

MINISTERE DEL'ENSEIGNEMENTS SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET DEL'INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE

Mémoire de fin d'études

*En vue de l'obtention du diplôme de Master en Electronique,
Option : Télécommunication et Réseaux*

Thème

*Conception et réalisation d'une application Web pour
la gestion du département d'Electronique de
l'UMMTO.*

Proposé et dirigé par :

Mr. M.LAHDIR

Etudié et réalisé par:

MENDACI Sara

Promotion LMD: 2011/2012

Remerciements

Je tiens avant tout à remercier M. Mourad LAHDIR mon promoteur, pour m'avoir permis de réaliser ce projet de fin d'étude et pour la confiance qu'il m'a accordé tout au long de la réalisation du projet.

Un merci tout particulier à mon mari Hocine, qui a beaucoup participé à ce travail, merci de m'avoir orienté, merci pour ton soutien.

Je remercie également tous les professeurs et enseignants qui ont collaboré à ma formation depuis mon premier cycle d'étude jusqu'à aujourd'hui.

Enfin, Que toute personne ayant contribué de près ou de loin à la réalisation de ce travail par une quelconque forme de contribution trouve ici le témoignage de ma plus profonde reconnaissance.

Dédicaces

Je dédie ce modeste travail :

A mes chers parents

A mes chères sœurs : Mina et Doucha

A mon Cher époux Hocine

A mes chers oncles et à ma très chère tante Amel

A toute ma famille

A tous mes ami(e)s et camarades d'études

Sommaire

| | |
|-----------------------------|---|
| Introduction générale | 1 |
|-----------------------------|---|

Chapitre I: Généralités

| | |
|--|----|
| I.1. Introduction | 2 |
| I.2. Architecture client/serveur | 2 |
| I.2.1. Présentation de l'architecture..... | 2 |
| I.2.2. Fonctionnement d'un système client/serveur..... | 2 |
| I.2.3. l'architecture client/serveur à 2 niveaux | 3 |
| I.2.4. l'architecture client/serveur à 3 niveaux | 3 |
| I.2.5. Comparaison des deux types d'architecture..... | 4 |
| I.2.6. L'architecture multi-niveaux..... | 5 |
| I.2.7. Avantages de l'architecture client/serveur..... | 5 |
| I.2.8. Inconvénients du modèle client/serveur | 5 |
| I.3. Notion d'Intranet | 5 |
| I.3.1. Présentation d'un intranet..... | 6 |
| I.3.2. L'utilité d'un intranet..... | 7 |
| I.3.3. Avantages d'un intranet | 7 |
| I.3.4. Mise en place de l'intranet | 8 |
| I.4. l'Extranet | 8 |
| I.5. L'Internet..... | 8 |
| I.5.1. Définition | 8 |
| I.5.2. Les différents services accessibles via Internet..... | 9 |
| I.5.3. L'internet et les protocoles | 9 |
| I.6. Domain Name System..... | 13 |
| I.6.1. Définition | 13 |
| I.6.2. Serveur DNS | 13 |
| I.7. Le World Wide Web | 14 |
| I.7.1. Définition | 14 |
| I.7.2. Les concepts du Web..... | 14 |
| I.7.3. Les langages du Web..... | 15 |
| I.7.4. World Wide Web Consortium..... | 16 |
| I.8. Site web | 16 |
| I.8.1. Site statique | 16 |
| I.8.2. Site dynamiques | 17 |
| I.9. Conclusion..... | 18 |

Chapitre II: Analyse et conception

| | |
|--|----|
| II.1. Introduction | 19 |
| II.2. Présentation du domaine d'application | 19 |
| II.3. Analyse fonctionnelle | 20 |
| II.3.1. Analyse de l'existant | 20 |
| II.3.2. Analyse des besoins | 20 |
| II.3.3. Identification des acteurs | 21 |
| II.3.4. Spécification des tâches | 22 |
| II.3.5. Spécification des scénarios | 23 |
| II.3.6. Diagramme de cas d'utilisation..... | 27 |
| II.4. La conception..... | 29 |
| II.4.1. Diagramme de séquence des cas d'utilisation..... | 29 |
| II.4.2. Diagramme de classe | 36 |
| II.4.3. Conception de la base de données..... | 37 |
| II.5. Conclusion | 37 |

Chapitre III: Implémentation et réalisation

| | |
|---|-------|
| III.1. Introduction | 38 |
| III.2. Environnement du travail | 38 |
| III.2.1. Les logiciels utilisés..... | 39 |
| III.2.1.1. XAMPP | 39 |
| III.2.1.2. Le CMS Joomla | 40 |
| III.2.1.3. Client FTP: Filezilla | 42 |
| III.2.2. Les outils utilisés | 42 |
| III.2.2.1. Le serveur web Apache | 42 |
| III.2.2.2. Le Serveur de bases de données MySQL | 43 |
| III.2.2.3. PhpMyAdmin | 43 |
| III.2.2.4. Jumi | 44 |
| III.2.2.5. FPDF..... | 45 |
| III.2.2.6. Toad for MySQL | 45 |
| III.2.1.7. Macromedia Dreamweaver 8..... | 45 |
| III.2.3. Les langages d'implémentation utilisés..... | 46 |
| III.3. Le modèle MVC (Model – View - Controler)..... | 48 |
| III.4. Principales interfaces de l'application | 49 |
| III.5. Conclusion | 55 |
| Conclusion générale..... | 56 |
| Annexes | 57-88 |

Introduction

Actuellement, le monde connaît une avancée technologique considérable dans tous les secteurs et cela grâce à l'informatique qui est une science qui étudie les techniques du traitement automatique de l'information. Elle joue un rôle important dans le développement de l'entreprise et d'autres établissements.

Les universités font partie intégrante des établissements que l'informatique pourra beaucoup aider. En effet, la croissance des étudiants et la variété des enseignements nécessitent d'introduire l'informatique dans les administrations universitaires et la mise en place d'une gestion rationnelle et rapide.

Ce travail porte sur « la conception et la réalisation d'une application web » pour apporter une solution aux problèmes de la gestion du département d'électronique de la faculté de Génie Electrique et d'Informatique de l'UMMTO. La façon actuelle de gestion, génère une lenteur et un manque d'efficacité. Ce projet comprend la mise en place de cette application qui doit être adaptée au département. Il s'agit de la concevoir et de la développer (en PHP & MySQL) en utilisant le CMS Joomla pour enfin trouver un moyen pour l'héberger et la mettre en ligne sur le web.

Le présent rapport se compose en trois chapitres. Le premier présente quelques généralités et définitions jugées nécessaires pour introduire notre travail. Le second chapitre s'attache à exposer une analyse au problème et la solution conçue exprimée principalement en diagrammes UML commentés brièvement. Le dernier chapitre traite la réalisation, les outils utilisés et le résultat à travers quelques captures d'écrans reflétant les fonctionnalités fournies par l'application.

I.1.Introduction

Le monde de l'informatique a connu un nouvel essor à la fin des années 90, avec la généralisation de l'internet qui a permis de communiquer, de dialoguer, d'échanger des documents et fournir des services divers permettant d'améliorer et de faciliter la vie quotidienne. Ce réseau mondial créé par l'armée américaine, puis utilisé par les chercheurs et autres scientifiques, a connu une croissance phénoménale auprès des entreprises et du grand public.

Ce chapitre a pour objectif de présenter quelques notions sur l'architecture client/serveur en premier lieu puis donnera un aperçu sur l'intranet, l'extranet et l'internet notamment le web et ses différents services.

I.2. Architecture client/serveur [1]

I.2.1. Présentation de l'architecture

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes (faisant partie du réseau) contactent un serveur, qui est une machine généralement très puissante en terme de capacités, qui leur fournit des services. Les services sont exploités par des programmes, s'exécutant sur les machines clientes. Lorsque l'on désigne un programme tournant sur une machine cliente, capable de traiter des informations qu'il récupère auprès d'un serveur (dans le cas du client FTP il s'agit de fichiers, tandis que pour le client de messagerie il s'agit de courrier électronique).

I.2.2. Fonctionnement d'un système client/serveur

Dans un système ou un réseau qui a une architecture client/serveur, on peut décrire le fonctionnement comme suit :

- Le client émet une requête vers le serveur grâce à son adresse IP 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 cliente et son port

Le schéma suivant nous montre encore le fonctionnement :

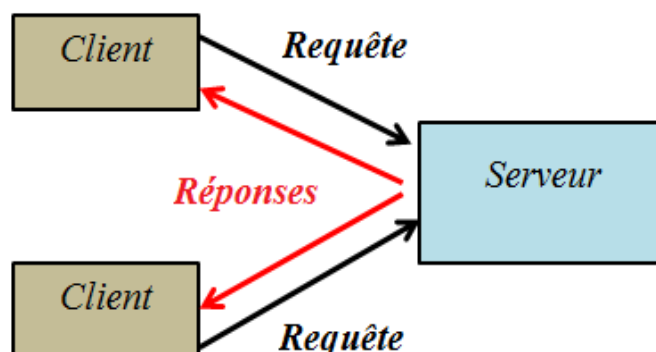


Figure 1.1 : Architecture client/serveur

I.2.3. l'architecture client/serveur à 2 niveaux

L'architecture à deux niveaux, aussi appelée *architecture 2-tier* (tier signifiant rangée en anglais) caractérise les systèmes clients/serveurs pour lesquels le client demande une ressource et le serveur la lui fournit directement, en utilisant ses propres ressources. Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir une partie du service.

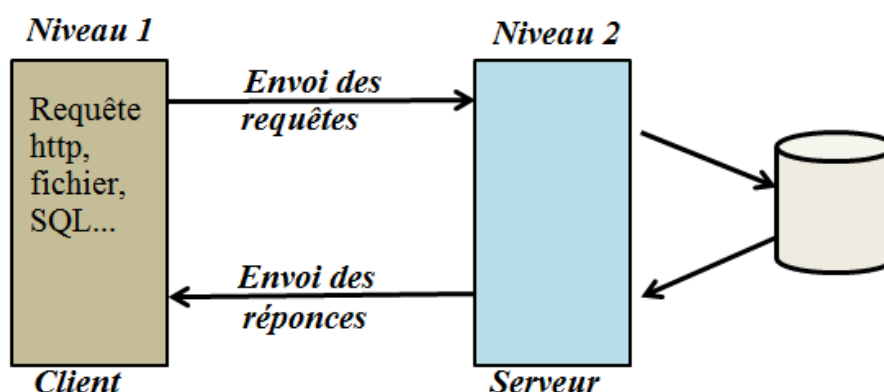


Figure 1.2 : Architecture à deux niveaux

I.2.4. l'architecture client/serveur à 3 niveaux

Dans l'architecture à 3 niveaux (appelée *architecture 3-tier*), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre un serveur, le client et un autre serveur secondaire.

- **Le client** : c'est-à-dire l'ordinateur demandeur de ressources, équipée d'une interface utilisateur (généralement un navigateur web) chargée de la présentation.
- **Le serveur d'applications** : chargé de fournir la ressource mais faisant appel à un autre serveur.
- **Le serveur secondaire** : généralement un serveur de base de données, fournissant au serveur d'applications les données dont il a besoin.

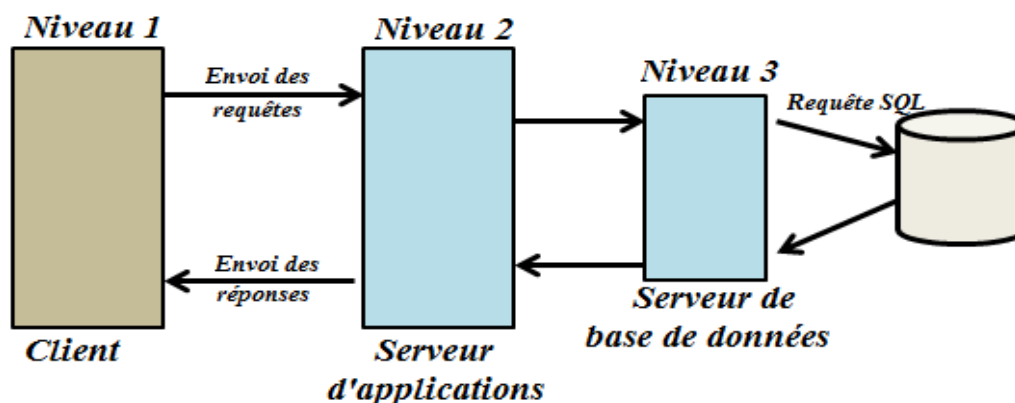


Figure 1.3 : Architecture à trois niveaux

I.2.5. Comparaison des deux types d'architecture

L'architecture à deux niveaux est donc une architecture client/serveur dans laquelle le serveur est polyvalent, c'est-à-dire 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'est-à-dire que chaque serveur est spécialisé dans une tâche (serveur web/serveur de base de données par exemple). Ainsi l'architecture à trois niveaux permet :

- Une plus grande flexibilité/souplesse.
- Une sécurité accrue car la sécurité peut être définie indépendamment pour chaque service, et à chaque niveau.
- De meilleures performances, étant donné le partage des tâches entre les différents serveurs.

I.2.6. L'architecture multi-niveaux

Dans l'architecture à 3 niveaux, chaque serveur (niveaux 2 et 3) effectue une tâche (un service) spécialisée. Un serveur peut donc utiliser les services d'un ou plusieurs autres serveurs afin de fournir son propre service. Par conséquent, l'architecture à trois niveaux est potentiellement une architecture à N niveaux...

I.2.7. Avantages de l'architecture client/serveur

Le modèle client/serveur est particulièrement recommandé pour des réseaux nécessitant un grand niveau de fiabilité, ses principaux atouts sont :

- **Des ressources centralisées** : étant donné que le serveur est au centre du réseau, il peut gérer des ressources communes à tous les utilisateurs, comme par exemple une base de données centralisée, afin d'éviter les problèmes de redondance et de contradiction.
- **Une meilleure sécurité** : car le nombre de points d'entrée permettant l'accès aux données est moins important.
- **Une administration au niveau serveur** : les clients ayant peu d'importance dans ce modèle, ils ont moins besoin d'être administrés.
- **Un réseau évolutif** : grâce à cette architecture il est possible de supprimer ou rajouter des clients sans perturber le fonctionnement du réseau et sans modification majeure.

I.2.8. Inconvénients du modèle client/serveur

L'architecture client/serveur a tout de même quelques lacunes parmi lesquelles :

- **Un coût élevé** dû à la technicité du serveur
- **Un maillon faible** : le serveur est le seul maillon faible du réseau client/serveur, étant donné que tout le réseau est architecturé autour de lui ! Heureusement, le serveur a une grande tolérance aux pannes.

I.3. Notion d'Intranet [12]

L'intranet est aujourd'hui un outil technologique adopté par un nombre croissant d'organisations. Il a un effet sur la communication, la gestion de l'information et aussi sur l'organisation du travail.

I.3.1. Présentation d'un intranet

Un intranet est un ensemble de services internet (par exemple un serveur web) internes à un réseau local, c'est-à-dire accessibles uniquement à partir des postes d'un réseau local, ou bien d'un ensemble de réseaux bien définis, et invisibles (ou inaccessibles) de l'extérieur. Il consiste à utiliser les standards client-serveur de l'internet (en utilisant les protocoles TCP/IP), comme par exemple l'utilisation de navigateurs internet (client basé sur le protocole HTTP) et des serveurs web (protocole HTTP), pour réaliser un système d'information interne à une organisation ou une entreprise.

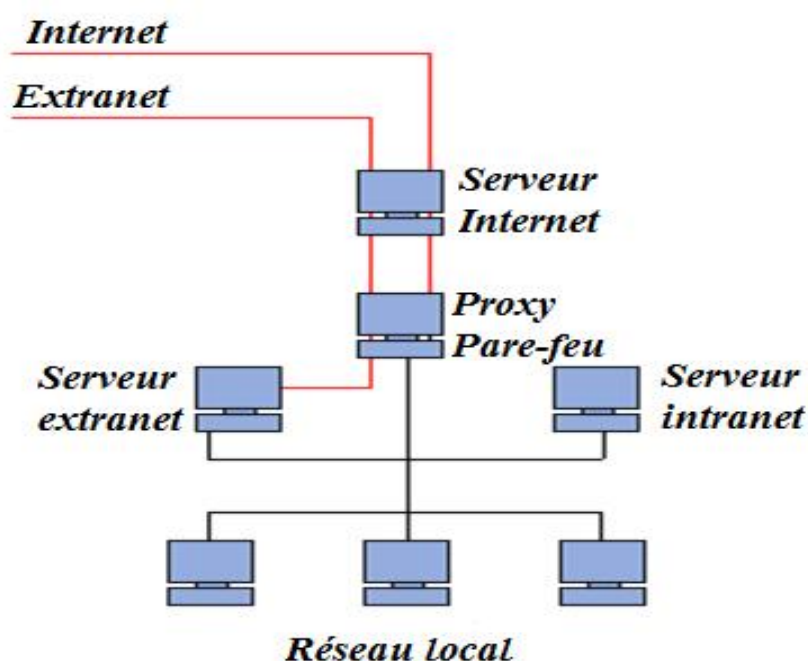


Figure 1.4 Architecture d'un Intranet

Un intranet repose généralement sur une architecture à trois niveaux, composée :

- de clients (navigateur internet généralement).
- d'un ou plusieurs serveurs d'application: un serveur web permettant d'interpréter des scripts CGI, PHP, ASP ou autres, et les traduire en requêtes SQL afin d'interroger une base de données.
- d'un serveur de bases de données.

Un intranet possède naturellement plusieurs clients (les ordinateurs du réseau local) et peut aussi être composé de plusieurs serveurs. Une grande entreprise peut par exemple posséder un

serveur web pour chaque service afin de fournir un intranet composé d'un serveur web fédérateur liant les différents serveurs gérés par chaque service.

I.3.2. L'utilité d'un intranet

Un intranet dans une entreprise permet de mettre facilement à la disposition des employés des documents divers et variés; cela permet d'avoir un accès centralisé et cohérent à la mémoire de l'entreprise, on parle ainsi de capitalisation de connaissances. De cette façon, il est généralement nécessaire de définir des droits d'accès pour les utilisateurs de l'intranet aux documents présents sur celui-ci, et par conséquent une authentification de ceux-ci afin de leur permettre un accès personnalisé à certains documents.

Des documents de tous types (textes, images, vidéos, sons, ...) peuvent être mis à disposition sur un intranet. De plus, un intranet peut réaliser une fonction de "**groupware**" très intéressante, c'est-à-dire permettre un travail coopératif. Voici quelques unes des fonctions qu'un intranet peut réaliser :

- Mise à disposition d'informations sur l'entreprise (panneau d'affichage)
- Mise à disposition de documents techniques
- Moteur de recherche de documentations
- Un échange de données entre collaborateurs
- Annuaire du personnel
- Gestion de projet, aide à la décision, agenda, ingénierie assistée par ordinateur
- Messagerie électronique
- Forum de discussion, liste de diffusion, chat en direct
- Visioconférence
- Portail vers internet

Ainsi, un intranet favorise la communication au sein de l'entreprise et limite les erreurs dues à la mauvaise circulation d'une information. L'information disponible sur l'intranet doit être mise à jour en évitant les conflits de version.

I.3.3. Avantages d'un intranet

Un intranet permet de constituer un système d'information à faible coût (concrètement le coût d'un intranet peut très bien se réduire au coût du matériel, de son entretien et de sa

mise à jour, avec des postes clients fonctionnant avec des navigateurs gratuits, un serveur fonctionnant sous Linux avec le serveur web *Apache* et le serveur de bases de données *MySQL*).

D'autre part, étant donné la nature universelle des moyens mis en jeu, n'importe quel type de machine peut être connecté au réseau local, donc à l'intranet.

I.3.4. Mise en place de l'intranet

Un intranet doit être conçu selon les besoins de l'entreprise ou de l'organisation (au niveau des services à mettre en place). Pour ce qui est de la mise en place matérielle, nous vous mettons en place un serveur web. Il suffit ensuite de s'y connecter par le navigateur web.

I.4. l'Extranet[2]

Un extranet est une extension du système d'information de l'entreprise à des partenaires situés au-delà du réseau. L'accès à l'extranet doit être sécurisé dans la mesure où cela offre un accès au système d'information à des personnes situées en dehors de l'entreprise. Il peut s'agir soit d'une authentification simple (authentification par nom d'utilisateur et mot de passe) ou d'une authentification forte (authentification à l'aide d'un certificat). Il est conseillé d'utiliser HTTPS pour toutes les pages web consultées depuis l'extérieur afin de sécuriser le transport des requêtes et des réponses HTTP et d'éviter notamment la circulation du mot de passe en clair sur le réseau.

Un extranet n'est donc ni un intranet, ni un site internet. Il s'agit d'un système supplémentaire offrant par exemple aux clients d'une entreprise, à ses partenaires ou à des filiales, un accès privilégié à certaines ressources informatiques de l'entreprise par l'intermédiaire d'une interface Web.

I.5. L'Internet

I.5.1. Définition [3]

Internet est le réseau informatique mondial qui rend accessibles au public des services variés comme le courrier électronique, la messagerie instantanée et le World Wide Web, en utilisant le protocole de communication IP (Internet Protocol). Son architecture technique qui repose sur une hiérarchie de réseaux lui vaut le surnom de réseau des réseaux.

Internet ayant été popularisé par l'apparition du World Wide Web, les deux sont parfois confondus par le public non averti. Le World Wide Web n'est pourtant que l'une des applications d'Internet.

L'accès à Internet peut être obtenu grâce à un fournisseur d'accès à Internet via divers moyens de télécommunication : soit filaire (réseau téléphonique commuté (bas débit), ADSL, fibre optique jusqu'au domicile...), soit sans fil (WiMAX, Internet par satellite, 3G+...). Un utilisateur d'Internet est désigné en français par le néologisme « internaute ».

I.5.2. Les différents services accessibles via Internet

Internet offre plusieurs services qui sont relatifs aux différents protocoles de communication parmi ces services nous citons:

- Ø **Messagerie électronique E-mail (Electronic-mail)** : c'est un service qui offre la possibilité à ses utilisateurs de communiquer et de recevoir des messages textes, des images, des programmes, des sons...par messagerie électronique.
- Ø **La messagerie instantanée** : service permettant l'échange instantané de messages textuels entre plusieurs ordinateurs.
- Ø **Les groupes de News : (News Group)** : appelé aussi les forums électroniques ou bien USENET, acronyme de Users Network (réseau d'utilisateurs). C'est une messagerie électronique qui permet à toute personne cherchant de l'aide ou faisant une recherche sur un sujet bien défini d'envoyer un courrier électronique vers un espace ouvert à tout le monde.
- Ø **La visioconférence** : permet aux utilisateurs de communiquer en directe sur Internet, avec échange de son et d'images, ainsi il pourra être considéré comme le plus grand média du futur.
- Ø **Le World Wide Web** : service permettant de consulter à l'aide d'un navigateur des documents multimédias reliés entre eux par des liens hypertextes.

I.5.3. L'internet et les protocoles [4]

Un protocole est une méthode standard qui permet la communication entre des processus (s'exécutant éventuellement sur différentes machines), c'est-à-dire un ensemble de règles et de procédures à respecter pour émettre et recevoir des données sur un réseau. Il en existe plusieurs selon ce que l'on attend de la communication. Certains protocoles seront par

exemple spécialisés dans l'échange de fichiers (le FTP), d'autres pourront servir à gérer simplement l'état de la transmission et des erreurs (c'est le cas du protocole ICMP), ...

Sur Internet, les protocoles utilisés font partie d'une suite de protocoles, c'est-à-dire un ensemble de protocoles reliés entre eux. Cette suite de protocole s'appelle TCP/IP.

Si on prend la présentation des couches réseaux TCP/IP, qui sont en principe structurées en quatre couches :

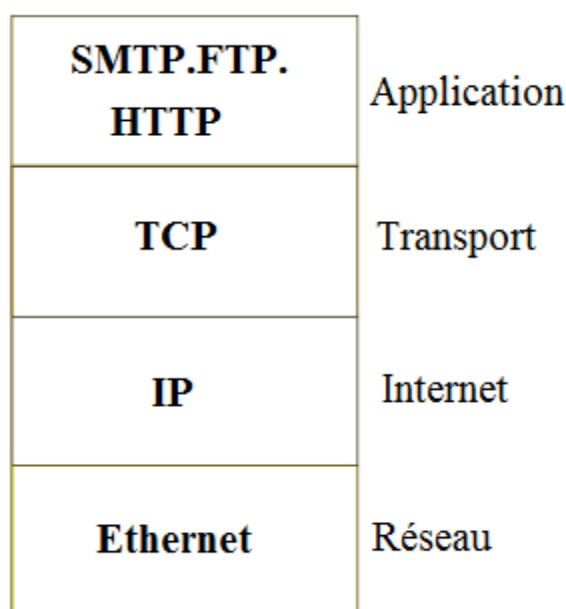


Figure 1.5 : Présentation famille TCP/IP

- **La couche Application** peut comporter des protocoles pour la messagerie (SMTP) ou le transfert de fichiers (FTP) ou la connexion entre ordinateur (Telnet) etc.
- **La couche transport** TCP (Transmission Control Protocol) assure la transmission des données sous forme de paquets (avec en-tête TCP) de bout en bout entre l'expéditeur et le destinataire, en utilisant les services de routage du protocole IP.
- **La couche internet ou routage** (Internet Protocol) ce dernier assure le routage des paquets TCP à travers le ou les réseaux jusqu'au destinataire.
- **La couche réseau** est spécifique au type de réseau gérant le transport des paquets sur le support physique (Ethernet, TokenRing, X25, etc...).

Pour savoir plus, prenant quelques exemples des différents protocoles :

Ø Le protocole FTP

Le *File Transfer Protocol* (protocole de transfert de fichiers), ou **FTP**, est un protocole de communication destiné à l'échange informatique de fichiers sur un réseau TCP/IP. Il

permet, depuis un ordinateur, de copier des fichiers vers un autre ordinateur du réseau, ou encore de supprimer ou de modifier des fichiers sur cet ordinateur.

FTP obéit à un modèle client-serveur, c'est-à-dire qu'une des deux parties, le *client*, envoie des requêtes auxquelles réagit l'autre, appelé *serveur*. En pratique, le serveur est un ordinateur sur lequel fonctionne un logiciel lui-même appelé serveur FTP, qui rend public une arborescence de fichiers. Pour accéder à un serveur FTP, on utilise un logiciel client FTP.

Ø Le protocole http

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

Le but du protocole HTTP est de permettre un transfert de fichiers (essentiellement au format HTML) localisés grâce à une chaîne de caractères appelée URL entre un navigateur (le client) et un serveur Web.

