

Université Sâad DAHLEB de Blida



Faculté des sciences

Département d' : Informatique

Mémoire présenté par :

CHAIRI Ryma Fatima Zahra

FORTAS Ahlem

En vue d'obtenir le diplôme de Master

Domaine : MI

Filière : Informatique

Spécialité : Informatique

Option : Ingénierie des logiciels

**Sujet : Développement d'un système de e-Learning via le
CMS Joomla**

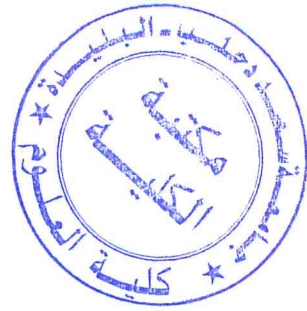
Soutenu le : 24 septembre 2013, devant le juré composé de :

Président : Mme BOUCETTA

Rapporteur : Mme TOUAHRI

Examineur : Mme MANCER

REMERCIEMENTS



En préambule à ce mémoire, nous remercions ALLAH tout puissant et miséricordieux, pour nous avoir donné le courage, la force ainsi que la capacité de pouvoir mener jusqu'à sa fin ce modeste travail.

Nous remercions aussi nos chers parents, sans qui, ce mémoire n'aurait pas pu voir le jour, ni aboutir, pour leur soutien moral et surtout pour leur disponibilité, compréhension et encouragement quotidien.
«Grand Merci ».

Qu'il nous soit permis d'exprimer nos vifs remerciements à notre promoteur Monsieur SIDOUMOU, qui, en tant que Directeur de mémoire, s'est toujours montré à l'écoute tout au long de la réalisation de ce mémoire.

Nous adressons de même nos remerciements aux membres du jury pour le temps consacré à la lecture de ce mémoire et leur aimable attention. Nous ferons l'honneur d'apprécier et de prendre connaissance de toutes éventuelles critiques et corrections de leur part.

Enfin, nous remercions toutes personnes ayant contribué de près ou de loin à la réalisation de ce travail.

Merci à tous et à toutes.



**A mes Chers
Parents, Grands Parents,
Ma tante Souhila,
Ma sœur Yousra,
Ahlem,
Ma Famille, mes Proches
Et tous mes
Amis ...**

Ryma



**A mes Chers :
Parents, Grands Parents,
Soeurs, Frère
Oncles et Tantes,
Ryma,
Ma famille et tous mes
Amis ...**

Ahlem

Résumé

L'apprentissage en ligne offre la possibilité de faire un apprentissage dépourvu de la contrainte de l'espace et parfois même du temps. C'est pour cela qu'il faut étendre l'apprentissage qu'on offre via des solutions de e-Learning afin d'alléger les contraintes de temps et d'espace.

De nombreux outils peuvent être utilisés pour compléter une solution de e-Learning comme : Quiz, examen en ligne, forums de discussions, commentaire, vidéo conférence, bureaux partagés,... etc.

Durant le processus de développement de notre projet nous avons implémenté une solution de e-Learning en y implémentant un ensemble des outils parmi ceux qui sont cités ci-dessus.

Nous avons utilisé l'un des outils nommé le CMS Joomla pour nous aider à contenir ce problème, un CMS des plus célèbres. Joomla nous a offert une facilité de travail ainsi qu'un certain respect des normes pour offrir un système prêt à être utilisé.

Cela nous a entraînés dans la nécessité d'étudier l'outil Joomla et ses fonctionnalités ainsi que son architecture basée sur le modèle MVC (Modèle-Vue-Contrôleur).

Mots clés :

Apprentissage en ligne, e-Learning, CMS, Joomla, MVC.

Abstract

Online learning offers the possibility to make a free learning space and even time. This is why it is necessary to extend the learning we offer via e-Learning solutions to alleviate the constraints of time and space.

Many tools can be used to complete a solution e-Learning as: Quiz, online review, discussion forums, comments, video conferencing, shared offices...

During the development process of our project we have implemented a solution for e-Learning are implementing a set of tools from those cited above.

We used one of them, named CMS Joomla to help us contain this problem, one of the most popular CMS tools. Joomla gave us a workability as well as a compliance to provide a system ready to use.

It has trained us to the need to study Joomla and its features and its tool based on the MVC (Model-View-Controller) architecture.

Keywords :

Online learning, e-learning, CMS, Joomla, MVC.

ملخص

التعلم عبر الإنترنت يوفر إمكانية التعلم في فضاء حر غير مقيد بالوقت. هذا هو السبب في أنه من الضروري تمديد التعلم الذي يقدم عبر حلول التعليم الإلكتروني (e-Learning) للتخفيف من قيود الزمان والمكان.

هناك العديد من الأدوات التي يمكن أن تستخدم لإكمال حل التعلم الإلكتروني كالمسابقات، الامتحانات عبر الإنترنت، منتديات المناقشة، التعليقات، مؤتمرات الفيديو، والمكاتب المشتركة،... الخ. أثناء عملية تطوير مشروعنا قمنا بتنفيذ حل للتعلم الإلكتروني وتنفيذ مجموعة من الأدوات من تلك المذكورة أعلاه.

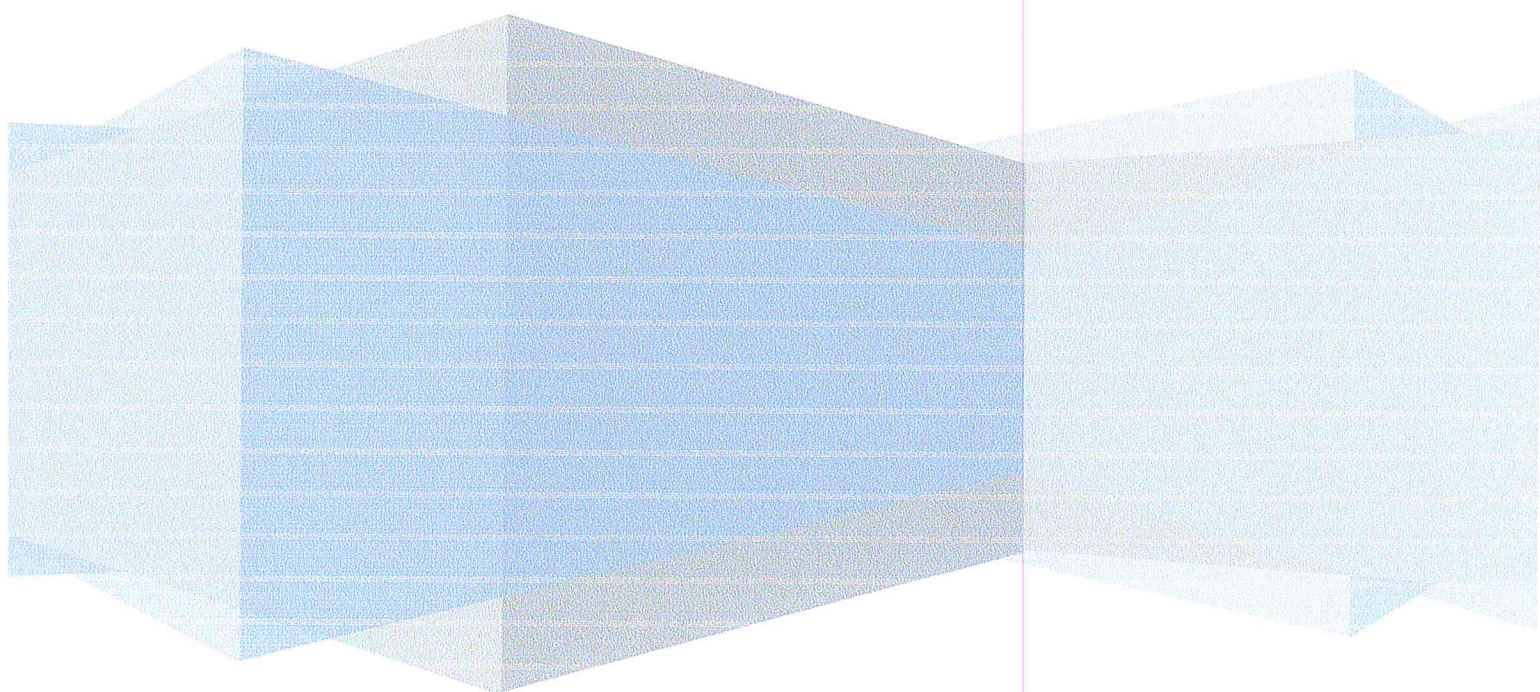
استخدمنا واحدة من الأدوات اسمها CMS جملة لمساعدتنا على احتواء هذه المشكلة. هي واحدة من CMS الأكثر شعبية.

جملة قامت بتسهيل عملنا وكذلك احترام المعايير لتقديم نظام جاهز للاستعمال. وهذه الأخيرة استدعتنا للتمرن على أدوات جملة وخصائصها وكذلك هندستها المبنية على نموذج MVC

كلمات البحث:

التعلم الإلكتروني، التعلم، MVC، جملة، CMS

Liste des figures



- P10 – **Figure 1**: exemple mécanisme du elearning.
- P22 - **Figure 2**: exemple d'interface d'administration Joomla 2.5.
- P23- **Figure 3** : Le fonctionnement de page web en PHP.
- P25- **Figure 4** : Les droits d'accès dans joomla.
- P27- **Figure 5** : Le positionnement des éléments dans page web joomla .
- P30- **Figure 6** : modèle MVC global.
- P31- **Figure 7** : diagramme de séquence d'un MVC simple.
- P32- **Figure 8** : fonctionnement d'un MVC dans le web.
- P39- **Figure 9** : Diagramme de cas d'utilisation globale.
- P40 - **Figure 10** : premier niveau: Authentification.
- P41- **Figure 11** : premier niveau: Gérer tous les articles.
- P41- **Figure 12**: premier niveau: Gérer le site.
- P42- **Figure 13** : second niveau: Gérer le contenu.
- P42- **Figure 14** : second niveau: Gérer les extensions.
- P43- **Figure 15** : second niveau: Gérer les utilisateurs.
- P44- **Figure 16** : Diagramme de séquence : Identification administrateur.
- P45- **Figure 17** : Diagramme de séquence Identification utilisateur.
- P46- **Figure 18** : Diagramme de séquence installation des extensions.
- P47- **Figure 19** : Diagramme états transition : cas d'articles.
- P49- **Figure 20** : Diagramme de class Analyse.
- P50- **Figure 21** : Diagramme de cas d'utilisation globale.
- P50 - **Figure 22** : premier niveau: acceder à la plate-forme.

- P51 - **Figure 23** : premier niveau: ajouter un contenu dans la plate-forme.
- P51 - **Figure 24** : premier niveau: Gérer la plate-forme .
- P52 - **Figure 25** : diagramme de séquence configuration forum .
- P53- **Figure 26** : diagramme de séquence configuration examen.
- P54- **Figure 27** : diagramme de séquence configuration cours.
- P55- **Figure 28** : diagramme de séquence création examen.
- P56- **Figure 29** : diagramme de séquence modifier examen.
- P57 - **Figure 30** : diagramme de séquence passer examen.
- P58 - **Figure 31** : diagramme de séquence ajouter un sujet.
- P58 - **Figure 32** : diagramme de séquence ajouter un commentaire.
- P59 - **Figure 33** : diagramme de séquence consulter un cours.
- P61 - **Figure 34** : digramme class forum/examen/cours.
- P62 - **Figure 35** : Le déroulement du composant Forum.
- P63 - **Figure 36** : Le déroulement du composant du composant cours pédagogique et examen.
- P68 - **Figure 37** : La page d'accueil du site.
- P69 - **Figure 38** : Menu cours pédagogiques.
- P69 - **Figure 39** : Le contenu d'un chapitre.
- P70 - **Figure 40** : Espace forum.
- P71 - **Figure 41** : installation réussite des composants.
- P72 - **Figure 42** : Sous menu cours pédagogique.
- P73 - **Figure 43** : Modifier un chapitre.

Il est plus que compréhensible que de nombreuses personnes ne prenaient alors pas en charge cette tâche eux-mêmes et confiaient la création de leur site web à une agence web, ou ne se lançaient tout simplement pas dans le projet.

Grâce aux kits tels que Google Sites et d'autres sites et plates-formes d'apprentissage en ligne, la création de pages Web simples est devenue relativement facile, mais si vous voulez quelque chose d'unique, vous devez vous familiariser avec un système de gestion de contenu.

Pour cela, Joomla! propose tout ce dont vous avez besoin pour créer votre propre site web personnel. C'est un outil comprenant beaucoup de possibilités et vous pouvez utiliser le système dans une grande variété de configurations, en fonction de vos idées et de vos souhaits.

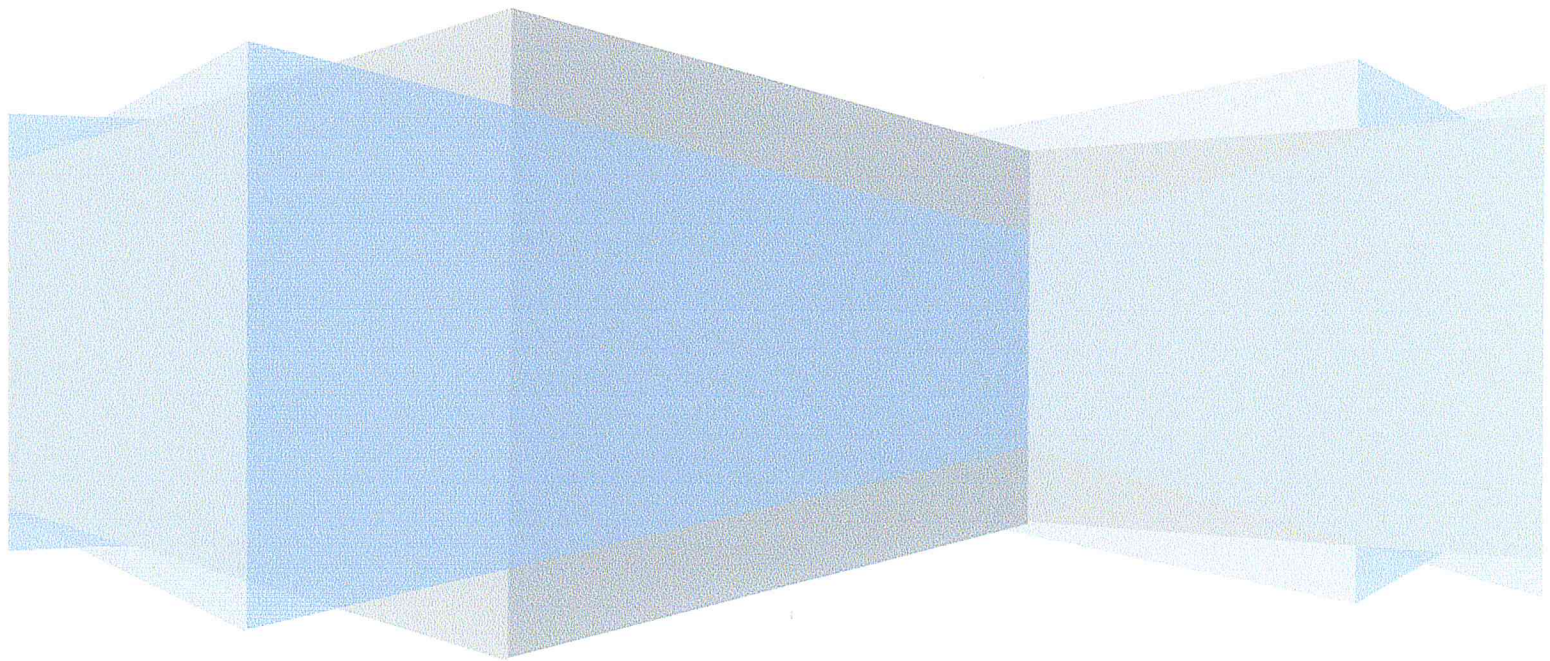
L'objectif de notre projet est de *développer un système de e-Learning via le CMS Joomla*.

