

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET LA RECHERCHE SCIENTIFIQUE
Université MOULOUD MAMMERI de Tizi-Ouzou
Faculté de Génie Electrique et Informatique
DEPARTEMENT INFORMATIQUE



MEMOIRE

De fin d'études

En vue de l'obtention du diplôme de
Master en Informatique

Thème

**Conception et réalisation d'une application web pour
la gestion de la scolarité**

Dirigé par :

Mr. A.DIB

Réalisé par :

TALEB YAZID

ALILECHE YAMINA

Promotion 2014-2015

Remerciement

Nous remercions le bon dieu pour le courage, la patience qui nous ont été utiles tout au long de notre parcours.

Nous tenons à exprimer notre profonde gratitude à notre promoteur, Monsieur A. Dib pour nous avoir encadrés et guidés tout au long de notre projet, pour ses conseils judicieux et minutieusement prodigués. Aussi nous tenons à lui reconnaître le temps précieux qu'il nous a consacré.

Les membres du jury pour l'honneur qu'il nous fait en acceptant de juger notre travail.

Ces remerciements ne seraient pas complets si on n'a pas pensé à les destiner, avec notre profonde reconnaissance, à nos parents qui nous ont offert un environnement favorable pour mener à terme notre travail.

Enfin nous remercions tous ceux qui ont contribués de près ou de loin afin que notre travail puisse voir le jour.

Dédicaces

Je dédie ce modeste travail à :
A la mémoire de mon grand frère Hakim
A mes très chers parents qui mon beaucoup
soutenu durant mon travail.
A mes frères et sœurs , neveu et nièce
A mes Proches
A tous mes Amis(es)
A toutes la promotion d'Informatique
2014/2015 spécialement amine

« Yazid »

Dédicaces

Je dédie ce modeste travail à :

A mes très chers parents.

A la Mémoire de ma Sœur : Nadia.

A mes frères et sœurs

A tous mes Amies : Amine, Zohra, Lycia

A mon Binôme : Yazid

« Yamina »

Sommaire

Introduction générale.....	1
Chapitre I : L'organisme d'accueil	
Introduction.....	2
I. Définition de l'université.....	2
II. L'université Mouloud Mammeri.....	2
II.1. Présentation de l'université Mouloud Mammeri (UMMTO).....	2
II.2. Histoire de l'UMMTO.....	2
II.3. Le système organisationnel.....	3
II.3.1. Les organes centralisés.....	4
➤ Conseil d'orientation.....	4
➤ Conseil scientifique de l'université.....	4
➤ Le rectorat.....	4
II.3.2. Les organes décentralisés.....	8
II.4. Les facultés de l'UMMTO.....	8
III. L'organisme d'accueil..... ;	9
III.1. Présentation de l'organisme d'accueil.....	10
III.2. Les services.....	11
1. Le service d'enseignement et de la scolarité.....	11
2. Le service de la recherche scientifique.....	11
III.3. Missions de département.....	11
III.4. Suivi des programmes d'enseignement.....	11
1. Un cursus.....	12
2. Formations.....	12
3. Système LMD.....	12
a. Licence.....	12
b. Master.....	12
c. Doctorat.....	13
4. Unités d'enseignement(UE).....	13
5. Matières enseignées.....	14
6. Suivi pédagogique des enseignants	14
Conclusion.....	14

Chapitre II : Généralités sur les réseaux, Internet et Web

Introduction.....	15
I. Les réseaux informatiques.....	15
I.1. Définition d'un réseau informatique.....	15
I.2. Objectif des réseaux	15
I.3. Classification des réseaux informatiques.....	15
a. Selon l'envergure.....	15
b. Selon la topologie.....	16
I.4. Architecture des réseaux.....	19
a. L'architecture OSI (Open System Inter connexion).....	19
b. L'architecture TCP/IP (Transmission Control Protocol / Internet protocol)....	21
I.5. Définition de l'architecture client /serveur.....	24
I.5.1. Caractéristiques du modèle client/serveur.....	24
I.5.2. Technique du dialogue Client /Serveur.....	26
I.5.3. Architecture client/serveur multi niveaux.....	26
I.5.4. Les types de serveurs.....	29
I.5.5. Le fonctionnement du modèle client/serveur.....	30
II. Internet.....	31
II.1 Définition de l'Internet.....	31
II.2. Histoire de l'Internet.....	31
II.3. Principaux services offerts par l'Internet.....	32
III. Le web.....	33
III.1. Définition.....	33
III.2. Le principe du Web.....	33
III.3. Histoire de web.....	33
III.4. L'évolution de web.....	34
III.5. Les concepts de web.....	35
III.5.1. Les pages Web.....	35
III.5.2. Un site Web (ou site Internet).....	35
III.6. Les principales caractéristiques du web.....	36
III.7. Les standards de web.....	37

III.7.1. URL (Universal/Uniform Resource Locator).....	37
III.7.2. Le langage HTML.....	37
III.7.3. Le protocole HTTP	38
III.8. Communication entre navigateur et serveur.....	39
Conclusion.....	40
Chapitre III : Framework	
Introduction.....	41
I.C'est quoi un Framework ?.....	41
II. Utilisation d'un framework	41
II.1. Inconvénients d'utilisation d'un framework.....	42
III. Quelques critères objectifs dans le choix d'un framework.....	42
IV. Les différents types de frameworks	43
1. Framework d'infrastructure système	43
2. Framework d'intégration intergicielle (middleware).....	43
3. Framework d'entreprise	43
4. Framework de gestion de contenu.....	43
V. Quelques caractéristiques que peut fournir un framework.....	43
VI. Un Framework PHP.....	44
VI.1. Définition	44
➤ Modèle MVC.....	44
➤ Templates	44
➤ Cache	44
➤ Gestion des SGBDR.....	44
➤ ORM.....	44
➤ Scaffolding.....	44
➤ Conventions.....	44
➤ URL conviviales.....	44

VI.2. Critères du meilleur framework PHP.....	45
VI.3. Les différents frameworks existants.....	46
➤ CakePHP	46
➤ CodeIgnite.....	46
➤ Symfony	46
➤ Zend Framework.....	47
VI.4. Comparaison des framework php.....	47
VI.5. Les avantages d'un Framework PHP.....	48
1. La pérennité.....	48
2. Vitesse de développement..... ;.....	49
3. Un support étendu.....	49
VII. Le meilleur framework est celui que vous choisissez	49
VII.1.Définition de Zend framework	49
VII.2.Les principales caractéristiques de Zend Framework	49
VII.3.Avantages Zend Framework.....	50
Conclusion	51

Chapitre IV : Analyse et conception

Introduction.....	52
I. Présentation de l'UML.....	52
I.1. Définition..... ;.....	52
I.2. Extension de l'UML pour le web.....	53
I.3.La modélisation UML	53
I.3.1. Les diagrammes UML.....	53
I.4 Diagramme de contexte.....	55
II. Analyse	56

II.1. Spécification des besoins.....	56
II.2. Identification des acteurs.....	56
a-Définition d'un acteur.....	56
b-Extraction des acteurs de notre système..... ;	56
II. 3. Identification des cas d'utilisation.....	56
II.3.1. Définition.....	56
II.3.2. Spécification des tâches.....	57
II.4. Spécification des scénarios	58
II.5. Spécification de quelques cas d'utilisation	61
III. La conception.....	64
III.1. Diagrammes de cas d'utilisation.....	64
III.2. Diagrammes de séquence de réalisation des cas d'utilisateur	68
III.3. Diagrammes d'activités	73
III.4. Diagrammes de classe	79
III.5. Conception de la base de données.....	82
III.5.1. Les tables de la base de données.....	82
a. Le niveau logique de la base de données.....	82
b. Le niveau physique de données	83
Conclusion.....	86

Chapitre V : Réalisation et implémentation

Introduction.....	87
I. Environnement de développement	87
II. Outils utilisés.....	88
II.1. Langages utilisés	88
II.1.1.HTML	88
II.1.2. CSS.....	88
II.1.3. PHP	88
II.1.4. SQL	89
II.1.5. Java.....	89
II.1.6.Java Script.....	89

II.2. Logiciels utilisés	90
II.2.1. WampServer	90
II.2.2. Dreamweaver.....	91
III. Présentation de quelques interfaces	92
III.1. Page d'accueil.....	92
III.2. La page Authentification.....	93
III.3. Page d'ajout d'un module	93
III.4. Page d'ajout d'une spécialité	94
III.5. Page d'affectation d'un enseignant.....	94
Conclusion	95

Chapitre I : L'organisme d'accueil

Figure I.1. Organigramme du Rectorat UMMTO.....7
Figure I.2. Organigramme général de département informatique.....9

Chapitre II : Généralités sur les réseaux, Internet et Web

Figure II.1 : Classification des réseaux informatiques selon l'envergure.....16
Figure II.2. Topologie en étoile.....17
Figure II.3. Topologie en bus.....17
Figure II.4. Topologie en anneau.....18
Figure II.5. Topologie maillé.....19
Figure II.6 Les couches du modèle OSI.....20
Figure II.7 La pile de protocole TCP/IP avec sa correspondance OSI.....22
Figure II.8. Le dialogue client/serveur.....24.
Figure II.9. Architecture client/serveur deux niveaux.....27
Figure II.10 : Architecture client/serveur trois niveaux.....28

Chapitre III : Framework

Figure III.1.Les différents types des frameworks.....46
Figure III.2.La courbe de recherche relative aux frameworks.....47
Figure III.3. : Tableau comparatif des 5 Frameworks.....48

Chapitre IV : Analyse et conception

Figure IV.1. La démarche de modélisation et application.....52
Figure IV.2. Les trois composantes de la modélisation UML.....54
Figure IV.3. Diagramme de contexte de l'application55
Figure IV.4 : Diagramme global de cas d'utilisation relatif à l'étudiant.....65
Figure IV.5 : Diagramme global de cas d'utilisation relatif à l'enseignant.....66

Figure IV .6 : Diagramme global de cas d'utilisation relatif à l'administrateur.....	67
Figure IV.7 : Diagramme de séquence de réalisation de cas d'utilisation « Authentification étudiant ».....	69
Figure IV .8 : Diagramme de séquence de réalisation de cas d'utilisation « enregistrement des notes ».....	70
Figure IV.9 : Diagramme de séquence de réalisation de cas d'utilisation « Changement de mot de passe ».....	71
Figure IV.10 : Diagramme de séquence de réalisation de cas d'utilisation « Ajout d'un module ».....	72
Figure IV.11 : Diagramme d'activités du d'utilisation « authentification étudiant ».....	74
Figure IV.12 : Diagramme d'activités du cas d'utilisation « Consulter les annonces ».....	75
Figure IV.13 : Diagramme d'activités du d'utilisation « enregistrement des notes des étudiant ».....	76
Figure IV.14 : Diagramme d'activités d'utilisation « changement de mot de passe »	77
Figure IV.15 : Diagramme d'activités du cas d'utilisation « Ajout d'un module ».....	78
Figure IV.16. Diagramme de classe général du cas d'utilisation «Authentification ».....	79
Figure IV.17. Diagramme de classe général du cas d'utilisation « Ajouter un étudiant ».....	80
Figure IV.18. Diagramme de classe général du cas d'utilisation.....	80
« Changer mot de passe»	
Figure IV.19 : Diagramme de classe global.....	81

Chapitre V : Réalisation et implémentation

Figure V.1. Interface de WampServer.....	90
Figure V.2. Interface de Dreamweaver CC.....	91
Figure V.4. Page d'accueil.....	92
Figure V.5 Page authentification.....	93
Figure V.6. Ajout d'un module.....	93
Figure V.7. Ajout d'une spécialité	94
Figure V.8. Affectation d'un enseignant.....	94

Introduction
générale

Il ne fait désormais plus aucun doute que l'informatique représente la révolution la plus importante qui a marqué la vie de l'humanité en ce siècle passé, jusqu'à nos jours.

En effet, l'informatique vient nous apporter de multiples confort à notre mode de vie. Cette technologie touche tous les domaines d'activités et apporte tant de services aussi bien pour l'entreprise ou l'administration que pour le personnel.

Au cours de ces dernières années, les nouvelles technologies de l'information et de la communication ont connu un bouleversement marqué par l'apparition de l'Internet et par sa croissance exponentielle.

Ces années ont aussi été marquées par l'entrée en scène de World Wide Web (WWW), l'une des applications qui a popularisé l'Internet, et qui a fait exploser le nombre de ses utilisateurs.

Cette évolution a aussi touchée l'univers des études et en plus particulier le monde de l'université.

C'est dans ce cadre d'idées que s'inscrit notre projet de fin d'études qui consiste à la conception et la réalisation d'un site web pour la gestion de scolarité.

Afin de mener à bien notre projet, nous avons reparti le contenu de notre travail en quatre chapitres comme suit :

- Le premier chapitre définit l'organisme d'accueil qui est le département informatique.
- Le second chapitre est consacré aux nouvelles technologies de l'information et de la communication.
- Le troisième chapitre est dédié à l'analyse et la conception de notre site.
- Le quatrième est dédié à la réalisation de notre site.

Chapitre I

Organisme d'accueil

Introduction :

L'une des missions principales de l'université est de contribuer au développement du capital humain et de diffuser le savoir. Pour y arriver, elle doit se prévaloir de la diversité de son offre de formation en équilibre avec les besoins des divers secteurs de la vie économique et du développement technologique.

Dans ce chapitre on présente l'université (UMMTO) en premier et le département informatique en particulier qui représente l'organisme d'accueil.

I. Définition de l'université :

Une université est un établissement qui fédère en son sein la production, la conservation et la transmission de différents domaines de la connaissance. Le mot université vient du latin, il désigne à l'origine la corporation des enseignants et des étudiants d'une même ville. [A] Les universités sont aujourd'hui divisées en département académiques, écoles ou facultés, qui sont centrées sur la seule transmission d'un domaine bien défini de la connaissance. L'université est sans doute le lieu où l'étudiant se spécialise dans un champ du savoir.

II. L'université Mouloud MAMMARI (UMMTO) :**II.1. Présentation de l'université UMMTO :**

L'université Mouloud Mammeri UMMTO, est une université algérienne située dans la ville de Tizi-Ouzou en Kabylie. Elle porte le nom de Mouloud Mammeri un écrivain, poète, anthropologue et linguiste algérien.

L'université Mouloud Mammeri de Tizi-Ouzou est un établissement public à caractère scientifique, culturel et professionnel doté d'une personnalité morale et d'une autonomie financière, dépendant du secteur de la fonction publique. Sa mission est d'assurer la formation en gradation et en post-graduation dans divers domaines et spécialités, en plus de la recherche scientifique.

II.2. Histoire de l'UMMTO : [B]

L'université de Tizi-Ouzou est créée en 1977 sous forme de centre universitaire rattaché à l'université d'Alger. En 1989 le C.U.T.O devient une université à part entière. Quelque 490 étudiants y firent leur entrée en 1977 mais ce nombre ne cesse de croître année après année pour atteindre en 2009 quelque 42000 étudiants.

En 1984, le centre universitaire de Tizi-Ouzou éclate en 9 instituts :

- INES des Sciences Juridiques et Administratives
- INES d'Agronomie
- INES de Biologie
- INES des Lettres et Littérature Arabes
- INES de Génie Civil
- INES des Sciences Economiques
- INES des Sciences Médicales
- INES d'Electronique
- INES d'Informatique

En 1989, cinq nouveaux départements y sont créés :

- Département d'Architecture
- Département d'Electronique
- Département des Langues Etrangères
- Département des Sciences Exactes
- Département de Génie Mécanique

En 1991, les quatre premiers départements cités deviennent des instituts. Quant au cinquième, il n'est érigé en institut qu'en 1995.

En 1990, le département des langues et cultures amazigh a vu le jour par arrêté ministériel.

L'université compte actuellement 9 facultés réparties sur plusieurs sites notamment Boukhalfa, Hasnaoua, Bastos, Tamda

II.3. Le système organisationnel :

L'université est dotée d'organes centralisés et des organes décentralisés prévus par le statut type de l'université.

II.3.1. Les organes centralisés :**➤ Le conseil d'orientation :**

Est composé de tous les représentants des ministères, des doyens de facultés, du secrétaire général de l'université, du responsable de la bibliothèque universitaire et des représentants des trois composantes de l'université les enseignants, les travailleurs et les étudiants.

➤ Le conseil scientifique de l'université :

Est composé du recteur de l'université, des vice-recteurs, des doyens des facultés, des présidents des conseils scientifiques des facultés, un représentant élu des enseignants par faculté, le secrétaire général de l'université et le responsable de la bibliothèque universitaire.

➤ Le rectorat :

Il est placé sous l'autorité du recteur, il comprend quatre vices rectorats et le secrétariat général, qui est organisé comme suit

1. Vice Rectorat de la formation supérieure de graduation, de la formation continue et des diplômes
 - Suivre les questions se rapportant au déroulement des enseignements et des stages organisés par l'université
 - Veiller à la cohérence des offres de formation présentées par les facultés et instituts avec le plan de développement de l'université.
 - Veiller au respect de la réglementation en vigueur en matière d'inscription, de réinscription, de contrôle des connaissances et de progression des étudiants.
 - Suivre les actions de formation à distance assurée par l'université et promouvoir les activités de formation continue.
 - Veiller au respect de la réglementation et de procédure en vigueur en matière de délivrance de diplôme et d'équivalence.

2. Vice Rectorat de la formation supérieure de post-graduation, de l'habilitation universitaire et de la recherche scientifique
 - Suivre les questions liées au déroulement des formations de post-graduation spécialisée et à l'habilitation universitaire et veiller à l'application de la réglementation en vigueur en la matière.
 - Suivre les activités de recherche des unités et laboratoires de recherches et élaborer le bilan, en coordination avec les facultés et instituts.
 - Mener toute action de valorisation des résultats de la recherche.

- Assurer le suivi du fonctionnement du conseil scientifique de l'université et conserver les archives.
 - Collecter et diffuser les informations sur les activités de recherche menées par l'université.
3. Vice Rectorat des relations extérieures, de la coopération, de l'animation et de la communication et des manifestations scientifique
- Promouvoir les relations de l'université avec son environnement socio-économique et d'initier des programme de partenariat.
 - Initier toute action de promotion des échanges interuniversitaires et de la coopération dans les domaines de l'enseignement et de la recherche.
 - Organiser et promouvoir les manifestations scientifiques.
 - Assurer le suivi des programmes de perfectionnement et de recyclage des enseignants.
4. Vice Rectorat du développement, de la prospective et de l'orientation
- Réunir les éléments nécessaires à l'élaboration des projets de plans de développement de l'université.
 - Effectuer toute étude prospective sur les prévisions d'évolution des effectifs étudiants de l'université et proposer toute mesure pour leur prise en charge notamment en matière d'évolution d'encadrement pédagogique.
 - Procéder à l'élaboration de tout support d'information avant les aider dans leur choix d'orientation.
 - Entreprendre des actions d'information des étudiants.
 - Suivre les programmes de construction et assurer la mise en œuvre des programmes d'équipement de l'université en relation avec les services concernés.
- 5- Secrétariat Générale
- Assurer la gestion des carrières des personnels de l'université dans le respect des attributions de la faculté et de l'institut.
 - Préparer le projet du budget de l'université et d'en suivre l'exécution.
 - Assurer le suivi du financement des activités des laboratoires et des unités de recherche.
 - Veiller au bon fonctionnement des services communs de l'université.
 - Mettre en œuvre les programmes d'activités culturelles et sportives de l'université et de les promouvoir.
 - Assurer le suivi et la coordination des plans de sûreté interne de l'université en relation avec le bureau ministériel de sûreté interne.

- Assurer la gestion et la conservation des archives et de la documentation du rectorat.

Le secrétariat général, comprend quatre (04) sous direction :

- La sous direction des personnels et de la formation.
- La sous direction du budget et de la comptabilité.
- La sous direction des moyens et de la maintenance.
- La sous direction des activités scientifique, culturelles et sportives.

❖ Organigramme du Rectorat UMMTO :

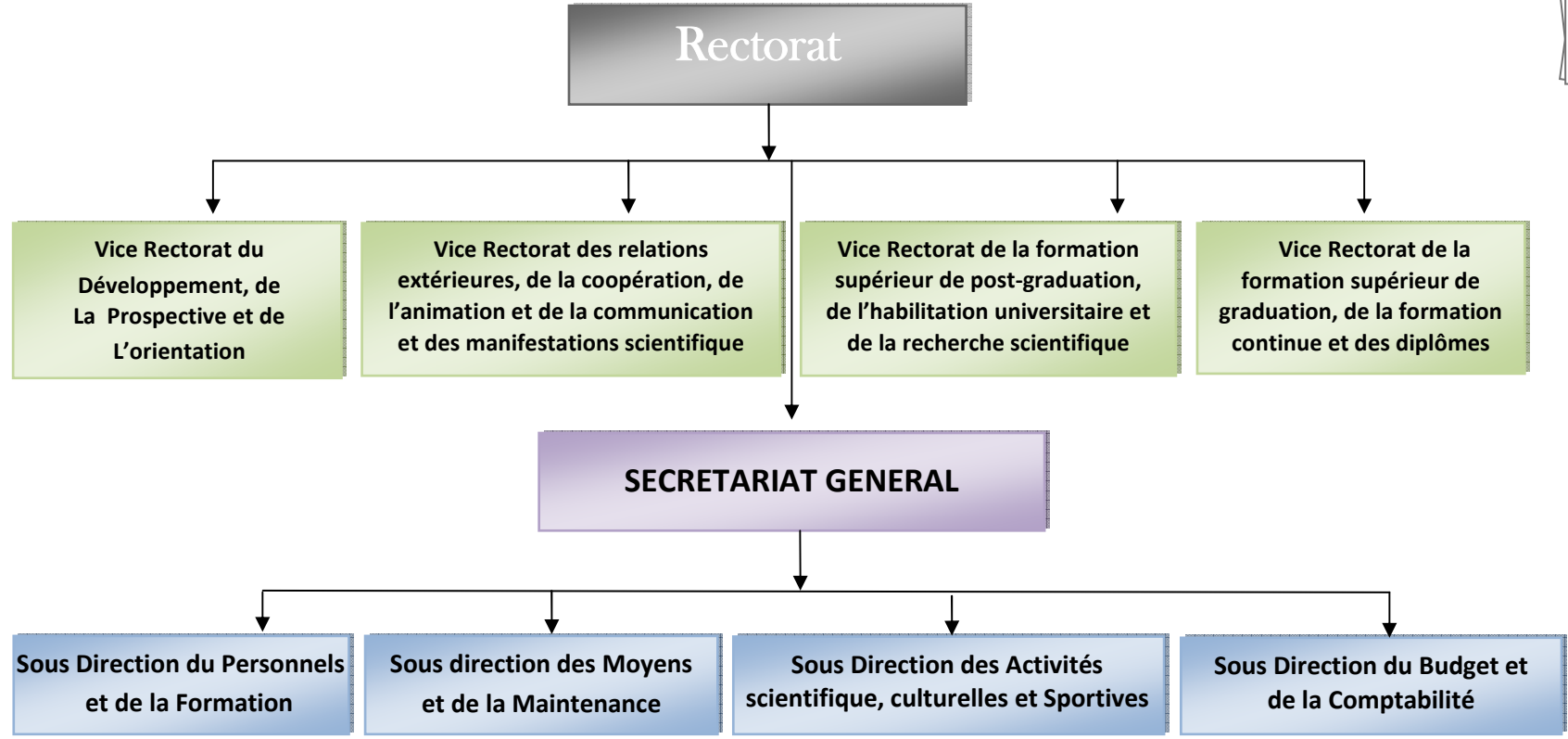


Figure I.1. Organigramme de l'UMMTO

II.3.2. Les organes décentralisés : Les facultés

La faculté est une unité d'enseignement et de recherche de l'université dans le domaine de la science et de la technologie, elle est composée de deux vice-doyens et du secrétariat général

La faculté est composée de départements et comporte une bibliothèque.

Le doyen est responsable du fonctionnement de la faculté et assure la gestion de ses moyens humains, financiers et matériels.

Il est assisté dans ses tâches par le vice-doyen chargé des études et des questions liées aux étudiants, le vice-doyen chargé de la post-graduation, de la recherche scientifique et des relations extérieures, le secrétaire général de la faculté, les chefs des départements et le responsable de la bibliothèque de la faculté.

Remarque :

La structure organisationnelle des différentes facultés est la même, pour toutes les facultés.

II.4. Les facultés de l'UMMTO :

L'université comprend neuf facultés et plusieurs départements à savoir :

- La faculté des sciences sociales et humaines.
- La faculté de médecine rassemble trois types de formations : doctorat en médecine, doctorat en chirurgie dentaire, doctorat en pharmacie.
- La faculté des sciences économiques et des sciences commerciales et des sciences de gestion regroupe trois types de formations à savoir : sciences économiques, sciences de gestion et sciences commerciales.
- La faculté des sciences biologiques et des sciences agronomiques comprennent deux filières (biologie et agronomie)
- La faculté de génie de la construction rassemble trois filières : génie mécanique, génie civil et architecture.
- La faculté de droit et sciences politiques assure une formation de deux types : droit et sciences politiques.
- La faculté des sciences est composée de quatre départements à savoir : département de physique, département des mathématiques, département de chimie, département tronc commun LMD.

- La faculté des lettres et des langues est composée de sept branches (anglais, français, traduction interprétariat, langue et culture amazighes, lettres arabes, psychologie, sciences de l'éducation, sciences sociales).
- La faculté de génie électrique et informatique est composée de quatre départements a savoir : département d'électrotechnique, département d'automatique et département d'électronique.

