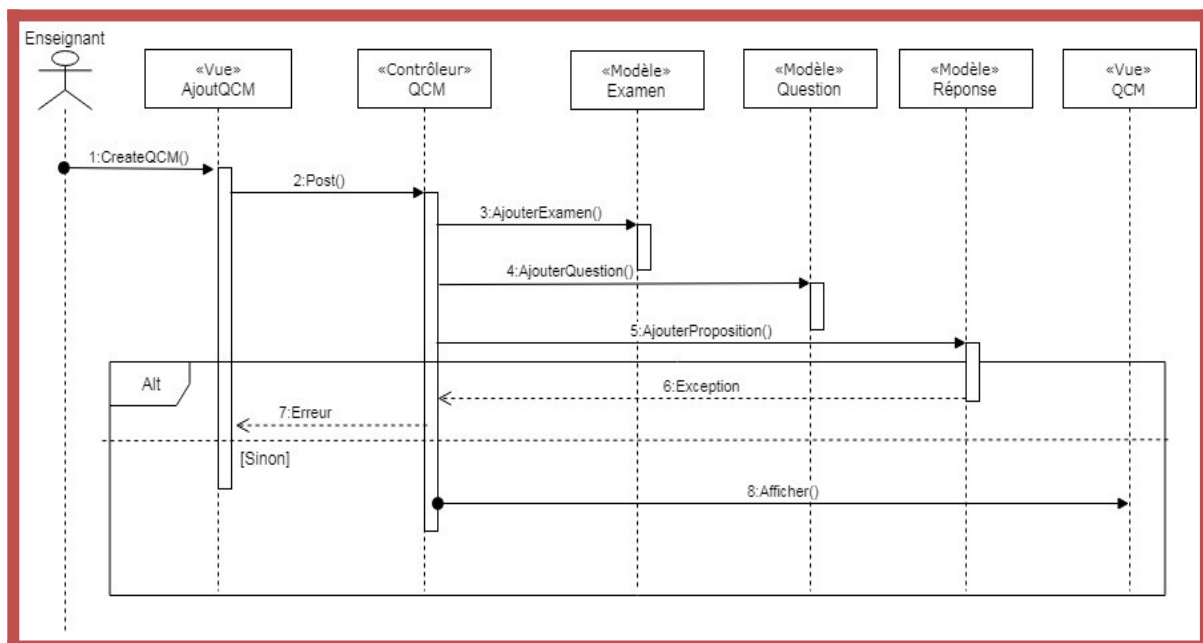


Figure III.6. Diagramme de séquence «préparation examen (création QCM)».

III.4.2.5. 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.

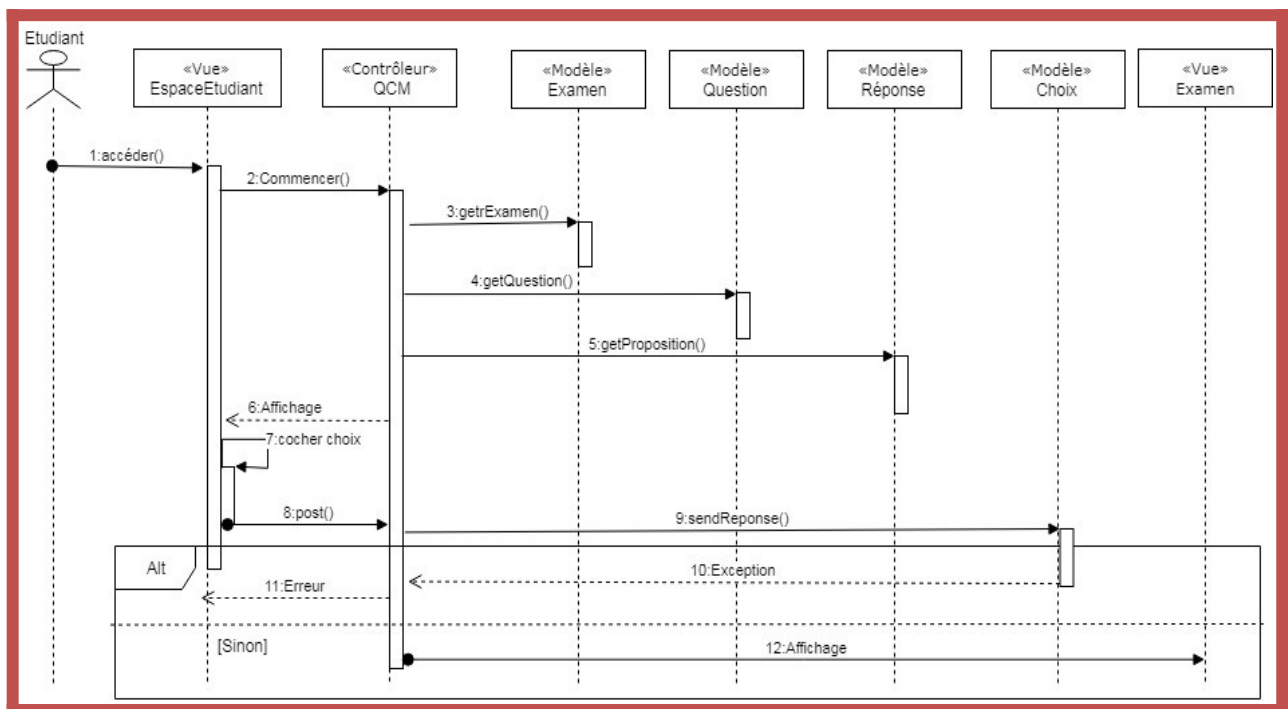


Figure III.7 : Diagramme de séquence de « Passer examen ».

III.4.2.6. Envoi 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.

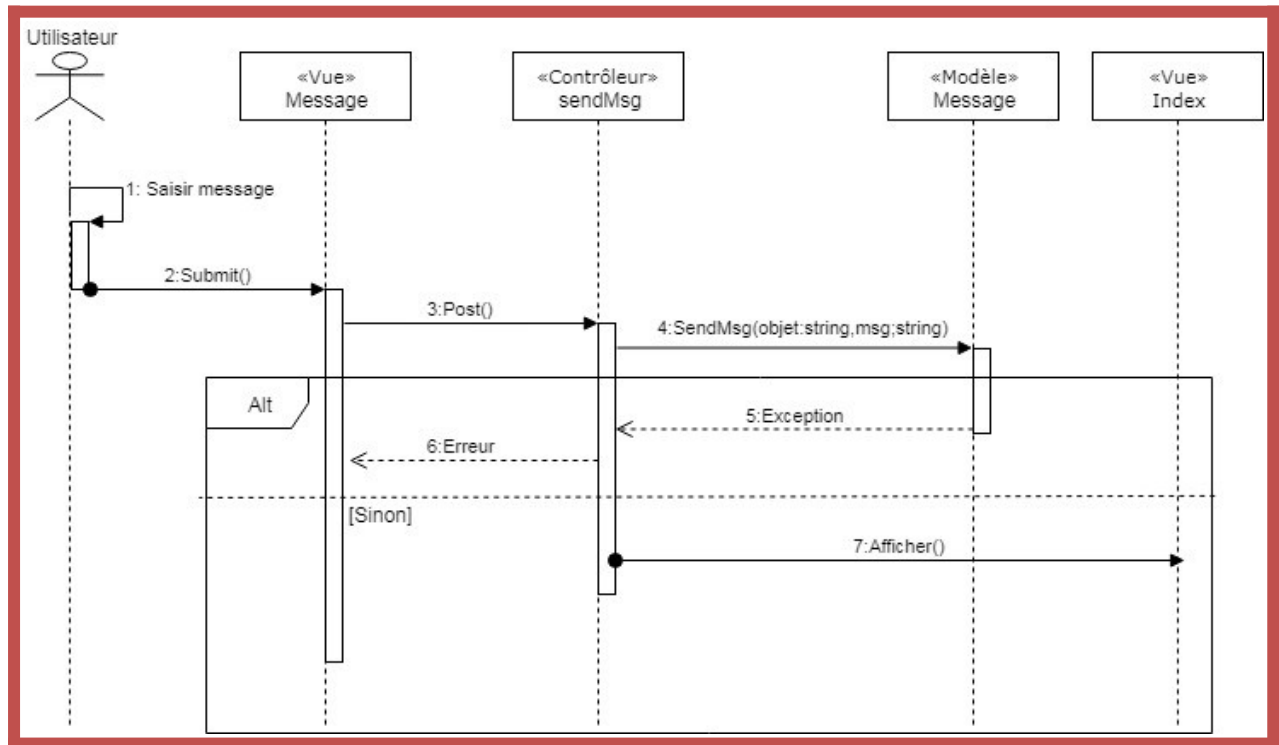


Figure III.8 : Diagramme de séquence «Envoi Message».

III.4.3. 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.

