



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOULoud MAMMERI TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE



DEPARTEMENT D'INFORMATIQUE

Mémoire

de fin d'étude

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

Spécialité : Conduite de Projets Informatiques

Thème :

***Systeme Informatique pour la gestion
de la scolarité de l'UMMTO
Cas : Nouveaux départements des
sciences et technologies TAMDA***

Proposé et dirigé par :

M^r M.DAOUI

Réalisé par:

M^r D.LALIAM

M^r F.TIDJEDAM

Promotion: 2018/2019

« *REMERCIEMENTS* »

Remerciements

Nous tenons à exprimer notre profonde gratitude à notre promoteur, monsieur DAOUI Mehammed pour nous avoir encadrés durant cette année, Nous tenons, également, à lui exprimer notre profonde reconnaissance pour le temps précieux qu'il nous a consacré ainsi que pour ses conseils judicieux et généreusement prodigués.

Que les membres du jury trouvent ici nos plus vifs remerciements pour avoir accepté d'honorer par leur jugement notre travail.

Un grand merci aussi à toute personne qui de près ou de loin a contribué à ce que ce modeste travail voit le jour.

Dédicaces

Je dédie ce modeste travail

À Mes très chers parents qui m'ont toujours apportés leur soutien inconditionnel tout au long de mes études, ainsi qu'à mes frères et mes sœurs.

À La mémoire de ma très chère et regrettée sœur que Dieu ait son âme.

À Saïd Minouh et Alicia.

À mes collègues de travail.

Et enfin à toutes les personnes ayant contribué de près ou de loin au bon déroulement et à l'aboutissement de notre travail.

L.Djamel

Dédicaces

Je dédie ce modeste travail

À Mes très chers parents qui m'ont toujours apportés leur soutien inconditionnel tout au long de mes études, ainsi qu'à mes frères et mes sœurs.

À ma femme et mes filles INES et SYRINE.

À Mes collègues de travail.

Et enfin à toutes les personnes ayant contribué de près ou de loin au bon déroulement et à l'aboutissement de notre travail.

T. Ferhat

« *SOMMAIRE* »

Introduction Générale.....1

Chapitre I : Introduction aux technologies réseaux et web

1-Les réseaux informatiques :

I.1 Introduction2

1.2. Définition2

1.3. Classifications des réseaux2

1.3.1. Réseaux local (LAN)2

1.3.2. Réseau métropolitain2

1.3.3. Réseaux étendus (WAN)2

1.4. Caractéristiques d'un réseau.3

1.5. Le modèle OSI3

1.6. Le modèle TCP/IP.....4

1.7. Sécurité et administration6

1.8. L'architecture Client-serveur.....7

1.8.1. Définition7

1.8.2. Présentation de l'architecture d'un système client/serveur7

1.8.3. Notions de bases7

1.8.4. Fonctionnement d'un système Client/serveur7

1.8.5. Objectifs de cette architecture8

1.8.6. L'évolution de l'architecture Client/serveur.....8

2-Technologies web

2.1. Le Web.....9

2.2. Historique.....9

2.3. Facteurs de succès du Web9

2.4. Principaux termes du Web9

2.5. Types de sites Web11

2.6. Les sites Web statiques12

2.7. Sites Web dynamiques.....12

2.8. Conclusion12

Chapitre II : Analyse et Conception

III.1. Introduction	
II.1 Présentation de l'organisme d'accueil	
II.1.1 Historique	13
II.1.2. Moyens matériels et humains.....	14
II.1.3. Mode d'organisation.....	14
II.1.4. Organisation et mission d'un département.....	16
II.1.5. Problématique.....	16
II.1.6. Objectifs.....	16
II.2 Analyse et conception	
II.2.1. Introduction	17
II.2.2. Phase d'Analyse	17
II.2.2.1. Définition des besoins	17
II.2.2.2. Identification des acteurs	18
II.2.3. Identification des espaces	18
II.2.4. Diagramme de contexte	19
II.3. Phase de Conception	19
II.3.1. Niveau Applicatif	20
a. Diagramme des cas d'utilisations.....	20
-Identification des tâches	21
a.1. Diagramme du cas d'utilisation global	22
a.2. Diagramme du cas d'utilisation « Espace Administrateur »	23
a.3. Diagramme du cas d'utilisation « Espace Chef de Département»	23
a.4. Diagramme du cas d'utilisation « Espace agent de la scolarité»	24
a.5. Diagramme du cas d'utilisation « Espace Enseignant»	24
a.5. Diagramme du cas d'utilisation « Espace Etudiant»	25
b. Diagrammes de séquences	25
b.1. Diagramme de séquence du cas d'utilisation «s'authentifier»	26
b.2. Diagramme de séquence du cas d'utilisation «Création de compte»	26
b.3. Diagramme de séquence du cas d'utilisation «Modifier mot de passe»	27
b.4. Diagramme de séquence du cas d'utilisation «Ajouter examen»	27

b.5. Diagramme de séquence du cas d'utilisation «éditer Pv modulaires»	28
b.6. Diagramme de séquence du cas d'utilisation «Afficher PV modulaires	28
b.7. Diagramme de séquence du cas d'utilisation «Consulter ses notes»	29
c. Diagramme d'activités.....	30
c.1. Diagramme d'activités du cas d'utilisation «Page principale»	30
c.2. Diagramme d'activités du cas d'utilisation «authentification utilisateur»	31
c.3. Diagramme d'activités du cas d'utilisation «Ajouter étudiant»	32
c.4. Diagramme d'activités du cas d'utilisation «éditer pv modulaire»	33
c.5. Diagramme d'activités du cas d'utilisation «Consulter les notes»	34
c.6. Diagramme d'activités du cas d'utilisation «création compte»	35
d. Diagramme de classes.....	35
II.3.2. Niveau de données	37
II.3.2. 1. Définition d'une base de données	37
II.3.2. 2. Définition d'un SGBD	37
II.3.2. 3. Modèle relationnel	37
II.4. Conclusion	43

Chapitre III : Réalisation

III.1. Introduction.....	44
III.2. Environnement de développement.....	44
III.3. Outils utilisés.....	44
III.3.1. Les logiciels utilisés.....	44
a. Windows 7.....	44
b. Visual Studio Code version.....	44
c. Le serveur web Apache.....	45
d. Le serveur de Bases de Données MySQL.....	46
e. PhpMyAdmin :	47
f. WAMP Server :	48
g. Les langages de programmation	49
g.1. HTML	49

Sommaire

g.2. CSS	49
g.3. PHP	49
g.4. JavaScript	50
g.5. SQL	50
III.4. Présentation de quelques interfaces de l'application	51
III.4.1. Interface principale.....	51
III.4.1. Interface modifier mot de passe	51
III.4.1. Interface Ajout d'un étudiant	52
III.4.1. Interface affichage des étudiants	52
III.4.1. Interface saisie des enseignements.....	53
III.4.1. Interface saisie de la fiche de vœux d'un enseignant.....	53
III.4.1. Interface Affichage des notes.....	54
III.6. Conclusion.....	54
Conclusion générale.....	55
Annexe A.....	56
Annexe B.....	67

« *TABLES DES*
FIGURES »

Figure I.1 : Diagramme du modèle OSI.

Figure I.2 : Comparaison des modèles OSI et TCP/IP.

Figure I.3 : Le mode Client/serveur

Figure I.4: Page Web.

Figure I.5: Langage HTML

Figure II.1 : Diagramme de contexte.

Figure II.2 : Démarche de modélisation.

Figure II.3 : Diagramme du cas d'utilisation global.

Figure II.4 : Diagramme du cas d'utilisation 'Espace Administrateur'.

Figure II.5 : Diagramme du cas d'utilisation 'Espace Chef de département'.

Figure II.6 : Diagramme du cas d'utilisation 'Espace Agent de la scolarité'.

Figure II.7 : Diagramme du cas d'utilisation 'Espace Enseignant'.

Figure II.8 : Diagramme du cas d'utilisation 'Espace Etudiant'.

Figure II.9: Diagramme de séquence cas d'utilisation « s'authentifier»

Figure II.10 : Diagramme de séquence cas d'utilisation « création de compte »

Figure II.11 : Diagramme de séquence cas d'utilisation « Modifier mot de passe »

Figure II.12 : Diagramme de séquence cas d'utilisation « Ajout examen »

Figure II.13 : Diagramme de séquence cas d'utilisation «Editer PV modulaire »

Figure II.14: Diagramme de séquence cas d'utilisation « Afficher PV modulaire »

Figure II.15: Diagramme de séquence cas d'utilisation « Consulter les notes»

Figure II.16 : Diagramme d'activité cas d'utilisation « Page principale »

FigureII.17: Diagramme d'activité cas d'utilisation « Authentification utilisateur»

Figure II.18 : Diagramme d'activité cas d'utilisation « Ajouter étudiant»

Figure II.19 : Diagramme d'activité cas d'utilisation « Editer PV modulaire»

Figure II.20 : Diagramme d'activité cas d'utilisation « Consulter les notes»

Figure II.21 : Diagramme d'activité cas d'utilisation « Création du compte»

Figure II.22 : Diagramme de classe.

Figure III.1 : Interface Visual Studio Code .

Figure III.2 : Accueil PhpMyAdmin.

Figure III.3 : Accueil WAMP Server.

Figure III.4 : Page Principale.

Figure III.5 : Page Modifier mot de passe.

Figure III.6 : Page Ajout d'un étudiant.

Figure III.7 : Page Affichage des étudiants.

Figure III.8 : Page saisie des enseignements.

Figure III.9 : Page saisie de la fiche de vœux d'un enseignant.

Figure III.10 : Page Affichage des notes.

CHAPITRE 1
INTRODUCTION AUX
TECHNOLOGIES RESEAUX ET WEB.

Introduction Générale

La gestion d'un service de scolarité d'une université de grande envergure tel que le pôle universitaire de Tamda, qui est en pleine extension notamment sur le nombre d'inscriptions qui augmente de façon exponentielle chaque année et le nombre de nouveaux départements en construction, demande de la rigueur face au mouvement souvent considérable du personnel et la gestion du suivi des étudiants.

Vu le nombre important d'anomalies et de difficultés que rencontrent le personnel de ce site au quotidien pour mener à bien le suivi et la gestion du cursus de ses étudiants, induit par le grand nombre de documents qui circulent quotidiennement en interne dans l'université, cette dernière doit se doter d'une solution lui permettant de gérer de façon rigoureuse et optimale son service de scolarité. Les différents services de scolarités de l'université se trouvent en quelque sorte dépassés par le nombre colossal de données à traiter ainsi que les différentes prestations fournies aux étudiants de l'université. Pour faire face à cette situation complexe et ardue, les départements des sciences et technologies voudraient s'équiper d'une stratégie efficace en la matière, en mettant en œuvre un système informatique qui procurera un environnement convivial au personnel et aux étudiants afin de répondre aux exigences du personnel et de l'ensemble des étudiants (contraintes de fiabilité, efficacité et disponibilité).

L'objectif de notre travail au sein du département des Sciences et Technologies de Tamda de l'UMMTO consiste en l'informatisation de la gestion de la scolarité ainsi que le suivi du cursus des étudiants. Pour la réalisation de ce système informatique, nous avons opté pour une solution web en créant une application qui prendra en charge les enseignements et les étudiants au sein du département des sciences et technologies. Cette application permettra aux différents acteurs intervenants dans le processus de gestion de la scolarité, le bon déroulement des traitements dont la plupart sont encore fait à la main. Cette solution permettra un meilleur rendement et un certain confort de travail et surtout un énorme gain de temps. Notre projet s'articule sur 3 chapitres comme suit :

Le premier chapitre est consacré aux nouvelles technologies et quelques notions de base concernant l'architecture logicielle, Internet, le web et le modèle Client/serveur.

Lors du deuxième chapitre, nous commencerons par une présentation de l'université de Tamda qui est notre organisme d'accueil et plus précisément le département des sciences et technologies qui sera notre champ d'étude. Puis nous passerons à l'analyse et la conception de notre système à travers son architecture globale puis détaillée en définissant les différentes fonctionnalités attendues du système. Pour cela, nous avons choisie le langage UML qui nous permettra de mieux représenter et de modéliser les besoins des utilisateurs.

Le troisième chapitre et le dernier quant à lui, est consacré à la réalisation de la solution définie précédemment. Pour cela, nous allons commencer par décrire les outils et l'environnement de développement utilisés, les différentes interfaces et une brève explication pour chacune d'elle pour donner une vue générale et complète sur le système réalisé.

Nous terminerons ce mémoire par une conclusion générale et des perspectives.

1-Technologies réseaux

Les réseaux informatiques :

1.1 Introduction :

L'architecture des applications modernes d'entreprise est le résultat d'un processus de sélection naturelle rapide où les critères de compétitivité s'appellent modularité, maintenance, portabilité, efficacité, évolutivité... Sous la pression de ces critères et du continuel renouvellement des technologies, l'architecture des applications a progressivement évolué du système Mainframe à l'application Web en passant par le client / serveur et les systèmes distribués. Et Le terme générique « réseau » définit un ensemble d'entités (objets, personnes, etc.) interconnectées les unes avec les autres.

Un réseau permet ainsi de faire circuler des éléments matériels ou immatériels entre chacune de ces entités selon des règles bien définies.

1.2. Définition :

Un réseau est un ensemble de matériels et de logiciels permettant de connecter deux ou plusieurs ordinateurs géographiquement dispersés afin d'échanger des données. Les données circulent dans le réseau sous forme de flux de bits, ces bits sont transférés sous forme de paquets de données qui se déplacent rapidement et qui traversent plusieurs réseaux pour arriver à la destination. Quelque soit la durée de déplacement, l'information doit circuler d'une manière fiable et atteindre sans erreurs la destination voulue.

1.3. Classifications des réseaux [1]

La principale classification des réseaux est faite selon leur taille en : réseau local (LAN), Réseaux métropolitain (MAN) et réseaux étendu (WAN).

1.3.1. Réseaux local (LAN : Local Area Network)

Il est formé d'un ensemble de stations situées dans une même zone géographique limitée (par exemple : un ensemble de bureau, un immeuble).

1.3.2. Réseau métropolitain (MAN : Métropolitain Area Network)

C'est un réseau situé dans une zone géographique plus petite que les réseaux étendus. Les réseaux métropolitains permettent des communications entre différents grands comptes ou à partir d'ordinateurs personnels.

1.3.3. Réseaux étendus (WAN: Wide Area Network)

Relie plusieurs réseaux locaux entre eux. Les réseaux étendus couvrent une zone géographique importante et les liaisons entre les réseaux locaux se font à travers des lignes de communication internationales et des routeurs. Ces réseaux étendus relient des machines hétérogènes qui utilisent des systèmes d'exploitation différents.

1.4. Caractéristiques d'un réseau :

Les réseaux locaux sont des infrastructures complexes et pas seulement des câbles entre stations de travail. Si l'on énumère la liste des composants d'un réseau local, on sera surpris d'en trouver une quantité plus grande que prévue :

_ Le câblage constitue l'infrastructure physique, avec le choix entre paires téléphoniques, câble coaxial ou fibre optique. Il détermine le type de concentrateurs (Switch, HUB, point d'accès Wifi,...etc.) utilisés. Ces équipements constituent les nœuds dans le cas de réseaux en étoile.

_ La méthode d'accès décrit la façon dont le réseau arbitre les communications des différentes stations sur le câble : ordre, temps de parole, organisation des messages. Elle dépend étroitement de la topologie et donc de l'organisation spatiale des stations les unes par rapport aux autres. La méthode d'accès est essentiellement matérialisée dans les cartes d'interfaces, qui connectent les stations au câble.

_ Les protocoles de réseaux sont des logiciels qui « tournent » à la fois sur les différentes stations et leurs cartes d'interfaces réseaux. C'est le langage de communication. Pour que deux structures soient connectées sur le réseau, ils doivent « parler » le même protocole.

_ Le Système d'exploitation du serveur réseau, souvent nommé gestionnaire du réseau, est installé sur le ou les serveurs. Il gère les partages, droits d'accès, ...etc. Pour Microsoft on trouve Windows NT serveur, Windows 2000 serveur, Windows 2003, 2008, 2008 R2, 2012, 2012 R2, 2016, 2019. Ce sont des versions spécifiques. Linux est utilisé sous différentes versions serveurs. Novell Netware est un système dédié principalement efficace comme serveur de fichier.

_ Le système de sauvegarde est un élément indispensable qui fonctionne de diverses manières soit en recopiant systématiquement tous les fichiers du ou des serveurs, soit en faisant des sauvegardes régulières, éventuellement automatisées.

_ Un pont, Un routeur ou passerelle constituent les moyens de communication qui permettent à un de ses utilisateurs de « sortir » du réseau local pour atteindre d'autres réseaux locaux ou des serveurs distants, Internet ou autres.

_ Le système de gestion et d'administration du réseau envoie les alarmes en cas d'incidents, comptabilise le trafic, mémorise l'activité de réseau et aide le superviseur à prévoir l'évolution de son réseau. Cette partie est typiquement software.

1.5. Le modèle OSI [2]:

Le modèle OSI (Open System Interconnexion Model) défini en 1977 régit la communication entre 2 systèmes informatiques selon 7 niveaux. A chaque niveau, Les deux systèmes doivent communiquer « compatibles ». En matériel réseau, nous n'utilisons que les couches inférieures, jusqu'à niveau 3. Ces niveaux sont également appelés couches. L'OSI est un modèle de base normalisé par l'International Standard Organisation (ISO).

Ce tableau montre les différentes couches du modèle OSI :

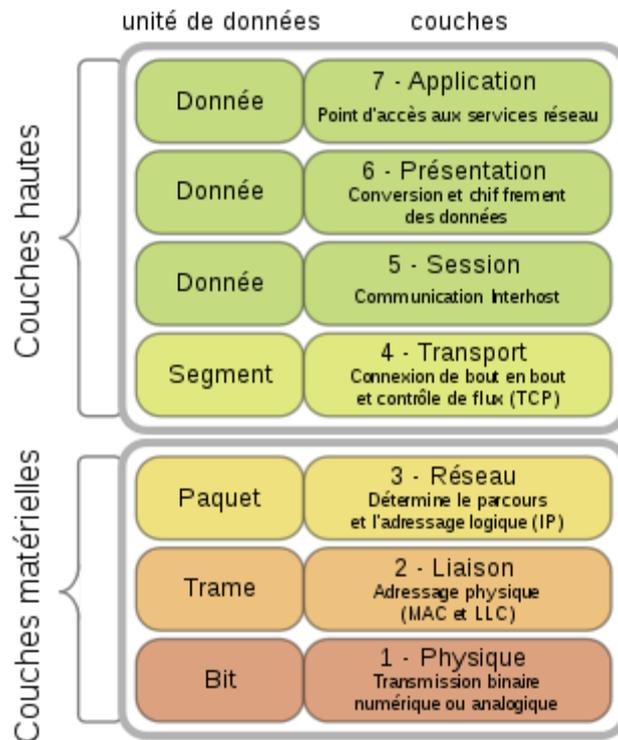


Figure I.1 : Diagramme du modèle OSI.

- **Niveau 7 (application)** : gère le format des données entre logiciels.
- **Niveau 6 (Présentation)** : met les données en forme, éventuellement de l'encryptage de la compression, par exemple mise en forme des textes, images et vidéo.
- **Niveau 5 (session)** : gère l'établissement, la gestion et coordination des communications.
- **Niveau 4 (Transport)** : s'occupe de la gestion des erreurs, notamment avec le protocole TCP.
- **Niveau 3 (réseau)** : sélectionne les routes de transport (routage) et s'occupe du traitement et du transfert des messages : gère par exemple les protocoles IP (adresse et le masque de sous-réseau) et ICMP. Utilisé par les routeurs et les switches manageables.
- **Niveau 2 (liaison de données)** : Utilise les adresses MAC. Le message Ethernet à ce stade est la trame, il est constitué d'un en-tête et des informations. L'en-tête reprend l'adresse MAC de départ, celle d'arrivée et une indication du protocole supérieur.
- **Niveau 1 (Physique)** : gère les connections matérielles et la transmission, définit la façon dont les données sont converties en signaux numériques : ca peut-être un câble coaxial, paires sur RJ45, onde radio, fibre optique,...etc.

1.6. Le modèle TCP/IP:

Le modèle TCP/IP s'inspire du modèle OSI auquel il reprend l'approche modulaire mais réduit le nombre à quatre. Les trois couches supérieures du modèle OSI sont souvent utilisées par une même application. Ce n'est pas le cas du modèle TCP/IP. C'est actuellement le modèle théorique le plus utilisé.

Le tableau suivant montre les deux modèles OSI et TCP/IP :

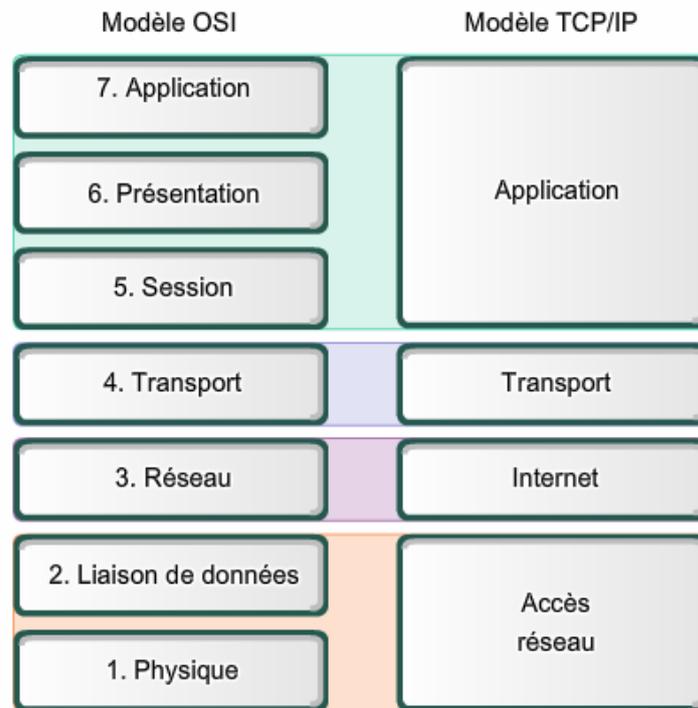


Figure I.2 : Comparaison des modèles OSI et TCP/IP.

De nouveau, on ajoute à chaque niveau un entête, les dénominations des paquets de données changent à chaque fois :

- _ Le paquet de données est appelé message au niveau de la couche application
- _ Le message est ensuite encapsulé sous forme de segment dans la couche transport. Le message est découpé en morceau avant envoi pour respecter une taille maximum suivant le MTU.
- _ Le segment une fois encapsulé dans la couche Internet prend le nom de datagramme
- _ Enfin, on parle de trame envoyée sur le réseau au niveau de la couche accès réseau.

Les couches du modèle TCP/IP sont plus générales que celles du modèle OSI.

A. Couche application :

Elle reprend les applications standards en réseau informatique et internet :

- _ SMTP : « Simple Mail Transport Protocol » gère le transfert de mails entre serveurs
- _ POP : gère le transfert des mails entre un serveur de messagerie et un ordinateur client
- _ TELNET : connexion sur une machine distante (serveur) en tant qu'utilisateur
- _ FTP (File Transfert Protocol), transfert des fichiers via Internet et beaucoup d'autres.

B. Couche transport:

La Couche transport permet le transfert des données et les contrôles qui permettent de vérifier l'état de la transmission.

Les protocoles des couches suivantes permettent d'envoyer des données issues de la couche application. On ne définit pas réellement les logiciels qui communiquent, mais des numéros de ports associés au type d'application (numéro variant de 0 à 65535,216). Par exemple, la navigation Internet utilise le port TCP 80, l'http, le 443, le FTP utilise le 21, ...etc.

La couche transport gère 2 protocoles de transport des informations, indépendamment du type de réseau utilisé :

_ TCP est orienté connexion (il vérifie la bonne transmission de données par des signaux d'accusées de réception – a knowledge – du destinataire), il assure ainsi le contrôle des données.

_ UDP, archaïque et non orienté connexion, n'assure aucun contrôle de transmission des données, par exemple utilisé en streaming.

Ces 2 types (orienté connexion ou pas) sont une notion utilisée pour les firewalls. Si vous fermez un port en TCP, l'envoi d'un message ne renvoie pas de signal de retour (a knowledge), faisant croire que l'adresse IP est libre, non utilisée. En UDP au contraire, un port fermé ne renvoie pas d'informations, faisant croire à une adresse IP utilisée. Le protocole UDP renvoie uniquement un message si le port est en erreur (ne répond pas).

C. Couche Internet:

La couche Internet est chargée de fournir le paquet des données. Elle définit les datagrammes et gère la décomposition/recomposition des segments.

La couche Internet utilise 5 protocoles, seuls les 3 premiers sont importants :

1. Le protocole IP : gère les destinations des messages, adresse du destinataire.
2. Le protocole ARP (Adresse Resolution Protocole) : gère les adresses des cartes réseaux et la correspondance avec l'adresse IP. Chaque carte a sa propre MAC d'identification codée sur 48 bits.
3. Le protocole ICMP (Internet Control Message Protocol) gère les informations relatives aux erreurs de transmission. ICMP ne les corrige pas, il signale uniquement que le message contient des erreurs, utilisé par exemple par la commande DOS Ping.
4. Le protocole RARP (Reverse Address Resolution Protocol) gère l'adresse IP pour les équipements réseaux qui ne peuvent en récupérer une automatiquement par lecture d'information dans un fichier de configuration ou via un serveur DHCP. Lorsqu'un équipement réseau démarre, le gestionnaire réseau lit l'adresse IP à utiliser, ce qui est impossible pour certains équipements qui ne possèdent pas de disque dur (principalement les terminaux).
5. Le protocole IGMP (Internet Group Management Protocol) permet d'envoyer le même message à des ordinateurs qui font partie d'un groupe. Il permet aussi à ces machines de s'abonner et se désabonner d'un groupe. La principale application HARDWARE de l'IGMP se trouve dans les SWITCH managés. Ce protocole permet de regrouper des stations.

D. Couche Accès réseau :

La couche accès réseau spécifie la forme sous laquelle les données doivent être transmises. Elle prend en charge les notions suivantes :

- Type de réseau (Ethernet, Token Ring, ...), y compris les cartes réseau
- Transfert des données
- Synchronisation de la transmission de données
- Mise en forme (format) des données
- Conversion analogique/numérique pour les modems téléphoniques
- Contrôle des erreurs.

1.7. Sécurité et administration :

Un des aspects importants d'un réseau informatique local est la centralisation de l'administration des données. Ceci permet de sauvegarder et sécuriser les données sur une seule machine. La sécurité reprend un ensemble de mesures contre les intrusions et virus, la gestion des privilèges et droits d'accès, la sauvegarde quotidienne des données, des équipements redondants en cas de panne, ...

Il n'y a pas de solutions idéales pour la sécurité des réseaux (et pour la sécurité informatique en générale). Trois solutions sont envisageables : les solutions matérielles que nous verrons, des solutions

basées sur Linux et des solutions basées sur Windows ou des programmes rajoutés sur ces stations sont d'ailleurs complémentaires. Sur un gros réseau « sensible », mettre un VPN hardware n'est pas suffisant. Une sécurité logicielle complémentaire incluant des contrôles d'accès au niveau administration serveur (serveur, dossier, droits d'accès) et logiciels de sécurités vérifiant le trafic sur le réseau interne n'est pas superflu.

_ Les routeurs peuvent être remplacés par une solution basée sur un serveur 2003 ou 2008, par un logiciel proxy comme Wingate ou par un ordinateur configuré spécifiquement en Linux.

_ Un serveur proxy est parfois intégré dans les routeurs (mais généralement sous Windows ou Linux)

_ Les firwalls sont intégrés dans certains routeurs mais des logiciels assurent (presque) des fonctions équivalentes, souvent intégrés dans l'anti-virus (ex : Symantec, ZoneAlarm, McAfee au niveau des stations)

_ Les réseaux privés intégrés (VPN) sont intégrés dans certains systèmes d'exploitation serveurs mais peuvent également être des équipements spécifiques.