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

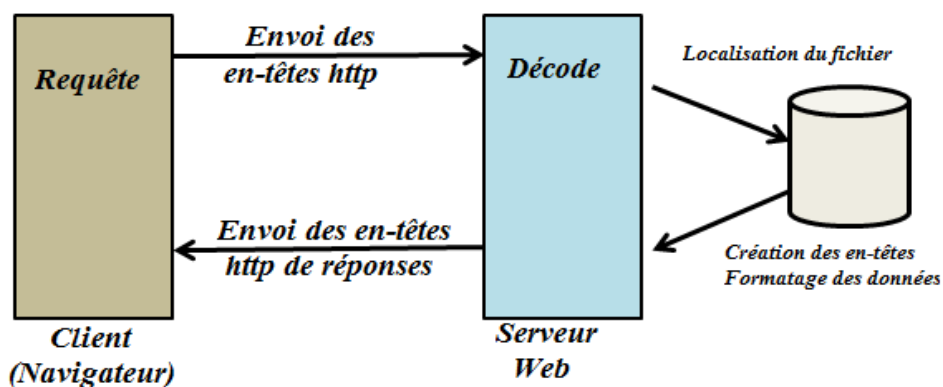


Figure 1.6 : La communication entre le client et le serveur

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

Ø Le protocole SMTP

Le protocole SMTP (*Simple Mail Transfer Protocol*, traduisez Protocole Simple de Transfert de Courrier) est le protocole standard permettant de transférer le courrier d'un serveur à un autre en connexion point à point.

Il s'agit d'un protocole fonctionnant en mode connecté, encapsulé dans une trame TCP/IP. Le courrier est remis directement au serveur de courrier du destinataire.

Ø Le protocole POP3

Le protocole **POP** (*Post Office Protocol* que l'on peut traduire par "protocole de bureau de poste") permet comme son nom l'indique d'aller récupérer son courrier sur un serveur distant (le serveur POP). Il est nécessaire pour les personnes n'étant pas connectées en permanence à Internet afin de pouvoir consulter les mails reçus hors connexion.

Ø Le protocole IMAP

Le protocole **IMAP** (*Internet Message Access Protocol*) est un protocole alternatif au protocole POP3 mais offrant beaucoup plus de possibilités :

- IMAP permet de gérer plusieurs accès simultanés
- IMAP permet de gérer plusieurs boîtes aux lettres
- IMAP permet de trier le courrier selon plus de critères

Ø Le protocole TCP

TCP (qui signifie *Transmission Control Protocol*, soit en français: Protocole de Contrôle de Transmission) est un des principaux protocoles de la couche transport du modèle TCP/IP. Il permet, au niveau des applications, de gérer les données en provenance de la couche inférieure du modèle (c'est-à-dire le protocole IP). Lorsque les données sont fournies au protocole IP, celui-ci les encapsule dans des datagrammes IP.

Ø Le protocole IP

Le protocole IP fait partie de la couche Internet de la suite de protocoles TCP/IP. C'est un des protocoles les plus importants d'Internet car il permet l'élaboration et le transport des datagrammes IP (les paquets de données). En réalité, le protocole IP traite les datagrammes IP indépendamment les uns des autres en définissant leur représentation, leur routage et leur expédition.

Le protocole IP détermine le destinataire du message grâce à 3 champs :

- Le champ adresse IP : adresse de la machine
- Le champ masque de sous-réseau : un masque de sous-réseau permet au protocole IP de déterminer la partie de l'adresse IP qui concerne le réseau
- Le champ passerelle par défaut : Permet au protocole Internet de savoir à quelle machine remettre le datagramme si jamais la machine de destination n'est pas sur le réseau local

I.6. Domain Name System

I.6.1. Définition [5]

Le Domain Name System (ou DNS, système de noms de domaine) est un service permettant de traduire un nom de domaine en informations de plusieurs types qui y sont associées, notamment en adresses IP de la machine portant ce nom.

I.6.2. Serveur DNS[6]

Un serveur DNS est un serveur qui permet de relier un nom de domaine avec une adresse IP. Plus concrètement, lorsque vous souscrivez à une offre d'hébergement pour votre site Internet, l'hébergeur vous donne la liste des serveurs DNS qui seront à installer sur votre nom de domaine afin de relier ce dernier à l'hébergement que vous avez souscrit et donc à votre site.

Le schéma suivant nous montre le fonctionnement :

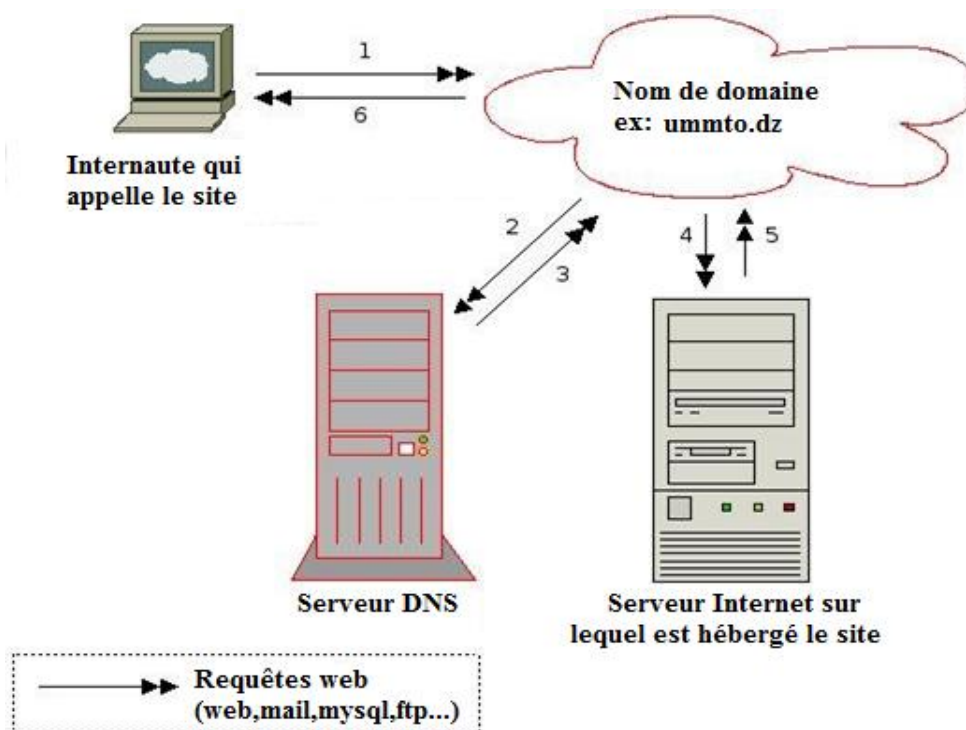


Figure 1.7 : Le fonctionnement d'un serveur DNS

Dans le schéma ci-dessus, l'internaute appelle, via son navigateur (requête 1), le nom de domaine qui va interroger son serveur DNS (requête 2 et 3) afin de connaître l'adresse IP du

serveur internet sur lequel est hébergé le site. Le nom de domaine va ensuite pouvoir échanger avec le serveur web (requête 4 et 5) afin de rendre visible le site pour l'internaute (requête 6).

I.7. Le World Wide Web

I.7.1. Définition

Le World Wide Web (WWW), littéralement la " toile (d'araignée) mondiale ", communément appelé le web, est un système hypertexte public fonctionnant sur internet qui permet de consulter avec un navigateur des pages accessibles sur des sites. L'image de la toile d'araignée vient des hyperliens qui lient les pages web entre elles.

Le Web n'est qu'une des applications d'Internet. Le Web a été inventé plusieurs années après Internet, mais c'est lui qui a rendu les médias grand public attentifs à Internet. Depuis, le Web est fréquemment confondu avec Internet ; en particulier, le mot Toile est souvent utilisé dans les textes non techniques sans qu'il soit clairement exprimé s'il désigne le Web ou Internet.[7]WWW a été développé par le CERN (laboratoire européen de recherche en physique des particules) à Genève. Le projet a démarré en mars 1989. L'objectif était de proposer une solution pour la communication de l'information dans la communauté de la physique des hautes énergies en utilisant l'internet.

I.7.2. Les concepts du Web

Le Web introduit plusieurs terminologies, dont on définit les suivantes :

- Ø **Page Web**: est l'unité de consultation de Word Wide Web. C'est un document informatique qui peut contenir du texte, des images, des formulaires à remplir et divers autres éléments multimédias et interactifs.
- Ø **Site Web** : c'est un ensemble de pages Web hyper liées entre elles et mises en ligne à une adresse Web.
- Ø **Navigateur Web** : est un logiciel (tel que internet explorer) conçu pour consulter le Word Wide Web, l'ensemble des sites Web.
- Ø **URL (Uniform Resource Locator)**: L'URL identifie l'endroit où se trouve une ressource sur le web.
- Ø **HTTP (Hyper Text Transfert Protocol)** : est le protocole de communication communément utilisé pour transférer les ressources du Web.

I.7.3. Les langages du Web

A. Coté client :

- Ø **HTML:** C'est le langage universel utilisé sur les pages Web lisibles par tous les navigateurs web. Ce langage fonctionne suivant l'assemblage et la combinaison de balises permettant de structurer et donner l'apparence voulue aux données textes, images et multimédias suivant la mise en page voulue.
- Ø **CSS:** Les feuilles de styles (en anglais "Cascading Style Sheets", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML ou XML.
- Ø **JavaScript:** JavaScript est un langage de programmation de scripts principalement utilisé pour les pages web interactives. C'est une extension du langage HTML qui est incluse dans le code. Ce langage est un langage de programmation qui permet d'apporter des améliorations au langage HTML en permettant d'exécuter des commandes.

B. Côté serveur :

- Ø **CGI (Common Gateway Interface) :** Est une interface permettant l'exécution de programmes externes par un serveur d'information de type http. Une CGI permet de faire communiquer un serveur HTTP et un programme pour générer d'une manière dynamique des documents HTML.
- Ø **PHP :** est un langage de programmation web côté serveur, ce qui veut dire que c'est le serveur qui va interpréter le code PHP (langage de scripts) et générer du code HTML qui pourra être interprété par votre navigateur. Le PHP permet d'ajouter des fonctionnalités de plus en plus complexe, d'avoir des sites dynamiques, de pouvoir gérer une administration de boutique en ligne, de modifier un blog, de créer des réseaux sociaux.
- Ø **ASP (Active Server Pages) :** Langage de programmation développé par Microsoft permettant l'exécution de script sur le serveur, de la même façon que le PHP ou le Perl.
- Ø **JSP (Java Server Protocol) :** est une technique basée sur Java qui permet aux développeurs de créer dynamiquement du code HTML, XML ou tout autre type de page web. Cette technique permet au code Java et à certaines actions prédéfinies d'être ajoutés dans un contenu statique.

Ø **Servlet:** Les servlets sont des applications java fonctionnant du côté serveur au même titre que les CGI et les langages de script côté serveur. Elles permettent donc de gérer des requêtes http et de fournir au client une réponse http dynamique.

I.7.4. World Wide Web Consortium [15]

Le W3C est un sigle utilisé pour définir le World Wide Web Consortium qui est une organisation non lucrative permettant de définir des standards pour les technologies liées au web. Il est important de noter que du point de vue européen, les standards fournis par cet organisme ne sont que des recommandations et non des normes standardisées. Cela permet de guider les technologies du web dans une même direction sur le long terme et ainsi améliorer leur compatibilité. Par la même occasion cela permet de mettre en place des directives sur des sujets variés. Cet organisme supervise un ensemble de standards tel que HTTP, HTML, XHTML, CSS, URL, XML, SVG, PNG et bien d'autres.

I.8. Site web

Le site web est un ensemble de pages Web et d'éventuelles autres ressources, liées dans une structure cohérente, publiées par un propriétaire (une entreprise, une administration, une association, un particulier, etc.) et hébergées sur un ou plusieurs serveurs Web. On considère qu'il existe 2 types de sites web : les sites **statiques** et les sites **dynamiques**.

I.8.1. Site statique

Les sites statiques ce sont des sites réalisés uniquement à l'aide des langages (X)HTML et CSS. Ils fonctionnent très bien mais leur contenu ne peut pas être mis à jour automatiquement, il faut que le propriétaire du site (le webmaster) modifie le code source pour y ajouter des nouveautés. Ce n'est pas très pratique quand on doit mettre à jour son site plusieurs fois dans la même journée. Les sites statiques sont donc bien adaptés pour réaliser des sites "vitrine", pour présenter par exemple son entreprise, mais sans aller plus loin. Ce type de site se fait de plus en plus rare aujourd'hui, car dès que l'on rajoute un élément d'interaction (comme un formulaire de contact), on ne parle plus de site statique mais de site dynamique. [9]

La figure qui suit illustre le fonctionnement d'un site statique.



Figure 1.8 : Le fonctionnement d'un site statique [9]

Lorsque le site est statique, cela se passe en deux temps :

1. Le client demande au serveur à voir une page web.
2. Le serveur lui répond en lui envoyant la page réclamée.

Sur un site statique, il ne se passe rien d'autre. Le serveur stocke des pages web et les envoie aux clients qui les demandent sans les modifier.

I.8.2. Site dynamiques

Un site dynamique est généralement associé au couple PHP/MySQL, sur lequel s'appuie bon nombre de CMS. Le contenu du site est désormais stocké dans une base de données (MySQL par exemple).

Le serveur Web reçoit une requête HTTP émise par le visiteur, il interroge la base de données puis construit une page de résultat qu'il restitue au visiteur via le navigateur. La page restituée pourra être différente en fonction de l'utilisateur connecté sur le site (accès autorisé ou pas à certaines parties du site par exemple). Le contenu du site pourra être mis à jour dynamiquement; par exemple, pour une page listant les articles de votre site, vous n'aurez à pas ajouter manuellement un lien vers un article fraîchement créé. Tout ce traitement est effectué par le serveur Web. [8]

La figure qui suit illustre le fonctionnement d'un site dynamique.



Figure 1.9 :Le fonctionnement d'un site dynamique [9]

Lorsque le site est dynamique, il y a une étape intermédiaire : la page est générée.

- Le client demande au serveur à voir une page web.
- Le serveur prépare la page spécialement pour le client.
- Le serveur lui envoie la page qu'il vient de générer.

La page web est générée à chaque fois qu'un client la réclame en sollicitant des bases de données. C'est précisément ce qui rend les sites dynamiques vivants : le contenu d'une même page peut changer d'un instant à l'autre.

I.9. Conclusion

Dans ce premier chapitre, nous avons présenté le modèle le plus utilisé pour l'échange des applications sur internet qui est le modèle client/serveur puis nous avons donné quelques notions générales sur les nouvelles technologies de l'information et de la communication .

Notre travail dans le cadre de ce mémoire concerne la conception et l'implémentation d'une application web dédiée à la gestion du département électronique de l'UMMTO. Le chapitre suivant est dédié à l'analyse et la conception de ce site.

II.1.Introduction

Depuis quelques années, la modélisation objet avec le langage UML est devenue incontournable sur la plupart des projets informatiques. L'un des domaines où l'UML est très utilisé est le développement web. Surtout que ce dernier (développement web) voit une complexité de plus en plus croissante.

Dans cette section du rapport, une idée générale sur la façon de gestion du projet et la modélisation de l'application est exposée. Il est à noter que, généralement, pour modéliser une application web trois diagrammes sont largement suffisants pour décrire statiquement et dynamiquement l'application.

II.2.Présentation du domaine d'application

Le département d'électronique est l'un des quatre départements de la faculté de Génie Electrique et d'Informatique(FGEI) de l'Université Mouloud Mammeri de Tizi-Ouzou(UMMTO). Il regroupe l'ensemble des filières de formation et un laboratoire de recherche destiné pour l'analyse des phénomènes aléatoires « Laboratoire d'Analyse et de Modélisation des Phénomènes Aléatoires » ou (LAMPA).

Le département assure les formations LMD (Licence, Master, Doctorat) suivantes :

Ø **Licence** : Le département reçoit des étudiants ayant passé avec succès les quatre semestres du domaine Sciences et techniques et après étude de fiches de vœux et sur la base des places pédagogiques disponibles.

- Licence professionnelle spécialité électronique industrielle
- Licence académique spécialité électronique

Ø **Master** : Le département propose une suite de Masters, dont l'entrée est sélective sur la base d'étude de dossiers par la commission Masters et de critères normalisés par le conseil de direction de l'université.

- Master professionnelspécialitéélectronique industrielle
- Master académiquespécialité télécommunication et réseaux

Ø **Formation en post graduation**: conduisant aux métiers de la recherche et du développement, de l'enseignement supérieur.

II.3. Analyse fonctionnelle

Le développement d'une application ne s'agit pas de se jeter sur l'écriture de code en omettant de formaliser les besoins des utilisateurs et d'élaborer une architecture et une modélisation qui nous guideront tout au long du projet. Mais d'autre côté, le but n'est pas de faire une modélisation juste pour le plaisir, mais bien de produire le plus rapide possible une application qui répond au mieux aux besoins des utilisateurs.

L'analyse est une activité qui débute par l'examen des diagrammes des cas d'utilisations et leurs scénarios ainsi que les besoins fonctionnels du système. A partir de cette analyse, on peut passer à la conception.

II.3.1. Analyse de l'existant

Ce paragraphe résume sous forme de points, le processus suivi actuellement par le département d'Electronique.

- La non-existence d'une application pour la gestion automatique des traitements ;
- Le principal outil utilisé au département d'Electronique est une application en Excel propre au département développé par ses enseignants, cette application permet de gérer les relevés de notes des étudiants;
- Les enseignants envoient par mail au chef de département les notes des étudiants sous forme de fichier Excel ;
- Le chef de département récupère les notes des étudiants envoyés par mail ;
- Le chef département consolide tous les fichiers Excel redus dans un autre fichier global, ce dernier contient des macros permettant d'éditer les relevé de notes des étudiants ;
- L'affichage de notes, d'emplois du temps, de plannings d'examens se fait au niveau de la scolarité du département d'électronique sous forme de papier ;
- Le processus nécessite plusieurs manipulations manuelles ce qui augmente la possibilité d'erreur.

II.3.2. Analyse des besoins

Un besoin est ce qu'on attend du système faire, ou l'objectif qui doit être accomplis par le système.

Une étude faite sur le département d'électronique, du quelle on a déduit le cahier des charges contenant les différents besoins et différentes contraintes auxquelles l'application doit répondre. Autrement dit ; le système doit atteindre un certain nombre de besoins.

Dans le but d'améliorer la gestion du département d'Electronique de l'UMMTO, nous proposons d'analyser, de concevoir et d'implémenter une application Web permettant la gestion du département. Pour ce faire notre système devra :

Ø Permettre aux étudiants de :

- Consulter les cours, les articles publiés par les enseignants.
- Consulter les emplois du temps et les plannings d'examens.
- L'envoi de mails.
- Accéder au service de la scolarité
- Consulter ses notes.
- Télécharger son relevé de note en PDF.
- Télécharger un document.
- L'accès au forum.

Ø Permettre aux enseignants de :

- Saisir et afficher les notes des étudiants.
- Déposer les cours, la documentation, des articles.

Ø Permettre au chef de département la gestion du département

Ø Permettre à l'administrateur de gérer tout le site.

II.3.3. Identification des acteurs

Après l'analyse de la situation, nous sommes arrivés à distinguer six acteurs qui vont agir sur notre système, et qui sont :

- **Le visiteur** (le grand public) : toute personne connectée au site et non authentifiée est considérée comme visiteur, ses droits de navigation sont assez limités.
- **L'étudiant** : sont les étudiants du département D'Electronique qui sont inscrits comme membres où chacun peut accéder à son espace personnel après son authentification. ils bénéficient des différents services offerts par l'application.
- **L'enseignant** : un membre du département qui contribue à enrichir le site en apportant des documents, cours, articles...
- **L'enseignant responsable** : aussi un membre comme tout autre enseignant, de plus il est responsable du module et il peut gérer les notes de son module.
- **Le chef de département** : la personne qui se charge de la gestion du département.

- **L'administrateur** : il a droit à accéder et modifier toutes les informations fonctionnelles de l'application et les différentes informations de configuration et de paramétrage.

A partir des acteurs identifiés ci-dessus, on peut construire le diagramme de contexte suivant.

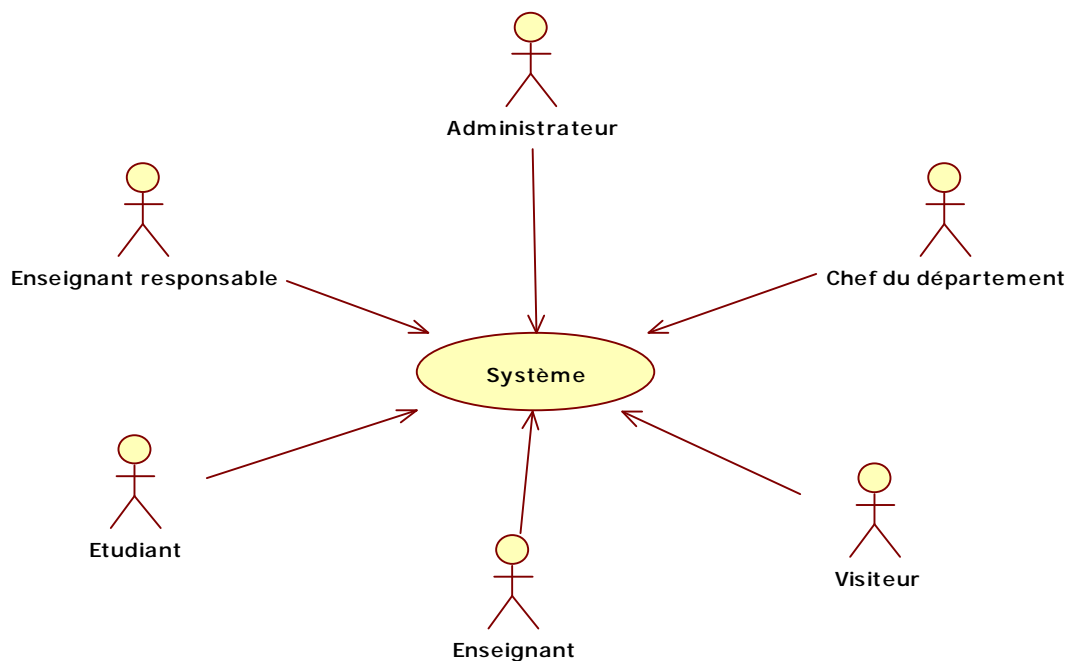


Figure 2.1 : diagramme de contexte du système

II.3.4. Spécification des tâches

Chacun des acteurs que nous avons définis précédemment, effectue un certain nombre de tâches qu'on résume dans le tableau suivant :

N.B : on symbolise la tâche par la lettre « T », exemple : Tache1 → T1

| Acteurs | Taches |
|-----------------|---|
| Visiteur | T0 : se connecter T1 : naviguer sur le site T2 : consulter des articles sur le site |
| Etudiant | T0, T1, T2 T3 : s'authentifier T4 : consulter cours |

| | |
|-------------------------------|--|
| | <p>T5 : consulter articles</p> <p>T6 : consulter mémoire</p> <p>T7 : consulter emplois du temps</p> <p>T8 : consulter plannings d'examens</p> <p>T9 : consulter les notes</p> <p>T10 : télécharger le relever de note</p> <p>T11 : demander un document</p> <p>T12: accéder à la messagerie</p> <p>T13: accéder au forum</p> |
| Enseignant | <p>T0-T8, T11-T13</p> <p>T14 : afficher les notes</p> <p>T15 : publier la documentation</p> <p>T16 : convoquer les étudiants</p> |
| Enseignant responsable | <p>T0-T8, T11-T16</p> <p>T17 : saisir et modifier les notes</p> |
| Chef de département | <p>T0-T8, T11-T17</p> <p>T18 : gérer les enseignants</p> <p>T19 : gérer les étudiants</p> <p>T20 : gérer les spécialités</p> <p>T21 : gérer les modules</p> <p>T22 : éditer planning</p> <p>T23 : consolider les notes</p> |
| Administrateur | <p>T0-T23</p> <p>T24 : gérer le paramétrage de l'application</p> <p>T25 : gérer les comptes des utilisateurs</p> <p>T26 : gérer la base de données</p> |

II.3.5. Spécification des scénarios

Chacune des tâches effectuées par un ou plusieurs acteurs sera décrite par un ensemble de scénarios. Les scénarios décrivant chacune des tâches définies auparavant sont récapitulés dans le tableau suivant :

N.B : on symbolise le scénario par la lettre « S », exemple : Scénario 1 → S1.

| Acteur | Tache | Scénario |
|-----------------|---|--|
| Visiteur | T0 : Se connecter | S0 : saisir l'URL dans la barre d'adresse du navigateur |
| | T1 : Naviguer sur le site | S1 : sélectionner les liens S2 : accéder aux pages |
| | T2 : Consulter des articles sur le site | S3 : sélectionner le lien S4 : visualiser les articles |
| Etudiant | T3 : S'authentifier | S0 S5: sélectionner le lien « mon compte » S6 : saisir le formulaire et valider |
| | T4 : Consulter cours | S7 : sélectionner le lien « documentation » S8 : sélectionner le lien « cours » S9: visualiser les cours |
| | T5 : Consulter articles | S7 S10 : sélectionner le lien « articles » S11 : visualiser les articles |
| | T6 : Consulter mémoire | S7 S12 : sélectionner le lien « mémoires » S13 : visualiser les mémoires |
| | T7 : Consulter emploi du temps | S14 : sélectionner le lien « services de la scolarité » S15 : sélectionner le lien « emploi du temps» S16 : visualiser l'emploi du temps |
| | T8 : Consulter plannings d'examens | S14 S17 : sélectionner le lien « calendrier d'examens» S18 : visualiser le planning |
| | T9 : Consulter les notes | S19 : sélectionner le lien « services de la scolarité » S20 : sélectionner le lien « consulter les notes » |

| | | |
|-------------------|---------------------------------|---|
| | | S21 : remplir un formulaire et valider S22 : visualiser les notes |
| | T10 : Editer le relever de note | S19 S23 : sélectionner le lien « relevé de note » S24 : renseigner des informations nécessaire au chargement du relevé de note. S25 : visualiser le relevé de note en PDF S26: imprimer le relevé de note |
| | T11 : Demander un document | S19 S27 : sélectionner le lien « document » S28 : visualiser le document en PDF S29 : imprimer le document |
| | T12: Accéder à la messagerie | S30 : sélectionner le lien « messagerie » S31 : consulter les messages S32 : écrire un message |
| | T13: Accéder au forum | S33 : sélectionner le lien « forum » S34 : ajouter un sujet S35 : consulter les messages S36: écrire un message S37 : reprendre à un message existant |
| Enseignant | T14 : Afficher les notes | S0, S5, S19 S38 : sélectionner le lien « afficher les notes » S39 : renseigner des informations nécessaires et valider. |
| | T15 : publier la documentation | S40 : sélectionner le lien « publier » S41 : renseigner des informations nécessaires et valider. |
| | T16 : Convoquer les étudiants | S42: sélectionner le lien « convocation » |

| | | |
|-------------------------------|-----------------------------|---|
| | | S43: remplir un formulaire et valider |
| Enseignant responsable | T17 : Saisir les notes | S0, S5, S19 S44: remplir un formulaire et valider S45 : Saisir le document affiché |
| Chef de département | T18 : Gérer les enseignants | S0, S5 S46 : sélectionner le lien « Gestion des enseignements » S47 : sélectionner le lien « enseignant » S48 : choisir un lien afin de pouvoir effectuer ; un ajout, suppression et modification. |
| | T19 : Gérer les étudiants | S46 S49 : sélectionner le lien « étudiants » S50 : choisir un lien afin de pouvoir effectuer ; un ajout, suppression et modification. |
| | T20 : Gérer les spécialités | S46 S51 : sélectionner le lien « spécialités » S52 : choisir un lien afin de pouvoir effectuer ; un ajout, suppression et modification. |
| | T21 : Gérer les modules | S46 S53 : sélectionner le lien « modules » S54 : choisir un lien afin de pouvoir effectuer ; un ajout, suppression et modification. |
| | T22 : Editer planning | S55 : sélectionner le lien «planning » S56 : choisir un lien afin de pouvoir effectuer un ajout ou suppression |
| | T23 : Consolider les notes | S57 : sélectionner le lien «validation » S58 : sélectionner le lien «notes » S59 : cliquer sur « valider » |

| | | |
|-----------------------|---|---|
| Administrateur | T24 : Gérer le paramétrage de l'application | S60 : sélectionner le lien «paramétrage » S61 : choisir l'opération à effectuer |
| | T25 : Gérer les comptes des utilisateurs | S 62: sélectionner le lien «Gérer Comptes » S 63: choisir l'opération à effectuer |
| | T26 : Gérer la base de données | S64 : choisir un lien qui correspond à une table dans la base de données afin de pouvoir effectuer ; un ajout, suppression et modification. |