Pour ce faire, nous avons structuré notre document en quatre chapitres :

- Dans le premier chapitre : nous présentons les concepts de base du e-Learning, les facteurs de motivation et parler des plates-formes de formation en ligne.
- Au second chapitre, nous allons parler du CMS Joomla, de son historique et de son principe de fonctionnement ainsi que le modèle MVC et les étapes de création d'un composant Joomla.
- Le troisième chapitre, consiste à faire une étude conceptuelle qui se répartie selon deux phases : la première est « Analyse » qui explique ce que nous avons extrait de la base de données Joomla, et la deuxième « Conception » qui explique ce que nous avons ajouté à la base de données Joomla.
- Le quatrième chapitre est la mise en œuvre de notre application avec des illustrations.
- Enfin, nous terminerons avec une conclusion générale.

Chapitre 1

Le e-Learning



1. Introduction

Les technologies Internet intéressent depuis longtemps les acteurs de la formation à distance. Qu'elles soient utilisées en visioconférence, en classe virtuelle ou intégrées à un module d'auto-formation, ces technologies permettent de développer de nouvelles méthodes pour assurer l'apprentissage. Néanmoins, les choix technologiques interviennent dans un contexte technique et économique d'un projet qu'il faut prendre en compte en amont de la mise en place d'un dispositif de formation en ligne.

Le e-Learning est un domaine révolutionnaire, un changement dans le champ de l'apprentissage. Les instructions qu'il offre en ligne peuvent être fournies **n'importe quand et n'importe où** par une gamme très vaste de solutions d'apprentissages électroniques telles que les groupes de discussions, les cours virtuels « en live », vidéo et audio, Web chat, simulations.

Aujourd'hui, le e-Learning est lié spécialement à deux secteurs :

- Des grandes entreprises qui l'utilisent avec le KM (Knowledge Management ou management des connaissances) pour former et actualiser les connaissances de ses employés de manière plus rapide.
- Des universités qui offrent des alternatives de formation à distance. ^{[1]p11}

2. Définition du e-Learning :

Il existe plusieurs définitions du e-learning. Certaines réduisent le e-Learning à l'apprentissage par internet, d'autres ont un champ d'application plus étendu telle que celle-ci, extraite du "Guide des solutions pour le e-learning": « *E-learning. Les activités, processus et actions liés à l'éducation et à l'apprentissage, formel ou informel, qui font appel à des moyens de communication électroniques tels qu'Internet, intranet, extranet, les cédéroms, la vidéo, la télévision, le téléphone portable, les agendas électronique, etc.* » [2]p4

Le e-Learning est un processus d'apprentissage à distance, qui repose sur la mise à disposition de contenus pédagogiques via un réseau de type Internet ou Intranet et permet ainsi à une ou plusieurs personnes de se former à partir d'un ordinateur.

Les supports multimédias utilisés peuvent combiner du texte, des graphismes, du son, de l'image de synthèse, de l'animation et même de la vidéo. Ces supports permettent une nouvelle approche pédagogique, avec l'emploi de méthodes plus attrayantes où l'interactivité joue un grand rôle, et avec la possibilité de s'adapter davantage au processus d'apprentissage de l'apprenant.

L'utilisateur peut se former à son rythme, en fonction de ses besoins et de ses disponibilités, ce qui est particulièrement important à une époque où la formation se décline tout au long d'une vie.

Le e-Learning est une vraie démarche de l'entreprise qui doit être conduite comme un projet venant de la direction pour avoir une chance d'aboutir favorablement. [3]p5

3. Concept de base du e-Learning :

Le e-Learning consiste à utiliser les ressources de l'informatique et de l'Internet pour acquérir, à distance, des connaissances. Ce type d'apprentissage, constituant une des facettes du E-Learning, se différencie par l'utilisation de la technologie Web. Ces applications et ces contenus sont dispensés via un ordinateur relié à un intranet ou à Internet. Les contenus constituent une étape plus avancée que l'utilisation du CD-Rom ou les cours sont dispensés en salle.

Le E-Learning est un domaine révolutionnaire, un changement dans le champ de l'apprentissage. Les instructions qu'il offre en ligne peuvent être fournies n'importe quand et n'importe où par une variété de solutions d'apprentissages électroniques. [4]p1

Quand on parle du e-Learning, il ne faut pas oublier une partie importante que sont les Apprenants → *Est-ce que les objectifs d'apprentissage ont été basés sur les besoins du public cible ? Est-ce que les apprenants sont-ils satisfaits avec les résultats ?*

La réussite d'un cours à distance est directement liée à la motivation et au comportement de l'apprenant. Pour qu'il puisse arriver à la fin de la formation, il aura besoin de tutorat et de support technique. [1]p11

4. Les facteurs de motivation:

Il est important de bien identifier les facteurs qui motivent les entreprises à mettre en place des projets de formation à distance:

- Proposer une formation homogène et personnalisable : Une première motivation concerne la possibilité de distribuer un message homogène pour l'ensemble des apprenants : tous suivent la même formation avec les mêmes contenus. Le eLearning offre aussi la possibilité de personnaliser le parcours de formation de l'utilisateur en lui permettant d'assister aux cours de son choix suivant ses besoins.

- Offrir une formation juste à temps: La formation à distance permet à un utilisateur de se former lorsqu'il en a besoin, à n'importe quel moment, sur un sujet pertinent par rapport à son métier sans avoir à attendre la programmation d'un cours à un moment précis. Pour l'entreprise, cela signifie une plus grande réactivité avec par exemple la possibilité de former ses employés dès la sortie d'un nouveau produit. Ceci représente une des conditions de la réussite de la mise en place d'un projet eLearning, car une formation est efficace uniquement si elle est rapidement mise en oeuvre.

- Réduire les frais de vie La formation à distance permet une réduction des frais de déplacement et d'hébergement des apprenants, et des frais liés à la prestation du formateur. De plus, le temps d'apprentissage personnel est réduit. En se consacrant uniquement aux points qu'il souhaite approfondir, l'apprenant se forme plus rapidement. Le coût de son indisponibilité est donc fortement réduit.

- Un rythme d'apprentissage libre La formation à distance permet également de tenir compte des disponibilités de chacun. Contrairement à une formation classique en présentielle, le temps n'est plus une contrainte. Chacun peut se former à son rythme.

- Formation sur le lieu de travail Les modes de diffusion de la formation à distance offre la possibilité de recevoir la formation directement sur le lieu de travail : dans une salle dédiée à la formation ou sur le poste de travail de l'apprenant. Ainsi, il n'y a plus de déplacement et il est possible de se former sans quitter son établissement.

- Une formation plus universelle Avec le eLearning il devient possible de former les populations d'employés qui sont habituellement exclus des plans de formation de l'entreprise, comme les intérimaires ou les vacataires extérieurs.

Comme on peut le constater la liste est longue et il serait possible d'ajouter de nouveaux bénéfices comme par exemple la capitalisation des savoirs faire, la gestion de la connaissance... [3]p9

5. Les plates-formes de formation en ligne :

Une plateforme pédagogique est un portail qui fournit un soutien pour l'apprentissage en ligne. Elle sert à créer des communautés d'apprenants autour de contenus didactiques et d'activités pédagogiques ; elle permet la gestion d'étudiants, d'apprenants au sens large du terme et facilite la mise en œuvre de stratégies pédagogiques.

Plus précisément, elle est un logiciel qui assiste la conduite des formations à présent et à distance, basée sur des techniques de travail collaboratif et regroupe les outils nécessaires aux trois principaux acteurs de la formation : apprenant, tuteur, administrateur.

Ses principales fonctions sont pédagogiques ou communicatives, elles permettent de créer un environnement d'apprentissage en ligne. Par l'intermédiaire du réseau (internet ou intranet), cette application permet de créer des interactions entre des pédagogues, des apprenants et des ressources pédagogiques.

Autres noms de la plateforme pédagogique : plateforme d'apprentissage, plateforme e-Learning.

Le schéma1 représente d'une façon générale le mécanisme du E-Learning et l'interaction entre les pédagogues et les apprenants au tour des ressources pédagogiques.

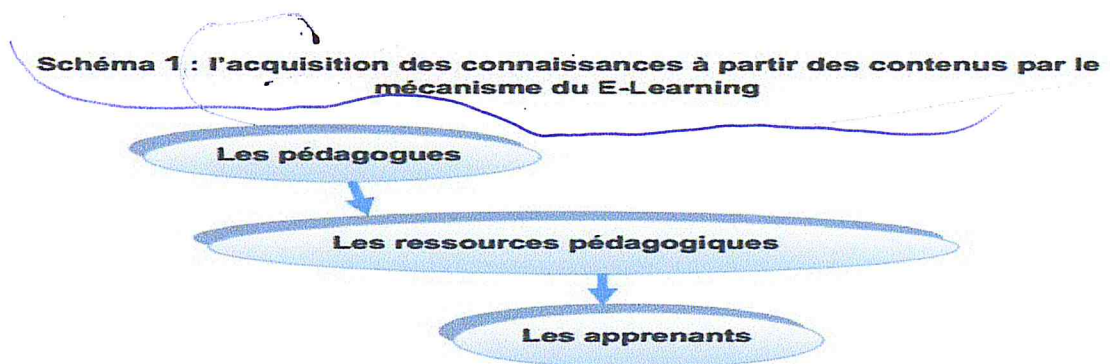


Figure1: exemple mécanisme du elearning

Une plateforme d'apprentissage fournit à chaque acteur un dispositif qui a pour première finalité l'accès à distance au contenu pédagogique, l'auto apprentissage, l'auto évaluation et la télé tutorat via l'utilisation des moyens de travail et de communication à plusieurs : vidéoconférence, e-mail, forums, chats, etc.

Le but est donc de combler la perte de cohésion et de stimulation de la salle que peut sentir l'apprenant devant sa machine.

L'usage de ces systèmes est relativement standard, le tuteur crée des parcours de formation typé, incorporé des ressources pédagogiques multimédias et de suivi des activités des apprenants.

L'apprenant, peut consulter en ligne ou télécharger les contenus pédagogiques qui lui sont recommandés, effectuer des exercices, s'auto évaluer et transmettre des travaux à son tuteur pour les corriger.

La communication entre apprenant et tuteur peut être individuelle ou en groupe. Il est possible de créer des thèmes de discussion et collaborer à des travaux communs en utilisant des moyens de travail et de communication à plusieurs. L'administrateur, de son côté, assure l'installation et la maintenance du système, gère les droits d'accès, crée des liens vers d'autres systèmes et ressources externes.

Ainsi, une plate-forme peut comporter des fonctionnalités relatives à la gestion des compétences, à la gestion des ressources pédagogiques, à la gestion de la qualité de la formation, etc. Avec l'évolution des techniques, des infrastructures de réseau et des normes, le nombre de plateformes et environnements de formation ont augmenté de manière significative. ^{[4]p2}

Par la suite, nous mentionnons les critères de sélection des plates formes.

6. Les principaux critères que doivent vérifier les plates-formes

- ✓ Le coût total de la mise en place de la plate-forme (gratuite ou non). Dans le cas d'une plate-forme payante, vérifier les prestations fournies.
- ✓ Les besoins de maintenance de la plate-forme d'un point de vue strictement informatique, tout ce qui devra être géré par les administrateurs (voir également le coût en personnel de maintenance).
- ✓ La sécurité de la plate-forme d'un point de vue purement informatique et au niveau des cours (login, mot de passe).
- ✓ Le nombre d'utilisateurs maximum que pourra supporter la plate-forme.
- ✓ La possibilité de mettre en place des QCM (Questions à Choix Multiples) sur la plate-forme, s'il est possible d'insérer une image, si le système de questions est interactif (avec animations), s'il y a possibilité de faire un tirage aléatoire des questions.
- ✓ Un forum de discussion (dont l'accès sera étroitement lié à la sécurité interne de la plateforme).
- ✓ La technologie utilisée, du point de vue du langage pour pouvoir éventuellement faire évoluer le système et sa capacité d'accueil (XML, MySQL, PHP...).
- ✓ Un système de gestion de mails, pour qu'un professeur puisse éventuellement envoyer des mails à l'une de ses classes, également lui permettre de gérer des groupes.
- ✓ La façon de poster un cours (à partir de documents PowerPoint, enregistrer au format HTML, différents formats).
- ✓ Les moyens requis pour consulter la plate-forme (browser, OS).
- ✓ La possibilité de diffuser les cours par de la vidéo soit en temps réel, soit en différé. ^{[4]p2}

7. Outils de communication

Généralement les plates-formes intègrent un certain nombre d'outils de communication. Ces outils sont souvent très usités sur le Web et ne sont en rien spécifique au E-Learning, ils ne nécessitent pas de configuration spécifique sur les postes des apprenants (par exemple, il n'est pas nécessaire de configurer un client messagerie pour profiter de ce service). On peut citer :

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

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

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

Les documents pédagogiques partagés (Une zone commune peut être utilisée pour mettre à disposition du groupe différents documents : Documents sous forme numérique (Word, Excel, PDF, etc.) pour l'ensemble du groupe,

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

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

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

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

Les FAQs (Cet espace permet aux tuteurs et aux administrateurs d'industrialiser la réponse à des questions récurrentes).^{[4]p3}