III.4.3.1. Gestion utilisateur :

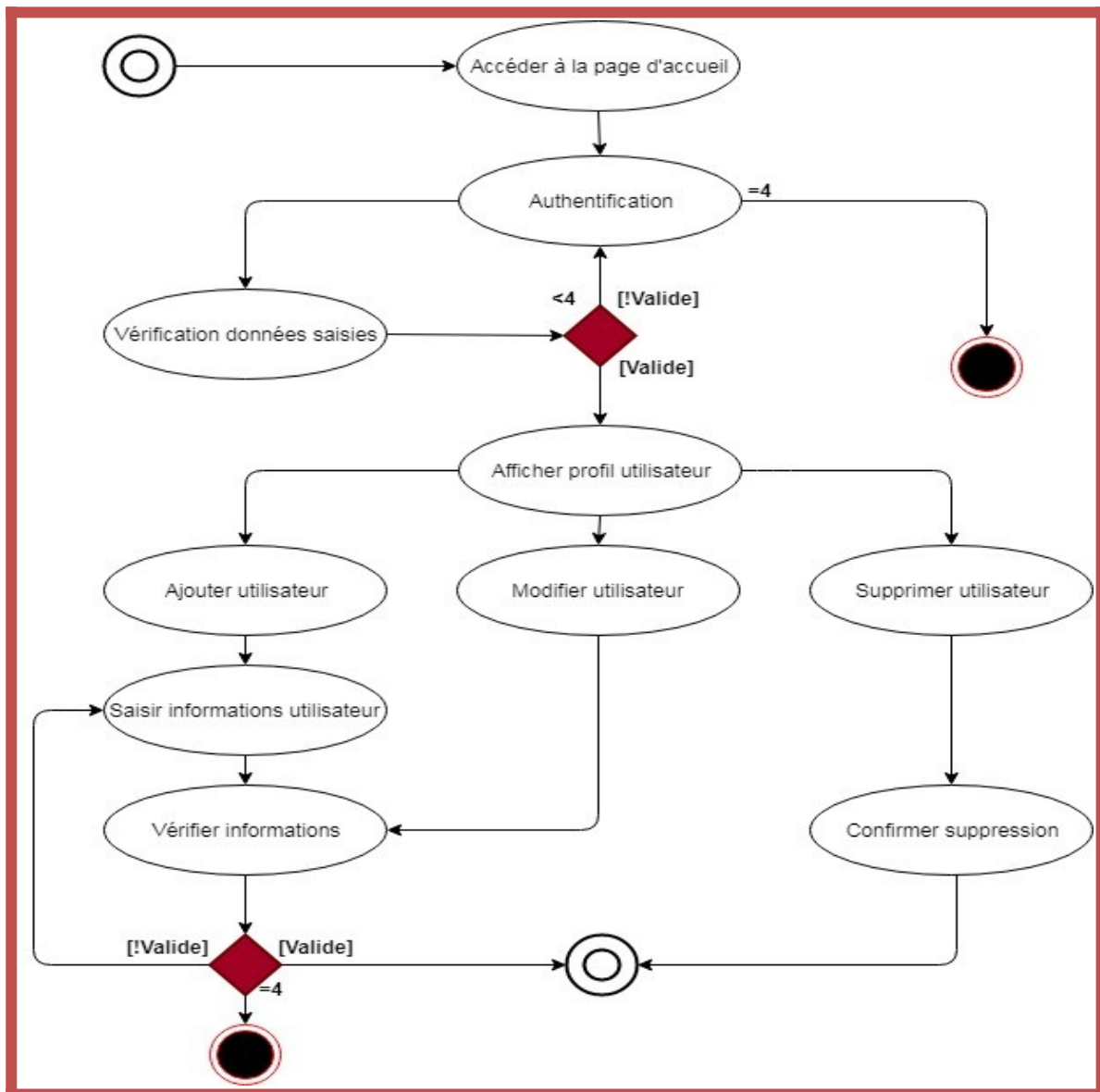


Figure III.9 : Diagramme d'activité «Gestion utilisateur».

III.4.3.2. Gestion des cours :

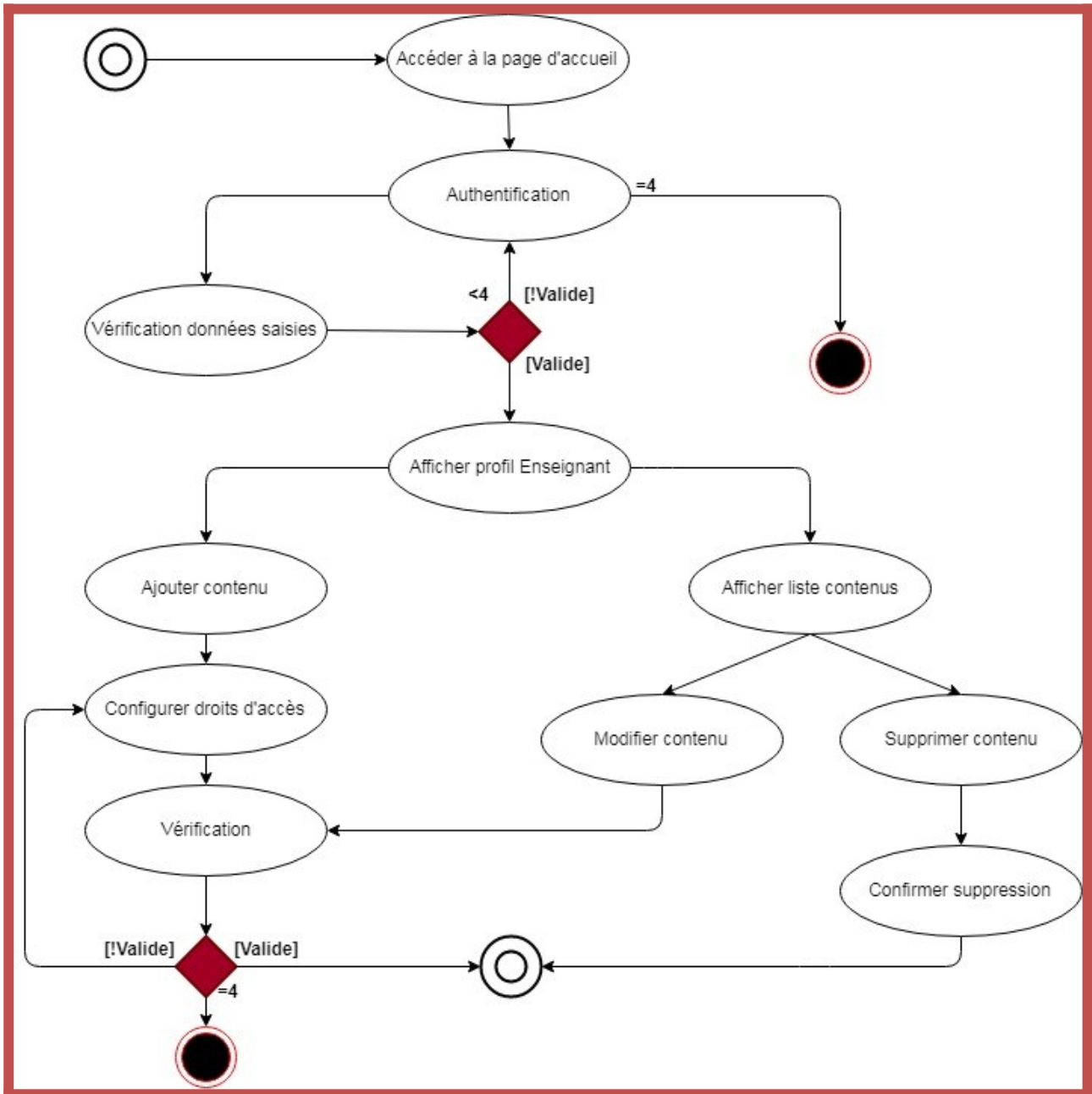


Figure III.10 : Diagramme d'activité «Gestion des cours».

III.4.3.3. Consultation messagerie :

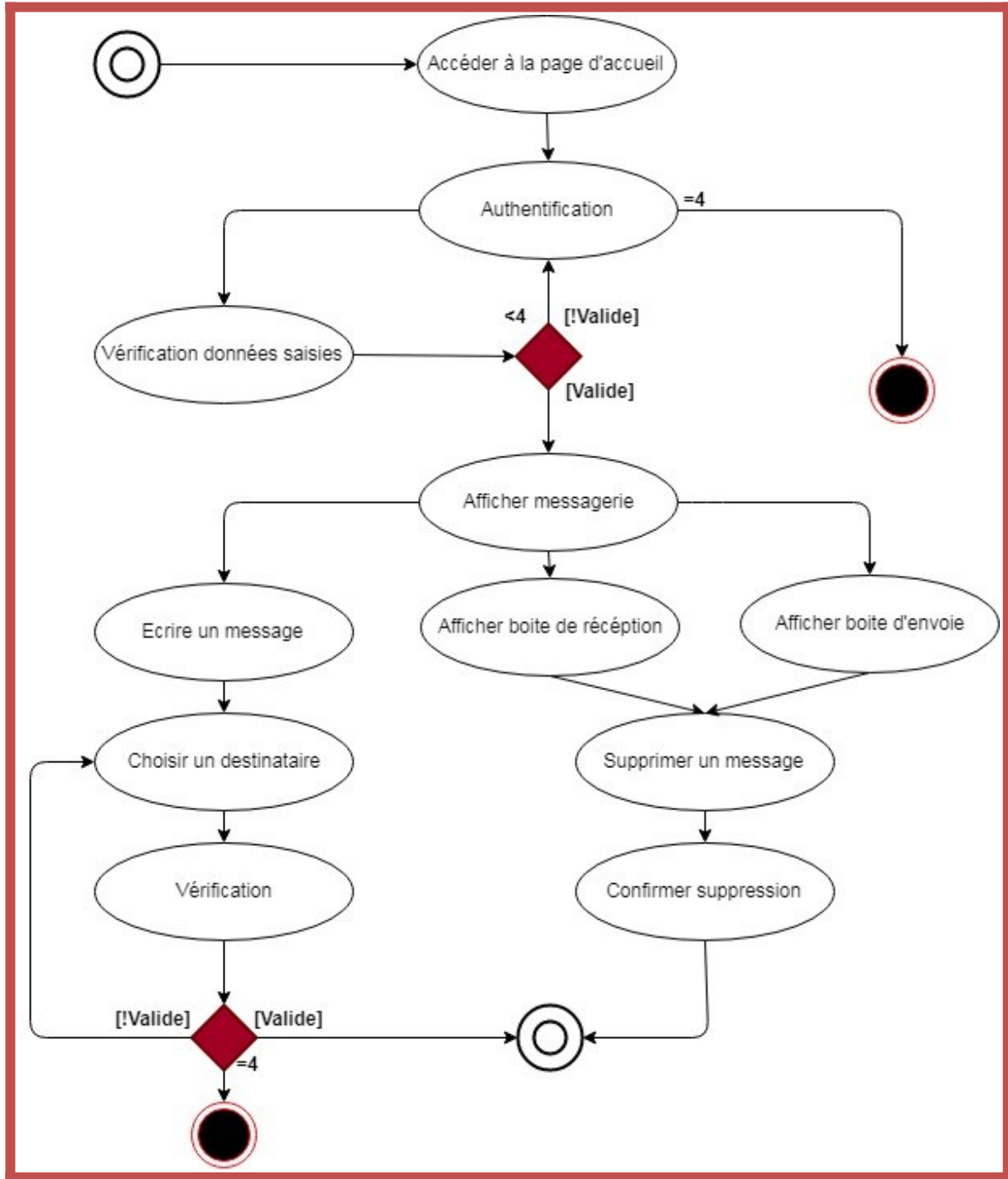


Figure III.11 : Diagramme d'activité «Consultation messagerie».