II.3.6. Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation permet d'identifier les possibilités d'interaction entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire toutes les fonctionnalités que doit fournir le système. Il permet aussi de délimiter le système. [10]

Le diagramme de cas d'utilisation général de notre application est présenté dans la figure ci-dessous. Il permet de schématiser nos attentes du système à développer :

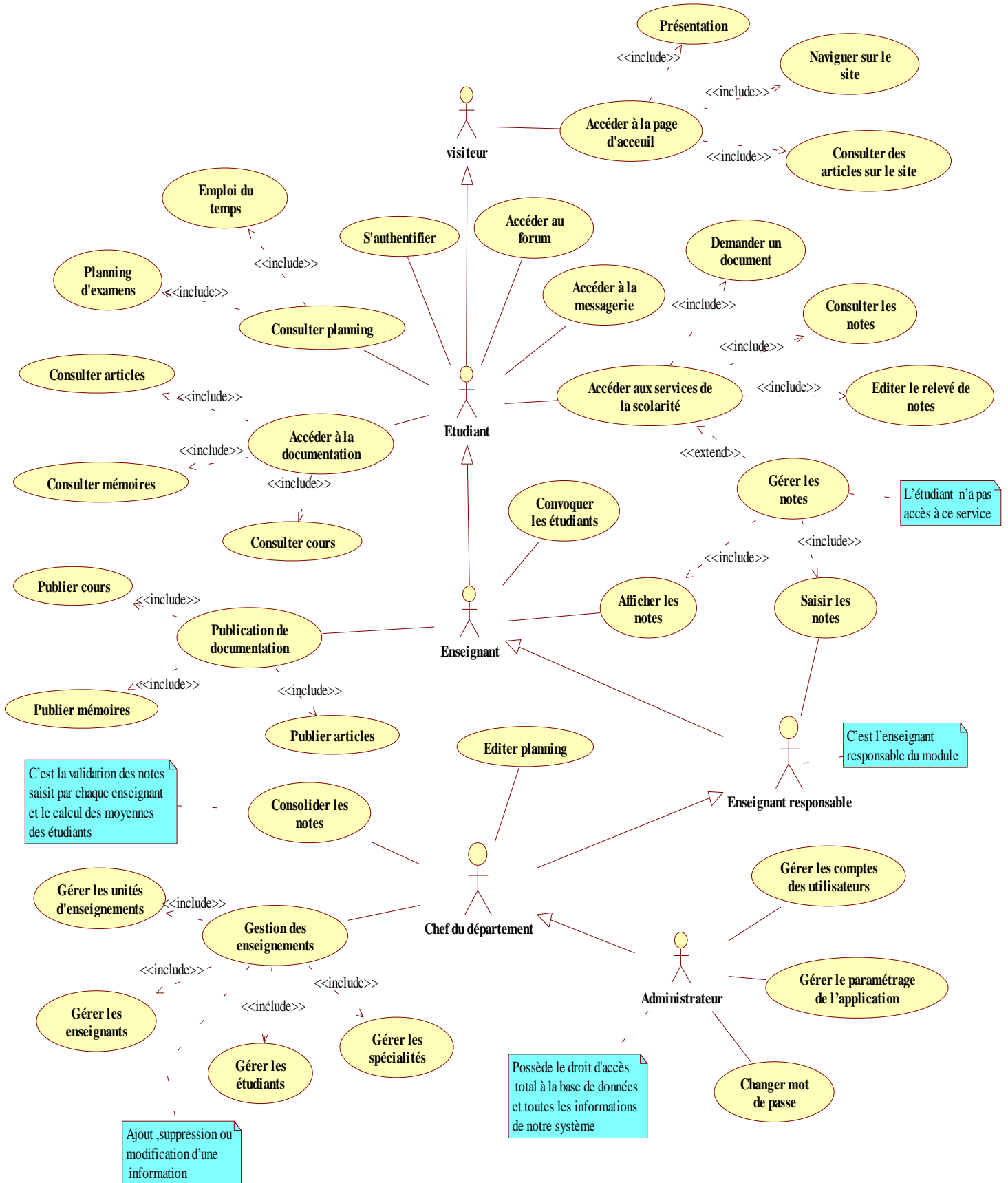


Figure 2.2 : diagramme de cas d'utilisation globale

II.4. La conception

L'analyse fonctionnelle du projet a fait apparaître les axes principaux de l'interface du système avec les utilisateurs, et la manière avec laquelle chaque personne utilise le système, nous allons maintenant spécifier le déroulement interne de ces opérations, et comment les différentes classes interagissent entre elles et avec la base de données pour mener à bien les différents processus.

II.4.1. Diagramme de séquence des cas d'utilisation

Il permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets. Un diagramme de séquence montre une interaction présentée en séquence dans le temps. En particulier, il montre aussi les objets qui participent à l'interaction par leur "ligne de vie" et les messages qu'ils échangent présentés en séquence dans le temps. Voici quelques notions de base du diagramme :

- **Scénario** : une liste d'actions qui décrivent une interaction entre un acteur et le système.
- **Interaction** : un comportement qui comprend un ensemble de messages échangés par un ensemble d'objets dans un certain contexte pour accomplir une certaine tâche.
- **Message** : Un message représente une communication unidirectionnelle entre objets qui transporte de l'information avec l'intention de déclencher une réaction chez le récepteur.

Dans cette partie, et pour détailler nos cas d'utilisation qui sont un peu généralistes, on appuie chaque scénario avec un diagramme de séquences système.

Ø Le cas d'utilisation « authentification d'un utilisateur »

- **Acteur principal** : personnel du département qui pourra être : l'administrateur, chef de département, enseignant et étudiant.
- **Objectif** : permettant à l'utilisateur d'accéder à son espace personnel et aux différents services offerts par l'application.
- **Scénario** :
 - L'utilisateur saisit l'URL
 - Le système affiche la page d'accueil
 - L'utilisateur sélectionne un lien qui nécessite une authentification par exemple : « identification »
 - Le système affiche une fenêtre d'authentification
 - L'utilisateur saisit son login avec son mot de passe et les valide

- Le système vérifie que ce nom d'utilisateur et le mot de passe correspondant existe dans sa base de données et ouvre à l'utilisateur les droits sur le site qui correspondent à ce nom d'utilisateur.

- **Diagramme de séquence**

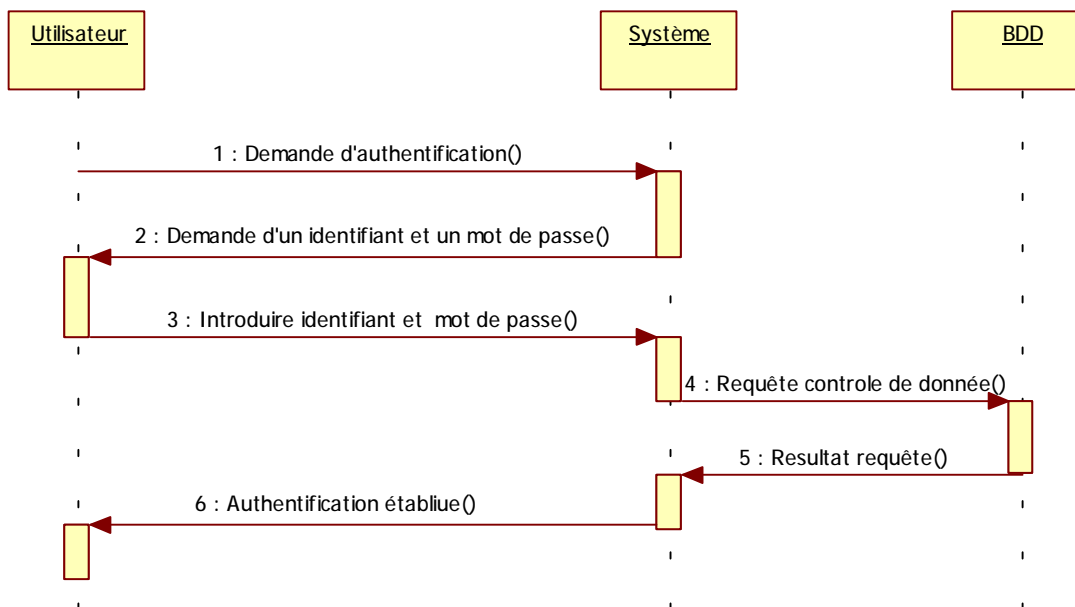


Figure 2.3 : Diagramme de séquence du cas d'utilisation «authentification »

Ø **Le cas d'utilisation «consulter les notes »**

- **Acteur principal:** étudiant.
- **Objectif :** cette fonctionnalité permet aux étudiants du département d'électronique de consulter leurs notes affichées par les enseignants.
- **Scénario:**
 - L'étudiant saisie l'URL
 - Le système affiche la page d'accueil
 - L'étudiant s'authentifie et accède à son espace
 - L'étudiant clique sur le lien « Accéder aux services de la scolarité »
 - Le système affiche la liste de services
 - L'étudiant clique sur le lien « Consulter les notes »
 - Le système affiche un formulaire à renseigner permettant la récupération des notes de l'utilisateur
 - L'étudiant saisie le formulaire puis clique sur valider

- Le système envoie une requête de recherche dans la base de données
- Le SGBD effectue la requête et envoie le résultat au système
- Le système affiche les notes publiées par l'enseignant

- **Diagramme de séquence**

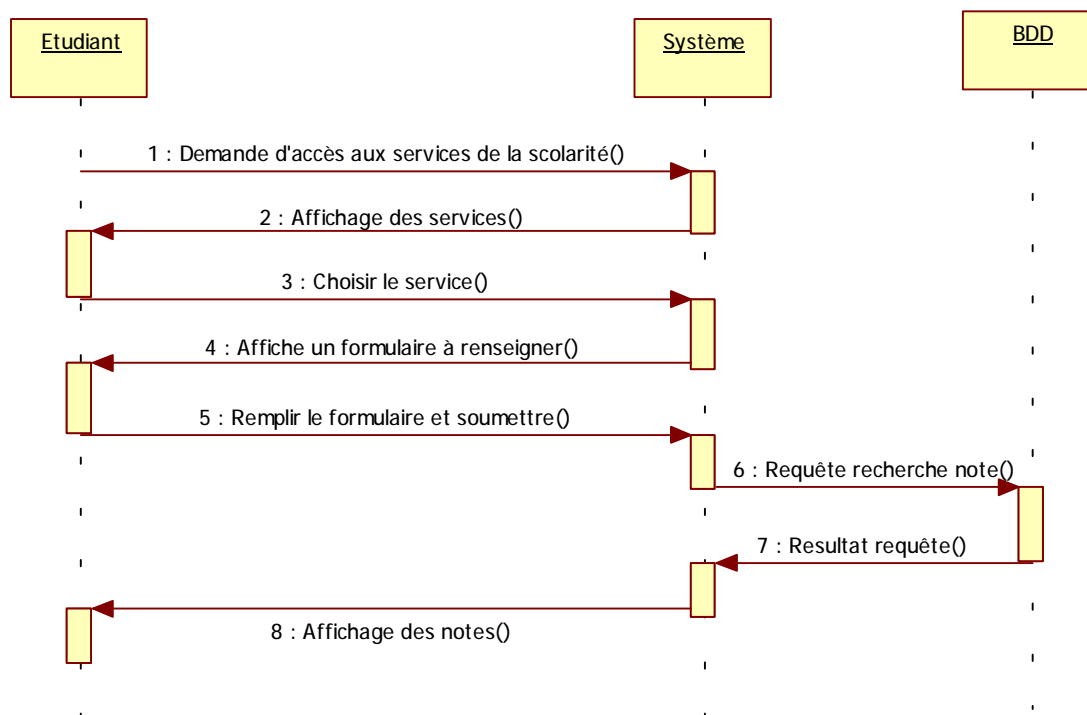


Figure 2.4 : Diagramme de séquence du cas d'utilisation «consulter les notes »

Ø **Le cas d'utilisation «afficher les notes »**

- **Acteur principal :** il pourra être l'enseignant ou l'enseignant responsable
- **Objectif :** cette fonctionnalité permet aux enseignants d'afficher les notes des étudiants.
- **Scénario :**
 - L'enseignant saisie l'URL
 - Le système affiche la page d'accueil
 - L'enseignant s'authentifie et accède à son espace
 - L'enseignant clique sur le lien « Accéder aux services de la scolarité »
 - Le système affiche la liste de services
 - L'enseignant clique sur le lien « afficher les notes »
 - Le système affiche un formulaire à renseigner

- L'enseignant saisie le formulaire et il joint un fichier puis clique sur valider
- Le Système met à jour les données et affiche un message de confirmation

- **Diagramme de séquence**

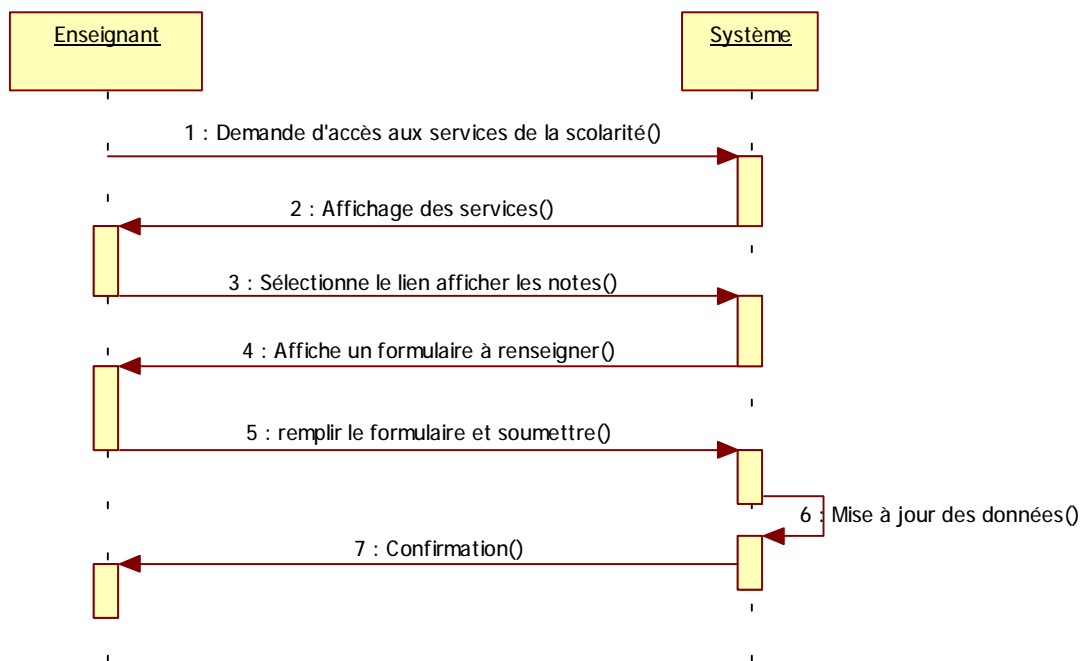


Figure 2.5 : Diagramme de séquence du cas d'utilisation «afficher les notes »

Ø **Le cas d'utilisation «saisir les notes »**

- **Acteur principal :** c'est l'enseignant responsable.
- **Objectif :** cette fonctionnalité permet aux enseignants responsables du module d'ajouter les notes des étudiants dans la base de données.
- **Scénario :**
 - L'enseignant responsable saisie l'URL
 - Le système affiche la page d'accueil
 - L'enseignant responsable s'authentifie et accède à son espace
 - L'enseignant responsable clique sur le lien « Accéder aux services de la scolarité »
 - Le système affiche la liste de services
 - L'enseignant clique sur le lien « saisir les notes »
 - Le système affiche un formulaire à renseigner
 - L'enseignant responsable saisie le formulaire et il joint un fichier.csv puis clique sur valider

- Le système envoie une requête pour ajouter les données dans la base de données
- Le SGBD effectue la requête et envoie le résultat au système
- Le système affiche un message de confirmation

- **Diagramme de séquence**

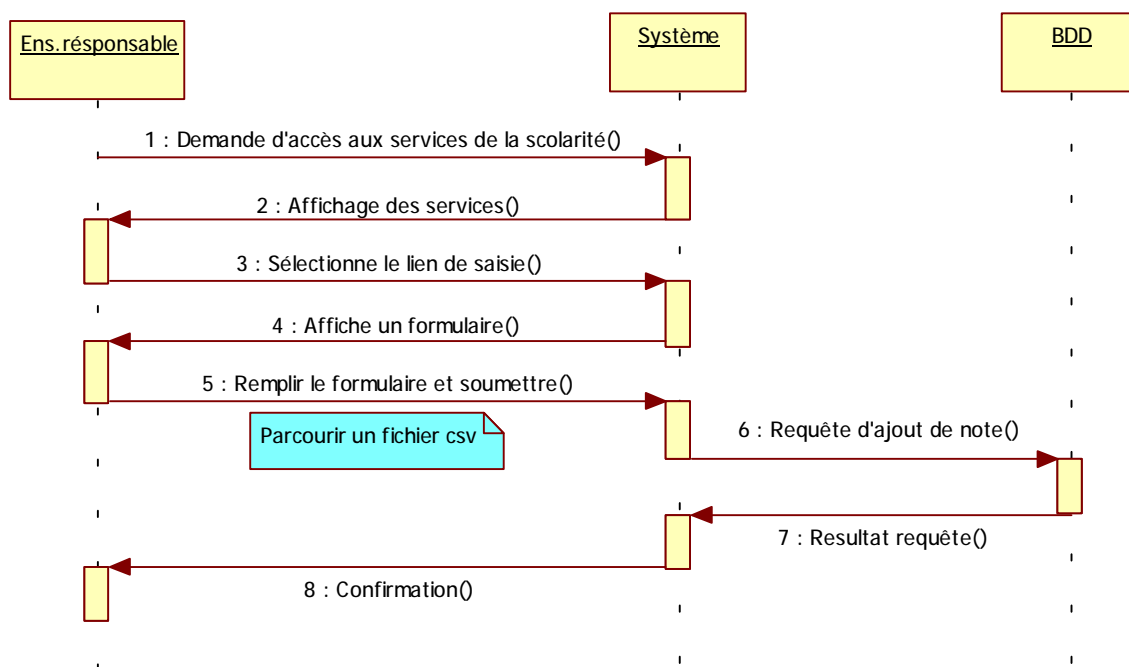


Figure 2.6 : Diagramme de séquence du cas d'utilisation « saisir les notes »

Ø **Le cas d'utilisation «visualisation d'un article »**

- **Acteur principal :** personnel du département qui pourra être : l'administrateur, chef de département, enseignant et étudiant.
- **Objectif :** cette fonctionnalité permet au personnel du département de consulter et de télécharger les articles publiés.
- **Scénario :**
 - L'utilisateur saisie l'URL
 - Le système affiche la page d'accueil
 - L'utilisateur s'authentifie et accède à son espace
 - L'utilisateur clique sur le lien « documentation »
 - Le système affiche la liste de documentation existante, il y a trois types de documentation : cours, articles et mémoires.
 - L'utilisateur sélectionne un article pour ouverture ou téléchargement

- Le système ouvre l'article et affiche les informations relatives

- **Diagramme de séquence**

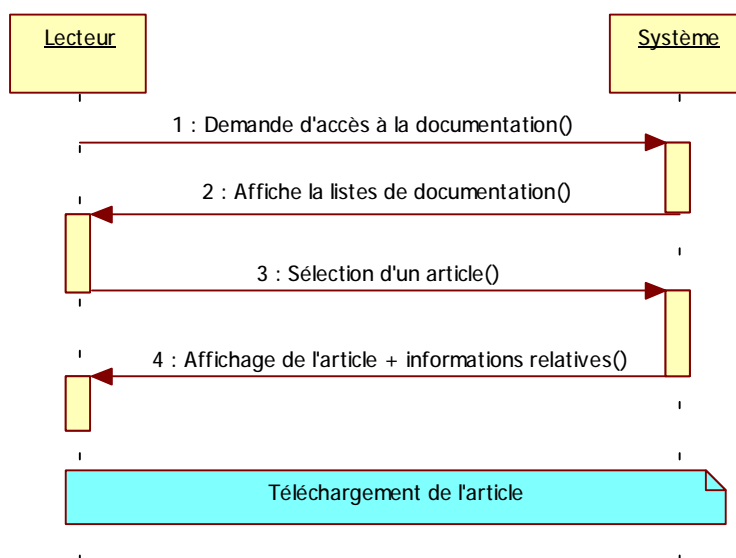


Figure 2.7: Diagramme de séquence du cas d'utilisation «visualisation d'un article »

Ø **Cas d'utilisation «Gestion des étudiants»**

- **Acteur principal** : c'est le chef de département.
- **Objectif** : permettant au chef de département d'ajouter, modifier ou supprimer des informations concernant les étudiants.
- **Scénario** :
 - Le chef de département saisie l'URL
 - Le système affiche la page d'accueil
 - Le chef de département s'authentifie et accède à son espace
 - Le chef de département clique sur le lien « Gestion des étudiants »
 - Le système affiche trois actions possibles à effectuer :
 - Ü Consultation détaillée
 - Ü Modification des différentes informations concernant les étudiants.
 - Ü L'ajout ou suppression d'une information, qui doit s'effectuer après une confirmation
 - Le chef de département choisit l'action à effectuer
 - Le système confirme l'action effectuée

• Diagramme de séquence

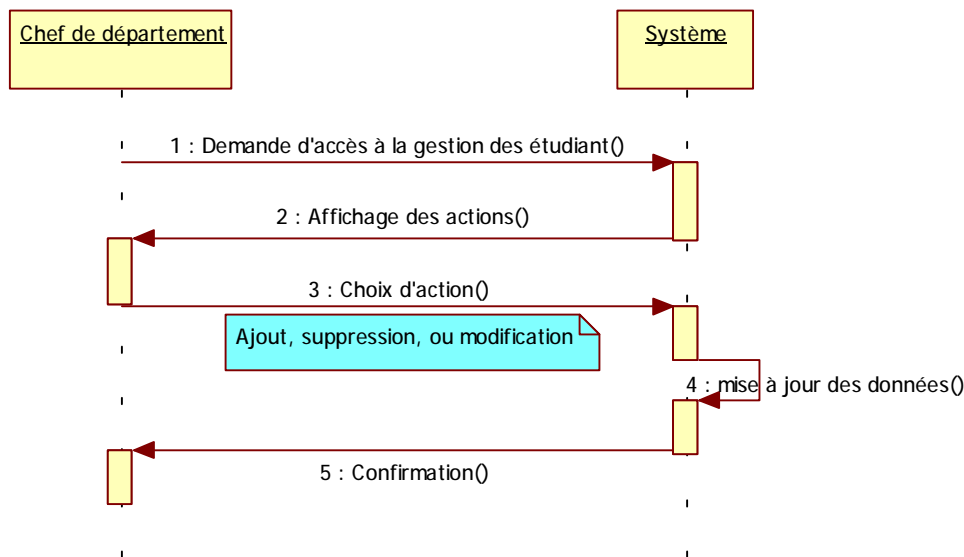


Figure 2.8: Diagramme de séquence du cas d’utilisation «gestion des étudiants»

Ø Cas d’utilisation « gestion des comptes utilisateurs »

- **Acteur principal** : administrateur.
- **Objectif** : la gestion des comptes offre la possibilité de créer, modifier, et supprimer des comptes, elle regroupe principalement la gestion des différents utilisateurs du système et des groupes à lesquels ils appartiennent.
- **Scénario** :
 - L’administrateur s’authentifie et accède à son espace
 - L’administrateur clique sur le lien « gestion des utilisateur »
 - Le système affiche trois actions possibles à effectuer :
 - ü Création de comptes en les affectant à des utilisateurs, en définissant les profils (étudiant, enseignant, enseignant responsable du module, chef de département, administrateur),
 - ü Mise à jour des comptes,
 - ü Suppression, qui doit s’effectuer après une confirmation
 - L’administrateur choisit l’action à effectuer
 - Le système confirme l’action effectuée

- Diagramme de séquence

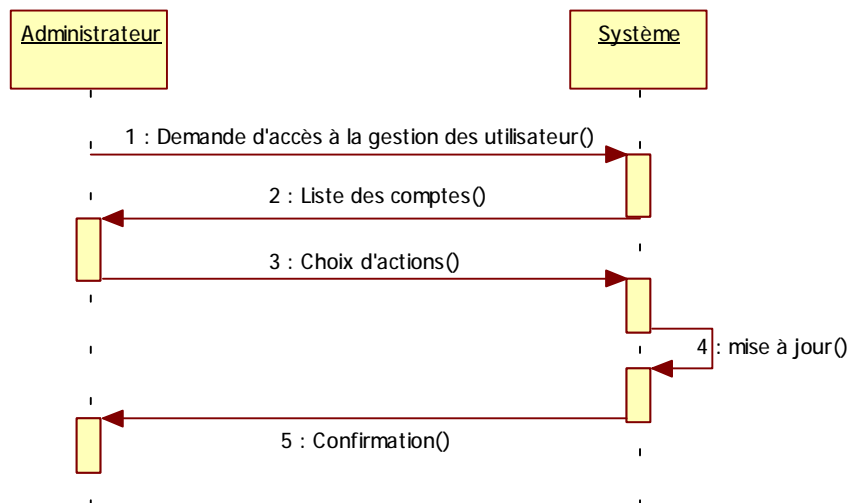


Figure 2.9: Diagramme de séquence du cas d'utilisation «gestion des comptes»

II.4.2. Diagramme de classe

Un diagramme de classe représente la structure statique d'un système. Il constitue un élément important de la modélisation des données. Il identifie la structure des classes du système y compris les propriétés et les méthodes de chacune. Les diverses relations qui peuvent exister entre les classes y sont également représentées. Un diagramme de classe proprement réalisé permet de structurer le travail de développement de manière très efficace.