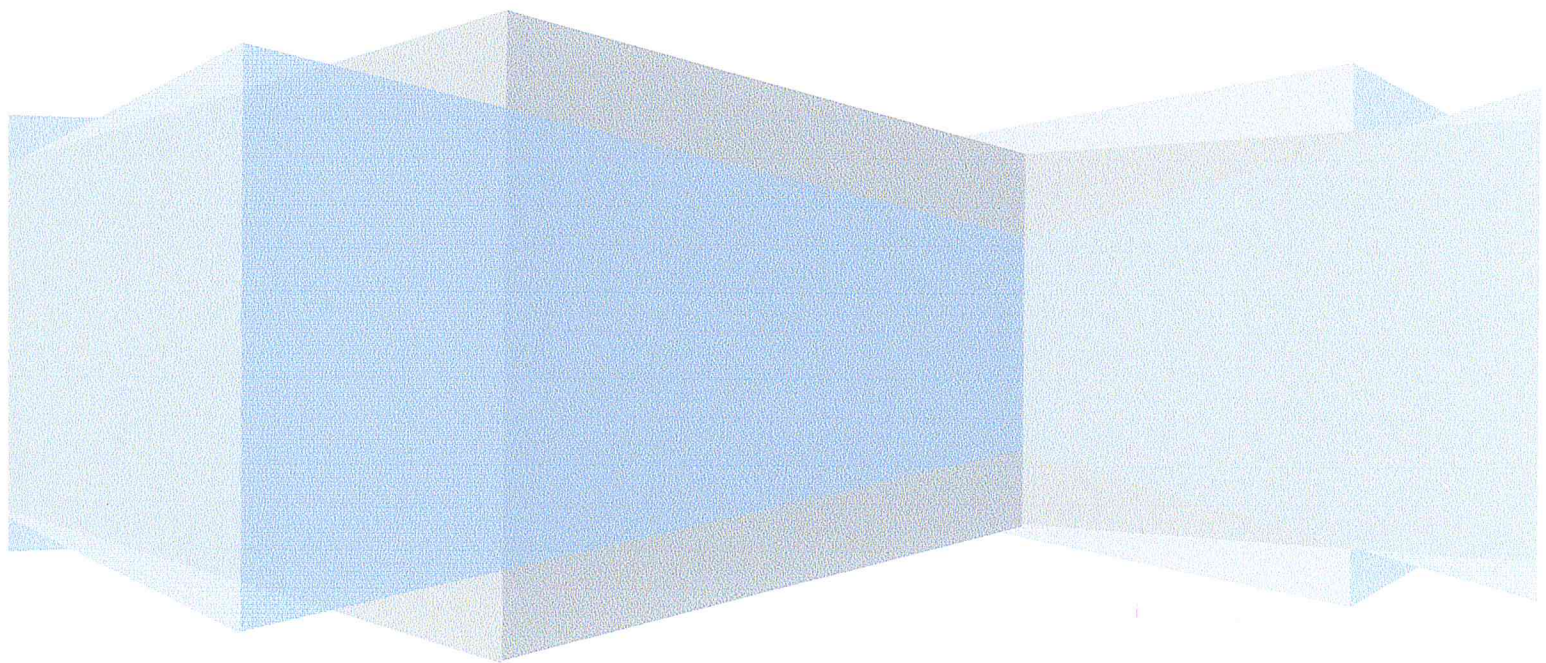
8. Conclusion

La fonction principale des plateformes e-Learning est de fournir à l'utilisateur les bonnes activités avec les bons outils au bon moment en fonction de ses besoins.

Le e-Learning promet de révolutionner la formation en réduisant les contraintes de temps et d'espace, ce qui peut conduire à une diffusion rapide du savoir. Les outils et les plates-formes du e-Learning progressent rapidement et visent à donner aux enseignants et élèves les mêmes réflexes que ceux qui existent dans le monde physique.

Chapitre 2

Le CMS Joomla et le MVC



1. Introduction:

Avant de développer des extensions, il est très utile d'aborder c'est quoi Joomla et comprendre la structure des composants, des modules, plugins et ainsi les fonctions de base d'un site créer avec CMS Joomla et model MVC, et pour le faire nous devons d'abord connaitre c'est quoi un CMS en générale.

2. C'est quoi un CMS?

Un CMS est un systeme de gestion de contenu (Content Management System), c'est une famille de logiciels destinés à la conception et à la mise à jour dynamique de sites Web ou d'applications multimédia.

Il existe des centaines de CMS dans le monde de l'Open Source proposés gratuitement par des passionnés qui rivalisent d'ingéniosité pour offrir aux apprentis webmasters les outils qui les aideront à réaliser leurs sites. Chacun choisira le plus proche de son besoin immédiat.

Un CMS est en général composé d'un « noyau » (architecture du logiciel) avec les fonctionnalités de base (faire des pages, gérer les utilisateurs, etc...) sur lequel se greffent des modules (ou extensions) spécifiques comme par exemple un forum, une newsletter, annuaire, glossaire, actualité, et bien d'autres choses.

Doués de qualités absolument remarquables, les CMS ne souffrent pas moins de certains problèmes qu'il faut absolument connaitre avant de se lancer dans la production d'un site avec un outil de travail (logiciel) de ce type et surtout de le mettre en production en ligne utilisable par le public et si on veut choisir un CMS, il faudra donc passer beaucoup de temps à analyser les besoins de son site, ainsi que les solutions fournies par chaque CMS. C'est évidemment plus facile à dire qu'à faire, car même si on a une bonne idée des fonctionnalités que l'on désire implémenter sur son site, celles-ci peuvent évoluer avec le temps.

Pour cela la première chose à faire une fois que vous aurez trouvé le CMS qui correspond aux besoins de votre site aujourd'hui mais aussi à ses besoins demain vous devrez continuer à apprendre continuellement sur ses capacités car plus vous connaîtrez le fonctionnement de votre CMS plus vous vous rendrez compte qu'au final le choix d'un CMS est dérisoire vu qu'avec un peu de débrouille et de savoir-faire vous trouverez toujours une magouille pour arriver à vos fins quelques soit l'outil choisi et simplement certains objectifs seront plus facile à réaliser avec tel ou tel CMS .

Joomla! propose tout ce dont vous avez besoin pour créer votre propre site web personnel. C'est un outil comprenant beaucoup de possibilités et vous pouvez utiliser le système dans une grande variété de configurations, en fonction de vos idées et de vos souhaits.^[A]

3. Historique Joomla:

Le mot Joomla! est dérivé du mot « Jumla » qui signifie “tous ensemble” en Swahili, une langue Africaine. Le choix s'est porté vers ce nom car l'intégralité de l'équipe de développement était unanime dans la volonté de protéger les intérêts des créateurs et de la communauté. [5]p5

Le Projet Joomla! est le résultat d'une discussion animée entre la Fondation Mambo, fondée en Août 2005, et son équipe de développement. Joomla! est une évolution du brillant système Mambo. Joomla! est utilisé de partout à travers le monde pour des sites allant de la simple page personnelle à des sites beaucoup plus complexes développés pour des entreprises. Il est très facile à installer, facile à gérer et très fiable.

L'équipe Joomla! s'est organisée et réorganisée au cours de ces 5 dernières années.

- De 2005 à 2009, Joomla 1.0 a été développé jusqu'à la version 1.0.15 et ce développement a officiellement été stoppé en Septembre 2009.
- De 2005 jusqu'à aujourd'hui, Joomla 1.5 est toujours développé, il a été présenté comme version stable en janvier 2008 et officiellement stoppé en avril 2012.
- De 2008 jusqu'à 2011, Joomla 1.6 a été développé. Une version stable est disponible depuis Janvier 2011.
- En Juillet 2011, Joomla 1.7 est sorti.
- Joomla 2.5 en tant que première version de support à long terme (Long Term Release LTR) depuis Joomla 1.5 est sorti en Janvier 2012.

Les utilisateurs de Joomla! sont restés. Beaucoup d'entre eux ont migré leur site web de Mambo vers Joomla! et ils ont appris beaucoup au fil des ans. De nombreux utilisateurs sont montés à bord ces dernières années mais il reste encore des gens dans le monde qui ne connaissent pas ce système. Joomla!, avec Drupal et Wordpress, est l'un des Systèmes de Gestion de Contenu (en Anglais: Content Managment System) les plus utilisés dans le monde. [6]p15

4. Description Joomla:

Joomla! est un système de gestion de contenu, open source, qui est écrit en langage de script PHP et qui utilise MySQL comme base de données. Ce CMS peut être utilisé pour développer toute une gamme d'applications Web allant du simple site Web, aux applications web les plus complexes. Joomla! a une architecture modulaire fiable et souple, et offre des interfaces faciles à utiliser. Les interfaces installées dans le navigateur peuvent être utilisées pour gérer facilement tous les aspects du site.

Les meilleures caractéristiques qui font de Joomla la meilleure solution pour la conception web sont :

- **Fonction de recherche :** Basé sur le Finder, qui a été développé par JXtended, Joomla 2.5 a ajouté une toute nouvelle fonction de recherche en langage naturel. Cette fonction utilise l'auto complémentation: retrait du suffixe pour rechercher toutes les occurrences en rapport avec le mot, même dérivées. Cette nouvelle recherche est plus rapide et plus polyvalente que la recherche standard. Les données sont indexées pour obtenir cette souplesse et cette rapidité.

- **Capacité d'utiliser d'autres bases de données :** Joomla peut actuellement s'interfacer avec les bases de données MySQL et MS SQL, avec PostgreSQL, Oracle. SQLite et PDO seront disponibles bientôt.

- **Notifications de mise à jour de Joomla :** joomla.org est la possibilité pour Joomla d'alerter l'administrateur du site lorsque celui-ci doit être actualisé, Ce dernier sera désormais en mesure de voir, dès qu'il se connecte si sa version de Joomla est obsolète. Il pourra alors cliquer sur le bouton et être guidé pour la mise à jour "en un clic" de Joomla.

- **Corriger la version en base de données :** Une nouvelle fonctionnalité a été ajoutée qui va vérifier si votre base de données a été mise à jour avec les modifications des schémas ajoutés lors des mises à jour de Joomla. Si ce n'est pas le cas, un message vous en informe et vous pouvez mettre à jour la base de données en cliquant sur bouton «Corriger» disponible dans la barre d'outils.

- **CAPTCHA de base :** Le CAPTCHA utilise des textes brouillés pour empêcher les robots spammeurs de vous inonder de messages plus ou moins publicitaires.

- **Lier de nouveaux menus avec un module :** vous aurez un lien dans le gestionnaire de menu qui vous permettra de créer un nouveau module pour ce menu. Vous avez alors le choix de sauver ce module menu standard ou d'annuler pour utiliser un module différent.

- **Un chargeur automatique pour le cms :**La plateforme Joomla est devenu une entité distincte, le CMS avait besoin d'un endroit pour mettre des fichiers / classes qui sont universels pour le CMS, mais qui ne sont pas une partie de la plateforme. Avec ce chargeur automatique, les développeurs n'auront pas besoin de s'inquiéter de savoir si la classe dont ils ont besoin se trouve dans les fichiers plateforme (librairies/joomla) ou dans les fichiers de CMS (librairies/CMS).^[B]

5. Les interfaces dans Joomla :

Pour adapter le contenu aux utilisateurs ou pour gérer un site et partager les tâches que cela implique, on doit attribuer à des groupes des accès et des droits d'exécution différents.

Chaque utilisateur enregistré sur le site possède un identifiant unique (nom d'utilisateur) et un mot de passe pour se connecter, ce qui permet de l'assigner à un groupe spécifique.

Joomla est constitué de deux interfaces distinctes :

- **Frontal (*frontend*)** : partie visible du site, par laquelle il est aussi possible de créer et gérer des articles ainsi que les contenus de certaines extensions.
- **Administration (*backend*)** : partie réservée à la configuration globale, à la structure des contenus, la gestion des utilisateurs, des extensions, etc.

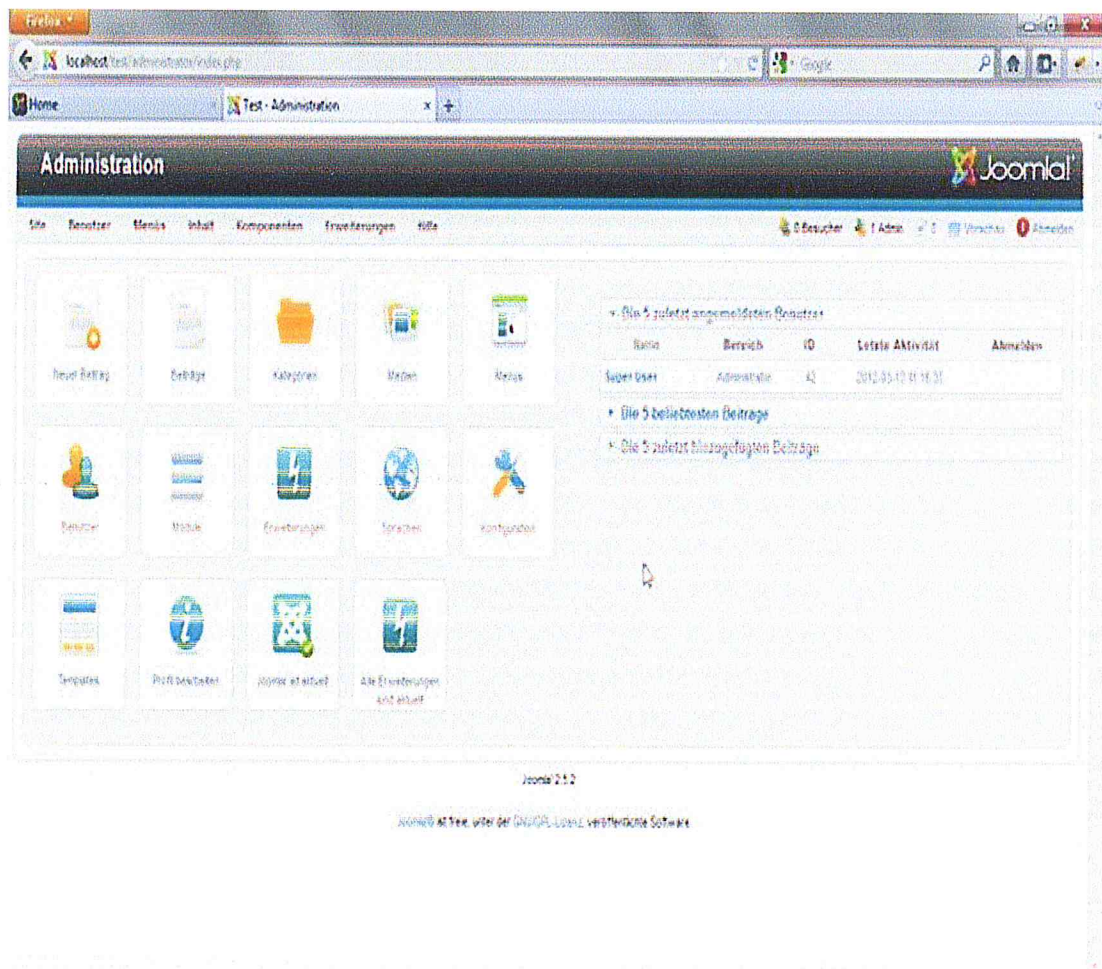


Figure 2: exemple d'interface d'administration Joomla 2.5

6. Principe de fonctionnement :

Le principe est le suivant: À partir d'un logiciel installé sur un serveur web, on gère son site web. Contrairement à la technique classique, l'utilisation d'un CMS ne repose pas sur une synchronisation entre le site local (créé sur son ordinateur) et le site distant (hébergé sur un serveur web et disponible au public).