III.4.4. 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 :

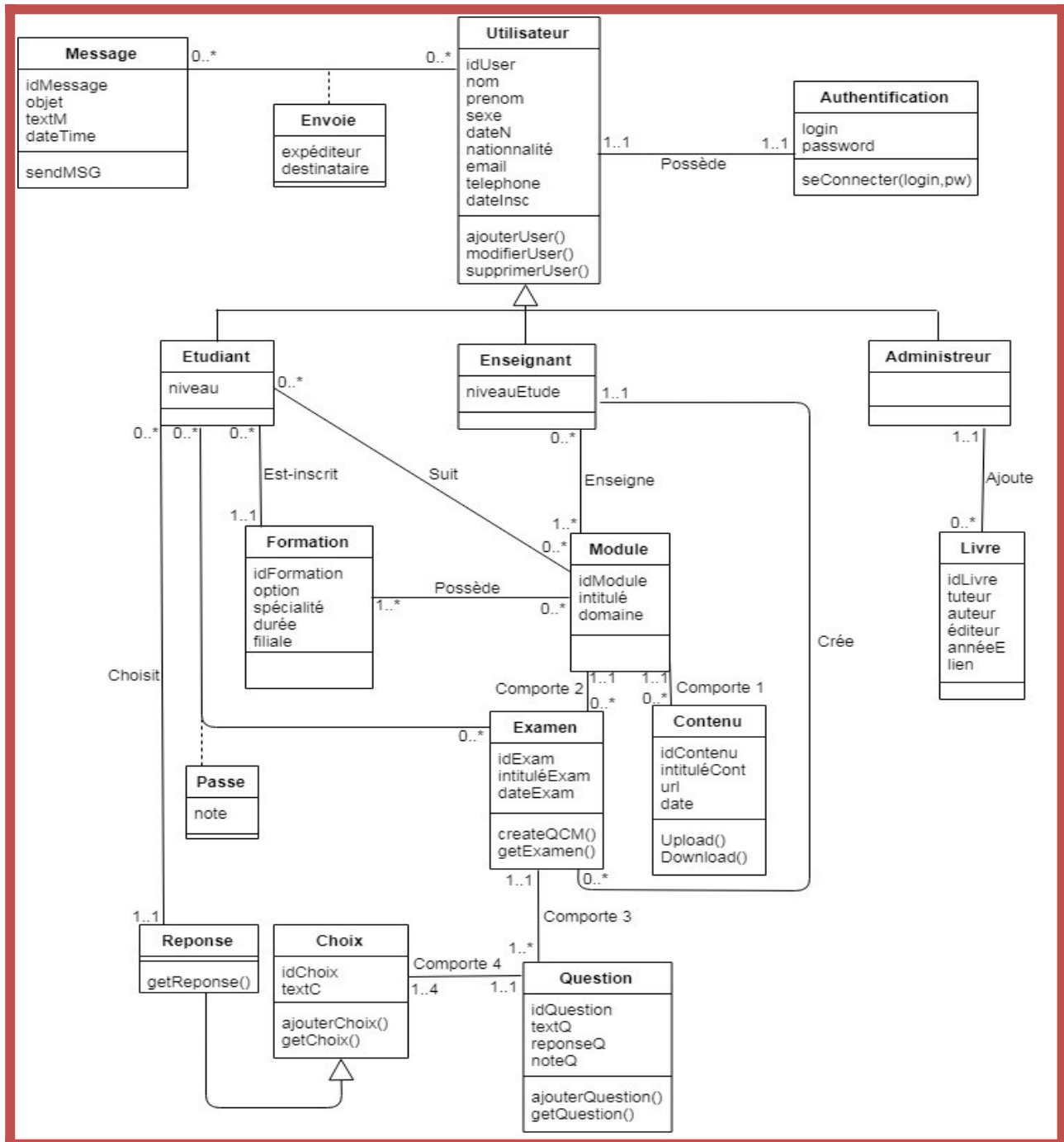


Figure III.12 : Diagramme de classes général.

III.4.4.1. Description des classes :

Nous décrivons en détaille toutes les classes utilisé dans le diagramme de classes précédant.

Classe Personne:

La classe personne 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 de l'institut INSIM, elle hérite de la classe Personne. 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 de l'école INSIM, elle hérite de la classe personne. 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 les administrateurs de l'INSIM, elle hérite de la classe Personne. 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 comparé à la réponse juste déclaré par l'enseignant.

Classe Contenu :

La classe contenu 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 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 par le groupe INSIM, chaque formation appartient à une catégorie (BTS, Licence, Master ...) et est caractérisé par une durée qui peut être généralement plusieurs mois. La formation donc a plusieurs modules, ainsi chaque formation appartient à une seule filiale du groupe INSIM.

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.

Chapitre IV

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.

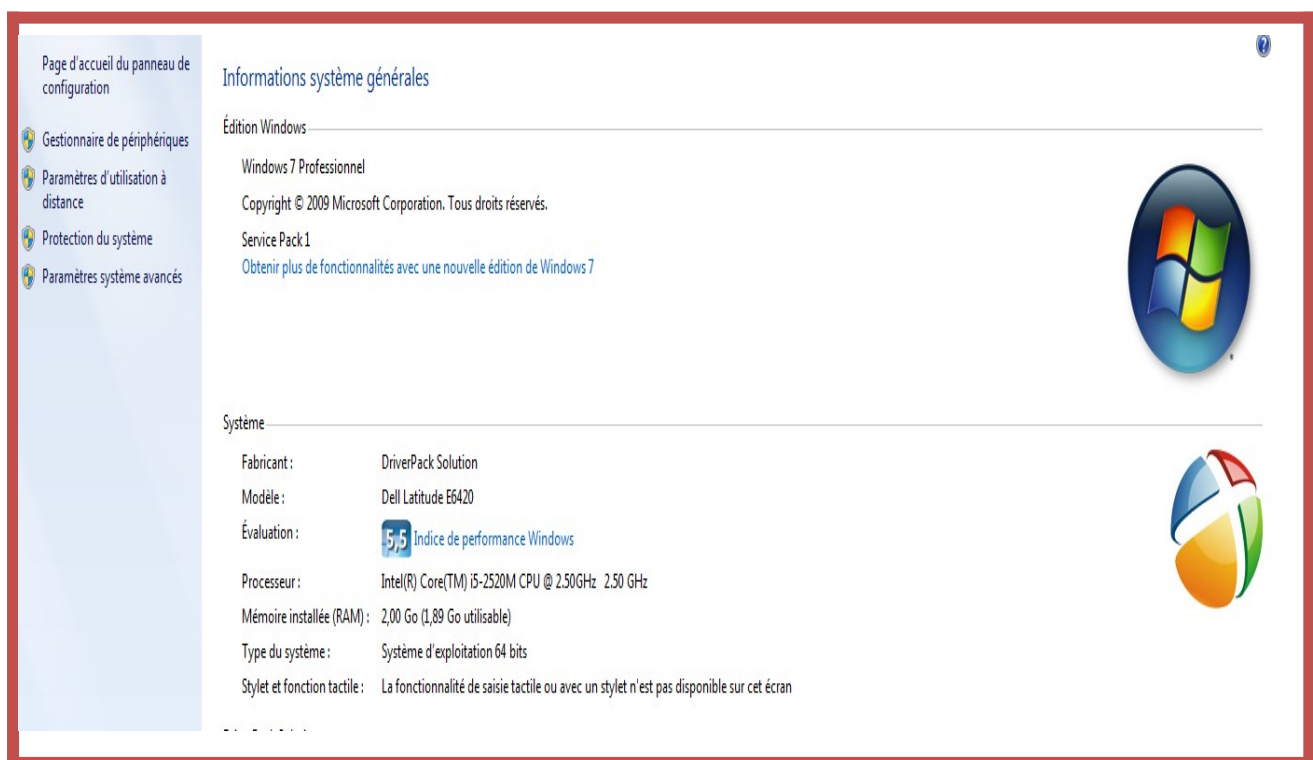
IV.1. Outils de développement et Langages de programmation :

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 :


IV.1.1. Outils de développement :

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

IV.1.1.1. Caractéristiques de l'ordinateur :



The screenshot displays the Windows 7 System Information window. On the left, there is a navigation pane with the following options: 'Page d'accueil du panneau de configuration', 'Gestionnaire de périphériques', 'Paramètres d'utilisation à distance', 'Protection du système', and 'Paramètres système avancés'. The main content area is titled 'Informations système générales' and includes the following details:

- Édition Windows:** Windows 7 Professionnel, Copyright © 2009 Microsoft Corporation. Tous droits réservés. Service Pack 1. [Obtenir plus de fonctionnalités avec une nouvelle édition de Windows 7](#)
- Système:**
 - Fabricant: DriverPack Solution
 - Modèle: Dell Latitude E6420
 - Évaluation:  **5,5** indice de performance Windows
 - Processeur: Intel(R) Core(TM) i5-2520M CPU @ 2.50GHz 2.50 GHz
 - Mémoire installée (RAM): 2,00 Go (1,89 Go utilisable)
 - Type du système: Système d'exploitation 64 bits
 - Stylet et fonction tactile: La fonctionnalité de saisie tactile ou avec un stylet n'est pas disponible sur cet écran

On the right side of the window, there are two circular icons: the Windows logo and the Windows Performance Index icon.