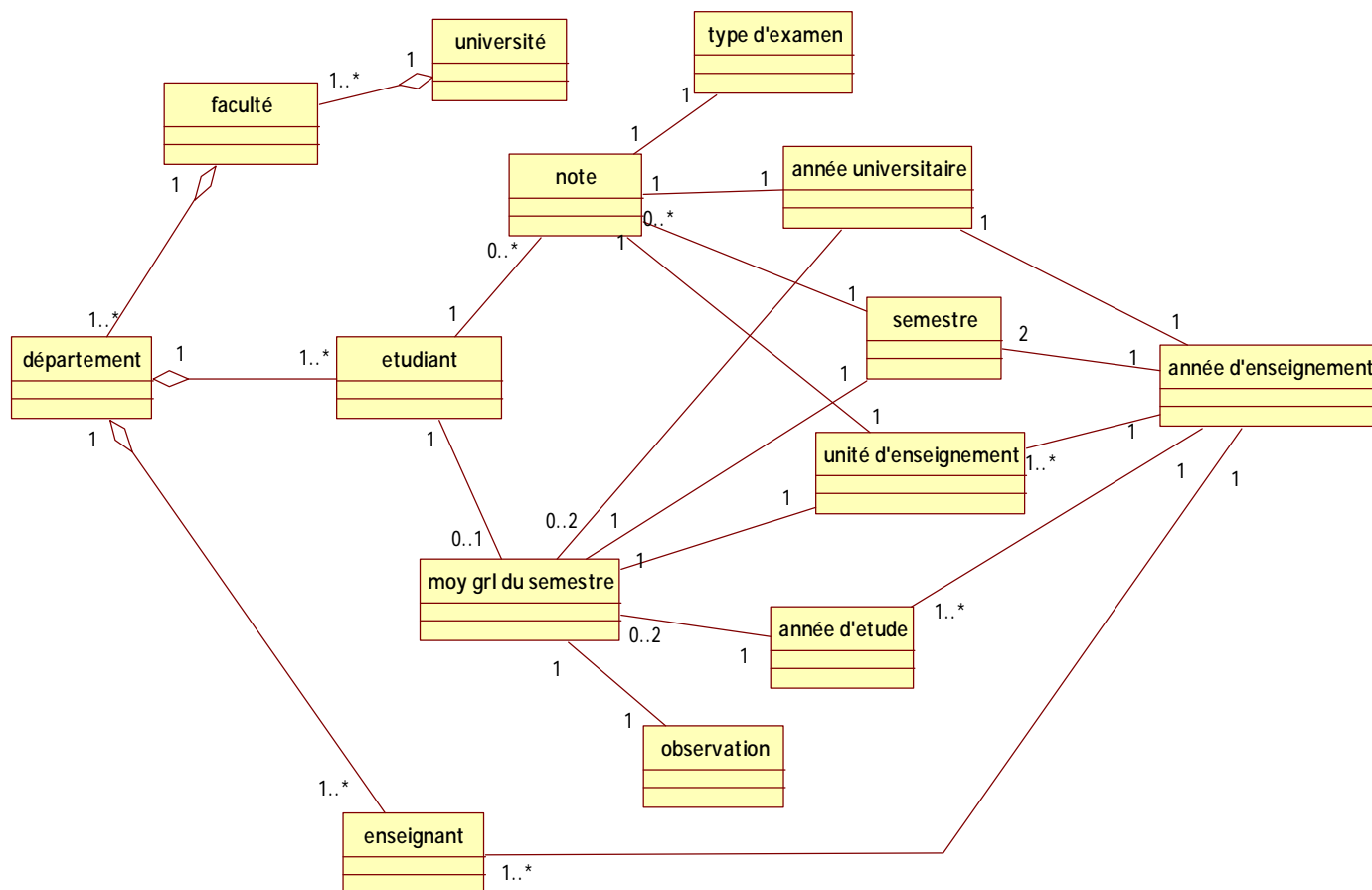


Figure 2.10 :Extraitdu diagramme de classes.

II.4.3. Conception de la base de données

Après avoir modélisé notre application avec les différents diagrammes vu précédemment ; nous nous intéressons maintenant aux données nécessaires pour le fonctionnement de l’application. Pour les obtenir, nous suivons de près le déroulement de chaque cas d’utilisation et le diagramme de classe qui nous renseigne sur les données nécessaires, qui seront stockées dans des tables de la base de données. (Voir annexe B).

II.5. Conclusion

La méthode de modélisation UML nous a permis de décrire statiquement et dynamiquement notre application et d’arrêter une solution globale nous permettant de passer à l’action et mettre en œuvre cette solution.

Le chapitre qui suit sera consacré à l’implémentation et la réalisation de notre application.

III.1. Introduction

La mise en œuvre de l'étude théorique et fonctionnelle que nous avons faite précédemment nous permet d'avoir une vue assez complète afin de réaliser notre application. Nous allons présenter dans ce chapitre l'environnement de développement, les outils choisis pour l'implémentation de notre application, ainsi que des captures d'écrans de quelques interfaces présentant les différentes fonctionnalités de notre système vu dans le chapitre précédent.

III.2. Environnement du travail

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

Ø Configuration logicielle :

- Xampp, permettant l'installation et la configuration de :
 - Le SGBD MySQL
 - Un serveur Apache
 - Une interface web PHPMyAdmin
- Le CMS Joomla.
- Client FTP: Filezilla
- Outils de développements:
 - Toad for MySQL
 - Macromedia Dreamweaver 8.

Ø Langages de programmation :

- HTML : pour la gestion et création des interfaces.
- PHP : pour le traitement des données.
- CSS : pour définir et de contrôler les styles d'affichage.
- SQL : pour l'interrogation de la base de données.
- JavaScript : pour la vérification des champs des formulaires etc...

On donne ci-après un bref aperçu sur l'ensemble des logiciels, outils et langages qu'on vient de citer.

III.2.1. Les logiciels utilisés

III.2.1.1. XAMPP [11]

XAMPP est un ensemble de logiciels permettant de mettre en place facilement un serveur Web et un serveur FTP. Il s'agit d'une distribution de logiciels libres (XApache MySQL Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide. Ainsi, il est à la portée d'un grand nombre de personnes puisqu'il ne requiert pas de connaissances particulières et fonctionne de plus, sur les systèmes d'exploitation les plus répandus.

Cette distribution se chargera donc d'installer l'ensemble des outils dont vous pourriez avoir besoin lors de la création d'une application Web. Plus d'une dizaine d'utilitaires sont intégrés, comme MySQL, PHP, Perl ou encore phpMyAdmin.

Ø Installer XAMPP

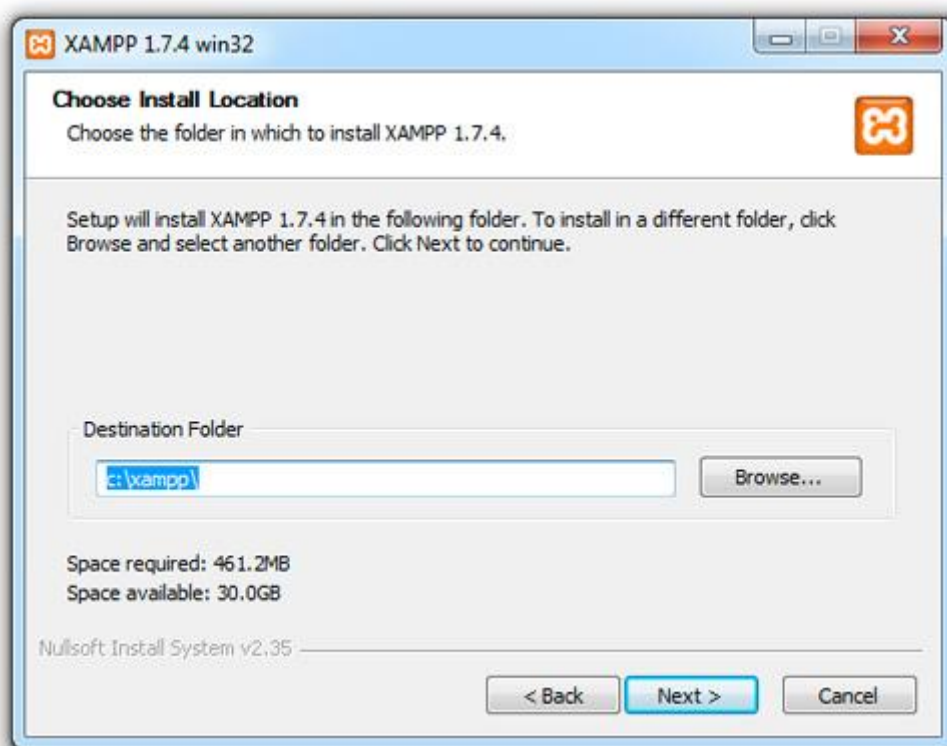


Figure 3.1 : L'assistant d'installation de XAMPP win32.

Quand l'installation est complétée, vous trouverez XAMPP sous Démarrer / Programmes / XAMPP. Grâce au panneau de contrôle XAMPP, vous pouvez démarrer/arrêter chacun des serveurs et installer/désinstaller les services.

Ø Lancer XAMPP

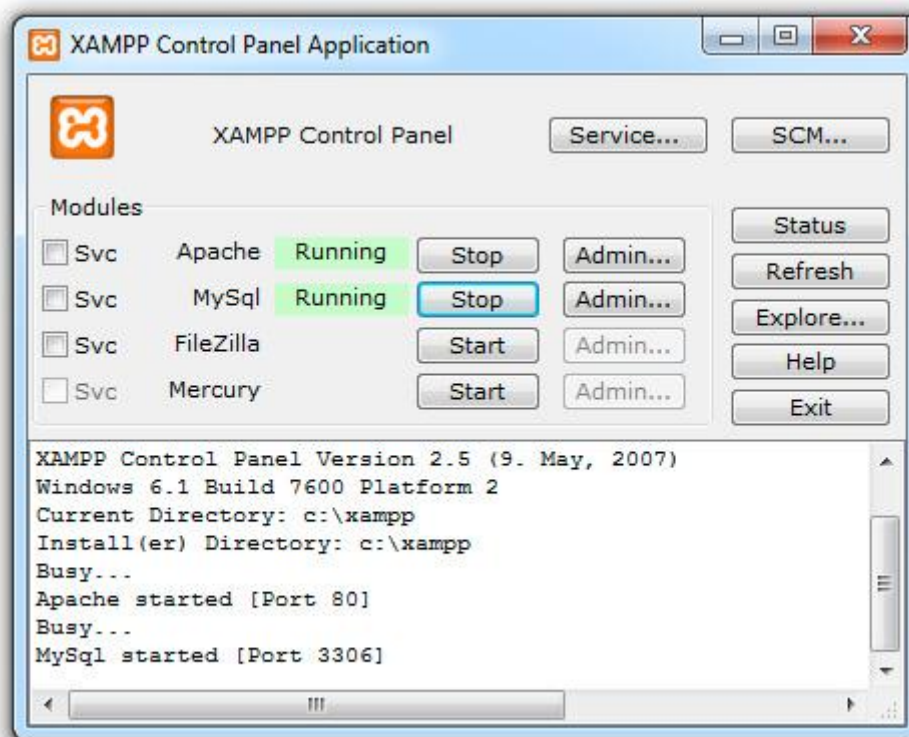


Figure 3.2 : lancement de XAMPP

Le panneau de contrôle XAMPP pour arrêter/démarrer Apache, MySQL, FileZilla et Mercury ou installer ces serveurs en tant que services. Pour tester : après le démarrage d'Apache, visitez <http://localhost> ou <http://127.0.0.1> et examinez tous les exemples et outils XAMPP.

III.2.1.2. Le CMS Joomla[13]

Ø Définition d'un CMS

Un système de gestion de contenu ou SGC ((en Anglais) Content Management System ou CMS) est une famille de logiciels destinés à la conception et à la mise à jour dynamique de site web ou d'application multimédia. Ils partagent les fonctionnalités suivantes:

- Ils permettent à plusieurs individus de travailler sur un même document ;
- Ils fournissent une chaîne de publication (workflow) offrant par exemple la possibilité de mettre en ligne le contenu des documents ;
- Ils permettent de séparer les opérations de gestion de la forme et du contenu ;
- Ils permettent de structurer le contenu (utilisation de FAQ, de documents, de blogs, de forums de discussion, etc.) ;

Et pour l'administration de notre projet on a choisi le CMS *Joomla*.

Ø Présentation du CMS Joomla

Joomla est un système de gestion de contenu (en anglais CMS pour Content Management System) libre, open source et gratuit. Il est écrit en PHP et utilise une base de données MySQL. Joomla inclut des fonctionnalités telles que des flux RSS, des news, une version imprimable des pages, des blogs, des recherches. Joomla est sous licence GNU GPL. Il est créé à partir du CMS Open Source Mambo en août 2005 suite aux désaccords d'une majorité des développeurs open source avec la société propriétaire du nom Mambo. Celle-ci pour accroître sa notoriété voulait utiliser ce nom pour le CMS propriétaire qu'elle développait en parallèle. La très grande majorité des utilisateurs a rapidement suivi le mouvement.

Dans le concours 2008 du meilleur CMS, Joomla est arrivé 2^{ème} dans la catégorie Overall Winner Open Source CMS Award (derrière Drupal) et 2^{ème} dans la catégorie Best PHP Open Source Content Management System (derrière Drupal).

La figure qui suit représente l'interface Administration de Joomla ! :

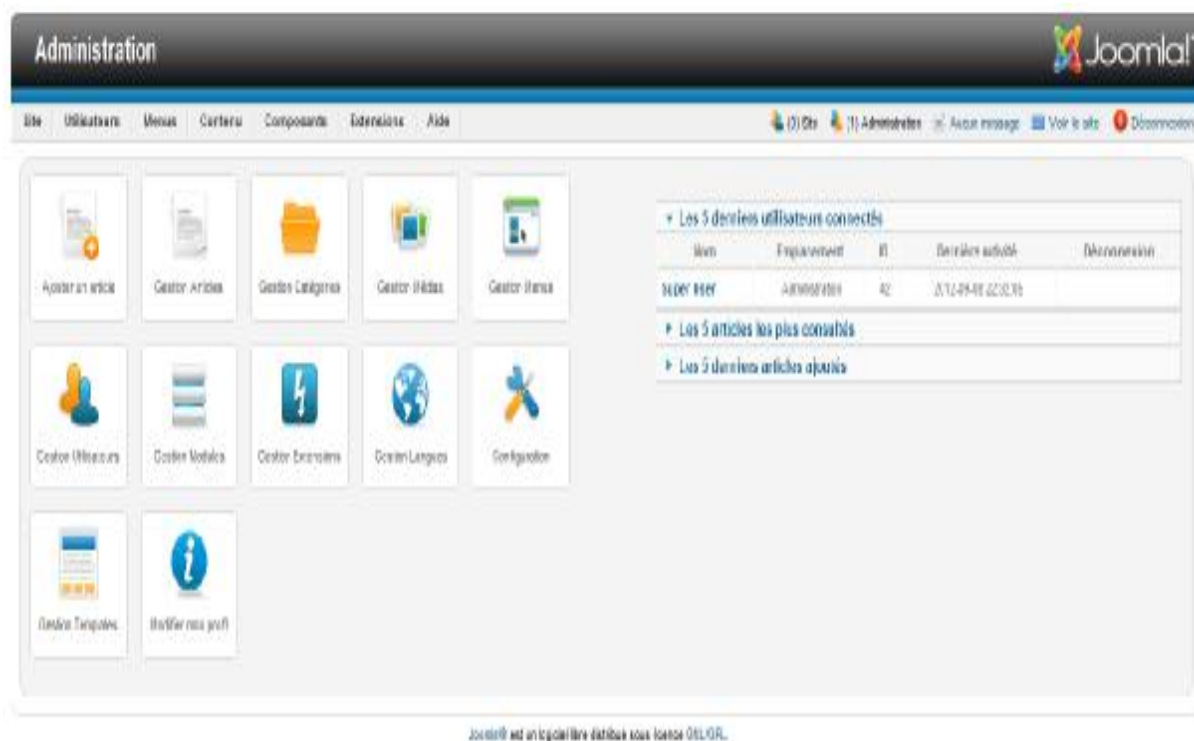


Figure 3.3 : L'interface Administration de Joomla

Avec ce CMS nous avons les avantages et options suivantes:

- Site dynamique entièrement géré par une base de données ;
- Gestion des utilisateurs ;
- Des menus, sous menus, sections, catégories, dynamiques modifiables directement en ligne ;
- Édition du contenu par un éditeur de texte de type Word ;
- Charte graphique personnalisable et échangeable ;
- Possibilité de site multi-langues ;
- Des modules fonctionnels prêt à l'emploi ;
- Possibilité d'évolution rapide par l'ajout de nouvelles fonctionnalités: Blog, Forum, News ;

III.2.1.3. Client FTP: Filezilla

Afin de transférer les fichiers sur le serveur, on a utilisé FileZilla, Ce logiciel client FTP nous servira à uploader (transférer) les fichiers sources d'installation de Joomla de notre micro-ordinateur vers le serveur de notre hébergeur, dans le cadre d'une installation distante.

III.2.2. Les outils utilisés

III.2.2.1. Le serveur web Apache

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

Apache 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. [14]

Le serveur HTTP Apache jouit d'une notoriété indiscutable dans le monde de l'Internet. D'après une étude sur les parts de marché des différents types de serveurs Web, Apache domine largement ses concurrents avec une utilisation qui dépasse 60% des serveurs web publics dans le monde. Ce succès est non seulement dû à sa gratuité, mais également à sa robustesse et à son extensibilité.

III.2.2.2. Le Serveur de bases de données MySQL

MySQL, le plus populaire des serveurs de bases de données SQL OpenSource, est développé et distribué par MySQL AB. MySQL est un système de gestion de bases de données relationnelles (SGBDR) libre fonctionnant sous diverses plates-formes telles que UNIX, Linux et Windows, et permettant de manipuler des instructions adressées à la base de données sous forme de requêtes SQL.

Le serveur de base de données MySQL est très souvent utilisé avec le langage de création de pages web dynamiques : PHP.

Les données manipulées par le site seront stockées dans une base de données MySQL. Le script PHP, se servira de commandes MySQL pour gérer et extraire des informations de la base de données afin d'engendrer les pages HTML qui seront interprétées par le navigateur.

La version utilisée pour notre application est la version 5.5.16.

Et pour l'administration de notre base de données, PhpMyAdmin 3.4.5 est utilisé.

III.2.2.3. PhpMyAdmin

PhpMyAdmin est un outil logiciel libre écrit en PHP, destiné à gérer l'administration de MySQL sur le World Wide Web. PhpMyAdmin supporte une large gamme d'opérations avec MySQL. Les opérations les plus fréquemment utilisés sont pris en charge par l'interface utilisateur (les bases de données de gestion, des tables, des champs, les relations, les index, les utilisateurs, les permissions, etc), alors que vous avez toujours la possibilité d'exécuter directement n'importe quelle instruction SQL.

Les fonctions de PhpMyadmin :

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

La figure qui suit illustre l'espace de travail de PhpMyAdmin

III.2.2.5. FPDF

Pour la création des documents destinés à l'impression, on a utilisé la bibliothèque FPDF qui est une classe PHP qui permet de générer des fichiers PDF en pur PHP, c'est-à-dire sans utiliser la librairie PDFlib. Le F de FPDF signifie Free : vous êtes libre de l'utiliser et de la modifier comme vous le souhaitez.

III.2.2.6. Toad for MySQL

Toad for MySQL est un outil de développement gratuit qui nous permet de créer et d'exécuter des requêtes rapidement, d'automatiser la gestion des objets de base de données, et de développer du code SQL de manière plus efficace. Il fournit des utilitaires pour comparer, extraire, et rechercher des objets, gérer des projets; import / export de données, et d'administrer la base de données. Toad for MySQL augmente considérablement la productivité et permet d'accéder à une communauté active d'utilisateurs.

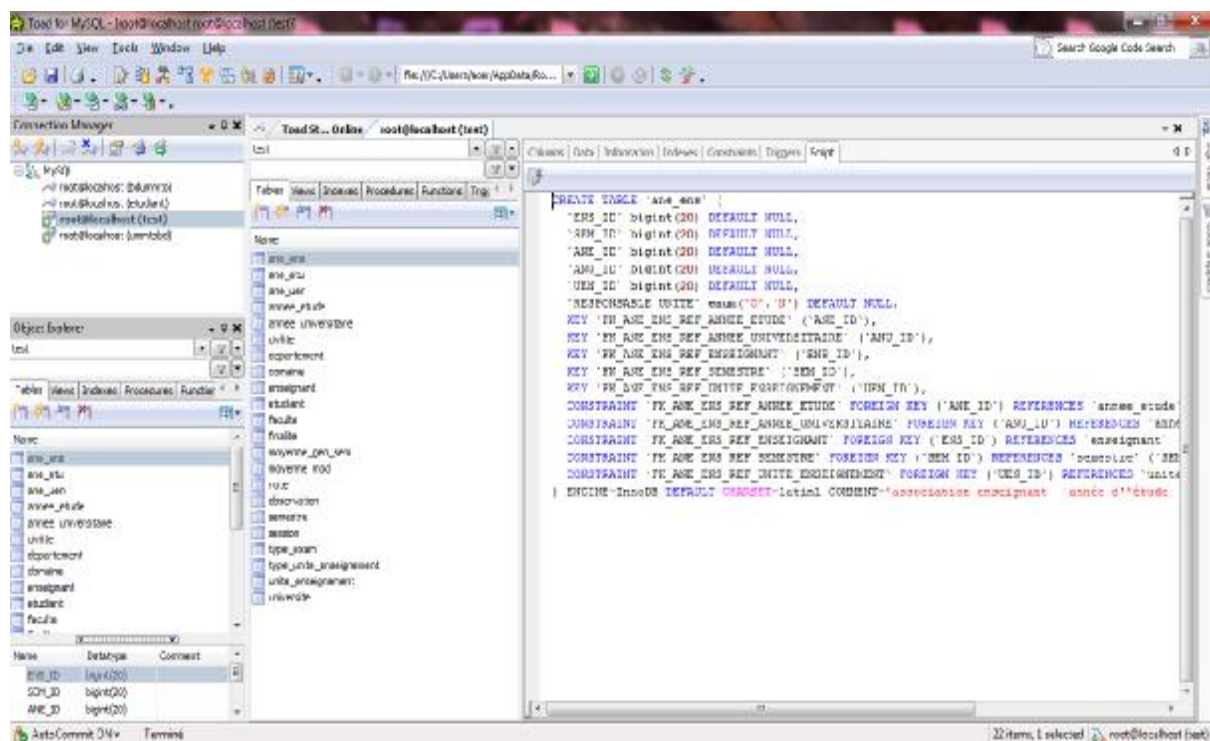


Figure 3.5 : l'interface de Toad for MySQL

III.2.1.7. Macromedia Dreamweaver 8

Macromedia Dreamweaver 8 est un éditeur HTML professionnel destiné à la conception, au codage et au développement de sites, de pages et d'applications Web. Quel que soit

l'environnement de travail utilisé (codage-manuel HTML ou environnement d'édition visuelle), Dreamweaver propose des outils qui nous aideront à créer des applications Web.

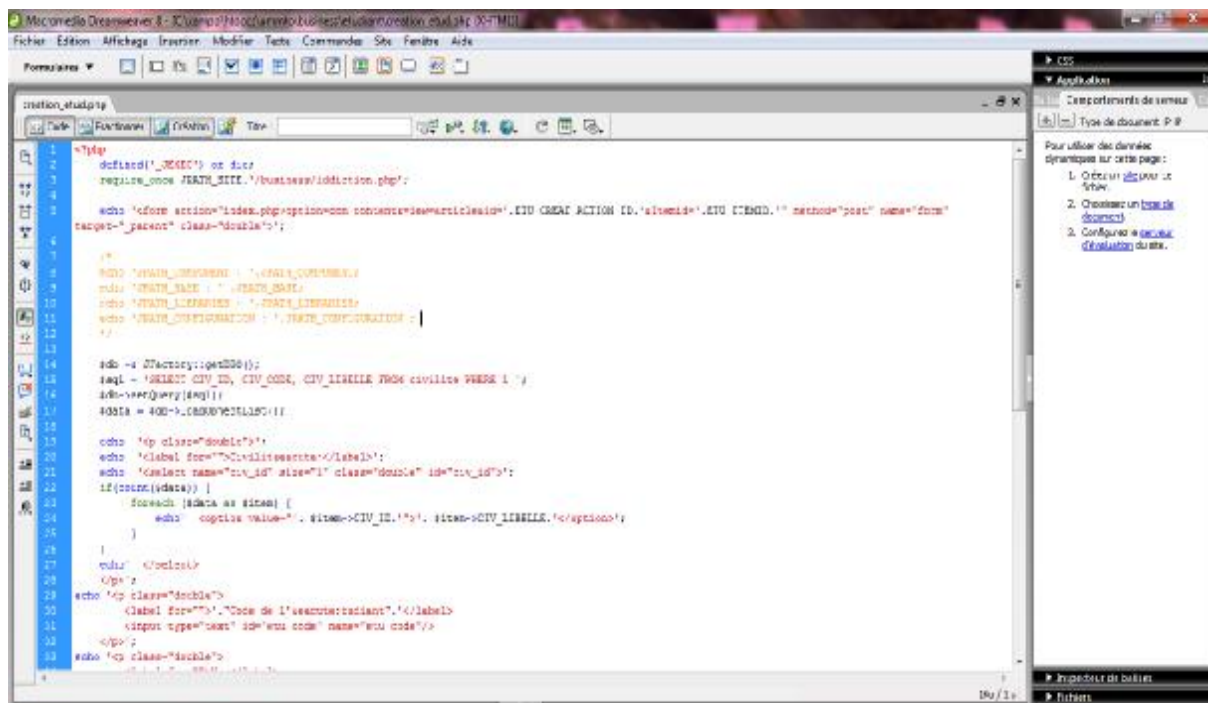


Figure 3.6 : l'interface de Macromedia Dreamweaver

III.2.3. Les langages d'implémentation utilisés :

Afin de réaliser notre projet, nous avons utilisé les langages décrits ci-dessous :

- **Le langage HTML**

Le langage html, décrit dans le premier chapitre, est utilisé pour structurer et mettre en page, le contenu des pages, d'inclure les ressources multimédia dont les images, des formulaires de saisie.