_ Les anti-virus sont le plus souvent des logiciels, mais peuvent être implémentés dans des routeurs qui vérifient tout le trafic extérieur.

1.8. L'architecture Client-serveur.

1.8.1. Définition [3]

C'est un modèle informatique basé sur le traitement distribué selon le quel un utilisateur lance un logiciel client à partir d'un ordinateur relié à un réseau, déclenchant simultanément le lancement d'un logiciel serveur situé dans un autre ordinateur possédant les ressources souhaitées par l'utilisateur.

1.8.2. Présentation de l'architecture d'un système client/serveur

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes (des machines faisant partie du réseau) contactent un serveur, une machine généralement très puissante en termes de capacités d'entrée-sortie, qui leur fournit des services. Les services sont exploités par des programmes, appelés programmes clients, s'exécutant sur les machines clientes. On parle ainsi de client FTP, client de messagerie, lorsque l'on désigne un programme, tournant sur une machine cliente, capable de traiter des informations qu'il récupère auprès du serveur. (Dans le cas du client FTP il s'agit de fichiers, tandis que pour le client messagerie il s'agit de courrier électronique). Dans un environnement purement Client/serveur, les ordinateurs du réseau (les clients) ne peuvent voir que le serveur, c'est un des principaux atouts de ce modèle (chaque machine est soit client, soit serveur).

1.8.3. Notions de bases

- **Client** : C'est le 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** : C'est un processus accomplissant une opération sur demande d'un client.
- **Requête** : C'est un message transmis par un client à un serveur décrivant l'opération à exécuter pour le compte d'un client.
- **Réponse** : C'est un message transmis par un serveur à un client suite à l'exécution d'une opération contenant les paramètres de retour de l'opération.
- **Middleware** : C'est le logiciel qui, au milieu assure les dialogues entre les clients et les serveurs souvent hétérogènes.

1.8.4. Fonctionnement d'un système Client/serveur

Un système client/serveur fonctionne selon le schéma suivant :

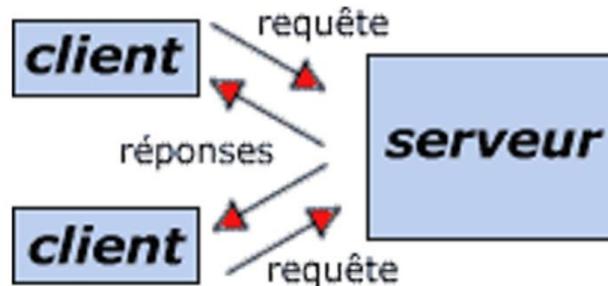


Figure I.3 : Le mode Client/Serveur

Le client émet une requête vers le serveur grâce à son adresse et le port, qui désigne un service particulier du serveur. Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine client et son port.

Remarque : *l'architecture à deux niveaux est donc une architecture client/serveur dans laquelle le serveur est polyvalent, c.-à-d. qu'il est capable de fournir directement l'ensemble des ressources demandées par le client. Dans l'architecture à trois niveaux par contre les applications au niveau serveur sont délocalisées, c.-à-d. Que chaque serveur est spécialisé dans une tâche (serveur Web/serveur de base de données par exemple). Ceci nous conduit à définir l'architecture multi-niveaux, qui est une architecture à N niveaux (tout comme l'architecture à 3-tiers mais avec plusieurs serveurs intermédiaires).*

1.8.5. Objectifs de cette architecture

- Une plus grande flexibilité/souplesse.
- Une plus grande sécurité (la sécurité peut être définie pour chaque service).
- De meilleures performances (les tâches sont partagées).

1.8.6. L'évolution de l'architecture Client/Serveur [4]

En espace de quelques années, les progrès réalisés dans les technologies de l'information nous ont ait entré dans une nouvelle ère, celle de l'informatique ouverte au grand public, et du client/serveur.

Le client/serveur apparaît aujourd'hui comme le point de convergence des besoins qu'expriment les entreprises depuis des années, celui d'intégrer les nouvelles technologies sans remettre en cause les systèmes existants, de fournir aux utilisateurs rapidité, souplesse et confort dans l'utilisation quotidienne de l'outil informatique et plus généralement de disposer d'un système informatique adapté aux fortes exigences qu'impose le contexte économique actuel.

Le modèle client/serveur constitue une évolution majeure de l'informatique. Le principe de base est de décomposer un processus informatique en au moins deux tâches moins complexes (le client et le serveur) associés à un mécanisme de communication leur permettant de coopérer.

2-Technologies web

2.1 Le Web [5]

World Wide Web (toile d'araignée mondiale) est un système d'information hypermédia sur Internet. Il est fondé sur l'extension de la technique de l'hypertexte aux réseaux, et a créé le premier réseau hypermédia réparti de couverture mondiale, donnant ainsi la possibilité d'accès à de très grandes quantités et variétés de documents de façon très simple. Le WEB repose sur une architecture client-serveur distribuée et une interface de navigation offrant la puissance d'hypertexte en environnement graphique.

2.2. Historique

Le WWW est inventé en 1989 et développé en 1992, au CERN (Centre Européen de Recherche Nucléaire) à Genève, par Tim Berners Lee, un chercheur atomiste en quête d'un outil de partage de connaissances scientifiques avec les autres chercheurs de toute l'Europe. Il était le premier à proposer de relier les documents sous forme d'une toile d'araignée c.-à-d. des documents reliés avec des liens hypertextes. En 1993, une première interface graphique, Mosaïc, est livrée au NCSA (National Center For Supercomputing Applications).

2.3. Facteurs de succès du Web : Ces facteurs sont principalement :

- *Universalité* : Ceci grâce à :
 - _ La navigation hypertexte étendue : les liens peuvent s'appliquer à des documents hébergés sur d'autres sites au sens géographique du terme.
 - _ Support multimédia : images fixes, animées, sons, vidéos, etc.
 - _ Multiplateforme : les codages utilisés permettent d'utiliser toutes les machines existantes.
- *Simplicité* : est assurée par :
 - _ L'unité de visualisation du contenu (navigateur)
 - _ Les liens hypertextes
 - _ Les systèmes d'adresses URL.
 - _ Intégration de tous les services qui existent (messagerie, FTP, news, etc.)
- *Gratuité des protocoles* : Sont offerts au domaine public comme –http- .

2.4. Principaux termes du Web

- 1- **Page Web** : C'est une série de mots disposés linéairement et d'objets graphiques, sonores ou vidéo. Certains mots ou graphiques (souvent en surbrillance ou soulignés) pointent vers d'autres pages, c'est ce qu'on appelle hyperlien qui peuvent être obtenues en cliquant sur ces mots ou graphiques. Cette technique d'hypertexte permet la lecture non linéaire d'un document (navigation, surf).

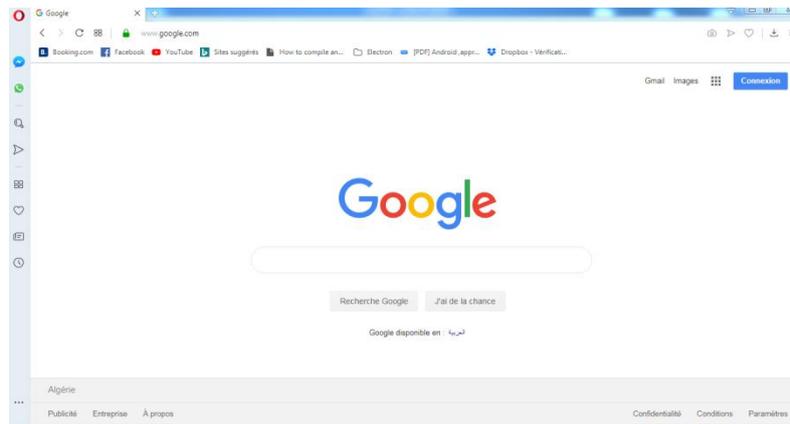


Figure I.4: Page Web.

- 2- **Site Web** : C'est un ensemble de pages Web reliées les unes aux autres par des hyperliens (liens hypertextes).
- 3- **Page d'accueil** : C'est la page d'introduction d'un site Web qu'on peut aussi appeler page de couverture. Elle est très importante car elle détermine la première impression de l'internaute.
- 4- **Site Internet** : C'est un ensemble de données (texte, image, son, vidéo, etc.) Hébergées sur un serveur ayant une adresse IP et accessible aux internautes. Il existe différents types de sites selon leurs contenus et la nature des services qu'ils offrent. Citons les sites FTP, WEB, etc.
- 5- **Serveur Internet** : C'est l'ensemble des ressources matérielles (espace disque, connexion Internet, contenu du site, etc.) et logicielles (les programmes) qui servent les clients d'où émanent les demandes de services.
- 6- **URL (Uniform Resource Locator)** : C'est une adresse universelle d'une page WEB. Elle a le format suivant : Protocole utilisé : // nom du serveur/ chemin d'accès au fichier, Une URL s'apparente à un nom de serveur et une information sur le type de protocole de réseau à utiliser pour atteindre la ressource.
- 7- **HTML (HyperText Markup Language)** : Les pages Web (ayant l'extension html) sont écrites en HTML. C'est un langage de balisage hypertextuel permettant d'inclure outre du texte, des graphiques, sons, vidéos et des pointeurs vers d'autres pages du même site ou d'un site distant.

```

1 <html>
2 <head>
3 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="s
4 <meta charset="UTF-8"></head>
5 <link rel="stylesheet" href="style.css">
6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7
8 <body>
9
10 <!-- bootstrap javascript -->
11 <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>
12 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js"></script>
13 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
14
15 <nav class="navbar navbar-expand-md bg-info navbar-dark">
16 <!-- Brand -->
17 <a class="navbar-brand" href="entete.php">Accueil</a>
18
19 <!-- Links -->
20 <ul class="navbar-nav">
21 <!-- Dropdown -->
22 <li class="nav-item dropdown">
23 <a class="nav-link dropdown-toggle" href="#" id="navbardrop" data-toggle="dropdown">
24 Domaine
25 </a>
26 <div class="dropdown-menu">
27 <a class="dropdown-item" href="afficher_dom.php">Afficher liste</a>

```

Figure I.5: Langage HTML

- 8- **Hypertexte** : C'est une forme non séquentielle non narrative d'organisation et d'accès à l'information représenté par un réseau de liens associatifs qui forment une structure d'hypertexte
- 9- **Hypermédia** : Le terme hypermédia désigne un hypertexte dont on insiste sur la nature multimédia de son contenu informationnel.
- 10- **Hyperdocument** : il désigne un ensemble de documents multimédia ayant une organisation et un fonctionnement hypertextuel.
- 11- **Navigateur** : C'est un logiciel qui offre une interface d'accès aux différentes informations disponibles sur le Web et l'exploration de ce dernier, Ce dernier interprète le texte et les commandes de formatage qu'il contient et affiche à l'écran la page correctement formatée.
- 12- **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 accès à l'information ainsi trouvée.
- 13- **Lien hypertexte** : C'est un pointeur qui pointe sur un nœud de destination, souvent il est sous forme d'une icône, symbole, petite image, un mot dans un texte, etc. Il permet d'explorer le Web avec un simple clic de souris. On distingue :
 - Les liens internes qui renvoient à d'autres pages du même site
 - Les liens externes qui renvoient à d'autres pages d'un autre site
 - Les liens vers un signet qui renvoient à un signet de la même page ou d'une autre page
 - Les liens vers un e-mail qui permettent à l'internaute d'envoyer un e-mail directement depuis la même page.
 - Liens de téléchargement qui permettent de télécharger des documents depuis la même page.

2.5. Types de sites Web [4]

Les sites Web sont difficiles à classer, les limites entre les différents types varient en fonction de l'objectif et du public visé. En général on retrouve la classification ci-après

1. Sites personnels (home page) : Se caractérisent par leur manque d'uniformité, leur présentation est liée aux capacités et aux possibilités de leurs créateurs, qui peuvent être très variés. Evidemment, il est difficile d'obtenir un tout harmonieux à partir de motifs provenant de styles variés. En ce qui concerne leur contenu, tous les thèmes sont traités.
2. Sites d'informations : Ces sites sont axés sur le texte offrant souvent de nombreux liens et peuvent se composer de plusieurs milliers de pages qui traitent des thèmes différents.
3. Sites pour présentation d'entreprises : Ils se composent généralement de quelques pages qui servent à présenter l'entreprise. La présentation est souvent rigide et le sérieux est de rigueur.
4. Sites "boutiques" : Ce sont des boutiques qui commercialisent leurs produits en ligne en offrant des informations sur chaque produit.
5. Sites de bases de données : Ces bases de données présentent leur contenu seulement sur demande, en remplissant un formulaire de requête.
6. Sites portail : Un portail est une collection de liens qui est conçue autour d'une fonction de recherche et se compose souvent de contenus prêtés

2.6. Les sites Web statiques [6]

Ce sont des pages Web écrites en HTML invariantes, proposées à l'avance, autrement dit, à chaque fois qu'un client affiche une page cette dernière se présente de la même manière tout au moins tant que le code HTML correspondant n'a pas été mis à jour par l'auteur.

Les fichiers de code HTML ne sont pas en mesure d'interagir avec le serveur. Si par exemple, on veut récupérer des informations liées à l'utilisateur pour les stocker dans une base de données, il va falloir mettre en œuvre d'autres outils capables de supporter ce type d'interaction. La mise à jour des pages HTML est très difficile et doit s'effectuer via un éditeur, de nombreux fichiers sont liés et la cohérence des liens est difficile à gérer.

2.7. Sites Web dynamiques

Les lourdeurs de la consultation et la mise à jour des pages Web statiques ainsi que le faible volume d'informations qu'elles contiennent ont rapidement conduit les concepteurs à proposer des sites Web dynamiques. Dans ce cas, les pages ne sont pas fabriquées à l'avance. Cette composante dynamique peut recouvrir des aspects tels que la diffusion d'informations évoluant dans le temps ou la récupération des renseignements saisis par l'utilisateur pour décider de déclencher telle ou telle autre action.

Les auteurs chargés de la conception de telles pages Web doivent faire appel à des ressources de programmation dépassant le cadre du HTML, qui connaît quelques limitations dont la principale est de proposer des pages statiques. Ces sources de programmation peuvent être des langages de scripts, les outils d'interfaçage Web /base de données ou autre (que nous allons aborder plus tard).

Voici une liste forcément incomplète des fonctionnalités dynamiques de base :

- Pages d'accueil personnalisables.
- Accès à des bases de données.
- Transactions commerciales.
- Formulaires de renseignements.
- Services d'inscription.

2.8. Conclusion :

Dans ce premier chapitre, nous avons présenté les notions de base sur les nouvelles technologies de l'information et de la communication (réseaux, internet, web...) nécessaire dans le cadre de ce projet. Les différents concepts traités dans ce chapitre nous aiderons à mieux comprendre notre mode d'opération et les notions fondamentales pour mener à bien notre application.

Le prochain chapitre, sera consacré à la présentation de l'organisme d'accueil, l'analyse et la conception de notre application.

CHAPITRE 2
ANALYSE ET CONCEPTION.

II.1 Présentation de l'organisme d'accueil

II.1.1 Historique :

L'Université Mouloud Mammeri de Tizi-Ouzou (U.M.M.T.O) créée par l'ordonnance N° -17-77 du 20 Juin 1977 en tant que centre universitaire (C.U.T.O) dont le premier noyau fut réalisé dans l'ex-centre de personnes âgées. La première rentrée universitaire de septembre 1977 avait accueilli 490 étudiants encadrés par 27 jeunes enseignants dans une structure de quatre départements. Ceux-ci évoluaient dans une salle de cinéma de la ville et un sous-sol de la mairie improvisés alors en amphithéâtres, 07 salles de travaux dirigés, 06 laboratoires et 02 laboratoires de langue. En 1989, cet important pôle a été élevé au rang d'Université de Tizi-Ouzou (U.T.O.) par le Décret Exécutif N°89-139 du 01/08/89 avec neuf instituts. En 1991, l'université de Tizi-Ouzou enrichit son offre de formation par la création du Département de Langue et Culture Amazighes. Entre temps, l'université de Tizi-Ouzou fut baptisée à juste titre du nom de l'illustre écrivain Mouloud Mammeri, décédé le 26 février 1989. Elle a du fonctionner ainsi jusqu'en 1998 où un décret présidentiel remodèle en profondeur l'organisation de l'université algérienne d'une manière générale, y compris dans la gestion de ses aspects pédagogiques.

Actuellement, avec des effectifs progressant rapidement, l'Université Mouloud Mammeri de Tizi-Ouzou ne cesse d'étoffer son offre de formation par la création de nouvelles filières et spécialités tant en graduation qu'en post-graduation. Même si elle demeure en phase de construction, l'université Mouloud Mammeri affirme chaque année un peu plus son identité et consolide sa tendance à jouer un rôle de pôle scientifique important tant à l'échelle nationale qu'internationale, grâce notamment aux potentialités qu'elle recèle.

Avec cet effectif, l'université de Tizi-Ouzou a continué à s'étendre au sud-est de la ville pour s'implanter dans un premier temps sur le campus de Bastos, pour abriter tout le pôle de Technologie et plusieurs cités universitaires, et dans un second temps sur le nouveau campus de Tamda inauguré en 2010. Ce dernier est le site le plus récent de l'université Mouloud MAMMERI, il se situe dans la commune de Ouaguenoun à 15 kilomètres de l'est de la wilaya Tizi-Ouzou.

Au titre de l'année universitaire 2017/2018, l'Université Mouloud MAMMERI de Tizi-Ouzou (U.M.M.T.O.) comprend neuf (09) Facultés regroupant vingt neuf (35) départements, et assure la formation de 49392 étudiants

II.1.2. Moyens matériels et humains : *[Statistiques 2012, Vice Rectorat D.P.O]*

L'U.M.M.T.O dispose de nombreux moyens matériels et humains, à savoir :

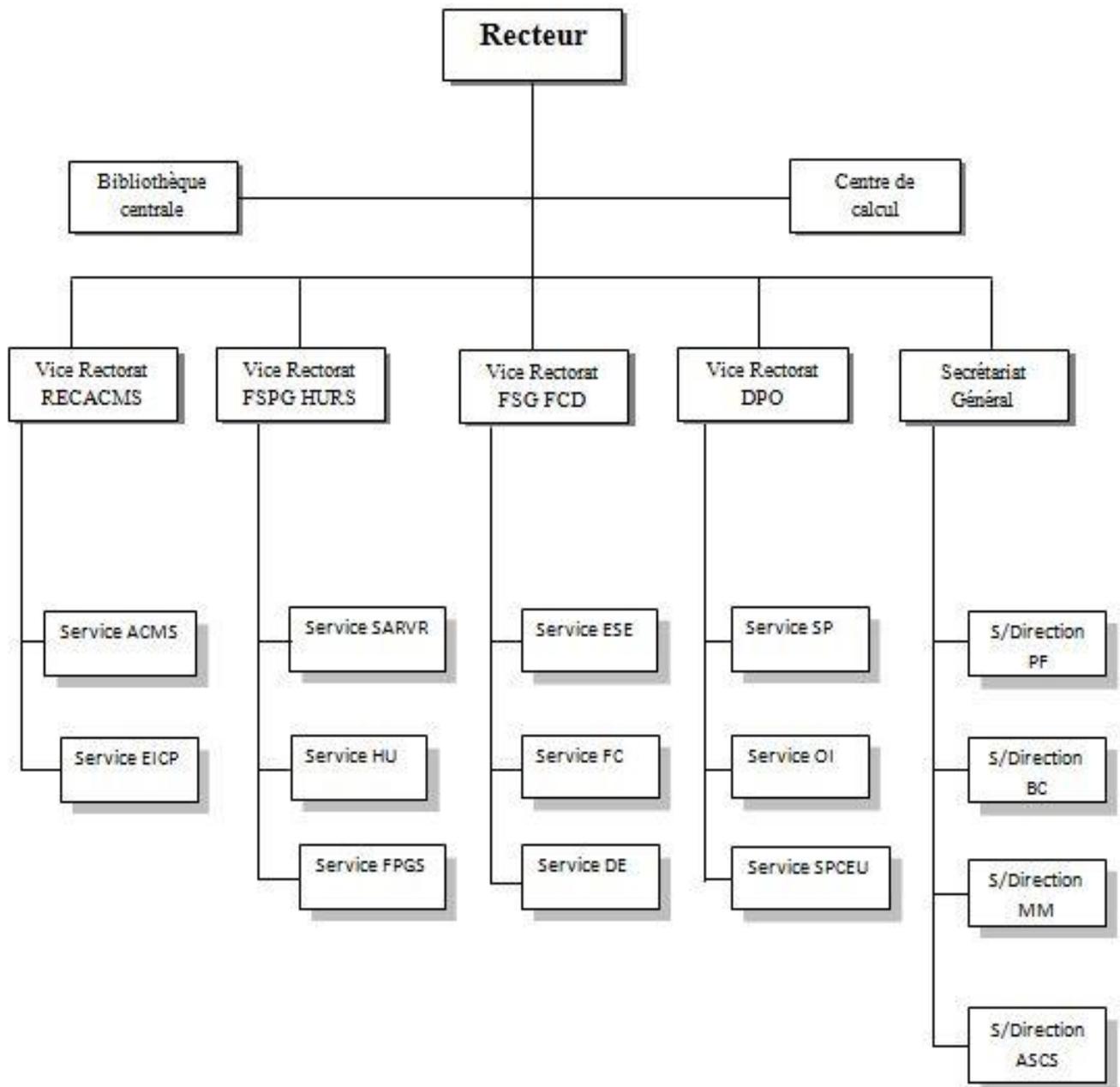
- ✓ 09 facultés répartis sur 35 départements ;
- ✓ 1882 enseignants dont :
 - 97 Professeurs ;
 - 139 Maitres de Conférence classe A ;
 - 120 Maitres de Conférences classe B ;
 - 785 Maitres Assistants classe A ;
 - 508 Maitres Assistants classe B ;
 - 32 Assistants ;
 - 13 Professeurs Hospitalo-Universitaire ;
 - 07 Maitres de Conférence Hospitalo-Universitaire classe A ;
 - 02 Maitres de Conférence Hospitalo-Universitaire classe B ;
 - 167 Maitres Assistants Hospitalo-Universitaires classe A ;
- ✓ 1424 personnels administratifs et techniques ;
- ✓ 49392 étudiants

II.1.3. Mode d'organisation :

La direction de l'UMMTO est structurée comme suit :

- ✓ Rectorat
 - Vice Rectorat des Relations Extérieures, Coopération, Animation, Communication et Manifestations Scientifiques (RE CACMS)
 - Service Animation, Communication et Manifestations Scientifiques (ACMS)
 - Service Echanges Interuniversitaires, Coopération et Partenariat (EICP)
 - Vice Rectorat de Formation Supérieure de Post-Graduation, d'Habilitation Universitaire et de Recherche Scientifique (FSPG HURS)
 - Service de Suivi des Activités de Recherche et de Valorisation de ses Résultats (SARVR)
 - Service Habilitation Universitaire (HU)
 - Service Formation Post-Graduation et Post-Graduation Spécialisée (FPGS)
 - Vice Rectorat de Formation Supérieure de Graduation, de Formation Continue et des Diplômes (FSG FCD)
 - Service Enseignements, Stages et Evaluation (ESE)
 - Service de Formation Continue (FC)
 - Service Diplômes et Equivalences (DE)
 - Vice Rectorat de Développement, Prospective et Orientation (DPO)
 - Service Statistiques et Prospective (SP)
 - Service Orientation et Information (OI)
 - Service de Suivi des Programmes de Construction et Equipement de l'Université (SPCEU)
 - Secrétariat Général
 - Sous Direction de Personnels et Formation (PF)
 - Sous Direction du Budget et Comptabilité (BC)

- Sous Direction Moyens et Maintenance (MM)
 - Sous Direction des Activités Scientifiques, Culturelles et Sportives (ASCS)
 - Bibliothèque Centrale
 - Centre des Systèmes et Réseaux d'Informations de Communication de Téléenseignement et Enseignement à Distance (Centre de Calcul)
- Cette hiérarchie est modélisée par la figure II.1



II.1.4. Organisation et mission d'un département : [Journal Officiel]

Un département est structuré comme suit :

- ✓ Chef de département : Il est assisté par
 - Le chef de département adjoint chargé de la scolarité et des enseignements de graduation : Il est assisté par
 - le chef de service de la scolarité ;
 - le chef de service du suivi des enseignements et de l'évaluation.

Il a pour tâches :

- de suivre les opérations d'inscription et de réinscription des étudiants de graduation ;
- de veiller au bon déroulement des enseignements ;
- de veiller au bon déroulement des examens et épreuves de contrôle des connaissances.
 - Le chef de département adjoint chargé de la post-graduation et de la recherche scientifique Il est assisté par
 - le chef de service de la formation supérieure de post-graduation et de la poste graduation spécialisée ;
 - le chef de service du suivi des activités de recherche.

Il a pour taches :

- de veiller au déroulement des enseignements de post-graduation ;
- de veiller au déroulement des formations de post-graduation spécialisée ;
- d'assurer le suivi des activités de recherche ;
- d'assurer le suivi du fonctionnement du comité scientifique de département.

II.1.5. Problématique :

Au cours de nos différents entretiens avec les responsables du département mathématique et informatique, nous avons constaté quelques problèmes quant à la gestion de la scolarité de ce dernier, tel que :

- ✓ Difficulté de recherche d'informations due à l'utilisation d'une grande masse de documents ;
- ✓ Le traitement manuel, ce qui implique la lenteur du travail et engendre parfois de l'incohérence ;
- ✓ Risque de perte de l'information due aux mauvaises conditions d'archivage car les informations sont archivées sur papier et la manipulation fréquente du papier finit par détériorer les documents;
- ✓ Perte de temps souvent due à l'affichage tardif des PV ;
- ✓ Lenteur et difficulté de mise à jour en cas d'erreur ;
- ✓ Difficulté d'établir les Procès-verbaux de délibérations.

II.1.6. Objectifs :

Dans le but de remédier aux périples de ce système manuel, il fallait penser à un moyen fiable, efficace, sûr et surtout compatible avec les moyens des départements.

Les principales caractéristiques de l'application seront :

- Gestion des utilisateurs ;
- Gestion des sections ;
- Gestion des modules ;
- Gestion des examens ;
- Gestion des étudiants ;
- Gestion des fiches de vœux ;
- Gestion des notes des étudiants ;
- Consultation des notes et des PV de délibération ;

L'application est destinée à l'administration, aux enseignants, ainsi qu'à tous les étudiants.

II.2 Analyse et conception

II.2.1 Introduction :

La phase de conception permet de décrire de manière non ambiguë, le plus souvent en utilisant un langage de modélisation, le fonctionnement futur du système, afin d'en faciliter la réalisation. C'est pourquoi, le langage UML s'impose avec ses multiples diagrammes qui nous permettent un passage progressif de la description du problème (domaine et fonctionnalités souhaitées) à la solution (spécifications du système).

Une méthode d'analyse et de conception a pour objectif de permettre de formaliser les étapes préliminaires du développement d'un système afin de le rendre plus fidèle aux besoins du client. Pour ce faire, on part d'un énoncé informel (le besoin tel qu'il est exprimé par le client, complété par des recherches d'informations auprès des experts du domaine fonctionnel, comme par exemple les futurs utilisateurs d'un logiciel), ainsi que de l'analyse de l'existant éventuel (c'est-à-dire la manière dont les processus à traiter par le système se déroulent actuellement chez le client).

La phase d'analyse de l'existant permet de lister les résultats attendus, en termes

- de fonctionnalités,
- de performance,
- de robustesse,
- de maintenance,
- de sécurité,
- d'extensibilité, ... etc.

II.2.2. Phase d'Analyse :

Dans cette partie, les interactions significatives (Interaction entre le système et les acteurs, interaction entre les objets) vont être spécifiées d'une manière claire et détaillée du point de vue de l'application de gestion de scolarité.

Pour cela, en premier lieu il faudra déterminer d'une manière globale ce qui se trouve dans le champ de l'application.