Figure IV.1 : *Caractéristiques de l'ordinateur.*

IV.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.

IV.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 es 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 IV.2 : Page d'accueil serveur «Apache Tomcat 9.0.16».

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.

IV.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 *tray icon* (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].

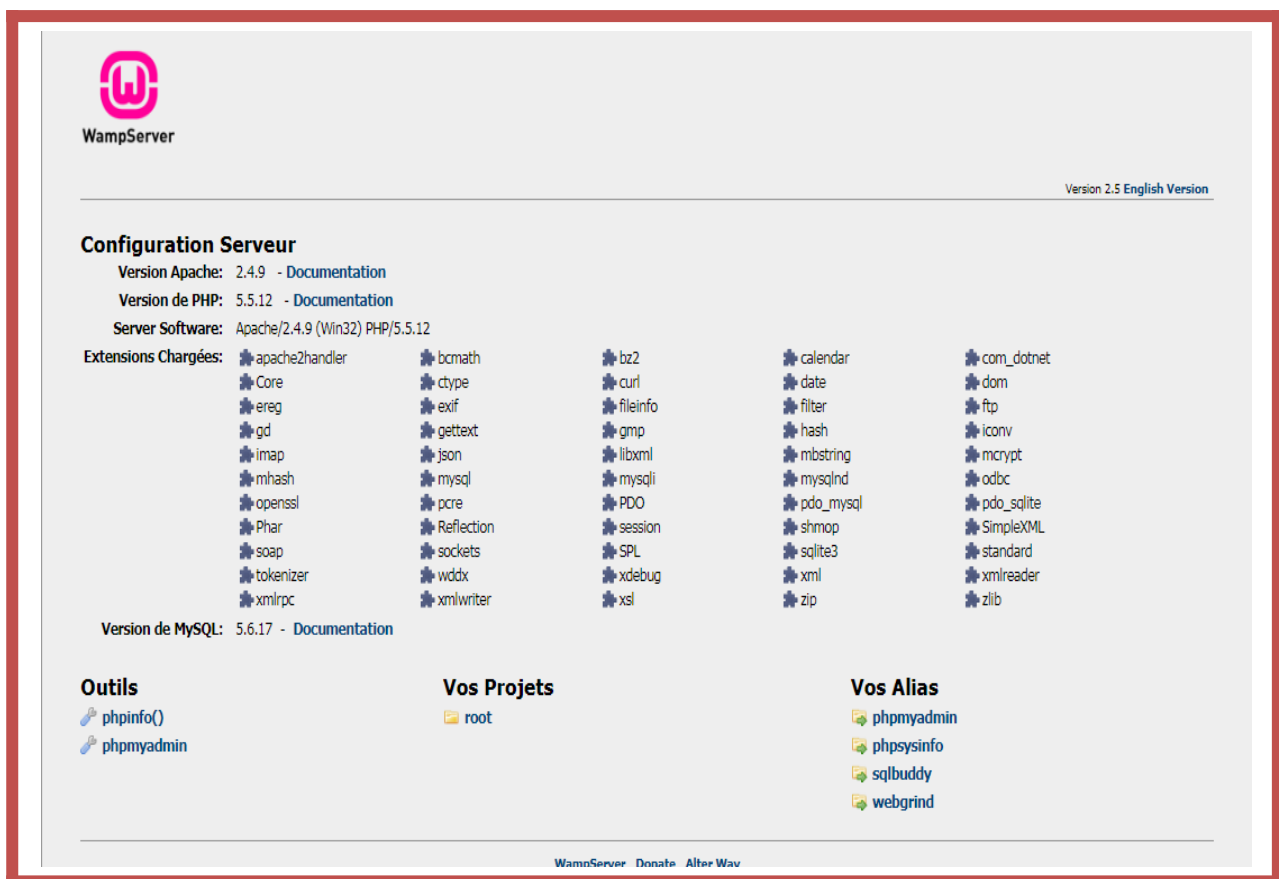


Figure IV.3 : Page d'accueil «WampServer».

IV.1.1.2.3. Jboss :

Jboss est un serveur d'application J2EE développé à partir de 1999 par un français Marc FLEURY. Ce serveur est écrit en Java et distribué sous licence LGPL. [4]

Jboss fournit un certains nombres de modules :

- JBossServer qui comporte une infrastructure constituée des conteneurs EJB, ainsi que du Java Management Extension (JMX).
- JBossMQ pour la gestion des messages JMS (Java Messaging service).
- JBossTX pour la gestion des transactions avec les API JTA(Java Transaction API) et JTS(Java Transaction Service).
- JBossCMP pour la persistance CMP.
- JBossSX pour la sécurité basée sur JAAS (Java Authentication and Authorization Service).
- JBossCX pour la gestion des connecteurs avec JCA (J2EE Connector Architecture).
- Tomcat ou Jetty pour le support des servlets et des pages JSP.

Jboss permet grâce au JMX de chargé les différents modules, conteneurs ou plugin en fonction des besoins. Bien entendu, Jboss permet d'implémenter ses propres services.

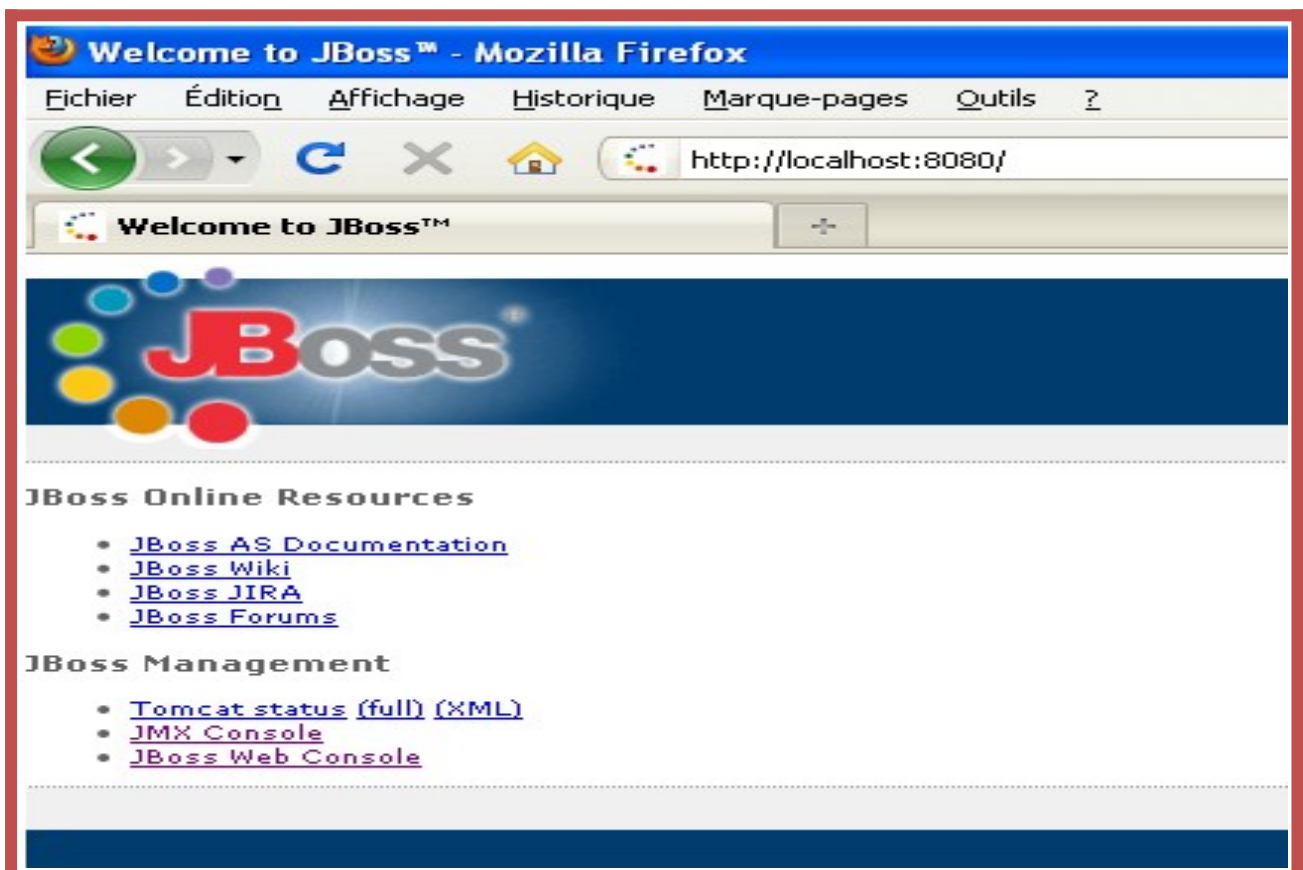


Figure IV.4 : Page d'accueil «Jboss».

IV.1.1.2.4. GlassFish :

GlassFish est désormais le serveur de référence JavaEE développé par Oracle en forte progression ces dernières années. GlassFish utilise en standard le moteur de persistance et propose plusieurs outils et services afin de faciliter la création et la maintenance d'applications Web.

GlassFish est principalement formé des composants suivants :

- Un serveur Web nommé Grizzly, permettant de répondre aux requêtes statiques des clients (pages HTML, images, JavaScripts...).
- Un serveur d'applications JavaEE conteneur de Servlets, pages JSP ou JSF.
- Un conteneur d'Entreprise JavaBean permettant la gestion des session beans, entity beans et message-driven beans.
- Une implémentation de l'API de persistance JPA basée sur EclipseLink.



Figure IV.5 : Page d'accueil «GlassFish».