III. L'organisme d'accueil : département informatique

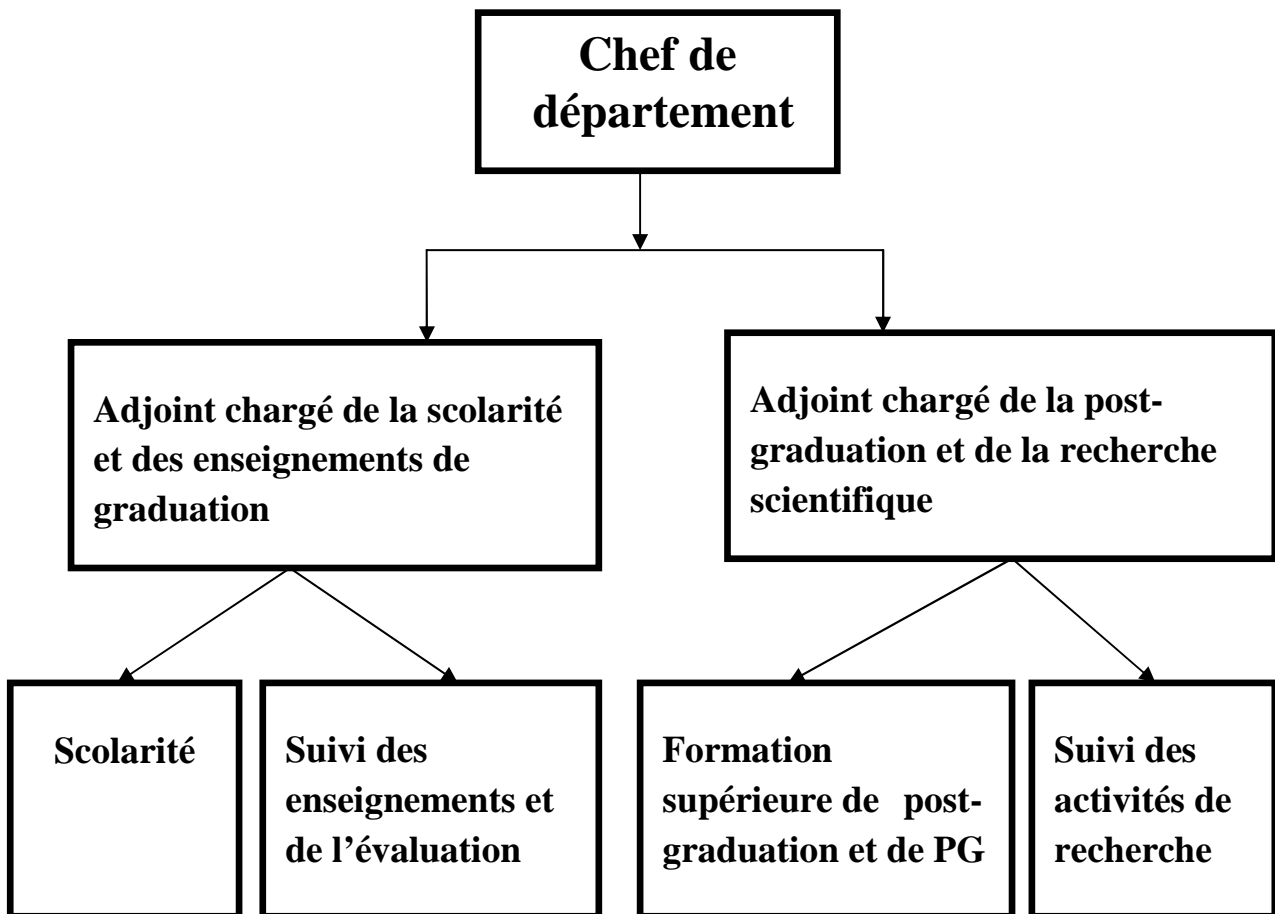


Figure I.2. Organigramme général du département informatique

III.1. Présentation de l'organisme d'accueil :

Le département d'informatique est l'un des départements géré par la faculté de génie électrique et informatique de l'UMMTO sous la direction d'un doyen, et par un chef de département, assisté par d'autres responsables chacun d'eux avec ses propres fonctions par rapport à son service.

* Le département est chargé d'assurer la programmation, la réalisation, l'évaluation et le contrôle des activités d'enseignement et de recherche dans son domaine de formation.

* Le chef de département est responsable du fonctionnement pédagogique et administratif du département.

* Il est assisté par le chef de département adjoint chargé de la scolarité et des enseignement de graduation, et le chef de département adjoint chargé de la post-graduation et de la recherche scientifique.

Le chef de département a pour taches notamment :

- de veiller à l'application et au respect de la réglementation en vigueur concernant toutes les activités universitaires.
- de préparer le planning des activités pédagogiques et d'assurer leur répartition entre les enseignants.

Le planning des activités pédagogiques comprend, notamment :

- les emplois du temps des différents enseignements dispensés et l'utilisation locaux d'enseignement.
- les activités pédagogiques du personnel enseignant et l'élaboration des documents et supports pédagogiques.
- la mise en place des comités pédagogiques et le suivi de leurs activités : déroulement des études et programmes pédagogiques, assiduité des étudiants et des enseignants.
- l'organisation et le bon déroulement des différentes épreuves d'évaluation des étudiants.
- la préparation des procès-verbaux de délibération, la tenue et le fonctionnement des jurys de délibération.

III.2. Les services :

On trouve au sein de ce département deux services qui sont :

1. **Le service d'enseignement et de la scolarité** sous la direction d'un chef adjoint à la pédagogie. Ce service assure la gestion des études et toutes les tâches concernant les enseignants et les étudiants du département.
2. **Le service de la recherche scientifique** sous la direction d'un chef de département adjoint à la recherche scientifique. Il gère les enseignants chercheurs et les étudiants en post-graduation. Les enseignants chercheurs organisent des réunions de travail pour exposer et étudier des thèmes de recherche, suivent les nouvelles technologies et animent des conférences pour l'information des étudiants.

Les activités reliées au suivi pédagogique dans le département de l'informatique sont :

- Participe aux délibérations des jurys d'examen
- Participe aux travaux de son équipe ou de son comité pédagogique
- Reçoit les étudiants pour les conseiller et les orienter
- Assure le bon déroulement des examens dont il a la charge
- Assure l'encadrement des étudiants
- Assure les travaux dirigés ou les travaux pratiques selon le volume horaire
- Corrige les copies des examens dont il a la charge

III.3. Missions du département :

Le département assure un suivi pédagogique des cycles de graduation et de post-graduation ; gère la scolarité des étudiants (inscription, évaluation, présence à l'enseignement, absences et sanctions) et des enseignants (matières enseignées, affectation des modules, volume horaire, emploi du temps, planning des examens, absences, saisie des notes, délibérations.....)

III.4. Suivi des programmes d'enseignement :

Le département consacre un temps énorme pour le suivi des programmes d'enseignement pour chaque cursus envers l'ensemble du collectif pédagogique.

1. Un cursus :

Est un programme universitaire débouchant sur un diplôme. Il montre l'ensemble des enseignements suivis par un étudiant de sa première inscription administrative à la délivrance du diplôme.

2. Formations :

Le département reçoit des étudiants, ayant passé avec succès les deux semestres du domaine mathématique et informatique et après étude des fiches de vœux et sur la base des places pédagogiques disponibles, dans les licences et masters.

3. Système LMD (Licence, Master, Doctorat) :**a. Licence :**

La formation de licence s'étale sur trois années : Licence 1ere année, Licence 2eme année, Licence 3eme année (L1, L2, L3).

Chaque année universitaire pédagogique est composée de deux semestres S1 et S2, chacun est composé d'un certain nombre de modules.

Le département assure deux spécialités en licence

- Licence Informatique
- Licence Informatique et Multimédia

b. Master :

La poursuite des études au cycle de master n'est autorisée que pour les étudiants ayant eu un bon classement au premier cycle. Cette seconde formation est composée de trois semestres d'études théoriques clôturée par un projet de mémoire de fin de cycle.

Chaque année est composée de deux semestres :

Pour les années de la licence l'évaluation est annuelle, pour le master l'évaluation est semestrielle.

Le département assure plusieurs spécialités en master :

- Master Conduite de Projets Informatiques
- Master Réseaux Mobilité et Systèmes Embarqués
- Master Systèmes Informatiques
- Master Ingénierie des Systèmes d'Information (Professionnelle)

c. Doctorat :

En fin le diplôme de doctorat est assuré pour les candidats de master ayant subi un concours au niveau national.

Le département assure plusieurs formations en doctorat LMD :

- Intelligence artificielle et systèmes
Réseaux, Mobilité et Systèmes embarqués

4. Unités d'enseignement (UE) :

L'enseignement d'un semestre s'articule autour d'unité d'enseignement. Les unités d'enseignement sont de quatre types : fondamental, découverte, spécialité ou transversal.

Une UE est dotée d'un crédit (points) porté par les éléments pédagogiques et lui est affecté un coefficient de pondération par rapport aux autres unités du même semestre.

Un semestre est généralement composé de trois à quatre UE. On calcule la moyenne de l'UE. Ensuite la moyenne du semestre est obtenue à partir des moyennes calculées des UE pondérées par leurs coefficients. Les UE d'un semestre se compensent entre elles. Une UE est généralement constituée d'une à quatre matières dispensées par toute forme d'enseignement. Les semestres d'une même année se compensent entre eux, dans le cas de la licence, par le calcul de la moyenne des notes des UE qui les composent affectées de leurs coefficients respectifs.

5. Matières enseignées :

Une matière (module) est dotée d'un crédit matière et lui est affecté un coefficient matière. A chaque matière correspond un examen final et/ou contrôle continu. Ce dernier peut être évalué par des notes de TD, TP, exposé, séminaire, projet ou toute autre forme arrêtée par l'équipe pédagogique. L'examen final et le contrôle continu sont pondérés par des coefficients. Généralement en cas de résultat insuffisant, l'étudiant est admis à une deuxième session organisée en fin de semestre ou en fin d'année universitaire.

En fonction du mode de calcul du rattrapage avec ou sans le contrôle continu, la moyenne de la matière est recalculée, le résultat le plus favorable est retenu pour l'étudiant.

6. Suivi pédagogique des enseignants :

Un enseignant est chargé d'assurer des séances de cours magistral, de TD ou de TP repartis au cours du jour. Une journée pédagogique est composée de six séances d'une heure et demi chacune.

On rencontre des enseignants permanents, associés ou vacataires. Chaque enseignant possède son propre grade de professeur, maître de conférences, maître assistant, ou enfin professeur-ingénieur.

Conclusion :

Dans ce chapitre nous avons présenté notre organisme d'accueil, afin de connaître toutes les informations concernant le département informatique de l'UMMTO.

Ces informations seront utiles pour la réalisation de notre projet. Le chapitre suivant fait l'objet d'une étude sur les réseaux, l'internet et le web.

Chapitre II

*Généralités sur les réseaux,
Internet et Web*

Introduction :

Une véritable révolution s'est produite dans le monde de l'informatique ; c'est l'explosion d'internet plus particulièrement la fondation sur laquelle elle est construite : le web, considéré comme outil de communication, de recherche/publication d'information et de création de nouveaux services.

I. Les réseaux informatiques :

Un réseau est un ensemble de moyens matériels et logiciels, géographiquement dispersés, destinés à offrir un service comme les réseaux téléphoniques ou à assurer le transport des données ; Les réseaux devront véhiculer toute sorte d'information avec des contraintes temporelles plus au moins fortes avec des volumes extrêmement variés.

I.1. Définition d'un réseau informatique :

C'est une interconnexion d'ordinateurs, terminaux, serveurs, qui permettent un échange d'information entre ces derniers. Cette connexion est réalisée par des câbles électriques, fibres optiques et satellites. Dans un premier temps, ces communications étaient destinées au transport de données informatiques, aujourd'hui les réseaux tendent à intégrer également la parole et la vidéo.

I.2. Objectif des réseaux :

Les réseaux permettent :

- Le partage de données (fichiers, BDD,...).
- Le partage de périphériques.
- La communication entre utilisateurs (messagerie électronique).

I.3. Classification des réseaux informatiques : [1]

La classification des réseaux se distingue en deux :

a. Selon l'envergure:

- **les réseaux personnels ou PAN (Personal Area Network) :**

Désignent un réseau restreint d'équipements informatiques habituellement utilisés dans le cadre d'une utilisation personnelle. Les bus utilisés les plus courants sont l'USB, les technologies sans fil telles que Bluetooth ou IR (infra rouge).

- **Les réseaux locaux ou LAN (Local Area Network) :**

Sont des réseaux informatiques à une échelle géographique relativement restreinte, par exemple une salle informatique, une habitation particulière, un bâtiment ou un site d'entreprise.

- **Les réseaux métropolitains ou MAN (Metropolitan Area Network) :**

Désignent des réseaux étendus d'ordinateurs habituellement utilisés dans les campus ou dans les villes. Ces réseaux utilisent généralement des fibres optiques. Par exemple, une université peut avoir un MAN qui lie plusieurs réseaux locaux.

- **Les réseaux étendus ou WAN (Wide Area Network) :**

Sont des réseaux informatiques couvrant une grande zone géographique, typiquement à l'échelle d'un pays, d'un continent, voir de la planète entière. Le plus grand WAN est le réseau Internet.

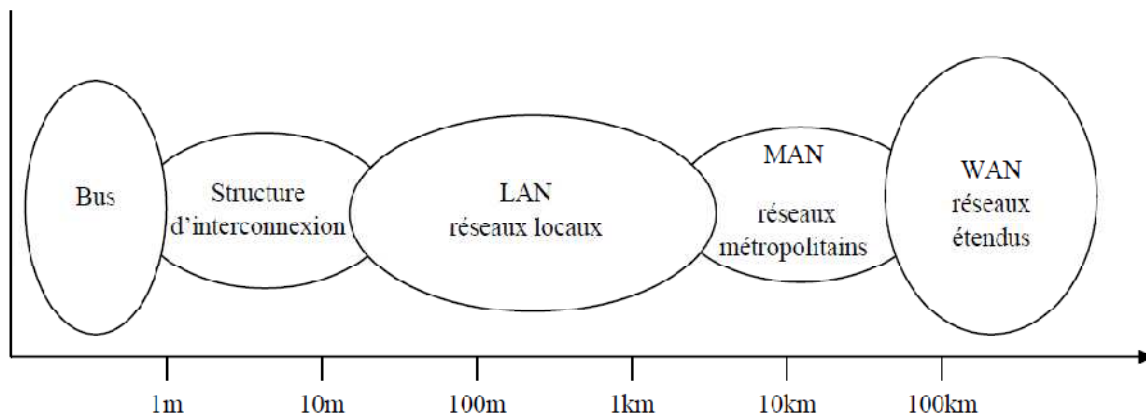


Figure II.1 : Classification des réseaux informatiques selon l'envergure

b. Selon la topologie:

Une topologie est une définition de l'architecture et structure des nœuds composant le réseau, elle donne une certaine disposition, arrangement et hiérarchie des différents postes informatiques.

On trouve les topologies suivantes :

- **Réseau en étoile :**

Les stations du réseau communiquent bi directionnellement avec le serveur. C'est une liaison point à point entre le serveur et chaque machine. Deux stations peuvent échanger des données à condition d'en passer par le serveur.

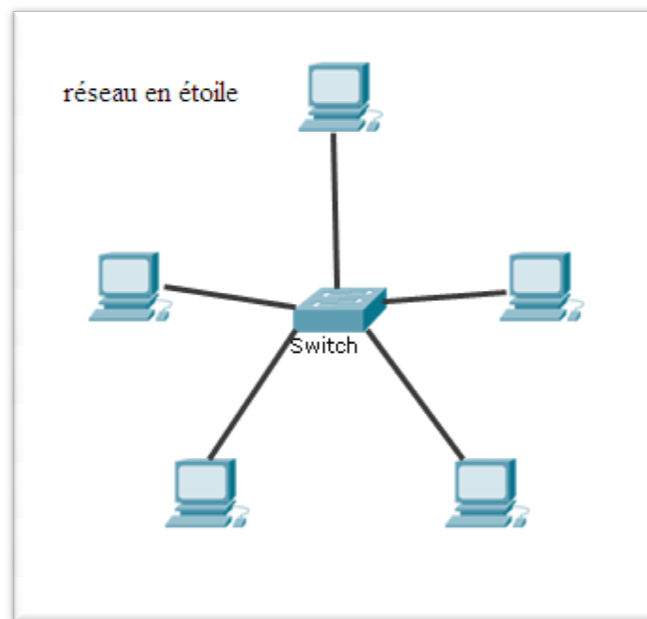


Figure II.2. Topologie en étoile

- **Réseau en bus :**

Chaque station, serveurs compris, communique en permanence de manière bidirectionnelle sur un support de transmission unique. Les liaisons ne sont pas point à point, c'est un réseau à diffusion. On peut également y pratiquer le multipoint car une machine peut adresser un même paquet de données à plusieurs stations simultanément. Dans cette configuration, il est important de fixer une politique d'accès au bus si on peut éviter que deux stations ne le fassent en même temps

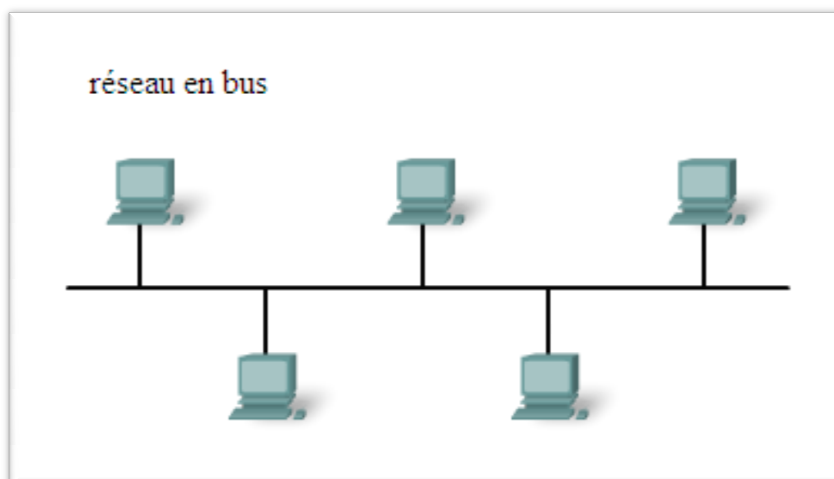


Figure II.3. Topologie en bus

- **Réseau en anneau :**

Toutes les stations reçoivent toutes les trames d'information et les font passer à leurs voisines. Nous avons ici l'exemple de liaisons point à point monodirectionnelles entre chacune des machines. La circulation de l'information se fait en sens unique.

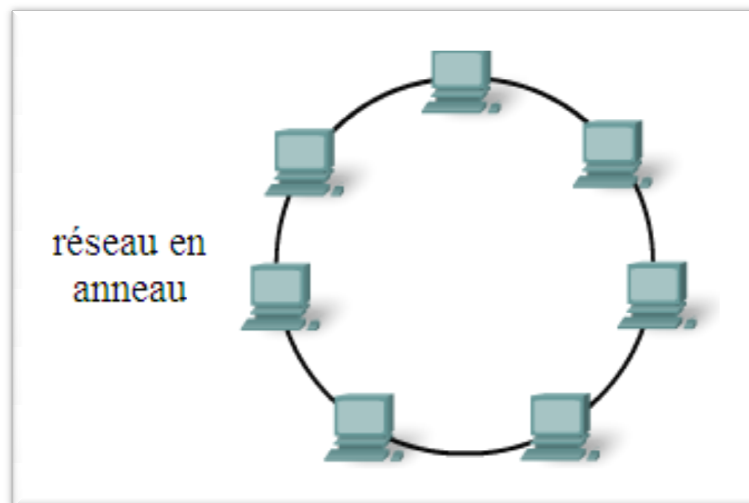


Figure II.4. Topologie en anneau

- **Réseau maillé :**

Un réseau maillé est caractérisé par le fait que deux nœuds quelconques soient reliés l'un à l'autre. Ce type de réseau permet plus de souplesse et de fiabilité dans son utilisation, mais il est difficilement envisageable pour un grand nombre de nœuds. Généralement, le maillage n'est pas parfait, on dit que c'est un réseau partiellement maillé entre les nœuds. Nous avons ici des liaisons points à point entre machines.

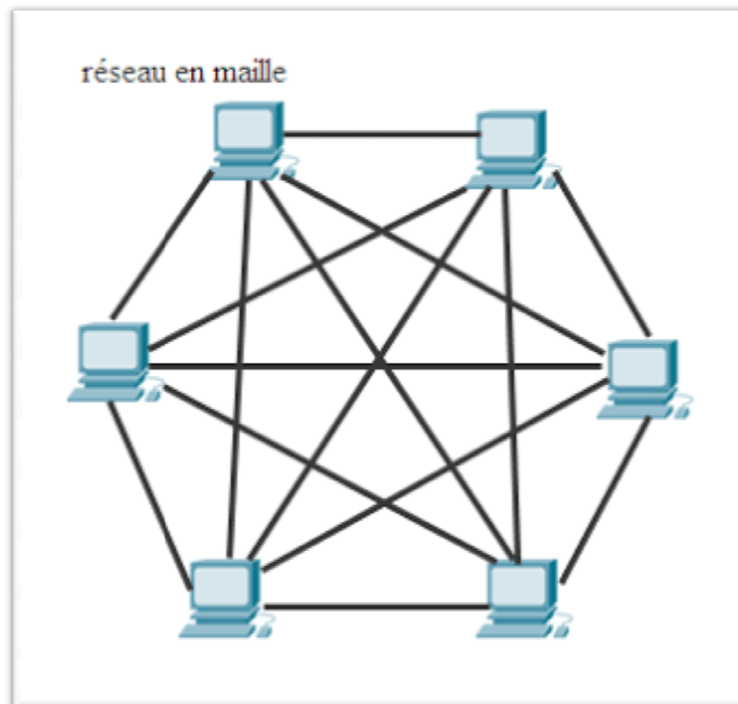


Figure II.5. Topologie maillé

Remarque :

Toute topologie adoptée doit faire au préalable l'objet d'une étude prenant en compte plusieurs facteurs : Nombre de stations à connecter, flux des données, coût, distance entre entités communicantes, évolution possible, résistance aux pannes et lignes de secours, administration ...etc.

I.4. Architecture des réseaux :

Le transport des données d'une extrémité à l'autre d'un réseau nécessite un support physique de communication. Cependant, pour que ces données arrivent correctement au destinataire, avec la qualité de service exigée, il faut une architecture logicielle. L'ensemble de protocoles nécessaires constitue une architecture. Un protocole est l'ensemble de règles qui définissent les modalités de fonctionnement d'une communication entre deux ordinateurs.

a. L'architecture OSI (Open System Inter connexion) : [2]

Que l'on appelle open system Interconnexion ou Interconnexion des Systèmes ouverts, c'est une architecture qui forme le modèle de référence qui comporte sept couches.

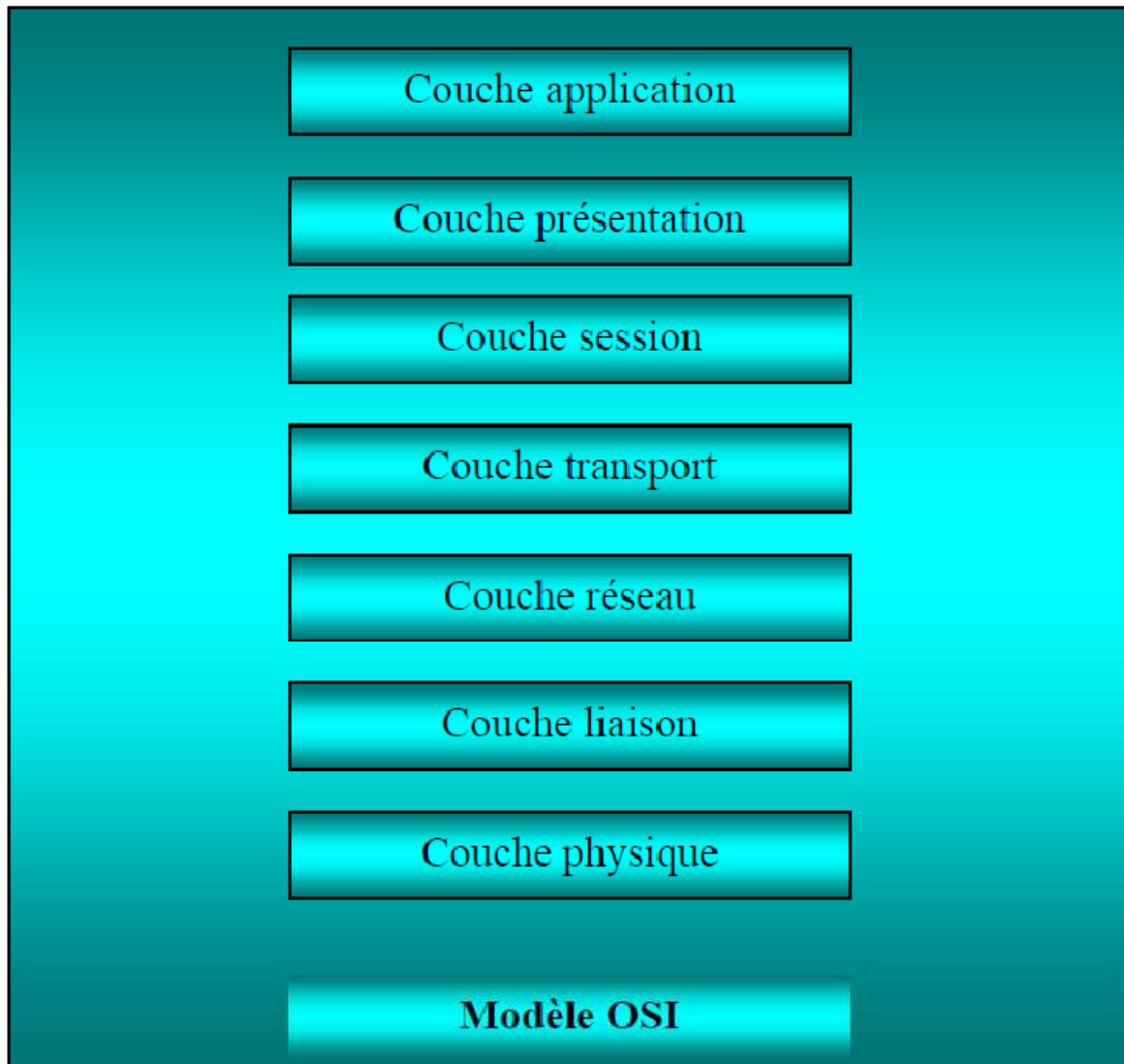


Figure II.6 Les couches du modèle OSI.

➤ **La couche physique :**

Elle est responsable de la manipulation des détails mécaniques et électriques de la transmission physique d'un flot de bits sur le canal de communication.

➤ **La couche de liaison :**

Sa tâche principale est de détecter et de corriger les erreurs de transmission, l'émetteur traite également les accusés de réception émis par le récepteur.

➤ **La couche réseau :**

Elle s'occupe du routage et du contrôle de congestion. L'information stockée au niveau de cette couche est structurée sous forme de paquets de données

➤ **La couche transport :**

Elle est responsable du transfert de données entre les entités de sessions. Ce transport devant être transparent, c'est-à-dire indépendant de la succession des caractères et même des éléments binaires transportés.

➤ **La couche session :**

Elle permet à des utilisateurs de différentes machines d'établir des sessions entre eux. Parmi ses services : la gestion du dialogue, le transfert de fichiers, ...

➤ **La couche présentation :**

Elle gère la syntaxe et la sémantique des informations transmises (le codage, la compression, la cryptographie des données, ...).