II.2.2.1. Définition des besoins :

Tous les jours, chaque département de l'UMMTO, en l'occurrence le département des sciences et technologies, assure la prise en charge des enseignements ainsi que les cursus des étudiants, à savoir : l'inscription et l'affectation des étudiants en début de chaque rentrée universitaire et l'édition des PV de délibération à la fin de chaque semestre par l'agent de la scolarité. La saisie de la fiche de vœux semestrielle et la saisie des PV modulaires à la fin des examens par l'enseignant. La consultation des PV de délibérations et l'affectation des enseignants par le chef de département, ainsi que la consultation de la liste des formations et des notes par l'étudiant.

Pour faciliter ces tâches, la réalisation et l'implémentation d'une nouvelle application web pour l'automatisation des procédures de gestion de scolarité est capitale.

II.2.2.2. Identification des acteurs :

Après une analyse de l'existant, on a pu définir les acteurs du nouveau système qui sont :

Administrateur (admin.): Il s'occupe de la gestion des comptes des utilisateurs.

Chef de Département : Il dispose de tous les privilèges (afficher, ajouter, modifier, supprimer, chercher...)

Agent de la scolarité : il s'occupe de la gestion du cursus des étudiants, création des comptes pour les enseignants et les étudiants ainsi que les différentes éditions

Enseignant : Il peut remplir sa fiche de vœux, chercher un étudiant ou un enseignant, ajouter un examen et une évaluation, ainsi que modifier l'évaluation en cas d'erreur d'insertion.

Etudiant: Les étudiants peuvent consulter leurs notes, visualiser la liste des enseignants, chercher un enseignant. Voir Les modules enseignées dans l'université, les enseignements qui existent, les filières enregistrées.

II.2.3. Identification des espaces :

A chaque acteur correspond un espace qui regroupe toutes les actions et les tâches qu'il peut effectuer. Pour notre application, cinq espaces sont identifiés :

- *Espace Administrateur.*
- *Espace Chef de département.*
- *Espace Agents de la scolarité.*
- *Espace Enseignant.*
- *Espace Etudiant.*

II.2.4. Diagramme de contexte :

C'est un diagramme représentant les acteurs externes qui interagissent avec le système. Il est considéré comme la plus haute vue de ce système.

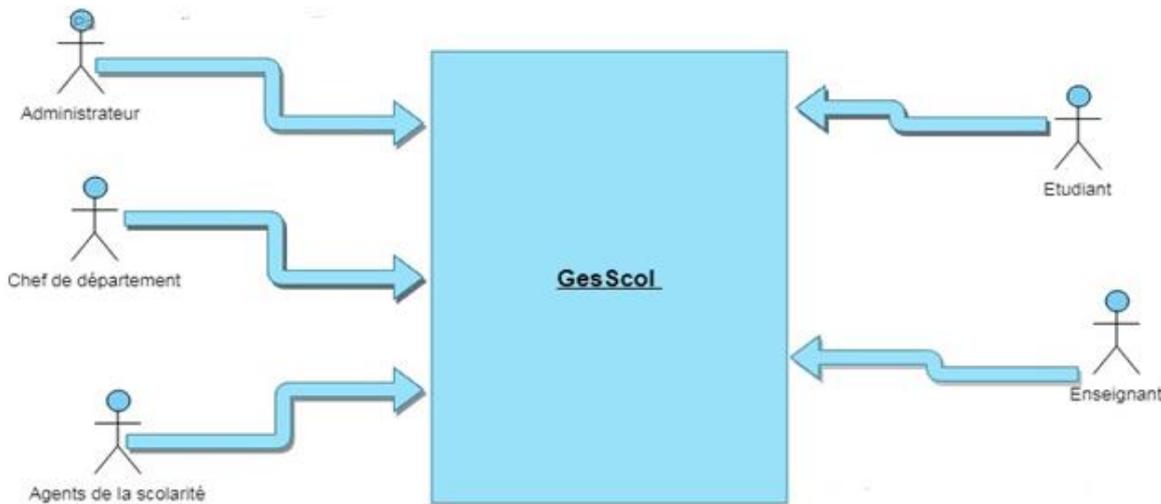


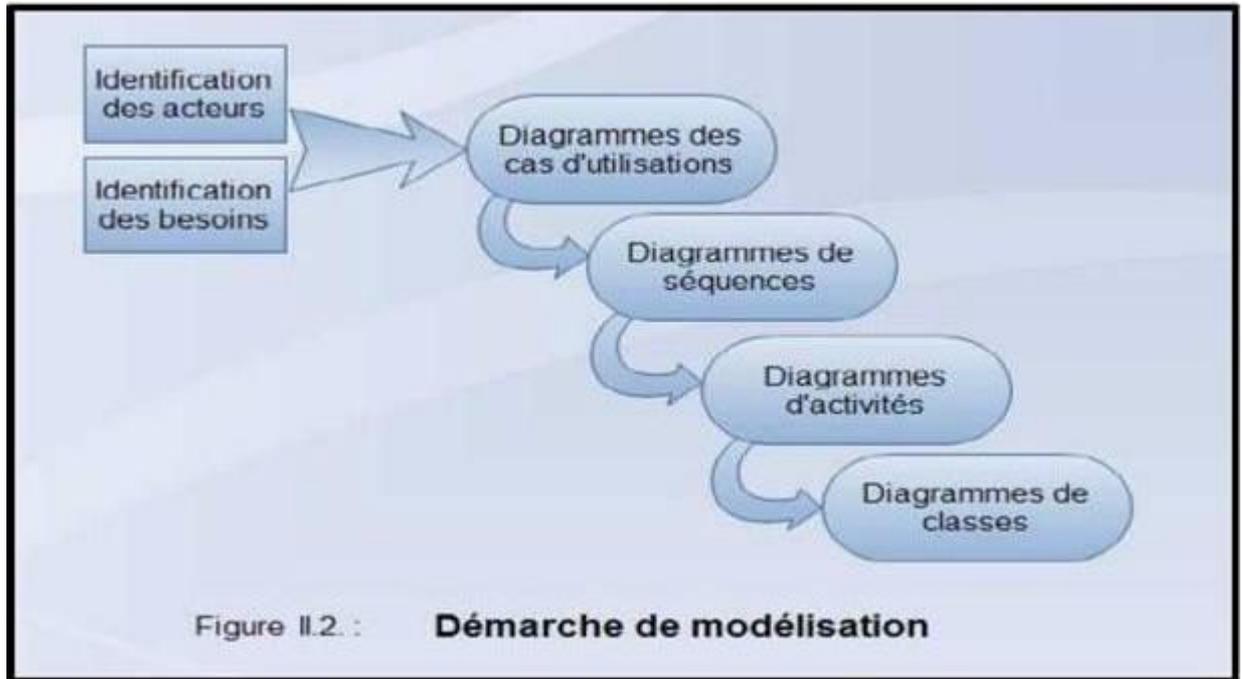
Figure II.1 : Diagramme de contexte.

II.3. Phase de Conception :

Le processus de conception de notre système comprend deux (02) niveaux : Niveau applicatif, Niveau données.

Niveau applicatif : S'appuie essentiellement sur quelques diagrammes de l'extension pour les applications Web du langage de modélisation UML. A cet effet, nous avons adopté la démarche suivante : **[Voir annexe A]**

- Mettre en évidence les cas d'utilisations mis en œuvre par les différents acteurs du système ;
- Elaborer le diagramme de cas d'utilisation global, puis les diagrammes de cas d'utilisation détaillés ;
- Formaliser graphiquement les scénarios décrivant chaque cas d'utilisation et ce grâce aux diagrammes de séquence ;
- Elaborer le diagramme d'activité permettant la représentation dynamique du système d'information ;
- Identifier les classes en s'inspirant des diagrammes de séquences, pour élaborer des diagrammes de classes.



Niveau de Données : Concerne l'organisation conceptuelle, logique et physique des données. Les données de l'application sont identifiées au niveau applicatif, les classes significatives seront dégagées. À ce stade, la conception de la base de données peut être élaborée.

II.3.1. Niveau Applicatif :

a. Diagramme des cas d'utilisations [7]:

Un cas d'utilisation est un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier.

Un cas d'utilisation modélise un service rendu par le système, exprime les interactions acteurs/système et apporte une valeur ajoutée notable à l'acteur concerné. Il permet de décrire ce que le futur système devra faire, sans spécifier comment le faire.

Un diagramme de cas d'utilisation est une vue statique du système représentant un ensemble de relations entre les cas d'utilisations et les acteurs.

- Identification des cas d'utilisations (tâches)

Acteur	Les tâches (T)
Administrateur	T : Se connecter au site. T : S'authentifier. T : Gestion des comptes des utilisateurs. T : Changer son mot de passe.

Acteur	Les tâches (T)
Chef de département	T : Se connecter au site. T : S'authentifier. T : Gestion des enseignants. T : Afficher la liste des étudiants. T : Gérer les PV de délibérations. T : Changer son mot de passe.

Acteur	Les tâches (T)
Agent de la scolarité	T : Se connecter au site. T : S'authentifier. T : Gestion des formations. T : Affectation des étudiants. T : Gestion des comptes. T : Edition des listes des étudiants. T : Edition des relevés de notes. T : Edition des PV globaux de délibérations. T : Changer son mot de passe.

Acteur	Les tâches (T)
Enseignant	T : Se connecter au site. T : S'authentifier. T : Consultation de la liste des étudiants. T : Consultation de la liste des enseignants. T : Consultation du planning des examens. T : Edition et affichage des PV modulaires. T : Consultation des PV de délibérations T : Changer son mot de passe.

Acteur	Les tâches (T)
Etudiant	T : Se connecter au site. T : S'authentifier. T : Consultation du planning des examens. T : Consultation des notes. T : Changer son mot de passe.

a.1. Diagramme du cas d'utilisation global :

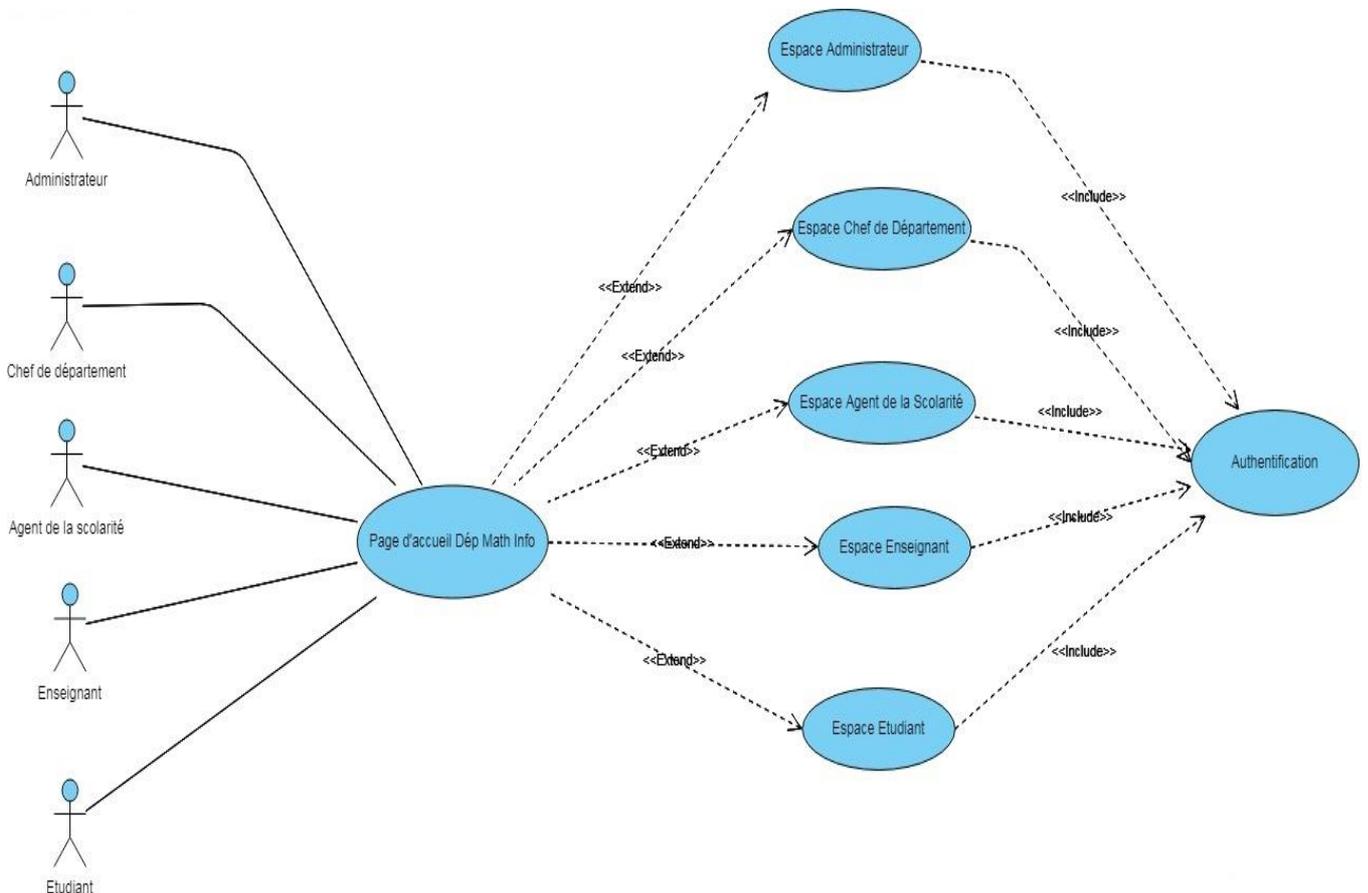


Figure II.3 : Diagramme du cas d'utilisation global.

a.2. Diagramme du cas d'utilisation « Espace Administrateur » :

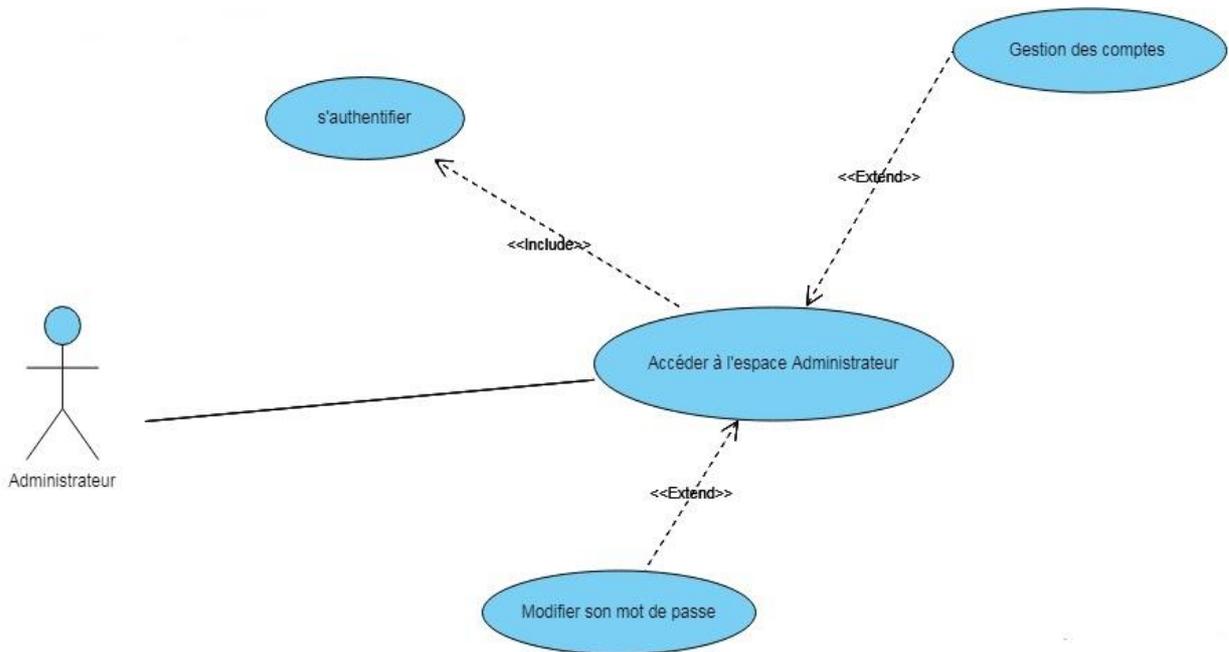


Figure II.4 : Diagramme du cas d'utilisation 'Espace Administrateur'.

a.3. Diagramme du cas d'utilisation « Espace Chef de Département» :

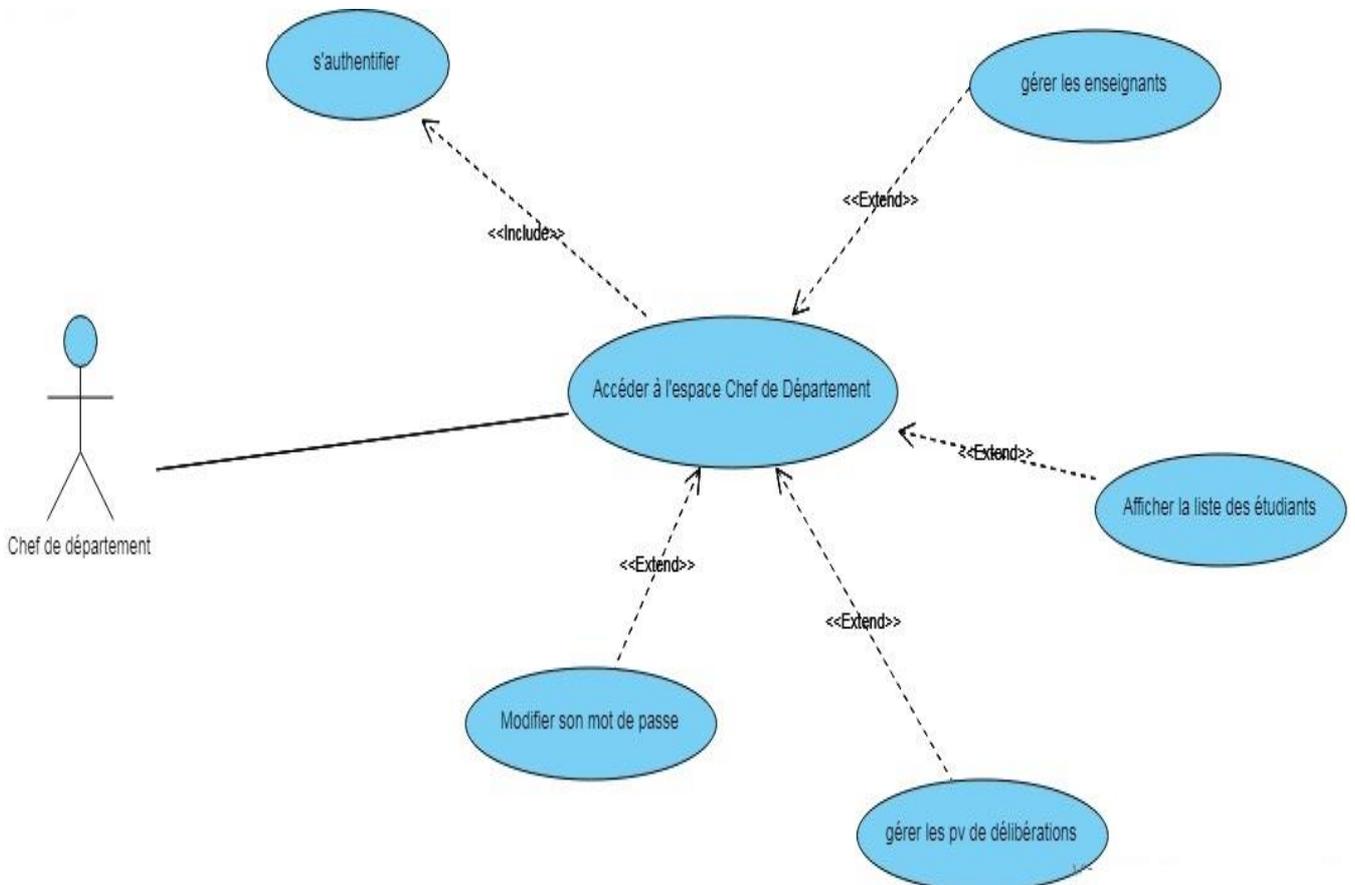


Figure II.5 : Diagramme du cas d'utilisation 'Espace Chef de département'.

a.4. Diagramme du cas d'utilisation « Espace agent de la scolarité» :

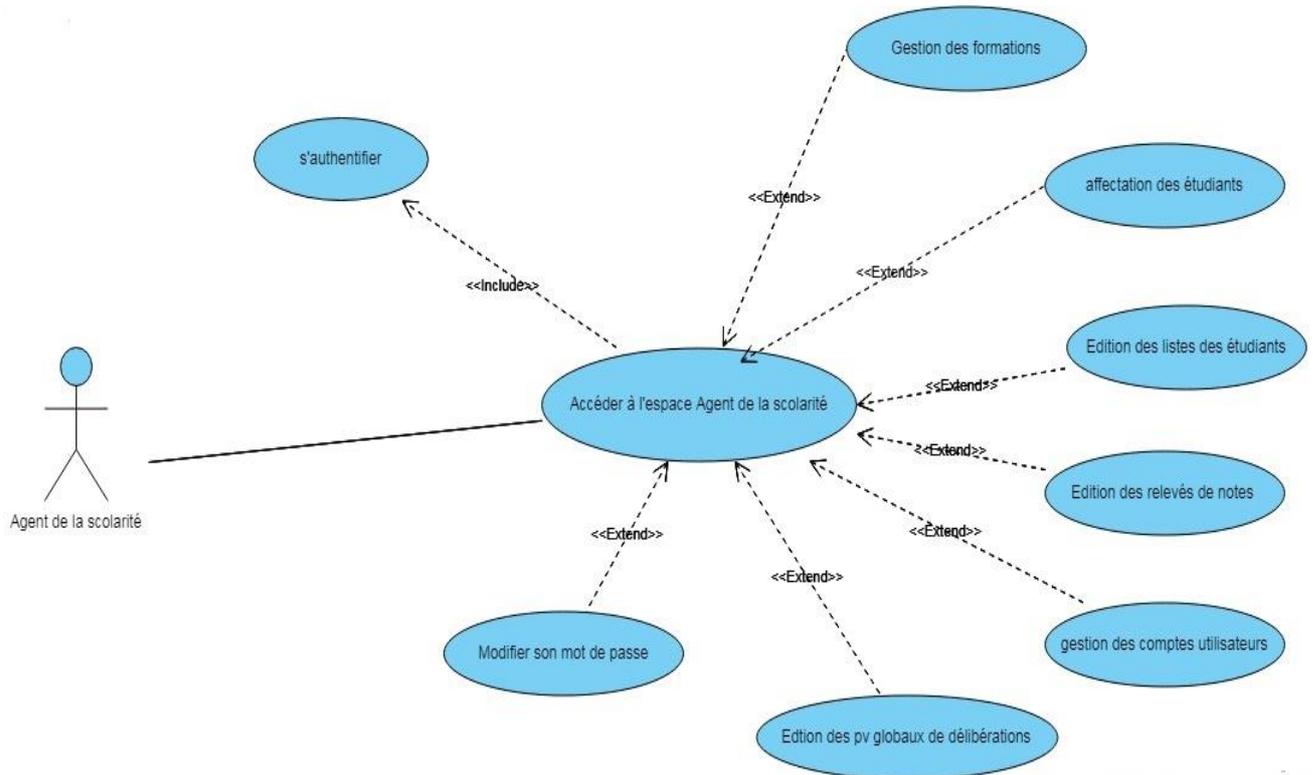


Figure II.6 : Diagramme du cas d'utilisation 'Espace Agent de la scolarité'.

a.5. Diagramme du cas d'utilisation « Espace Enseignant» :

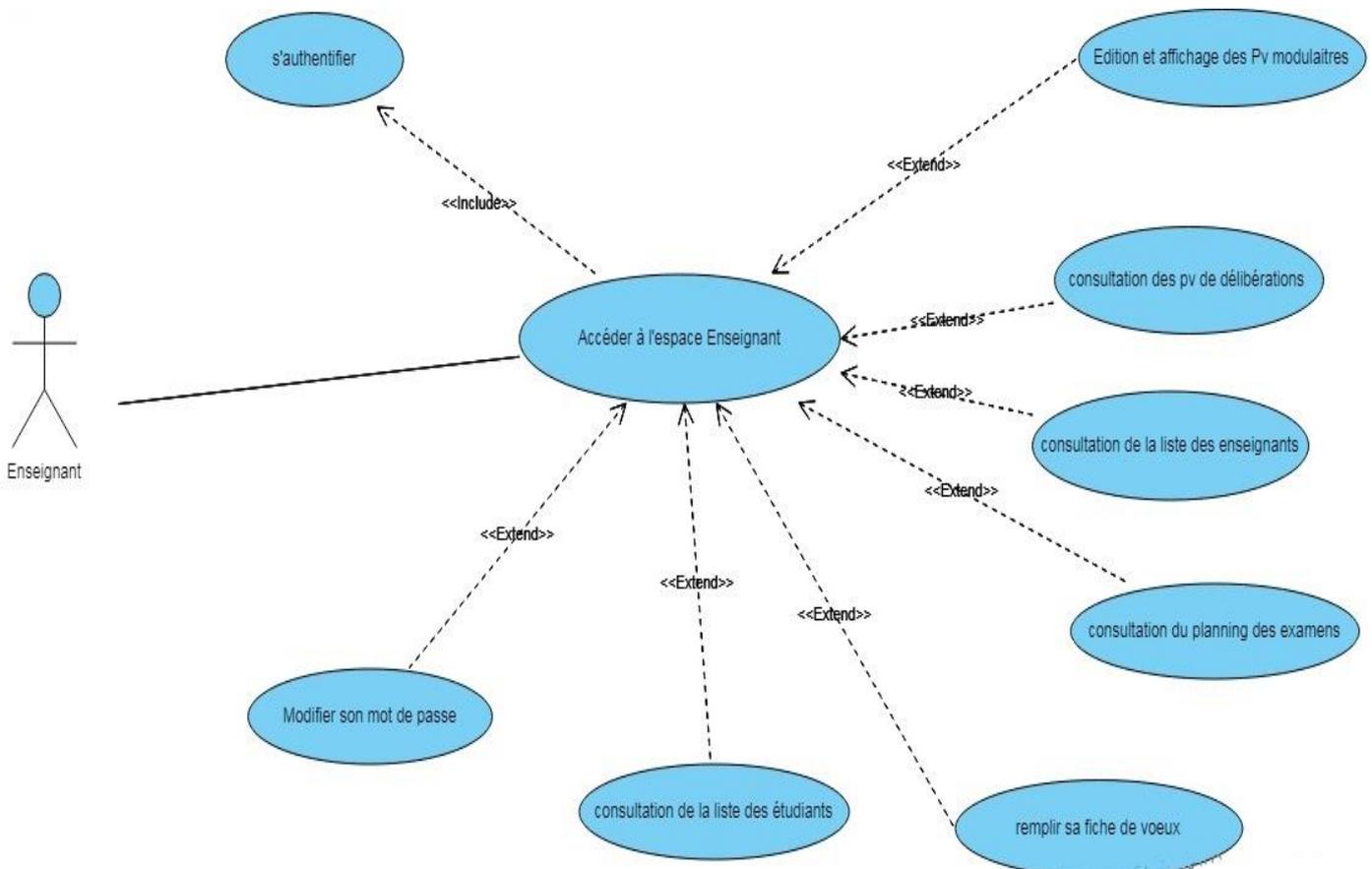


Figure II.7 : Diagramme du cas d'utilisation 'Espace Enseignant'.

a.5. Diagramme du cas d'utilisation « Espace Etudiant » :

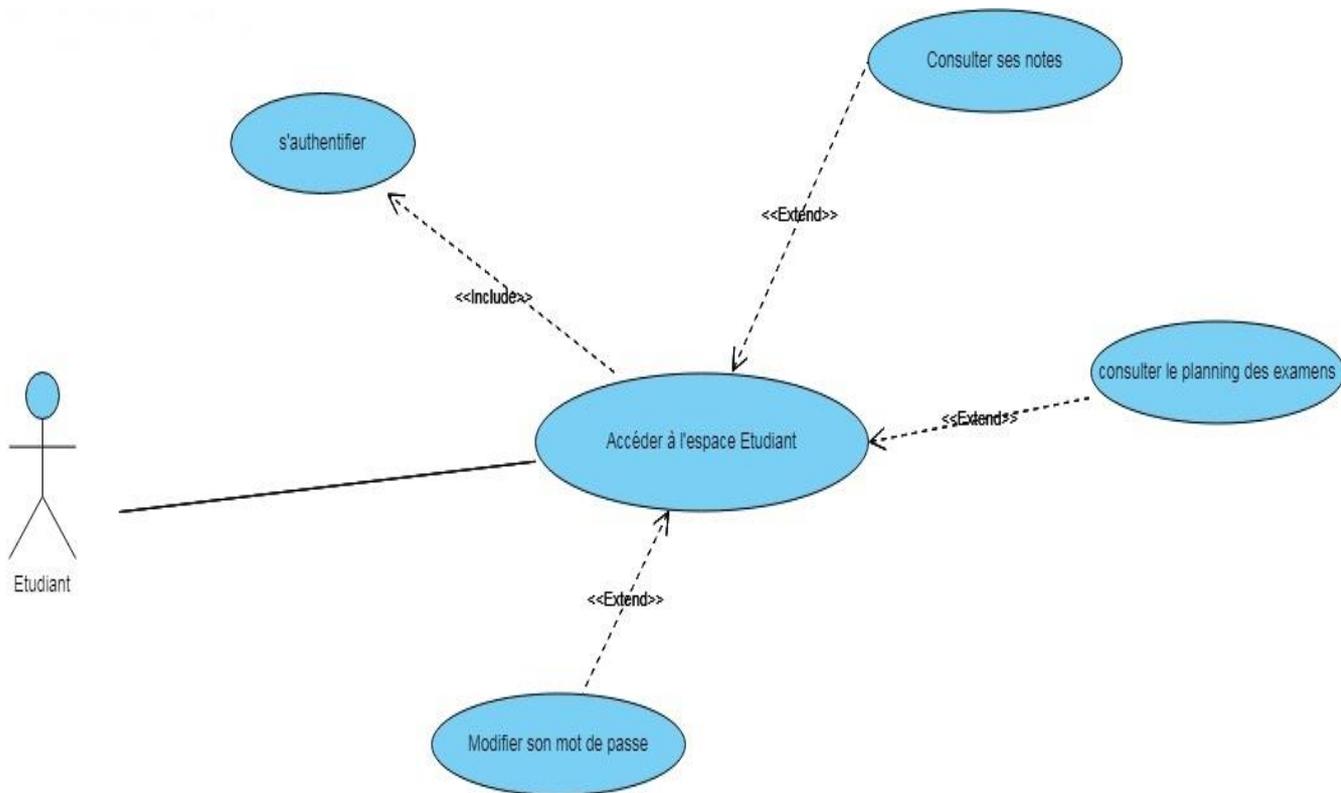


Figure II.8 : Diagramme du cas d'utilisation 'Espace Etudiant'.

b. Diagrammes de séquences [8]:

Après avoir décrit les cas d'utilisations détaillés à travers lesquels on peut identifier aisément les scénarios ; nous allons élaborer les diagrammes de séquences.