IV.1.1.2.5. WebLogic :

Propriété de Oracle Corporation, Oracle WebLogic consiste en une famille de produits sur la plate-forme Java EE (auparavant J2EE) incluant :

- Un serveur d'applications J2EE, WebLogic Application Server.
- Un portail, WebLogic Portal.
- Une plate-forme Enterprise Application Integration.
- Un serveur transactionnel, WebLogic Tuxedo.
- Une plate-forme de télécommunications, WebLogic Communication Platform.
- Un serveur web HTTP.

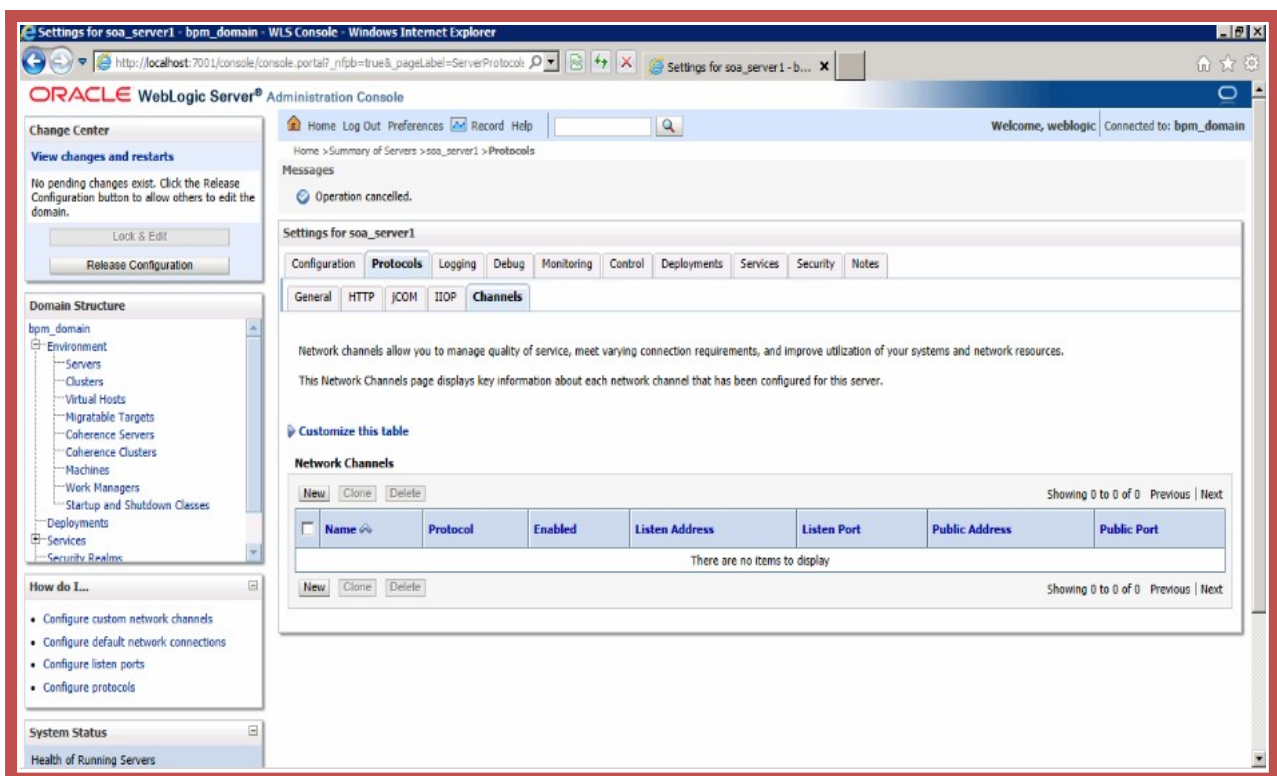


Figure IV.6 : Page d'accueil «WebLogic».

IV.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.

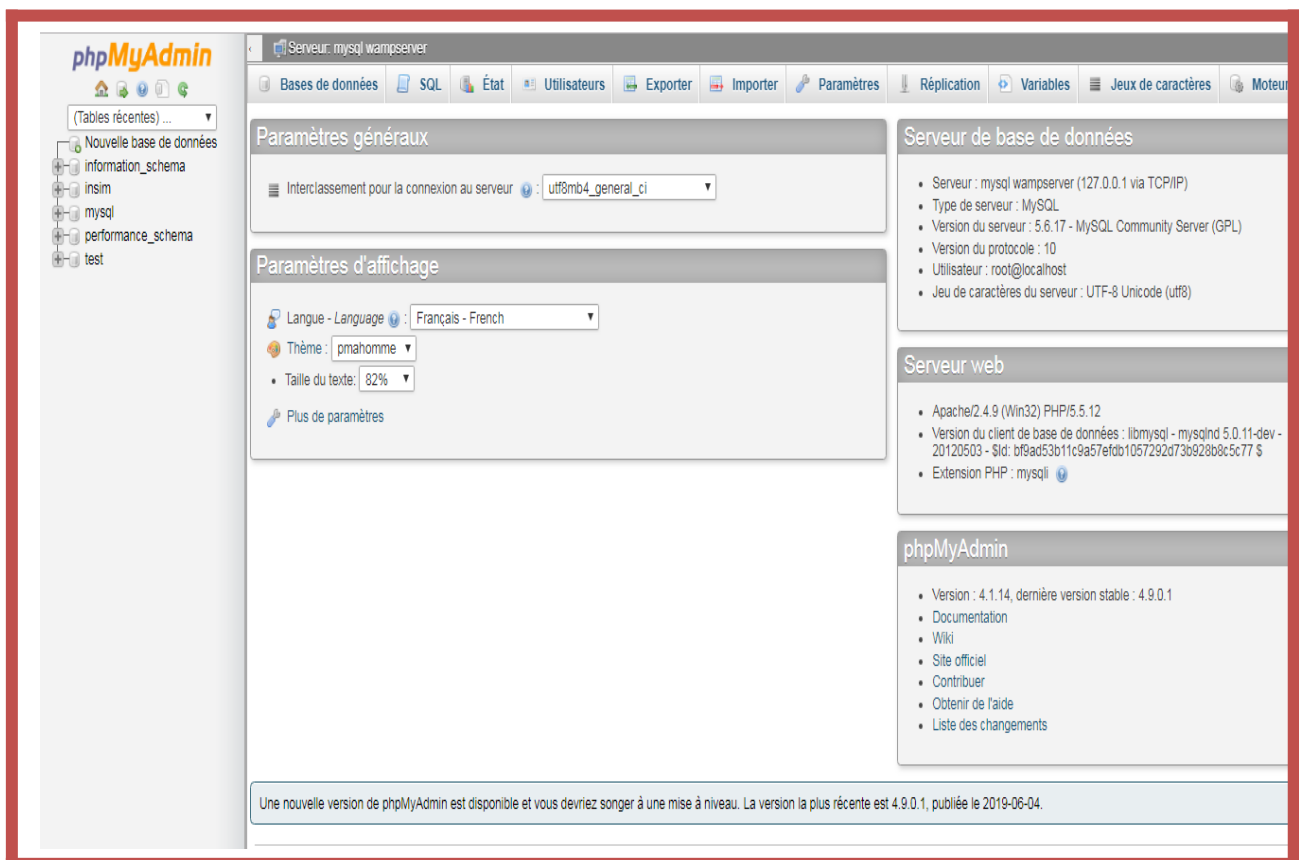


Figure IV.7 : Page d'accueil de «PhpMyAdmin».

IV.1.1.4. Eclipse IDE (Integrated Development Environment) :

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 œuvre 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.

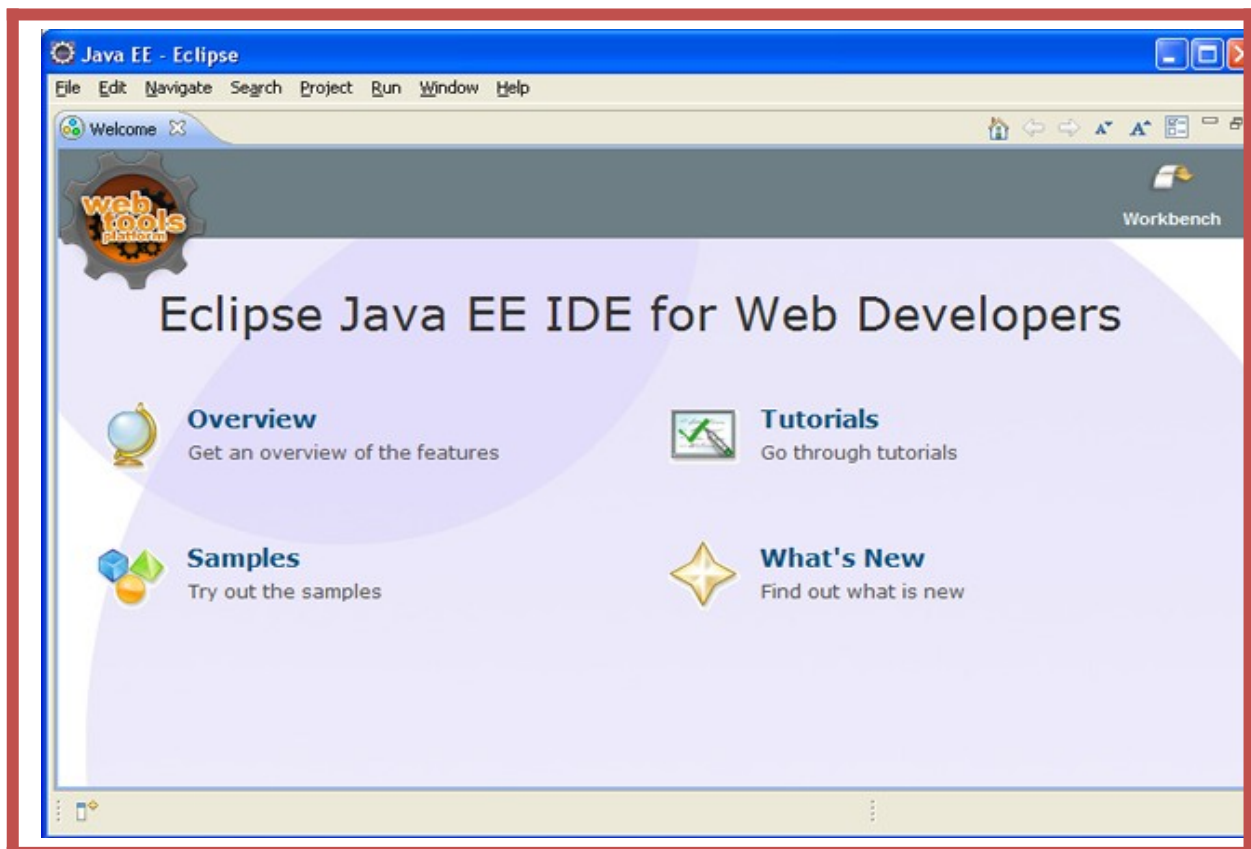


Figure IV.8 : page d'accueil «Eclipse IDE».