➤ **La couche application :**

Elle est responsable de l'interaction directe avec les utilisateurs. Elle traite les protocoles de connexion à distance, le courrier électronique ainsi que la définition des terminaux virtuels.

b. L'architecture TCP/IP (Transmission Control Protocol / Internet protocol): [3]

Dans les années 70, DOD (Department Of Defense), devant le foisonnement de machines utilisant des protocoles de communication différents et incompatibles, décide de définir sa propre architecture. L'architecture TCP/IP est à la source du réseau Internet de nombreux réseaux privés. Cette architecture comme celle d'OSI est répartie en couches.

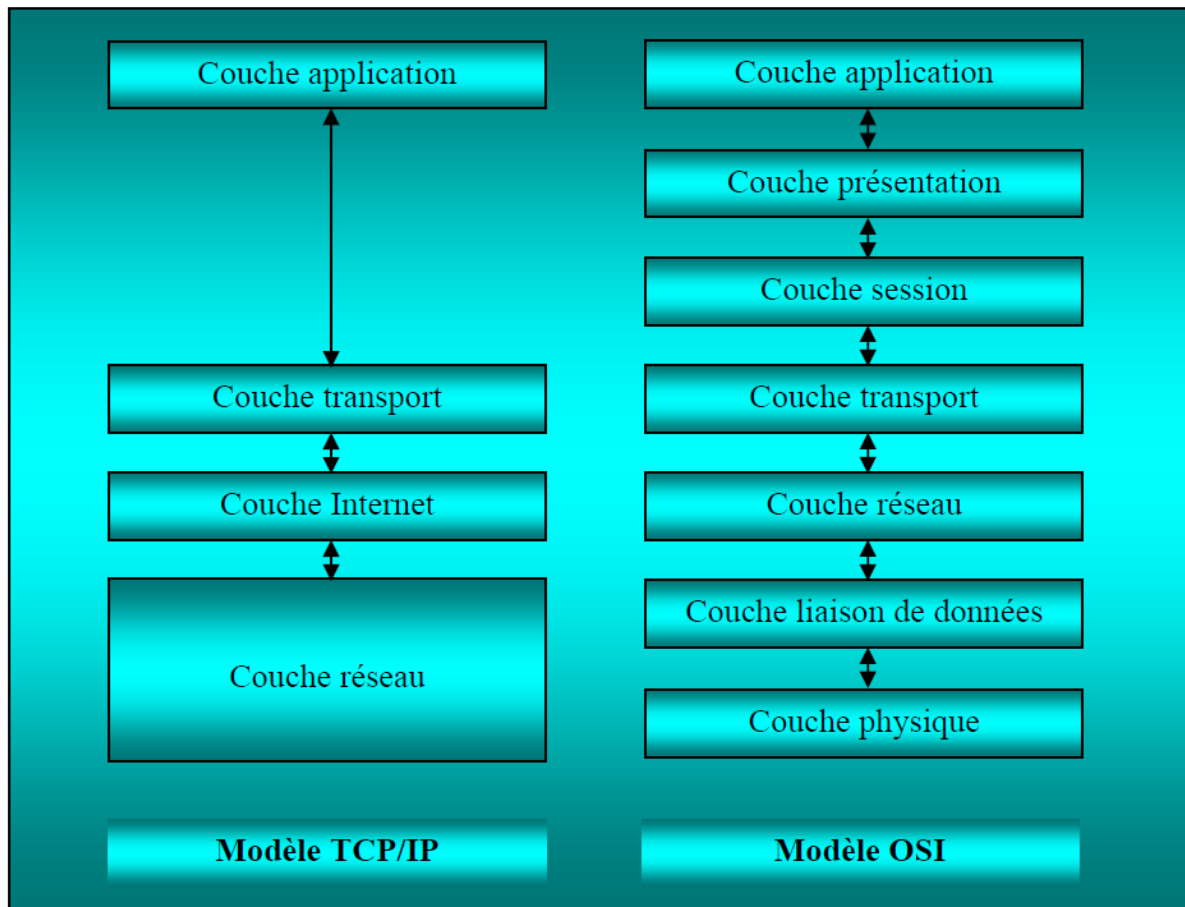


Figure II.7 La pile de protocole TCP/IP avec sa correspondance OSI.

➤ **La couche application :**

Elle englobe les applications standards du réseau :

SMTP (Simple Mail Transport Protocol) : permet d'échanger du courrier entre deux serveurs de messagerie.

TELNET : Protocole permettant de se connecter sur une machine distante (serveur) en tant que utilisateur.

FTP (File Transfert Protocol) : Est un protocole permettant d'échanger des fichiers via Internet.

DNS (Domain Name Service) : Service de nom du domaine. Il permet de convertir le nom d'une machine en une adresse réseau (ou IP) et vice versa.

➤ **La couche transport :**

Elle assure l'acheminement des données sur le réseau local et les mécanismes permettant de connaître l'état de la transmission. Les protocoles de cette couche permettent d'envoyer des informations d'une machine à une autre.

TCP : Il assure le contrôle de données, orienté connexion.

UDP (User Datagram Protocol) : Archaïque et non orienté connexion, il n'assure aucun contrôle de transmission de données.

➤ **Couche Internet :**

La couche Internet est chargée de fournir le paquet de données. Elle contient cinq protocoles, les trois premiers sont les plus importants :

IP : gère les destinations des messages, adresse du destinataire.

ARP (Address Resolution Protocol) : gère les adresses des cartes réseau. Chaque carte a sa propre adresse d'identification codée sur 48 bits.

ICMP (Internet Control Message Protocol) : gère les informations relatives aux erreurs de transmission. Il ne corrige pas les erreurs, mais signale aux autres couches que le message contient des erreurs.

RARP (Reverse Address Resolution Protocol) : gère l'adresse IP pour les équipements, qui ne peuvent s'en procurer une, par lecture d'information dans un fichier de configuration. En effet, lorsque le PC démarre, la configuration réseau lit l'adresse IP qu'elle va utiliser. Ceci n'est pas possible dans certains équipements qui ne possèdent pas de disques durs (imprimantes, terminaux).

IGMP (Internet Group Management Protocol) : permet d'envoyer le même message à des machines faisant parti d'un groupe. Ce protocole permet également à ces machines de s'abonner ou de se désabonner d'un groupe. Ceci est utilisé par exemple dans la vidéo conférence à plusieurs machines, envoi de vidéos,...

I.5. Définition de l'architecture client /serveur : [4]

Client/serveur désigne un mode de communication entre des programmes informatiques:

- a. Un programme tient le rôle de « serveur » : il est conçu de façon à pouvoir répondre à des requêtes extérieures.
- b. Les autres programmes « client » contacte le serveur pour lui soumettre des demandes, et attendre sa réponse.

Un serveur peut fournir un ou plusieurs services, et ceux, à un ou plusieurs clients, le programme serveur et /ou les programmes client peuvent se trouver sur la même machine ou sur des machines différentes, reliées entre elles par réseau.

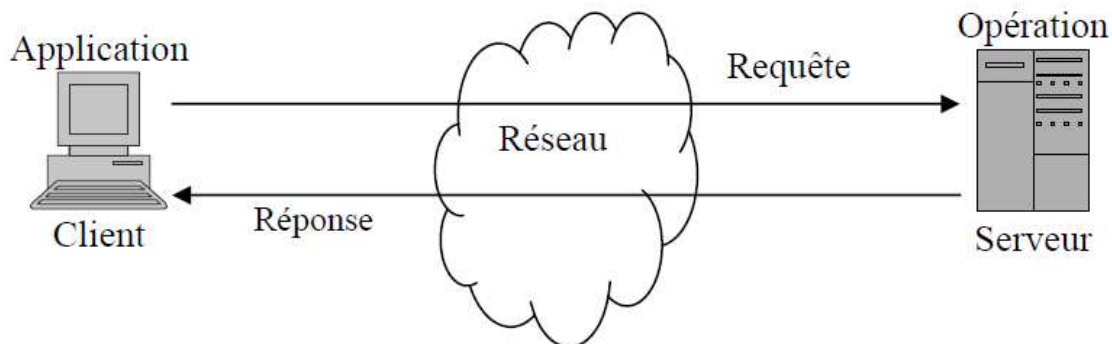


Figure II.8. Le dialogue client/serveur.

I.5.1. Caractéristiques du modèle client/serveur :

Bien que le client/serveur soit une forme d'informatique distribuée, il n'est pas supposé d'appeler système client/serveur que ceux qui partagent les caractéristiques suivantes :

- ✓ **Service** : Le modèle Client/serveur est bâti sur une relation entre des processus tournants sur des machines séparées. Le processus serveur est un fournisseur de services, le processus client est un consommateur de services. Le modèle établit ainsi une répartition claire des fonctions à partir de la notion de service.
- ✓ **Partage des ressources** : Un serveur peut traiter plusieurs clients en même temps et contrôler leurs accès aux ressources.

- ✓ **Asymétrie des protocoles** : La relation entre clients et serveur est de type plusieurs vers un. C'est toujours le client qui déclenche le dialogue en demandant un service. Les serveurs attendent passivement les requêtes des clients.
- ✓ **Transparence à la localisation** : Les processus serveur et client peuvent résider sur la même machine ou, par l'intermédiaire d'un réseau, sur deux machines différentes interconnectées. Le logiciel Client/serveur masque aux clients la localisation du serveur en redirigeant les demandes de service si nécessaire. Un programme peut être client, serveur ou les deux.
- ✓ **Assemblage multi vendeur** : Le logiciel Client/serveur idéal est indépendant de la plate-forme matérielle ou du système d'exploitation. On doit toujours pouvoir mélanger et apparier les plates-formes client et serveur.
- ✓ **Échange de messages** : Client et serveur sont des systèmes à liaison épisodique qui interagissent au moyen de messages. Le message est le mécanisme d'émission des demandes de services et des réponses à celles-ci.
- ✓ **Encapsulation des services** : Le serveur est un « spécialiste », un message lui indique quel service est requis. C'est à lui de décider comment rendre ce service. Les serveurs peuvent être mis à niveau sans effet sur les clients tant que l'interface des messages reste la même.
- ✓ **Redimensionnement** : Les systèmes Client/serveur peuvent être redimensionnés horizontalement ou verticalement. Le redimensionnement horizontal correspond à l'ajout ou au retrait de stations clientes, avec un léger impact sur les performances. Le redimensionnement vertical correspond à la migration du serveur vers une machine plus puissante ou consiste à distribuer la charge de traitement sur plusieurs serveurs.
- ✓ **Intégrité** : Le code et les données du serveur sont gérés de façon centralisée, ce qui garantit un moindre coût de maintenance et une meilleure intégrité des données partagées. De l'autre côté, les clients restent individuels et indépendants.

Les caractéristiques ci-dessus facilitent la répartition de l'intelligence sur tout le réseau. Elles constituent une base pour la conception d'applications en réseau à liaisons épisodiques.

I.5.2. Technique du dialogue Client /Serveur :**I.5.2.1. Notions de base et fonctionnement de client/serveur :**

- **Client (Client) :** Processus demandant l'exécution d'une opération à un autre processus par envoi d'un message contenant le descriptif de l'opération à exécuter et attendant la réponse à cette opération par un message en retour.
- **Serveur (Server) :** Processus accomplissant une opération sur demande d'un client et transmettant la réponse à ce client.
- **Requête (Request) :** Message transmis par un client à un serveur décrivant l'opération à exécuter pour le compte du client.
- **Réponse (Reply) :** Message transmis par le serveur à un client suite à l'exécution d'une opération contenant les paramètres de retour de l'opération.

Un client exécute une application et demande l'exécution d'une opération à un serveur par le biais d'une requête. Il reçoit une réponse, lui indiquant par exemple que l'opération a été bien exécutée.

Les appels au service de transport mis en jeu sont au nombre de quatre :

- 1) **Send Request () :** Permet au client d'émettre le message décrivant la requête à une adresse correspondant à la porte d'écoute du serveur.
- 2) **Receive Request () :** Permet au serveur de recevoir la requête sur sa porte d'écoute.
- 3) **Send Reply () :** Permet au serveur d'envoyer la réponse sur la porte d'écoute du client.
- 4) **Receive Reply () :** Permet au client de recevoir la réponse en prévenance du serveur.

Remarque :

Un client ne dialogue qu'avec un serveur à la fois, alors qu'un serveur peut gérer plusieurs connexions simultanément.

I.5.3. Architecture client/serveur multi niveaux :

Les applications client/serveur se différencient par la façon dont les fonctions distribuées se partagent entre le client et le serveur. Le modèle oriente serveur place plus de fonctionnalités sur le serveur tandis que le modèle oriente client, le gros de l'application tourne du côté client.

Les spécialistes du client/serveur préfèrent utiliser le terme d'architecture client/serveur à deux niveaux ou à trois niveaux plutôt que ceux d'orienté client ou d'orienté serveur.

I.5.3.1 Architecture client/serveur à deux niveaux :

L'architecture à deux niveaux (appelé aussi architecture 2-tiers) caractérise les systèmes Client/serveur dans lesquels le client demande une ressource et le serveur la lui fournit directement sans faire appel à un autre serveur intermédiaire.

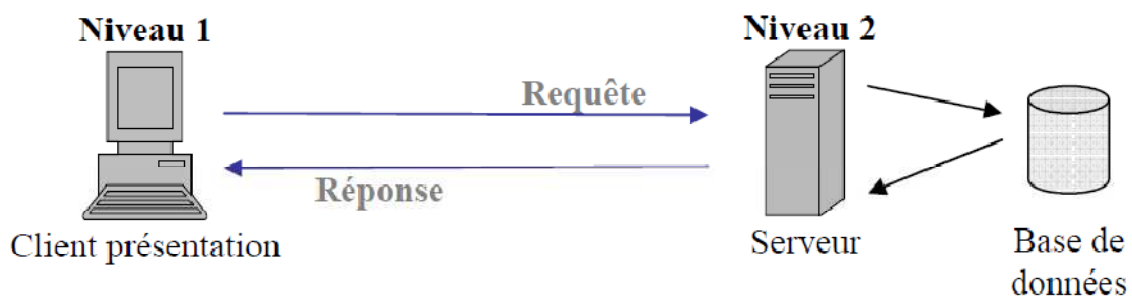


Figure II.9. Architecture client/serveur deux niveaux.

La simplicité est le facteur principal qui explique la popularité de l'architecture client/serveur à deux niveaux, cette dernière offre une grande souplesse et facilite la création d'outils frontaux permettant aux utilisateurs de créer leur application.

Avec l'arrivée des nouvelles technologies (internet, web) l'architecture client/serveur deux niveaux a évolué vers l'architecture trois niveaux.

I.5.3.2 Architecture trois niveaux :

L'architecture à trois niveaux (appelé aussi architecture à 3-tiers) est généralement partagée entre :

- **Un client** : le demandeur de ressource.
- **Un serveur d'application (middleware)** : qui est chargé de fournir la ressource en faisant appel à un autre serveur.

- **Un serveur secondaire** : fournissant un service au premier serveur (généralement un serveur de base de données).

Les applications de l'architecture client/serveur à trois niveaux sont plus faciles à déployer et à gérer sur le réseau, elles essayent de minimiser les échanges sur le réseau en créant des niveaux de service. L'architecture client serveur à trois niveaux est recommandée pour diverses raisons dont on cite celle de l'Internet

L'architecture à trois niveaux devient de plus en plus à N niveaux, cela est dû au fait que le niveau intermédiaire est implémenté sous forme d'une collection de composants. Chaque composant se charge d'une fonction relativement petite exécutée suite à une demande d'un client, la demande d'un client peut nécessiter plusieurs composants qui s'entraident pour la satisfaire, d'où le modèle à trois niveaux est en fait à N niveau.

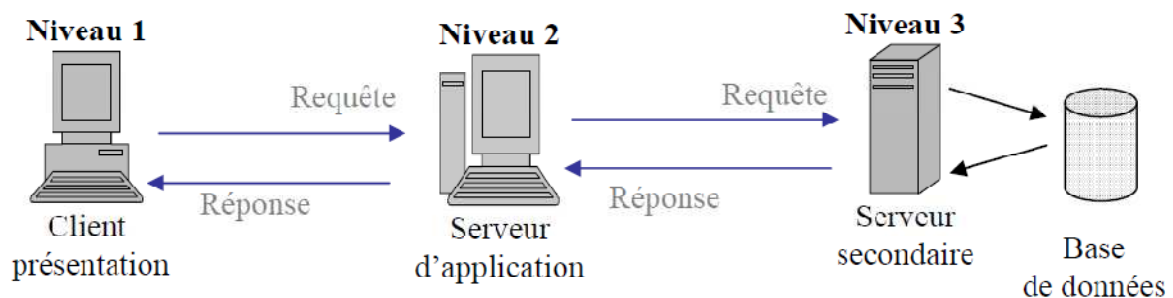


Figure II.10 : Architecture client/serveur trois niveaux.

I.5.3.3. Architecture en couche :

Les réseaux d'ordinateurs sont conçus de façon très structurée. Pour réduire la complexité de leur conception, ils sont organisés en séries de couches, chacune étant construite sur la précédente. Cependant, dans chaque réseau, l'objet de chaque couche de niveau (n) est d'offrir certains services à la couche de niveau (n+1), en leur épargnant les détails de la mise en œuvre de ses services.

I.5.4. Les types du serveur :

Il existe plusieurs variantes des technologies client/serveur, comme les décrivent les paragraphes suivants :

I.5.4.1. Serveur de fichiers :

Dans le cas de serveur de fichiers, le client requiert des enregistrements de fichiers en émettant des requêtes au serveur de fichiers. Les serveurs de fichiers sont utiles pour partager des fichiers sur un réseau et ils sont indispensables pour créer des banques de documents, d'images, ...etc. mais l'obtention de l'information nécessite de nombreux échanges de message sur le réseau.

I.5.4.2. Serveur de base de données :

Dans le cas d'un serveur de base de données, le client émet des requêtes SQL sous forme de message en direction du serveur, le résultat est renvoyé au client. Le serveur utilise sa capacité de traitement pour rechercher les données demandées au lieu de transmettre tous les articles au client et de le laisser en faire la sélection, donc la puissance est répartie et utilisée de façon beaucoup plus efficace.

I.5.4.3. Serveur de transaction :

Dans ce modèle, le client invoque des procédures distantes résidant sur le serveur qui comporte un moteur de base de données SQL. Ces procédures distantes exécutent un ensemble d'instructions SQL. L'échange sur le réseau consiste en un seul message requête/réponse (contrairement à l'application serveur base de données pour laquelle le message requête/réponse est émis pour chaque instruction SQL). Pour ce type de serveurs l'application Client/serveur nécessite du code source au niveau du serveur.

I.5.4.4. Serveur d'application Web :

La world wide web est la première application Client/serveur intergalactique, ce modèle consiste en des clients, léger, portable qui communiquent avec de très gros serveurs. Un serveur web renvoie des documents lorsque le client les demandes

I.5.4.5. Serveur groupware :

Le groupware s'intéresse à la gestion d'informations semi-structurées telles que le texte, l'image, courrier, messagerie et ordonnancement de tâches.

I.5.4.6. Serveur d'application objet :

Dans le cas d'un serveur objet, l'application Client/serveur est écrite sous forme d'un jeu d'objets communicants. Les objets client communiquent avec les objets serveur au moyen d'un courtier d'objets ou ORB (Object Request Broker). Le client invoque une méthode sur un objet distant, l'ORB localise une instance de la classe, appelle la méthode demandée et renvoie les résultats à l'objet Client.

I.5.5. Le fonctionnement du modèle client/serveur :

La communication entre le client et le serveur est de type transactionnel et coopératif. Le client transmet sa demande et interrompt son traitement jusqu'à ce qu'il reçoive la réponse du serveur. Le mode de fonctionnement du client est dit bloquant. Le serveur quant à lui ne fait rien jusqu'à ce qu'il reçoive une requête. On dira que le serveur est en état d'écoute. Lorsque la demande a été traitée par le serveur, celui-ci se remet à l'écoute de nouvelles demandes. Il s'agit du mode de fonctionnement le plus simple entre client et serveur, signalons qu'il existe d'autres mécanismes qui permettant au client de continuer son traitement après l'envoi d'une requête, il s'agit alors d'un fonctionnement non bloquant. Dans ce cas, il n'est pas nécessaire de disposer de mécanismes de synchronisation entre client et serveur. Le client écrit sa requête dans un tampon, une fois la demande de communication au serveur acceptée, il se met en attente de réponse (en mode non bloquant, le client peut continuer son traitement celui-ci ne dépend pas de résultat de la requête). Le serveur lit le contenu du tampon et traite la requête (lui-même ou fait intervenir d'autres serveurs). Une fois le traitement terminé, la réponse écrite dans un tampon accessible par tous les serveurs.

II. Internet :**II.1. Définition de l'Internet : [5]**

L'internet ou inter Connected Network, constitue un réseau de réseaux qui relie, dans le monde entier, des ordinateurs en utilisant un protocole de transmission et de communication constituant un langage commun permettant la connexion de toutes les machines (Pc, Mac, Unix)

Ce langage commun s'appelle Protocol /Internet, Protocol (TCP/IP)

L'internet vaut avant tout par ses applications ou ses services .On a trop souvent tendance à penser l'internet comme un lieu qu'il s'agit simplement d'un réseau sur lequel se greffent des applications ; il convient de parler de « service de l'internet »

II.2. Histoire de l'Internet

L'internet remonte au développement de réseau Arpanet (Advanced Research Projects Agency), liée à la défense américaine ; crée en 1968 par le département américain de la défense dans un but stratégique, pour relier ses centres de recherche.

L'objectif était de faire fonctionner un réseau censé résister à une attaque nucléaire et à des sabotages, en palliant à la centralisation excessive des infrastructures existantes.

Le réseau initialement ne permettait que l'échange de textes, ce qu'on appelle le courrier électronique. Ce réseau a été développé et étendu par des universitaires à la fin des années 70.

En 1977, Internet devient réalité car le TCP/IP est effectivement utilisé pour relier divers réseaux à Arpanet.

En 1990, c'est la naissance d'Internet, avec l'arrivée d'un système d'interface graphique développé par Tim Berners-Lee du CERN à Genève appelée World Wide Web, qui permet d'ouvrir le réseau au grand public en simplifiant les procédures de consultation des sites.

Comme on le constate, l'Internet a évolué, passant d'un moyen de sécurité gouvernementale et militaire à un moyen de communication accessible au "Village Global de la Planète".

II.3. Principaux services offerts par l'Internet : [6]

Nous avons vu qu'Internet s'appuyait sur le protocole TCP/IP. Il existe de nombreux autres protocoles en usage sur ce réseau.

Les principaux services, offerts par Internet, sont le web, la messagerie électronique (e-mail), les forums, le chat et le FTP. Ils utilisent chacun un protocole différent :

Protocoles :	Services :
HTTP (HyperText Transfer Protocol)	Visualisation de pages Web
FTP (File Transfer Protocol)	Transferts de fichiers
SMTP (Simple Mail Transfer Protocol) POP (Post Office Protocol)	Transferts de courriers électroniques (e-mail) Récupération des e-mails sur le serveur
NNTP (Network News Transfer Protocol)	Forums de discussions en temps différés
IRC (Internet Relay Chat)	Dialogue en temps réel

- Le Web autorise la navigation sur des pages multimédia. Il a été créé en 1989 au CERN de Genève par Tim Berners-Lee, Robert Cailliau et leur équipe. Il s'appuie sur le protocole HTTP. Ce sigle figure au début de l'adresse du serveur où vous vous connectez pour accéder à une page Web
- Le FTP est un protocole de transfert de fichiers. Ce transfert s'effectue en établissant une connexion entre un serveur FTP et un client FTP situé sur votre ordinateur. Les fichiers échangés sont des fichiers informatiques de tous types (texte, images, sons, logiciels, ...)
- La messagerie électronique permet aux internautes dotés d'une adresse E-Mail de s'expédier mutuellement des messages dans le monde entier. Les messages sont expédiés et reçus grâce à deux protocoles (SMTP et POP), mais ils peuvent aussi transiter par le Web.
- Les news sont des espaces de discussions où des groupes de connectés discutent en temps différé. Les news sont regroupés par thèmes d'intérêt, appelés newsgroups ou forums.
- Le chat : la messagerie électronique instantanée, c'est-à-dire une communication écrite en temps réel et interactive

III. Le web :

Le Web a été inventé plusieurs années après Internet, mais c'est lui qui a contribué à l'explosion de l'utilisation d'Internet par le grand public, grâce à sa facilité d'emploi. Depuis, le Web est fréquemment confondu avec Internet alors qu'il n'est en réalité qu'un des services.

III.1. Définition : [7]

On appelle World Wide Web (noté WWW) ou tout simplement Web (mot anglais signifiant toile) la "toile virtuelle" formée par les différents documents appelés « pages web » liés entre eux par des hyperliens.

Les pages web sont généralement organisées autour d'une page d'accueil, jouant un point central dans la navigation à l'aide des liens hypertextes. Cet ensemble cohérent de pages web liées par des liens hypertextes et articulées autour d'une page d'accueil commune est appelée site web.

III.2. Le principe du Web :

Le web est une énorme archive vivante composé de pages web stockées sur des serveurs web c'est-à-dire des machines connectées à Internet en permanence et chargées de fournir les pages web demandées. Chacune des pages web, et plus généralement toute ressource en ligne (image, vidéo, musique, animation, etc.), est repérée par une adresse unique appelée url.

L'élément clé pour la navigation au sein des pages web est le navigateur (en anglais browser, parfois également appelé fureteur, fouineur, feuilleteur ou butineur au Québec), c'est-à-dire le logiciel client capable d'interroger les serveurs web, d'exploiter leurs résultats et de mettre en page les informations grâce aux instructions contenues dans la page HTML.

III.3. Histoire du Web : [C]

Le concept de World Wide Web a été créé à partir de 1989 au cern centre européen de recherche nucléaires par Tim Berners Lee, puis développé par lui-même et Roberte Cailliau en 1990 dans le but de concevoir un système permettant de naviguer simplement d'un espace à un autre d'internet à l'aide de liens hypertextes et grâce à un navigateur .

En 1993, un navigateur web graphique nommé mosaïc reposant sur les principes de la toile tels qu'ils ont été formulés par l'équipe du Cern de Tim Berners Lee, notamment le http est développé par Eric Bina et Marc Andrsen.

NCSA mosaic jette les bases de l'interface graphique des navigateurs modernes et cause un accroissement exponentiel de la popularité du web

III.4. L'évolution du Web :

Le Web est caractérisé par une évolution constante du fond et de la forme des pages Web. Dans sa conception initiale, le Web dit Web 1.0 comprenait des pages statiques au contenu codé en HTML qui était rarement mises à jour, voir jamais, ces pages sont non-interactives et ne disposent que de peu d'informations.

Une première évolution fut réalisée par des solutions se basant sur un Web dynamique appelé Web 1.5. Ce Web dynamique est généralement basé sur l'association des langages de programmation PHP et des bases de données MySQL.