Sans connaissances particulières en programmation, Joomla! Permet de créer un site web et de le personnaliser plus facilement que par la manière classique (site dit «statique»).

Joomla ! est un logiciel écrit en langage php et fonctionne avec une base de données (BD). Son fonctionnement peut être résumé par le schéma suivant:

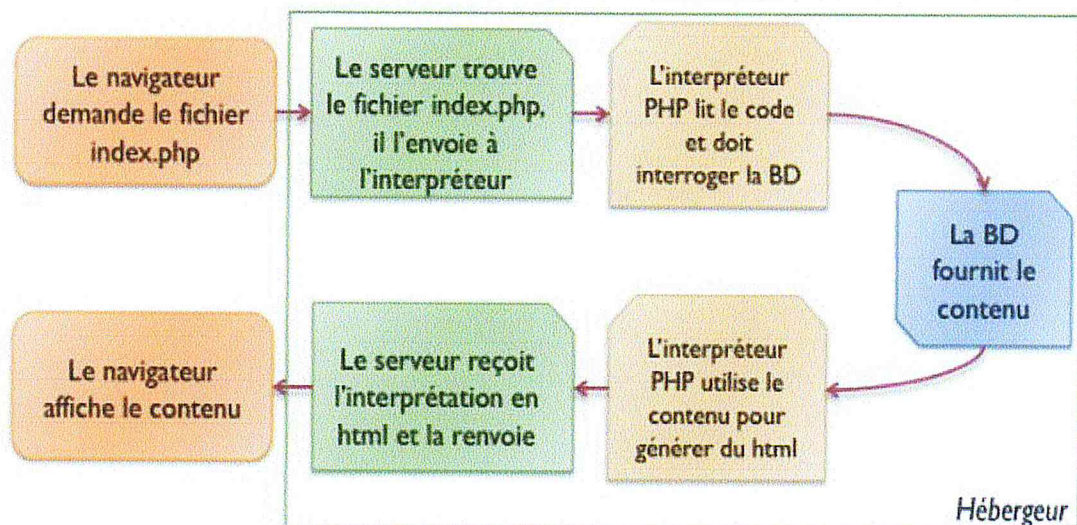


Figure 3 : Le fonctionnement de page web en PHP

L'idée principale à retenir de ce schéma est la dissociation entre «aspect» du site et «contenu» du site. Le contenu est stocké de manière brute dans la base de données, l'aspect est géré par le logiciel (principalement à l'aide des templates, ou chartes graphiques).^{[7]p4}

6.1. le contenu :

L'appellation "contenu" dans Joomla concerne presque exclusivement les articles et leur organisation, même si par extension ce terme regroupe toutes les zones de texte avec du formatage HTML .Le contenu du site est symbolisé par trois boutons principaux sur la page d'accueil de votre administration Catégories, Articles.

Catégories permettent d'organiser les articles du site de manière plus claire dans des rubriques.

Classer les articles du site permet de les afficher sur des pages réunissant par exemple plusieurs articles d'une catégorie. En publiant et dépubliant tout le contenu d'une catégorie, on peut également faire apparaître ou disparaître du site tout ce qu'elle contient.

6.2. les utilisateurs :

Avec un minimum d'apprentissage du fonctionnement de Joomla, vous comprendrez que la gestion des utilisateurs de Joomla fonctionnent depuis toujours avec :

- des utilisateurs.
- des groupes.
- des niveaux d'accès.

Si l'association des utilisateurs aux groupes semble simple d'approche, la notion de niveaux d'accès peut paraître plus complexe. Son paramétrage dans Joomla vous demandera donc de bien comprendre sa philosophie.

D'un coté nous avons les utilisateurs et les groupes, d'un autre coté nous avons les fonctions et les contenus dans Joomla.

Les niveaux d'accès servent de pont entre ces 2 mondes : les personnes et les données.

Dans une architecture de droits d'accès Joomla, nous avons donc :

- **des Utilisateurs** : les comptes des membres de votre site.
- **des Groupes** : qui regroupent les utilisateurs par fonction (auteurs, gestionnaires, administrateurs ...). Un utilisateur peut désormais faire partie de plusieurs groupes !
- **des Niveaux d'accès** : qui définissent quel groupe a accès (public, enregistré, spécial) a quel contenu (articles, catégories, modules ...)

- **des Permissions** : qui accordent des actions (login, création, suppression, configuration, publication ...) a des groupes d'utilisateurs.
- **des contenus** : toutes les données de votre sites qui peuvent soumis à la gestion des droits de Joomla (articles, catégories, modules ...) ^[C]

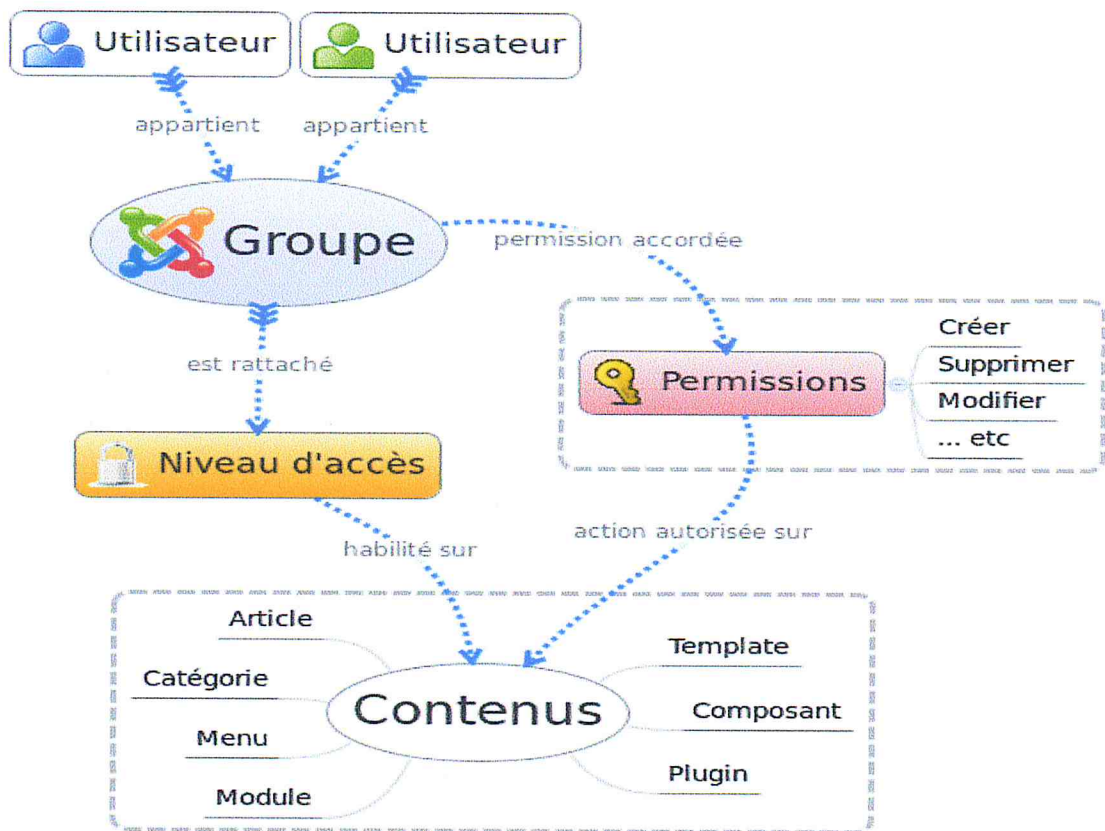


Figure 4 : Les droits d'accès dans joomla

6.3. 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 catégories: les composants, les modules, les plugins, et les Templates.

6.3.1 Les composants : 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, de même, il existe un composant pour authentifier les utilisateurs ...

Les composants constituent la majeure partie de vos pages ! Les composants de base sont fournis avec Joomla!. D'autres composants peuvent être installés par la suite (forums, livre d'or, galerie d'images, gestionnaire de newsletter, gestionnaire de formulaires... et bien d'autres encore).

Exemple : `com_content` (gestion des contenus) et `com_registration` (enregistrement des utilisateurs)

6.3.2 Les modules : 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...

Les modules sont souvent associés à des composants, comme par exemple le module qui affiche une photo aléatoire tiré d'une galerie d'images géré par un composant. Exemple : `mod_banners` (affichage des bannières), `mod_mainmenu` (affichage d'un menu).

6.3.3 Les plugins : ce sont des morceaux de code activés sur un événement. L'exécution de n'importe quelle partie de Joomla, que ce soit le noyau, un module ou un composant, peut déclencher un événement et alors les plugins associés à cet événement s'exécuteront. Par exemple, le plugin de recherche `content.searchbot` est un script qui permet d'effectuer une recherche sur tout votre site.

Autre exemple, quand vous saisissez le texte d'un article, le plugin TinyMCE permettra de convertir le texte que vous saisissez en code HTML interprétable par votre navigateur: un événement est l'appui sur une touche du clavier, l'action est la génération du code html adéquat.

6.3.4 Le Template : Il rassemble tous les éléments qui constituent l'apparence d'un site. Il gère toute la partie graphique d'un 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 » d'un site. Il se constitue la plupart du temps :

- d'une page dynamique *index.php* qui structure la page en colonnes, en lignes, et grâce au langage HTML, et va intégrer le contenu de la base de données à cette structure au moyen du code PHP.
- d'une ou plusieurs feuilles de style (fichiers CSS) qui définissent la mise en forme du contenu : couleurs et polices de caractères, fonds, aspect des cadres, des tableaux, des menus, etc.
- de fichiers images appelés pour illustrer ou fabriquer la page : images de fond de page, de fond de tableau, logo, bandeaux, etc.^[D]

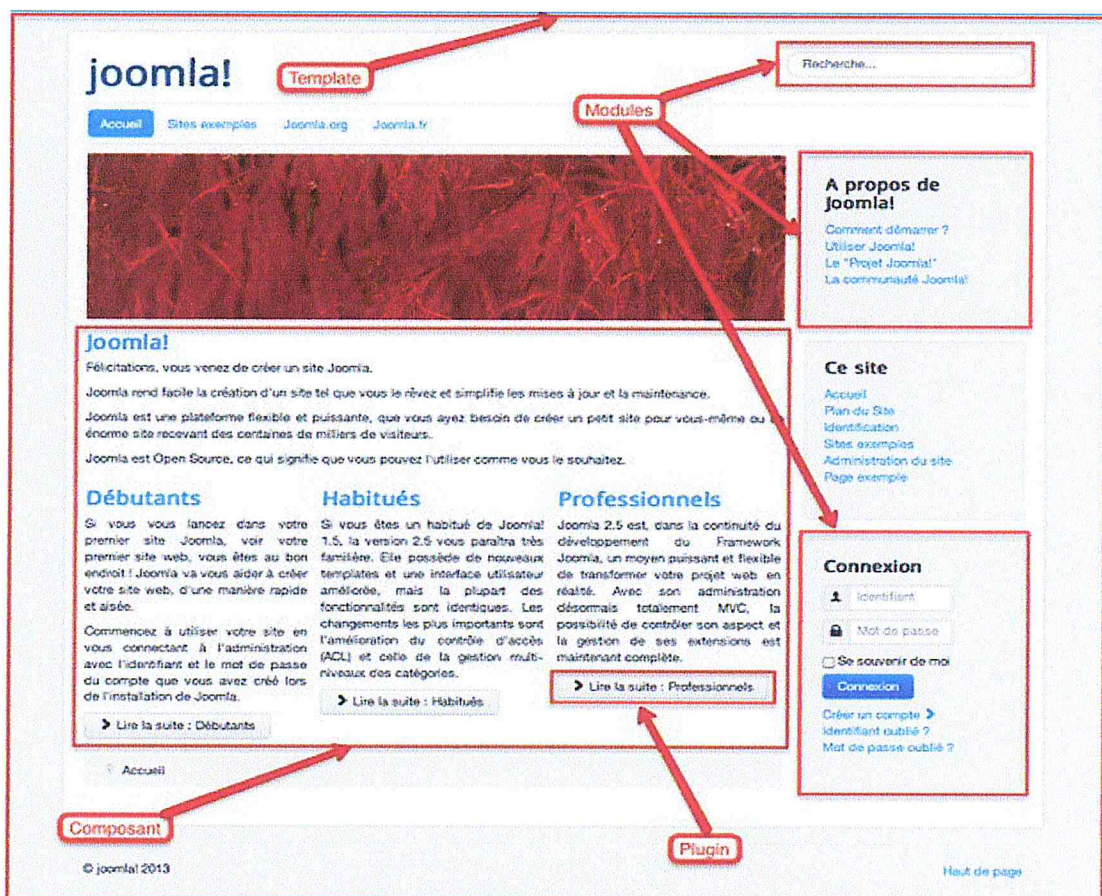


Figure 5 : Le positionnement des éléments dans page web joomla

7. Le model MVC :

Joomla! est construit sur l'architecture modèle-vue-contrôleur (MVC) qui a été décrite la première fois pour les interfaces utilisateur d'un langage de programmation appelé Smalltalk en 1979.