IV.1.2. Langages de programmation :

IV.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].

IV.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].

IV.1.2.3. CSS3 et CSS4 :

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].

IV.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 activé côté serveur. L'exécution du code est effectuée par votre navigateur internet tel que Chrome Firefox ou Internet Explorer.

L'une des choses primordiale 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 utile et servent surtout à ajouter un effet esthétique à la page. L'intérêt du JavaScript est d'exécuté un code sans avoir à recharger une nouvelle fois la page.

La technique AJAX (Asynchronous JavaScript And XML) utilise grandement le JavaScript dans le but d'interagir sur la page de manière dynamique. [10].

IV.1.2.5. JQuery :

Jquery est un framework JavaScript sous licence libre qui permet de faciliter des fonctionnalités communes de JavaScript.

L'utilisation de cette bibliothèque permet de gagner du temps de développement lors de l'interaction sur le code HTML d'une page web,

l'AJAX ou la gestion des évènements. JQuery possède par la même occasion l'avantage d'être utilisable sur plusieurs navigateurs web (cf. Internet Explorer, Firefox, Chrome, Safari ou Opéra).

La bibliothèque jQuery possède les fonctionnalités suivantes :

- Manipulation du DOM (HTML ou CSS).
- Gestion des évènements (clic, survol, soumettre un formulaire ...).
- AJAX.
- Effet d'animation.

IV.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 modèle relationnel des SGBDR). [11].

IV.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.

IV.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);

IV.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:

IV.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)	
dateN	Date de naissance de l'utilisateur	Date	
nationalité	Pays de naissance 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 IV.1 : Table «Utilisateur».

IV.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 IV.2 : *Table «Authentification».***IV.2.3.3. Table «Etudiant»:**

Nom de champ	Désignation	Types 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 IV.3 : *Table «Etudiant».***IV.2.3.4. Table «Enseignant»:**

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

Tableau IV.4 : *Table «Enseignant».***IV.2.3.5. Table «Administrateur»:**

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

Tableau IV.5 : *Table «Administrateur».*

IV.2.3.6. Table «Livre» :

Nom de champ	Désignation	Types 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)	
annéeE	L'année d'édition du livre.	Year	
lien	L'url du livre.	Varchar(100)	

Tableau IV.6: Table«Livre».

IV.2.3.7. Table «Message» :

Nom de champ	Désignation	Types 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 IV.7 : Table«Message».

IV.2.3.8. Table «Formation» :

Nom de champ	Désignation	Types 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 IV.8 : Table«Formation».

IV.2.3.9. Table «Module» :

Nom de champ	Désignation	Types 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 IV.9 : Table «Module».

IV.2.3.10. Table «Examen» :

Nom de champ	Désignation	Types 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 IV.10 : Table «Examen».

IV.2.3.11. Table «Contenu» :

Nom de champ	Désignation	Types 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 IV.11 : Table «Contenu».

IV.2.3.12. Table «Question» :

Nom de champ	Désignation	Types 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 IV.12 : Table «Question».

IV.2.3.13. Table «Choix» :

Nom de champ	Désignation	Types 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 IV.13 : Table «Choix».

IV.2.3.14. Table «Réponse» :

Nom de champ	Désignation	Types 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 IV.14 : Table «Réponse».

IV.3. Présentation de quelques interfaces :

En ce qui suit nous allons présenter quelques interfaces de notre application réalisée au profit de l'*INSIM* de Tizi-Ouzou :

IV.3.1. Page d'accueil :



Figure IV.9 : Page d'accueil.

IV.3.2. Interface d'authentification:

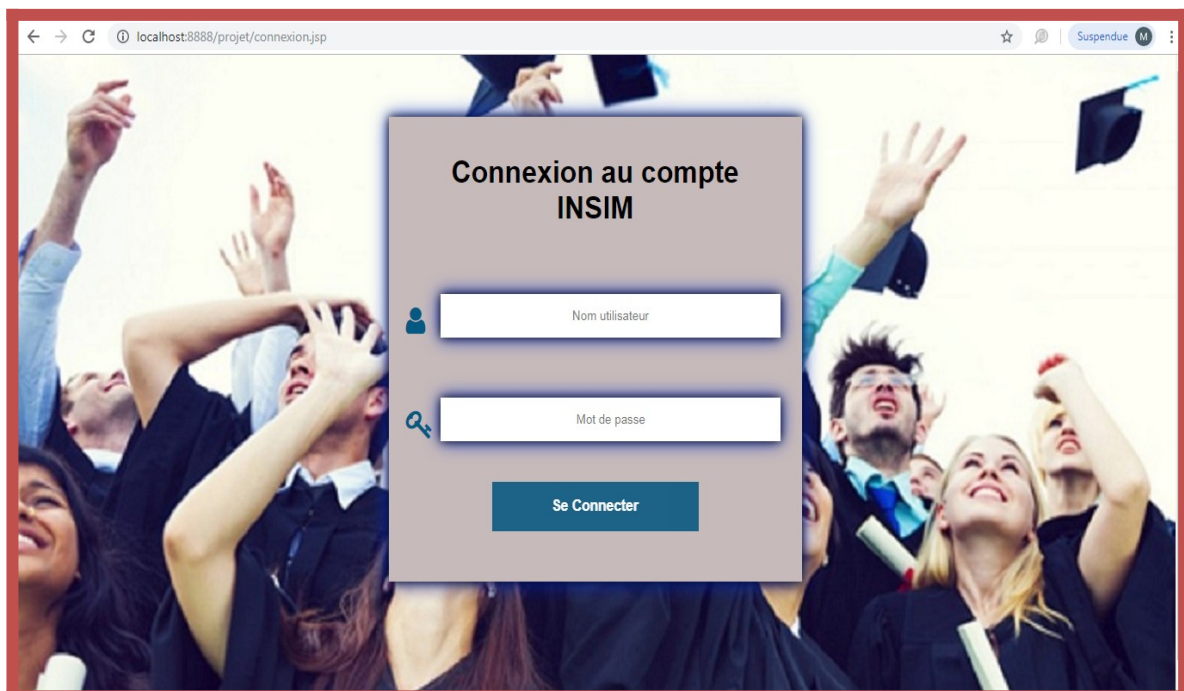


Figure IV.10 : Page «Authentification utilisateur».

IV.3.3. Interface «Ajouter enseignant» :

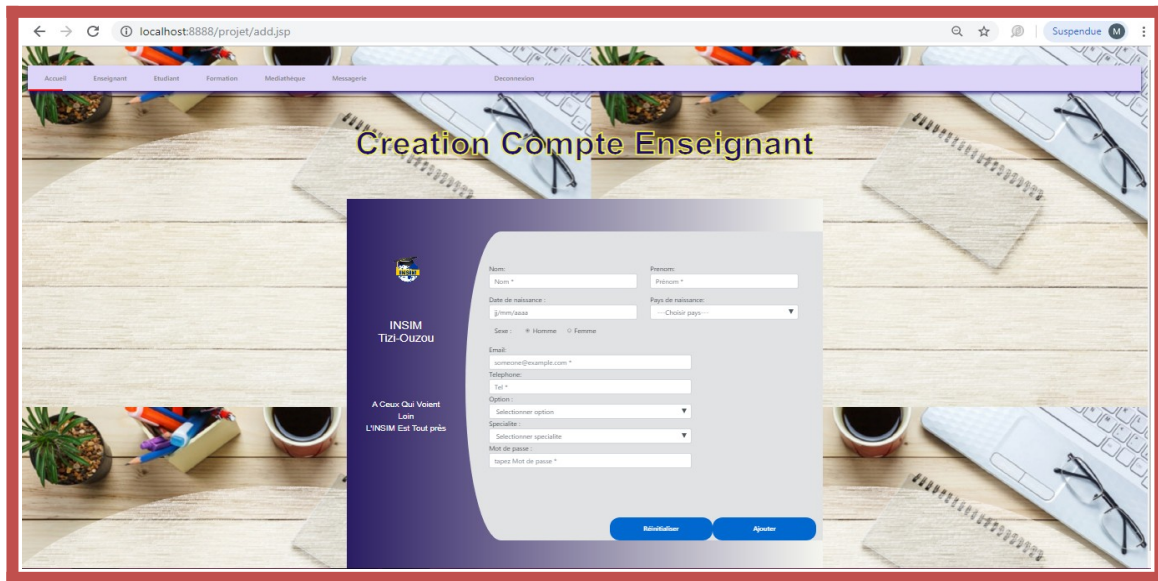


Figure IV.11 : Page «Ajouter enseignant».