L'objectif de ce type de diagramme offert par UML est de représenter graphiquement les interactions entre les acteurs et le système selon un ordre chronologique. On montre ces interactions dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but étant de décrire comment se déroulent les actions entre acteurs ou objets. Les périodes d'activité des classes sont symbolisées par des rectangles.

Vu le nombre important de diagrammes de cas d'utilisation qu'on a, nous nous limitons aux diagrammes de séquences pour les cas d'utilisation représentés dans les figures qui vont suivre :

b.1. Diagramme de séquence du cas d'utilisation «s’authentifier»

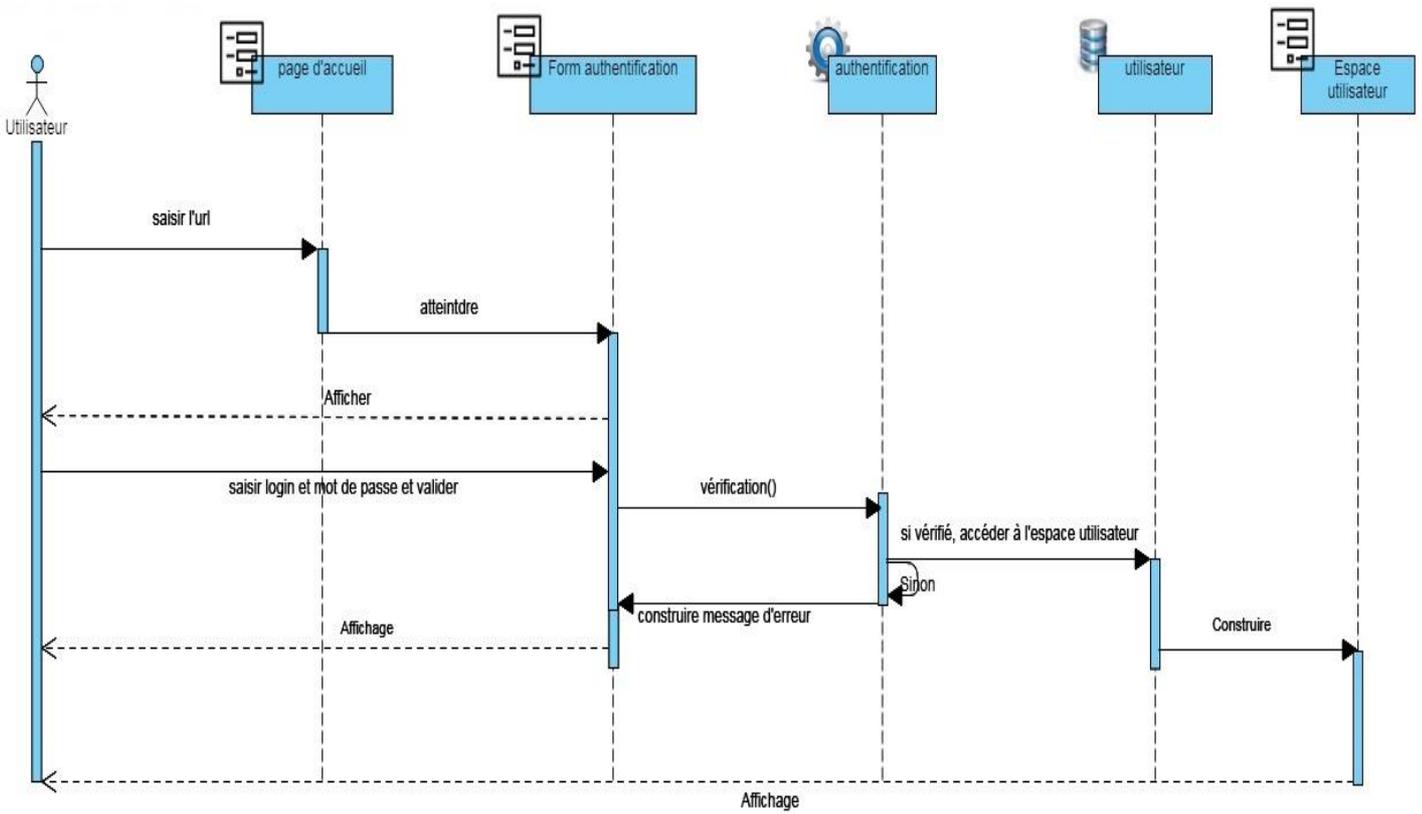


Figure II.9: Diagramme de séquence cas d'utilisation « s’authentifier»

b.2. Diagramme de séquence du cas d'utilisation «Création de compte» :

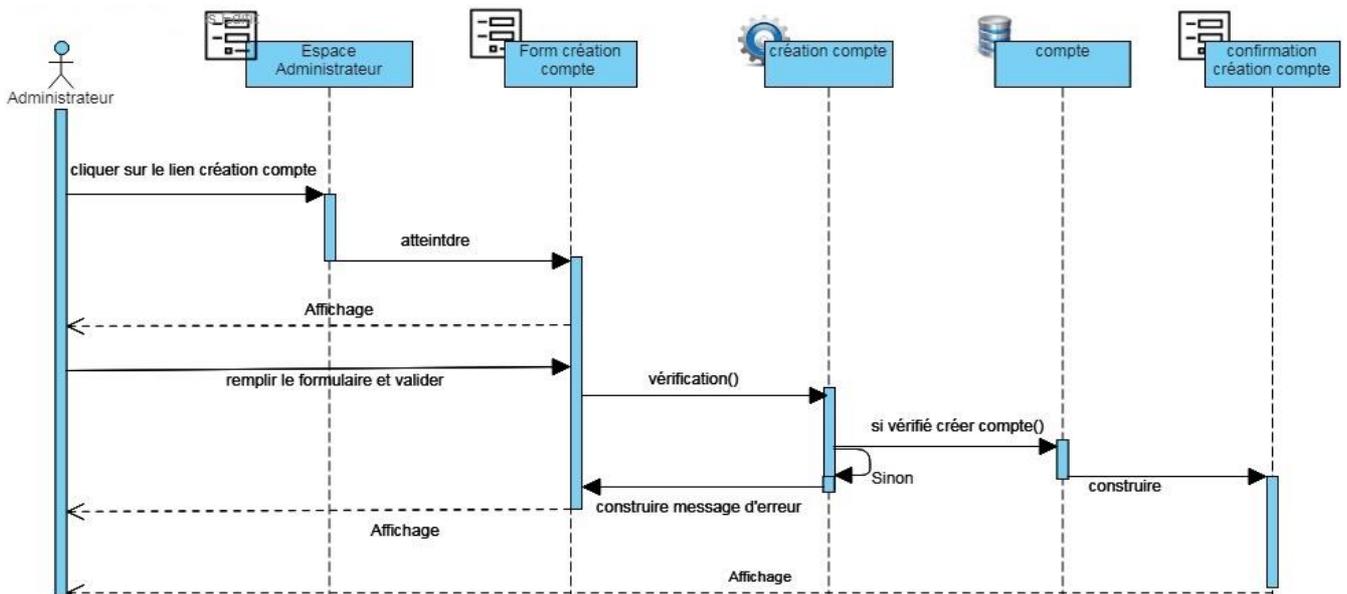
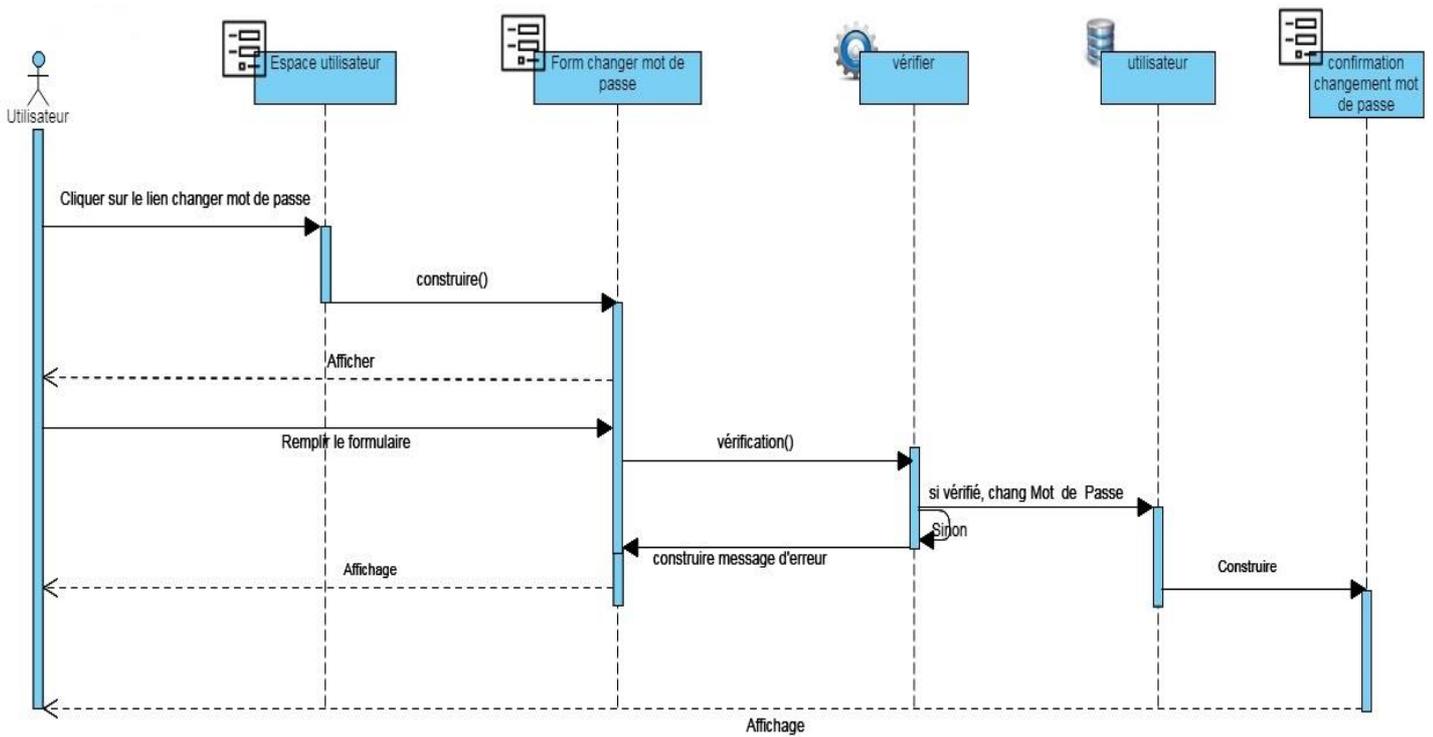


Figure II.10 : Diagramme de séquence cas d'utilisation « création de compte »

b.3. Diagramme de séquence du cas d'utilisation «Modifier mot de passe» :



Utilisateur : (Administrateur, chef de département , Agent de la scolarité, Enseignant, Etudiant)

Figure II.11 : Diagramme de séquence cas d'utilisation « Modifier mot de passe »

b.4. Diagramme de séquence du cas d'utilisation «Ajouter examen» :

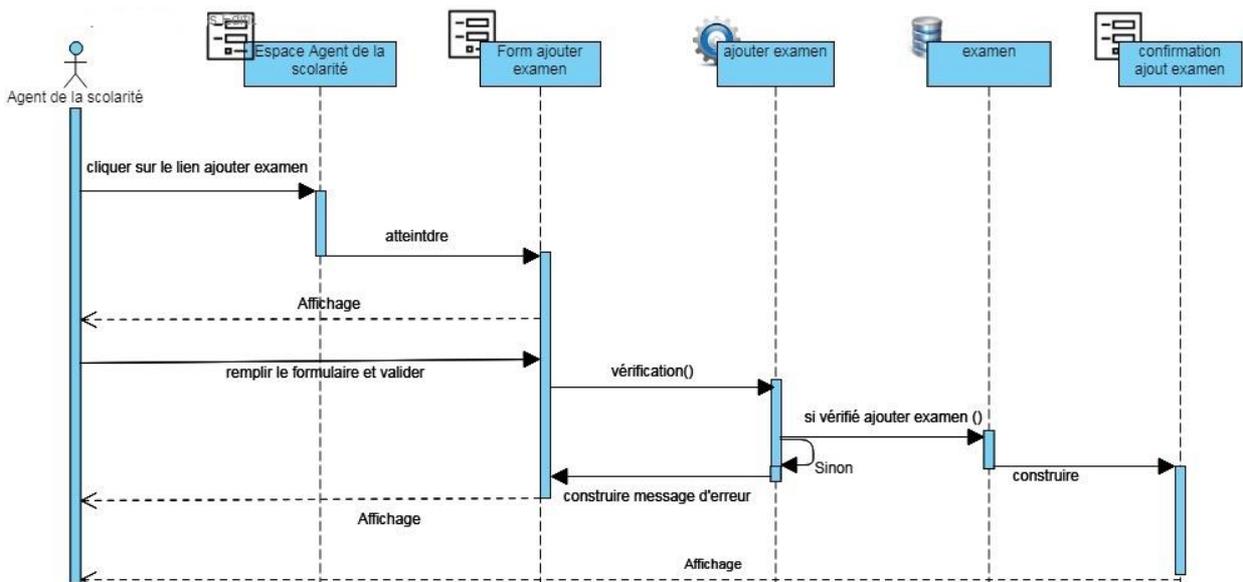


Figure II.12 : Diagramme de séquence cas d'utilisation « Ajout examen »

b.5. Diagramme de séquence du cas d'utilisation «éditer Pv modulaires» :

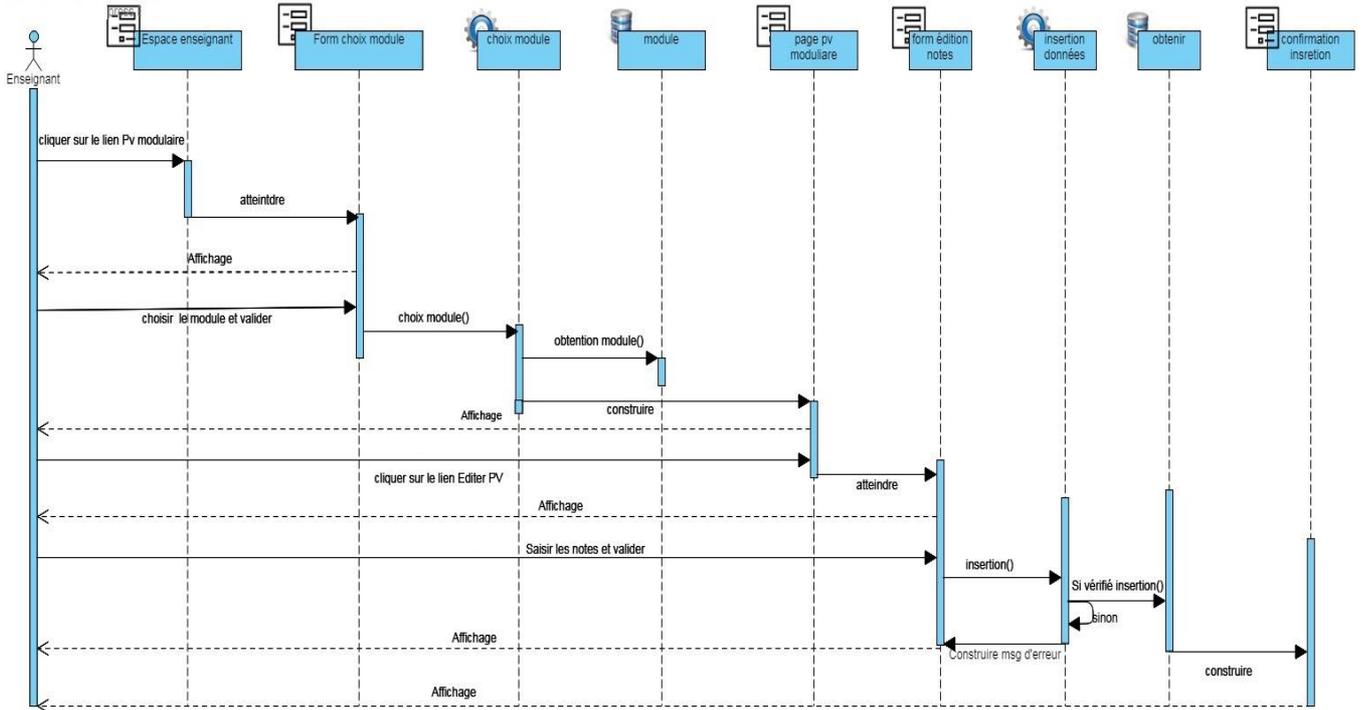


Figure II.13 : Diagramme de séquence cas d'utilisation «Editer PV modulaire »

b.6. Diagramme de séquence du cas d'utilisation «Afficher PV modulaires :

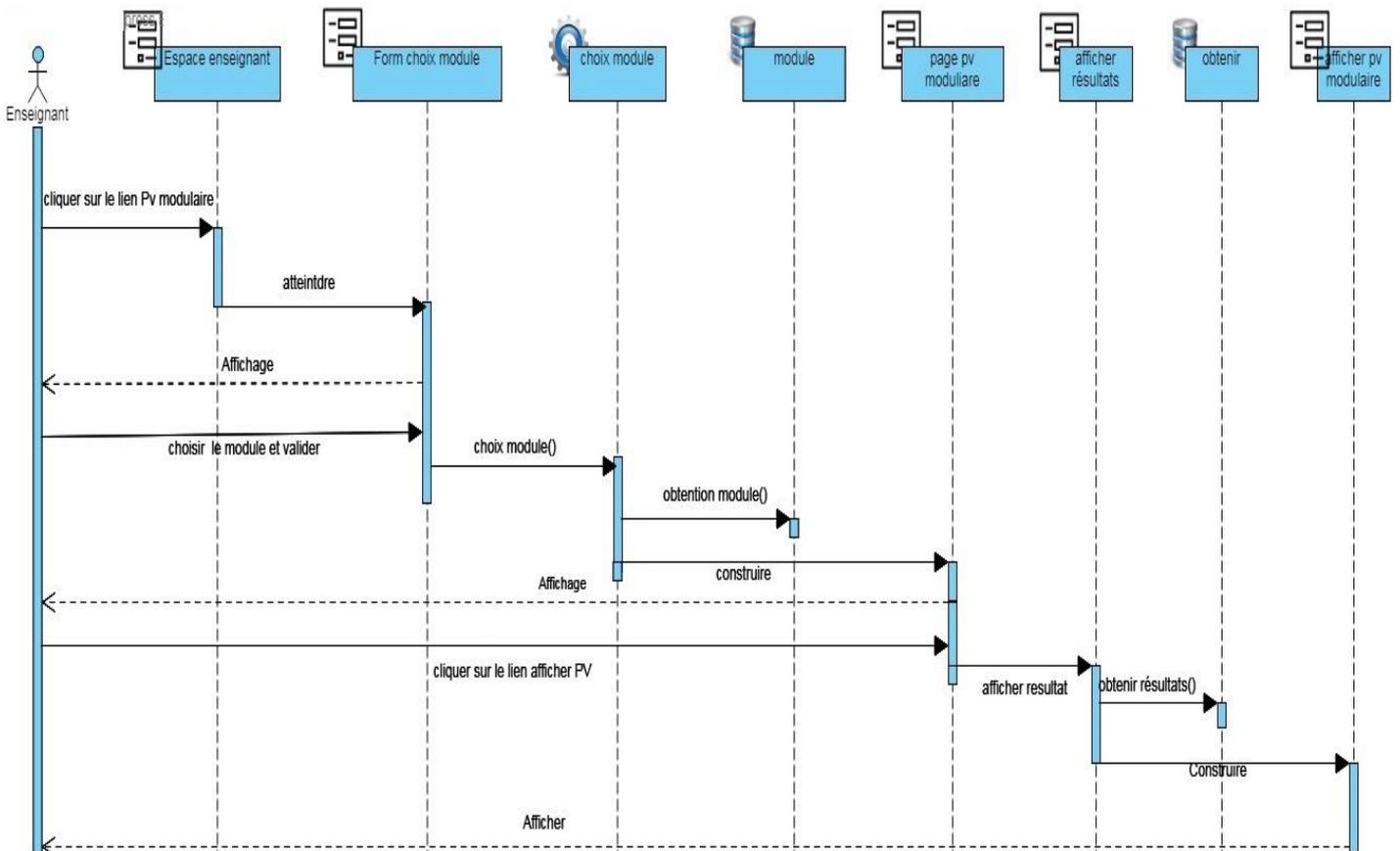


Figure II.14 : Diagramme de séquence cas d'utilisation « Afficher PV modulaire »

b.7. Diagramme de séquence du cas d'utilisation « Consulter ses notes » :

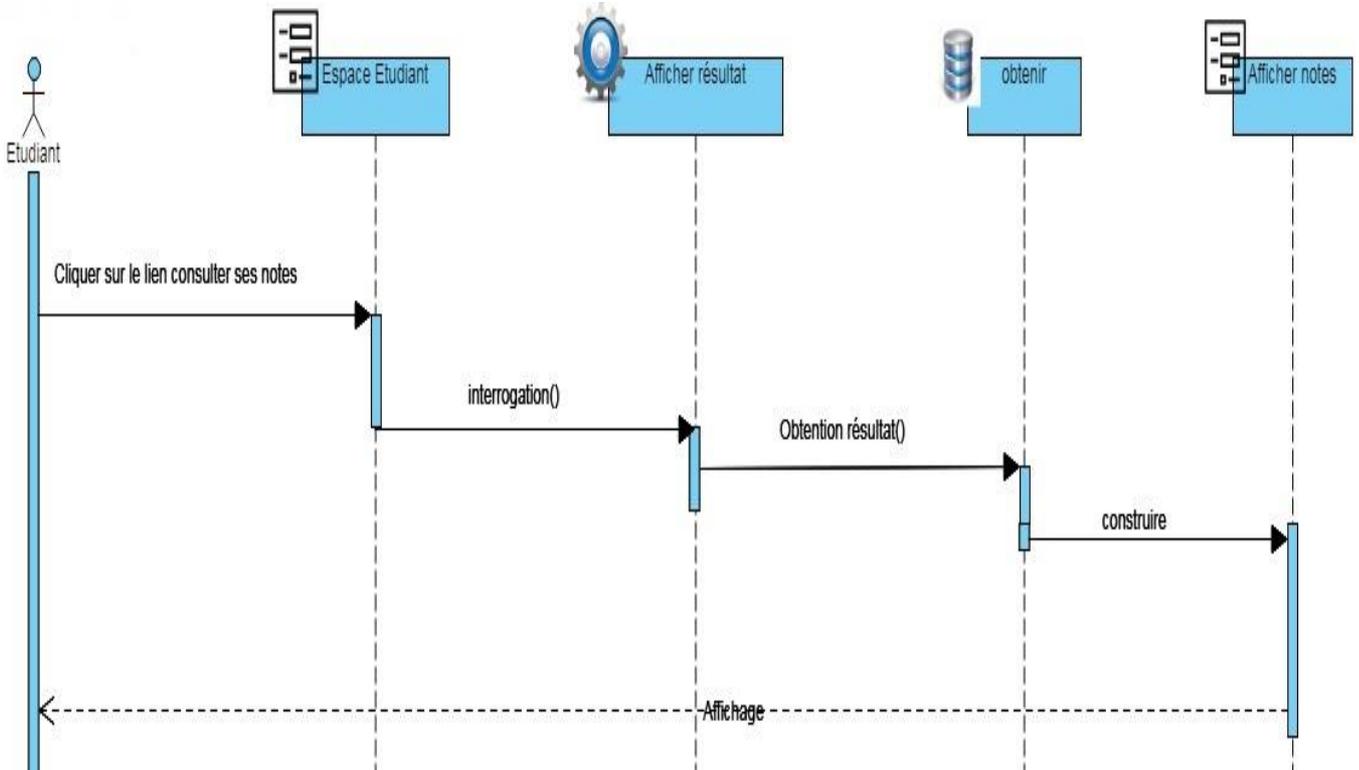


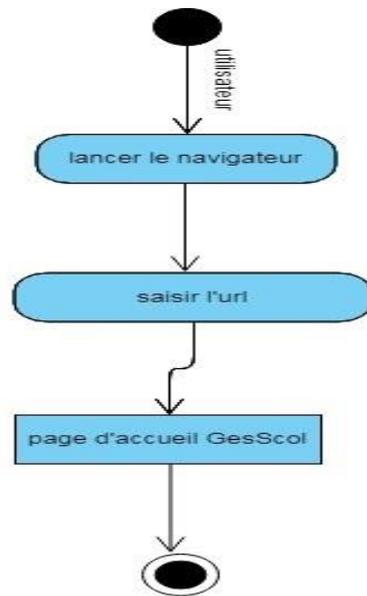
Figure II.15 : Diagramme de séquence cas d'utilisation « Consulter les notes »

c. Diagramme d'activités [8]:

Un diagramme d'activité permet de modéliser un processus interactif, global ou partiel pour un système donné (logiciel, système d'information). C'est une représentation proche de l'organigramme ; la description d'un cas d'utilisation par un diagramme d'activités correspond à sa traduction algorithmique.