Lorsque l'internaute accède au site dynamisé, il fait exécuter sur le serveur le langage PHP qui va chercher l'information dans la base de données pour la retranscrire dans la page HTML sur le poste utilisateur.

Le Web subi une nouvelle évolution avec l'apparition de nouvelles technologies comme le langage AJAX qui rend les pages interactives et fluides et le flux RSS, qui permet de rester informé des actualités d'une interface Web. C'est l'avènement du Web collaboratif, interactif et participatif. Ce Web dit Web 2.0, rend l'internaute acteur. Ainsi, il lui est possible sur certains sites Web de modifier, rajouter ou d'effacer du contenu et d'échanger des informations par des techniques synchrones comme les messageries instantanées, la téléphonie sur internet, ...ou des méthodes asynchrones comme les forums, les wikis, les blogs.....

Le Web qui est en cours de développement est le Web 3.0. Ce sera l'arrivée du Web sémantique ou les informations ne seraient plus stockées mais « comprises » par les ordinateurs afin d'apporter à l'utilisateur ce qu'il cherche vraiment. Le Web sémantique est ce que l'on pourrait appeler l'avènement d'outils permettant de transformer automatiquement les données en informations, et les informations en savoir. L'enjeu du Web sémantique est donc de réussir à étiqueter de manière pertinente le contenu disponible sur la toile, pour permettre un accès intelligent. Le principe repose sur l'intelligence collective des utilisateurs.

III.5. Les concepts du Web :

Le principe du Web repose sur l'exploration d'une myriade de pages Web (fichiers HTML) contenant du texte mis en forme, des images, du son, des vidéosL'exploration est rendue possible grâce à un logiciel appelé navigateur (Firefox, Internet Explorer,) qui offre une interface d'accès aux différentes informations disponibles sur le Web.

III.5.1. Les pages Web :

Sont liées les unes aux autres par des liens hypertextes(ou hyperliens). En pratique un lien hypertexte est un simple mot affiché sur un document permettant de passer à un autre document lorsqu'on clique dessus. Le document lié peut être un fichier à transférer, un son, une animation.

III.5.2. Un site Web (ou site Internet) :

Est ainsi un ensemble logiquement structuré de pages web reliées entre elles et formant un tout (le site) dédié à une organisation donnée. La création et la mise à jour d'un site peut être confiée soit à une personne, le webmaster, ou à une entreprise spécialisée, l'agence Web. Pour être accessible sur internet, un site web doit être hébergé sur un serveur web.

Le site web est habituellement architecturé autour d'une page centrale, appelée «Page d'accueil» et proposant des liens vers les autres pages du site, et parfois des liens dits externes, vers des pages étrangères au site pour rechercher une information sans connaître a priori le site susceptible de la fournir, il est nécessaire d'utiliser des outils de recherche.

Il existe deux types de système de recherche sur Internet :

- les moteurs de recherche
- les annuaires web

➤ Un moteur de recherche :

Un moteur de recherche est un programme qui indexe le contenu de différentes ressources Internet, et plus particulièrement de sites Web, qui permet à l'internaute de rechercher de l'information à partir d'un navigateur Web, selon différents paramètres, en ce servant de mots clés, et qui permet d'avoir à l'information ainsi trouvée.

➤ Un annuaire web :

Est un site Internet dont l'objet est le référencement et la classification de sites web, en différents catégories (ou rubriques). L'utilisateur peut alors parcourir l'annuaire rapidement grâce à un système de sélection de catégories et de sous catégories qui permet d'affiner la recherche. Bien qu'étant de plus en plus performants, les moteurs et annuaires de recherche n'indexent pas la totalité des documents présents sur internet .Il existe en effet, une grande

masse d'informations inaccessibles pour les outils de recherche "traditionnels" : c'est le web invisible. Le web invisible comprend des bases, banques de données et bibliothèque en ligne gratuites ou payantes

III.5.2.1. Types de site web :

On considère qu'il existe deux types de sites web : les sites statiques et les sites dynamiques.

- **Les sites statiques** : ce sont des sites réalisés uniquement à l'aide des langages HTML et CSS. Ils fonctionnent très bien mais leur contenu ne peut pas être mis à jour automatiquement : il faut que le propriétaire du site (le webmaster) modifie le code source pour y ajouter des nouveautés. Ce n'est pas très pratique quand on doit mettre à jour son site plusieurs fois dans la même journée ! Les sites statiques sont donc bien adaptés pour réaliser des sites « vitrine », pour présenter par exemple son entreprise, mais sans aller plus loin.
- **Les sites dynamiques** : plus complexes, ils utilisent d'autres langages en plus de HTML et CSS, tels que PHP et MySQL. Le contenu de ces sites web est dit « dynamique » parce qu'il peut changer

III.6. Les principales caractéristiques du Web :

Le Web comporte un certain nombre de caractéristiques spécifiques qui font le système d'information le plus perfectionné qui fait parti d'internet à ce jour :

- **Le Web est un system Hypertexte :**

Un système hypertexte est un système contenant des nœuds liés entre eux par hyperliens permettant de automatiquement d'un nœud à autre .Un document hypertexte est un document qui contient des hyperliens et des nœuds .Un nœud est une « unité minimale d'information », qui signifie simplement que l'information d'un nœud sera toujours présentée entière .les liens entre les parties du texte sont gérés par ordinateur et permettent d'accéder à l'information.

- **Le Web est un système Multimédia :**

Au début d'internet, la plus puissantes des explorateurs du WEB est mémoires sur le réseau se présentait sous forme de texte, le web peut contenir les élément suivent : texte, caractères spéciaux, image, séquence audio (son, musique ...) et séquence vidéo .

- **Le Web assure l'interface avec les systèmes de base de données :**

Une des fonctions les plus puissantes des explorateurs du WEB est leur capacité d'agir comme interface avec des systèmes de base de données reliée à internet.

- **Le Web est un système de traitement distribué :**

Dans le web les documents peuvent être stockés dans toutes les mémoires reliées au réseau, cela signifie que les documents qui y sont mémorisés peuvent être distribués dans tout le réseau.

III.7. Les standards du Web :

Trois technologies ont du être développées pour le World Wide Web :

- **Les URL** pour pouvoir identifier toute ressource dans un hyperlien.
- **Le langage HTML** pour écrire des pages Web contenant des hyperliens.
- **Le protocole de communication HTTP** utilisé entre les navigateurs et les serveurs Web.

III.7.1. URL (Universal/Uniform Resource Locator): [7]

Une URL (Uniform Resource Locator) est un format de nommage universel pour désigner une ressource sur Internet. Il s'agit d'une chaîne de caractères ASCII imprimables qui se décompose en cinq parties :

- Le nom du protocole
- Identifiant et mot de passe
- Le nom du serveur
- Le numéro de port
- Le chemin d'accès à la ressource

III.7.2. HTML :

Le HTML (« HyperText Mark-Up Language ») est un langage dit de « marquage » (de « structuration » ou de « balisage ») dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage. Les balises permettent d'indiquer la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents.

Le langage HTML permet notamment la lecture de documents sur internet à partir de machines différentes, grâce au protocole HTTP, permettant d'accéder via le réseau à des documents repérés par une adresse unique, appelée URL

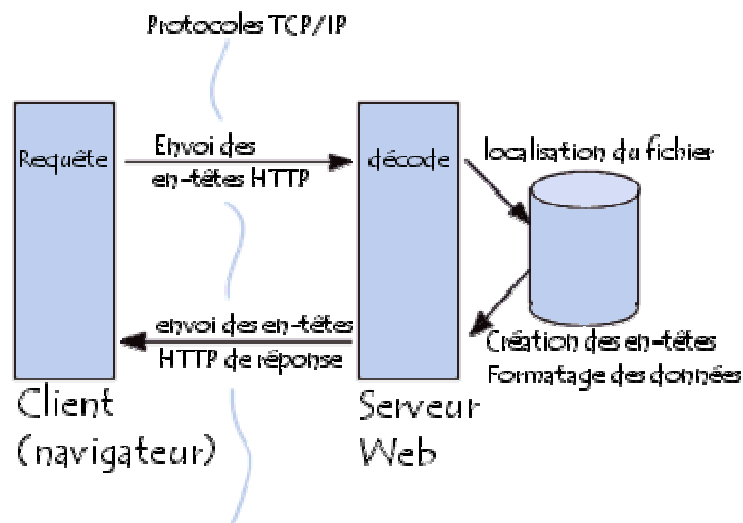
III.7.3. Le protocole http :

Le protocole HTTP (HyperText Transfer Protocol) est le protocole le plus utilisé sur Internet depuis 1990. La version 0.9 était uniquement destinée à transférer des données sur Internet (en particulier des pages Web écrites en HTML. La version 1.0 du protocole (la plus utilisée) permet désormais de transférer des messages avec des en-têtes décrivant le contenu du message en utilisant un codage de type MIME.

Le but du protocole HTTP est de permettre un transfert de fichiers (essentiellement au format HTML) localisés grâce à une chaîne de caractères appelée URL entre un navigateur (le client) et un serveur Web (appelé d'ailleurs httpd sur les machines UNIX.)

III.8. Communication entre navigateur et serveur :

La communication entre le navigateur et le serveur se fait en deux temps :



- Le navigateur effectue une requête HTTP
- Le serveur traite la requête puis envoie une réponse HTTP

Requête HTTP

Une requête HTTP est un ensemble de lignes envoyé au serveur par le navigateur. Elle comprend :

- **Une ligne de requête:** c'est une ligne précisant le type de document demandé, la méthode qui doit être appliquée, et la version du protocole utilisée. La ligne comprend trois éléments devant être séparés par un espace :
 - La méthode
 - L'URL
 - La version du protocole utilisé par le client (généralement HTTP/1.0)
- **Les champs d'en-tête de la requête:** il s'agit d'un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la requête et/ou le client (Navigateur, système d'exploitation, ...). Chacune de ces lignes est composée d'un nom qualifiant le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête
- **Le corps de la requête:** c'est un ensemble de lignes optionnelles devant être séparées des lignes précédentes par une ligne vide et permettant par exemple un envoi de données par une commande POST lors de l'envoi de données au serveur par un formulaire

Réponse HTTP

Une réponse HTTP est un ensemble de lignes envoyées au navigateur par le serveur. Elle comprend :

- **Une ligne de statut:** c'est une ligne précisant la version du protocole utilisé et l'état du traitement de la requête à l'aide d'un code et d'un texte explicatif. La ligne comprend trois éléments devant être séparés par un espace :
 - La version du protocole utilisé
 - Le code de statut
 - La signification du code
- **Les champs d'en-tête de la réponse:** il s'agit d'un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la réponse et/ou le serveur. Chacune de ces lignes est composée d'un nom qualifiant le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête
- **Le corps de la réponse:** il contient le document demandé

Conclusion :

Dans ce chapitre, on a passé en revue quelques généralités sur les réseaux avant de présenter le modèle Client/Serveur; modèle de base des systèmes distribués. On déploie aujourd'hui des applications Client/Serveur servant des milliers de clients dans l'entreprise. Ces applications tournant souvent sur de nombreux serveurs et consistent en centaines de composants. Elles sont au cœur fonctionnement de l'entreprise. Les serveurs peuvent accepter des requêtes de l'un des milliers de PC (clients) connectés.

On a aussi donné un aperçu de ce qu'est l'internet, des services qu'il offre parmi lesquels le Web, nous avons alors les concepts du Web et les technologies associées.

Chapitre III

Framework

Introduction :

Pour créer un site web performant, sécurisé, avec toutes les fonctionnalités des applications modernes, sans réinventer la roue, ni passer sa vie à taper du code, le mieux est d'utiliser un framework. Ce dernier s'adresse à un développeur, il est là pour lui faciliter la vie, lui faire gagner du temps, le guider dans l'organisation de son code, l'aider à le tester et le mettre à jour.

I. C'est quoi un Framework ?[D]

C'est la bonne question à se poser au départ. Qu'est-ce qu'un framework ?

Un framework est un ensemble d'outils et de composants logiciels organisés conformément à un plan d'architecture et des patterns, l'ensemble formant ou promouvant un squelette de programme.

Détaillons un peu cette définition :

- **un ensemble d'outils** : on aborde directement l'aspect instrumental. Un framework n'est pas un pur concept mais un ensemble concret d'outils.
- **un ensemble de composants logiciels** : on dispose donc de « briques » logicielles toutes prêtes pour bâtir nos applications.
- **organisés conformément à un plan d'architecture et des patterns** : les outils et les briques logicielles ne sont pas empilés en désordre mais organisés selon une certaine logique, chaque élément étant bien à sa place. Cette architecture implique un modèle de développement, une sorte de guide, donc on ne dispose pas seulement d'éléments prêts à l'emploi mais en plus on nous accompagne dans la façon de les utiliser.
- **l'ensemble formant ou promouvant un squelette de programme** : en fait un framework nous offre l'essentiel de notre application, l'image du squelette est éloquente, on a une structure solide, bien établie, à nous d'ajouter de la vie,

II. Utilisation d'un framework :[E]

Un framework est, comme son nom l'indique en anglais, un « cadre de travail ». L'objectif d'un framework est généralement de simplifier le travail des développeurs informatiques en leur offrant une architecture « prête à l'emploi » et qui leur permette de ne pas repartir de zéro à chaque nouveau projet.

Les frameworks doivent être utilisés pour construire des applications Web. Toute demande qui comporte l'appel à une base de données, des formulaires, sessions, cookies ou l'appel à un service à distance comme twitter ou facebook gagnera en performance en partant d'un framework existant et utilisé par de nombreux autres systèmes.

En revanche, il n'est pas nécessaire d'utiliser un framework pour un site qui a seulement une ou deux pages.

II.1. Inconvénients de l'utilisation de frameworks :[G]

« Inconvénient » n'est peut être pas le terme approprié, mais il existe en tout cas quelques points de vigilance à observer lorsqu'il est question d'adopter un framework, notamment dans un contexte d'entreprise :

- ❖ **Courbe d'apprentissage :** Il faut d'abord savoir que tous les frameworks n'ont pas la même courbe d'apprentissage : certains frameworks permettent aux développeurs de monter en compétence rapidement, d'autres demandent un temps d'apprentissage beaucoup plus long

- ❖ **Connaissances préalables :**

Les frameworks implémentent généralement des design patterns et bonnes pratiques de développement : c'est très bien, mais une méconnaissance de ces concepts généraux (indépendants d'un langage ou d'un framework, comme par exemple MVC, les factories, les décorateurs, les singletons, etc.) est généralement synonyme de mauvaise compréhension/interprétation des principes de base des frameworks, et donc de choix techniques peu judicieux.

- ❖ **Couplage et dépendance :**

Il faut savoir que l'utilisation d'un framework est également synonyme de dépendance (d'une certaine manière, de couplage) vis-à-vis de ce framework. La réingénierie nécessaire au changement de framework est généralement très lourde et coûteuse (et proportionnelle à la taille du SI...).

III. Quelques critères objectifs dans le choix d'un framework : [D]

On rencontre souvent dans la question de savoir quel est le meilleur framework. La réponse la plus intelligente à donner est que le meilleur est celui qu'on connaît bien et qui répond à la fois à nos besoins et notre compétence. Il y a toutefois des critères objectifs dans le choix d'un framework

- ❖ **La documentation :** un framework mal documenté fait perdre plus de temps qu'il n'en fait gagner.
- ❖ **La communauté :** lorsqu'on rencontre une difficulté il est bon de recevoir une aide rapide pour ne pas rester bloqué longtemps sur un problème de détail.

- ❖ **La durée d'apprentissage** : un temps d'apprentissage trop long fait perdre l'intérêt d'un framework même si on factorise notre connaissance en multipliant les cas d'application.
- ❖ **La pérennité** : apprendre un framework pour apprendre quelques mois plus tard qu'il est abandonné est plutôt frustrant.
- ❖ **La richesse des outils** : certains framework sont intéressants mais nécessitent de nombreuses greffes pour devenir vraiment opérationnels.
- ❖ **La simplicité** : utiliser un framework et se retrouver à écrire un code plus complexe que celui qu'on aurait utilisé avec simplement du PHP n'est pas une opération judicieuse.
- ❖ **Les mises à jour** : on est jamais à l'abri d'une faille de sécurité, le fait d'utiliser un framework est un peu à double facettes : le code a été normalement bien écrit, validé, mais il est lisible par tout le monde et devient par le fait vulnérable, d'où la nécessité de procéder à des mises à jour en cas de découverte d'une faille.
- ❖ **Les standards de programmation** : le code d'un « bon » framework est écrit en suivant les « bonnes pratiques » et il nous invite à faire de même.
- ❖ **Les possibilités d'extension** : on doit pouvoir facilement étendre les fonctionnalités d'un framework sans détruire son organisation.

IV. Les différents types de frameworks:[F]

1. **Framework d'infrastructure système** :
Pour développer des systèmes d'exploitation, des interfaces graphiques, des outils de communication (exemple : Framework .Net, Struts).
2. **Framework d'intégration intergicielle(middleware)** :
Pour fédérer des applications hétérogènes. Pour mettre à disposition différentes technologies sous la forme d'une interface unique.
3. **Framework d'entreprise** :
pour développer des applications spécifiques au secteur d'activité de l'entreprise ;
4. **Framework de gestion de contenu**:
Sont les fondations d'un système de gestion de contenu pour la création, la collecte, le classement, le stockage et la publication de « biens numérisés ».

V. Quelques caractéristiques que peut fournir un framework :[E]

- ❖ **Couche d'abstraction de la base de données** : Un framework peut générer une couche d'abstraction ou interface entre le code de l'application et les appels natifs réalisés vers le serveur de base de données, permettant de prendre en compte les

caractéristiques des fournisseurs SQL. Ainsi on peut modifier la base de données SQL sans avoir à réécrire du code. Certains frameworks peuvent également gérer les relations de base de données automatiquement, ce qui vous évite d'avoir à écrire les requêtes de jointure.

- ❖ **Gestion des formulaires** : Symfony2 et CakePHP permettent au développeur de définir les propres formulaires en PHP.
- ❖ **Authentification** : La plupart des frameworks sont livrés avec un module d'authentification utilisateur générique. Ils gèrent la connexion et déconnexion, l'enregistrement, la gestion des sessions et des autorisations. Il est généralement très facile de modifier et ajouter vos propres champs personnalisés.
- ❖ **Internationalisation** : Les frameworks vous permettent de créer facilement des versions spécifiques suivant les langues qu'on souhaite gérer, grâce à l'internationalisation en natif dans ces outils.

VI. Un Framework PHP :

VI.1. Définition :[H]

Au moment d'industrialiser ses développements, toute entreprise cherche le moyen d'avoir la meilleure solution pour gérer son environnement. Cela passe donc bien entendu par le choix d'un EDI (environnement de développement intégré), le choix d'un gestionnaire de version, mais surtout par le choix d'un framework PHP. Celui-ci permet en effet de forcer les développeurs à utiliser des conventions dans leur façon de coder, et donc à n'importe quel développeur de pouvoir prendre en main l'ensemble du projet rapidement par la suite. Il permet aussi de guider les développeurs dans le choix technologique et d'architecture afin de pouvoir ajouter des « briques » au projet facilement par la suite.

Un framework PHP répond normalement à de nombreuses demandes des développeurs. Il doit par exemple prendre en compte nativement les éléments suivants :

- ❖ **Modèle MVC** : modèle vue contrôleur. Il permet de structurer une application en distinguant la partie présentation, la partie base de données et la partie applicative.
- ❖ **Templates** : Il permet de séparer le code applicatif de la présentation.
- ❖ **Cache** : Il permet de stocker les pages afin d'optimiser leur temps de chargement.
- ❖ **Gestion des SGBDR** : (Système de Gestion de Bases de Données Relationnelles) : Il doit pouvoir gérer plusieurs types de base de données. MySQL bien entendu, mais éventuellement PostgreSQL ou Oracle en fonction des besoins.
- ❖ **ORM** : Mapping de relation objet. Cela permet de gérer la base de données sous forme d'objets.
- ❖ **Scaffolding** : échafaudage. Permet de créer un espace d'administration d'un site sans aucun développement, uniquement à partir de l'ORM.
- ❖ **Conventions** : Oblige les développeurs à utiliser les mêmes conventions de codage afin d'avoir un code uniforme.
- ❖ **URL conviviales** : règles de redirection. Il doit pouvoir gérer les URLs facilement.

VI.2. Critères du meilleur framework PHP :[H]

Il convient avant tout de définir des critères avant d'élire le meilleur framework PHP. En voici quelques uns :

- **Prise en main** : dans tous les frameworks, il est nécessaire de passer quelques temps afin de le prendre en main. Cette période peut être plus ou moins longue (de quelques heures à plusieurs semaines).
- **Flexibilité** : le framework peut imposer des contraintes très fortes pour obliger tous les développeurs à développer de la même façon. Mais cela peut finir par être limitant.
- **Taille des projets** : les petits projets peuvent être développés avec un framework ultralight.
- **Documentation** : une documentation complète est un réel plus !
- **Performance** : certaines frameworks sont trop gourmands. Même le moindre « Hello World » peut nécessiter l'appel à plus de 100 fichiers différents.
- **Communauté** : un forum actif sera synonyme de personne prêt à aider en cas de problème
- **Évolutivité** : de quand date la dernière mise à jour ?

Tous ces critères peuvent être plus ou moins pondérés en fonction de ce qu'on attend d'un framework. De préférence un framework soit simple à prendre en main(en l'espace de quelques jours maximums), assez flexible et très bien documenté

VI.3. Les différents frameworks existants :



Figure III.1. : Les différents frameworks existants

❖ **CakePHP :**

Sa prise en main est rapide, mais il manque de fonctionnalité. Et sa communauté est de plus en plus réduite.

❖ **CodeIgniter :**

Ce framework est « léger » : il laisse beaucoup de souplesse au développeur. Cela peut être un point faible comme un point fort. Sa prise en main est extrêmement rapide (quelques heures), et la communauté est de plus en plus disponible. De plus on trouve sur Github des dizaines de bibliothèques pour gérer tous les besoins classiques d'un développeur.

❖ **Symfony :**

Ce framework est assez complexe : le développeur doit se plier à ses nombreuses contraintes, de plus celui-ci est assez lent. Il est recommandable dans le cas de projet conséquent (au delà de 1000j/h) et dans le cas où les développeurs ont déjà une expérience avec.

❖ **Zend Framework :**

Il a pendant très longtemps très peu évolué : la communauté ne pouvait que très difficilement contribuer au framework, la version 2 a du corriger cela. Par ailleurs il est très contraignant et demande aussi un long temps d'apprentissage.

Voici la courbe des recherches relative à ces frameworks durant les dernières années :

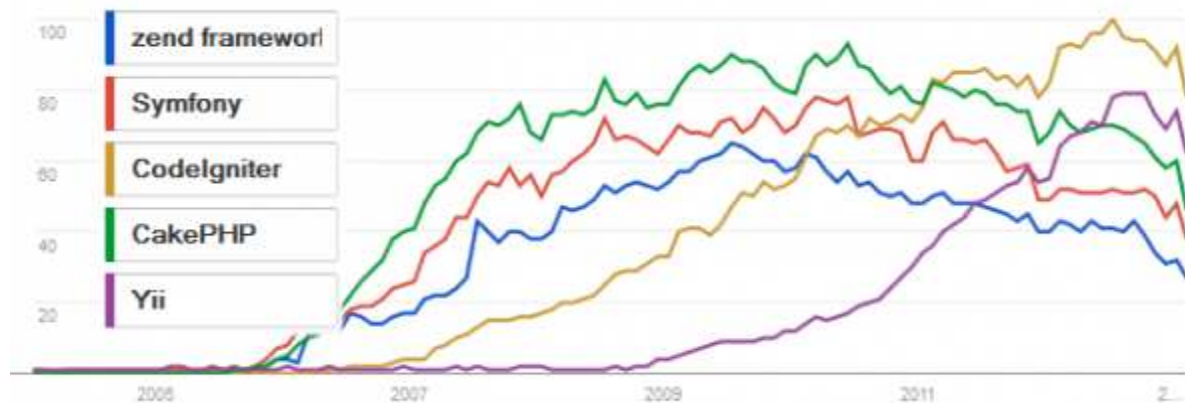


Figure III.2. : La courbe des recherches relative aux frameworks

On constate que ces frameworks ont décollé fin 2005. CakePHP a longtemps été le plus utilisé, mais CodeIgniter a pris les devants début 2011. Yii est actuellement en train de cartonner et pourrait même finir par dépasser CodeIgniter dans les prochaines années. Si on regarde les chiffres pour la France : Symfony est largement devant, mais CodeIgniter commence à monter en puissance.

VI.4. Comparaison des framework php : [Q]

Parmi la multitude d'excellents frameworks PHP nous avons choisi de comparer cinq d'entre eux, adoptant tous la structure MVC (Model-View-Controller ou Modèle-Vue-Contrôleur) :

Le Modèle : c'est le comportement de l'application.

La Vue : c'est l'interface manipulée par l'utilisateur.

Le Contrôleur : c'est le gestionnaire des événements de mise à jour de la vue et de synchronisation des informations par le modèle.

Ces frameworks sont parmi les plus connus et renommés :

- "Cake Php"
- "Code Igniter"
- "Jelix framework"
- "Framework Symfony"
- "Zend Framework"

Nom	PHP	Modèle	ORM	Template	Cache	Url conviviales	Validation de formulaire	Ajax	Extension	Générateur de code
Cake PHP	4/5	MVC	AR	PHP	0	0	0	0	0	0
CodeIgniter	4/5	MVC	AR	PHP	0	0	0	N	0	N
Jelix	5	MVC	DAO	jTpl	0	0	0	0	0	0
Symfony	5	MVC	Propel	PHP ou smarty	0	0	0	0	0	0
Zend	5	MVC	/	PHP	0	0	N	N	N	N

Figure III.3. : Tableau comparatif des 5 Frameworks

0 = oui

N= non

AR= Active Record

jTpl = Moteur de templates Jelix

MVC = Modèle-Vue-Contrôleur

Propel= Mapping objet relationnel (ORM)

DAO= Data Access Object

VI.5. Les avantages d'un Framework PHP : [I]

1. La pérennité :

Quand nous livrons un site internet, il est important de s'assurer qu'il soit stable et qu'il fonctionne (bien entendu), mais aussi que nous pourrions facilement assurer sa maintenance et sa longévité, puisqu'un projet réussi est avant tout un projet pérenne. En utilisant une méthodologie de développement unique sur l'ensemble de nos projets, on gagnera en temps de réalisation en acquérant au fur et à mesure des automatismes, et il sera d'autant plus facile pour un développeur de reprendre un projet en cours de route ou d'assurer son maintien.