IV.3.4. Interface «Ajouter contenu» :

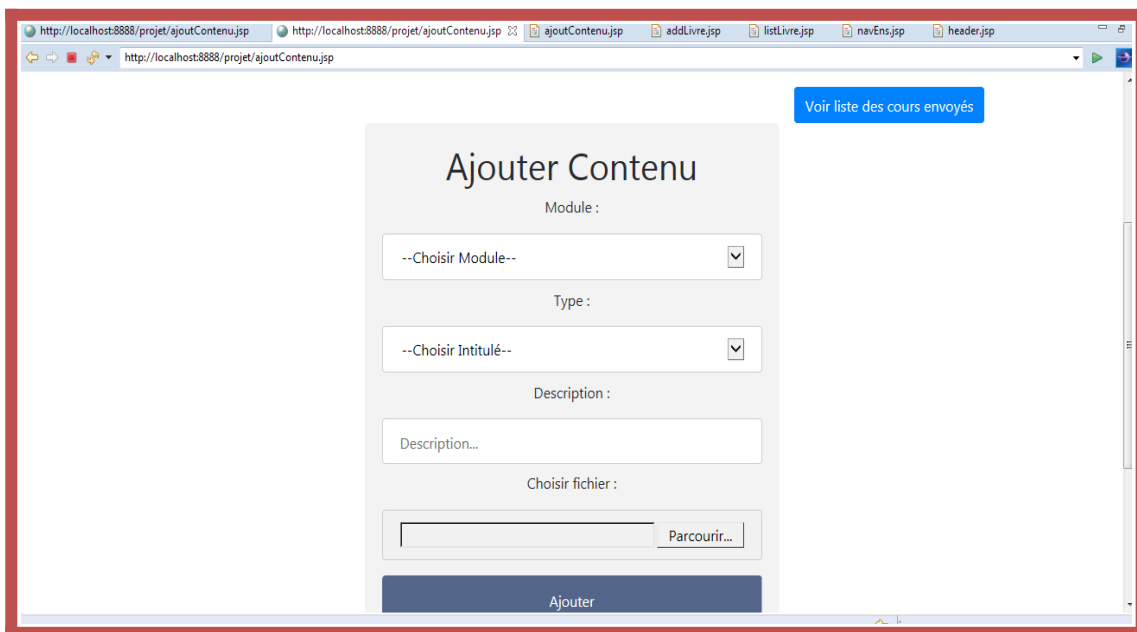


Figure IV.12 : Page «Ajouter contenu».

IV.3.5. Interface «Créer QCM» :



The screenshot shows a web browser window with the URL `localhost:8888/projet/question.jsp`. The page contains a form titled "Ajouter question:". The form includes a "Note" field with a red error message: "L'examen doit être doté de dix (10) questions au juste...!". Below this is a "Question N°:" field with the placeholder text "Intitulé de la question...". There are three "Choix" fields labeled "A", "B", and "C", each with a placeholder "choix N° 1", "choix N° 2", and "choix N° 3" respectively. A "Réponse" field has radio buttons for "A", "B", and "C". At the bottom of the form is a "Note" field and an "Ajouter" button.

Figure IV.13 : Page «Créer QCM».

IV.3.6. Interface «Accueil administrateur» :



Figure IV.14 : Page «Accueil administrateur».

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 au profit de l'institut de formation professionnelle INSIM 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' autres langues dans la plateforme.
- Intégration d'un moteur de recherche par mots clés.
- Mettre en place tout les parcours de formations suivis à l'INSIM.
- 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 :

De nos jours, le génie logiciel nous offre des nouvelles technologies qui ont les potentialités de révolutionner le monde des conceptions des systèmes informatiques, un bon logiciel ne verra jamais le jour sans utiliser une bonne architecture logicielle qui respect le critère de fiabilité, sécurité, portabilité

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

I. Architecture Logicielle :

II.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 ».

II.2. Critères de qualité logicielle :

II.2.1. L'intégrité :

Exprime la faculté du logiciel à protéger ses fonctions et ses données d'accès non autorisés.

II.2.2. La fiabilité :

Exprime la faculté du logiciel à gérer correctement ses propres erreurs de fonctionnement en cours d'exécution.

II.2.3. La maintenabilité :

Exprime la simplicité de correction et de modification du logiciel, et même, parfois, la possibilité de modification du logiciel en cours d'exécution.

II.2.4. La réutilisabilité :

Exprime la capacité de concevoir le logiciel avec des composants déjà conçus tout en permettant la réutilisation simple de ses propres composants pour le développement d'autres logiciels.

II.2.5. L'extensibilité :

Exprime la possibilité d'étendre simplement les fonctionnalités d'un logiciel sans compromettre son intégrité et sa fiabilité.

II.2.6. L'efficacité :

Exprime la capacité du logiciel à exploiter au mieux les ressources offertes par la ou les machines où le logiciel sera implanté.

II.2.7. L'interopérabilité extrinsèque :

Exprime la capacité du logiciel à communiquer et à utiliser les ressources d'autres logiciels comme les documents créés par une certaine application.

II.2.8. L'interopérabilité intrinsèque :

Exprime le degré de cohérence entre le fonctionnement des commandes et des modules à l'intérieur d'un système ou d'un logiciel.

II.2.9. La portabilité :

Exprime la possibilité de compiler le code source et/ou d'exécuter le logiciel sur des plates-formes (machines, systèmes d'exploitation, environnements) différentes.

II.2.10. La compatibilité :

Exprime la possibilité, pour un logiciel, de fonctionner correctement dans un environnement ancien (compatibilité descendante) ou plus récent (compatibilité ascendante).

II.2.11. La validité :

Exprime la conformité des fonctionnalités du logiciel avec celles décrites dans le cahier des charges.

II.2.12. La vérifiabilité :

Exprime la simplicité de vérification de la validité.

II.2.13. L'autonomie :

Exprime la capacité de contrôle de son exécution, de ses données et de ses communications.

II.2.14. La transparence :

Exprime la capacité pour un logiciel de masquer à l'utilisateur (humain ou machine) les détails inutiles à l'utilisation de ses fonctionnalités.

II.2.15. La composabilité :

Exprime la capacité pour un logiciel de combiner des informations provenant de sources différentes.

II.2.16. La convivialité :

Décrit la facilité d'apprentissage et d'utilisation du logiciel par les usagers.

II. Les modèles d'architecture :**III.1. Le modèle des 4+1 vues :**

Le modèle de Kruchten5 dit modèle des 4 + 1 vues est celui adopté dans le processus unifié (Unified Process). 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.

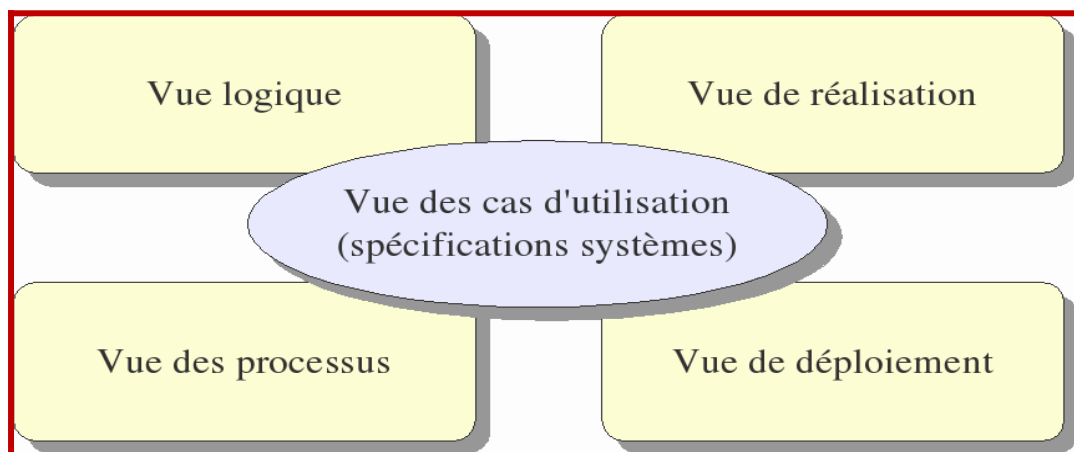


Figure : Modèle d'architecture de Kruchten.

III.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.

III.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 cœur 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.

III.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.

III.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.

III.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.

III. 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 :

- **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 voire DHTML avec l'utilisation du langage client JavaScript.
- **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 :

IV.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.

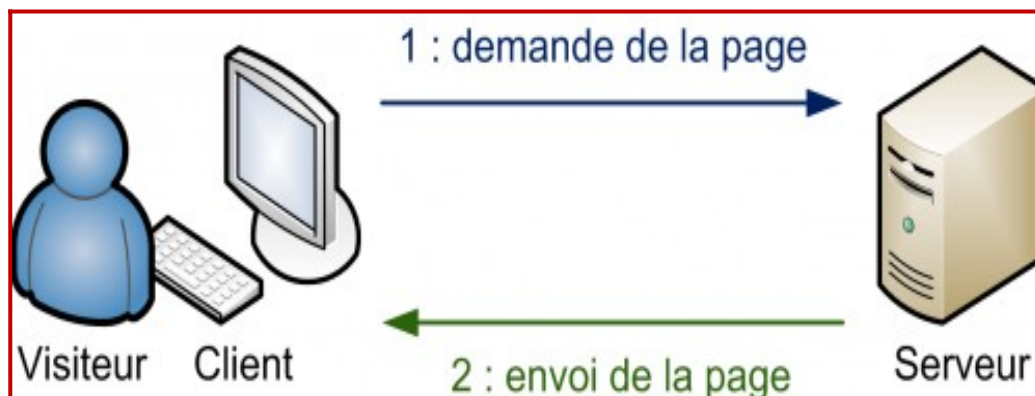


Figure 04 : Fonctionnement de l'architecture Client/Serveur.

IV.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 :

VI.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).

VI.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.

VI.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.