Vu le nombre important de diagrammes de cas d'utilisation qu'on a, nous nous limitons aux diagrammes d'activités pour les cas d'utilisation représentés dans les figures qui vont suivre :

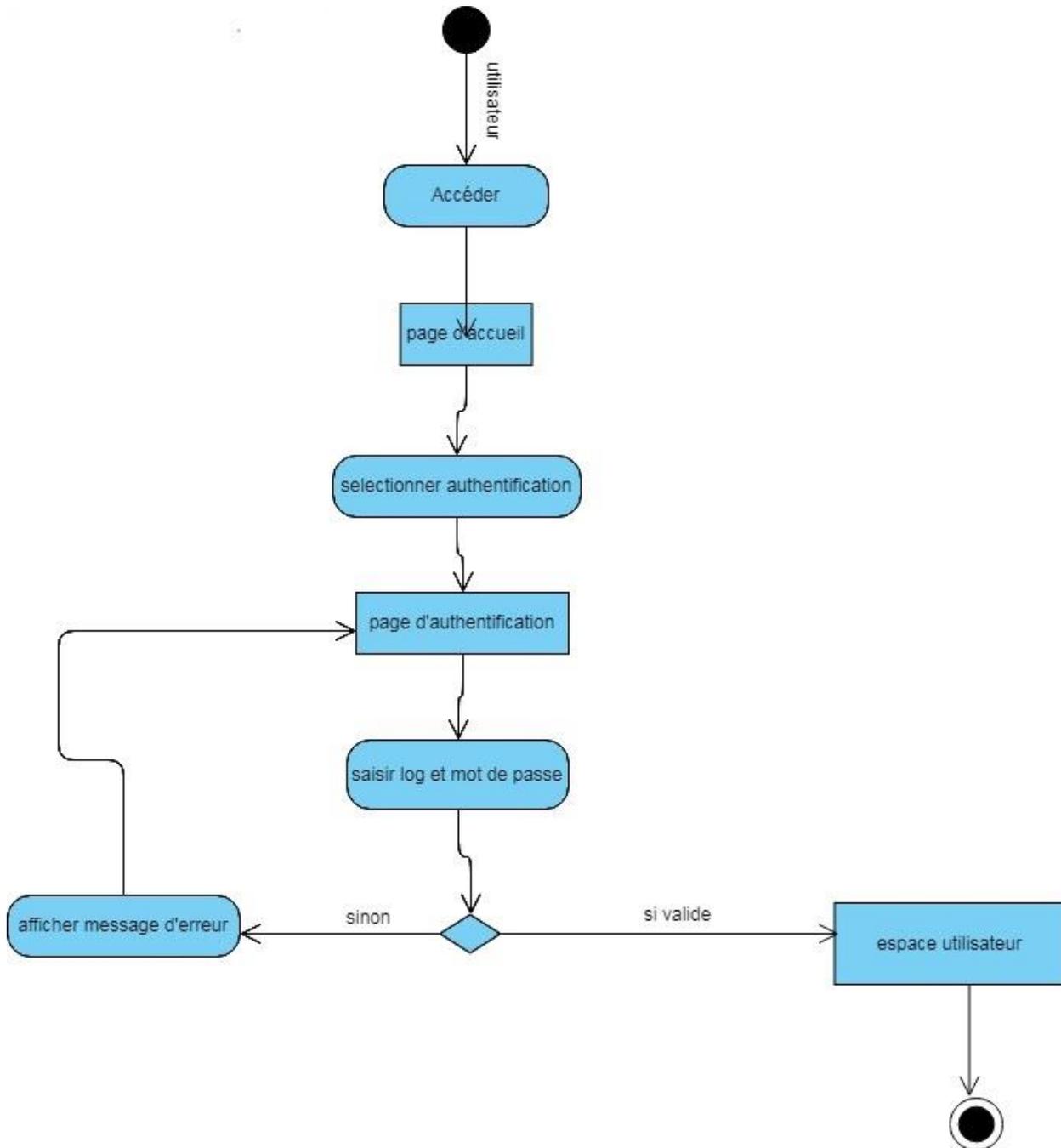
c.1. Diagramme d'activités du cas d'utilisation «Page principale » :



Utilisateur(Administrateur, chef de département, agent de la scolarité, enseignant, étudiant)

Figure II.16 : Diagramme d'activité cas d'utilisation « Page principale »

c.2. Diagramme d'activités du cas d'utilisation «authentification utilisateur» :



Utilisateur(Administrateur, chef de département, agent de la scolarité, enseignant, étudiant)

Figure II.17: Diagramme d'activité cas d'utilisation « Authentification utilisateur»

c.3. Diagramme d'activités du cas d'utilisation «Ajouter étudiant» :

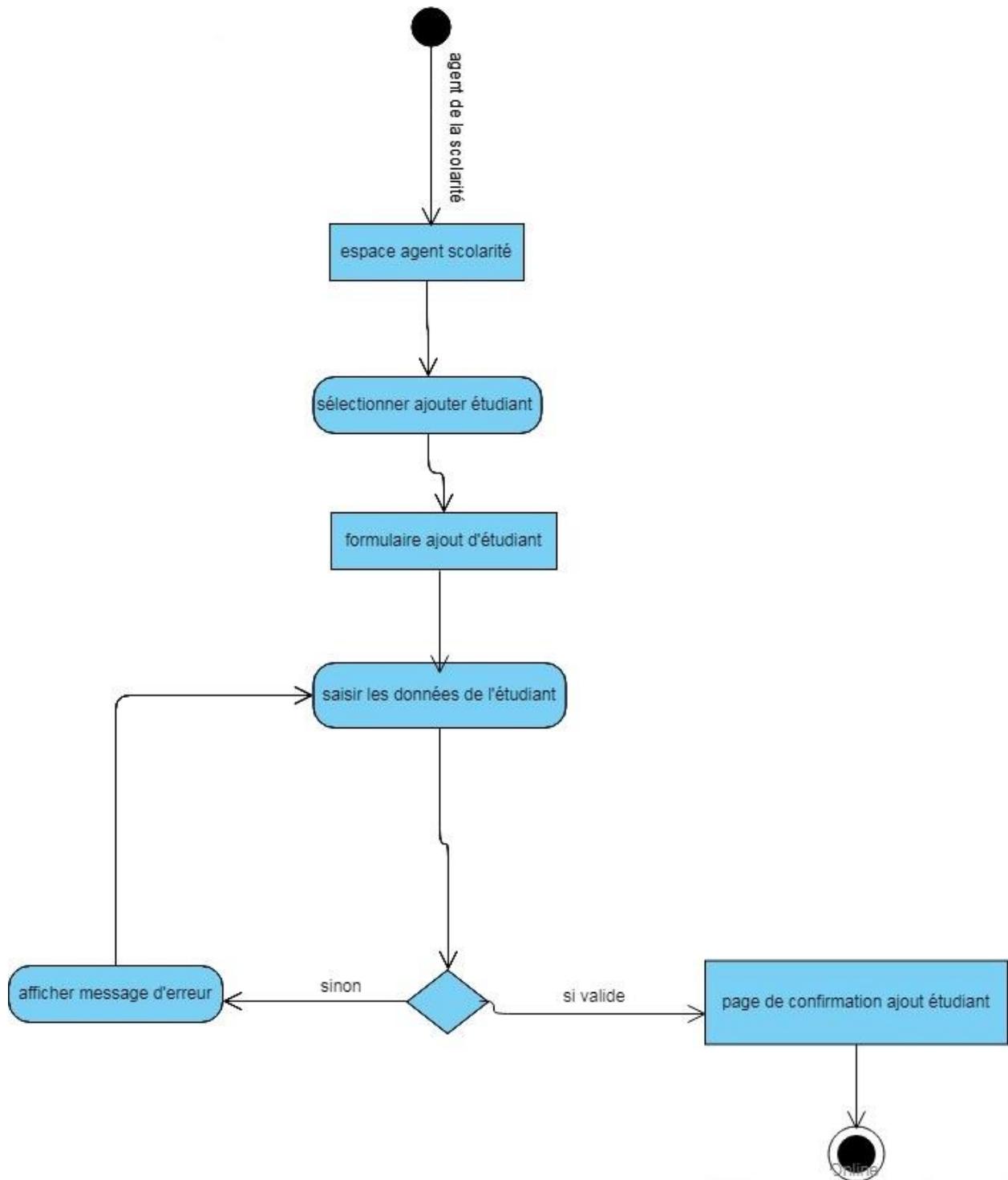


Figure II.18 : Diagramme d'activité cas d'utilisation « Ajouter étudiant»

c.4. Diagramme d'activités du cas d'utilisation «éditer pv modulaire» :

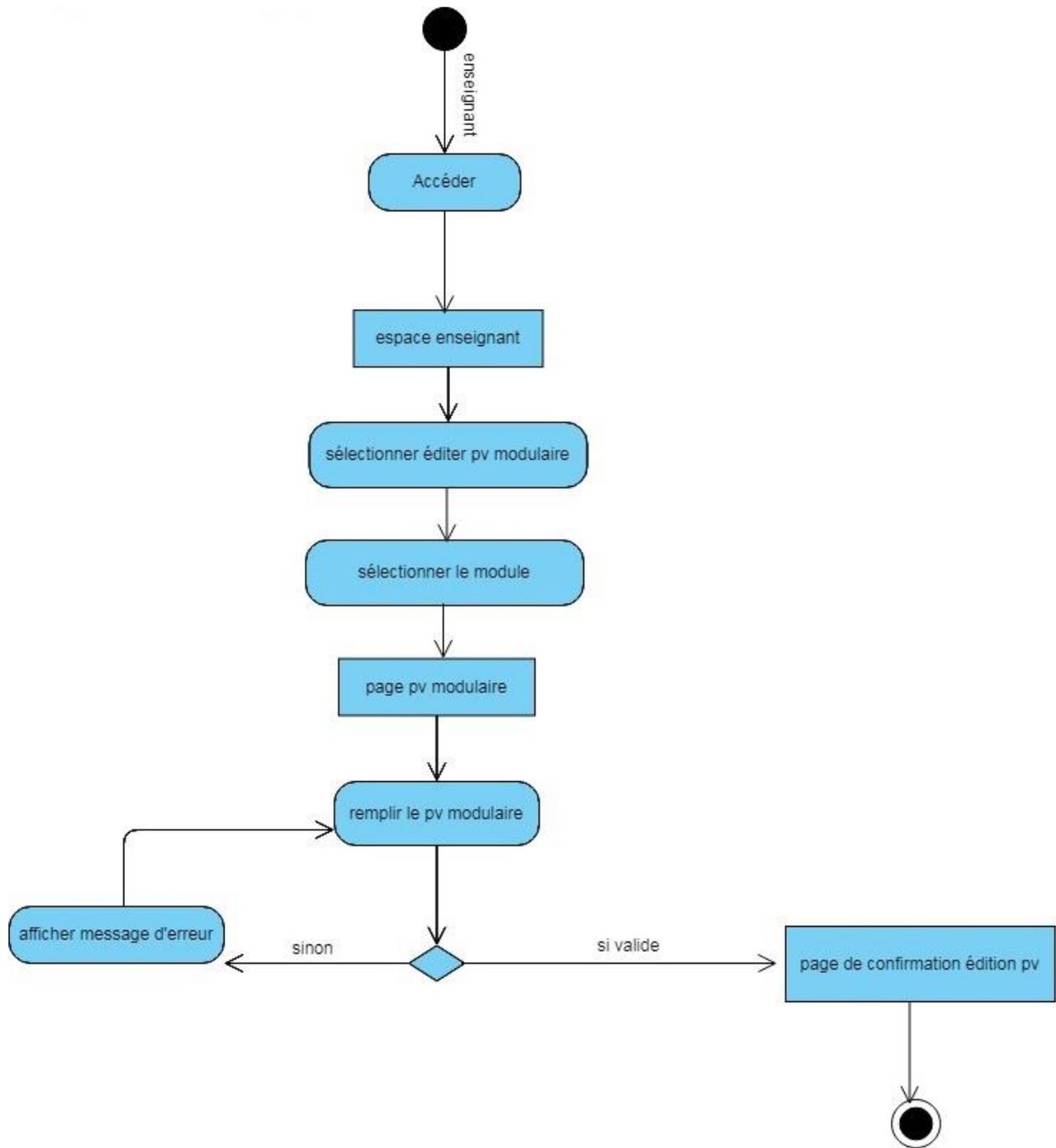


Figure II.19 : Diagramme d'activité cas d'utilisation « Editer PV modulaire»

c.5. Diagramme d'activités du cas d'utilisation «Consulter les notes» :

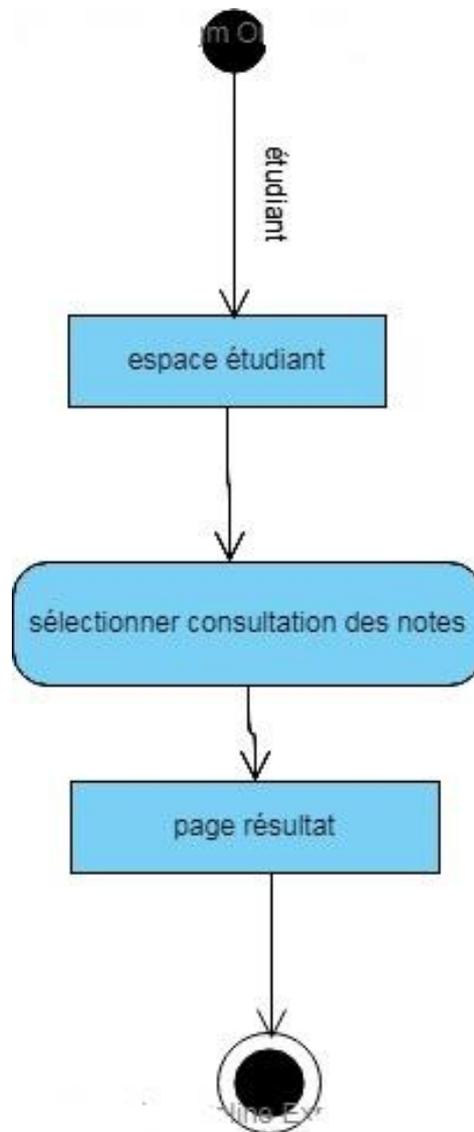


Figure II.20 : Diagramme d'activité cas d'utilisation « Consulter les notes »

c.6. Diagramme d'activités du cas d'utilisation «création compte» :

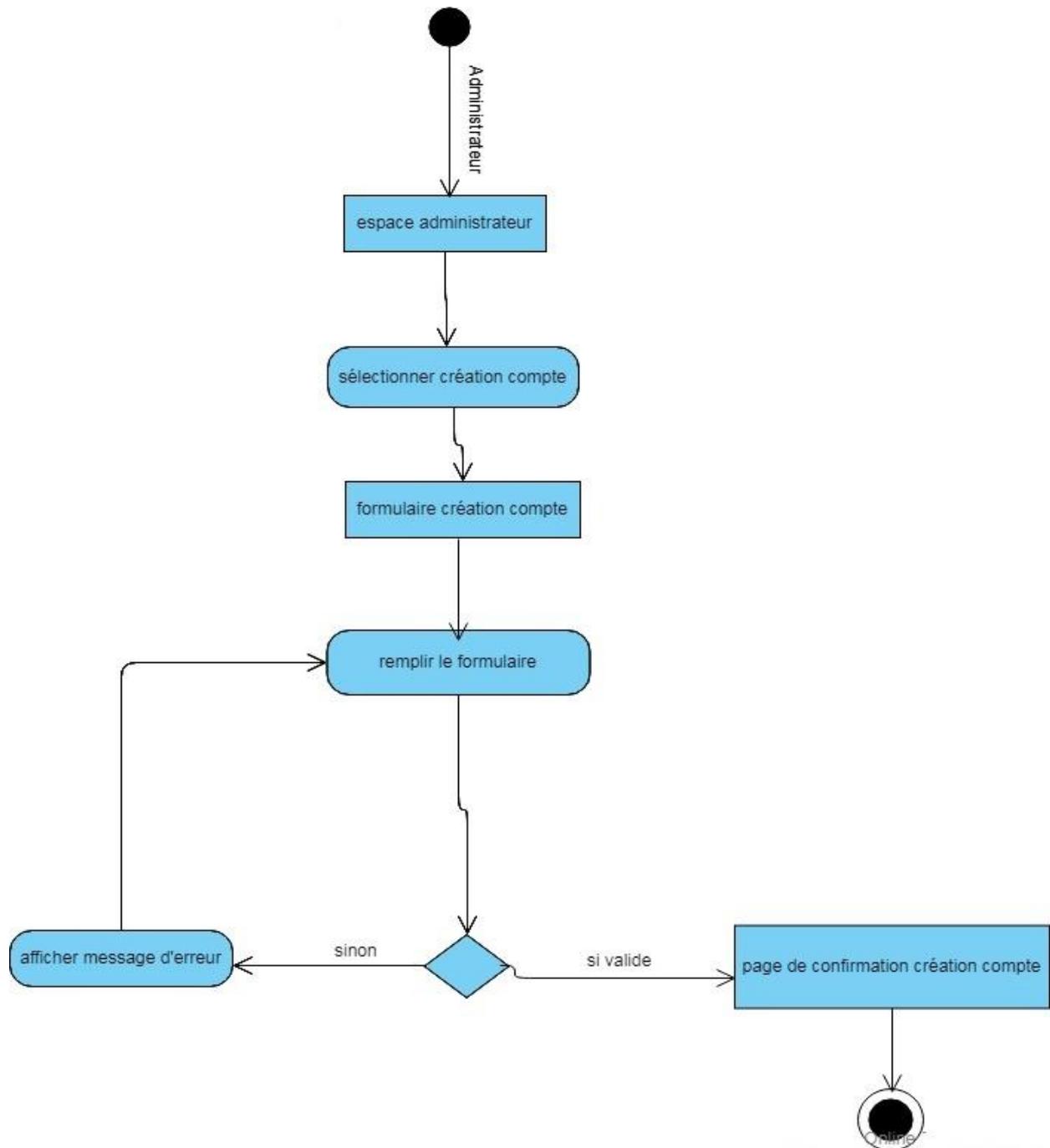


Figure II.21 : Diagramme d'activité cas d'utilisation « Création du compte»

d. Diagramme de classes [10] :

Le diagramme de classes constitue un élément très important de la modélisation du système orienté objet. Il permet de définir quelles seront les composantes du système final, mais ne permet pas de définir le nombre et l'état des instances individuelles.

Néanmoins, on constate souvent qu'un diagramme de classes proprement réalisé permet de structurer le travail de développement d'une manière très efficace.

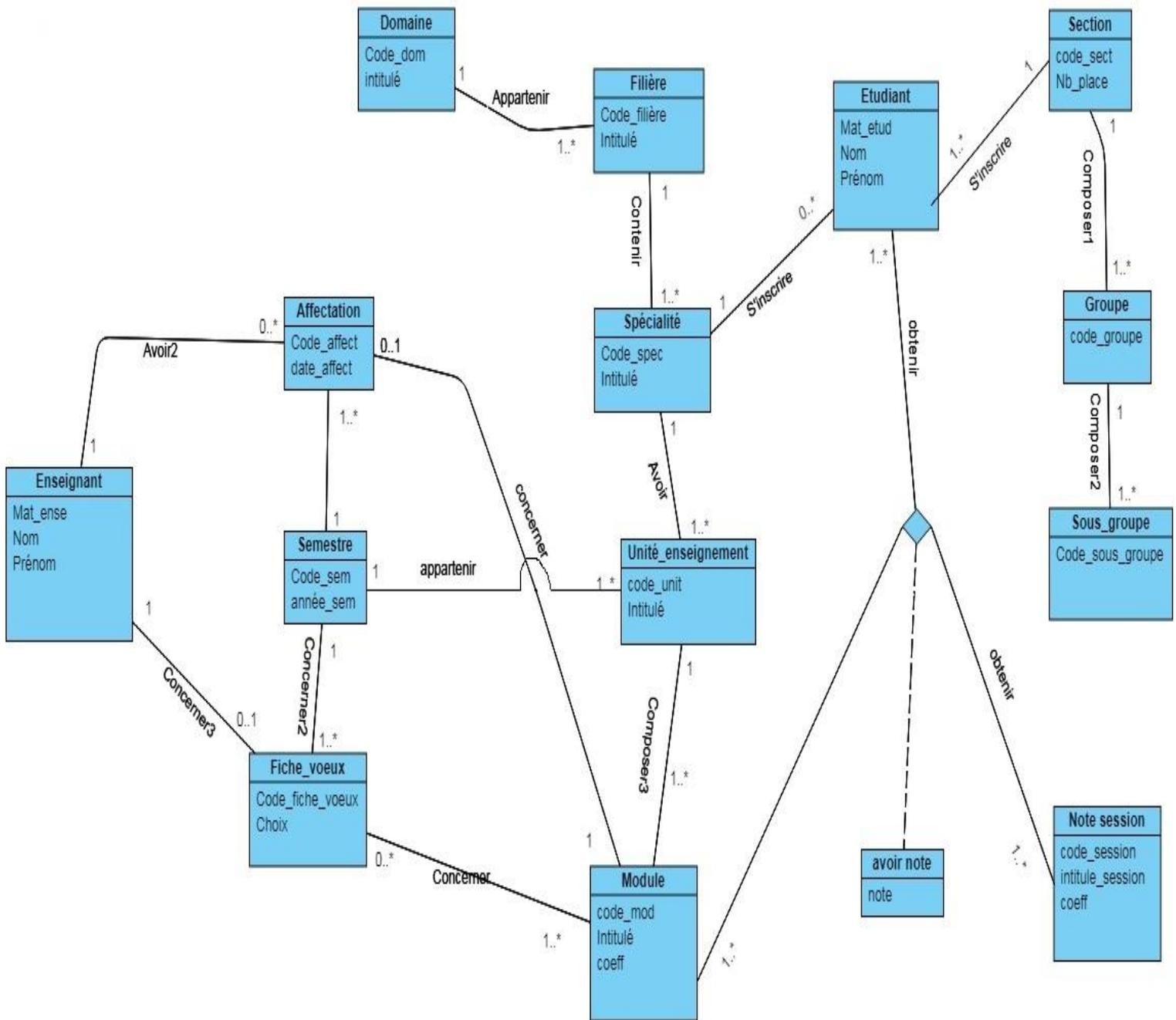


Figure II.22 : Diagramme de classe.

II.3.2. Niveau de données :

II.3.2. 1. Définition d'une base de données

Une Base de Données (BDD) est un ensemble de données structurées, avec une redondance calculée et accessibles aisément par plusieurs programmes, qui les utilisent selon des objectifs distincts, les propriétés des données ainsi que les relations sémantiques entre ces données sont spécifiées en utilisant les concepts proposés par le modèle de donnée, cette organisation à pour objectif de pouvoir effectuer des inférences sur ces données sous forme de requêtes. Les bases de données sont manipulées par un système de gestion de base de données (SGBD)

II.3.2. 2. Définition d'un SGBD

Un Système de Gestion de Base de Données (SGBD) est un outil permettant d'insérer, de modifier et de rechercher des données spécifiques dans une masse d'informations. C'est aussi une interface entre l'utilisateur et la mémoire secondaire permettant l'exploitation des informations stockées. Un SGBD est composé de trois couches : un système de gestion de fichier, un SGBD interne et un SGBD externe.

Il existe une multitude de types SGBD, ils se rejoignent sur un ensemble d'aspects fondamentaux. Parmi les SGBD les plus connus : Oracle, Microsoft SQL Server, MySQL, Sybase SQL, IBM DB2.

II.3.2. 3. Modèle relationnel

A partir du diagramme de classe globale nous construisons le modèle relationnel qui va nous permettre de créer la base de donnée sur la quelle s'appuie notre application comme suit :

enseignant (mat_ense, `nom`, `prenom`, `date_nais`, `lieu_nais`, `commune_nais`, `wilaya_nais`, `sexe`, `adresse`, `grade`, `specialite`, `email`, `num_tel`, `login_ense`)

fiche_voeux (code_fiche_voeux, `#mat_ense`, `#code_sem`, `#code_mod`, `#code_spec`)

etudiant (mat_etud, `#code_sect`, `#code_filiere`, `nom`, `prenom`, `date_nais`, `lieu_nais`, `commune_naiss`, `wilaya_naiss`, `sexe`, `situation_fam`, `nationalite`, `groupe_sang`, `nom_mere`, `prenom_mere`, `prenom_pere`, `adresse`, `wilaya_resid`, `num_tel`, `serie_bac`, `libelle_bac`, `annee_obtent_bac`, `code_filiere_bac`, `wilaya_bac`, `login_etud`)

affectation (code_affect, `#mat_ense`, `#code_mod`, `#code_sem`, `date_affect`)

domaine (code_dom, `intitule_dom`)

filiere (code_filiere, `#code_dom`, `intitule_filiere`)

specialite (code_spec, `#code_filiere`, `intitule`)

section (code_sect, `nb_place`, `intitule`)

groupe (code_groupe, `#code_sect`, `nb_etudiant`, `intitule_groupe`)

sous_groupe (code_sous_group, `#code_groupe`, `nb_place`, `intitule`)

unite_enseignement (code_unit, `#code_spec`, `intitule`, `nature`, `nb_heures`, `credit`, `#code_sem`)

module (code_mod, `#code_unite`, `intitule_mod`, `coeff`, `credit`, `vh_cours`, `vh_td`, `vh_tp`)

avoir_note (code_session, `mat_etud`, `code_mod`, `note`)

note_session (code_session, `intitule_session`)

semestre (code_sem, `anne_sem`, `intitule`)

Notre base de données contient les tables suivantes :

- Table Enseignant :

Structure de table | Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	mat_ense	int(8)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	nom	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
3	prenom	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
4	date_nais	date			Oui	NULL			Modifier Supprimer Plus
5	lieu_nais	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
6	commune_nais	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
7	wilaya_nais	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
8	sexe	char(1)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
9	adresse	char(50)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
10	grade	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
11	email	char(20)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
12	num_tel	decimal(8,0)			Oui	NULL			Modifier Supprimer Plus
13	cours_dispense	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
14	login_ense	varchar(40)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

- Table Etudiant :

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	mat_etud	int(8)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	code_sect	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	code_filiere	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
4	nom	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
5	prenom	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
6	date_nais	date			Oui	NULL			Modifier Supprimer Plus
7	lieu_nais	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
8	commune_naiss	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
9	wilaya_naiss	char(20)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
10	sexe	char(1)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
11	situation_fam	varchar(255)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
12	nationalite	char(20)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
13	groupe_sang	char(2)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
14	nom_mere	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
15	prenom_mere	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
16	prenom_pere	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
17	adresse	char(50)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
18	wilaya_resid	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
19	num_tel	int(12)			Oui	NULL			Modifier Supprimer Plus
20	serie_bac	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
21	libelle_bac	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
22	annee_obtent_bac	date			Oui	NULL			Modifier Supprimer Plus
23	code_filiere_bac	char(4)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
24	wilaya_bac	char(20)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
25	logon_etud	varchar(40)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Table Affectation :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/>	1	code_affect			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
<input type="checkbox"/>	2	mat_ense			Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	3	code_mod			Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	4	code_sem			Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	5	date_affect			Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Table Fiche de vœux :

Parcourir Structure SQL Rechercher Insérer Exporter Importer Privilèges Opérations Déclencheurs

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/>	1	code_fiche_voeux			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
<input type="checkbox"/>	2	mat_ense			Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	3	code_sem			Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	4	choix	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
<input type="checkbox"/>	5	code_mod			Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	6	code_spec			Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	7	username	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
<input type="checkbox"/>	8	privilege			Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Table Domaine :

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/>	1	code_dom			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
<input type="checkbox"/>	2	intitule_dom	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter colonne(s) après intitule_dom Exécuter

Table Filière :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_filiere	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	code_dom	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	intitule_filiere	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après intitule_filiere Exécuter

Table Groupe :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_groupe	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	code_sect	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
3	nb_etudiant	int(2)			Oui	NULL			Modifier Supprimer Plus
4	intitule_groupe	varchar(30)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après intitule_groupe Exécuter

Table Sous_groupe :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_sous_groupe	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	code_groupe	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	nb_place	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
4	intitule	varchar(30)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après intitule Exécuter

Table Unité d'enseignement:

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_unit	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	code_spec	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	intitule	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
4	nature	varchar(25)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
5	nb_heures	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
6	credit	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
7	code_sem	int(11)			Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après code_sem Exécuter

Table module :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_mod	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	code_unite	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
3	intitule_mod	varchar(30)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	coeff	int(1)			Non	Aucun(e)			Modifier Supprimer Plus
5	credit	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
6	vh_cours	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
7	vh_td	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
8	vh_tp	int(2)			Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après vh_tp Exécuter

Table Spécialité :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_spec	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	code_filiere	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	intitule	char(30)	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après intitule Exécuter

Table Section :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_sect	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	nb_place	int(2)			Oui	NULL			Modifier Supprimer Plus
3	intitule	varchar(30)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après intitule Exécuter

Table Avoir_note :

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_session	varchar(10)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
2	mat_etud	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
3	code_mod	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
4	note_emd	int(2)			Oui	NULL			Modifier Supprimer Plus
5	note_td	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
6	note_tp	int(2)			Non	Aucun(e)			Modifier Supprimer Plus
7	note_finale	int(2)			Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Table Note_session :

Parcourir Structure SQL Rechercher Insérer Exporter Importer Privilèges Opérations Déclencheurs

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_session	varchar(10)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
2	intitule_session	varchar(30)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après intitule_session Exécuter

Table Semestre:

Structure de table Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	code_sem	int(2)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	anne_sem	text	utf8_general_ci		Oui	NULL			Modifier Supprimer Plus
3	intitule	varchar(30)	utf8_general_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

Imprimer Suggérer des optimisations de structure Déplacer des colonnes Normaliser

Ajouter 1 colonne(s) après intitule Exécuter

II.4. Conclusion :

Le présent chapitre a été consacré à l'analyse et à la conception du système. On a recensé en premier lieu les besoins des utilisateurs tout en s'inspirant des entretiens et interviews qu'on a effectués avec eux. Ces besoins ont été traduits par la suite, en formalismes graphiques à l'aide des diagrammes offerts par le langage UML ; en s'appuyant sur l'analyse, la partie conception a mis en évidence les différents composants de l'application. Après réflexion et élaboration d'un plan issu de la conception, et donc place à la réalisation qui fera l'objet du prochain chapitre.