- **Les CSS**

CSS (Cascading Style Sheets, aussi appelées Feuilles de style) le rôle du CSS est de gérer l'apparence de la page web (agencement, positionnement, décoration, couleur, taille du texte...). Ce langage est venu compléter le HTML en 1996.

- **Le langage de script PHP**

Le PHP est actuellement le langage de script (côté serveur) le plus utilisé pour la réalisation de sites web dynamiques. Langage très complet, il permet de réaliser la plupart des applications, mais il est principalement employé pour mettre des bases de données en ligne dans les sites web. Les principaux atouts du PHP sont :

- La gratuité et la disponibilité du code source (PHP est distribué sous licence GNU GPL) ;
- La simplicité d'écriture de scripts qui sont exécutés directement sur le serveur, avant l'envoi de la page à l'internaute ;
- La possibilité d'inclure le script PHP au sein d'une page HTML ;
- La simplicité d'interfaçage avec des bases de données (de nombreux SGBD sont supportés, mais le plus utilisé avec ce langage est MySQL, un SGBD gratuit disponible sur les plateformes Unix, Linux, et Windows) ;
- La richesse fonctionnelle : PHP comporte plus de 1000 fonctions, parmi lesquelles les fonctions d'images, les protocoles Internet, etc.

- **Le langage de requête SQL :**

Pour communiquer avec une base de données, on a besoin de lui envoyer des commandes ou instructions appelées requêtes. Que ce soit pour la création, la suppression d'une table, la modification, l'insertion ou la sélection de données, le langage standard de requêtes est SQL.

SQL ou (*Standard QueryLanguage*) est un langage permettant d'interroger les bases de données de manière simple. Il est doté d'une syntaxe particulière que l'on doit respecter pour que la communication avec la base se passe au mieux. Son succès est dû essentiellement à sa simplicité et au fait qu'il énonce des requêtes en laissant le SGBD responsable de la stratégie d'exécution.

- **JavaScript**

JavaScript est un langage de programmation de scripts, principalement utilisé dans les pages web interactives. 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 équipés de constructeurs permettant de générer leurs propriétés.

III.3. Le modèle MVC(Model – View - Controller) [16]

Le Modèle Vue Contrôleur (MVC) est une architecture et une méthode de conception pour le développement d'applications logicielles qui sépare le modèle de données, l'interface utilisateur et la logique de contrôle.

Ce modèle d'architecture impose la séparation entre les données, les traitements et la présentation, ce qui donne trois parties fondamentales dans l'application finale : le modèle, la vue et le contrôleur.

Dans notre application, Joomla utilise le Modèle MVC pour concevoir et construire des extensions. Dans Joomla chaque élément du Modèle MVC est représenté par une classe abstraite du frameworkjoomla: JModel, JView, et JController de la bibliothèque 'joomla.application.component'.

- **Le Modèle**

Le modèle représente l'ensemble des données et des traitements de votre application. C'est l'ensemble des classes, librairies et fonctions nécessaires au fonctionnement de votre application, que ce soit l'enregistrement de vos données dans une base de données ou la modification de fichiers. D'une manière plus générale, le modèle définit le comportement général de votre composant. On appelle également cette partie la couche métier de votre application.

- **La Vue**

La vue est l'interface qui va interagir avec l'utilisateur. Dans Joomla, cela se traduit par du HTML renvoyé à votre navigateur. Dans la plupart des architectures MVC de type Web, l'utilisation de Javascript ou de tout autre langage exécuté côté client est une entorse au modèle MVC strict. Ainsi, il n'est pas rare d'avoir du code exécuté pour la vérification de données avant l'envoi au contrôleur. L'interface dispose d'un ensemble d'actions possibles se traduisant le plus souvent par une requête http envoyée à votre serveur web.

- **Le Contrôleur**

Le contrôleur est le lien entre le modèle et la vue. C'est lui qui est chargé de traiter les différents événements reçus par la vue (des requêtes http) et de les rediriger vers le traitement métier adéquate. Il ne réalise aucun traitement sur les données. Le cas échéant, le

contrôleurrécupère les données retournées par le modèle, se charge de les transférer à la vue avant de la renvoyer au client (navigateur).

Le schéma suivant nous montre encore le fonctionnement :

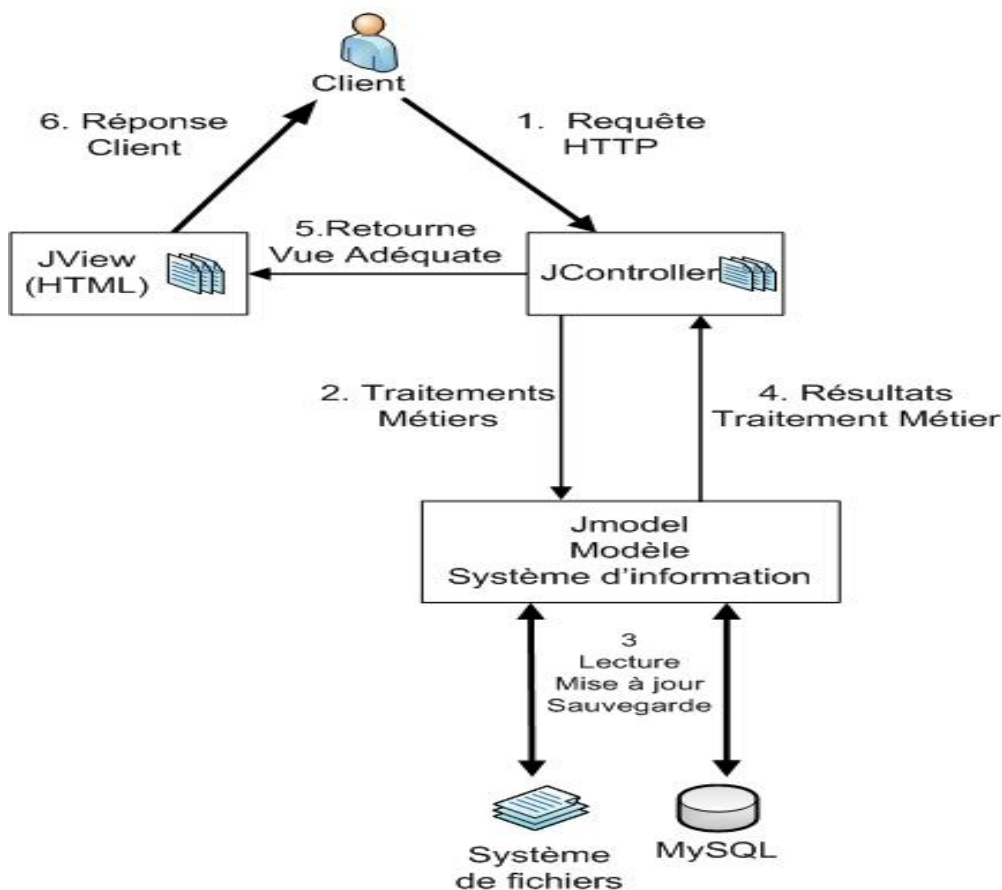


Figure 3.7 : le fonctionnement du modèle MVC Joomla

III.4. Principales interfaces de l'application

Les principales interfaces sont celles qui permettent d'agir sur les différentes informations qui concernent un projet donné.

La page d'accueil

The screenshot shows the homepage of the Faculty of Electrical Engineering and Informatics (FGEI) at the University of Mouloud Mammeri of Tizi-Ouzou. The page has a navigation menu at the top with links for ACCUEIL, FORMATIONS, EQUIPE PEDAGOGIQUE, MON COMPTE, and CONTACT. Below the menu is a banner with the faculty name and department name. The main content area includes a photo of the building and text describing the department's offerings and research.

Vous êtes ici : Accueil

Menu Membre

- » Services de la scolarité
- » Documentation
- » forum
- » Galeries photos

Ce site

- » Plan du Site
- » Profil

Faculté de Génie Electrique et d'Informatique
Département D'Electronique

Le département d'électronique est l'un des quatre départements de la faculté de Génie Electrique et d'Informatique (FGEI) de l'Université Mouloud Mammeri de Tizi-Ouzou (UMMTO). Il regroupe l'ensemble des filières de formation et un laboratoire de recherche destiné pour l'analyse des phénomènes aléatoires « Laboratoire d'Analyse et de Modélisation des Phénomènes Aléatoires » ou (LAMPA).

Le département assure les formations LMD (Licence, Master, Doctorat) suivantes :

- **Licence** : Le département reçoit des étudiants ayant passé avec succès les quatre semestres du domaine Sciences et techniques et après étude de fiches de vœux et sur la base des places pédagogiques disponibles.
 - Licence professionnelle spécialité électronique industrielle
 - Licence académique spécialité électronique
- **Master** : Le département propose une suite de Masters, dont l'entrée est sélective sur la base

Figure 3.8 : Page d'accueil de l'application

 Page d'authentification



Figure 3.9 : Interface pour l'authentification.

 L'espace étudiant

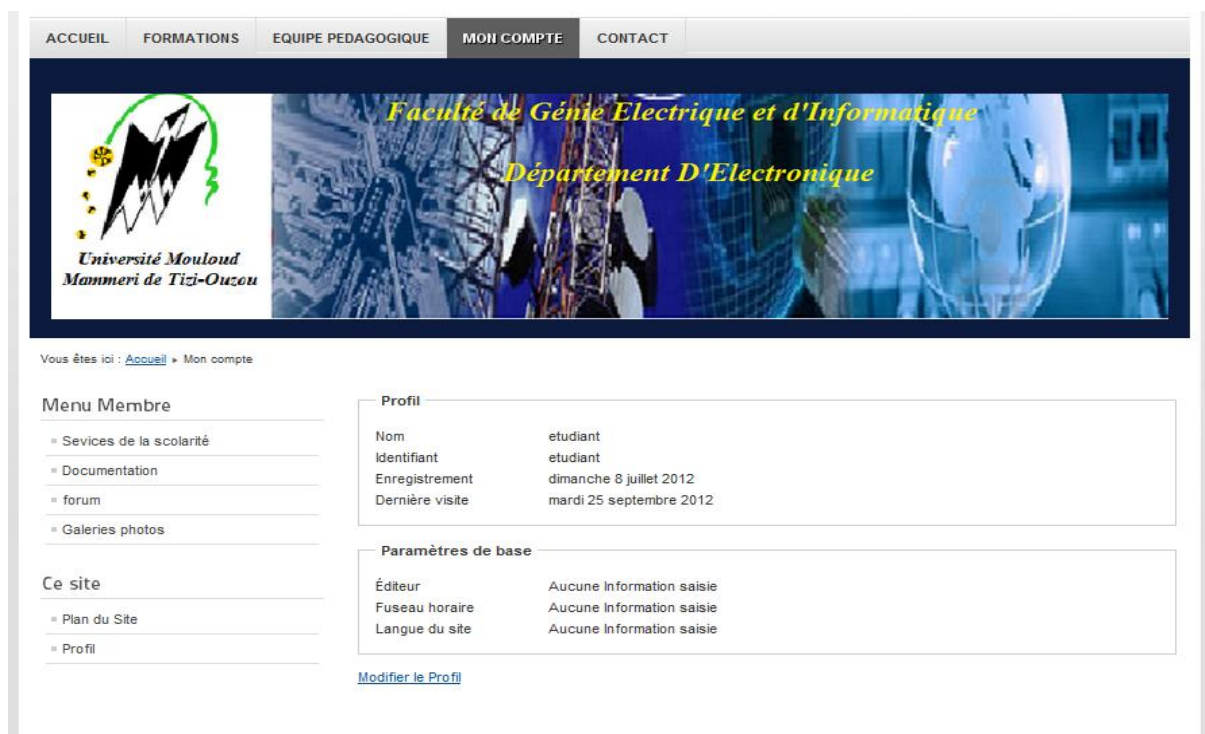


Figure 3.10 : Espace personnel d'un étudiant.



Figure 3.11 : La page informations et documents téléchargeables


 Espace enseignant



Figure 3.11 : L'ajout d'un document

Espace chef de département

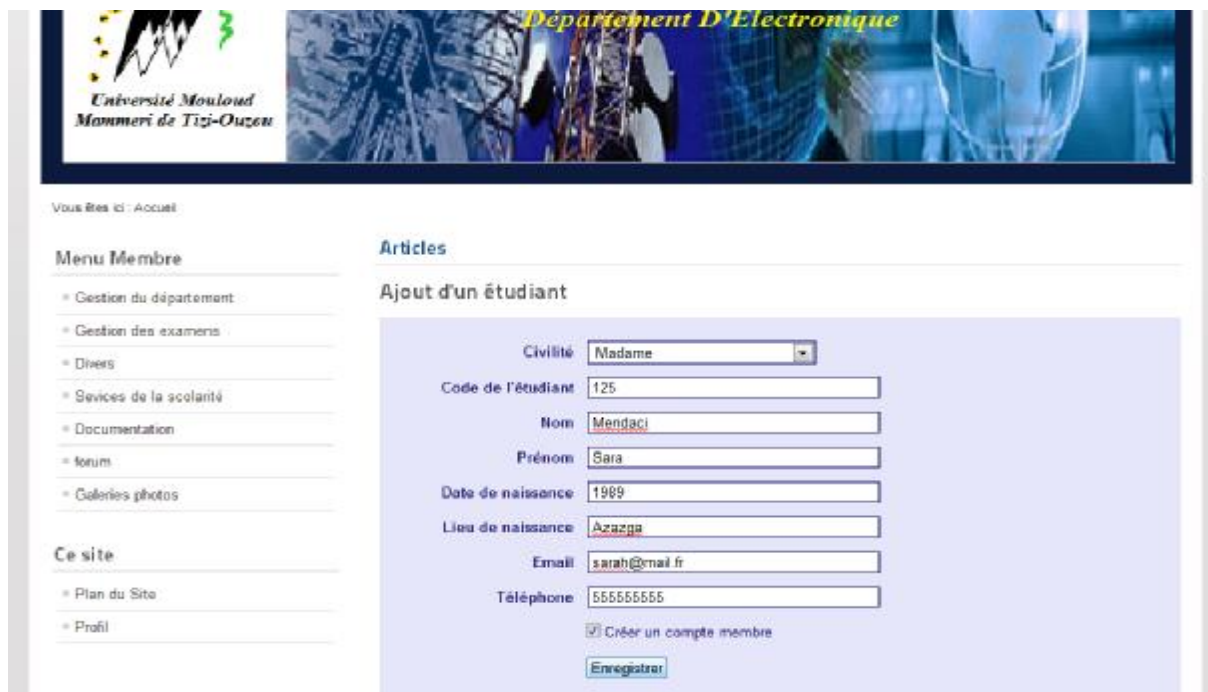


Figure 3.12 : L'ajout d'un étudiant

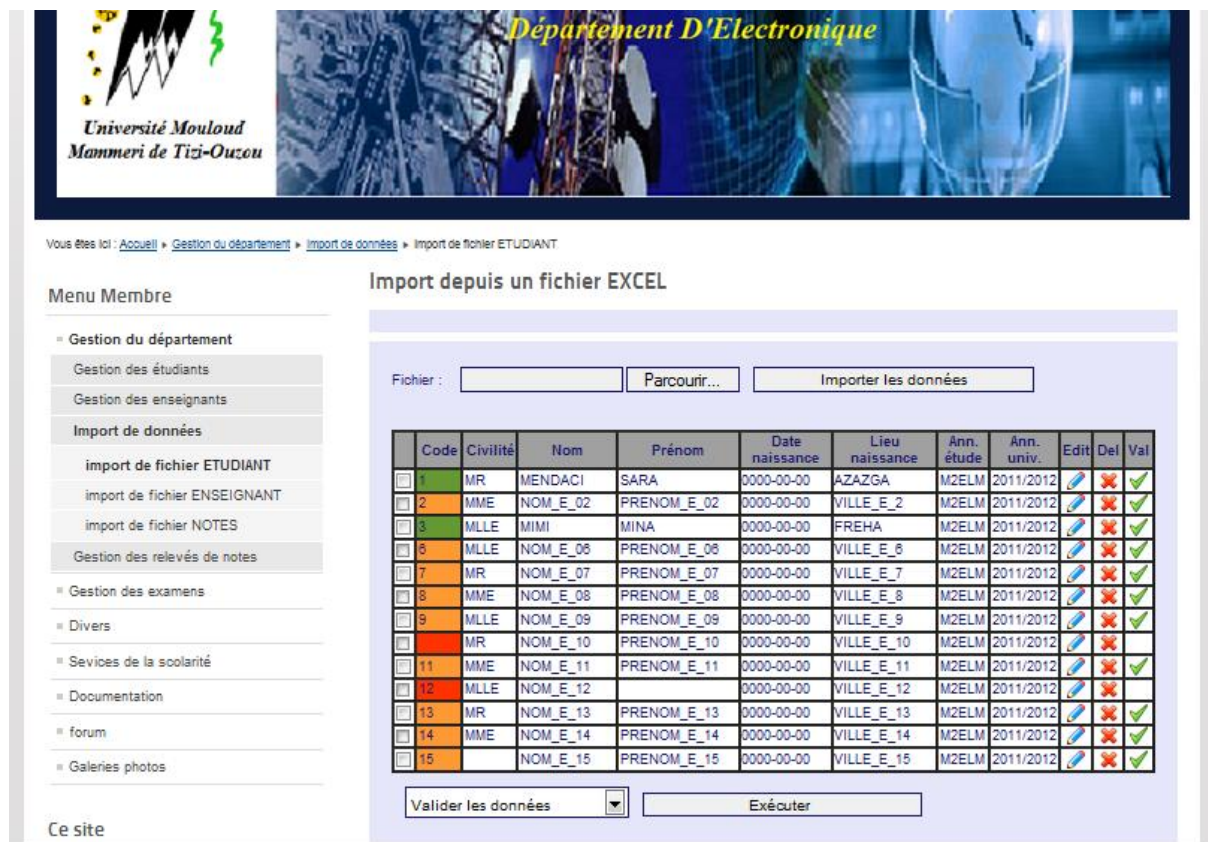


Figure 3.13 : Import de fichier étudiant

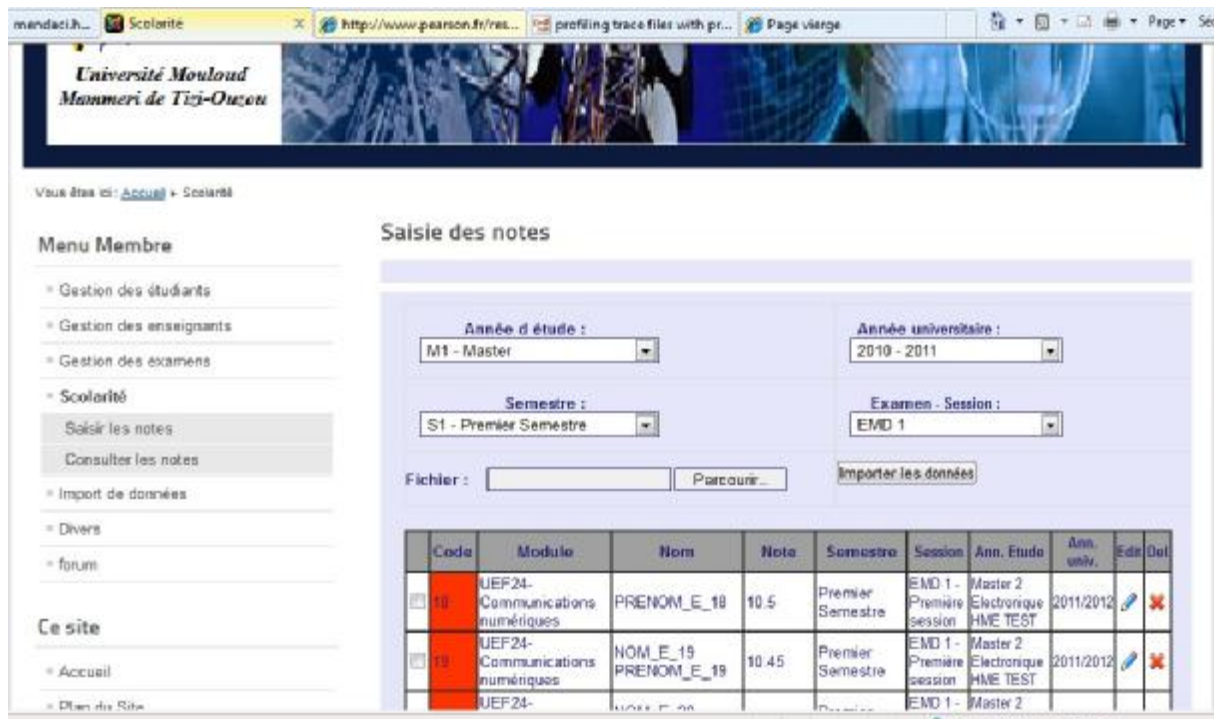


Figure 3.14 : L'ajout des notes des étudiants dans la base de données

L'interface forum

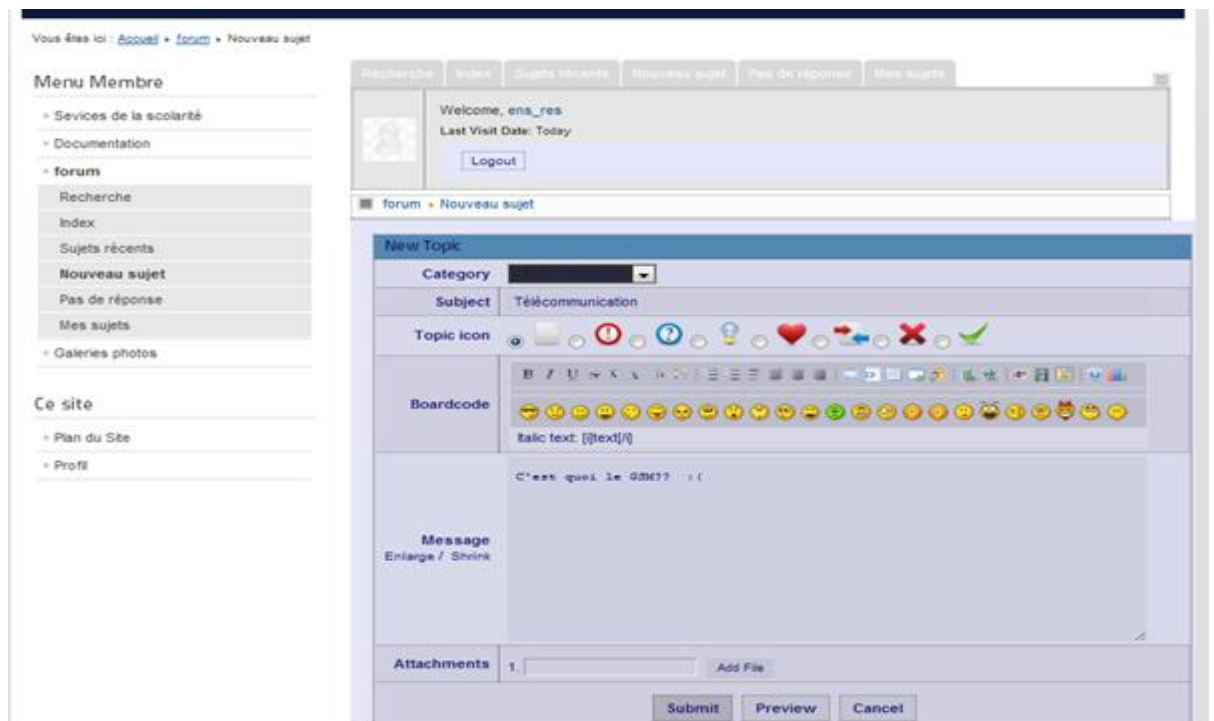


Figure 3.15 : L'ajout d'un nouveau sujet

L'interface galeries photos

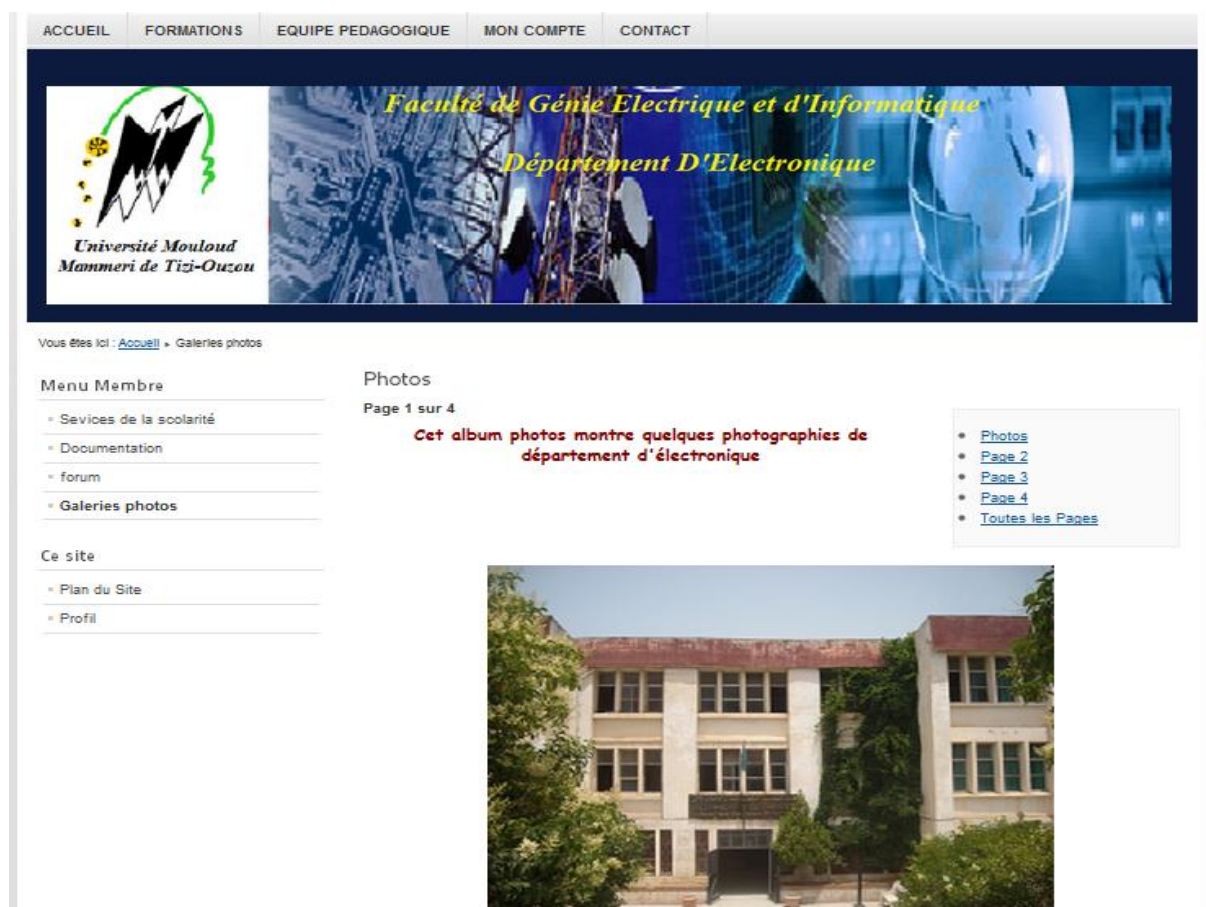


Figure 3.16 : Album photos

III.5. Conclusion

Dans ce chapitre nous avons, au premier lieu, présenté les différents outils et langages que nous avons utilisé pour implémenter notre application. Par la suite, nous avons présenté quelques interfaces de notre application.