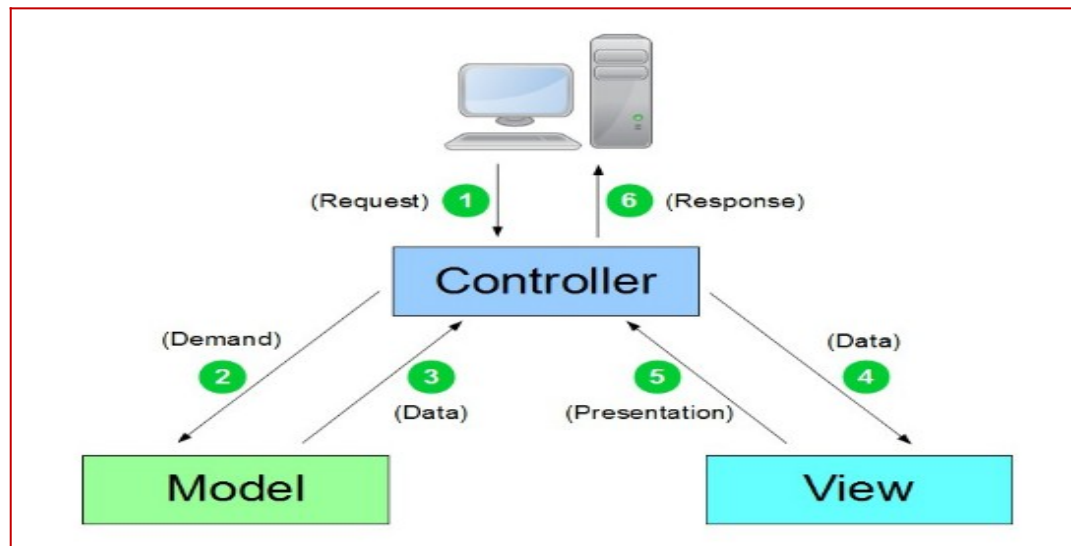


Figure 04 : *Fonctionnement de l'architecture MVC.*

Le contrôleur va demander au modèle les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue.

V. Java Enterprise 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.

V.2. 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 utilisées du code HTML et du code spécifique en général en JAVA.

V.2.1. 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.

V.2.2. 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).

V.2.3. 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.

VI. 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.

VII. 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 œuvre et des exemples d'utilisation
- Des normes de développement
- Des outils facilitant la mise en œuvre.

L'objectif d'un Framework est de faciliter la mise en œuvre 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.

VIII. 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.

IX. 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. Quand 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.

I. 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.

II. 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.

III.1. 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.

III.1.1. 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 :

III.1.1.1. Les acteurs principaux:

Les personnes qui utilisent les fonctions principales du système.

III.1.1.2. Les acteurs secondaires:

Les personnes qui effectuent des tâches administratives ou de maintenance.

III.1.1.3. Le matériel externe:

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

III.1.1.4. Les autres systèmes:

Les systèmes avec lesquels le système doit interagir.

III.1.2. 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.

III.1.2.1. 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.

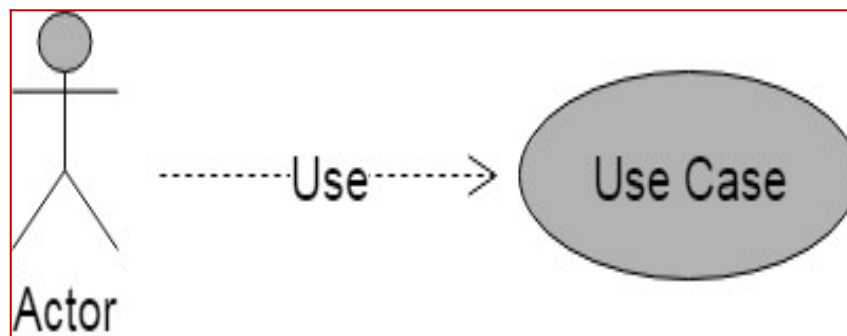


Figure 01 : *Relation généralisation.*

❖ 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).

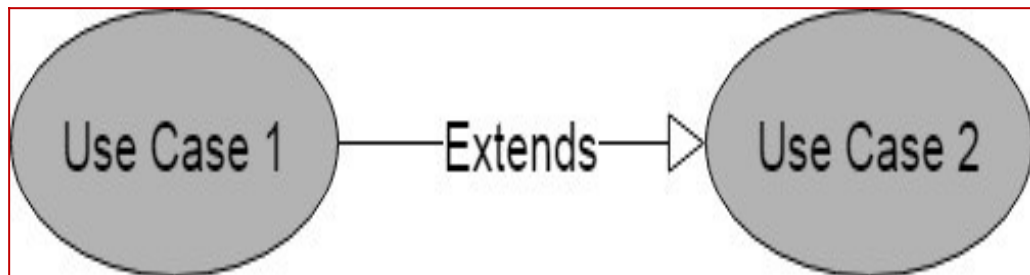


Figure 02 : *Relation d'extension.*

❖ **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.

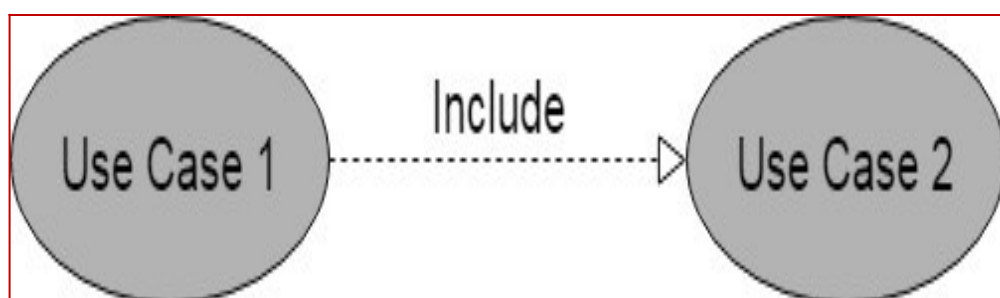


Figure 03 : *Relation d'inclusion.*

III.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.

III.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.

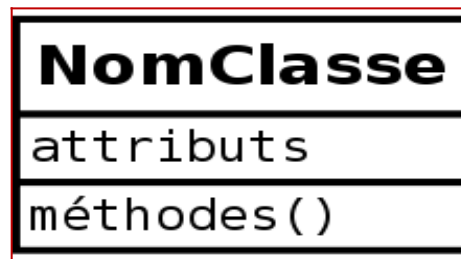


Figure 03 : représentation d'une classe UML.

III.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-aire, les classes sont reliées à un losange par des traits simples. Ces relations peuvent être nommées.

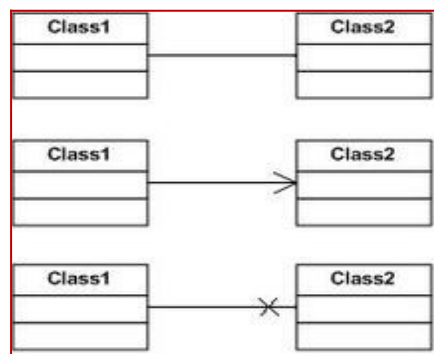


Figure 03 : représentation d'Association UML.

III.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.

III.3.1. Message synchrone et message asynchrone :

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

III.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.

III.3.2. 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.

III.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 .


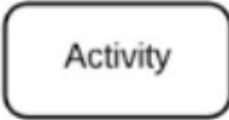

Symbole	Description
	Symbole de début : Représente le début d'un processus ou d'un flux de travail dans un diagramme d'activités. Il peut être utilisé seul ou avec un symbole de note qui explique le point de départ.
	Symbole d'activité : Indique les activités qui composent un processus modélisé. Ces 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.

Tableau 01 : Symboles d'un diagramme d'activité.

Conclusion :

Cette annexe a proposé un survol et une présentation brève des principaux concepts de modélisation UML ainsi que leur notation.

Références :

Références bibliographique :

- I. [1] : ODILE F, « Territoiles : les rencontres territoriales de la e-formation », Février 2011.
- I. [3] : « Avis sur la formation à distance », fédération étudiante universitaire du Québec, Année 2009, article
- IV. [1] : mémoire fin d'études Aouam Djamel & Chabane Hocine, «Conception et réalisation d'une plateforme E-learning», promotion 2010-2011.

Références webliographique :

- I. [2] : Définition de l'union européenne de l'E-learning.
- I. [4]: MOOC: https://fr.wikipedia.org/wiki/Massive_Open_Online_Course.
- I. [5]: <https://www.definitions-marketing.com/definition/tracking-internet>.
- I. [6]: <https://fr.wikipedia.org/wiki/Claroline>.
- I.[7]:
<http://sarremia.com/blog/index.php/ganesha-plate-forme-e-learning-scorm-2004-opensource/73>.
- I. [8]: <https://fr.wikipedia.org/wiki/Moodle>.
- III. [1] : https://www.memoireonline.com/02/09/1973/m_conception-et-developpement-dune-application-de-la-gestion-dune-bibliotheque5.html
- III. [2] : Cas d'utilisation : https://fr.wikipedia.org/wiki/Diagramme_des_cas_d%27utilisation.
- IV. [2] : https://fr.wikipedia.org/wiki/Apache_Tomcat.
- IV. [3] : <https://fr.wikipedia.org/wiki/WampServer>.
- IV. [4] : http://www-igm.univ-mlv.fr/~dr/XPOSE2003/alexandrebole/jboss_1.html
- IV. [5] : https://doc.fedorafr.org/wiki/PhpMyAdmin:gestion_de_vos_bases_de_donn%C3%A9es_MySQL.
- IV. [6] : <https://www.techno-science.net/definition/517.html>.
- IV. [7] : [https://fr.wikipedia.org/wiki/Java_\(langage\)](https://fr.wikipedia.org/wiki/Java_(langage))
- IV. [8] : https://fr.wikipedia.org/wiki/Hypertext_Markup_Language
- IV. [9] : <https://www.futura-sciences.com/tech/definitions/internet-css-4050/>
- IV. [10] : <http://glossaire.infowebmaster.fr/javascript/>
- IV. [11] : <http://dictionnaire.sensagent.leparisien.fr/Mod%C3%A8le%20relationnel/fr-fr/>
- IV. [12] : https://fr.wikipedia.org/wiki/Mod%C3%A8le_physique_des_donn%C3%A9es