Aujourd'hui MVC est de facto le standard dans le développement logiciel. Il peut paraître sous différentes formes, le flux de commande est généralement comme suit:

- L'utilisateur interagit avec l'interface utilisateurs d'une manière quelconque (par exemple, en cliquant sur un bouton de soumission).
- Le contrôleur manipule l'événement depuis l'interface utilisateurs, et le convertit en action d'utilisateur appropriée, compréhensible pour le modèle.
- Le contrôleur avise le modèle de l' action de l'utilisateur, qui peut entraîner un changement dans l'état du modèle. (Par exemple, le contrôleur met à jour la liste de Maisons de l'utilisateur.)
- Une vue interroge le modèle afin de générer une interface utilisateur approprié (par exemple, la vue liste les listes de maison). La vue obtient ses propres données du modèle.
- L'interface utilisateur attend les interactions utilisateur supplémentaires, qui redémarre le cycle de flux de commande.

Dans Joomla!, le modèle MVC est implémenté à l'aide de trois classes : JModel, JView et Jcontroller.

L'intérêt principal d'un modèle MVC est de séparer les couches de conception, de manière à ce que plusieurs groupes de personnes puissent travailler chacun sur sa couche, sans même qu'ils connaissent les personnes qui utilisent la couche du dessous (ou du dessus, ou d'à côté).

Il n'est cependant pas toujours facile de bien distinguer les couches, et le motif MVC reste sans aucun doute un motif complexe car, les développeurs habitués à ce modèle apprécient particulièrement la lisibilité du code. ^{[8]p20}

7.1. Principe du modèle MVC :

Le but de MVC est de séparer les couches d'une application (en 3 couches distinctes, au minimum, et le plus souvent). On va donc distinguer :

1. **Le modèle** : qui encapsule la logique métier et la manipulation des sources de données. Ici, des designs patterns peuvent intervenir.
2. **La vue** : présente les données au client les demandant (un utilisateur humain ou un autre programme). Ici aussi on peut introduire des motifs : Factory , Composite, TemplateView; par exemple
3. **Le contrôleur** : c'est tout le reste, c'est lui qui analyse la requête du client, accède aux données, formate le tout et l'envoie à la partie vue qui va présenter tout ça.

On va détailler un peu, la vue d'abord, c'est un TemplateView. On utilise PHP lui-même pour écrire nos variables dans du HTML , ou alors un moteur de template.

Le TransformView peut aussi être utilisé pour sortir en transformant les données renvoyées par les modèles, qui sont en XML, grâce à XSLT et les feuilles de style XSL. Une sortie en PDF peut aussi être utilisée ou encore une présentation JSON des données, dont les clients JavaScript et Ajax sont très friands.

Il est très important de noter que la vue ne fait que présenter des données et à aucun moment elle ne connaît leur provenance. C'est le modèle qui va fournir toutes les méthodes nécessaires à la récupération, et le contrôleur qui se chargera de cette récupération et de sa transformation éventuelle afin de coller à la vue.

Du côté du modèle, il est très important que celui-ci ne connaisse ni les contrôleurs, ni les vues. Les liens vont des contrôleurs vers le modèle, de temps en temps, des vues vers le modèle, mais jamais dans le sens inverse.

Le modèle étant totalement indépendant du reste, il pourra faire l'objet d'une réflexion et d'un développement à part. Il pourra de même être utilisé par d'autres composants, tels que des web services ou d'autres applications locales.

Les données manipulées par le modèle doivent persister entre toutes les requêtes HTTP, ainsi un support de stockage devra être utilisé. Dans la majeure partie des cas, la source de données est une base de données relationnelle et le mapping objet.

Il est possible que la vue accède au modèle. C'est un choix architectural qui devra être bien réfléchi, car il impose un plus fort couplage et un mélange des couches. On appelle ça l'approche pull. C'est souvent réservé aux opérations légères, comme par exemple afficher une liste de catégories d'éléments quelconques ; et aux opérations de lecture uniquement.

L'approche inverse, appelée push, interdit à la vue d'accéder au modèle. En général, on utilise les 2 choix en même temps : une instruction de lecture simple (récupérer la liste des membres) sera demandées directement par la vue, ou un de ses sous-composants. En revanche une opération de modification du modèle, sera demandée par un contrôleur.

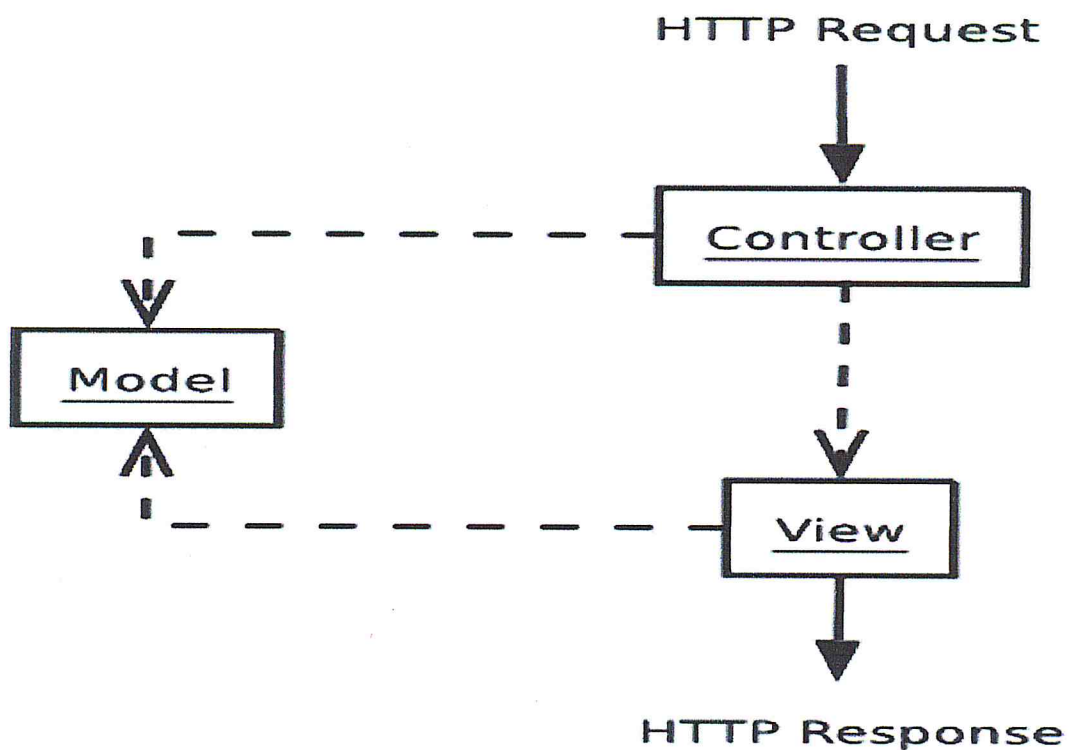


Figure 6 : modèle MVC global

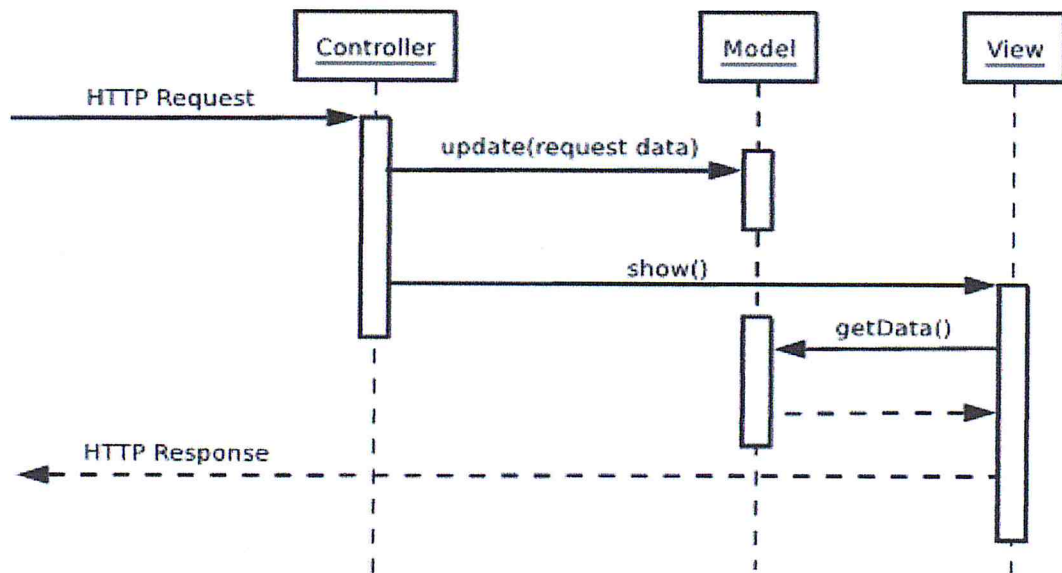


Figure 7 : diagramme de séquence d'un MVC simple

Et on arrive enfin au **contrôleur**, c'est le composant principal du modèle MVC. Il agit avec le modèle, la vue et d'autres contrôleurs tout à la fois.

Quel qu'il soit, le contrôleur reçoit et analyse la requête du client, accède au modèle, et présente la vue. [E]

7.2. Illustration du modèle MVC:

Le modèle MVC (Modèle-Vue-Contrôleur) cherche à séparer nettement les couches présentation, traitement et accès aux données. Une application web respectant ce modèle sera architecturée de la façon suivante :

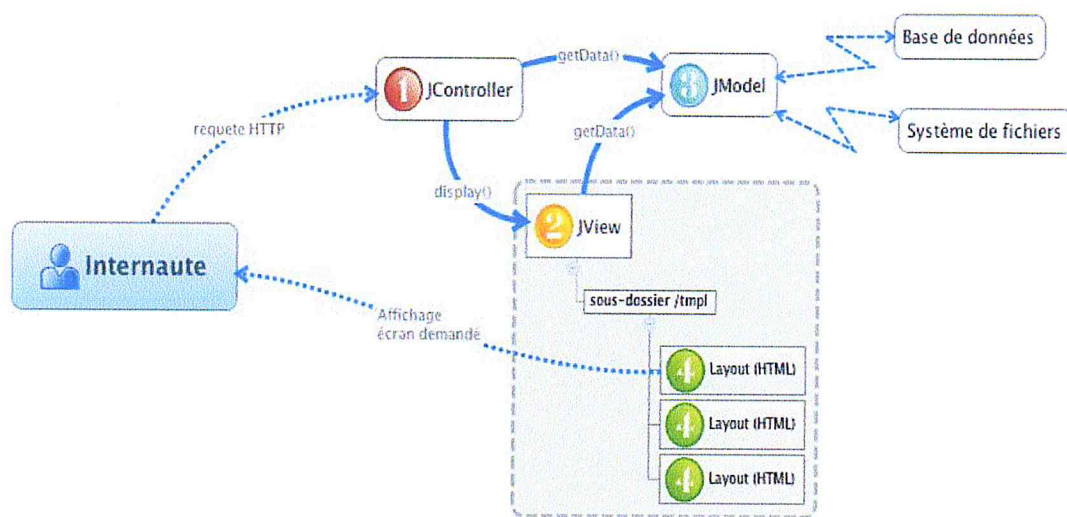


Figure 8 : fonctionnement d'un MVC dans le web

Une requête HTTP (URL) est chargée par un internaute

1) Votre classe contrôleur qui hérite de Jcontroller (la classe fournit par le framework) va traiter la demande afin de contrôler

- si toutes les conditions sont requises
- qualifier le chargement de la vue et/ou du modèle demandé

2) la vue charge les données nécessaires en appelant le modèle

3) le modèle accède aux données demandées et les fournis à l'appelant (la vue)

4) la vue renvoie vers le Layout demandé. Le layout s'exécute en formatant le rendu des données au format HTML puis le résultat HTML est renvoyé à l'appelant (l'internaute).^[F]

8. Etapes de création d'un composant Joomla:

Avant de vous attaquer à la création de composants sous Joomla, il vous faudrait commencer par étudier ce CMS. De nombreux tutoriels se trouvent sur Internet à ce sujet :

- Joomla Facile : <http://joomlafacile.com/>
- Joomla Help : <http://help.joomla.org/>
- ...

Il vous faudra aussi maîtriser le langage SQL, qui vous permettra d'attaquer une base de données (en l'occurrence, celle de Joomla). Et le plus important : Il vous faut des connaissances solides du PHP et du HTML.

8.1. L'Articulation:

Un composant s'articule d'une manière particulière, il contient des fichiers précis et Les scripts seront regroupés dans un ZIP selon une arborescence spécifique:

```
Com_exemple
manifest.xml
/admin
  manifest.xml
  install.mysql
  uninstall.mysql
  exemple.php
  controller.php
/models
  exemple1.php
/views
  /exemple1
    view.html.php
  /tmpl
    default.php
```

```
/site
  exemple.php
  controller.php
  /models
    exemple1.php
  /views
    /exemple1
      view.html.php
      /tmpl
        default.php
```

Un fichier XML appelé « **manifest** » va décrire les informations d'installation. Il contient une entête déclarative, la liste des fichiers à installer et des informations d'administration.

```
<files folder="site">
  <filename>index.html</filename>
  <filename>exemple.php</filename>
  <filename>controller.php</filename>
  <folder>models</folder>
  <folder>views</folder>
</files>
<administration>
  <menu link="option=com_exemple" img="templates/khepri/images/menu/icon-16-user.png">Exemple</menu>
  <files folder="admin">
    <filename>index.html</filename>
  </files>
</administration>
```

La balise `<files folder="site">` décrit les fichiers à copier (ici pour la partie site frontend).

La balise `<administration>` contient des informations comme le lien du menu avec le nom de votre composant "**exemple**".

Le script d'entrée :

"**exemple.php**" sera le point d'entrée de notre composant, il doit porter le nom du composant, avec l'extension "**php**". C'est ce fichier qui sera appelé lorsque le serveur recevra une requête du type :

`http:// www.nomDuSite.com/index.php?option= com_exemple.`

Le contrôleur principal :

"**controller.php**" va gérer l'aiguillage des appels vers les bons modèles et vues. Il implémente toujours la méthode **display()** pour gérer les appels par défaut.

La vue :

La vue doit déclarer une méthode publique "**display()**" appelée par "**controller.php**". Le fichier de vue doit s'appeler "**view.html.php**" être placé dans le dossier "**com_exemple/site/views/ exemple /**".

Le Modèle :

Le fichier **/models/ecran1.php** doit permettre de gérer toutes les requêtes SQL et accès BD pour la vue du même nom : **exemple1**.

Le Template :

La dernière étape avant de pouvoir lancer un composant est de créer une page de Template, en affichant simplement le contenu de la variable **\$variable**, fournie par le modèle et transmise au vue.

Notre fichier va s'appeler **default.php**, puisqu'il s'agira de l'affichage par défaut. Il sera situé dans "**/site/views/ exemple /tmpl**".

Ce script affiche les données qui vont construire l'écran au format HTML.