2. Vitesse de développement :

C'est le principe de base d'un Framework, il va nous permettre de nous concentrer sur les parties les plus complexes de notre application en évitant de réinventer la roue à chaque fois.

Certains Frameworks comme Symfony2 nous permettront même de simplifier encore plus nos développements grâce aux lignes de commande pour générer automatiquement nos fichiers PHP, elle n'est pas belle la vie ?

3. Un support étendu :

On est plus seul, parce que l'utilisation d'un Framework nous donne un point commun avec l'ensemble des développeurs qui l'utilisent aussi, on fait désormais partie d'une communauté et ça c'est plutôt un bon point. Il y a 99% de chance pour que plusieurs utilisateurs aient eu le même problème avant nous.

VII. Le meilleur framework est celui que vous choisissez !

Parmi les bons frameworks, tous ont des avantages et des inconvénients. Pour choisir un framework le secret est donc de tester pour construire sa propre opinion. Symfony 2 ou CakePHP sont d'excellents frameworks. Mais ce ne sera peut être pas votre avis, avec votre expérience et votre manière de penser. Vous préférez peut être FuelPHP ou Zend Framework.

Dans la création de notre application nous avons fait le choix d'utiliser zend framework

VII.1. Définition de Zend framework :

C'est un framework PHP 5 créé par la société Zend Technologies, appelée aussi The PHP Company car fondée par les pères du PHP moderne : Andi Gutmans et Zeev Suraski. Il est distribué open source sous la New BSD license.

De part son histoire, Zend tient une place particulière dans l'univers des Frameworks PHP.

Bénéficiant d'une communauté massive à travers le monde, il est aujourd'hui à n'en pas douter le plus connu et utilisé des frameworks.

VII.2. Les principales caractéristiques de Zend Framework :

- Compatible PHP 5
- Conception totalement orientée objet
- Intégration de l'architecture MVC (Modèle-Vue-Contrôleur)
- Faible dépendance des composants entre eux, permet leur utilisation individuelle
- Puissant, extensible et modulaire
- Imposante communauté hyper-active

- Fortement adapté à l'environnement professionnel
- Développeurs externes respectant la licence open source
- Alliés et contributeurs hors normes : Microsoft, Google...

VII.3. Avantages Zend Framework

Zend Framework peut être considéré actuellement comme le leader des Frameworks PHP, sa souplesse est telle, qu'à partir d'une base quelconque, il est possible de créer une liaison avec d'autres frameworks ou composants.

D'une puissance, robustesse et fiabilité à toute épreuve, la conception de Zend Framework est particulière et le différencie des autres frameworks PHP.

En effet, ses composants ont la faculté d'être utilisés individuellement, car très peu dépendants des uns envers les autres.

Par contre, utilisés ensemble dans la librairie standard de Zend Framework, ils donnent au framework sa redoutable puissance.

De même, bien que l'architecture MVC soit présente nativement dans Zend Framework, elle n'est pas imposée mais optionnelle.

Toujours à la pointe du développement, de multiples composants permettent l'accès aux services web les plus modernes.

Le principal moteur du projet reste évidemment Zend Technologies, mais certaines sociétés extérieures ont développé leur propres composants et les ont mis ensuite à la disposition de la communauté. Parmi ces contributeurs exceptionnels, il convient de citer Google et Microsoft, deux des plus importants acteurs du web.

Zend Framework est en perpétuel développement, et chaque entreprise intéressée peut développer ses composants, à la seule condition de les mettre gracieusement à la disposition des utilisateurs, dans le respect de la licence.

Le concept PHP reste l'esprit de Zend Framework : libre, ouvert, simple, efficace et puissant

Conclusion :

Les frameworks de développement ont de multiples avantages, et sont l'une des évolutions les plus importantes des pratiques de conception et développement informatique de ces 20 dernières années, les bons développeurs d'aujourd'hui sont généralement d'excellents intégrateurs de solutions de différents niveaux de granularité.

Un framework apportera beaucoup avec un investissement relativement mineur comparé aux bénéfices tirés.

Les bons frameworks ne sont pas que des boîtes à outils, des bibliothèques de modules réutilisables. Par leur incitation au développement selon les bonnes pratiques de l'état de l'art, ils permettent à leurs utilisateurs de développer des applications plus robustes, plus évolutives et plus industrialisables.

Chapitre IV

Analyse et conception

Introduction

Dans le but d’une meilleure organisation et une bonne maîtrise du travail, tout processus de développement d’application ou système informatique doit suivre une méthode ou démarche bien définie.

Dans notre application, nous procéderons à une modélisation avec l’UML, étendu pour le web, en décrivant les cas d’utilisation qui servent d’outil d’analyse et de la conception du système, ainsi on élabore les diagrammes de séquence et les diagrammes de classes.

La figure suivante donne la représentation graphique de la démarche de modélisation choisie pour la conception de notre application :

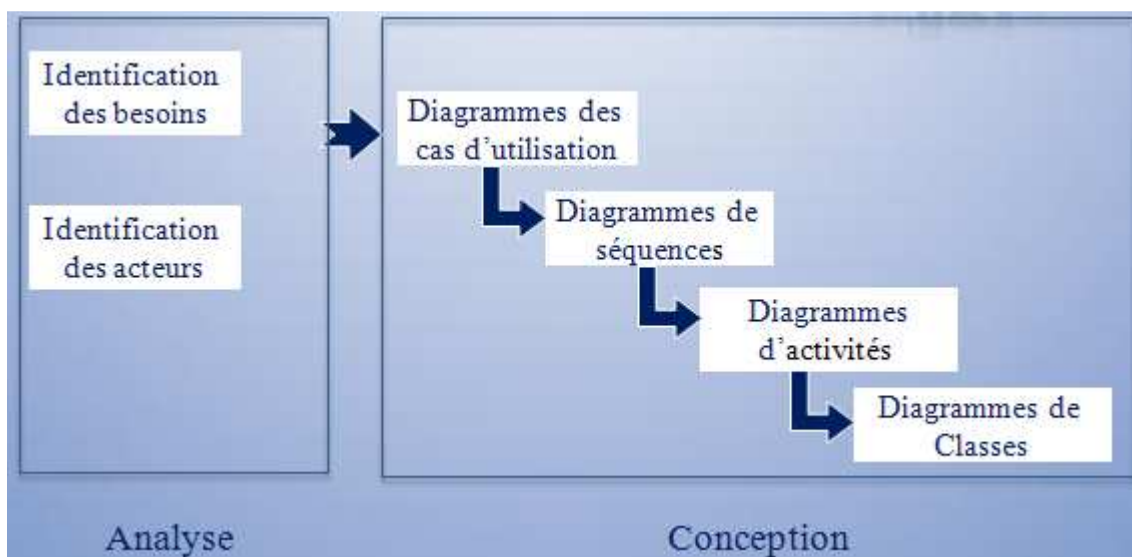


Figure IV.1. La démarche de modélisation et application

I. Présentation de l’UML: [8]

I.1. Définition

UML (Unified Modeling Language) est un langage de modélisation unifié (ou universel) et non pas une méthode.

UML a été conçu pour permettre la modélisation de tous les phénomènes de l’activité de l’entreprise (processus métier, systèmes d’informations, systèmes informatiques, composants logiciels...) employés par la suite.

I.2. Extension de l'UML pour le web :

Une extension d'UML définit un ensemble de stéréotypes et de contraintes, qui rend possible la modélisation d'application web.

- **Une stéréotype :** C'est une extension du vocabulaire d'UML, il permet d'associer une nouvelle signification à un élément du modèle. Les stéréotypes peuvent être appliqués après que tous les éléments du modèle et sont habituellement représentés par une chaîne de caractères entre guillemets (« »).
- **Une étiquette :** C'est une extension des propriétés d'un élément. Elle permet la description d'une nouvelle propriété d'un modèle. Elle est représentée dans un diagramme par une chaîne de caractères entre chevrons (<>).
- **Une contrainte :** C'est une extension de la sémantique d'UML. Elle prescrit la règle que le modèle doit vérifier pour être qualifié. Elle est représentée par une chaîne de caractères entre accolades ({ }).

Ces stéréotypes et ces contraintes sont appliqués sur des applications Web, permettant ainsi de les représenter au sein du même modèle et sur les mêmes diagrammes que ceux qui décrivent le reste du système.

Le principal élément spécifique des applications Web étant la page Web , plusieurs stéréotypes, qui lui sont destinés , sont décrits plus loin ainsi que d'autres , conçu pour les éléments tels que les cadres , les cibles et les formulaires représentant eux aussi des composants architecturaux significatifs.

I.3. La modélisation UML :

Schématiquement, UML peut être défini sur trois plans :

- Concepts dont la sémantique est complètement et formellement définie (Cas d'usage, scénarios, Objets, classe, associations, messages, opérations, état et transition)
- Diagrammes, utilisés pour spécifier les besoins et les systèmes.
- Mécanismes d'extension, pour intégrer aux processus des stéréotypes spécifique aux différentes méthodes et/ou contextes applicatifs.

I.3.1. Les diagrammes UML : [9]

Le langage UML est décrit par un méta-modèle, c'est-à-dire une représentation simple, mais rigoureuse, de ses constituants élémentaires, avec leurs règles d'utilisations et de syntaxe. Un diagramme est défini par les auteurs d'UML comme la représentation graphique d'un ensemble sélectionné de constituants UML.

UML possède 9 diagrammes de modélisation, tels que :

- Diagramme de cas d'utilisation
- Diagramme d'objets
- Diagramme de classes
- Diagramme de composants
- Diagramme de déploiement
- Diagramme de collaboration
- Diagramme de séquence
- Diagramme d'états-transitions
- Diagramme d'activités

Chacun de ces diagrammes étant dédié à la représentation des concepts particuliers d'un système logiciel. Ils sont répartis sur trois axes du niveau conceptuel :

- Fonctionnel.
- Structurel.
- Temporel.

Chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel.

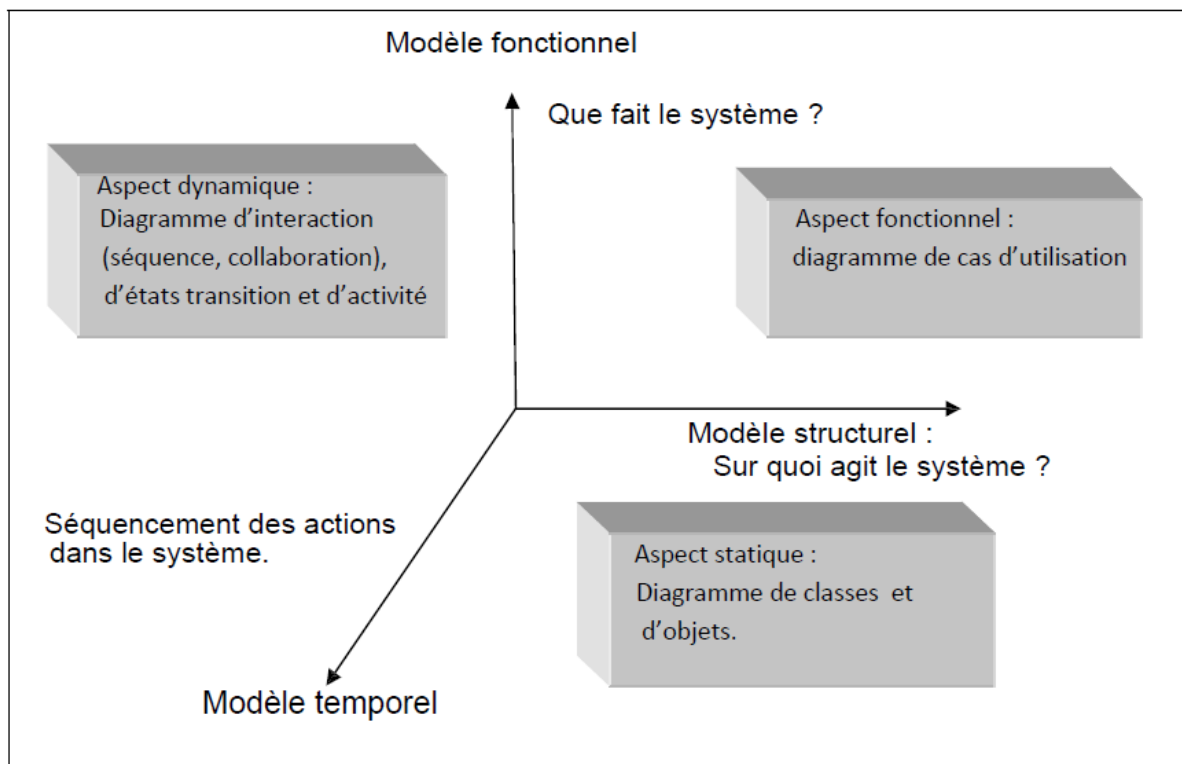


Figure IV.2. Les trois composantes de la modélisation UML

Une classification courante distingue les diagrammes qui traduisent la structure et les diagrammes qui présentent le comportement du système étudié.

I.4. Diagramme de contexte : [10]

Le diagramme de contexte est un modèle conceptuel qui permet d'avoir une vision globale des interactions entre le système et les liens avec l'environnement extérieur. Il permet aussi de délimiter le champ d'études.

Dans notre cas le diagramme de contexte est donné par la figure suivante :

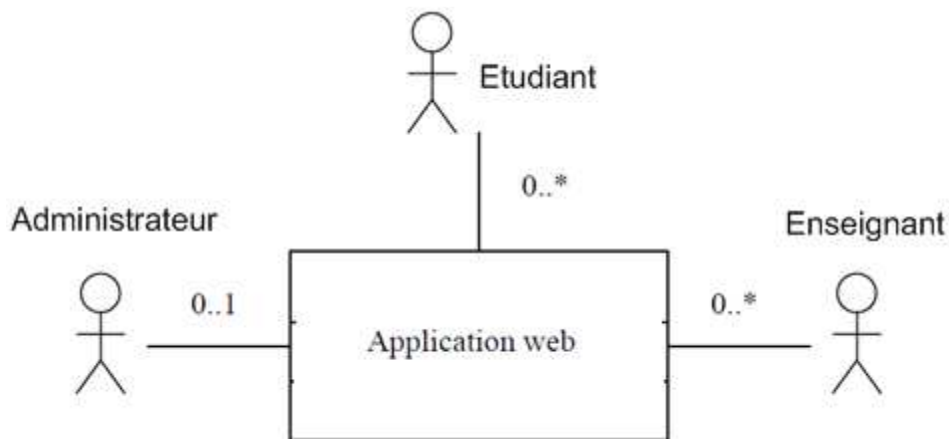


Figure IV.3. Diagramme de contexte de l'application

II. Analyse :**II.1. Identification des besoins :**

L'identification des besoins doit décrire sans ambiguïté le système à développer. L'expression des besoins doit donc proposer ce que le système devrait accomplir et non comment le faire.

II.2. Identification des acteurs :**a-Définition d'un acteur :**

Un acteur est toute entité externe (utilisateur, dispositif matériel ou autre système) qui interagit directement avec le système étudié, il peut consulter et/ou modifier directement l'état du système, en émettant et/ou recevant des messages susceptible d'être porteur des données, autrement dit un objet actif qui utilise les fonctions du système.

b-Extraction des acteurs de notre système :

Après l'étude de l'existant, on a pu identifier (extraire) les acteurs qui seront les futurs utilisateurs de notre application.

Dans notre application, nous recensons les acteurs suivant :

1. L'enseignant : C'est un rôle qui peut être joué par tous les enseignants du département.
2. L'étudiant : C'est un rôle qui peut être joué par tous les étudiants du département.
3. Administrateur : C'est un rôle qui peut être joué par la personne qui est chargée de l'administration de la base de données.

II. 3. Identification des cas d'utilisation :**II.3.1. Définition : [10]**

Un cas d'utilisation représente un ensemble de séquences qui sont réalisées par le système en réponse à une action d'un acteur et qui produisent un résultat observable. Les cas d'utilisation décrivent ce que le système devra faire sans spécifier comment le faire.

II.3.2. Spécification des tâches :

Pour chaque acteur nous spécifierons les tâches qu'il assure. Le tableau suivant résume ces tâches.

Acteurs	Taches
Etudiant	T1 : Se connecter au site T2 : S'authentifier. T3 : Consulter les notes. T4 : Consulter les plannings des examens. T5 : Consulter les annonces. T6 : Changer son mot de passe.
Enseignant	T1 : Idem que l'étudiant T7 : S'authentifier T8 : Gérer les notes des étudiants T9 : Annoncer les jours de consultation T10 : Consulter les plannings T11 : Changer son mot de passe
Administrateur	T1: Idem que l'étudiant T12 : S'authentifier T13 : Gérer les étudiants (ajouter, supprimer, modifier). T14 : Gérer les enseignants (ajouter, supprimer, modifier) T15 : Gérer les modules (ajouter, supprimer, modifier) T16 : Gérer les sections (ajouter, supprimer, modifier) T17 : Gérer les spécialités (ajouter, supprimer, modifier) T18 : Gérer l'affectation des enseignants. T19 : Gérer l'affectation des étudiants T20 : Annoncer les emplois de temps T21 : Annoncer les plannings des examens T22 : Changer son mot de passe.

Tableau IV.1 : Les différentes taches des acteurs

II.3.3. Spécification des scénarios :

Chacun des acteurs qu'on a définis effectue des taches et chaque tache est décrite par des scénarios qu'on résume comme suit :

Acteurs	Les taches	Scénarios
Etudiant	T1 : Se connecter au site	S0 : Saisir l'URL du site
	T2 : S'authentifier	S0 : S1 : Cliquer sur le lien « Etudiant » S2 : Remplir le formulaire et valider
	T3 : Consulter le relevé de notes	S0, S1, S2 S3 : Cliquer sur le lien « Relevé de notes »
	T4 : Consulter les plannings des examens	S0, S1, S2 S4 : Cliquer sur le lien« Plannings des examens » S5 : Choisir la section et la spécialité puis valider
	T5 : Consulter les annonces	S0, S1, S2 S6 : Cliquer sur le lien« Annonces »
	T6 : Changer son mot de passe	S0, S1, S2 S7 : Cliquer sur le lien « changement de mot de passe » S8 : Remplir le formulaire puis valider

Enseignant	T1, T7 : S'authentifier	S0 S9 : Cliquer sur le lien « Enseignant » S10 : Remplir le formulaire et valider
	T8 : Gérer les notes des étudiants	S0, S9, S10 S11 : cliquer sur le lien « gestion des notes » S12 : Choisir l'ajout, la suppression ou la modification S13 : Remplir le formulaire puis valider
	T9 : Annoncer les jours de consultation	S0, S9, S10 S14 : Cliquer sur le lien « Annonces » S15 : Editer l'annonce puis valider
	T10 : Consulter les plannings des examens	S0, S9, S10 S16 : Cliquer sur le lien « Plannings des examens » S17 : Choisir la section et la spécialité puis valider
	T11 : Changer son mot de passe	S0, S9, S10 S18 : Cliquer sur le lien « changement de mot de passe » S19 : Remplir le formulaire puis valider

Administrateur	T1, T12 : S'authentifier	S0, S20 : Cliquer sur le lien « Administrateur » S21 : Remplir le formulaire et valider
	T13 : Gérer les étudiants	S0, S20, S21, S22 : Cliquer sur le lien« Gestion des étudiants » S23 : Faire le choix (ajouter, consulter, supprimer) S24 : Remplir le formulaire et valider
	T14 : Gérer les enseignants	S0, S20, S21, S25 : Cliquer sur le lien « gestion des enseignants » S26 : Faire le Choix (ajouter, consulter, supprimer) S27 : Remplir le formulaire et valider
	T15: Gérer les modules (ajouter, supprimer)	S0, S20, S21, S28 : Cliquer sur le lien « gestion de modules » S29 : Faire le Choix (ajouter, consulter, supprimer) S30 : Remplir le formulaire et valider
	T16 : Gérer les sections (ajouter, consulter)	S0, S20, S21, S31 : Cliquer sur le lien « gestion de sections » S32 : Faire le Choix (ajouter, consulter, supprimer) S33 : Remplir le formulaire et valider
	T17: Gérer les spécialités (ajouter, consulter)	S0, S20, S21, S34 : Cliquer sur le lien « gestion de modules » S35 : Faire le Choix (ajouter, consulter, supprimer) S36 : Remplir le formulaire et valider
	T18 : Affecter les enseignants	S0, S20, S21, S37 : Cliquer sur le lien « Affecter les enseignants » S38 : Remplir le formulaire et valider
	T19 : Affecter les étudiants	S0, S20, S21, S39 : Cliquer sur le lien « Affecter les étudiants » S40 : Remplir le formulaire et valider
	T20 : Editer les emplois du temps	S0, S20, S21 S41 : Cliquer sur le menu « Gestion des emplois du temps » S42 : Faire le choix (ajouter, consulter, supprimer) S43 : Remplir le formulaire et valider
	T21 : Editer les plannings des examens	S0, S20, S21 S44 : Cliquer sur le lien« Gestion des emplois du temps » S45 : Faire le choix (ajouter, consulter, supprimer) S46 : Remplir le formulaire et valider
	T22 : Modifier leur mot de passe	S0, S20, S21 S47 : Cliquer sur le lien « Changement de mot de passe S48 : Remplir le formulaire et valider

Tableau IV.2 : Les scénarios des acteurs

II.3.4. Spécification de quelques cas d'utilisation :

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

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

✚ Etudiant :

➤ **Cas d'utilisation** : « Authentification »

Use Case : Authentification

Rôle : Etudiant

Scenarios : S0, S1, S2

Description:

1. L'étudiant clique sur le lien « Etudiant »
2. Le système affiche le formulaire d'identification
3. L'étudiant remplit le formulaire et valide
4. Le système affiche le message d'erreur si le formulaire est mal rempli sinon
5. Le système affiche l'espace étudiant

➤ **Cas d'utilisation** : « Consulter les plannings des examens »

Use Case : Consulter les plannings des examens

Rôle : Etudiant

Scenarios : S0, S1, S2, S4, S5

Description:

1. L'étudiant clique sur le lien « Consulter les plannings des examens »
2. Le système affiche le formulaire pour faire le choix (la section et la spécialité)
3. L'étudiant remplit le formulaire et valide
4. Le système affiche le planning

✚ Enseignant.

- **Cas d'utilisation** : « Gérer les notes des étudiants »

Use Case : Gérer les notes des étudiants (ajouter notes)

Rôle : Enseignant

Scenarios : S0, S9, S10, S11, S12, S13

Description:

1. L'enseignant clique sur le lien «ajouter des notes »
2. Le système affiche le formulaire
3. L'enseignant fait choisir la section, la spécialité, le module et valide
4. Le système affiche une liste des étudiants
5. L'enseignant saisit la note de chaque étudiant et valide
6. Le système affiche le message succès

- **Cas d'utilisation** : « Changer son mot de passe »


Use Case : Changer son mot de passe

Rôle : Enseignant

Scenarios : S0, S9, S10, S18, S19

Description:

1. L'enseignant clique sur le lien «Changement de mot de passe »
2. Le système affiche le formulaire de modification
3. L'enseignant remplit le formulaire et valide
4. Le système affiche le message d'erreur s'il y a une erreur sinon
5. Le système affiche le message succès

 Administrateur➤ **Cas d'utilisation** : « Gérer les étudiants»

Use Case : Gérer les étudiants

Rôle : Administrateur

Scenarios : S0, S20, S21, S22, S23, S24

Description :

1. L'administrateur clique sur le lien « Gestion des étudiants » et fait le choix (ajouter, supprimer, modifier)
2. Le système affiche le formulaire
3. L'administrateur remplit le formulait et valide
4. Le système affiche le message de succès.

➤ **Cas d'utilisation** : « gérer les modules »

Use Case : gérer les modules

Rôle: Administrateur.

Scenarios : S0, S20, S21, S28, S29, S30

Description :

1. L'administrateur clique sur le lien « Gestion des modules » et fait le choix (ajouter, supprimer, modifier) un module
2. Le système affiche le formulaire
3. L'administrateur remplit le formulait et valide
4. Le système affiche le message de succès.

III. La conception :

Le but principal de la conception est de rendre le modèle d'analyse réalisable sous forme logicielle, c'est ici que les abstractions métiers viennent pour la première fois, au contact de la réalité logicielle

Pour modéliser la conception de notre application, nous avons utilisé l'extension d'UML pour le web, en construisant les diagrammes des cas d'utilisation, les diagrammes de séquences suivis des diagrammes de classes.

III.1. Diagrammes des cas d'utilisation:

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un système logiciel.

Pour donner une vue graphique pour l'étape d'analyse nous avons utilisé les diagrammes des cas d'utilisation offerts par l'UML qui utilise les relations suivantes :

- Relation (include) :

Une relation d'inclusion d'un cas d'utilisation A par rapport à un cas d'utilisation B signifie qu'une instance de A contient le comportement décrit dans B, le cas d'utilisation A ne peut être utilisé seul.

- Relation (extend) :

Une relation d'extension d'un cas d'utilisation A par rapport à un cas d'utilisation B signifie qu'une instance de A peut être étendue par le comportement décrit dans B.

- Relation (use) :

Lorsqu'une ou plusieurs tâches sont utilisées régulièrement, on peut les factoriser dans un même use case et faire de telle sorte que d'autres use cases l'utilisent en pointant par une flèche.