CHAPITRE 3
R É A L I S A T I O N

III.1. Introduction :

Nous allons présenter dans ce chapitre l'environnement de développement et les différents outils utilisés pour réaliser notre application, puis nous expliquerons son fonctionnement en présentant quelques interfaces illustratives.

III.2. Environnement de développement :

Pour l'environnement logiciel, nous avons opté pour la configuration suivante :

Configuration logicielle :

- Système d'exploitation : *Windows 7* ;
- Visual Studio Code version 1.34.0;
- Serveur de base de données, contenant :
 - Un serveur *MySQL* ;
 - Un serveur *Apache 2.4.37* ;
 - Une interface graphique *phpMyAdmin*.

Langages de programmation :

- *HTML et CSS* ;
- *PHP* ;
- *JavaScript* ;
- *SQL*.

III.3. Outils utilisés :

III.3.1. Les logiciels utilisés :

a. Windows 7 :

Il est utilisé en raison de sa simplicité d'utilisation, sa popularité, sa convivialité et surtout pour sa puissance ainsi pour toutes les ressources et les options qu'il propose.

b. Visual Studio Code version 1.34.0:

Visual Studio Code est présenté lors de la conférence des développeurs Build d'avril 2015 comme un éditeur de code cross-platform, open source et gratuit, supportant une dizaine de langages.

Le code source est fourni sous la licence libre MIT (plus précisément la licence Expat) sur le site du projet sur Github. En revanche, l'exécutable est proposé sur le site officiel de Microsoft sous une licence privative.

C'est un éditeur *HTML*, destiné à la conception, au codage et au développement de sites, d'applications web et desktop. Quelque soit l'environnement de travail utilisé, il nous permet la visualisation de l'apparence des interfaces et l'intégration des scripts *PHP* et *JavaScript* dans la page *HTML*.

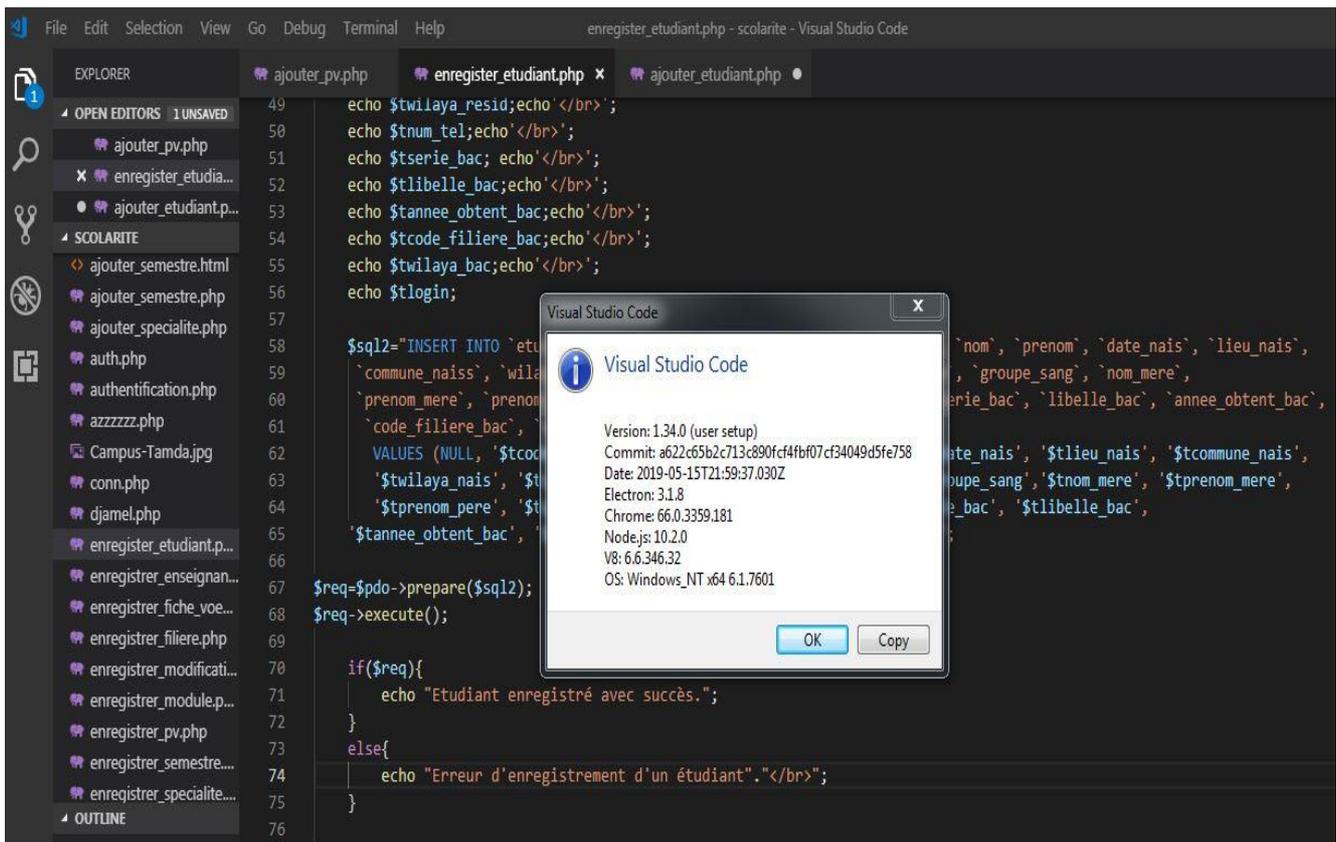


Figure III.1 : Interface Visual Studio Code .

c. Le serveur web Apache :

Tout développement de site web requiert un serveur web qui s'occupe du traitement des requêtes des clients et l'exécution des programmes sur la machine serveur.

Apache (www.apache.org) est le serveur le plus répandu sur Internet. Il s'agit d'une application fonctionnant à la base sur les systèmes d'exploitation de type Unix, mais il a désormais été porté sur de nombreux systèmes, dont Microsoft Windows. Le pack PHPdev (désormais EasyPHP) est ainsi téléchargeable, il regroupe les applications suivantes :

- le serveur web Apache
- le serveur de bases de données MySQL
- le serveur d'application PHP
- l'outil phpMyAdmin permettant de gérer des bases MySQL

La procédure d'installation de EasyPHP est décrite dans la section PHP de CCM.

Apache (prononcez à la française ou bien pour les puristes à l'anglophone « Apatchy ») tire son nom de la façon dont il a été mis au point (« A patchy server » traduisez « un serveur rafistolé ») car il est le fruit d'une multitude de correctifs logiciels afin d'en faire une solution très sûre. En

effet Apache est considéré comme sûr dans la mesure où peu de vulnérabilités le concernant sont publiées.

Ainsi, dès qu'un bug ou une faille de sécurité est décelée, ceux-ci sont rapidement corrigés et une nouvelle version de l'application est éditée.

Apache possède désormais de nombreuses fonctionnalités dont la possibilité de définir une configuration spécifique à chaque fichier ou répertoire partagé, ainsi que de définir des restrictions d'accès grâce aux fichiers htaccess.

d. Le serveur de Bases de Données MySQL :

MySQL (prononcer [maj.es.ky.ɛl]) est un système de gestion de bases de données relationnelles (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde³, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, PostgreSQL et Microsoft SQL Server.

Son nom vient du prénom de la fille du cocréateur Michael Widenius, My. SQL fait référence au *Structured Query Language*, le langage de requête utilisé.

MySQL AB a été acheté le 16 janvier 2008 par Sun Microsystems pour un milliard de dollars américains⁴. En 2009, Sun Microsystems a été acquis par Oracle Corporation, mettant entre les mains d'une même société les deux produits concurrents que sont Oracle Database et MySQL. Ce rachat a été autorisé par la Commission européenne le 21 janvier 2010.

Depuis mai 2009, son créateur Michael Widenius a créé MariaDB (Maria est le prénom de sa deuxième fille) pour continuer son développement en tant que projet Open Source.

e. PhpMyAdmin :

C'est un outil d'administration de bases de données MySQL via un navigateur web, il permet de créer et de gérer l'ensemble des tables de la base de données. Il possède plusieurs fonctions, parmi elles, on cite :

- Création et suppression de bases de données ;
- Création, modification et suppression de tables ;
- Edition, ajout et suppression d'attributs ;
- Exécution de commandes SQL ;
- Création d'index ;
- Gestion des utilisateurs et leurs privilèges.

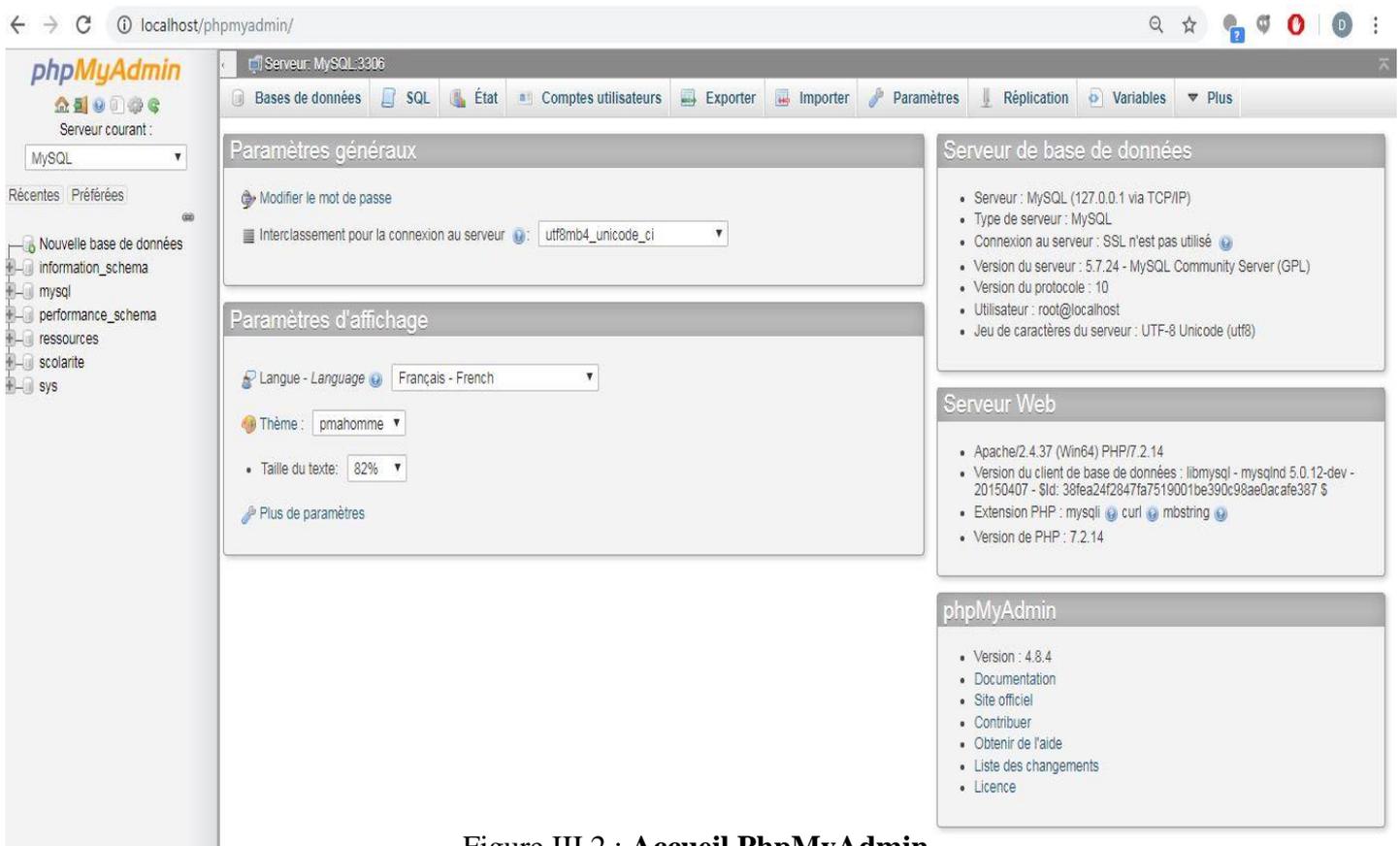


Figure III.2 : Accueil PhpMyAdmin.

f. WAMP Server :[Voir Annexe B]

Il s'agit d'une plateforme de développement web, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WAMP Server n'est pas en soi un logiciel, mais un environnement comprenant deux (02) serveurs ; un serveur web(Apache) et un serveur de bases de données (MySQL), un interpréteur de scripts (PHP), ainsi qu'une administration SQL (phpMyAdmin). Il dispose d'une interface d'administration permettant de gérer les alias (dossiers virtuels disponibles sur Apache), et le démarrage/arrêt des serveurs. Il permet donc d'installer en une seule fois tout le nécessaire au développement local du PHP.

Par défaut, le serveur Apache crée un nom de domaine virtuel (en local) 127.0.0.1 ou localhost. Ainsi, quand on choisit « Web local » dans le menu de WAMP Server, le navigateur s'ouvre sur cette URL et affiche la page index.php de ce site qui correspond en fait au contenu du dossier www de WAMP Server. WAMP Server peut être utilisé comme une application portable, c'est-à-dire lancé sur une clé USB.

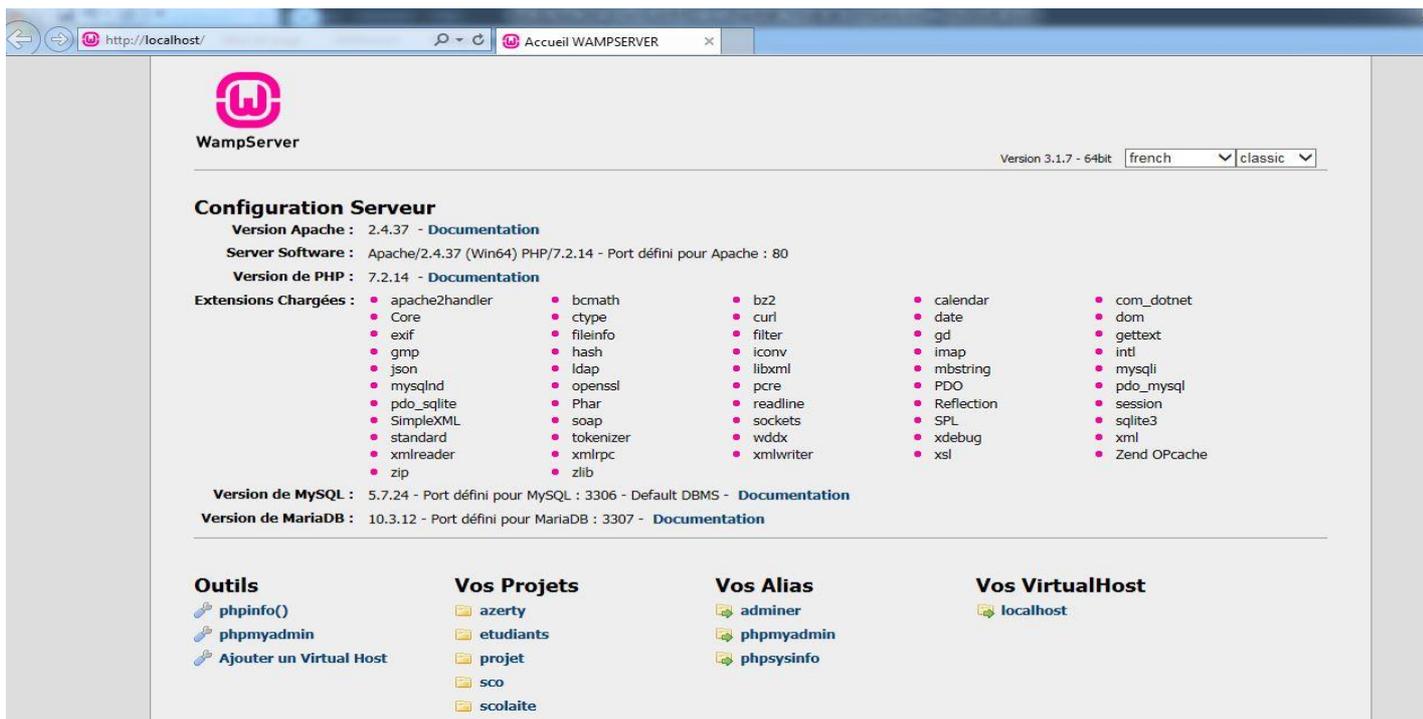


Figure III.3 : Accueil WAMP Server.

g. Les langages de programmation :

g.1. HTML (HyperText Markup Language): [11]

C'est un langage de balisage permettant d'écrire de l'hypertexte, d'où son nom. Il permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie et des éléments programmables tels que des applets. Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec des langages de programmation (JavaScript, PHP, ...etc) et des formats de présentation (Feuilles CSS).

g.2. CSS (Cascading Style Sheets) : [11]

Les **feuilles de style en cascade**, généralement appelées **CSS** de l'anglais *Cascading Style Sheets*, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites web et bien pris en charge par les navigateurs web dans les années 2000.

g.3. PHP (Hypertext PreProcessor) : [12]

PHP: Hypertext Preprocessor, plus connu sous son sigle **PHP** (sigle auto-référentiel), est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté objet.

PHP a permis de créer un grand nombre de sites web célèbres, comme Facebook, Wikipédia, etc. Il est considéré comme une des bases de la création de sites web dits dynamiques mais également des applications web.

g.4. JavaScript :

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs² avec l'utilisation (par exemple) de Node.js. C'est un langage orienté objet à prototype, c'est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas des instances de classes, mais qui sont chacun équipés de constructeurs permettant de créer leurs propriétés, et notamment une propriété de prototypage qui permet d'en créer des objets héritiers personnalisés. En outre, les fonctions sont des objets de première classe. Le langage supporte le paradigme objet, impératif et fonctionnel. JavaScript est le langage possédant le plus large écosystème grâce à son gestionnaire de dépendances npm, avec environ 500 000 paquets en août 2017.

JavaScript a été créé en 1995 par Brendan Eich. Il a été standardisé sous le nom d'ECMAScript en juin 1997 par Ecma International dans le standard ECMA-262. Le standard ECMA-262 en est actuellement à sa 8^e édition. JavaScript n'est depuis qu'une implémentation d'ECMAScript, celle mise en œuvre par la fondation Mozilla. L'implémentation d'ECMAScript par Microsoft (dans Internet Explorer jusqu'à sa version 9) se nomme JScript, tandis que celle d'Adobe Systems se nomme ActionScript.

Avec les technologies HTML et CSS, JavaScript est parfois considéré comme l'une des technologies cœur du World Wide Web. Le langage JavaScript permet des pages web interactives, et à ce titre est une partie essentielle des applications web. Une grande majorité des sites web l'utilisent⁶, et la majorité des navigateurs web disposent d'un moteur JavaScript dédié pour l'interpréter, indépendamment des considérations de sécurité qui peuvent se poser le cas échéant.

g.5. SQL (Structured Query Language) : [13]

C'est un langage informatique normalisé qui sert à demander des opérations sur des bases de données. La partie langage de manipulation de données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données. En plus du langage de manipulation de données, la partie langage de définition de données permet de créer et de modifier l'organisation des données dans la base de données, la partie langage de contrôle de transaction permet de commencer et de terminer des transactions, et la partie langage de contrôle de données permet d'autoriser ou d'interdire l'accès à certaines données à certaines personnes.

III.4. Présentation de quelques interfaces de l'application :

Nous allons dans ce qui suit, présenter quelques interfaces illustratives de notre application.

III.4.1. Interface principale :

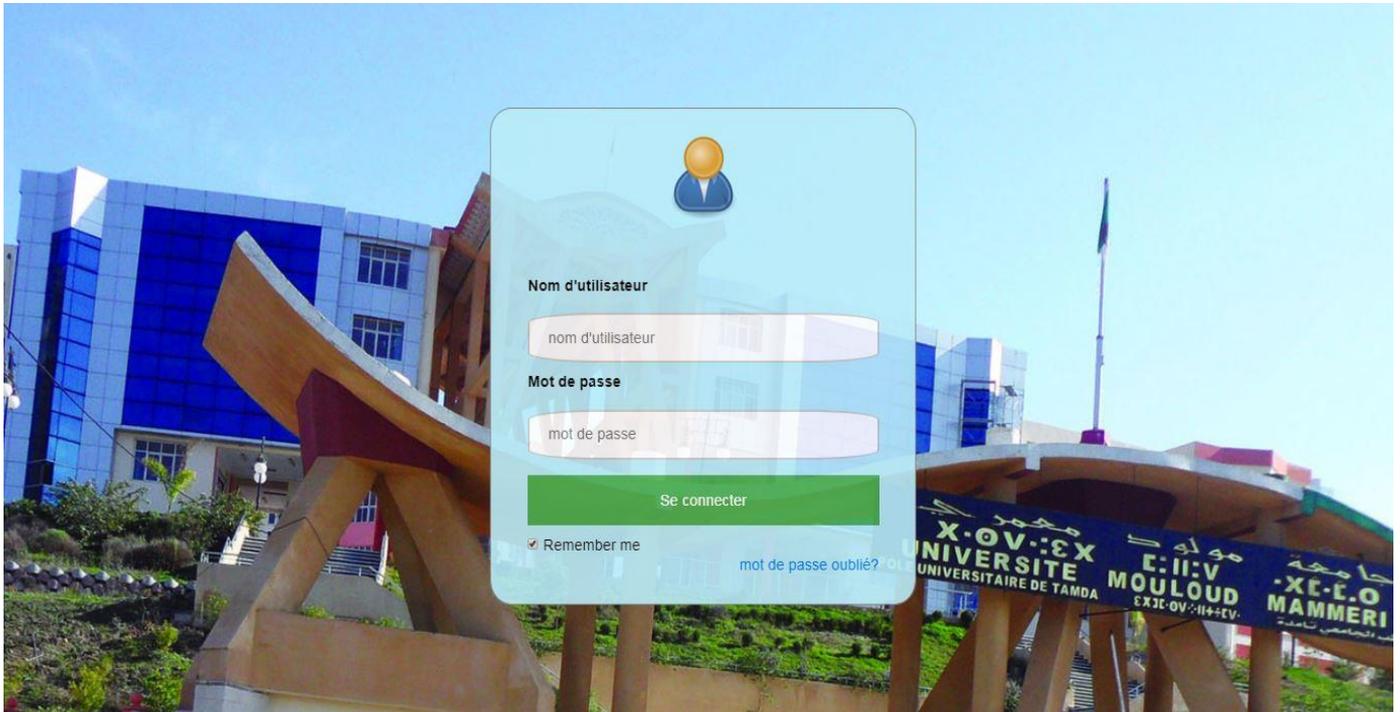


Figure III.4 : Page Principale.

III.4.2. Interface modifier mot de passe :

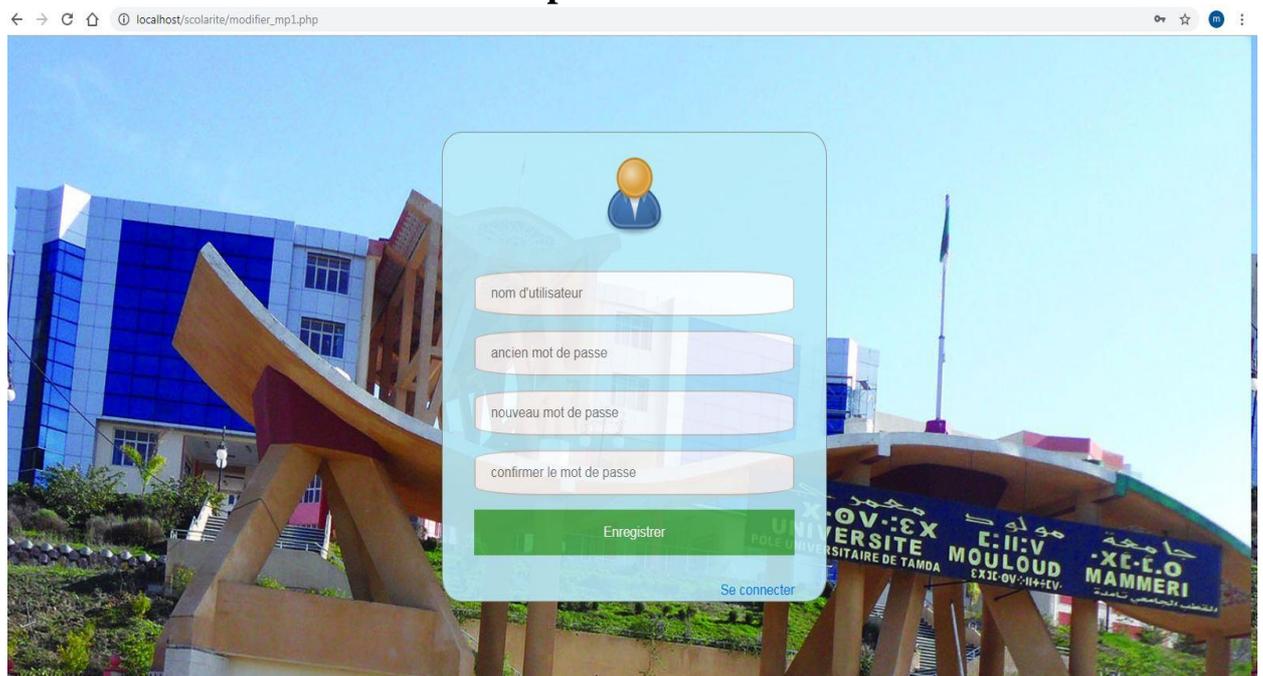


Figure III.5 : Page Modifier mot de passe.