Vous pouvez installer votre composant comme n'importe lequel, soit en indiquant à Joomla le chemin de votre dossier **"/exemple"**, soit en faisant un zip de votre dossier **"com_exemple"**.

Il suffit donc ensuite de vous connecter en admin, d'aller dans le gestionnaire d'extension, de cliquer sur "parcourir" dans l'encart "archive à transférer", indiquez votre fichier zip et cliquez sur "Envoyer et Installer".

9. Conclusion:

Joomla est un CMS (Content Management System) qui permet à des utilisateurs non expérimentés de gérer eux-mêmes leurs sites.

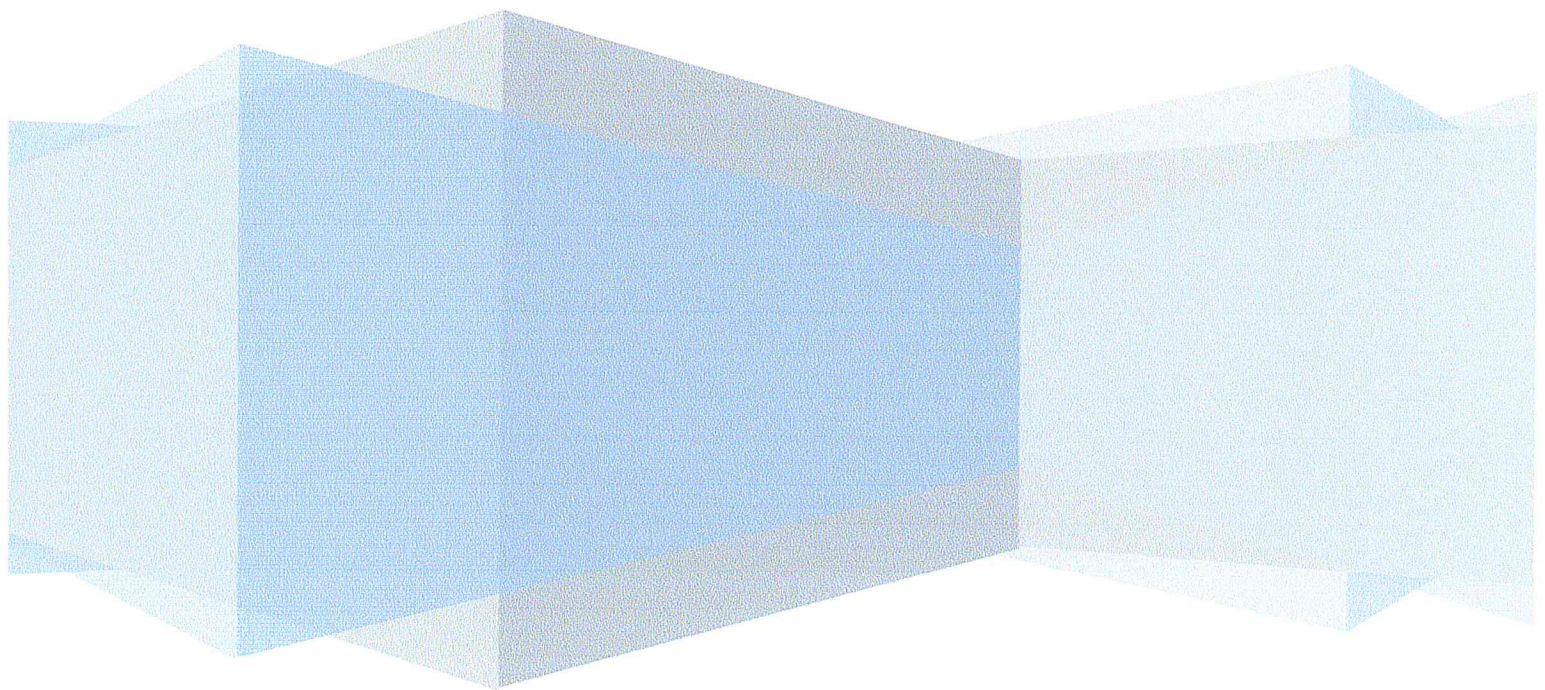
Joomla permet aux développeurs de créer des composants, des thèmes, des modules etc. Le but du composant est de créer une application au sein du système.

En effet, pour tout ce qui est gestion des actualités, des utilisateurs etc., on peut se reposer sur l'architecture joomla qui est très solide, pour mieux se concentrer sur la logique applicative à mettre en place.

Le modèle MVC (Model, View, Controller) correspond à un besoin de séparation entre les différentes facettes d'une application.

Chapitre 3

Etude conceptuelle



1. Introduction :

Nous allons modéliser notre système avec des diagrammes UML.

D'abord, nous allons traduire la liste des exigences en diagrammes de cas d'utilisation, qui permettent d'analyser et d'organiser les besoins, suivis par les diagrammes de séquences. Ensuite, nous allons étudier l'aspect statique avec les diagrammes de classes.

Mais tout d'abord, parlons d'UML :

UML (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>>. Utilisé surtout dans les projets logiciels, il peut aussi être appliqué à toutes sortes de systèmes et ne se limite pas au domaine informatique.

UML est né de la fusion des trois méthodes qui ont le plus influencé la modélisation objet au milieu des années 90. Ces méthodes sont l'OMT (Object Modeling Technique), Booch qui se réfère à son inventeur <<Grady Booch>> et OOSE « Object Oriented Software Engineering>>. Elaboré à partir d'un travail d'experts reconnus, l'UML est le résultat d'un large consensus.

A la fin de 1997, UML est devenu une norme OMG (Object Management Group) qui est un organisme à but non lucratif, son rôle est de promouvoir des standards qui garantissent l'interopérabilité entre des applications orientées et développées sur des réseaux hétérogènes. ^{[9] p14}

2. Analyse :

2.1. Diagrammes de cas d'utilisation :

Ces diagrammes permettent de recenser les grandes fonctionnalités d'un système, ils décrivent ce que le système devra faire, sans préciser comment il le fera. [10] p15

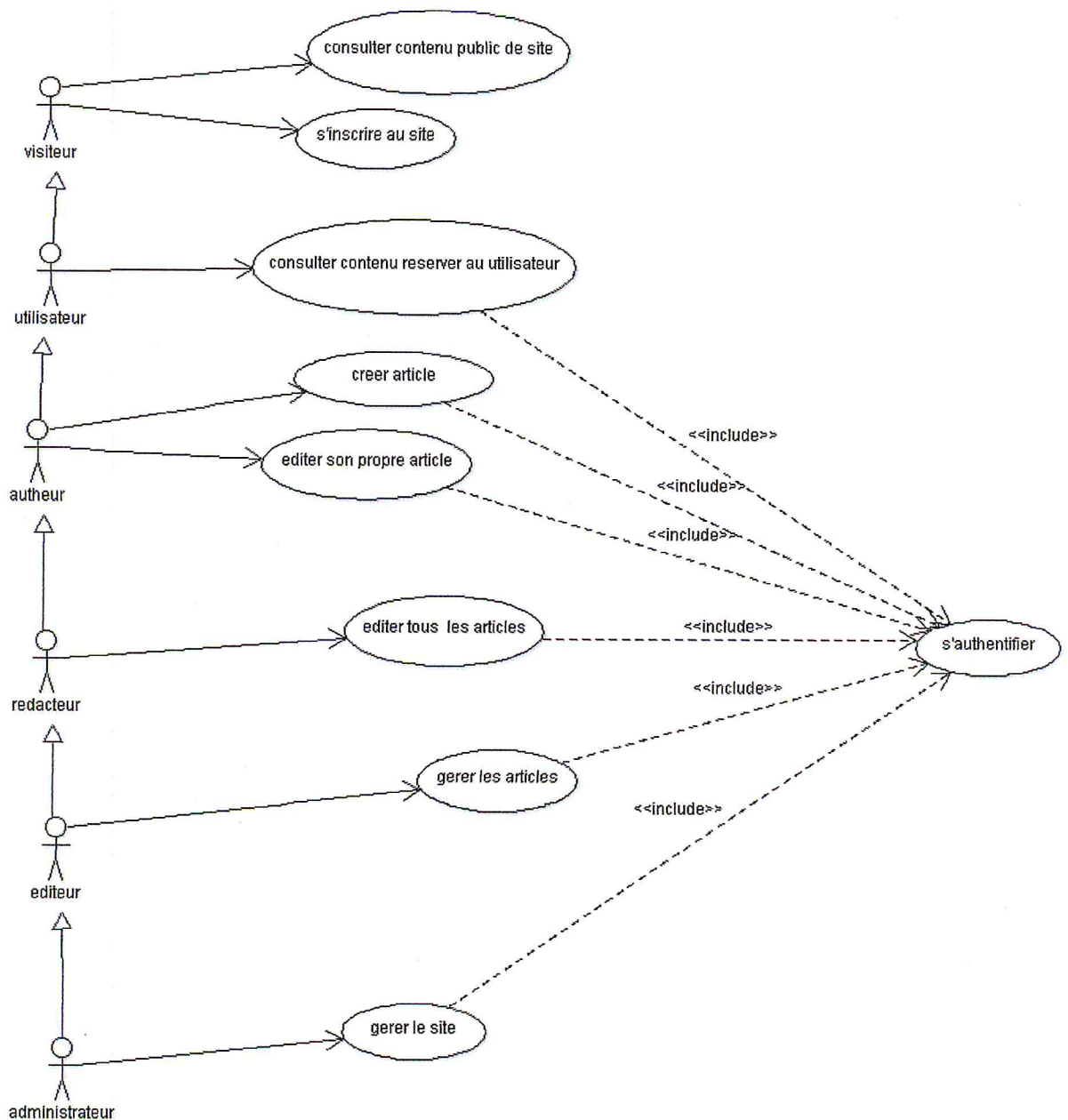


Figure 9 : Diagramme de cas d'utilisation globale

Les visiteurs ne peuvent que consulter les pages sur le site, via les différents menus.

Les utilisateurs peuvent visiter les pages du site, mais aussi consulter des pages qui possèdent un niveau d'accès défini par l'administrateur pour chacun des utilisateurs.

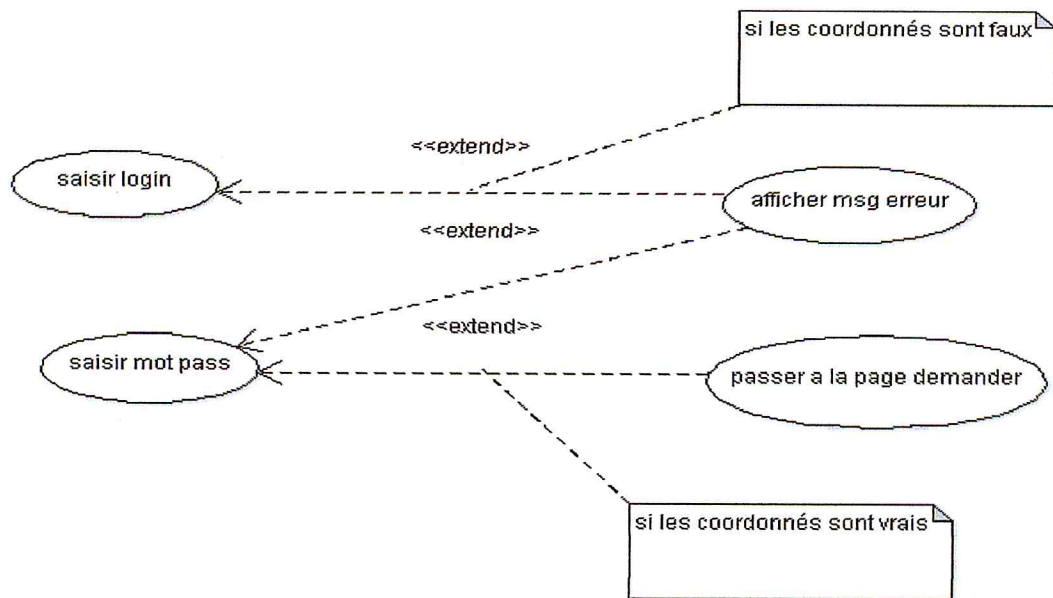


Figure 10 : premier niveau : Authentification

A ce niveau les utilisateurs peuvent s'authentifier en utilisant leur identifiant ainsi que leur mot de passe, si l'identifiant ou le mot de passe sont faux, un message d'erreur apparaît, sinon l'utilisateur a la possibilité d'accéder aux pages qui lui sont autorisé.

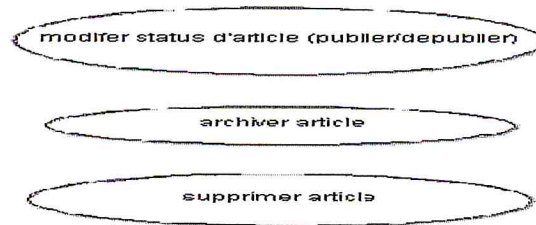


Figure 11 : premier niveau : Gérer tous les articles

Les éditeurs peuvent publier, dé-publier, archiver ou supprimer des éléments grâce à la permission « Modifier le statut ».

Les éditeurs sont enfants du groupe « Rédacteur ». Ils héritent donc des permissions « Modifier », « Créer », « Modifier ses éléments », et « Connexion au site »

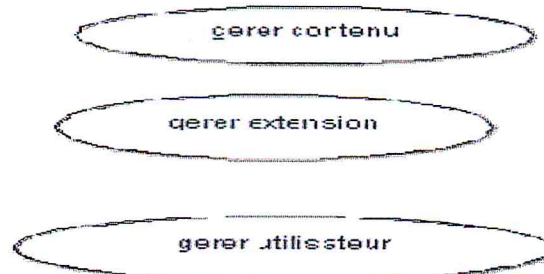


Figure 12: premier niveau : Gérer le site

L'administrateur peut gérer le site, il peut donc améliorer les performances en utilisant des extensions, ou bien gérer le contenu de site exemple : articles, gérer la notion des utilisateurs et leur affectation des niveaux accès.