Nous allons passer à la représentation des différents cas d'utilisation des différents acteurs

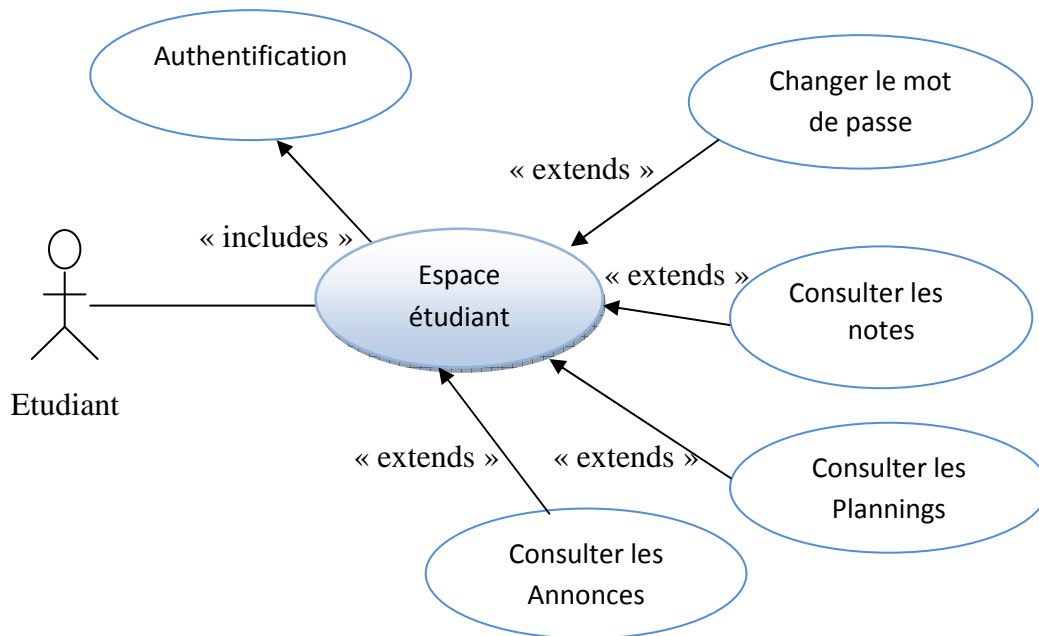


Figure IV.4: Diagramme global de cas d'utilisation relatif à l'étudiant

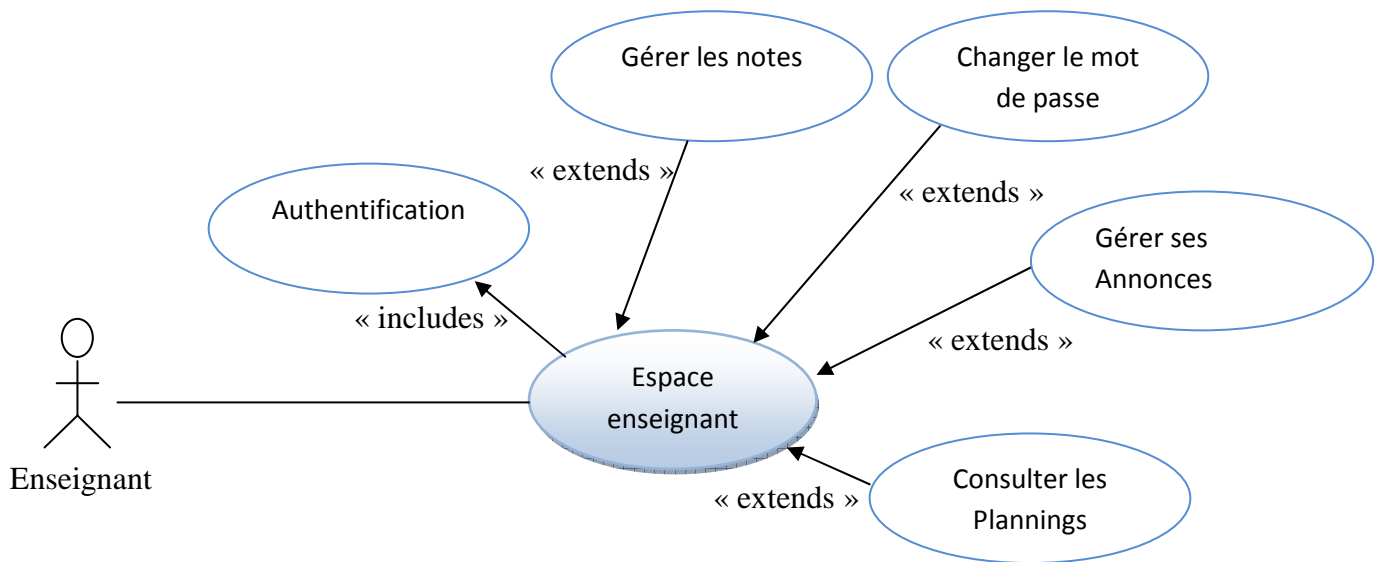


Figure .IV.5. Diagramme global de cas d'utilisation relatif à l'enseignant

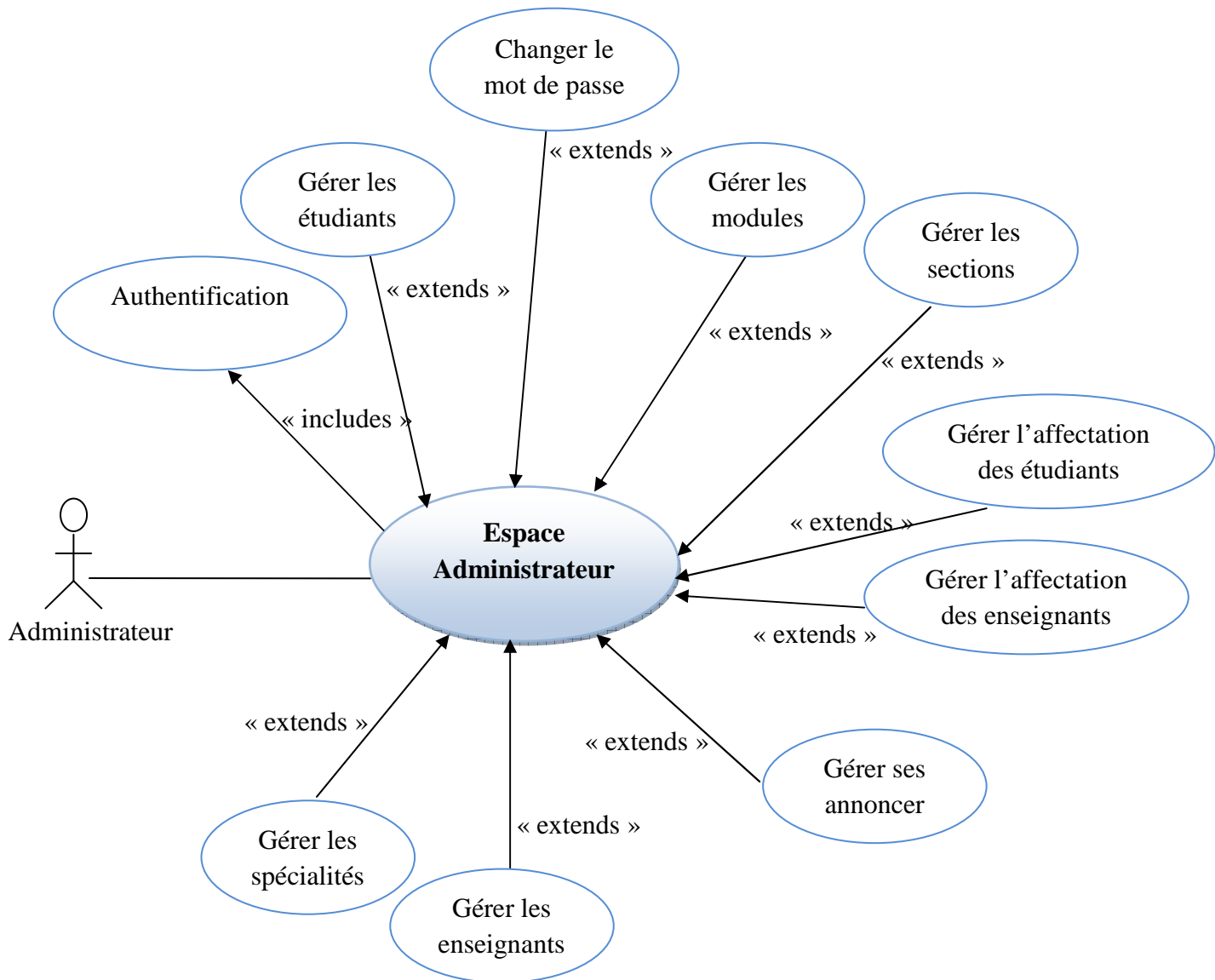


Figure IV .6 : Diagramme global de cas d'utilisation relatif à l'administrateur

III.2. Diagrammes de séquence de réalisation des cas d'utilisation :

Selon le formalisme UML (Unified Modeling Language), les diagrammes de séquences détaillés représentent graphiquement les interactions (acteur systèmes) dans l'évolution temporelle. Les acteurs interagissent donc avec les pages clients et les pages serveurs qui sont utiles pour les interactions quelques diagrammes de séquences cas utilisation important.

Etudiant :

➤ Cas d'utilisation « Authentification » :

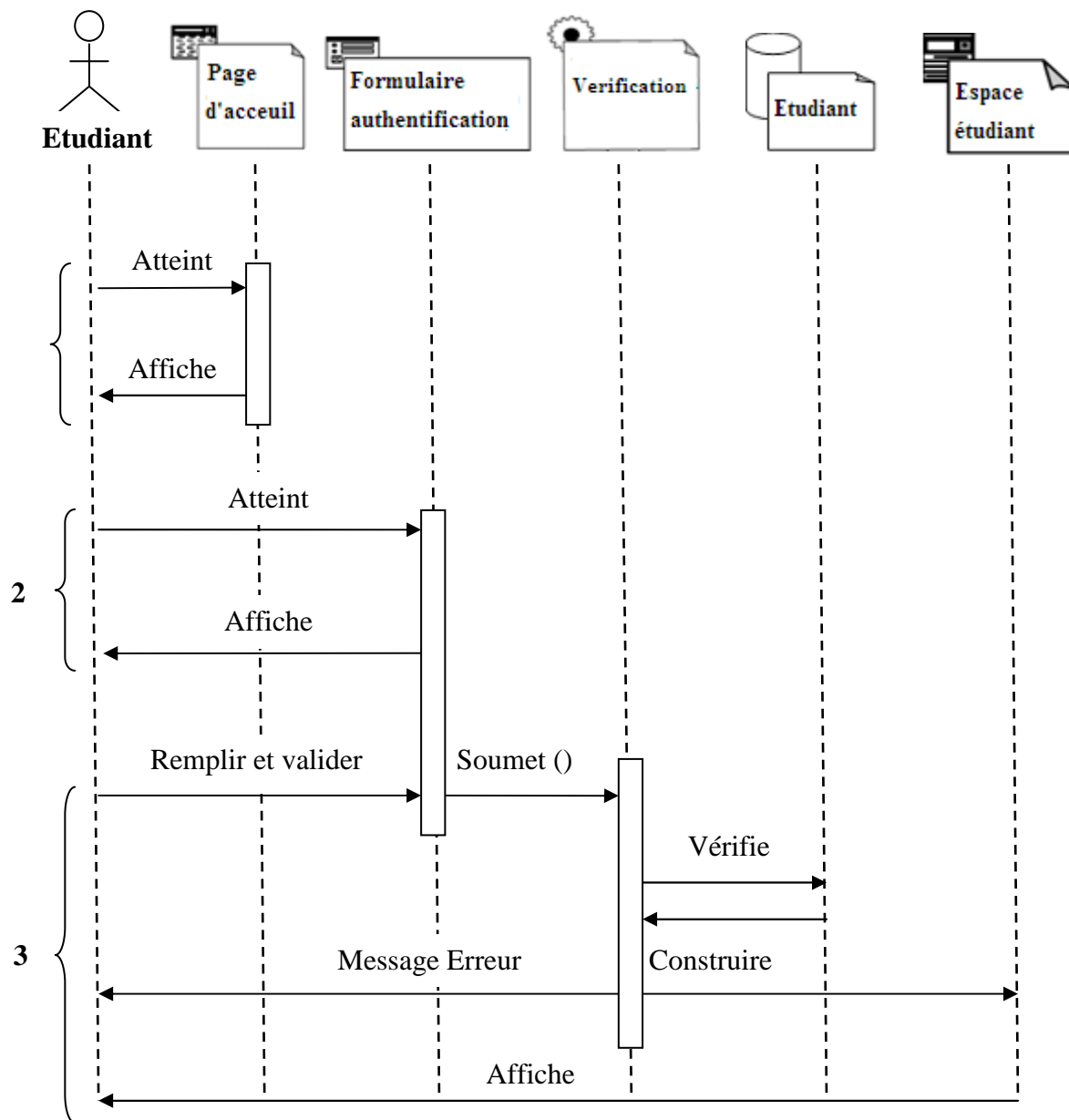


Figure IV.7 : Diagramme de séquence de réalisation de cas d'utilisation « Authentification étudiant »

- 1) L'acteur étudiant saisit l'url de l'application puis valide la saisie le système alors construit la page d'accueil de l'application puis l'affiche.
- 2) L'acteur étudiant clique sur le lien étudiant alors le système construit la page authentification étudiant puis l'affiche.
- 3) L'acteur étudiant saisit son login et son mot de passe dans le formulaire authentification puis le valide la saisie le système vérifie alors les données saisies dans la base de données ensuite il construit a page espace Etudiant puis l'affiche .Dans le cas ou les données login et /ou mot de passe ne sont pas justes le système construit un message d'erreur puis l'affiche.

Enseignant :

➤ **Diagramme de séquence de réalisation « enregistrement des notes des étudiant »**

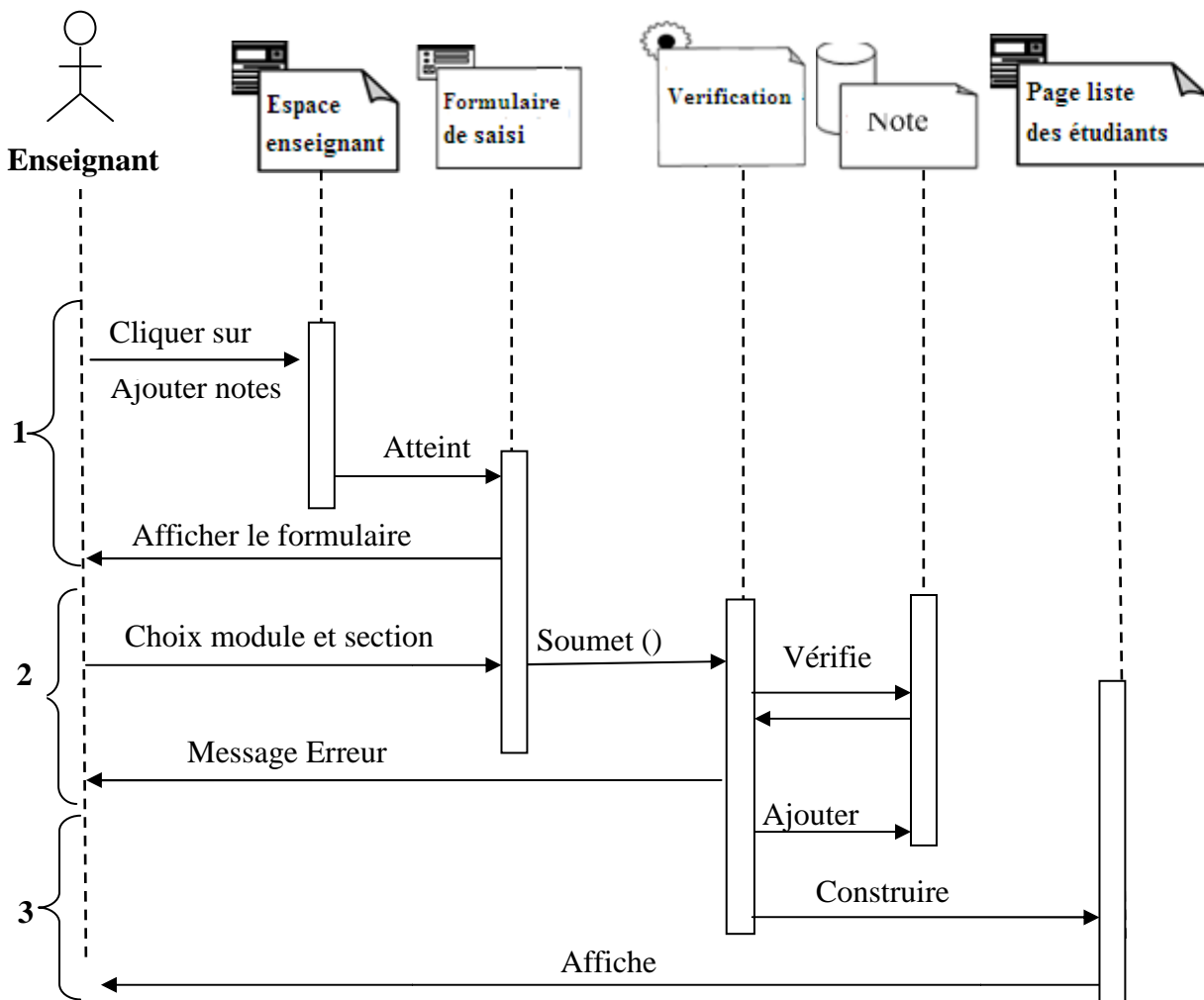


Figure IV .8 : Diagramme de séquence de réalisation de cas d'utilisation « enregistrement des notes »

- 1) L'acteur Enseignant clique sur le lien ajouter note qui se trouve dans son espace, le système construit alors le formulaire de choix du module, de la spécialité et la section puis l'affiche.
- 2) L'acteur enseignant choisit alors le module et la spécialité et la section puis le valide en cliquant sur le bouton Valider, le système cherche alors dans la base de données puis construit la liste des étudiants qui convient et l'affiche. Dans le cas ou les données ne sont pas justes le système construit un message d'erreur puis l'affiche.
- 3) L'acteur Enseignant saisit la note de chaque étudiant de cette liste et confirme en cliquant sur confirmer, le système affiche un message de succès pour l'enregistrement de ces notes.

➤ Cas d'utilisation « Changement de mot de passe »

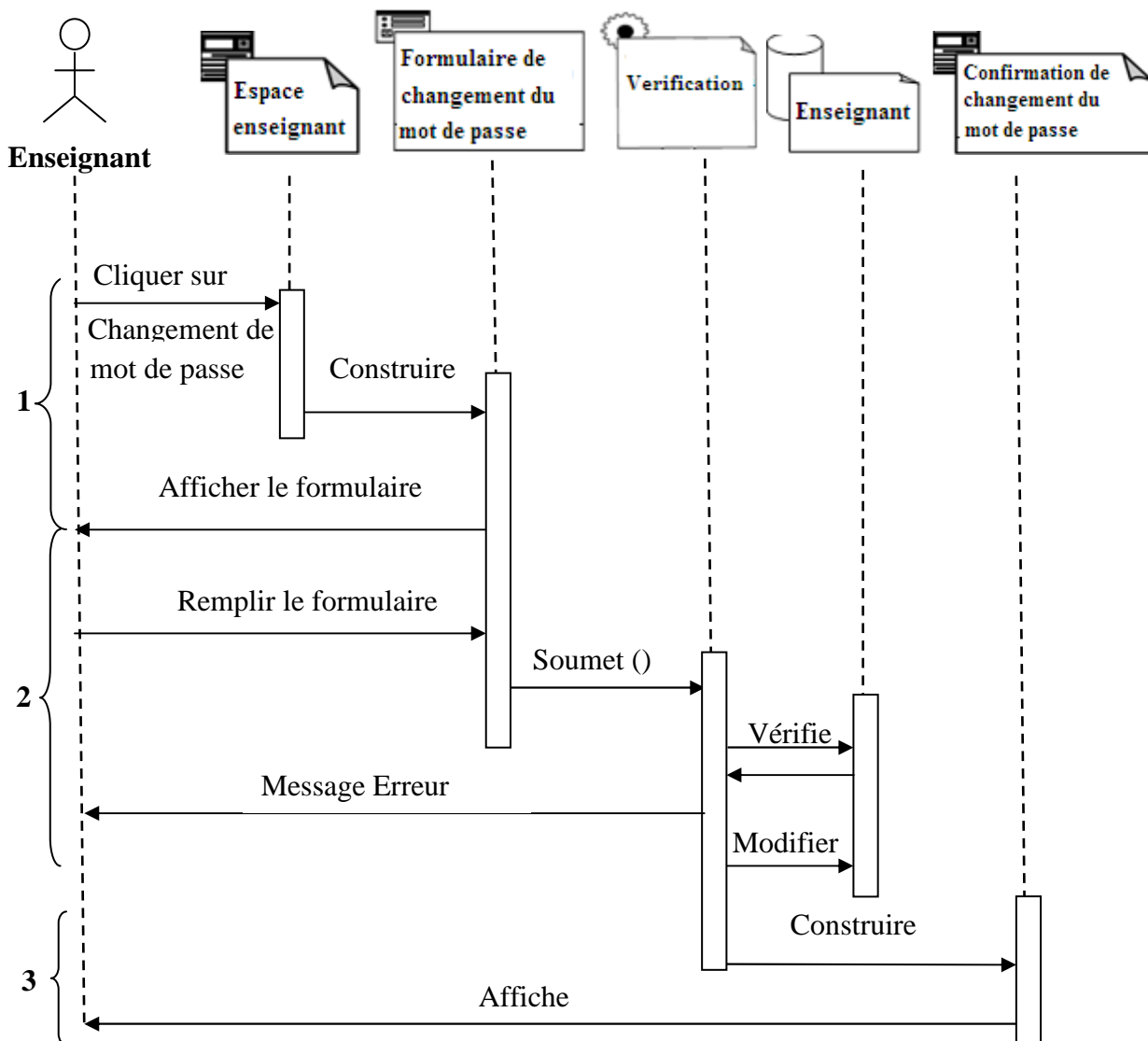


Figure IV.9 : Diagramme de séquence de réalisation de cas d'utilisation « Changement de mot de passe »

- 1) L'acteur enseignant clique sur le lien Changement de mot de passe qui se trouve dans son espace, le système affiche alors le formulaire du changement de mot de passe.
- 2) L'acteur enseignant saisit son login et son mot de passe actuelle ainsi que son nouveau mot de passe dans le formulaire du changement du mot de passe puis valide la saisie en appuyant sur le bouton valider , le système vérifie alors les données saisies dans la base de données puis procède a la modification du mot de passe si les données saisie sont exactes, sinon il construit un message d'erreur puis l'affiche.
- 3) Apres avoir modifié le mot de passe dans la base de données le système construit la page de confirmation puis l'affiche.

Administrateur :

➤ Cas d'utilisation « ajout d'un module »

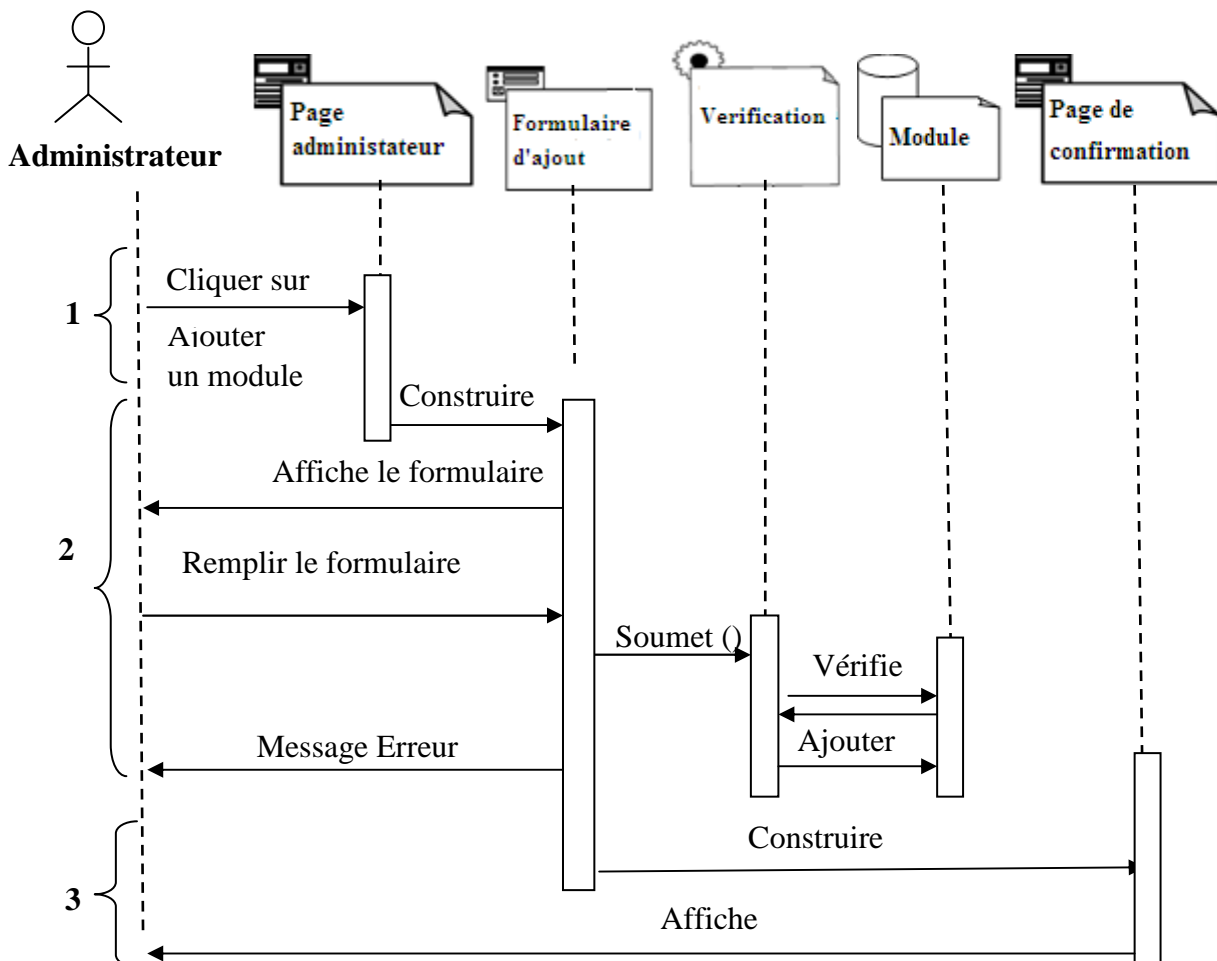


Figure IV.10 : Diagramme de séquence de réalisation de cas d'utilisation « Ajout d'un module »

- 1) L'acteur administrateur accède à son espace et sélectionne le lien « ajout d'un module »
- 2) Le système lui affiche un formulaire d'ajout, l'administrateur choisit et remplit le formulaire puis valide la saisie en appuyant sur le bouton valider.
- 3) Le système construit la page de confirmation puis l'affiche.

III.3. Diagrammes d'activités :

Un diagramme d'activités est semblable à un organigramme .Il se focalise sur le flot des activités impliquées dans un processus unique, il permet la représentation graphique du comportement d'une méthode ou le déroulement d'un cas d'utilisation.