III.4.3. Interface Ajout d'un étudiant :

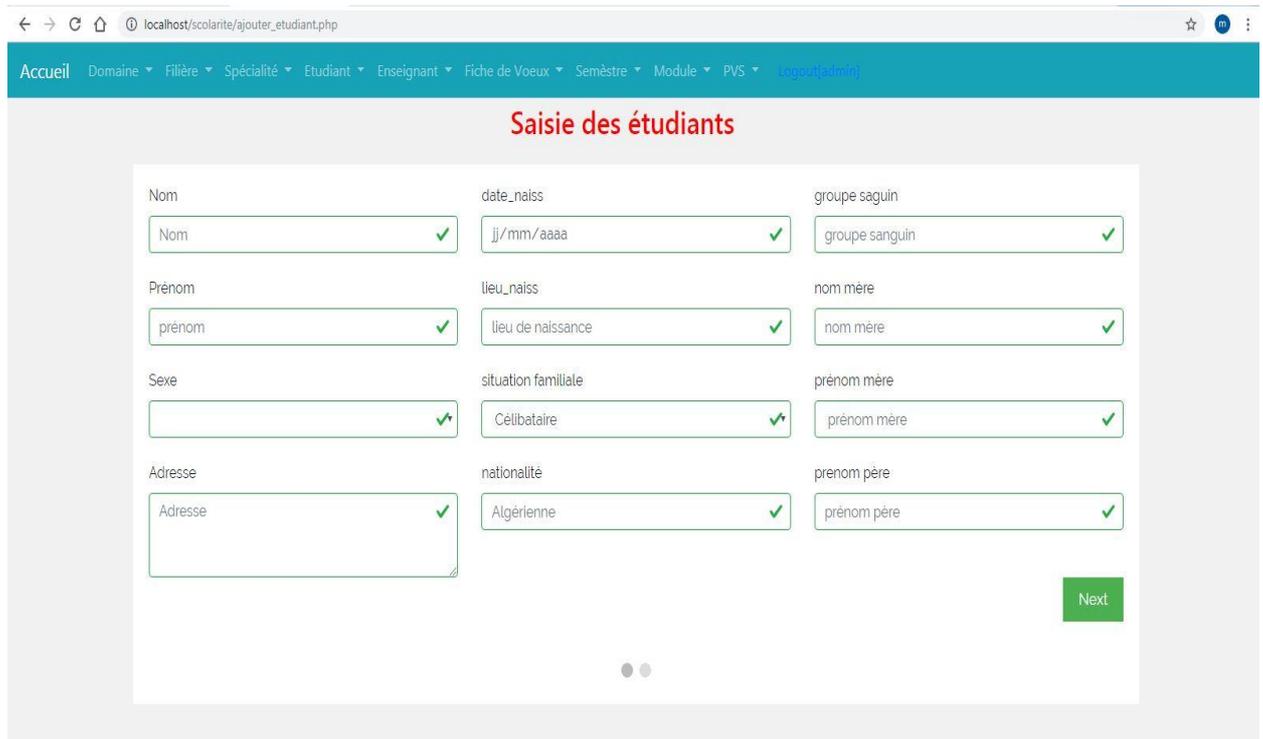


Figure III.6 : Page Ajout d'un étudiant.

III.4.4. Interface affichage des étudiants :

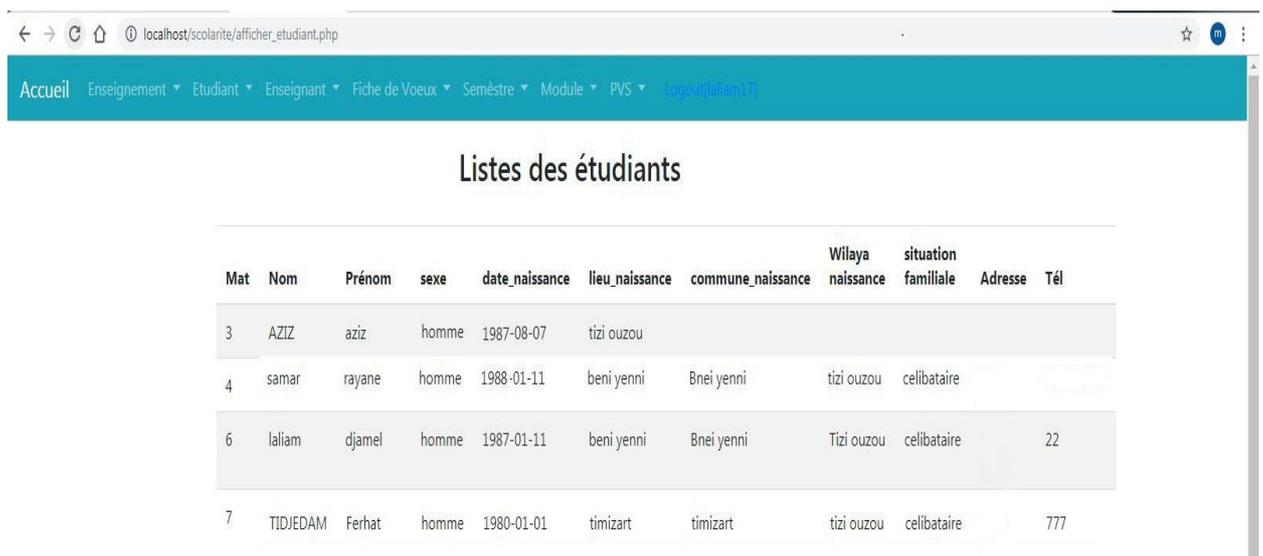


Figure III.7 : Page Affichage de la liste des étudiants.

III.4.5. Interface saisie des enseignements :

The screenshot shows a web browser at localhost/scolarité/ajouter_enseignement.php. The navigation bar includes: Accueil, Enseignement, Etudiant, Enseignant, Fiche de Voeux, Semestre, Module, PVS, and Logout(admin). The main content area has three yellow panels:

- Ajouter Domaine:** Input field for 'Intitulé du domaine' with a green checkmark and an 'Enregistrer' button.
- Ajouter Filière:** Input fields for 'nom du Domaine' (containing 'math info') and 'Intitulé de la filière' (containing 'intitulé'), both with green checkmarks, and an 'Enregistrer' button.
- Ajouter Spécialité:** Input fields for 'Nom filière' (containing 'INformatique') and 'Intitulé spécialité' (containing 'intitulé'), both with green checkmarks, and an 'Enregistrer' button.

Figure III.8 : Page saisie des enseignements.

III.4.6. Interface saisie de la fiche de vœux d'un enseignant :

The screenshot shows a web browser at localhost/scolarité/ajouter_enseignement.php. The navigation bar includes: Accueil, Domaine, Filière, Spécialité, Etudiant, Enseignant, Fiche de Voeux, Semestre, Module, PVS, and Logout(enseignant1). The main content area shows:

- Two tabs: 'Semestre 1' and 'Semestre 2'.
- Text: 'veuillez choisir une spécialité'.
- Input field for 'specialité' with the value 'choisir spécialité' and a green checkmark.
- 'Enregistrer' button.

Figure III.9 : Page saisie de la fiche de vœux d'un enseignant.

III.4.7. Interface Affichage des notes :



nom	prénom	Intitulé module	note	coefficient	crédit	spécialité	unité d'enseignement
AZIZ	aziz	BDD	8	3	6	CPI	UEF1
AZIZ	aziz	architecture	12	3	4	CPI	UEF1
AZIZ	aziz	cpi	14	3	5	CPI	UEM1

Figure III.10 : Page Affichage des notes.

III.6. Conclusion :

Dans ce dernier chapitre de notre mémoire, nous avons présenté les différents outils qu'on a utilisés pour réaliser notre application, ainsi que quelques interfaces illustrant quelques fonctionnalités de cette dernière.

Conclusion Générale :

Toute organisation, quelque soit sa vocation et son caractère, doit se mettre au diapason de la progression technologique et faire face par l'automatisation de ses structures et la formation de son personnel afin d'améliorer son rendement et son service.

L'étude et le développement de ce projet informatique nous ont permis, outre la familiarisation avec le domaine professionnel, d'acquérir de nouvelles compétences sur le plan pratique et technique , à savoir côté application Client/serveur, les outils de développement tels que Visual Studio Code et l'utilisation du serveur Web Apache, côté bases de données avec à la clé l'usage du SGBD MySQL, côté conception orientée objet faite à l'aide du langage UML et côté réalisation avec l'introduction des langages de programmation tels que XHTML, CSS, PHP, JavaScript et SQL.

L'application à laquelle nous avons abouti, traite la gestion de scolarité qui adopte l'architecture Client/Serveur. Elle offre aux différents utilisateurs plus de souplesse et d'efficacité dans l'exécution de leurs tâches.

Quoique notre application offre plusieurs services et rend aisé la tâche de gestion de scolarité, quelques améliorations et extensions peuvent être apportées. En guise de perspectives, les possibilités d'évolutions de ce projet sont nombreuses. Pour clore ce mémoire, nous espérons avoir répondu aux attentes et aux exigences de l'université, et que l'application informatique réalisée sera d'une grande utilité pour les départements des sciences et technologies du Campus de Tamda de l'Université Mouloud Mammeri de Tizi-Ouzou.

ANNEXE A

« *UML* »

Introduction

La notation UML est une fusion des notations d'OOD, OMT et OOSE. Les concepteurs de cette notation ont recherché avant tout la simplicité. Les symboles embrouillés, redondants ou superflus ont été éliminés en faveur d'un meilleur rendu visuel.

UML est un langage de modélisation fondé sur les concepts objet standard conçu pour l'écriture de plans d'élaboration de logiciels. L'objectif d'UML est de fournir une notation standard utilisable dans le développement de systèmes informatiques basés sur l'objet.

UML est adapté à la modélisation de systèmes, depuis les systèmes informatiques d'entreprise jusqu'aux applications distribuées basées sur le Web, c'est un langage très expressif qui couvre toutes les perspectives nécessaires au développement et au déploiement de tels systèmes. Pour apprendre à s'en servir efficacement, il faut d'abord s'appuyer sur une représentation conceptuelle de ce langage, ce qui nécessite l'assimilation de trois éléments fondamentaux qui sont :

Les briques de base d'UML, (Des éléments. Des relations. Des diagrammes) et les règles qui déterminent la manière de les assembler et quelques mécanismes généraux qui s'appliquent à ce langage.

UML s'articule autour de neuf diagrammes différents, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Par ailleurs, UML modélise le système suivant deux modes de représentation : l'un concerne la structure du système pris "au repos" **modèle statiques** (diagramme de cas d'utilisation, de classe...etc.) l'autre concerne sa **dynamique** de fonctionnement (diagramme de séquence, d'états transitions, d'activités...etc.).

Les deux représentations sont nécessaires et complémentaires pour schématiser la façon dont est composé le système et comment ses composantes fonctionnent entre elles.

1. Modèle conceptuel d'UML

Le modèle conceptuel d'UML comprend les notions de bases génériques du langage. Il définit trois sortes de briques de base :

- Les éléments, qui sont les abstractions essentielles à un modèle.
- Les relations, qui constituent des liens entre ces éléments.
- Les diagrammes, qui regroupent des éléments et des liens au sein de divers ensembles.

2. Quelques définitions [14]

2.1 Les interactions

Une interaction est un comportement qui comprend un ensemble de messages échangés au sein d'un groupe d'éléments, dans un contexte particulier, pour atteindre un but bien défini. Le comportement d'un ensemble d'objets ou celui d'une opération individuelle peut être précisé par une interaction..Comme le montre la **figure suivante** un message est représenté par une ligne fléchée, qui indique le nom de son opération.

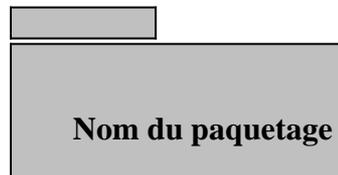
Atteint



2.2 Les éléments de regroupement

Les éléments de regroupements représentent les parties organisationnelles des modèles UML. Ce sont des boîtes dans lesquelles un modèle peut être décomposé. Il existe un seul type fondamental d'éléments de regroupement : le « paquetage ».

Un **paquetage** est un ensemble d'éléments de modélisation : des classes, des associations, des objets, des composants....



2.3 Les éléments d'annotation

Les éléments d'annotation représentent les parties explicatives des modèles UML. Ce sont les commentaires qui peuvent accompagner tout élément dans un modèle, à des fins de description, d'exploitation et de remarque. Il existe un type fondamental d'éléments d'annotation appelé « note » qui est simplement un symbole utilisé pour représenter les contraintes et les commentaires rattachés à un élément ou un ensemble d'éléments. Comme le montre la **figure suivante** une note est représentée par un rectangle écorné qui contient un commentaire textuel ou graphique.

Voici une note



Dans l'étude conceptuelle de notre plate-forme, nous n'avons utilisé que quatre diagrammes, à savoir le diagramme de cas d'utilisation, le diagramme de classe, le diagramme de séquence et le diagramme d'activités.

Nous allons décrire ces diagrammes ainsi que les éléments et les relations les constituant dans ce qui suit.

3. Les Diagrammes d'UML

Un diagramme est la représentation graphique d'un ensemble d'éléments qui constituent un système. Il se présente sous la forme d'un graphe connexe où les sommets correspondent aux éléments et les arcs aux relations. Les diagrammes servent à visualiser un système sous différents perspectives, ce sont des " vues " qui permettent d'isoler certaines parties des modèles pour les rendre plus aisément compréhensibles.

3.1 Description des cas d'utilisation [15]

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

a. L'acteur

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

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

Il existe 4 catégories d'acteurs :

1. **Les acteurs principaux** : les personnes qui utilisent les fonctions principales du système
2. **Les acteurs secondaires** : les personnes qui effectuent des tâches administratives ou de maintenance.
3. **Le matériel externe** : les dispositifs matériels incontournables qui font partie du domaine de l'application et qui doivent être utilisés.
4. **Les autres systèmes** : les systèmes avec lesquels le système doit interagir.

b. Le 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 use cases décrit les objectifs (le but) du système.

c. La relation

Elle exprime l'interaction existant entre un acteur et un cas d'utilisation.

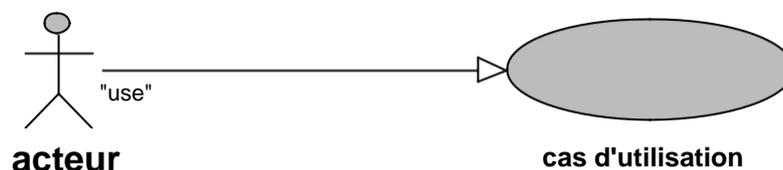
Il existe 3 types de relations entre cas d'utilisation :

1. la relation d'utilisation
2. la relation d'extension
3. la relation d'inclusion

c.1. La relation généralisation

Dans une relation de généralisation entre 2 cas d'utilisation, le cas d'utilisation enfant est une spécialisation du cas d'utilisation parent.

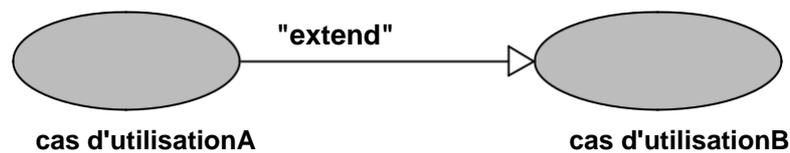
Formalisme :



c.2. La relation d'extension

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

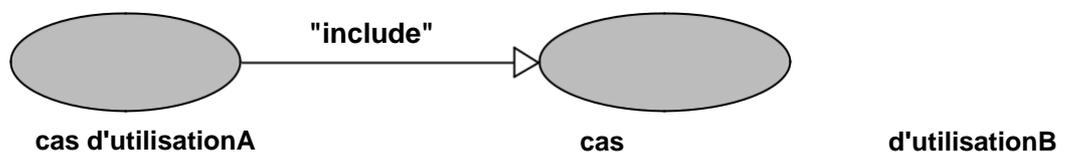
Formalisme :



c.3. La relation d'inclusion

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

Formalisme :



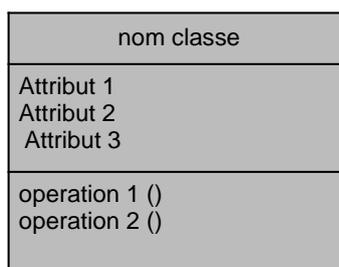
3.2 Le diagramme de classes:

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

3.2.1 Eléments de Base

3.2.1.1. Classe : C'est une description d'un ensemble d'objets qui partage les mêmes attributs, opérations, méthodes, relations et contraintes. Une instance d'une classe est appelée objet. Elle est représentée par un rectangle, qui, en règle générale, contient son nom, ses attributs et ses opérations.

Formalisme :



3.2.1.2 les relations

Il existe plusieurs types de relations entre classes :

- l'association
- la généralisation/spécialisation
- la dépendance

a. L'association

Une association est une relation structurelle bidirectionnelle qui décrit un ensemble de liens entre différents éléments (la nature des liens est ajoutée comme décoration).

Elle est représentée par une ligne qui peut être fléchée ; elle comprend parfois une étiquette et souvent d'autres décorations, comme la multiplicité et les noms de rôles.

b. La généralisation / spécialisation

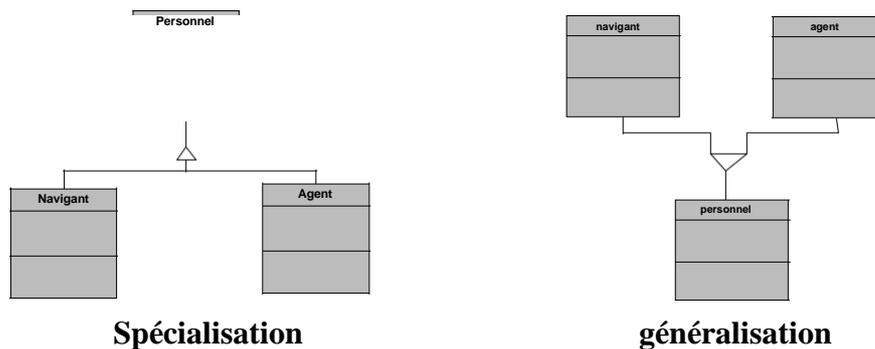
Le principe de généralisation / spécialisation permet d'identifier parmi les objets d'une classe (générique) des sous-ensembles d'objets (des classes spécialisées) ayant des définitions spécifiques. La classe plus spécifique est cohérente avec la classe plus générale c'est-à-dire qu'elle contient par **héritage** tous les attributs, les membres, les relations de la classe générale, et peut contenir d'autres.

Les relations de généralisation peuvent être découvertes de 2 manières :

- **la généralisation** : il s'agit de prendre des classes existantes déjà mises en évidence) et de créer de nouvelles classes qui regroupent leurs parties communes ; il faut aller du plus spécifique au plus général.

- **La spécialisation** : il s'agit de sélectionner des classes existantes (déjà identifiées) et d'en dériver des nouvelles classes plus spécialisées, en spécifiant simplement les différences.

Formalisme :



c. Relation de dépendance :

Relation entre éléments du modèle ne nécessitant pas forcément un lien entre objets.

Lorsque cette relation est réalisée par des liens entre objets, ces derniers sont limités dans le temps, contrairement à d'autres relations plus structurelles.

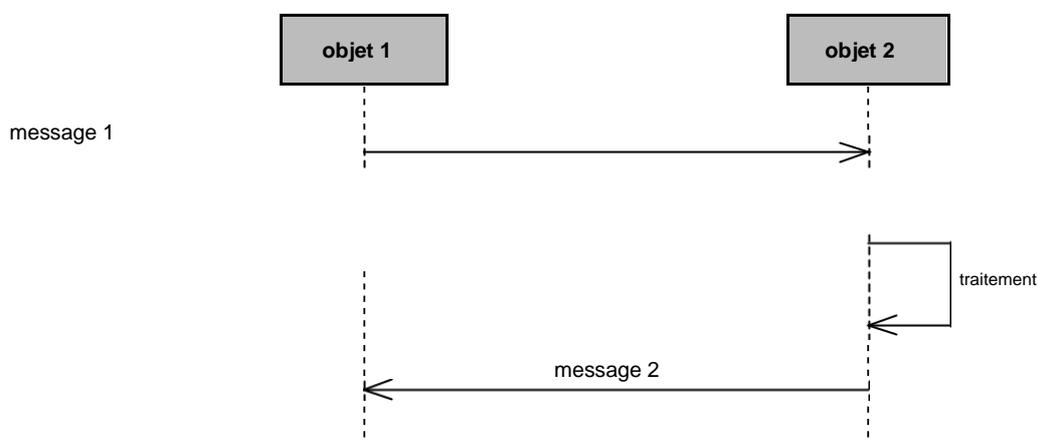
Un élément *A* *dépend* d'un élément *B*, lorsque *A* utilise des services de *B*. De ce fait, tout changement dans *B* peut avoir des répercussions sur *A*.

3.3. Diagramme de séquence [14]

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

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

Formalisme :

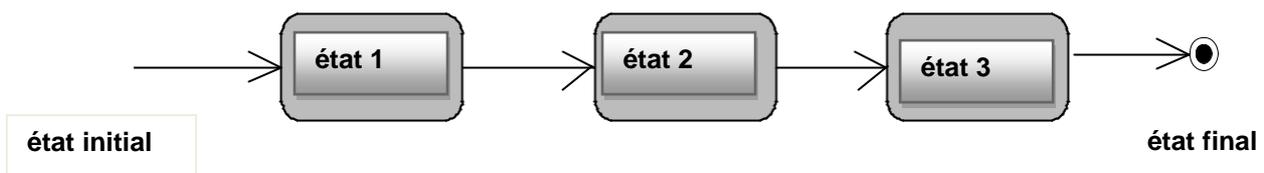


3.4. Diagrammes d'activités

Le diagramme d'activité est attaché à une catégorie de classe il décrit le déroulement des activités de cette catégorie et il décrit le comportement d'une opération en terme d'actions.

Le déroulement s'appelle "**flot de contrôle**". Il indique la part prise par chaque objet dans l'exécution d'un travail. Il sera enrichi par les conditions de séquençement.

Formalisme :



3.5. Les autres diagrammes UML :

- **Le diagramme de collaboration** qui est une représentation spatiale des objets, des liens et des interactions.
- **Le diagramme de composants** qui décrit les composants physiques d'une application.
- **Le diagramme de déploiement** qui décrit les composants sur les dispositifs matériels.
- **Le diagramme d'états transitions** qui décrit le comportement d'une classe en terme d'états.
- **Diagrammes d'objets** permet de mettre en évidence des liens entre les objets. Les objets, instances de classes, sont reliés par des liens, instances d'associations.

4. Extension d'UML pour le web [14]

L'extension d'UML pour le Web définit un ensemble de stéréotype, d'étiquettes et de contraintes, qui rend possible la modélisation Web.

4.1. Classe

4.1.1 Page serveur « Server page »

Icône



Description : Représente une page Web possédant des scripts qui interagissent avec des ressources serveur telles que les bases de donnée, ces scripts sont exécuté par le serveur.

Contraintes : Les pages serveur ne peuvent avoir de relation qu'avec des objets sur le serveur.

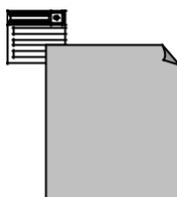
Étiquettes : Moteur de script qui peut être un langage ou le moteur qui doit être utilisé pour exécuter ou interpréter cette page.

4.1.2. Page client « client page »

Icône

Page

Client



Description : Une instance d'une page client est une page Web formatée en HTML. Les pages clients peuvent contenir des scripts interprétés par les navigateurs lorsque celles-ci sont restituées par ces derniers. Les fonctions des pages clients correspondent aux fonctions des scripts de la page web.

Contrainte : Aucune.

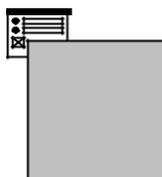
Étiquettes :

- _ Titre (Title) : Titre de la page tel qu'il est affiché par le navigateur.
- _ Base (Base) : URL de base pour déréférencer les URL relatives.
- _ Corps (Body) : ensemble des attributs de la balise <body>, qui définit des caractéristiques par défaut du texte et de l'arrière-plan.

4.1.3. Formulaire

Icône

Form



Description : Une classe stéréotypée « form » est un ensemble de champs de saisie faisant partie d'une page client. A une classe formulaire correspond une balise HTML <form>, les attributs de cette classe correspondent aux éléments de saisie d'un formulaire HTML (zone de saisie, zone de texte, boutons d'option.).

Un formulaire n'a pas d'opérations, puisqu'il peut les encapsuler. Toute opération qui interagit avec le formulaire appartient à la page qui la contient.

Contraintes : Aucune.

Étiquettes : Méthodes utilisées pour soumettre les données à l'URL de l'attribut

4.2. Association

4.2.1. Lien « link »

Icône : Aucune

Description : Un lien est un pointeur d'une page client vers une autre page. Dans un diagramme de classe, un lien est une association entre une page client et une autre page client ou une page serveur.

Contraintes : Aucune.

Étiquettes : Paramètres (paramètres) : liste de noms de paramètres qui doivent être passés avec la demande de la page liée.

4.2.2. Soumet « submit »

Icône : Aucune.

Description : submit est une association qui se trouve toujours entre un formulaire et une page serveur. Les formulaires soumettent les valeurs de leurs champs au serveur, par l'intermédiaire de pages serveur, pour qu'il les traite. Le serveur Web traite la page serveur, qui accepte et utilise les informations du formulaire.

Contraintes : Aucune.

Étiquettes : Paramètres (parameters) : une liste de noms de paramètres qui doivent être passés avec la demande de la page liée.

4.2.3. Construit « build »

Icône : Aucune.

Description : La relation « build » est une relation particulière qui fait le pont entre les pages client et les pages serveur. L'association « build » identifie quelle page serveur est responsable de la création d'une page client. C'est une relation orientée, puisque la page client ne connaît pas la page qui est à l'origine de son existence.

Une page serveur peut construire plusieurs pages client, en revanche, une page client ne peut être construite que par une seule page serveur.

Contraintes : Aucune.

4.2.4. Redirige « redirect »

Icône : Aucune

Description : Une relation « redirect », est une association unidirectionnelle avec une page web, peut être dirigée à partir d'une page client ou serveur vers une autre page client ou serveur.

Contraintes : Aucune.

Étiquettes : Délai (delay) : délai que doit observer une page client avant de rediriger vers la page destination.

5. Les points forts et points faible d'UML [16]

5.1. Les points forts

a. UML est un langage formel et normalisé, gain de précision, gage de stabilité encourage l'utilisation d'outils

b. UML est un support de communication performant, il cadre l'analyse, il facilite la compréhension de représentations abstraites complexes, son caractère polyvalent et sa souplesse font un langage universel.

5.2. Les points faibles

La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation.

La nécessité de s'accorder sur des modes d'expression communs est vitale en informatique. UML n'est pas à l'origine des concepts objets, mais en constitue une étape majeure, car il unifie les différentes approches et en donne une définition plus formelle.

Le processus (non couvert par UML) est une autre clé de la réussite d'un projet. Or, l'intégration d'UML dans un processus n'est pas triviale et améliorer un processus est une tâche complexe et longue

Les auteurs d'UML sont tout à fait conscients de l'importance du processus, mais l'acceptabilité industrielle de la modélisation objet passe d'abord par la disponibilité d'un langage d'analyse objet performant et standard.

Conclusion

Cette annexe a proposé une présentation brève des principaux concepts de modélisation d'UML et son extension pour le web ainsi que leur notation.

ANNEXE B

« *WAMP* »

Description

Wamp est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données.