Conclusion

L'objectif de notre projet a été de concevoir et de réaliser une application Web permettant la gestion du département d'Electronique de la faculté de Génie Electrique et d'Informatique de l'UMMTO, afin d'améliorer son système d'information pour pallier aux problèmes de la gestion classique.

L'application à laquelle on a abouti, répond à plusieurs de ces fonctionnalités, néanmoins quelques améliorations et extensions peuvent aussi être ajoutées. En effet, ce projet m'a permis d'apprendre énormément de choses sur le développement WEB. Il m'a donné la chance de toucher à divers langages tels que PHP, Javascript, HTML, CSS, SQL, UML ... Ainsi, il m'a permis de me familiariser avec un certain nombre d'outils de développement tels que Xampp et le CMS Joomla.

Ce Projet fut pour moi une expérience très enrichissante car il m'a permis d'améliorer mes compétences dans le domaine du développement Web.

Annexe A

Généralités sur l'UML

1. Introduction

Dans cette partie, on va définir et décrire en détail le langage utilisée pour la modélisation dans le deuxième chapitre de notre application.

2. Présentation d'UML

UML (sigle désignant l'Unified Modeling Language ou « langage de modélisation unifié ») est un langage de modélisation graphique à base de pictogrammes. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ». UML est couramment utilisé dans les projets logiciels.

UML représente le standard de modélisation objet le plus répandu et le plus utilisé aujourd'hui. Il s'articule autour de treize types de diagrammes chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont répartis en deux grands groupes ; des Diagrammes structurels et comportementaux.

3. La modélisation UML

UML fournit une panoplie d'outils permettant de présenter l'ensemble des éléments du monde objet (classe, objet,...etc.) ainsi que les liens qui les relient, il s'articule autour de treize types de diagrammes, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel.

4. Les diagrammes UML

La version 2 d'UML propose 13 types de diagrammes standards représentant autant de « vues » d'un système d'information. Ces diagrammes se répartissent en deux catégories : six représentent la structure statique de l'application (diagramme de classes, d'objets, de composants, de déploiement, de package, et de structure composite) ; sept représentent son comportement dynamique (diagramme de cas d'utilisation, d'états-transitions, d'activités, de séquence, de communication, d'interaction, de temps).

Les différents diagrammes sont représentés dans la figure suivante :

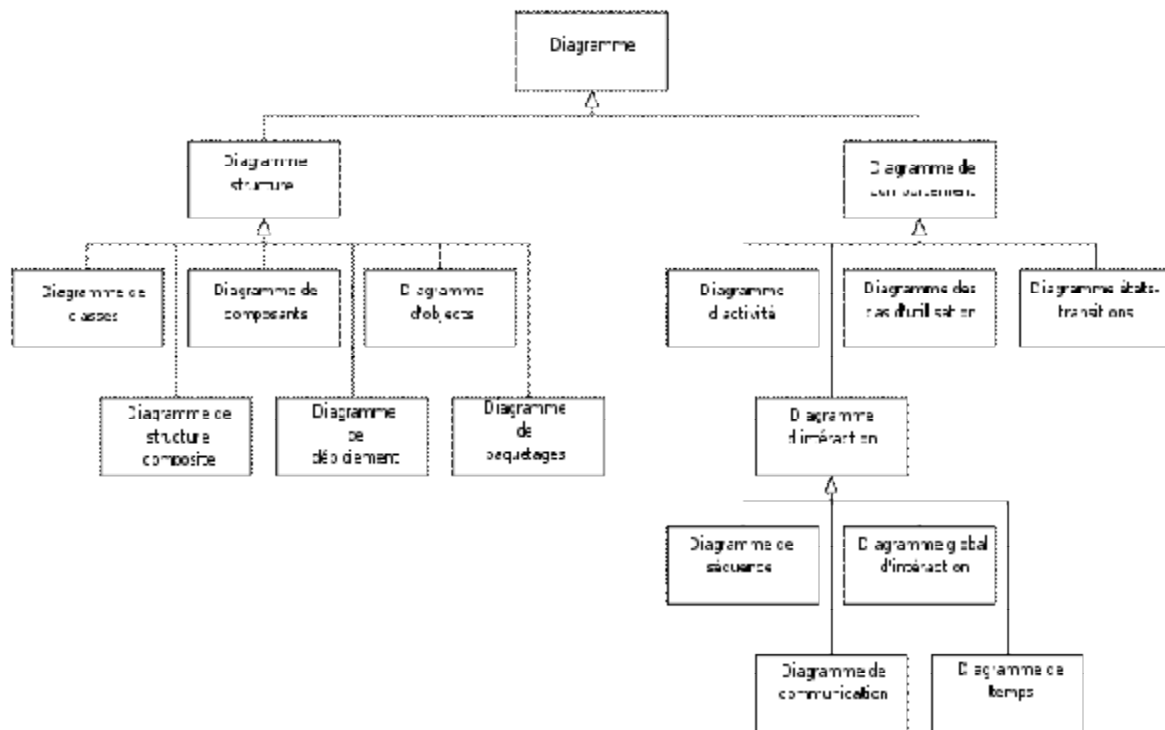


Figure : les diagrammes d'UML

4.1. Diagrammes Structurels ou Diagrammes statiques (Structure Diagram)

- Ø **Diagramme de classes (Class diagram)** : il représente les classes intervenant dans le système.
- Ø **Diagramme d'objets (Object diagram)** : il sert à représenter les instances de classes (objets) utilisées dans le système.
- Ø **Diagramme de composants (Component diagram)** : il permet de montrer les composants du système d'un point de vue physique, tels qu'ils sont mis en œuvre (fichiers, bibliothèques, bases de données...).
- Ø **Diagramme de déploiement**: il sert à représenter les éléments matériels (ordinateurs, périphériques, réseaux, systèmes de stockage...) et la manière dont les composants du système sont répartis sur ces éléments matériels et interagissent avec eux.
- Ø **Diagramme des paquetages (Package Diagram)**: permet de représenter la hiérarchie des paquetages du projet, leur organisation et leurs interdépendances, simplifie les diagrammes (donc plus simple à comprendre)
- Ø **Diagramme de structures composites (Composite Structure Diagram)** : permet de décrire la structure interne d'un objet complexe lors de son exécution (c'est à dire,

décrire l'exécution du programme), dont ses points d'interaction avec le reste du système.

4.2. Diagrammes Comportementaux ou Diagrammes dynamiques

- Ø **Diagramme des cas d'utilisation (Use case diagram):** il décrit les possibilités d'interaction entre le système et les acteurs, c'est-à-dire toutes les fonctionnalités que doit fournir le système.
- Ø **Diagramme états-transitions (State Machine Diagram):** il montre la manière dont l'état du système (ou de sous-parties) est modifié en fonction des événements du système.
- Ø **Diagramme d'activité (Activity Diagram):** variante du diagramme d'états-transitions, il permet de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallélisables (multithreads ou multi-processus).
- Ø **Diagramme d'interactions (Interaction Diagram):**
 - **Diagramme de séquence (Sequence Diagram):** la représentation séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou des acteurs.
 - **Diagramme de communication (Communication Diagram):** la représentation simplifiée d'un diagramme de séquence se concentrant sur les échanges de messages entre les objets.
 - **Diagramme global d'interaction (Interaction Overview Diagram):** variante du diagramme d'activité où les nœuds sont des interactions, permet d'associer les notations du diagramme de séquence à celle du diagramme d'activité, ce qui permet de décrire une méthode complexe.
- Ø **Diagramme de temps (Timing Diagram):** la représentation des interactions où l'aspect temporel est mis en valeur; il permet de modéliser les contraintes d'interaction entre plusieurs objets, comme le changement d'état en réponse à un événement extérieur.

5. Quelques définitions de base

Ø **Un acteur** : un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système.

Ø **Cas d'utilisation** : un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service.

Ø **Un stéréotype** : Un stéréotype représente une variation d'un élément de modèle existant.

Ø **Un scénario** : Il représente une succession particulière d'enchaînement qui s'exécute du début à la fin du cas d'utilisation, un enchaînement étant l'unité de description séquences d'actions.

Ø Relations entre cas d'utilisation

La relation d'inclusion : un cas A est inclus dans un cas B si le comportement décrit par le cas A est inclus dans le comportement du cas B : on dit alors que le cas B dépend de A. Cette dépendance est symbolisée par le stéréotype « *include* ».

La relation d'extension : on dit qu'un cas d'utilisation A étend un cas d'utilisation B lorsque le cas d'utilisation A peut être appelé au cours de l'exécution du cas d'utilisation B. Exécuter B peut éventuellement entraîner l'exécution de A : contrairement à l'inclusion, l'extension est optionnelle. Cette dépendance est symbolisée par le stéréotype « *extend* ».

La relation de généralisation : un cas A est une généralisation d'un cas B si B est un cas particulier de A. Cette relation de généralisation/ spécialisation est présentée dans la plupart des diagrammes UML et se traduit par le concept d'héritage dans les langages orientés objet.

Annexe B

Le modèle physique de données

Après la modélisation du site avec les différents diagrammes qu'on a vu dans le deuxième chapitre, le fonctionnement de l'application nécessite certaines données. Une base de données était créée pour ça, qui comporte des tables qui sont décrites dans le tableau suivant :

| Nom des tables Croissant | Nom des champs | Signification | Type de donnée |
|--------------------------|-------------------|------------------------------------|----------------|
| ane_ens | ENS_ID | Identifiant enseignant | bigint(20) |
| | SEM_ID | Identifiant semestre | bigint(20) |
| | ANE_ID | Identifiant année d'étude | bigint(20) |
| | ANU_ID | Identifiant année universitaire | bigint(20) |
| | UEN_ID | Identifiant unité d'enseignement | bigint(20) |
| | RESPONSABLE_UNITE | Responsable d'unité d'enseignement | enum('O','N') |
| ane_etu | ANE_ID | Identifiant année d'étude | bigint(20) |
| | ETU_ID | Identifiant étudiant | bigint(20) |
| | ANU_ID | Identifiant année universitaire | bigint(20) |
| ane_uen | SEM_ID | Identifiant semestre | bigint(20) |
| | ANE_ID | Identifiant année d'étude | bigint(20) |
| | ANU_ID | Identifiant année universitaire | bigint(20) |
| | UEN_ID | Identifiant unité d'enseignement | bigint(20) |
| | CREDIT | Crédit | int(11) |
| | COEFFICIENT | Coefficient | int(11) |
| annee_etude | ANE_ID | Identifiant années d'étude | bigint(20) |
| | ANE_CODE | Code de l'année d'étude | varchar(30) |
| | ANE_LIBELLE | Libellé de l'année d'étude | varchar(100) |
| | FIN_ID | Identifiant finalité | bigint(20) |

| | | | |
|----------------------------|-----------------|---|---------------|
| | DEP_ID | Identifiant département | bigint(20) |
| annee_universitaire | ANU_ID | Identifiant année universitaire | bigint(20) |
| | ANU_ANNEE_DEBUT | Année de début de l'année universitaire | varchar(4) |
| | ANU_ANNEE_FIN | Année de fin de l'année universitaire | varchar(4) |
| | ANU_MOIS_DEBUT | Mois de début de l'année universitaire | varchar(2) |
| | ANU_MOIS_FIN | Mois de fin de l'année universitaire | varchar(2) |
| Civilite | CIV_ID | Identifiant civilité | bigint(20) |
| | CIV_CODE | Code de civilité | varchar(20) |
| | CIV_LIBELLE | Libellé de civilité | varchar(100) |
| Departement | DEP_ID | Identifiant département | bigint(20) |
| | DEP_CODE | Code du département | varchar(20) |
| | DEP_LIBELLE | Libellé du département | varchar(200) |
| | FAC_ID | Identifiant faculté | bigint(20) |
| Document | DOC_ID | Identifiant document | bigint(20) |
| | DOC_TITRE | Titre de document | varchar(100) |
| | DOC_ANNEE | Année d'étude de document | varchar(100) |
| | DOC_NOM | Nom de documenta | varchar(200) |
| | DOC_DESCRIPTION | Description de document | varchar(1000) |
| Domaine | DOM_ID | Identifiant domaine d'étude | bigint(20) |
| | DOM_CODE | Code du domaine d'étude | varchar(20) |
| | DOM_LIBELLE | Libellé du domaine d'étude | varchar(200) |
| | UNI_ID | Identifiant université | bigint(20) |
| Enseignant | ENS_ID | Identifiant enseignant | bigint(20) |
| | ENS_CODE | Code de l'enseignant | varchar(20) |
| | ENS_NOM | Nom de l'enseignant | varchar(100) |
| | ENS_PRENOM | Prénom de l'enseignant | varchar(100) |
| | ENS_TELEPHONE | Numéro de tél de l'enseignant | varchar(20) |

| | | | |
|------------------------|--------------------|-----------------------------------|------------------------|
| | ENS_MAIL | Adresse mail de l'enseignant | varchar(1000) |
| | ENS_DATE_NAISSANCE | Date de naissance de l'enseignant | date |
| | CIV_ID | Identifiant civilité | bigint(20) |
| | USER_ID | Identifiant user | int(11) |
| Etudiant | ETU_ID | Identifiant etudiant | bigint(20) unsigned |
| | CIV_ID | Identifiant civilité | bigint(20) |
| | ETU_CODE | Code de l'étudiant | varchar(20) |
| | ETU_NOM | Nom de l'étudiant | varchar(100) |
| | ETU_PRENOM | Prénom de l'étudiant | varchar(100) |
| | ETU_DATE_NAISSANCE | Date de naissance de l'étudiant | date |
| | ETU_LIEU_NAISSANCE | Lieu de naissance de l'étudiant | varchar(50) |
| | ETU_MAIL | Adresse mail de l'étudiant | varchar(1000) |
| | ETU_TELEPHONE | Numéro de téléphone de l'étudiant | varchar(20) |
| | USER_ID | Identifiant user | int(11) |
| Faculte | FAC_ID | Identifiant faculté | bigint(20) |
| | FAC_CODE | Code de la faculté | varchar(20) |
| | FAC_LIBELLE | Libellé de la faculté | varchar(200) |
| | UNI_ID | Identifiant université | bigint(20) |
| | DOM_ID | Identifiant domaine | bigint(20) |
| Finalite | FIN_ID | Identifiant finalité | bigint(20) |
| | FIN_CODE | Code finalité | varchar(30) |
| | FIN_LIBELLE | Libellé finalité | varchar(100) |
| import_etudiant | ETU_ID | Identifiant etudiant | bigint(20) |
| | CIV_CODE | Code civilité | varchar(20) |
| | ETU_CODE | Code étudiant | varchar(20) |
| | ETU_NOM | Nom de l'étudiant | varchar(100) |

| | | | |
|--|--------------------|--|---------------|
| | ETU_PRENOM | Prénom de l'étudiant | varchar(100) |
| | ETU_DATE_NAISSANCE | Date de naissance de l'étudiant | date |
| | ETU_LIEU_NAISSANCE | Lieu de naissance de l'étudiant | varchar(50) |
| | ETU_MAIL | Adresse mail de l'étudiant | varchar(1000) |
| | ETU_TELEPHONE | Numéro de téléphone de l'étudiant | varchar(20) |
| | ETU_IMPORT | Importer étudiant | varchar(1) |
| | USER_ID | Identifiant user | bigint(20) |
| | ANE_CODE | Code de l'année d'étude | varchar(20) |
| | ANU_CODE | Code de l'année universitaire | varchar(20) |
| | IMP_ERR | Erreur import | varchar(50) |
| moyenne_gen_sem (la moyenne générale du semestre) | MGS_ID | Identifiant moyenne générale du semestre | bigint(20) |
| | ANE_ID | Identifiant année d'étude | bigint(20) |
| | ANU_ID | Identifiant année universitaire | bigint(20) |
| | SEM_ID | Identifiant semestre | bigint(20) |
| | ETU_ID | Identifiant étudiant | bigint(20) |
| | MGS_VALEUR | La valeur de la moyenne générale du semestre | decimal(2,2) |
| moyenne_mod (la moyenne du module) | MOM_ID | Identifiant moyenne du module | bigint(20) |
| | ANU_ID | Identifiant année universitaire | bigint(20) |
| | ANE_ID | Identifiant année d'étude | bigint(20) |
| | ETU_ID | Identifiant étudiant | bigint(20) |
| | SEM_ID | Identifiant semestre | bigint(20) |
| | SES_ID | Identifiant session | bigint(20) |
| | UEN_ID | Identifiant unité d'enseignement | bigint(20) |
| | OBS_ID | Identifiant observation | bigint(20) |
| | MOM_VALEUR | La valeur de la moyenne du module | decimal(2,2) |

| | | | |
|--------------------------------|----------------|---|--------------|
| Note | NOT_ID | Identifiant note | bigint(20) |
| | TEX_ID | Identifiant type d'examen | bigint(20) |
| | UEN_ID | Identifiant unité d'enseignement | bigint(20) |
| | ANU_ID | Identifiant année universitaire | bigint(20) |
| | SEM_ID | Identifiant semestre | bigint(20) |
| | ETU_ID | Identifiant étudiant | bigint(20) |
| | NOT_VALEUR | La valeur de la note | decimal(2,2) |
| observation | OBS_ID | Identifiant observation | bigint(20) |
| | OBS_CODE | Code observation | varchar(30) |
| | OBS_LIBELLE | Libellé observation | varchar(100) |
| semestre | SEM_ID | Identifiant semestre | bigint(20) |
| | SEM_CODE | Code du semestre | varchar(30) |
| | SEM_LIBELLE | Libellé du semestre | varchar(50) |
| | SEM_MOIS_DEBUT | Mois de début du semestre | varchar(20) |
| | SEM_MOIS_FIN | Mois de fin du semestre | varchar(20) |
| session | SES_ID | Identifiant session | bigint(10) |
| | SES_CODE | Code session | varchar(20) |
| | SES_LIBELLE | Libellé session | varchar(100) |
| type_exam | TEX_ID | Identifiant type d'examen | bigint(20) |
| | SES_ID | Identifiant session | bigint(20) |
| | TEX_CODE | Code de type d'examen | varchar(30) |
| | TEX_LIBELLE | Libellé de type d'examen | varchar(100) |
| type_unite_enseignement | TUE_ID | Identifiant type d'unité d'enseignement | bigint(20) |
| | TUE_CODE | Code de type d'unité d'enseignement | varchar(15) |
| | TUE_LIBELLE | Type d'unité d'enseignement | varchar(50) |
| unite_enseignement | UEN_ID | Identifiant unité d'enseignement | bigint(20) |
| | UEN_CODE | Code de l'unité d'enseignant | varchar(20) |

| | | | |
|-------------------|-------------|---|--------------|
| | UEN_LIBELLE | Libellé de l'unité d'enseignement | varchar(20) |
| | TUE_ID | Identifiant type d'unité d'enseignement | bigint(20) |
| universite | UNI_ID | Identifiant université | bigint(20) |
| | UNI_CODE | Code de l'université | varchar(20) |
| | UNI_LIBELLE | Libelle de l'université | varchar(150) |

Annexe C

Présentation du CMS Joomla

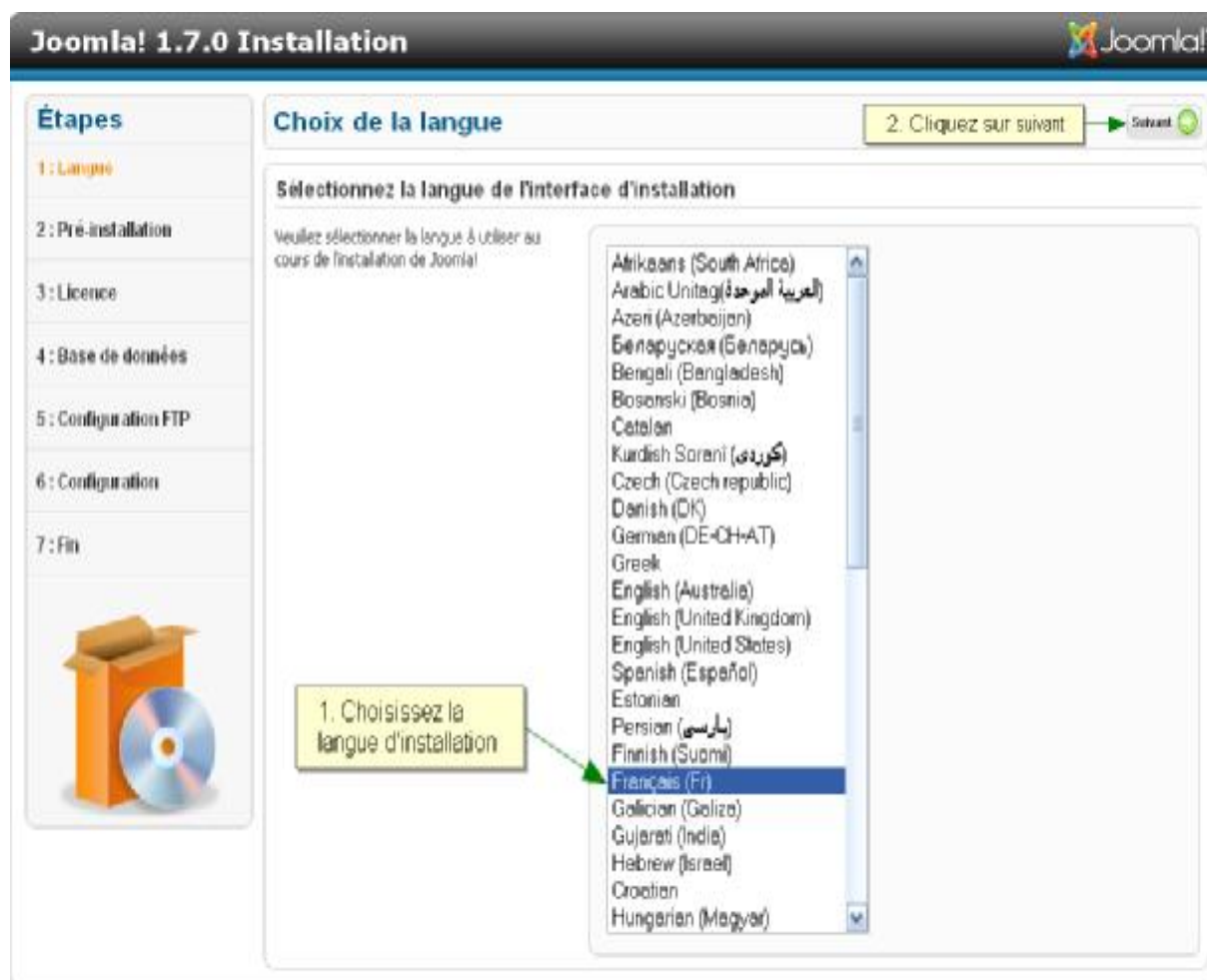
1.Présentation

Joomla (prononcé "djoumla") est une transcription phonétique du mot swahilien "Jumla". Sa traduction signifie "le tout" ou "tout en entier" et sous-entend la communauté des développeurs.

Dans notre application nous avons utilisé le CMS Joomla version 1.7

2.Installation de Joomla 1.7

Les 7 étapes de l'installation de Joomla! 1.7 :

 **Choix de la langue d'installation**

Pré-installation

Étapes

- Langue
- Pré-installation**
- Licence
- Base de données
- Configuration FTP
- Configuration
- Fin

Pré-installation [Vérifier à nouveau] [Précédent] [Suivant]

Vérification de la pré-installation pour Joomla! 1.7.0 Stable [Ember] 19-Jul-2011 14:00 GMT

Si certains de ces éléments ne sont pas en forme (marqués **Non**) veuillez faire en sorte de les corriger en contactant votre hébergeur.

Si vous rencontrez des problèmes, vous devez désinstaller les problèmes Joomla! et réinstaller avec des conditions initiales en utilisant Joomla!

Si vous ne possédez pas les droits d'écriture d' fichiers sur le dossier racine de votre site, le fichier configuration.php ne pourra pas être créé. Dans ce cas, créez-le à l'aide d'un simple éditeur de texte, laissez-le vide et ajoutez-le à la racine de votre site. Choisissez la langue Joomla! dans le temps de l'installation et de la configuration de Joomla!. Pour des raisons de sécurité, il est recommandé de faire la mise à jour de Joomla! à l'aide de la plus récente de cette version.

Vous pouvez également télécharger l'installation en utilisant ce processus, le contenu est affiché pour créer le fichier et le placer à la racine de votre site.

| | |
|---|-----|
| Version de PHP >= 5.2.4 | Oui |
| Support de la compression GZIP | Oui |
| Support de MySQL | Oui |
| Support de MySQL | Oui |
| Directive missing language par défaut | Oui |
| Directive missing translation d'attributs | Oui |
| INI Fileset Support | Oui |
| Support JSON | Oui |
| Configuration non modifiable | Oui |

Paramètres recommandés :

Ces paramètres PHP sont recommandés pour une compatibilité optimale avec Joomla!