❖ Diagramme d'activités du cas d'utilisation « S'authentifier » pour l'étudiant :

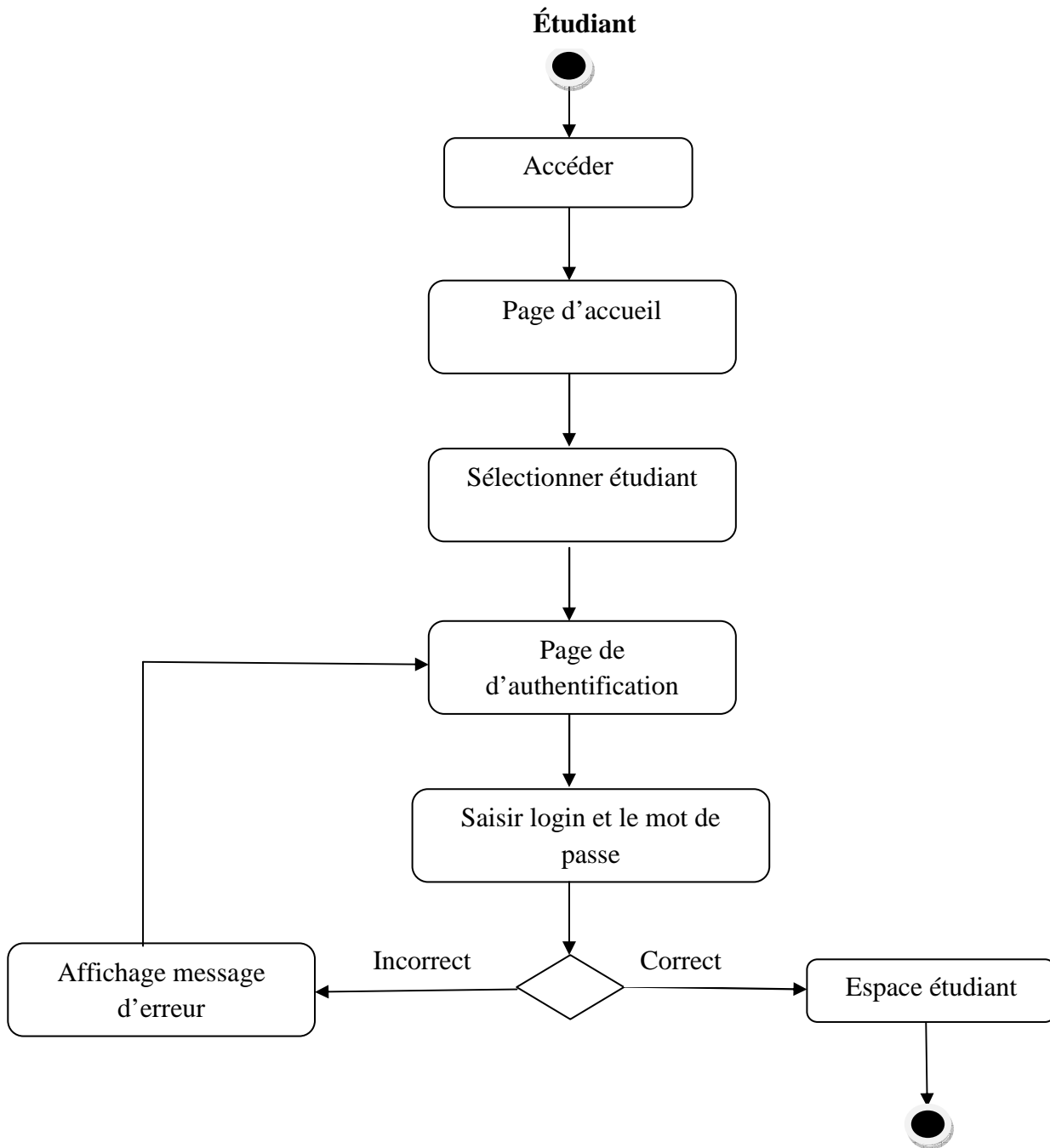


Figure IV.11 : Diagramme d'activités du d'utilisation « authentification étudiant »

❖ Diagramme d'activités du cas d'utilisation « Consulter les annonces » :

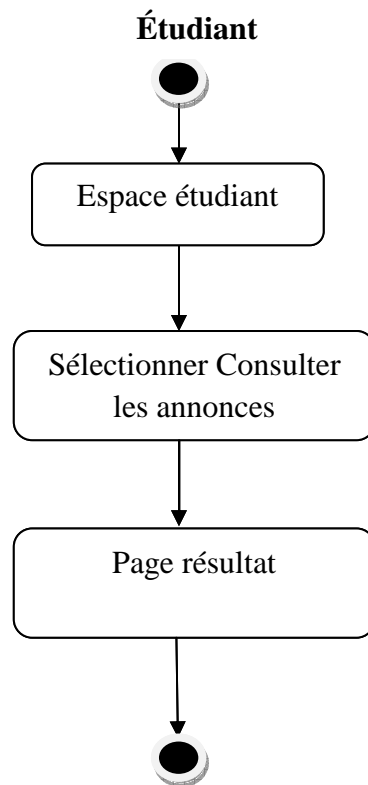


Figure IV.12 : Diagramme d'activités du cas d'utilisation « Consulter les annonces »

❖ Diagramme d'activités du cas d'utilisation « Enregistrer les notes des étudiants » :

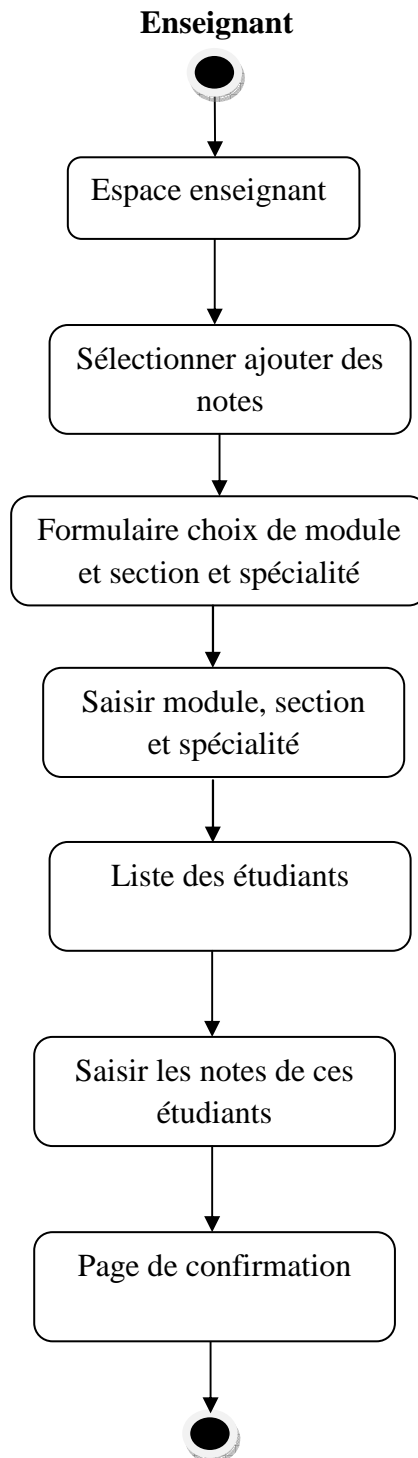


Figure IV.13 : Diagramme d'activités du d'utilisation « enregistrement des notes des étudiant »

❖ Diagramme d'activités du d'utilisation « Changement de mot de passe » :

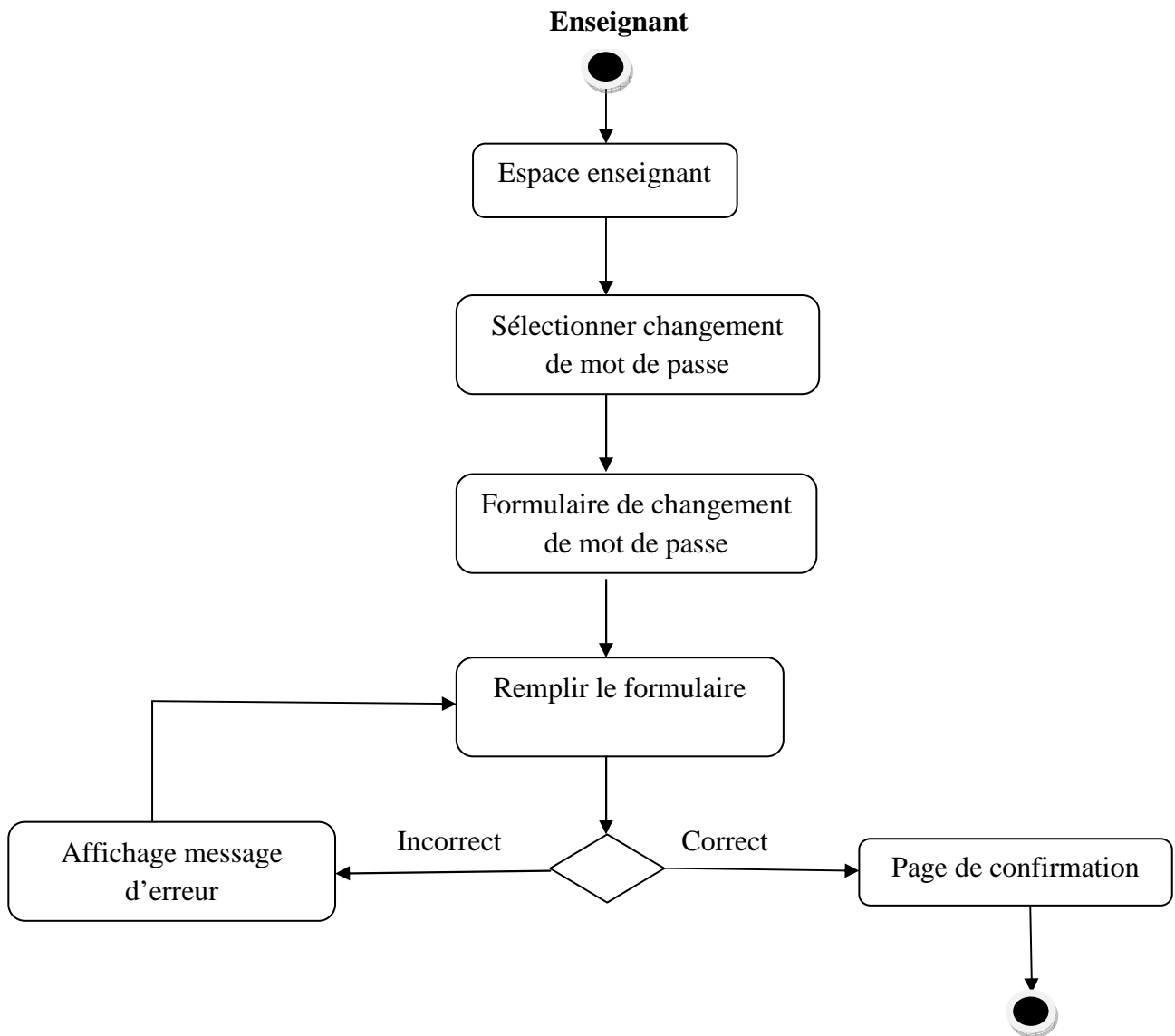


Figure IV.14 : Diagramme d'activités d'utilisation « changement de mot de passe »

❖ Diagramme d'activités du cas d'utilisation « ajouter un module » :

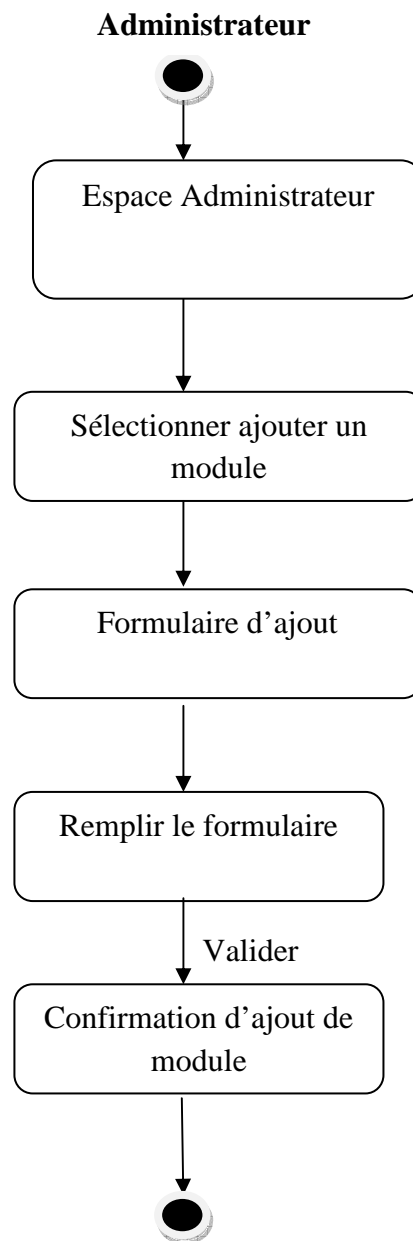


Figure IV.15 : Diagramme d'activités du cas d'utilisation « Ajout d'un module »

III.4. Diagrammes de classes :

Le diagramme de classe représente l'architecture conceptuelle du système : il décrit les classes que le système utilise, ainsi que leurs liens, que ceux-ci représentent un emboîtement conceptuel (héritage, marqué par une flèche terminée par un triangle) ou une relation organique (agrégation, marqué par une flèche, terminée par un diamant), Il est l'un des principaux diagrammes de l'UML.

Dans cette partie, nous allons élaborer un diagramme de classes montrant les relations entre les pages client et les pages serveur pour certains cas d'utilisation.

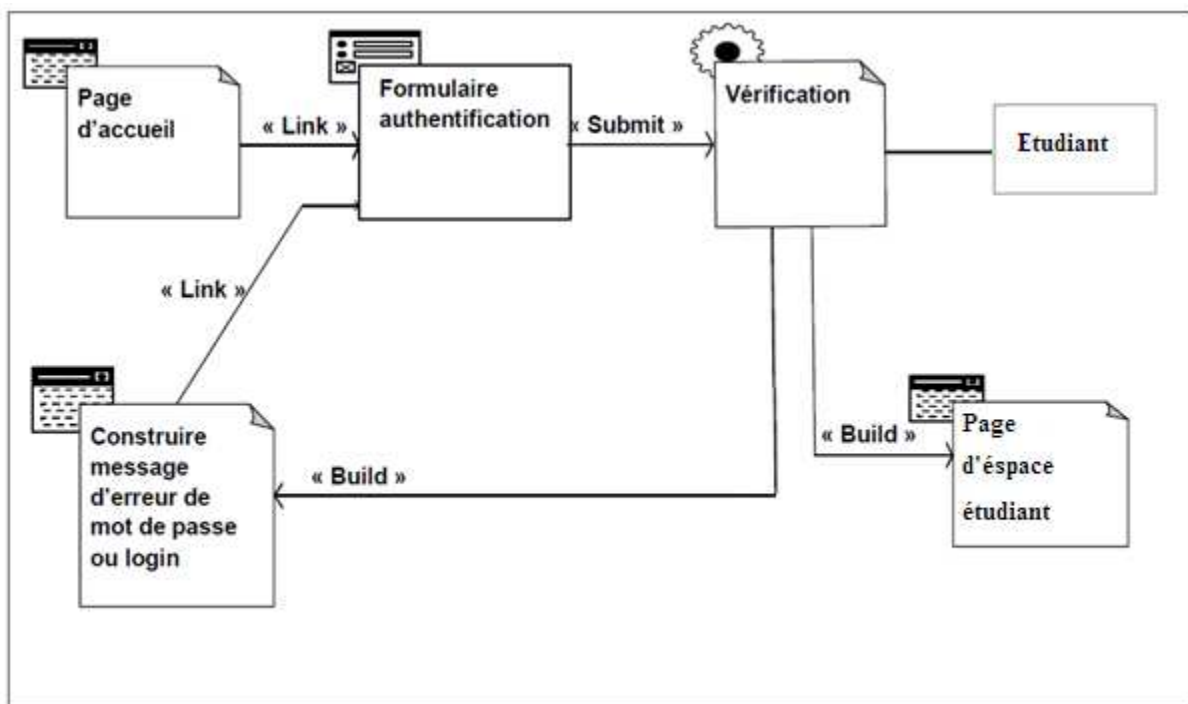


Figure IV.16. Diagramme de classe général du cas d'utilisation «Authentification »

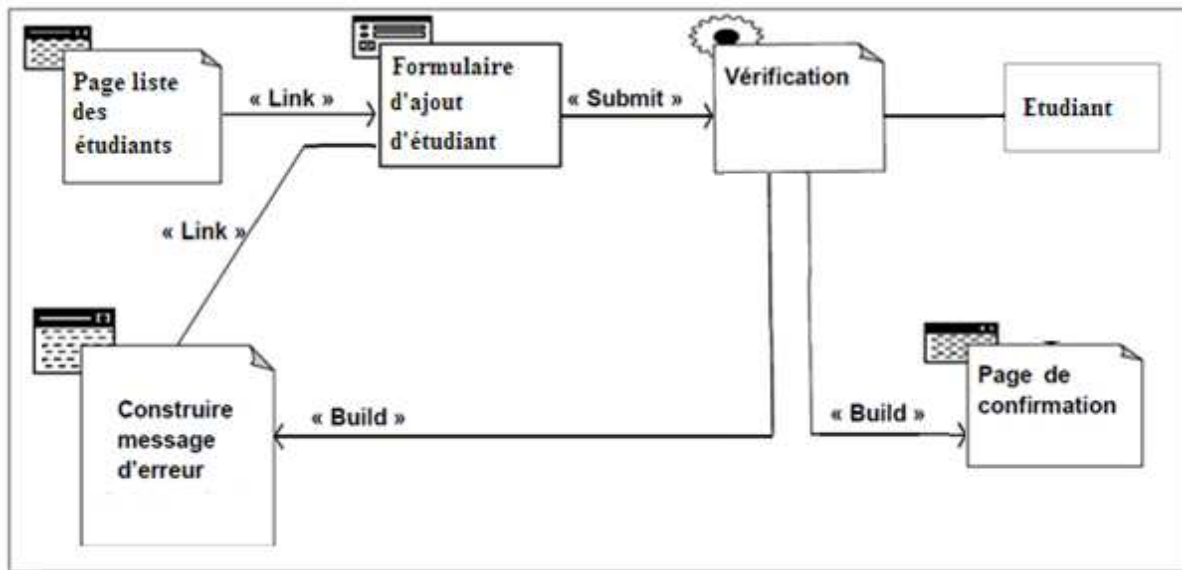


Figure IV.17. Diagramme de classe général du cas d'utilisation « Ajouter un étudiant »

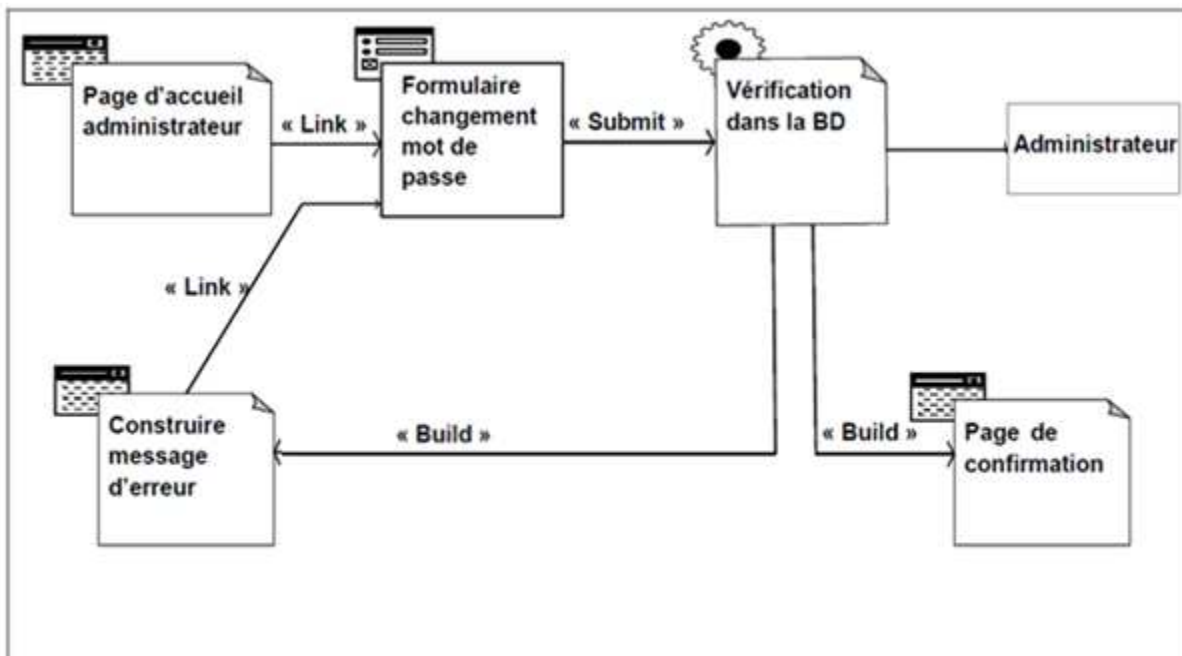


Figure IV.18. Diagramme de classe général du cas d'utilisation « Changer mot de passe »

III.4.1. Diagramme de classe global de données :

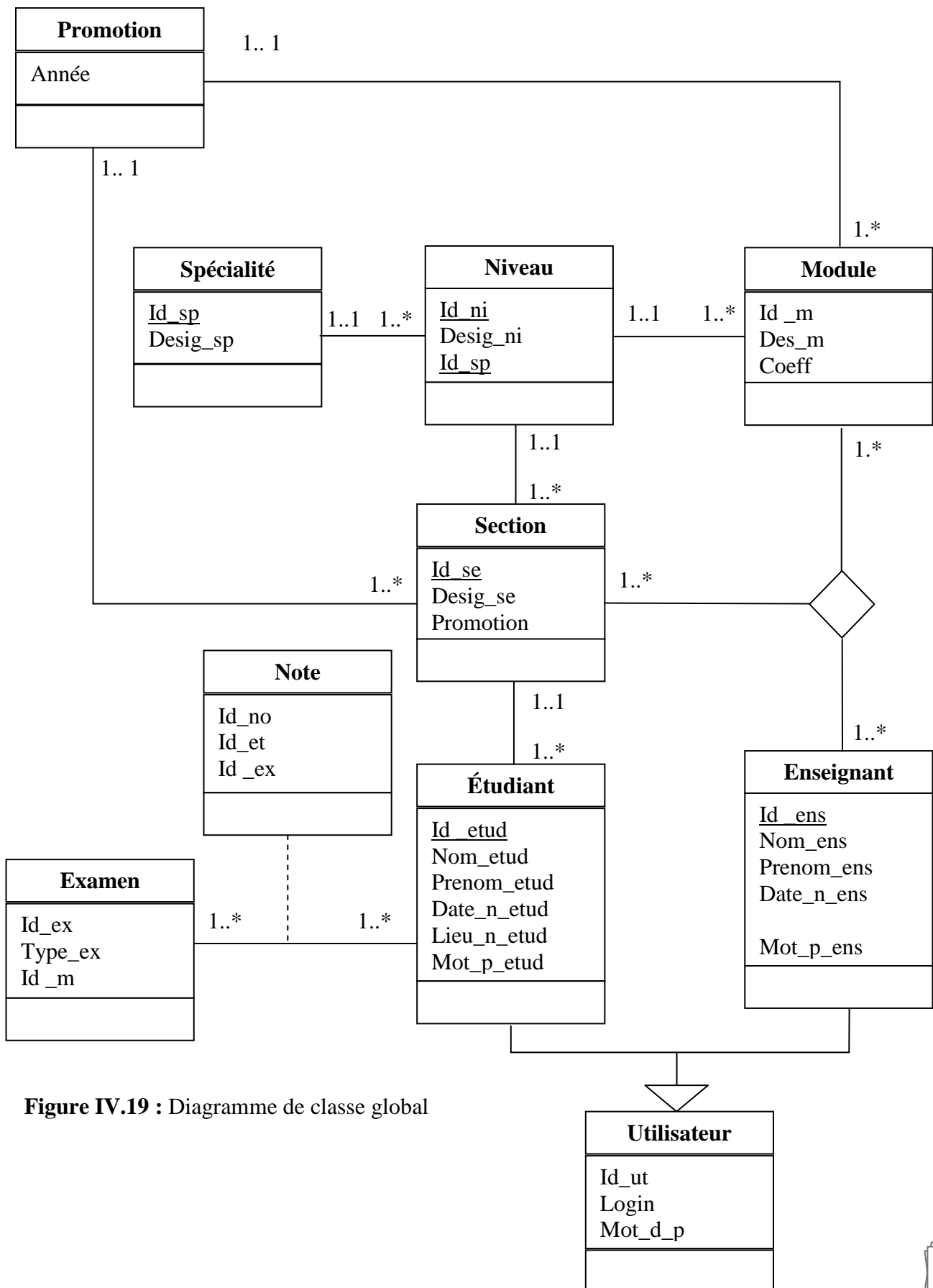


Figure IV.19 : Diagramme de classe global

III.5. Conception de la base de données :

Après avoir modélisé notre application Web avec les différents diagrammes offerts par le langage de modélisation UML, nous avons suivi le déroulement de chaque cas d'utilisation et nous avons utilisé les diagrammes de classes pour en extraire les différentes données nécessaires.

Pour l'implémentation de la base de données, on aura besoin d'élaborer un modèle logique de données qui contiendra :

III.5.1. Les tables de la base de données:

L'application dans tout ces cas d'utilisation manipule les données stockées dans la Base De Données, et elle contient les tables suivantes :

a. Le niveau logique de la base de données:

Après avoir élaboré le schéma conceptuel de la base de données, nous élaborons le schéma relationnel de la base de données en respectant les règles de passage du niveau conceptuel au niveau logique.

Le schéma logique :

Étudiant (Id _etud,Nom_etud,Prenom_etud,Date_n_etud, Lieu_n_etud,Mot_p_etud)

Enseignant (Id _ens,Nom_ens,Prenom_ens,Grade_ens,Date_n_ens,Lieu_n_ens
Mot_p_ens)

Module (Id _m,Des_m,Coeff)

Examen (Id_ex,Type_ex,Id _m)

Note (Id_no,Id_et,Id_ex>Note)

Spécialité (Id_sp,Desig_sp)

Section (Id_seDesig_se)

Promotion (année)

Affectation (Id_aff,Id_ens,Id_se,Id_sp,Id_mod)

Inscription (Id_ins,Id _etud,Id _section,Id _specialite, date_ins)

Séance (Id_se,Id_m,H_debut,H_fin,Salle,Type_se)

Niveau (Id_niDesig_ni)

b. Le niveau physique de données :

Le modèle physique de données est la traduction du modèle logique de données dans un langage de description de données spécifique au système de gestion des bases de données utilisé. Le modèle physique des données est l'implantation des données (tables) issues du modèle logique des données en machine afin d'aboutir à la description des fichiers de base de données.