Procédure d'Installation

Téléchargement

Il est possible de télécharger Wamp en passant par le site WAMPServer.com dans le menu de navigation du site cliquez sur **TÉLÉCHARGER**



La page devrait alors défiler vous amenant à une section téléchargement où vous aurez le choix entre 2 versions de WampServer, WAMP SERVER 64 Bits et WAMP SERVER 32 bits



Pour rappel, pour connaître le type de système de votre Windows, rendez-vous dans panneau de configuration > Système et sécurité > système

Système	
Évaluation :	5,8 Indice de performance Windows
Processeur :	Intel(R) Core(TM) i7-2640M CPU @ 2.80GHz 2.80 GHz
Mémoire installée (RAM) :	8,00 Go (7,88 Go utilisable)
Type du système :	Système d'exploitation 64 bits
Stylect et fonction tactile :	La fonctionnalité de saisie tactile ou avec un stylect n'est pas disponible sur cet écran

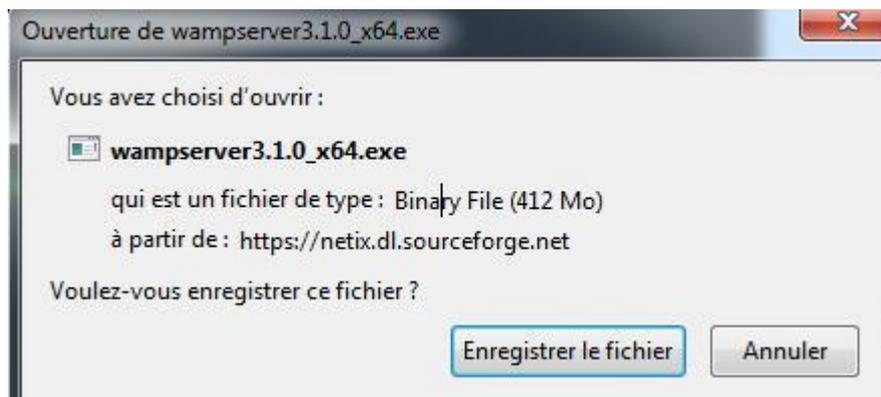
Votre type de système identifié, cliquez sur le bouton de téléchargement de la version de Wamp qui lui correspond, apparaît alors un questionnaire et quelques recommandations techniques. Il est à noter que vous pouvez choisir de ne pas remplir le questionnaire en cliquant sur « passer au téléchargement direct »

WampServer est disponible gratuitement (sous licence
actualités formation d'Alter Way, société éditrice, ainsi que
pas, vous pouvez **passer au téléchargement direct.**

Vous arrivez alors sur un autre site, vous présentant plusieurs choix, afin de lancer le téléchargement cliquer sur le bouton vert Download Latest Version.



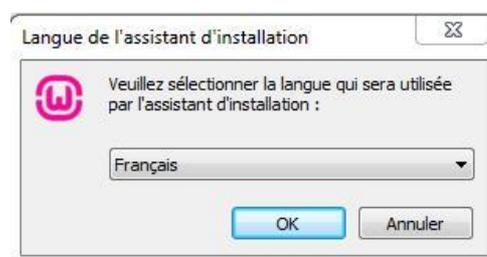
Après quelques secondes une fenêtre s'ouvrira, vous proposant d'enregistrer le programme WampServer sur votre ordinateur



Celle-ci téléchargé, exécuter l'application pour effectuer l'installation.

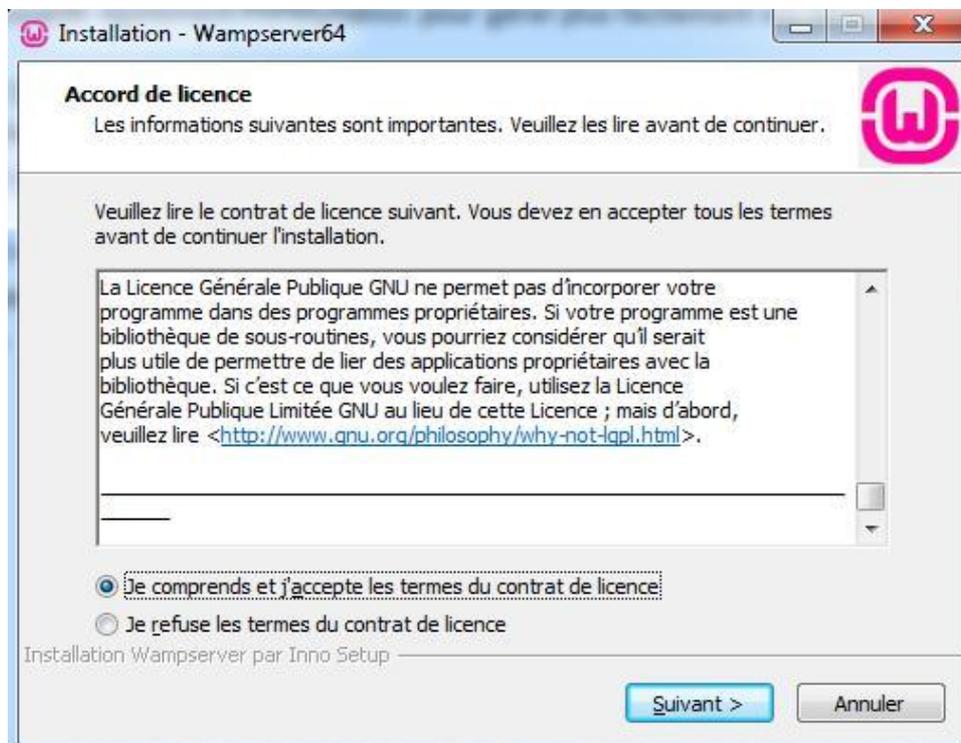
Installation

L'application lancée, une fenêtre s'affiche vous demandant de choisir la langue de votre installation.

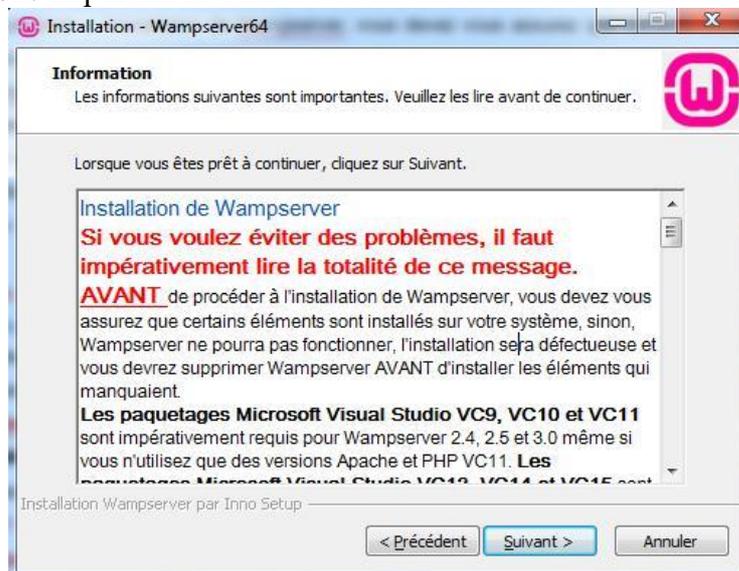


Valider en appuyant sur Ok, s'en suis la demande d'accord de licence, après l'avoir lu, cocher alors

« Je comprends et j'accepte les termes du contrat de licence » avant d'appuyer sur suivant.

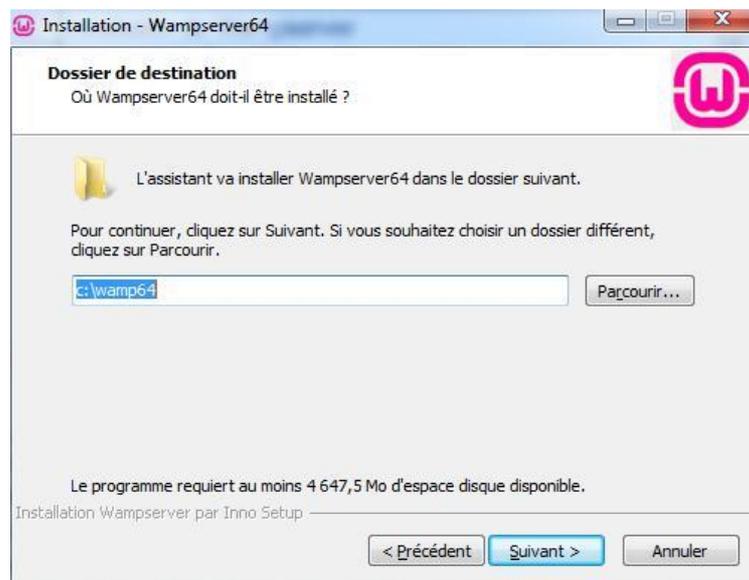


L'interface qui suit fournis des instructions essentielles pour le bon fonctionnement du logiciel, il est donc important de bien suivre les informations affichées à l'écran.

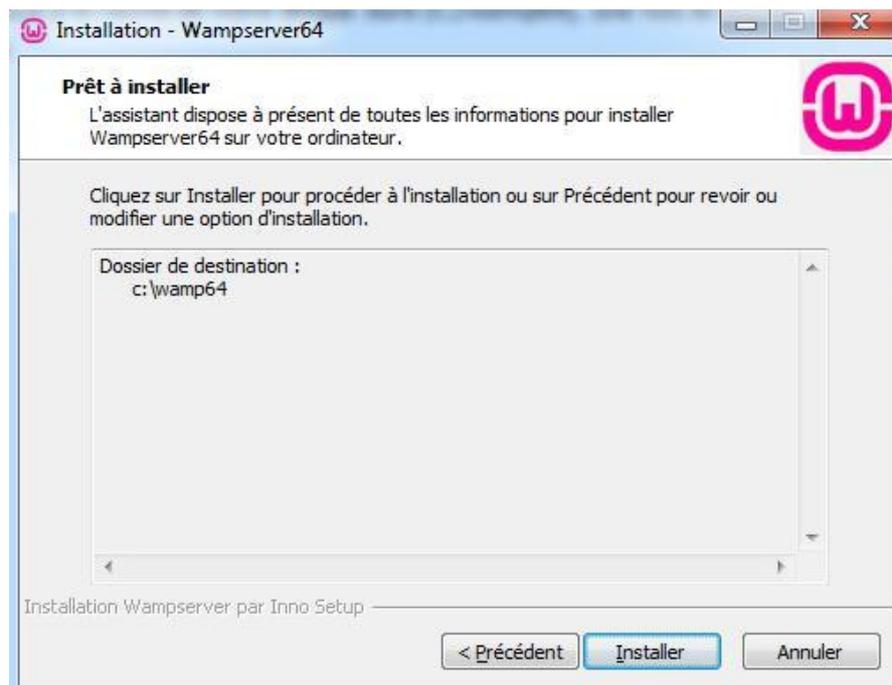


Ces instructions expliquent notamment qu'il est nécessaire d'installer **Les paquetages Microsoft Visual Studio VC13, VC14 et VC15** qui sont indispensables pour le fonctionnement de Apache 2.4.27 et PHP 7 ou 7.1. Dans ces instructions sont cités les liens permettant de télécharger les paquetages ainsi que quelques consignes supplémentaires en cas de problème lors du téléchargement et l'installation de ces dépendances. Une fois prêt à continuer, cliquez sur « suivant ».

La page qui suit vous propose de choisir le dossier de destination de WampServer, par défaut celui-ci sera placé à la racine de votre disque dur (C:/wamp64), une fois le dossier choisi, cliquez sur suivant.



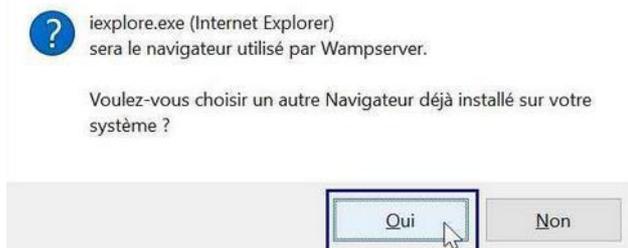
La page suivante récapitule votre choix de répertoire et vous invite à lancer l'installation en appuyant sur « Installer ».



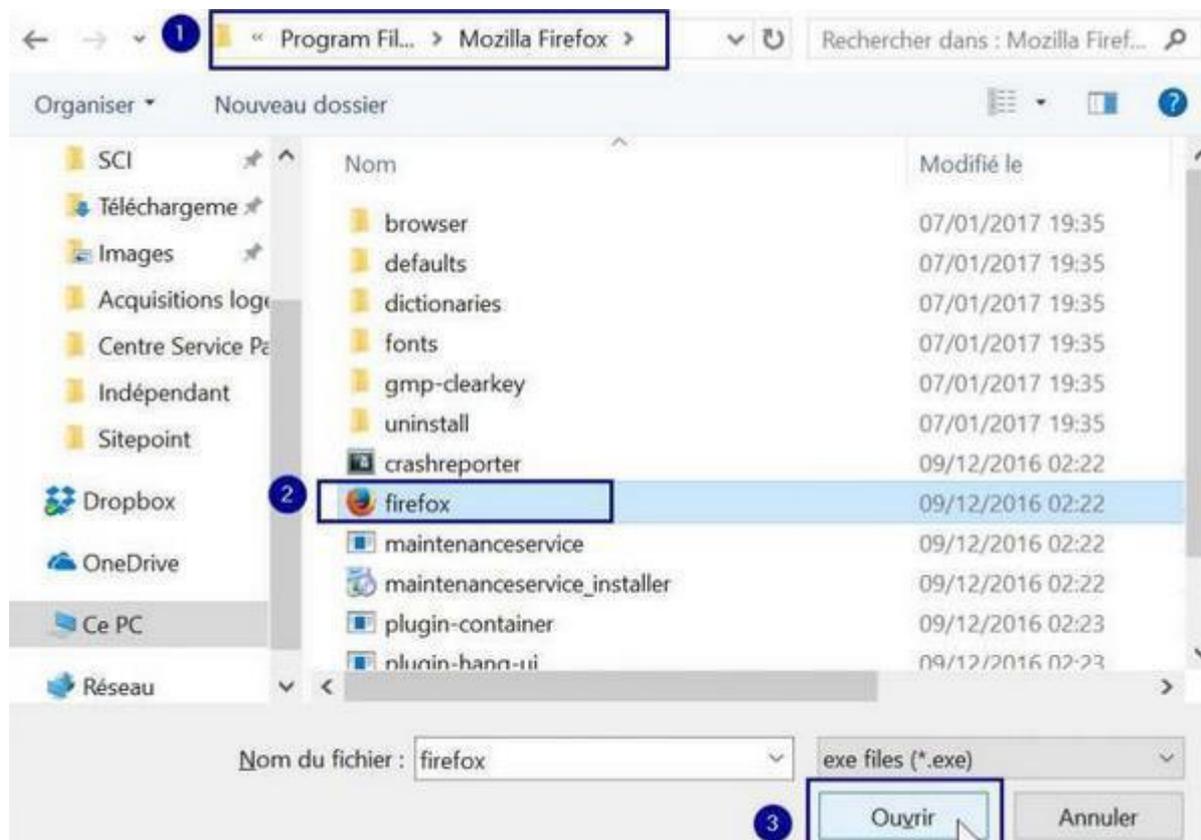
Une barre de progression de l'installation apparaît



Vers la fin de l'installation il vous sera demandé de choisir un navigateur par défaut pour utiliser WampServer. Voici comment choisir Firefox plutôt qu'Internet Explorer :

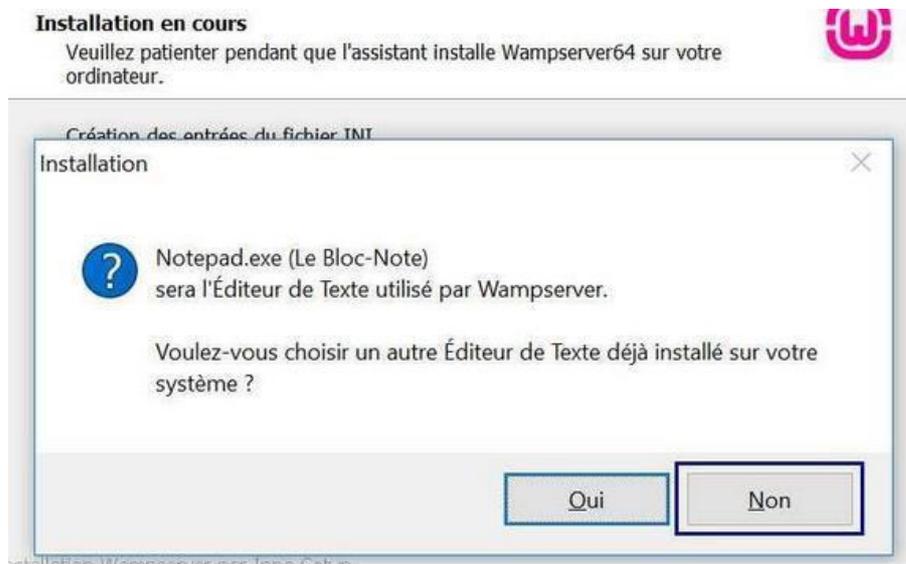


Choisir le répertoire contenant les fichiers programmes de « Mozilla Firefox », sélectionner l'exécutable de Firefox puis cliquer sur le bouton « Ouvrir »



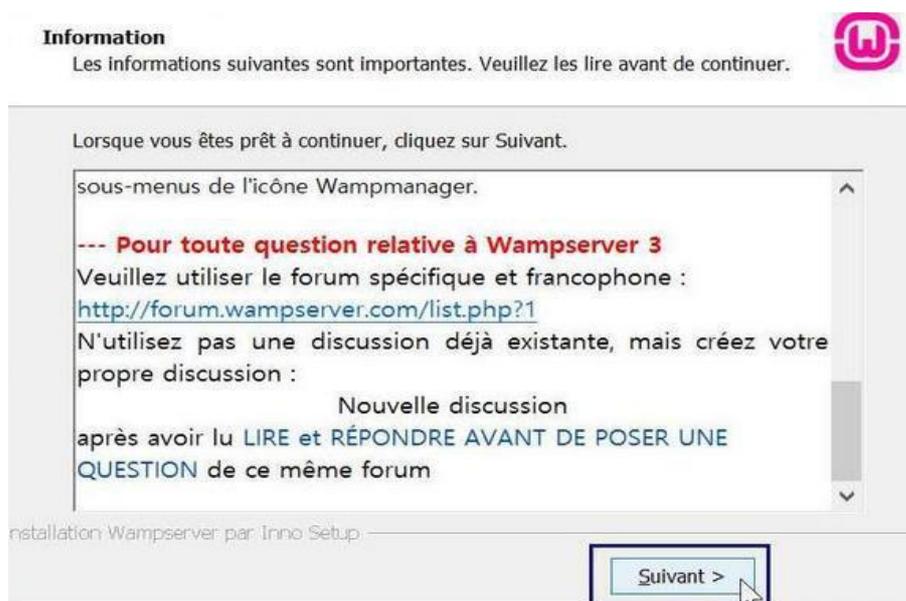
L'installateur propose ensuite de choisir un éditeur à utiliser par WampServer pour éditer des fichiers comme les journaux (logs).

Conservons l'éditeur de texte standard de Windows (si vous souhaitez utiliser un autre éditeur de texte cliquer sur « Oui » puis la procédure est similaire à celle vue précédemment pour choisir le navigateur.



Une fois l'installation achevée, l'installateur affiche des recommandations et informations pour l'utilisation de WampServer :

- L'utilisateur utilisé pour accéder à l'administration de WampServer est « root » et il n'a pas de mot de passe associé.
- Ne pas utiliser les touches de clavier pour naviguer dans les menus de WampServer.
- Un forum d'aide est disponible à l'adresse <http://forum.wampserver.com/list.php?1>



L'installation est terminée, cliquez sur le bouton Terminer.

Présentation de WampServer

Lancer WampServer

Il existe plusieurs possibilités pour lancer WampServer :

- En cliquant sur l'icône présent sur votre bureau :



- En lançant depuis le menu Démarrer :

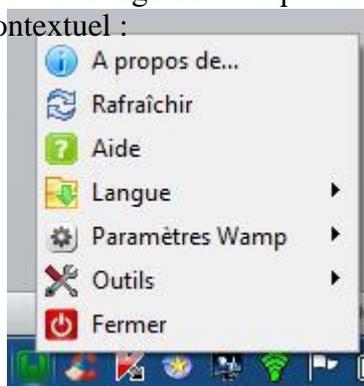


L'application se lance en arrière-plan, un icône de Wamp s'affiche dans la barre des tâches, indiquant l'état du serveur à travers sa couleur :

-  Rouge indique qu'il y a un problème.
-  Orange indique que le serveur n'est pas fonctionnel (exemple : en cours de démarrage).
-  Vert, le serveur est en route et fonctionnel.

Les interfaces

Un clic droit sur le logo de Wamp dans la barre des tâches vous permettra d'afficher un menu contextuel :

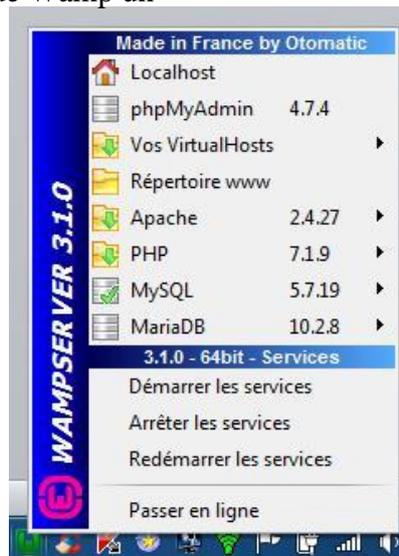


Voici un descriptif des différentes options :

- *A propos de...* donne des informations sur la version du serveur et de ses composants.
- *Rafraîchir* permet de redémarrer le serveur, permettant notamment d'actualiser des modifications.
- *Aide* vous ouvrira dans votre navigateur la page du forum d'aide de Wamp.
- *Langue* vous permettra de choisir la langue de WampServer.
- *Paramètre Wamp* vous propose des choix d'option de fonctionnement de WampServer.
- *Outils* permet de lancer des utilitaires de gestion de WampServer.
- *Fermer* permet quant à lui de fermer l'application WampServer.

En effectuant un clic gauche dans la barre des tâches sur le logo de Wamp un second menu contextuel apparaît, proposant d'autres options :

- *Localhost* ouvre dans votre navigateur l'adresse de WampServer.
- *phpMyAdmin* vous fait accéder à votre base de données.
- *Vos VirtualHosts* affiche la liste des hôtes virtuels de votre serveur.
- *Répertoire www* vous permet d'accéder au répertoire de stockage de vos projets.
- *Apache* propose des outils de configuration pour ce composant.
- *PHP* vous propose des outils de configuration pour PHP comme la possibilité de changer sa version en cour.
- *MySQL* vous permet de configurer ce composant.
- *MariaDB* propose des configurations pour ce composant.
- *Démarrer les services* permet de lancer les composants éteints.
- *Arrêter les services* stopper les composants en cour d'exécution sans arrêter WampServer.
- *Redémarrer les services* permet de relancer les services pour appliquer des modifications notamment.



La page d'accueil de WampServer se trouve à la page <http://localhost> ou <http://127.0.0.1>

Elle apporte de nombreuses informations sur votre serveur :

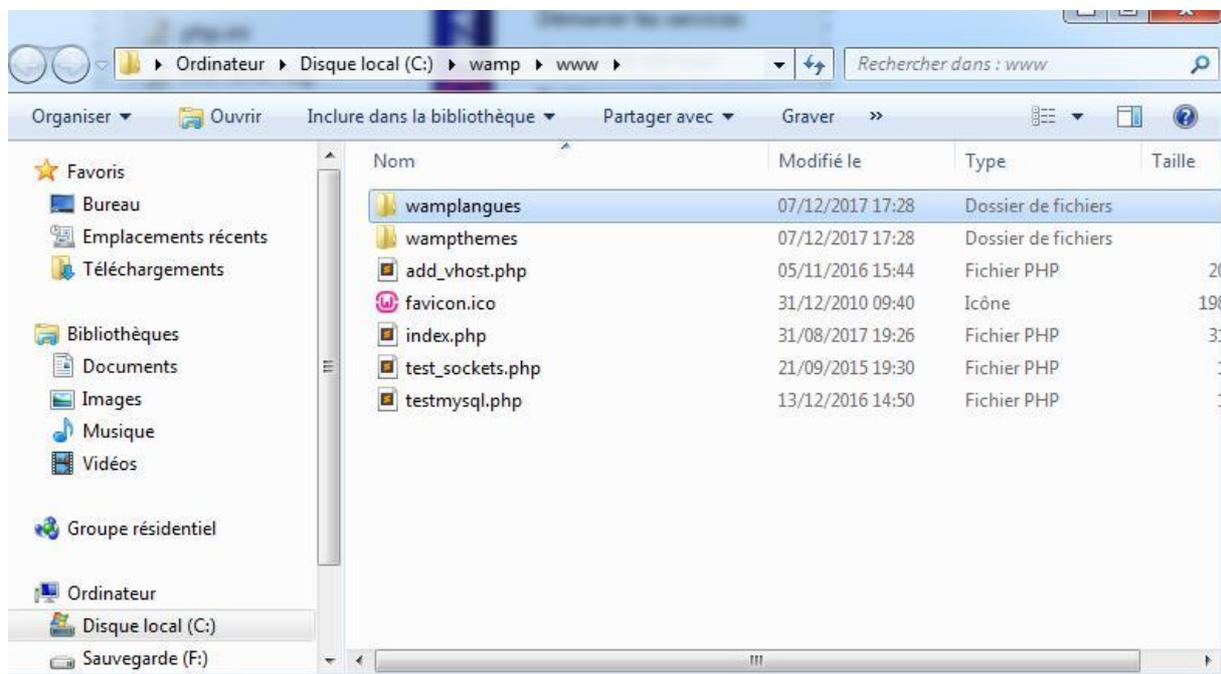
- La version Apache.
- La version PHP.
- La liste des extensions chargées.
- La version de MySQL, ainsi que le port défini.
- La version de MariaDB (une autre base de données), ainsi que le port défini.

Plus bas quelques liens important sont fournis :

- Une liste d'outil permettant de gérer votre serveur, notamment le lien vers PHPMyadmin.
- Vos projets, les fichiers situés dans le répertoire www de Wamp.
- Vos alias, d'autres outils intéressant.
- Vos VirtualHost, par défaut il n'y a que localhost, mais il est possible d'en ajouter grâce à l'outil Ajouter un Virtual Host.

Démarrer avec WampServer

Pour créer une application il vous faudra vous rendre dans le répertoire www situé dans le répertoire de Wamp, pour y accéder, clic gauche sur WampServer dans la barre des tâches > « Répertoire www » explorer s'ouvre alors dans le répertoire souhaité.



« *REFERENCES*

BIBLIOGRAPHIQUES »

Références

- [1]: Bertrand Petit, Architecture des réseaux.
- [2]: Jean, François, Dillon, Introduction au réseau local, Edition Dunod
- [3]: George et Olivier GARDARIN, "**Le Client-Serveur**".Eyrolles, 2000
- [4] :[http ://www.wikipedia.org](http://www.wikipedia.org).
- [5]: Introduction au World Wide Web et à XHTML, Pierre Senellart, 2007
- [6]: [http ://www.Algerie-dz.com](http://www.Algerie-dz.com).
- [7]: UML 2 pour l'analyse d'un système d'information, 4^{eme} édition, Dunod
- [8]: UML 2 Analyse et conception, J.Gabay & D.Gabay, 2008.
- [10]: L'orienté objet, 3eme édition, EYROLLES
- [11]: www.siteduzero.com
- [12]: www.php.net
- [13]: Mise en œuvre des bases de données Edition Eyrolles.
- [14]: Etude de cas UML de Anne-Laure VALDIVIA, David ROUSSE, Philippe CASSAN
- [15]: COURS UML livre de Frédéric Di Gallo/ Laurent Piechocki.
- [16]: UML, le langage de modélisation objet unifié Par Laurent Piechocki.