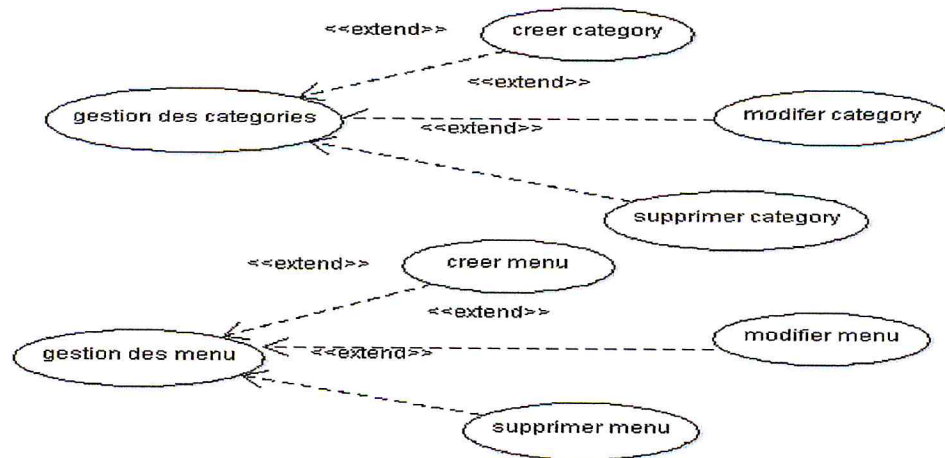


Figure 13 : second niveau : Gérer le contenu

La gestion du contenu s'exprime en différentes pages qui vont contenir différents articles qui peuvent être créés, modifiés ou supprimés. Cela est le même pour les catégories qui contiennent ces articles. La plupart du temps on accède aux différents articles par des liens dans des menus donc cela provoque un besoin de gérer les menus ainsi que leur lien.



Figure 14 : second niveau : Gérer les extensions

Pour améliorer le site de Joomla, l'administrateur a le droit d'installer des extensions, les configurer selon ses besoins ou même les désinstaller.



Figure 15 : second niveau : Gérer les utilisateurs

L'administrateur a le droit d'affecter un utilisateur à un groupe, modifier son statut ou même le supprimer.

2.2. Diagrammes de séquence :

Un diagramme de séquence est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique.

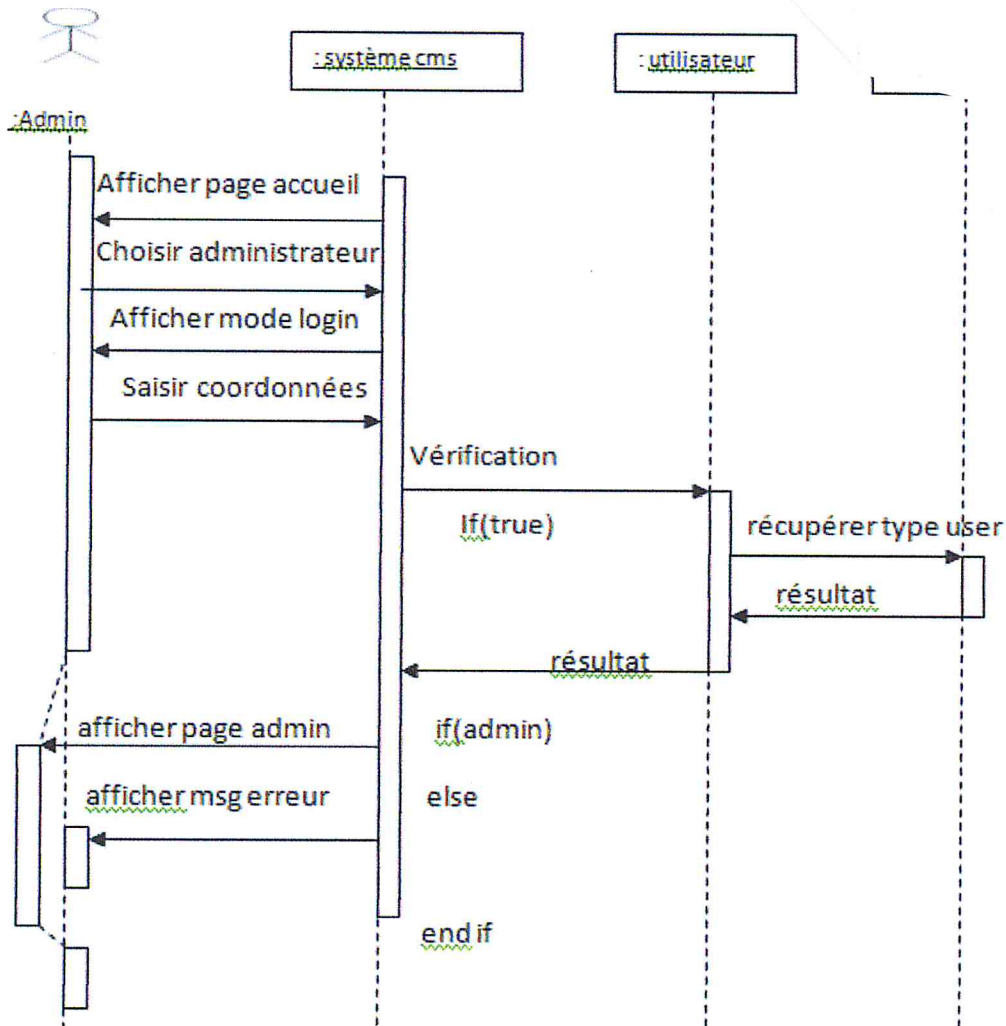


Figure 16 : Diagramme de séquence : Identification administrateur

Le système se connecte à la base de données pour vérifier si les informations rentrées sont exactes. Pour cela, il va vérifier dans la table users si le login existe, puis vérifier si le mot de passe entré est associé au login, ensuite il va consulter la table usergroupe pour vérifier l'appartenance du membre au groupe administrateur, Si c'est bon, alors l'administrateur est connecté, un message de confirmation va être affiché pour informer l'utilisateur qu'il est connecté au site.

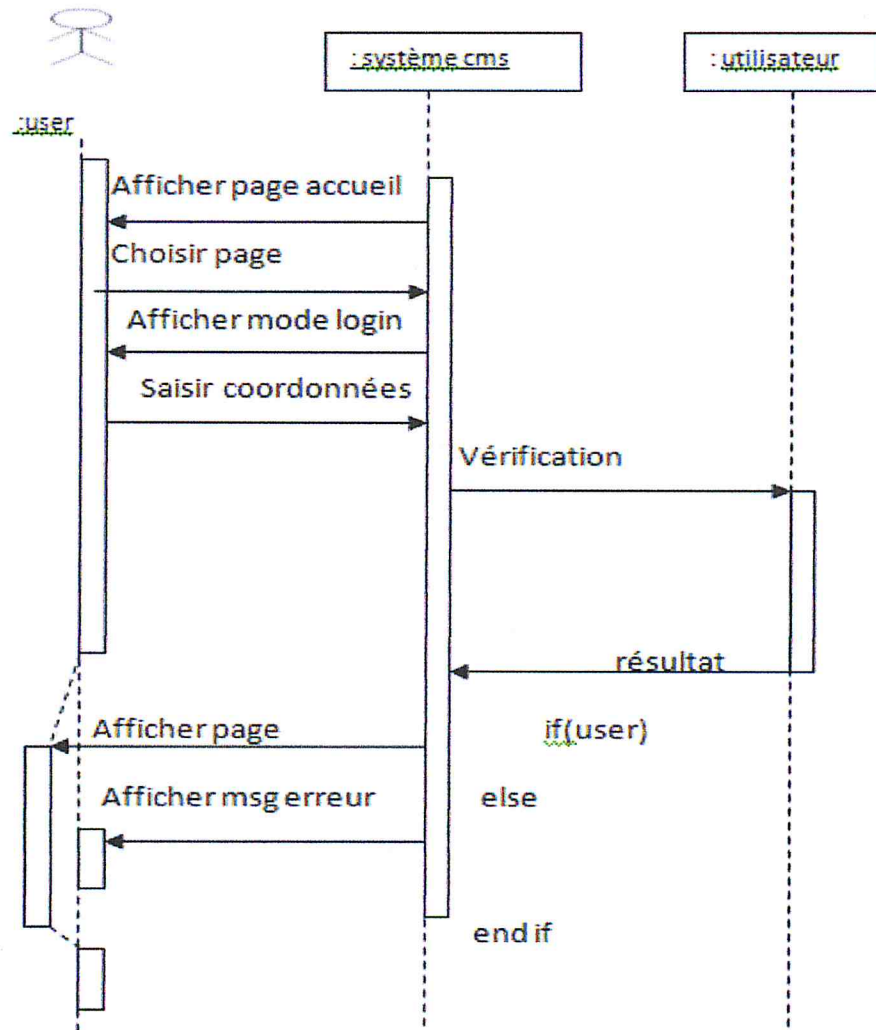


Figure 17 : Diagramme de séquence Identification utilisateur

Le système se connecte à la base de données pour vérifier si les informations rentrées sont exactes. Pour cela, il va vérifier dans la table users si le login existe, puis vérifier si le mot de passe entré est associé au login, Si c'est bon, alors l'utilisateur est connecté.

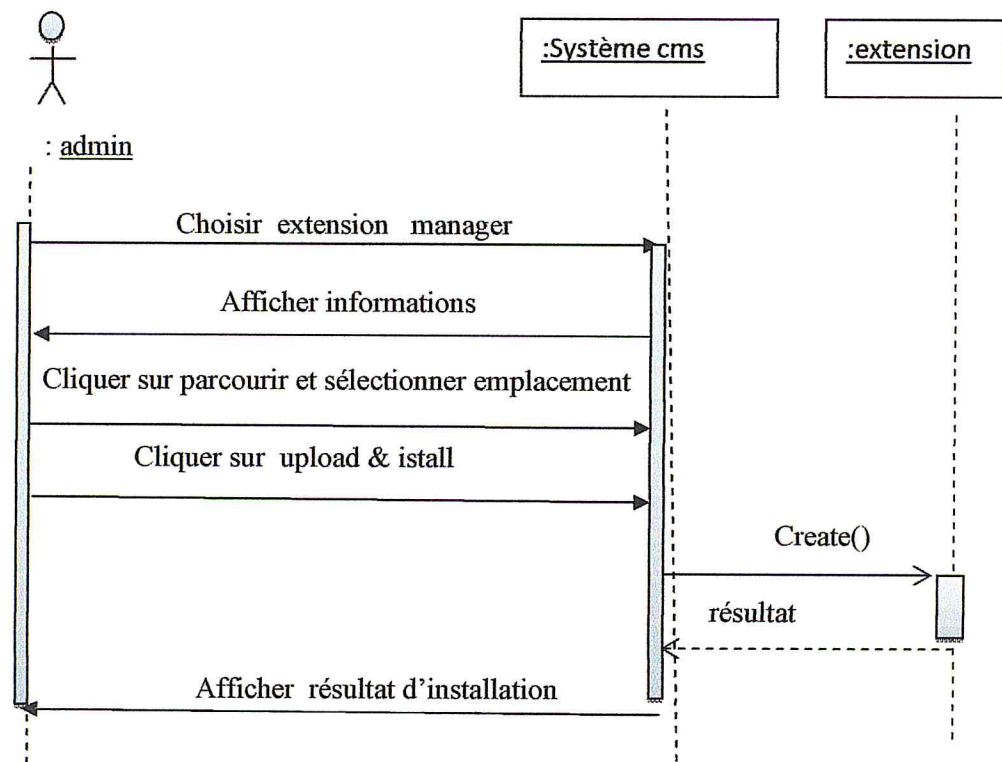


Figure 18 : Diagramme de séquence installation des extensions

Lorsque l'administrateur veut installer une extension, il doit cliquer sur l'icône extension manager, une page va s'apparaître avec des différentes informations, le premier onglet est associé à l'installation des extensions, l'administrateur doit parcourir les dossiers afin d'atteindre l'emplacement d'extension à installer, ensuite il doit valider l'installation, le système l'effectue en se connectant à la base de données pour insérer les nouvelles données dans la table extensions.

A la fin de l'installation, l'administrateur reçoit un message indiquant le résultat.

2.3. Diagrammes d'états transition :

Ces diagrammes utilisent le concept de machines à états finis pour s'intéresser au cycle de vie d'une instance d'une classe donnée et ses interactions avec le reste du système, et dans tous les cas possibles. [10] p90

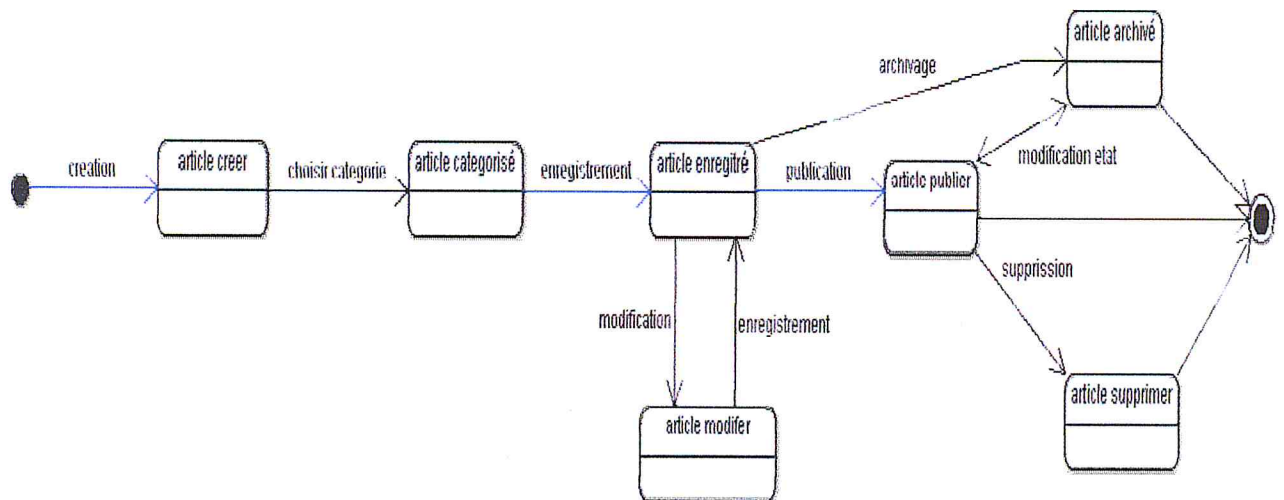


Figure 19 : Diagramme état transition : cas d'articles

2.4. Diagramme de classes :

Dans ce diagramme de classes, nous avons des utilisateurs enregistrés (user) chaque utilisateur est caractérisé par plusieurs attributs : un identifiant, un nom ainsi qu'un pseudo et un mot de passe afin de s'identifier, un email. Il peut consulter les articles (content) du site.

Chaque utilisateur est associé à un groupe d'utilisateurs par exemple : un utilisateur a le droit d'écrire des articles donc il est associé au groupe (auteurs).

La table user-groupe possède plusieurs attributs : un identifiant, un titre (type du groupe).

A chaque connexion d'un utilisateur une session lui y est associée avec un attribut identifiant et un attribut guest indiquant la nature de cet utilisateur (s'il est enregistré ou non).