Si ils ne sont pas tous en vert, Joomla! pourra tout de même fonctionner mais des extensions risquent de ne pas être utilisables.

| Directive | Recommandé | Actuel |
|---------------------------|------------|--------|
| Safe Mode | Off | Off |
| Afficher les erreurs | Off | Off |
| Transfer de fichiers | On | On |
| Write/execute/permissions | Off | Off |
| Register Globals | Off | Off |
| Output Buffering | Off | Off |
| Session Auto Start | Off | Off |

Configuration PHP recommandée (voir dans cet exemple)

Licence

Étapes

- Langue
- Vérification Pré-installation
- Licence**
- Base de données
- Configuration FTP
- Configuration
- Fin

Licence [Précédent] [Suivant]

Joomla! est sous licence GNU

Licence Publique Générale GNU :

Table des contenus

- GNU GPL - GNU LGPL
- Préambule
- TERMS AND CONDITIONS FOR COPYING AND DISTRIBUTION
- HOW TO APPLY THESE TERMS AND CONDITIONS

GNU GENERAL PUBLIC LICENSE

Version 2, June 1989

Copyright © 1989, 1991, 1993 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation.

Preamble

This license is intended to assure your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software - to make sure the software is free for all users. This General Public License applies to most of the Free Software Foundation's software and to any program which carries over from existing GNU software. Other Free Software Foundation software is licensed by the GNU Library General Public License instead. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you can make copies for others to use, that you are free to change the software to suit your needs, and that you are free to help others do the same.

You are free to copy or distribute verbatim copies of this license provided that you copy only the license text and the copyright notice, and you preserve this license and its copyright notice on all copies.

You are free to copy or distribute verbatim copies of this license provided that you copy only the license text and the copyright notice, and you preserve this license and its copyright notice on all copies.

You are free to copy or distribute verbatim copies of this license provided that you copy only the license text and the copyright notice, and you preserve this license and its copyright notice on all copies.

Configuration de la base de données

The screenshot shows the Joomla! 1.7.x installation interface for the 'Configuration de la base de données' step. The left sidebar lists the installation steps, with '4: Base de données' highlighted. The main content area is titled 'Paramètres de connexion' and contains a 'Paramètres de base' form. Annotations with colored boxes and arrows point to specific fields:

- Type de base de données:** Set to 'MySQL'. A note explains: 'Type de base de données (uniquement MySQL ou Microsoft SQL Server)'. A red box highlights the dropdown menu.
- Nom de serveur:** Set to 'localhost'. A note explains: 'Nom de serveur de base de données pour lequel est installé le Joomla!'. A purple box highlights the field.
- Nom d'utilisateur:** A note explains: 'Nom d'utilisateur utilisé pour accéder à la base de données MySQL ou Microsoft SQL Server'. A green box highlights the field.
- Mot de passe:** A note explains: 'Mot de passe utilisé pour accéder à la base de données MySQL ou Microsoft SQL Server. Un caractère spécial de passe est autorisé'. A green box highlights the field.
- Connexion à la base de données:** A note explains: 'Nom de la base de données pour Joomla!, ainsi que tous autres composants Joomla!'. A red box highlights the field.
- Préfixe des tables:** Set to 'joomla_'. A note explains: 'Le préfixe des tables est utilisé automatiquement pour créer la table de données Joomla!'. A red box highlights the field.
- Installation précédente:** Radio buttons for 'Sauvegarder' and 'Supprimer'. A note explains: 'Si vous sautez les tables, elles seront reconstruites à partir de la table "Joomla!". Assurez-vous que ce préfixe n'est pas utilisé pour les tables d'un autre système'. A blue box highlights the options.

A yellow box at the bottom states: 'Si vous rencontrez des problèmes lors de l'installation de Joomla!'. The bottom right corner shows 'Précédent' and 'Suivant' buttons.

Configuration FTP

The screenshot shows the Joomla! 1.7.x installation interface for the 'Configuration FTP' step. The left sidebar lists the installation steps, with '5: Configuration FTP' highlighted. The main content area is titled 'Configuration FTP (Optionnel - La plupart des utilisateurs peuvent passer cette étape - Appuyez sur "Suivant" pour continuer)'. The 'Paramètres de base' form has 'Activer la couche FTP si besoin' checked. Annotations with colored boxes and arrows point to specific fields:

- Activer la couche FTP:** A dropdown menu set to 'Oui'. A note explains: 'Attention: l'activation de cette option aura des effets sur les performances de votre Joomla! et peut entraîner des problèmes de sécurité'. A yellow box highlights the dropdown.
- Nom d'utilisateur FTP:** A note explains: 'Attention: l'activation de cette option aura des effets sur les performances de votre Joomla! et peut entraîner des problèmes de sécurité'. A yellow box highlights the field.
- Adresser IP:** Set to '127.0.0.1'. A note explains: 'Adresse IP ou nom de domaine de votre serveur Joomla!'. A yellow box highlights the field.
- Port FTP:** Set to '21'. A note explains: 'Port FTP par défaut'. A yellow box highlights the field.

The bottom right corner shows 'Précédent' and 'Suivant' buttons.

3. Les notions de base

Avant de se lancer dans la mise en œuvre de Joomla! et sa configuration, j'ai préféré de vous présenter Joomla avec son vocabulaire et quelques concepts de base.

3.1. La terminologie Joomla!

Voici une liste des termes les plus fréquemment utilisés dans la planète Joomla:

- **Article** : un article est une unité de contenu. Il comprend généralement du texte, des images et des liens ; il a certaines caractéristiques comme un titre, un auteur, une date de publication et tout un tas de paramètres qui seront décrits plus loin.
- **Catégorie** : une catégorie sera le point de rassemblement d'une série d'articles appartenant à une même thématique. Il est possible de créer des catégories à l'intérieur d'une catégorie.
- **Menu** : c'est une liste d'éléments, disposés de façon verticale ou horizontale selon le module choisi pour l'afficher et sa configuration. L'appui sur un élément du menu provoque l'affichage d'une page avec ses modules et ses composants.
- **Page d'accueil** : c'est la première page que voit un visiteur lorsqu'il saisit le nom de votre site.
- **Administration** : la partie administration – ou backend - est l'arrière-boutique de votre site ; l'interface d'administration va permettre de créer et mettre à jour vos articles mais aussi de gérer tout votre site.
- **Site** : La partie Site - ou frontend - c'est la boutique, ce que voient les visiteurs qui viennent sur votre site.
- **Cache** : pour rendre plus rapide l'affichage des pages de votre site, les éléments les plus souvent demandés (logos, images, page d'accueil) sont stockés dans un répertoire intermédiaire, encore appelé cache. Lorsqu'un utilisateur veut consulter une page comprenant un élément en cache, Joomla! n'a plus besoin d'aller le chercher dans la base de données ou un répertoire du site, il le prend directement dans le cache.
- **Editeur WYSIWYG** : comme son nom l'indique, il s'agit d'un éditeur qui va permettre de rédiger et de mettre en forme du texte comme vous le feriez avec un traitement de texte (What You See Is What You Get).
- **Publier / dépublier**: encore une notion importante à intégrer. Il s'agit de rendre visible ou pas sur le site un article, un lien dans un menu, un module entier, une section, une

catégorie, un article. Pour un article, il est par ailleurs possible de définir un calendrier de publication, date à partir de laquelle ou jusqu'à laquelle un article sera publié.

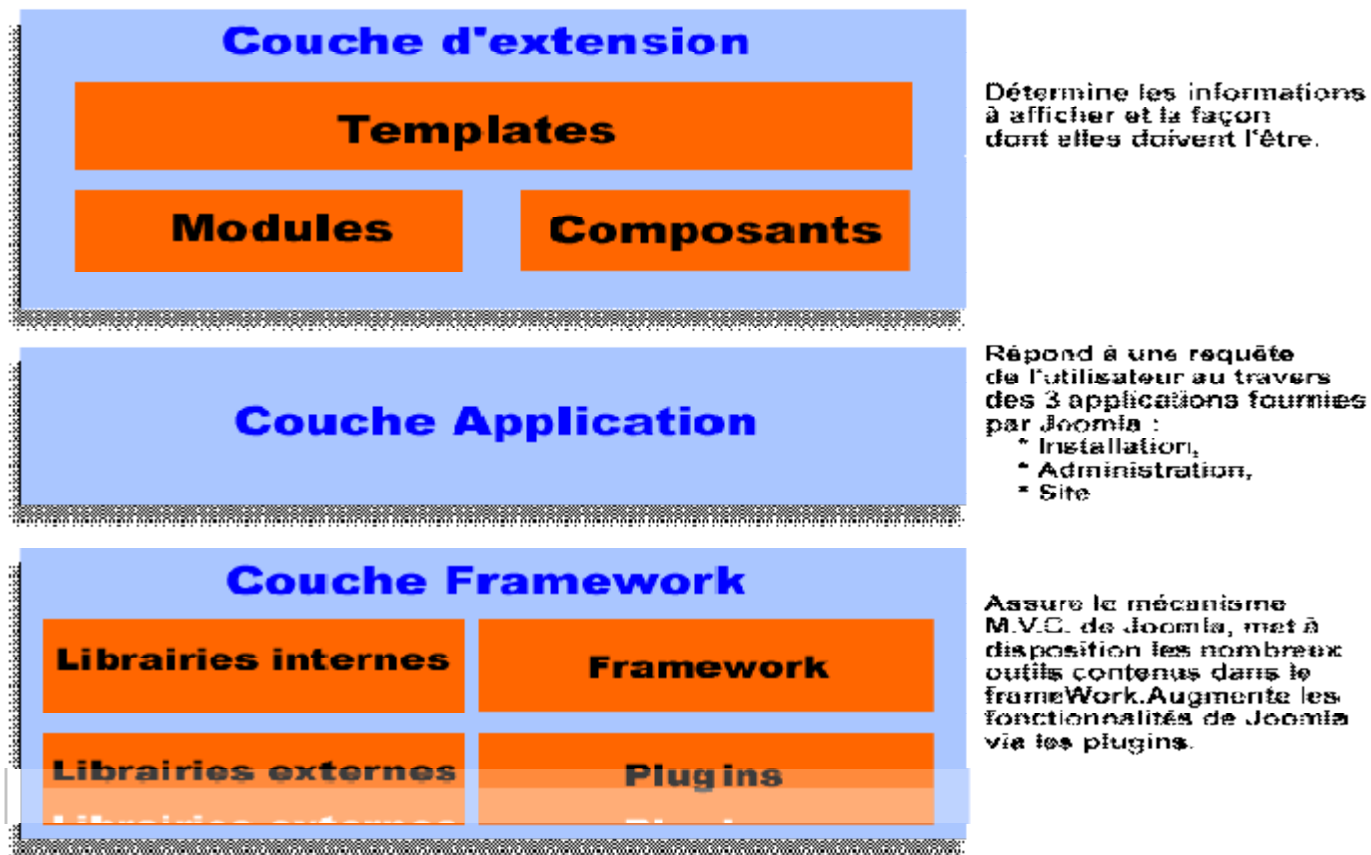
3.2. Les extensions

Joomla est un outil de gestion de contenu assez sophistiqué qui s'appuie sur des extensions, c'est-à-dire des programmes complémentaires pour gérer la mise en forme ou ajouter des nouveaux services. Ces extensions sont classées en 4 catégories : les composants, les modules, les plugins et les templates.

- **Composant** : c'est une mini application intégrée à votre site Joomla, qui dispose de sa propre interface de configuration dans la console d'administration Joomla. Ainsi à chaque fois qu'une page est chargée, Joomla fait appel à un composant pour générer le corps de la page. Les composants constituent la majeure partie de vos pages. Les composants de base sont fournis avec Joomla. D'autres composants peuvent être facilement installés par la suite.
- **Module** : pour faire simple, un module est un bloc que l'on trouvera généralement autour du corps de la page web, par exemple dans la colonne de gauche ou la colonne de droite de notre site. Ainsi le menu de gauche de votre site est placé dans un module ! De même que la bannière en haut de votre site, le bas de page ou le module d'identification.
- **Plugin** : ce sont des morceaux de code activés sur un évènement. L'exécution de n'importe quelle partie de Joomla, que ce soit le noyau, un module ou un composant, peut déclencher un évènement et alors les plugins associés à cet événement s'exécuteront.
- **Template** : un template gère toute la partie graphique de votre site : les couleurs des caractères et des fonds, la police des caractères, les cadres, les menus ... En changeant de template, vous changez le « look and feel » de votre site.

4. Architecture de Joomla

Joomla se repose sur 3 couches essentielles : couche d'extension, couche d'application et couche framework



5. L'administration de Joomla 1.7

5.1. Accès à la console d'administration

- Ø Tapez <http://www.nomdedomaine.fr/administrator> dans la barre d'adresse de votre navigateur.

Connexion à l'administration de Joomla!

Veuillez utiliser un identifiant et un mot de passe valides pour accéder à l'administration.

[Accéder à la page d'accueil du site.](#)



Identifiant

Mot de passe

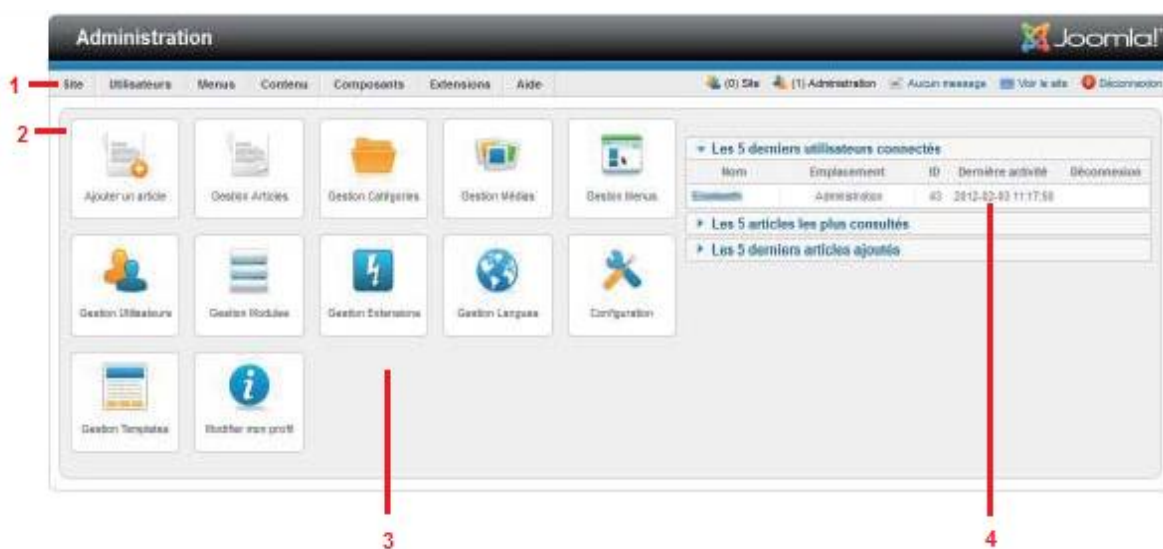
Langue ▼



- Ø Saisissez votre identifiant et votre mot de passe
- Ø Cliquez sur connexion

5.2. La page d'accueil de la console d'administration

Tous les éléments qui composent la page d'accueil de la console d'administration sont des modules administrateur que vous pourrez choisir de dépublier (sauf la barre de menu !) si vous ne souhaitez pas qu'ils soient affichés. A tout moment vous pouvez retourner sur la page d'accueil de la console d'administration en cliquant sur **Site > Panneau de configuration** tout à gauche de la barre de menus.





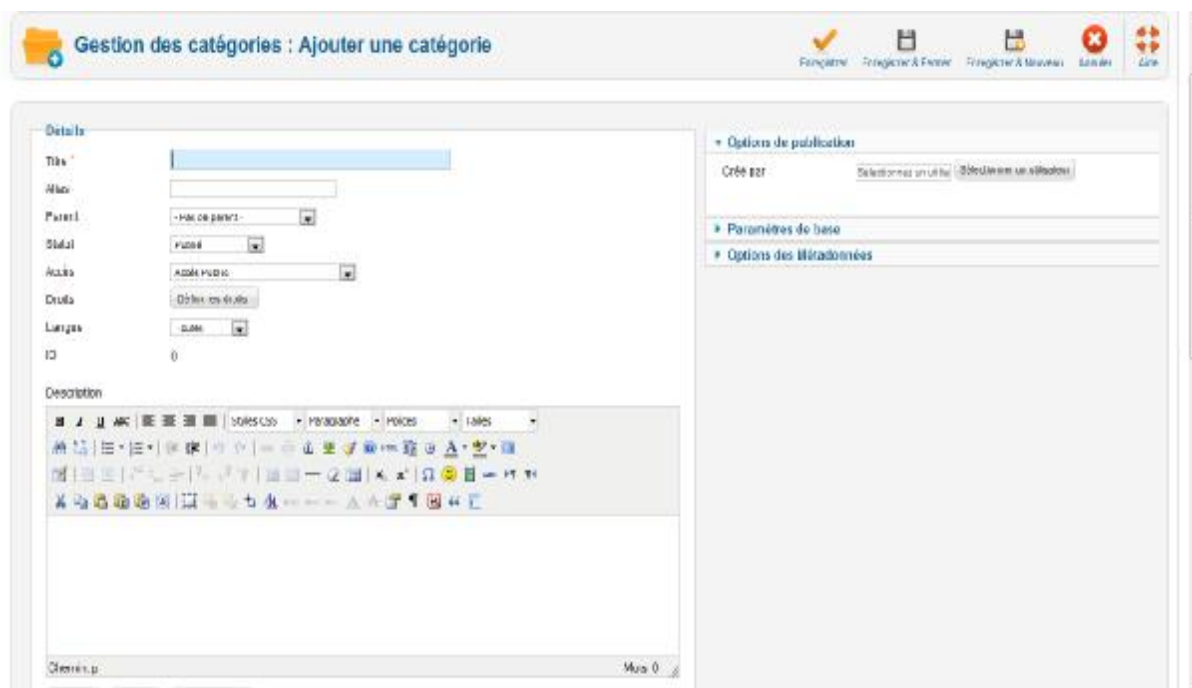
- 1 - La barre de menus vous permettra de naviguer facilement dans la console d'administration
- 2 - La barre d'infos vous indique l'endroit où vous vous trouvez dans la console d'administration
- 3 - Le panneau de contrôle affiche des icônes faisant office de raccourcis vers les différentes parties de la console d'administration.
- 4 - La vue par onglets affiche des statistiques sur votre site Joomla! (composants installés, derniers articles rédigés...)

6. Ajout et modification du contenu

6.1. Création d'une catégorie



Dans la console d'administration ouvrez le gestionnaire de Catégories «Contenu >Gestion des catégories»

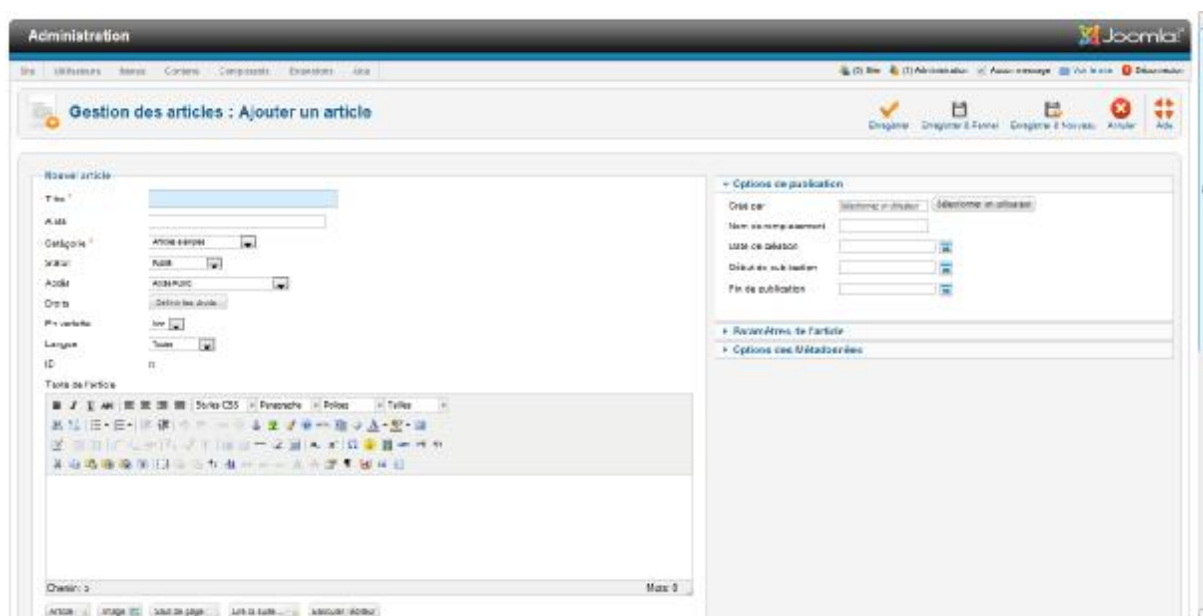
- Cliquez sur le bouton  « Nouveau »
- Saisissez un Titre et un Nom pour la catégorie, puis éventuellement une description, enfin, veillez à ce qu'elle soit bien rattachée à la catégorie parente souhaitée s'il s'agit d'une sous-catégorie
- Sauvegardez les modifications en cliquant sur le bouton «  Enregistrer »



6.2. Création d'un article

Dans la console d'administration ouvrez le menu Contenu / Gestion des articles :


- Cliquez sur le bouton «  Nouveau »
- Donnez un Titre à votre article,
- Sélectionnez la catégorie à laquelle il sera rattaché,
- Rédigez votre article (obligatoire)
- Sauvegardez en cliquant sur le bouton 



Le nouvel article a été créé avec succès, il s'affiche dans le gestionnaire des articles.

6.3. Modification d'un article

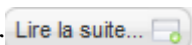
Dans la console d'administration ouvrez le menu Contenu / Gestion des articles :

- Cliquez sur le titre de l'article à modifier
- Modifiez le contenu et les options de votre article
- Enregistrez en cliquant sur le bouton 

La liste des articles réapparaît.

6.4. Créer un lien «Lire la suite»

Cette option vous permet de couper l'article en 2 : la première partie s'affichera dans un premier temps suivi d'un lien «Lire la suite», et l'article complet s'affichera une fois ce lien cliqué.

- Placez votre curseur à l'endroit du texte où vous souhaitez le couper.
- Cliquez sur 

L'éditeur affiche un trait qui sera remplacé en ligne par un lien «Lire la suite». Le texte placé à la suite du trait ne sera visible que si l'internaute clique sur le lien.




6.5. Supprimer un article

Il y a 4 façons de supprimer un contenu depuis la console.

Vous pouvez modifier le statut depuis la liste des articles ou dans l'édition d'un article.

Depuis la page «Gestion des articles» :



-  Dépublier - enlève seulement de l'affichage
-  Archiver - range l'article dans une archive (on peut afficher les archives à part)
-  Corbeille - déplace l'article dans la corbeille lors de l'édition d'un article.

| Options de publication | | |
|------------------------|--|--|
| Créé par | <input type="text" value="celine"/> | <input type="button" value="Sélectionner un utilisateur"/> |
| Nom de remplacement | <input type="text" value="Joomla!"/> | |
| Date de création | <input type="text" value="2011-01-01 00:00:01"/> | <input type="button" value="23"/> |
| Début de publication | <input type="text" value="2011-01-01 00:00:01"/> | <input type="button" value="23"/> |
| Fin de publication | <input type="text" value="0000-00-00 00:00:00"/> | <input type="button" value="23"/> |

- Dans la page de modification de l'article, à droite : Options de publication : Éditer l'article > Options de publication > Début de publication / Fin de publication : programmer une date de fin de publication dans les paramètres de l'article.

6.6. L'insertion d'un document

Vous voulez insérer un document au format pdf – ou dans un autre format - dans un article de votre site, il suffit de télécharger ce fichier, d'insérer le logo pdf par exemple dans votre article et un lien sur cette icône vers ce fichier.

Etape 1 : téléchargez le document sur votre site


L'outil le plus adéquat est le gestionnaire de médias :

- Créez votre formulaire sur votre micro-ordinateur avec un nom simple, sans blanc ni caractères spéciaux,

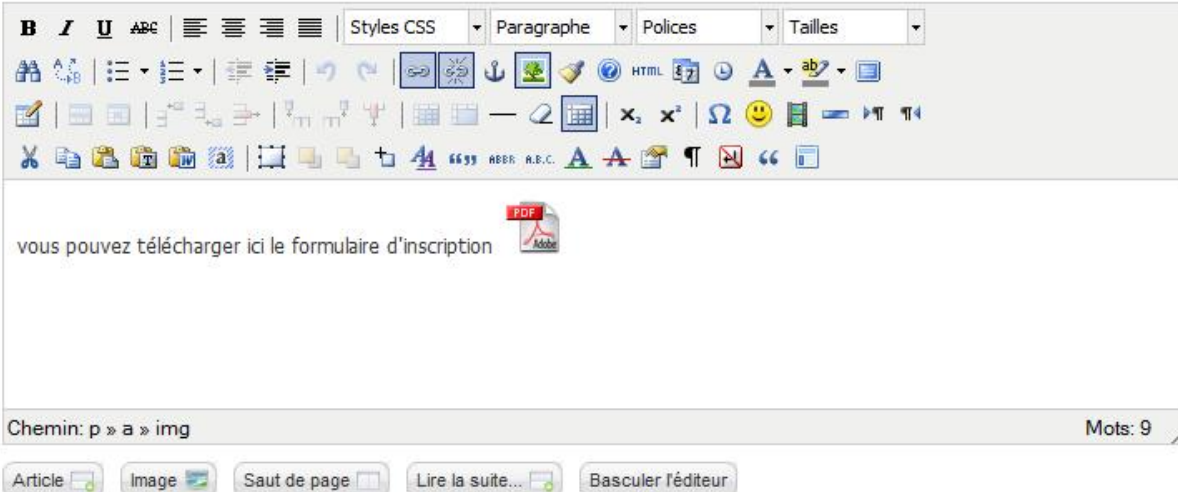
- Sélectionnez dans le menu Site l'item Gestion des médias,
- Avec le bouton Créer un dossier, créez un sous dossier documents,
- Parcourir le disque de votre micro-ordinateur pour sélectionner le fichier à télécharger et lancer le transfert.


Etape 2 : téléchargez l'icône pdf Vous pouvez créer le lien sur un bout de texte, mais, pour faire beau, vous pouvez aussi insérer dans l'article qui contient le formulaire l'icône qui représente un fichier pdf.