❖ **Table étudiant :**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_etud	Identifiant d'étudiant	int(11)	Primaire
Nom_etud	Le nom d'étudiant	Varchar(255)	
Prenom_etud	Le prénom de l'étudiant	Varchar(255)	
Date_n_etud	La date de naissance d'étudiant	Date	
Lieu_n_etud	Lieux de naissance d'étudiant	Varchar(255)	
Mot_p_etud	Le mot de passe d'étudiant	Varchar(255)	

❖ **Table enseignant :**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_ens	Identifiant de l'enseignant	int(11)	Primaire
Nom_ens	Nom de l'enseignant	Varchar(255)	
Prenom_ens	Prénom de l'enseignant	Varchar(255)	
Grade_ens	Grade de l'enseignant	Varchar(255)	
Date_n_ens	La date de naissance d'enseignant	Date	
Lieu_n_ens	Lieux de naissance d'enseignant	Varchar(255)	
Mot_p_ens	Le mot de passe d'enseignant	Varchar(255)	

❖ **Table module :**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_m	Identifiant de module	int(11)	Primaire
Des_m	Description de module	Varchar(255)	
Coeff	Le coefficient de module	int(11)	

❖ **Table examen:**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_ex	Identifiant de l'examen	int(11)	Primaire
Type_ex	type de l'examen	Varchar(255)	
Id_m	Identifiant de module	int(11)	Etrangère

❖ **Table note:**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_no	Identifiant de la note	int(11)	Primaire
Id_et	Identifiant d'étudiant	int(11)	Etrangère
Id_ex	Identifiant de l'examen	int(11)	Etrangère
Note	La note d'étudiant	int(11)	

❖ **Table spécialité:**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_sp	Identifiant de la spécialité	int(11)	Primaire
Desig_sp	Désignation de la spécialité	Varchar(255)	

❖ **Table Section :**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_se	Identifiant de la section	int(11)	Primaire
Desig_se	Désignation de la section	Varchar(255)	

❖ **Table promotion:**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Année	Année de la promotion	Date	Primaire

❖ **Table affectation :**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_aff	Identifiant de l'affectation	int(11)	Primaire
Id_ens	Identifiant de l'enseignant	int(11)	Etrangère
Id_se	Identifiant de la section	int(11)	Etrangère
Id_sp	Identifiant de la spécialité	int(11)	Etrangère
Id_mod	Identifiant de module	int(11)	Etrangère
date_aff	Date de l'affectation	Date	

❖ **Table inscription:**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_ins	Identifiant d'inscription	int(11)	Primaire
Id_etud	Identifiant d'étudiant	int(11)	Etrangère
Id_section	Identifiant de la section	int(11)	Etrangère
Id_specialite	Identifiant de la spécialité	int(11)	Etrangère
date_ins	Date d'inscription	Date	

❖ **Table séance:**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_se	Identifiant de la séance	int(11)	Primaire
Id_m	Identifiant de module	int(11)	Etrangère
H_debut	Heur de début de la séance	Varchar(255)	
H_fin	Heur de fin de la séance	Varchar(255)	
Salle	Salle de la séance	Varchar(255)	
Type_se	Type de la séance (TD, TP, Cour)	Varchar(255)	

❖ **Table niveaux:**

Nom du champ	Description du champ	Type de donnée	Clé(s)
Id_ni	Identifiant de niveau	int(11)	Primaire
Desig_ni	Désignation de niveau	Varchar(255)	

Conclusion :

Ce chapitre est consacré à l'analyse et la conception de l'application avec le langage de modélisation UML. On a commencé par la description des différentes fonctionnalités et scénarios, puis on a passé à la description quelques diagrammes, enfin on a terminé ce chapitre par la création de la base de données.

Après avoir effectué l'analyse et la conception du système, on va passer à l'étape de l'implémentation de notre application, qui fera l'objet du prochain chapitre

Chapitre V

Réalisation et implémentation

Introduction :

L'implémentation d'application Web nécessite la maîtrise simultanée de nombreuses technologies.

Le présent chapitre portera sur la description de l'environnement et les outils de développement de notre application, et nous terminerons par la présentation de ses fonctionnalités à travers ses différentes interfaces.

I. Environnement de développement :**I.1. WAMP:**

WAMP est un acronyme signifiant: **Windows Apache MySQL PHP**, les rôles de chacun de ces composants sont :

➤ **Windows:**

Assure l'attribution des ressources aux composants suivants.

➤ **Apache :**

C'est un serveur web qui répond aux requêtes du client web (le navigateur), il a été créé dans la volonté de développer et de maintenir un serveur http sécurisé, efficace et évolutif pour les systèmes d'exploitation modernes.

Apache est un serveur multi plateforme et gratuit, son installation et sa configuration sont faciles, rapide et son utilisation pas compliquée. En l'associant avec PHP, il devient un serveur web dynamique. [J]

➤ **MySQL :**

C'est un système de gestion de base de données (SGBD), un serveur de bases de données relationnelle basé sur le langage de requêtes SQL (Structured Query Language) qui est un langage standard pour le traitement des bases de données.

Le serveur MySQL peut fonctionner en mode client/serveur, il assure à plusieurs utilisateurs l'utilisation simultanée d'une même base de donnée. C'est un serveur qui offre de nombreuses et puissantes fonctionnalités, ses possibilités de connexion, sa rapidité et sa sécurité font de lui un serveur très adapté à l'internet.

➤ **PHP :**

C'est un langage de script généralisé pour la production de contenu web dynamique. En l'associant avec un serveur Apache et une SGBD MySQL on obtient une solution fiable, puissante et gratuite pouvant être déployée sur de nombreuses plates formes telle que Windows.

II. Outils utilisés :**II.1. Langages utilisés :****II.1.1.HTML « Hyper Text Markup Language » « langage hypertexte à balises 'marqueurs' » : [K]**

Est un langage dit de « marquage » (de « structuration » ou de « balisage ») dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage. Les balises permettent d'indiquer la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents.

Le langage HTML permet notamment la lecture de documents sur Internet à partir de machines différentes, grâce au protocole HTTP, permettant d'accéder via le réseau à des documents repérés par une adresse URL.

L'HTML n'est pas un langage de programmation au sens classique du terme, mais est essentiellement un ensemble de règles qui indiquent à un navigateur comment afficher une page du web. Il est souvent utilisé conjointement avec des langages de programmation telle que JavaScript et des formats de présentation (feuilles de style en cascade).

II.1.2. CSS : « Cascading Style Sheets » « feuilles de style en cascades » [L]

Le langage CSS est utilisé pour décrire la présentation d'un document écrit en HTML ou XML. Il décrit les couleurs, la police...Les feuilles de style ont été mises au point afin de compenser les manques du langage HTML en ce qui concerne la mise en page et la présentation. Elles permettent de rendre un document plus agréable à lire et mieux présenté. Il a donc pour rôle d'améliorer la mise en page, de séparer la structure de sa présentation et d'améliorer l'accessibilité des pages web, son principe consiste à regrouper dans un même document des caractéristiques de mise en forme associées à des groupes d'éléments. Il suffit de définir un ensemble de caractéristiques de mise en forme par un nom et d'appeler ce dernier pour l'appliquer à un texte.

II.1.3. PHP « PreHypertext Processor » « préprocesseur hypertexte » : [M]

Le PHP est un langage qui a été conçu en 1994 par Rasmus Lerdorf. Les premières versions (qui restaient privées) étaient utilisées afin de savoir qui venait consulter son CV en ligne. La première version publique fut disponible au début de l'année 1995. Elle fut connue sous le nom de "Personal Sommaire Page Tools". Elle était composée d'un analyseur extrêmement simple qui ne reconnaissait que quelques macros spéciales et d'un petit nombre d'utilitaires couramment utilisés dans les pages web.

PHP est un langage utilisé dans des applications web pour écrire des scripts HTML. L'essentiel de sa syntaxe est empruntée aux langages C, Java et Perl, mais y est ajouté plusieurs fonctionnalités uniques. Le premier but de ce langage est de permettre aux développeurs web de concevoir rapidement des sites contenant des pages dynamiques.

II .1.4. SQL « Structured Query Language » « Langage de requêtes structuré » [N]

SQL est un langage informatique mis au point par IBM , destiné à interroger ou piloter une base de données. La première version commerciale fut disponible en 1979. MySQL, Oracle, Microsoft SQL Server, PostgreSQL... sont Quelques exemples de bases de données utilisant SQL.

II.1.5. Java : [11]

Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton de Sun Microsystems. Mais c'est également un environnement d'exécution.

Java peut être séparé en deux parties. D'une part, le programme écrit en langage Java et d'autre part, une machine virtuelle(JVM) qui va se charger de l'exécution du programme Java. C'est cette plateforme qui garantit la portabilité de Java. Il suffit qu'un système ait une machine virtuelle Java pour que tout programme écrit en ce langage puisse fonctionner.

➤ **Les Servlets (technique de développement) :**

Une servlet est un programme java qui fonctionne sur un serveur Web et dont le rôle consiste à apporter une réponse à une requête .Elle constitue l'outil idéal pour l'implémentation du tiers médian dans les applications 3-tiers.

Une servlet peut ainsi recevoir une requête envoyée par un navigateur Web, négocier les informations demandées avec une base de données et renvoyer le résultat de la requête au navigateur.

Pour exécuter une servlet, il suffit de taper son URL dans la zone d'adresse du navigateur ou de l'interroger dans une page Web.

Une servlet peut être invoquée plusieurs fois en même temps pour répondre à plusieurs requêtes simultanées.

La servlet se positionne dans une architecture Client /Serveur trois tiers dans le tiers du milieu entre le client léger chargé de l'affichage et la source de données.

➤ **Les JSP :**

JSP=Java Server Pages, c'est un fichier contenant du code HTML et des fragments de code Java exécutés sur le moteur de servlets comparable aux langages cotés serveur de type PHP, ASP , les pages JSP sont converties en servlet par le moteur de servlet lors du premier appel à la JSP.

II.1.6. JavaScript:

Java Script (initialement appelé Live script) a été développé par Netscape puis a été repris par la firme Sun. Les scripts sont des instructions (des lignes de code) interprétées par les navigateurs Netscape et internet explorer.

Lé JavaScript n'a cessé d'évaluer avec les versions des navigateurs. Java Script est un langage de Script qui s'incorpore au langage HTML de présentation des pages web .Les script vont permettre de rendre une page web plus dynamique

II.2. Logiciels utilisés :

➤ **WampServer : [O]**

WampServer est une plate-forme de développement Web sous Windows. Il vous permet de développer des applications Web dynamiques à l'aide du serveur Apache, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin et SQLite Manager pour gérer plus facilement vos bases de données.

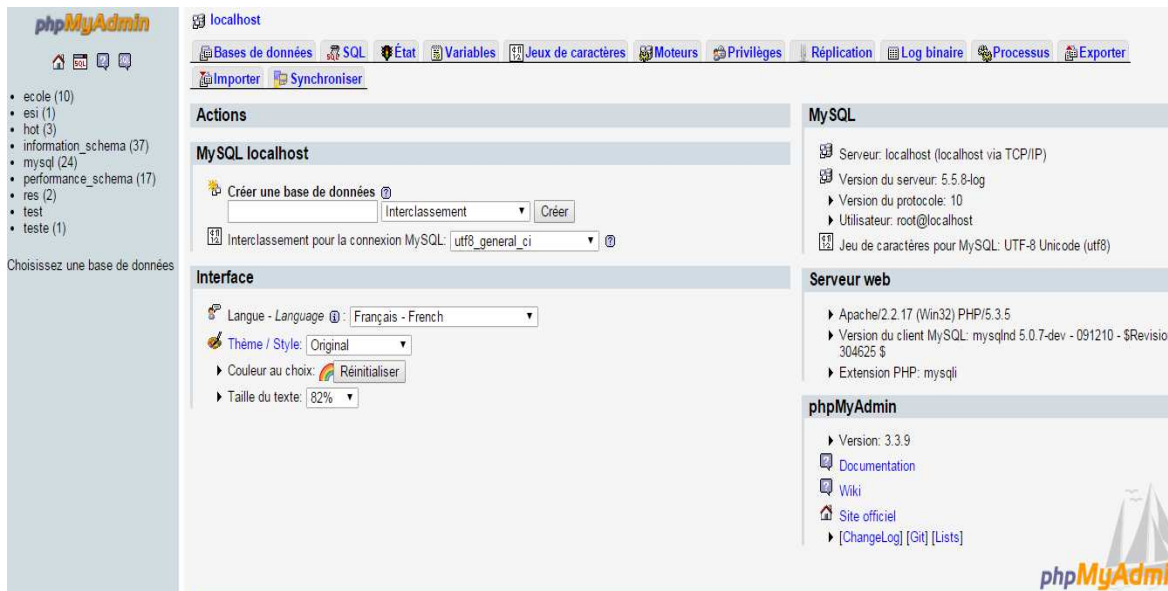


Figure V.1. Interface de WampServer

➤ **Dreamweaver : [P]**

Dreamweaver est un éditeur de pages web c'est à dire en quelques sortes un traitement de texte spécialisé dans la création de pages web. Lorsque vous utilisez ce type de logiciel, le HTML nécessaire à la création de vos pages est automatiquement généré. Néanmoins, la connaissance d'un tel logiciel ne vous dispensera pas (à terme, et pour l'utilisation de notions telles que CSS ou PHP) de l'étude du langage HTML.

Dreamweaver permet aussi d'obtenir une partie du Javascript nécessaire à quelques contrôles.

Ce logiciel vous permettra de concevoir un site web de manière efficace et conviviale intégrant des pages web traditionnelles, mais aussi des styles.

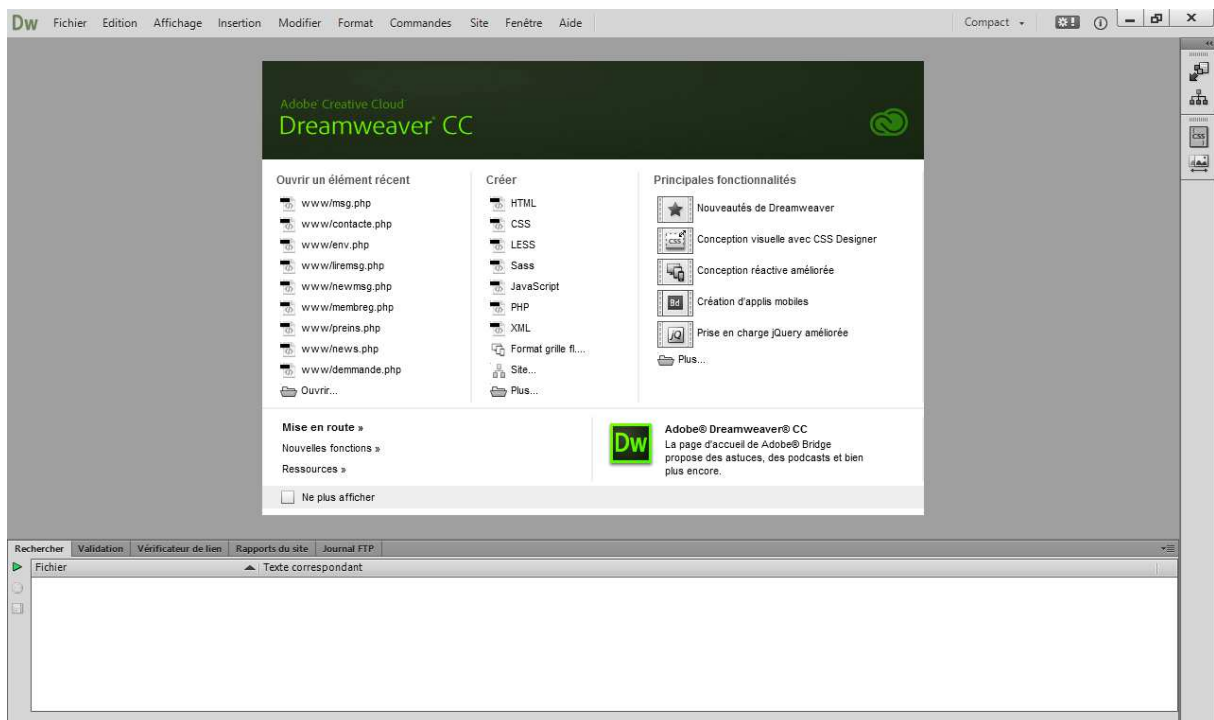


Figure V.2. Interface de Dreamweaver CC

III. Présentation de quelques interfaces :

Nous allons montrer dans cette partie les interfaces principales de notre application

III.1. Page d'accueil :

C'est la première page téléchargée et visualisée par les utilisateurs c'est la page principale du site car elle donne une idée générale sur les fonctionnalités de notre application.

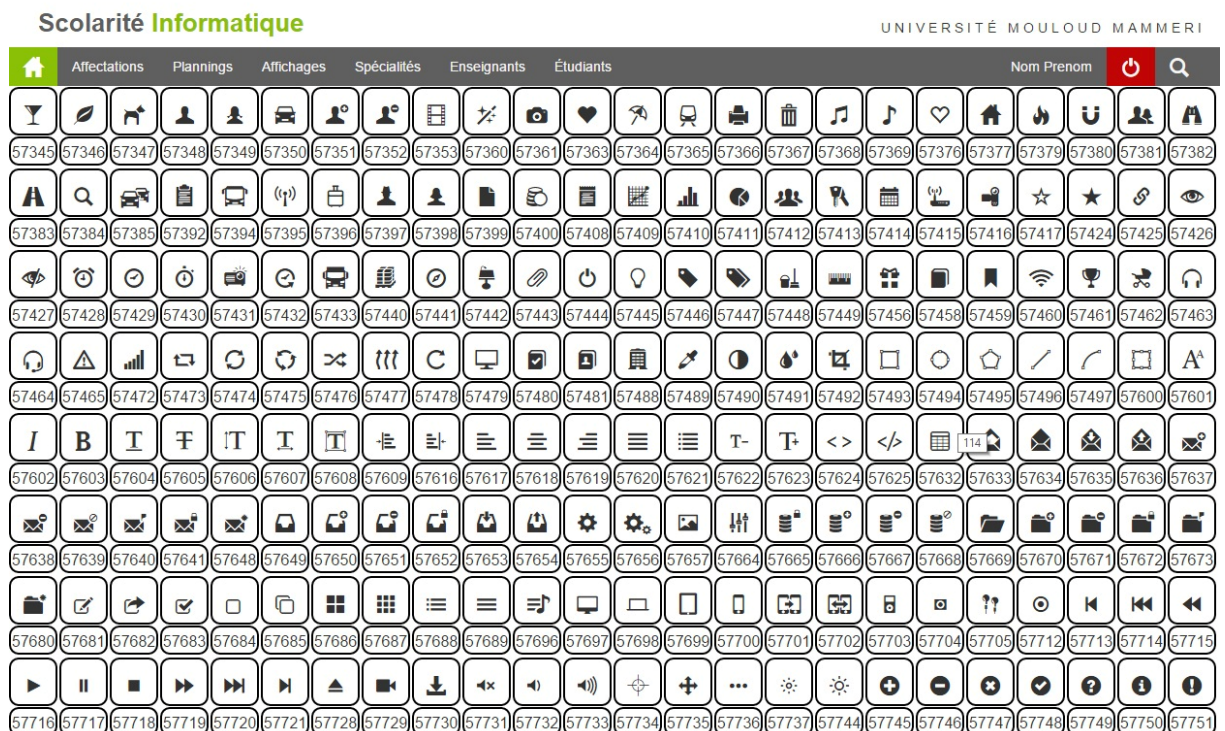
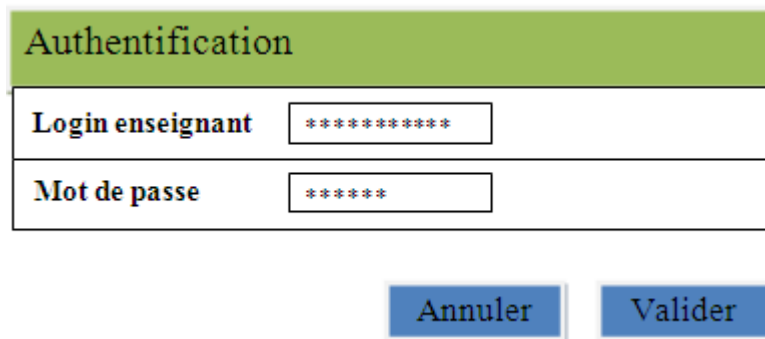


Figure V.4. Page d'accueil

III.2. La page Authentification:

Cette page est nécessaire pour l'authentification des différents acteurs qui vont utiliser cette application (étudiant, enseignant, administrateur), ce qui va leur permettre d'accéder à leur espace personnel et pour cela il suffit de cliquer sur le lien authentification ensuite remplir le formulaire (Login, Mot de passe) et valider la saisie.



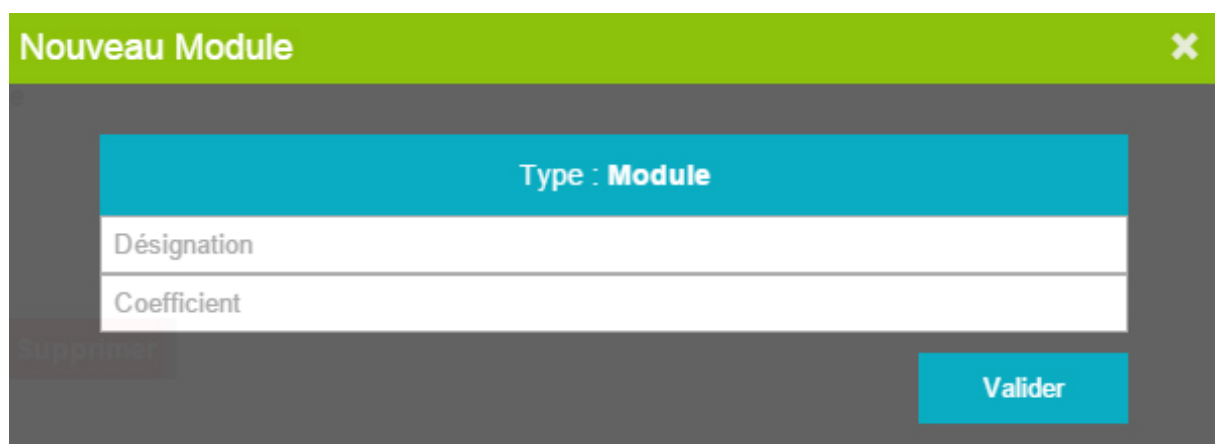
Authentification	
Login enseignant	*****
Mot de passe	*****

Annuler Valider

Figure V.5 Page authentification

III.3. Page d'ajout d'un module :

Cette page peut être atteinte par l'administrateur, après avoir effectué l'authentification, il pourra ajouter un module où cette interface le montre :



Nouveau Module	
Type : Module	
Désignation	
Coefficient	

Valider

Figure V.6. Ajout d'un module

III.4. Page d'ajout d'une spécialité :

L'administrateur peut accéder à la liste des spécialités existantes, et peut éventuellement ajouter une nouvelle.



The screenshot shows a modal window titled "Nouvelle Spécialité" with a close button (X) in the top right corner. The form contains a text input field labeled "Désignation" and a dropdown menu with the text "- Sélectionnez Licence/Master -". A blue "Valider" button is located at the bottom right of the form.

Figure V.7. Ajout d'une spécialité :

III.5. Page d'affectation d'un enseignant:

L'administrateur peut voir toute la liste des enseignants existants, et faire des nouvelles affectations.



The screenshot shows a modal window titled "Affectation" with a close button (X) in the top right corner. The form contains three dropdown menus: the first is labeled "S1", the second is "- Sélectionnez une Seance -", and the third is "- Sélectionnez un Enseignant-". A blue "Valider" button is located at the bottom right of the form.

Figure V.8. Affectation d'un enseignant

Conclusion :

Dans ce chapitre, nous avons présenté l'environnement du développement de notre application et les outils utilisés pour l'implémenter, puis nous avons présenté les différents langages de programmation utilisés, ainsi que quelques fonctionnalités qu'elle effectue notre application.

Conclusion

générale

Conclusion générale

L'objet de notre travail dans ce projet de fin d'études est d'assurer une partie importante et indispensable pour la mise en place d'une application web pour la gestion de scolarité du département d'informatique de l'université Mouloud Mammeri de Tizi Ouzou.

Notre problématique de départ était de répondre au mieux aux attentes des différents utilisateurs : l'étudiant, l'enseignant, l'administration. Par ailleurs, il fallait faciliter l'accès à l'information et éviter les traitements manuels source d'erreurs importantes.

Et pour cela on a essayé d'apporter des solutions à cette problématique comme : la consultation des notes, l'enregistrement et la modification des notes, l'édition de relevé de notes....

La réalisation de ce travail nous a permis d'acquérir de nouvelles connaissances sur les technologies web.

Bibliographie
&
webliographie

- [1] David James Clarke, CNE Guide de formation ,1995.
- [2] Cours réseaux : Le modèle OSI Frédéric Jacquenod Réseaux Edition : DUNOD 2007
Stéphane Canet.
- [3] François LAISSUS, Cours d'introduction à TCP/IP, 2002
- [4] L'architecture client/serveur [François Marie-Colonna, Octobre 2002]
- [5] Lionel Bocherberg « Internet et commerce électronique » 2ème Edition
- [6] D.Gonzalez « Les services de l'internet » édition 2002
- [7] Paolo Zanella et Yves Ligier « Architecture et technologie des ordinateurs » Edition
DUNOD 4ème édition 2005
- [8] Pascal Roques, « UML par la pratique », Edition EYROLLES 2ème édition 2004.
- [9] Grady Booth et al « Le guide de l'utilisateur d'UML», Edition EYROLLES 2003.
- [10] Pierre-allain Muller, « Modélisation objet avec UML », Edition EYROLLES 1997.
- [11] « Servelets et Java Server page le guide du développeur », P.Y.Saument, A.Mirecourt.

[A] <http://fr.Wikipedia.org/Wiki/Université>

[B] http://fr.wikipedia.org/wiki/Universit%C3%A9_Mouloud_Mammeri_de_Tizi_Ouzou

[C] [http://fr.wikipedia.Org/wiki/Web 2.0](http://fr.wikipedia.Org/wiki/Web_2.0)

[D] <http://laravel.sillo.org/presentation-du-framework-laravel/>

[E] <http://www.blog-nouvelles-technologies.fr/3484/pourquoi-devriez-vous-utiliser-un-framework-php/>

[F] <https://fr.wikipedia.org/wiki/Framework/>

[G] <http://www.formation-django.fr/generalites/framework.html>

[H] <http://www.inmediaveritas.com/comparatif-framework-php/>

[I] <https://www.eenov.com/developpement-open-source/pourquoi-utiliser-un-framework-php/>

[J] <http://www.ratiatum.com>

[K] <http://www.henri-ruch.ch/HTML/HTML.asp>

[L] <http://www.graphic-evolution.fr/definition/feuille-de-style-41.html>

[M] <http://damien.faucillon.free.fr/tutorial/PHPdef.htm>

[N] http://www.futura-sciences.com/fr/definition/t/informatique-3/d/sql_2524

[O] www.6ma.fr/tuto/wamp+server-454

[P] <http://pedagene.creteil.iufm.fr/internet/definit.htm>

[Q] <http://www.comparatif-framework.com/>