Etape 3 : créer votre article avec le lien vers le document

- Créez l'article où vous souhaitez, insérez le document, insérez l'image de l'icône pdf, sélectionnez l'image et choisissez la commande **Insérer/Modifier un lien** 

Texte de l'article



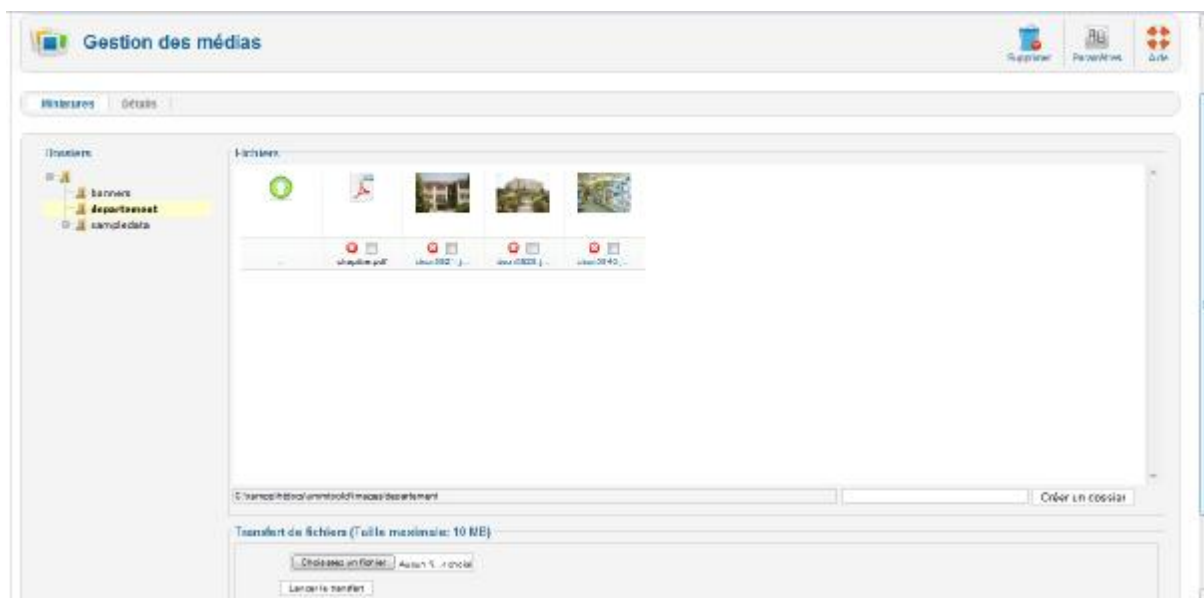
vous pouvez télécharger ici le formulaire d'inscription 

Chemin: p » a » img Mots: 9

Article Image Saut de page Lire la suite... Basculer l'éditeur

6.7. Le gestionnaire de médias

Vous pouvez utiliser le Gestionnaire de médias accessible depuis la console d'administration par le menu Contenu > Gestion des médias.



Ø Créer un nouveau répertoire

Vous avez la possibilité de créer des répertoires personnalisés pour y transférer vos images, vos pdf, vos animations flash ou vos vidéos.

- Cliquez sur le répertoire qui contiendra votre nouveau dossier.
- Sous la fenêtre de prévisualisation, renseignez le nom du dossier
- Cliquez sur «Créer un dossier».

Ø Télécharger une image sur le serveur

Pour charger une image à partir de votre disque dur, cliquez sur «Parcourir» dans la partie «Transfert de fichiers» puis sur «Lancer le transfert».

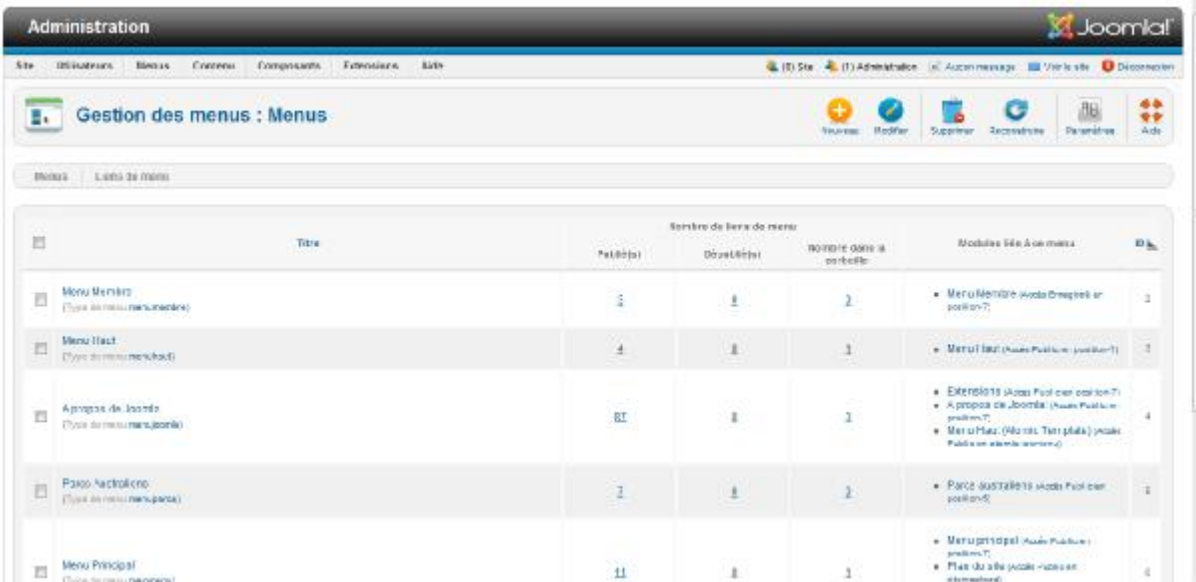
- Cliquez sur «Parcourir»
- Sélectionnez l'image à ajouter sur votre disque dur
- Renouvelez l'opération autant de fois que d'images souhaitées
- Cliquez sur «Lancer le transfert» seulement à la fin

Ceci étant fait, cliquez sur le menu Contenu / Gestion des articles puis éditez l'article dans lequel vous allez insérer une image (cliquez sur le titre de l'article pour passer en mode édition).

7. Menus et liens de menus

Pour que vos visiteurs accèdent à votre contenu, vous devez leur proposer des liens vers celui-ci dans un système de navigation


- Affichez le gestionnaire de menu (Menus > Gestion des menus)



| Titre | Nombre de liens de menu | | | Modules Site & Composants |
|---|-------------------------|-------------|----------------------------|--|
| | Publié(s) | Dépublié(s) | Éléments de menu à publier | |
| Menu Membres (Type de menu: menu_membre) | 1 | 1 | 2 | • Menu Membres (Module: MenuItems en position: 7) |
| Menu Haut (Type de menu: menu_haut) | 4 | 1 | 1 | • Menu Haut (Module: Public en position: 7) |
| À propos de Joomla (Type de menu: menu_propos) | 81 | 1 | 1 | • EXTENSIONS (Module: Public en position: 7) • À propos de Joomla (Module: Public en position: 7) • Menu Haut (Module: Titre public) (Module: Public en position: 7) |
| Page Nationale (Type de menu: menu_page) | 1 | 1 | 2 | • Page Nationale (Module: Public en position: 5) |
| Menu Principal (Type de menu: menu_principal) | 11 | 1 | 1 | • Menu principal (Module: Public en position: 7) • Plan du site (Module: Titre public en position: 7) |

7.1. Ajouter un lien

Sélectionnez le menu dans lequel vous allez ajouter des liens

- Cliquez sur le bouton «  Nouveau ».
- Sélectionnez le type de lien

La page qui s'ouvre contient 8 catégories par défaut .D'autres types de liens de menus peuvent apparaître lorsque des composants sont ajoutés.

Contacts

Liste des catégories de contacts
 Liste des contacts d'une catégorie
 Contact
 Contacts favoris

Articles

Articles archivés
 Article
 Liste de toutes les catégories
 Blog d'une catégorie
 Liste des articles d'une catégorie
 Blog des articles en vedette
 Proposer un article

Fils d'actualités (RSS)

Liste des catégories de fils d'actualités
 Liste des fils d'actualités d'une catégorie
 Fil d'actualités

Recherche

Formulaire de recherche

Gestion des utilisateurs

Connexion
 Profil de l'utilisateur
 Modification du profil
 Enregistrement
 Rappel de l'identifiant
 Réinitialisation du mot de passe

Liens web

Liste des liens de toutes les catégories
 Liste des liens d'une catégorie
 Proposer un lien web

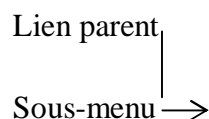
Fenêtre intégrée (IFrame)

Contenu externe

Liens divers

URL externe
 Alias de lien de menu
 Séparateur

- La suite des opérations dépend du type de lien choisi
- Le champ «menus» est automatiquement renseigné par le nom du menu préalablement sélectionné.

7.2. Créer un sous-menu


Le lien parent doit être créé en premier. Le sous-menu est créé ensuite comme n'importe quel lien puis rattaché au lien parent :

- Dans la liste «Lien parent», sélectionnez le lien auquel sera rattaché votre nouveau lien. Par défaut, positionné sur «Lien de menu racine», un lien sera au premier niveau du menu.
- La suite des opérations dépend du type de lien choisi

7.3. Ajouter un menu

Par défaut, dans Joomla!, si vous n'avez pas installé des données exemples à l'installation, est livré le Menu principal.

Il peut être nécessaire de créer un nouveau menu pour organiser sa navigation.

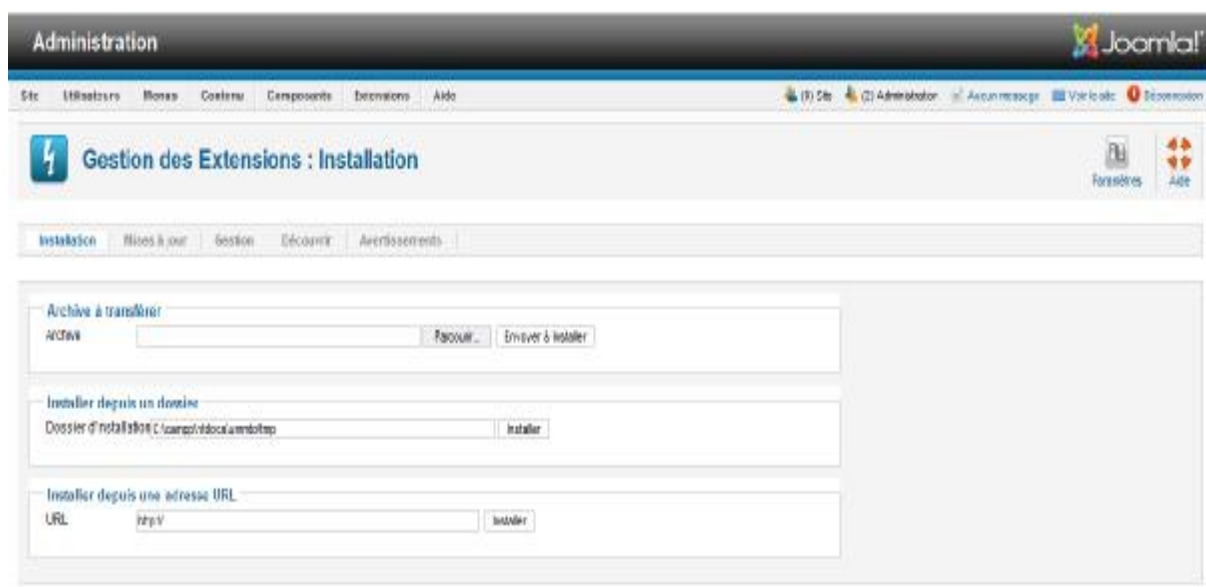
- Allez dans Menu > Gestion des menus
- Cliquez sur «  Nouveau »
 - ü Type de menu : nom unique, sans espaces ni caractères spéciaux
 - ü Titre : titre qui sera affiché au-dessus du menu
 - ü Description : facultatif
- Enregistrer

Ensuite, créer un module

8. Les extensions (composant, module, plugin, template)

8.1. Ajouter une extension

- Téléchargez l'extension désirée. Attention, elle doit être compatible avec la version Joomla! que vous utilisez.
- Affichez le menu Extensions > Gestion des extensions
- Cliquez sur «Parcourir» et sélectionnez le fichier archive de l'extension (.zip) sur votre disque dur.
- Cliquez sur «Envoyer & Installer»



- Rendez-vous sur «composant» ou «extension» pour le paramétrer
- N'oubliez pas de créer un lien dans un menu s'il s'agit d'un composant afin qu'il soit accessible aux visiteurs.

S'il s'agit d'un module ou d'un plugin, activez-le depuis l'onglet «Extensions >

Gestion des ...» (modules / plugin)

8.2. Les composants

Rappel :

Les composants est une mini application intégrée à votre site Joomla!, qui dispose de sa propre interface de configuration dans la console d'administration. Certains composants sont préinstallés. D'autres peuvent être facilement installés par la suite (forums, livre d'or, galerie d'images, gestionnaire de newsletter, gestionnaire de formulaires... et bien d'autres encore).

Les composants natifs de Joomla!

Lorsqu'on installe le pack Joomla, certains composants sont installés par défaut. Ils font aussi l'objet de mises à jour lorsque la communauté publie des correctifs.

Les 5 premiers de la liste suivante fonctionnent tous sur le même principe :

- Ø Des éléments sont créés pour afficher des informations
- Ø Des catégories (obligatoires) permettent de les classer et/ou de les afficher automatiquement en liste

Pour chacun, la démarche pour créer une catégorie sera la même :

- Rendez-vous dans le menu Composants > (nom du composant) > Catégories
- Cliquez sur « Nouveau »
- Renseignez le titre
- Renseignez une description si vous le souhaitez puis sauvez

8.3. Les modules

Rappel

pour faire simple, un module est un bloc que l'on trouvera généralement autour du corps de la page web, par exemple dans la colonne de gauche ou la colonne de droite de notre site. Certains modules sont pré-installés (module de menu principal, module d'identification...).

D'autres, le plus souvent associés à des composants (par exemple module affichant une photo aléatoire tirée d'un composant galerie d'images), peuvent être installés ultérieurement. Les menus sont des modules dont le contenu est géré dans l'onglet «menu» de la console.

Les positions

Elles sont définies par le template. Tout peut changer donc d'un template à un autre ! Le site est découpé en une grille dont chaque case est nommée.

Il peut y avoir plusieurs modules dans une même position. Ils se mettront alors les uns au-dessous des autres selon leur ordre dans la liste des modules. Ceci peut être modifié facilement dans le panneau de gestion des modules.



| Titre | Templates |
|---------------------|-----------------------------|
| atomic-bottomleft | • atomic (En bas à gauche) |
| atomic-bottommiddle | • atomic (En bas au centre) |
| atomic-search | • atomic (Recherche) |
| atomic-sidebar | • atomic (Barre latérale) |
| atomic-topmenu | • atomic (Top Menu) |
| atomic-topquote | • atomic (Top Citation) |
| banner | • js_jamba (Banner) |

Ajouter un module

Plusieurs paramètres sont réglables sur les modules :

- Ø Afficher ou non le titre du module
- Ø Sélectionner la position
- Ø Modifier le niveau d'accès (public, enregistrés, statut spécial, clients)
- Ø Afficher sur une ou plusieurs pages (par défaut, sur tout le site)
- Ø Changer l'ordre d'affichage

∅ ... Autres paramètres suivant les modules


Un module s'installe comme toute extension mais une fois installé, il peut être ajouté plusieurs fois dans le site et être paramétré différemment : plusieurs menus utiliseront le même module (menu) mais afficheront des liens différents dans des positions différentes.

Pour ajouter un module déjà en place dans le site

- Affichez le menu Extensions > Gestion des modules
- Cliquez sur « Nouveau »
- Sélectionnez le module à ajouter
- Cliquez sur « suivant »
- Sélectionnez sa position dans la liste déroulante
- Modifiez les paramètres (changent suivant le module)
- Enregistrez

Il est possible, comme pour les articles de paramétrer une date de début et/ou de fin de publication.

Il est possible également de dupliquer un module existant. Il conservera alors les paramètres de son « clone » et pourra être ou non personnalisé.

- Allez dans « Extensions > Gestion des modules »
- Sélectionnez avec la case à cocher le module à dupliquer
- Cliquez sur  Dupliquer
- Editez le nouveau module : le module créé est dépublié et comporte un numéro(N) à la suite de son nom.

8.4. Les plugins



Un plugin est un morceau de code qui peut être activé depuis n'importe quel composant, module et contenu de votre site Joomla! Il permet en général d'automatiser des actions. Certains plugins par exemple permettent de transformer dynamiquement un petit code en une galerie de photos. Joomla! est livré avec une multitude de plugins comme par exemple le plugin de recherche de contenu qui sera exécuté lors de la recherche de données depuis le module de

recherche. Les plugins peuvent être listés depuis l'onglet «Extensions > Gestion des plugins». Un plugin s'installe comme toute extension mais doit être activé pour fonctionner. Il serait long et fastidieux de lister ici tous les plugins déjà installés dans Joomla! Mais voici une liste des plugins qu'on pourra être amené à utiliser lors de l'administration d'un site Joomla! :

- Ø **Éditeur - TinyMCE**: vous pouvez paramétrer assez finement les boutons et comportements souhaités pour l'éditeur WYSIWYG que vous utilisez un peu partout dans Joomla!
- Ø **Recherche - (...)** : paramétrez ici quelques options d'affichage des résultats de recherche pour les contacts, articles, catégories, liens et fils d'actualités
- Ø **Système - Filtre de langue** : à activer si vous souhaitez gérer un site multilingue avec les options multilingues de Joomla!
- Ø **Utilisateur - Profi 1** : vous pouvez l'activer pour paramétrer les champs obligatoires ou facultatifs des fiches utilisateurs lors de l'inscription
- Ø **Utilisateur - Fiches de contact automatiques** : utile à activer si vous souhaitez automatiquement générer une fiche contact pour chaque utilisateur à l'enregistrement

9. La gestion des utilisateurs

9.1. Changer les infos de l'administrateur

Vous avez reçu votre identifiant et votre mot de passe administrateur. Vous pouvez modifier le nom et surtout l'identifiant de connexion du compte administrateur afin de sécuriser l'accès à la console d'administration.


- Affichez le Gestionnaire des utilisateurs (menu Site)
- Accédez aux propriétés de l'utilisateur Administrator (1 clic sur le nom)
- Modifiez les champs Nom et Utilisateur à votre convenance.

9.2. Créer un utilisateur

Il existe deux façons de créer un utilisateur dans Joomla!

- Ø **Depuis le frontend** : le visiteur de votre site pourra cliquer sur le lien Créer un compte dans le module d'identification s'il est affiché ; il devra fournir un nom complet, un

identifiant, une adresse mail et un mot de passe. Le compte nouvellement créé sera automatiquement ajouté au groupe d'utilisateurs par défaut paramétré. Vous pouvez

désactiver l'enregistrement des utilisateurs depuis le backend en cliquant sur  dans le gestionnaire des utilisateurs.

Ø **Depuis le backend** (interface d'administration) : seuls les utilisateurs membres du groupe Super Admin sont habilités à gérer les comptes (menu Site > Gestion des utilisateurs).

9.3. Gestion des accès

La gestion des droits utilisateurs a longtemps été l'une des grandes lacunes de Joomla!. Par défaut Joomla! propose trois niveaux d'accès et sept groupes d'utilisateurs prédéfinis.

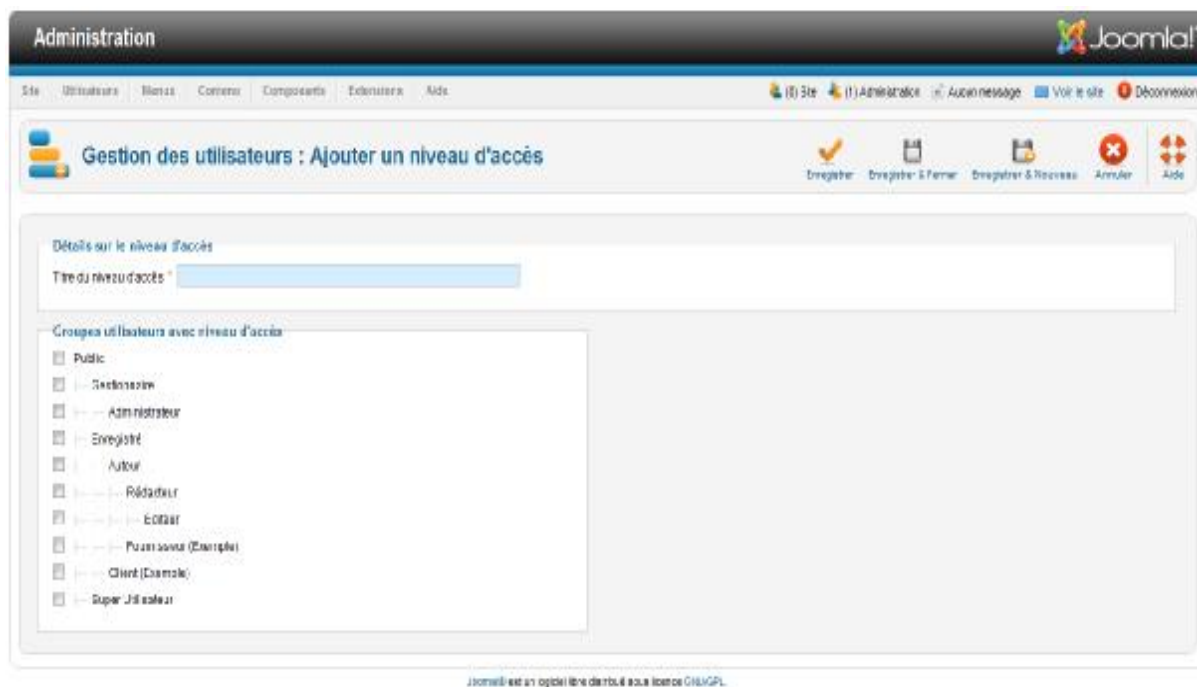
Il est possible maintenant de les modifier, les supprimer ou en créer de nouveaux. On peut aussi attribuer finement des droits à une personne ou un groupe de personnes : à 1 article, 1 catégorie, 1 composant...

Il est possible de paramétrer finement les droits, niveaux et groupes.

Joomla! dispose de trois niveaux d'accès par défaut, qui permettront de restreindre l'accès à certaines parties de votre site en fonction de l'appartenance à un ou plusieurs groupe(s) d'utilisateurs donné(s) :

- **Public** : l'élément du site (catégories, articles, liens, modules) auquel est attribué ce niveau d'accès est visible par tous les visiteurs ; il s'agit du niveau d'accès par défaut.
- **Enregistré** : ce niveau d'accès peut-être traduit par membre ou utilisateur enregistré ; l'élément du site (catégories, articles, liens, modules) auquel est attribué ce niveau d'accès sera visible uniquement par les utilisateurs membres du ou des groupes sélectionnés dans ce niveau qui se seront authentifiés sur votre site à l'aide d'un identifiant et d'un mot de passe.
- **Spécial** : l'élément du site (catégories, articles, liens, modules) auquel est attribué ce niveau d'accès sera visible uniquement par les utilisateurs membres des groupes sélectionnés dans ce niveau.

- Tout autre niveau d'accès qui pourra être créé depuis l'onglet «Utilisateurs > Niveaux d'accès»



Conclusion

L'objectif de notre projet a été de concevoir et de réaliser une application Web permettant la gestion du département d'Electronique de la faculté de Génie Electrique et d'Informatique de l'UMMTO, afin d'améliorer son système d'information pour pallier aux problèmes de la gestion classique.

L'application à laquelle on a abouti, répond à plusieurs de ces fonctionnalités, néanmoins quelques améliorations et extensions peuvent aussi être ajoutées. En effet, ce projet m'a permis d'apprendre énormément de choses sur le développement WEB. Il m'a donné la chance de toucher à divers langages tels que PHP, Javascript, HTML, CSS, SQL, UML ... Ainsi, il m'a permis de me familiariser avec un certain nombre d'outils de développement tels que Xampp et le CMS Joomla.

Ce Projet fut pour moi une expérience très enrichissante car il m'a permis d'améliorer mes compétences dans le domaine du développement Web.

Références

- [1] : George et Olivier GARDARIN "**Le client-serveur**". Eyrolles,1996
- [2] : <http://fr.wikipedia.org/wiki/Extranet>
- [3] : <http://fr.wikipedia.org/wiki/Internet>
- [4] : François Laissus, "**Cours d'introduction à TCP/IP**". <http://www.laissus.fr/cours/cours.html>, version 2009.
- [5] : http://fr.wikipedia.org/wiki/Domain_Name_System
- [6] : <http://faq.domaine-achat.fr/content/3/48/fr/serveur-dns-definition-.html>
- [7] : http://fr.wikipedia.org/wiki/World_Wide_Web
- [8] : Ivan GAUTREAU "**Joomla ! pour les nuls**".
- [9] : Mathieu Nebra"**Concevez votre site web avec PHP et MySQL**". <http://www.siteduzero.com/>, 2012
- [10] : <http://www.uml-sysml.org/diagrammes-uml-et-sysml/diagramme-uml/use-case-diagramme>
- [11] :<http://www.apachefriends.org/en/xampp.html>
- [12] : <http://www.commentcamarche.net/contents/entreprise/intranet.php3>
- [13] : Johann-Christian Hanke"**Compétence Micro N° 6, Joomla ! Version 1.5**". KnowwareEurl - 05/03/2009
- [14] : <http://www.commentcamarche.net/contents/apache/apacintro.php3>
- [15] : <http://glossaire.infowebmaster.fr/w3c/>
- [16] : <http://www.apprendre-joomla.com/mvc-joomla-15.html>

Livres et mémoires utilisés

- Christian Soutou"**UML 2 pour les bases de données**",Éditions Eyrolles.
- Pascal Roques"**UML2 Modéliser une application web**", Eyrolles.2007.
- Mihaly Marti "**Joomla ! Créez un site web complet**", Pearson.2010.
- Ibazizene Aziz"**Conception et réalisation d'un site web pour le laboratoire de recherche LAMPA**". Université Mouloud Mammeri de Tizi-Ouzou. Promotion : 2009/2010.
- Ndiaga Gueye"**Conception et Réalisation d'un portail web dynamique pour LA MAIRIE DE LA VILLE DE KAFFRINE**". Université Paris Ouest. 2009/2010.

Résumé

Ce travail porte sur « la conception et la réalisation d'une application web » pour apporter une solution aux problèmes de la gestion du département d'électronique de la faculté de Génie Electrique et d'Informatique de l'UMMTO. La façon actuelle de gestion, génère une lenteur et un manque d'efficacité. Ce projet comprend la mise en place de cette application qui doit être adaptée au département. Il s'agit de la concevoir et de la développer (en PHP & MySQL) en utilisant le CMS Joomla pour enfin trouver un moyen pour l'héberger et la mettre en ligne sur le web.

Le présent rapport se compose en trois chapitres. Le premier présente quelques généralités et définitions jugées nécessaires pour introduire notre travail. Le second chapitre s'attache à exposer une analyse au problème et la solution conçue exprimée principalement en diagrammes UML commentés brièvement. Le dernier chapitre traite la réalisation, les outils utilisés et le résultat à travers quelques captures d'écrans reflétant les fonctionnalités fournies par l'application.