Pour la table content (article), chaque article possède plusieurs attributs : un identifiant, un titre, une introduction du texte ainsi qu'un corps du texte, un attribut state qui indique si l'article sera affiché ou non, une date de création, un attribut hits qui sert à déterminer le nombre de fois que l'article a été consulté, un attribut Access qui sert à déterminer le niveau d'accès (ce niveau d'accès permet de savoir quel utilisateur pourra voir la page : soit les invités et les membres, soit uniquement les membres).

Les articles sont organisés selon des catégories, chaque catégorie possède comme attributs : un identifiant, un titre ainsi qu'un identifiant du créateur. On peut joindre chaque article par un lien menu qui est caractérisé par un identifiant, un titre ainsi qu'un attribut Link qui contient le chemin vers cet article.

Chaque menu possède un type-men caractérisé par un identifiant, un menu-type, un titre et une description du type.

Une table Admin qui hérite de la classe user.

Un administrateur peut créer, modifier et supprimer des pages ou des articles, ainsi qu'installer des extensions pour améliorer le site.

Chaque extension est caractérisé par : un identifiant, un nom, un manifest-cache qui englobe des informations relatif à cet extension ainsi qu'un type.

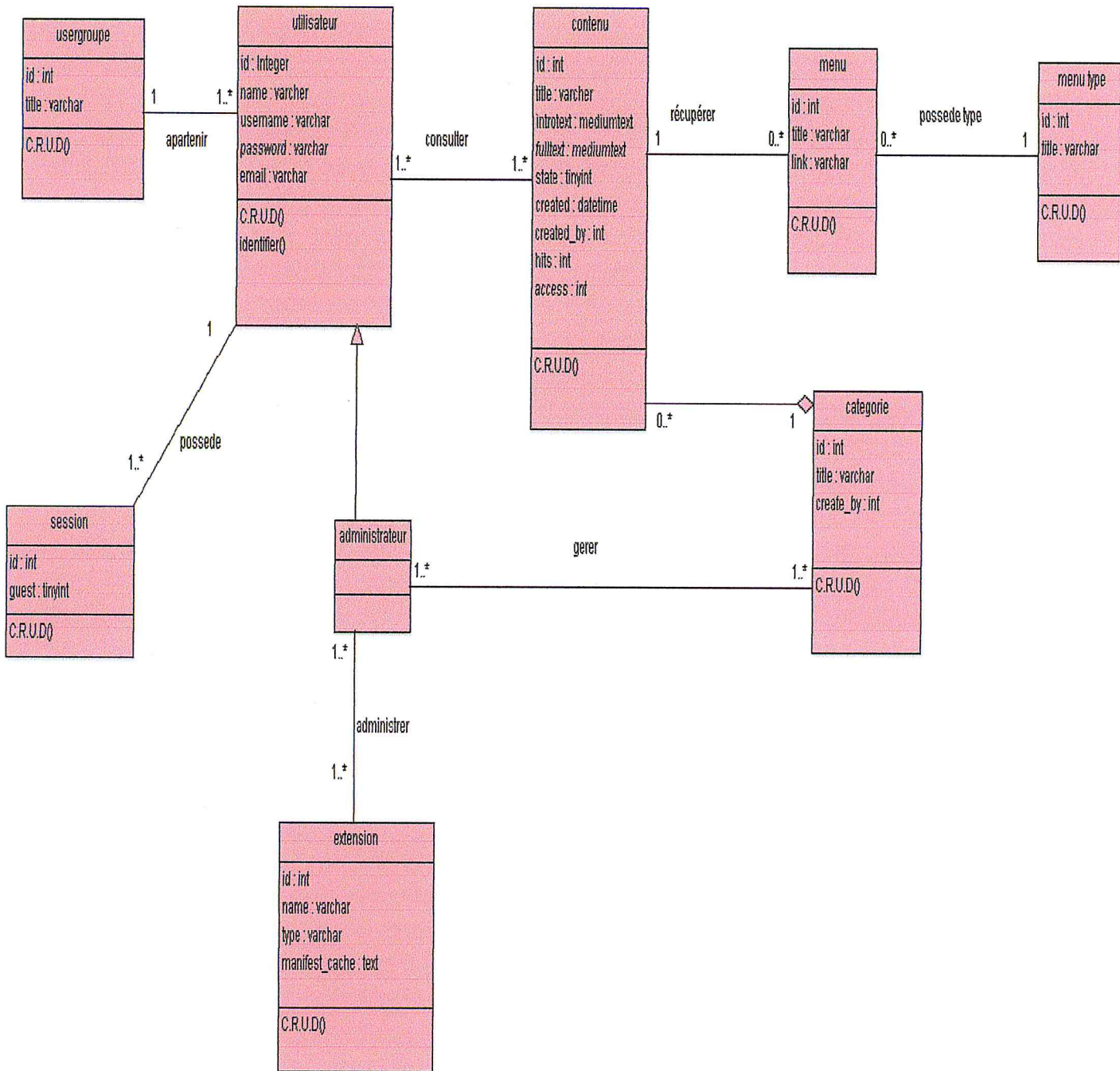
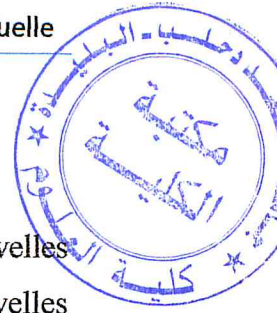


Figure 20 : diagramme de classes Analyse



3. Conception :

Dans cette partie on trouve le diagramme de classes qui contient les nouvelles classes qu'on a ajoutées ainsi que des diagrammes décrivant les nouvelles fonctionnalités.

3.1. Diagrammes de cas d'utilisation :

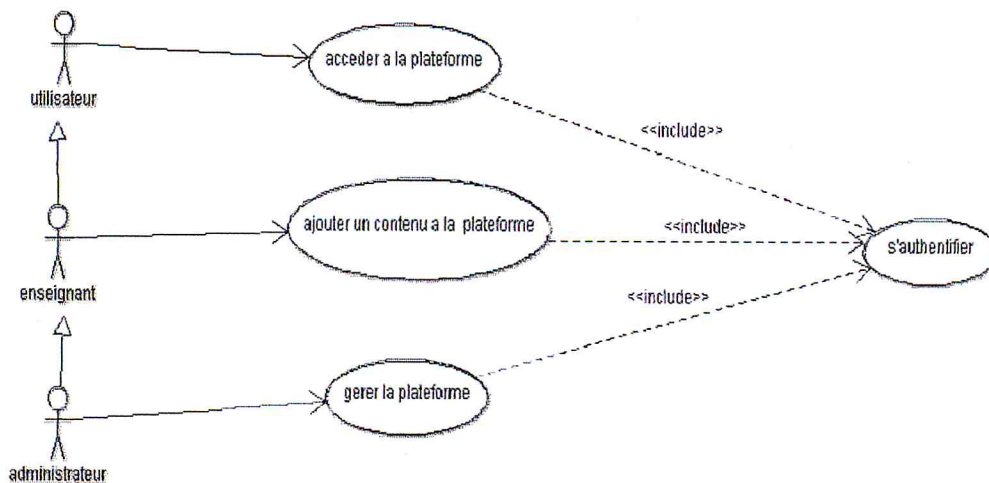


Figure 21 : Diagramme de cas d'utilisation globale

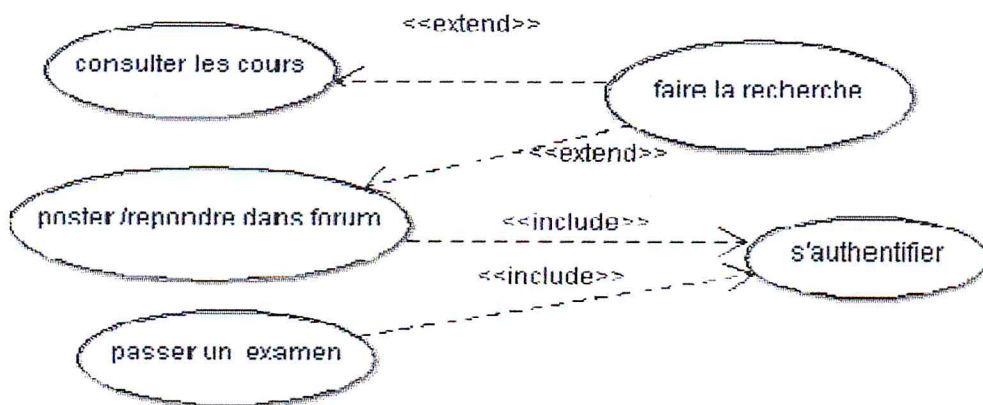


Figure 22 : premier niveau : accéder a la plate-forme

Quand un utilisateur accède a la plate-forme, il peut consulter des cours, poster/répondre dans un forum et même passer un examen en ligne

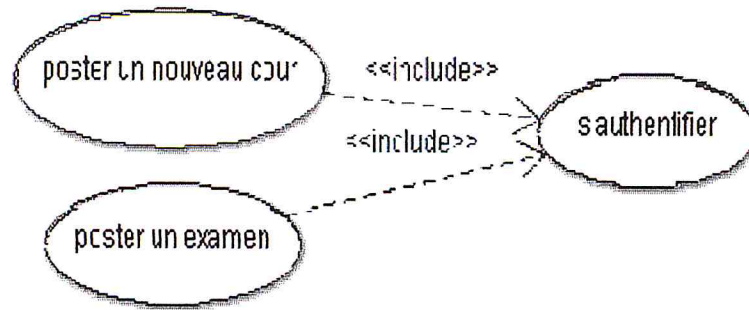


Figure 23 : premier niveau : ajouter un contenu dans la plate-forme

Ces fonctionnalités peuvent être réalisées seulement par l'administrateur et l'enseignant.

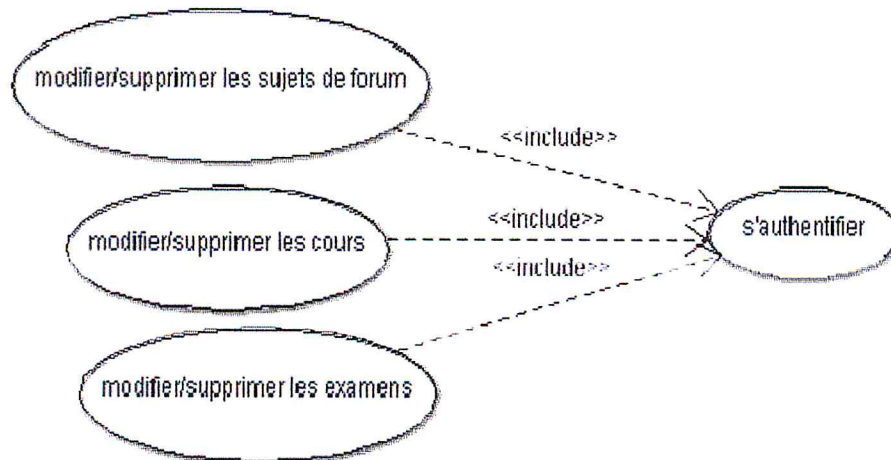


Figure 24 : premier niveau : gérer la plate-forme

Ces tâches sont réalisées par l'administrateur

3.2. Diagrammes de séquence :

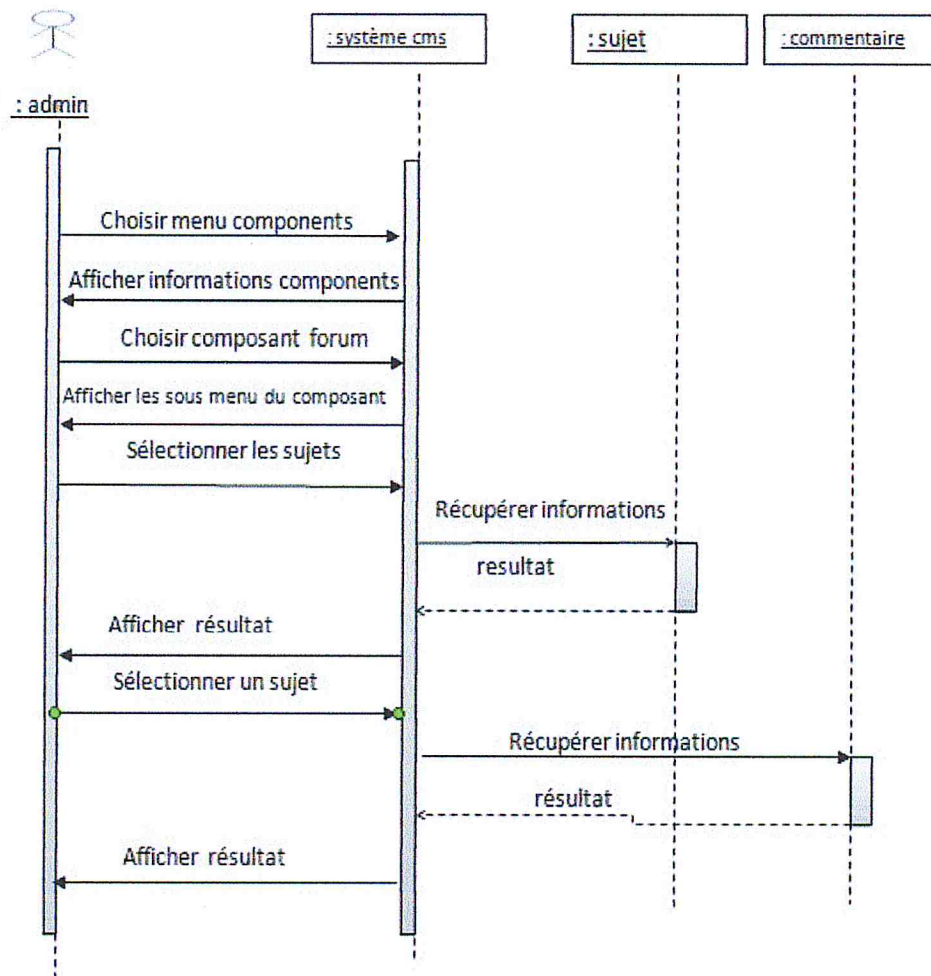


Figure 25 : diagramme de séquence configuration forum

L'administrateur choisit le menu components. Le système affiche les différents composants installés, pour que l'administrateur puisse sélectionner son choix.

L'administrateur sélectionne le composant Forum et là, le système lui affiche un sous menu de ce composant.

L'administrateur choisit le sous menu sujets. Le système vérifie avec la base de données et affiche la liste des sujets.

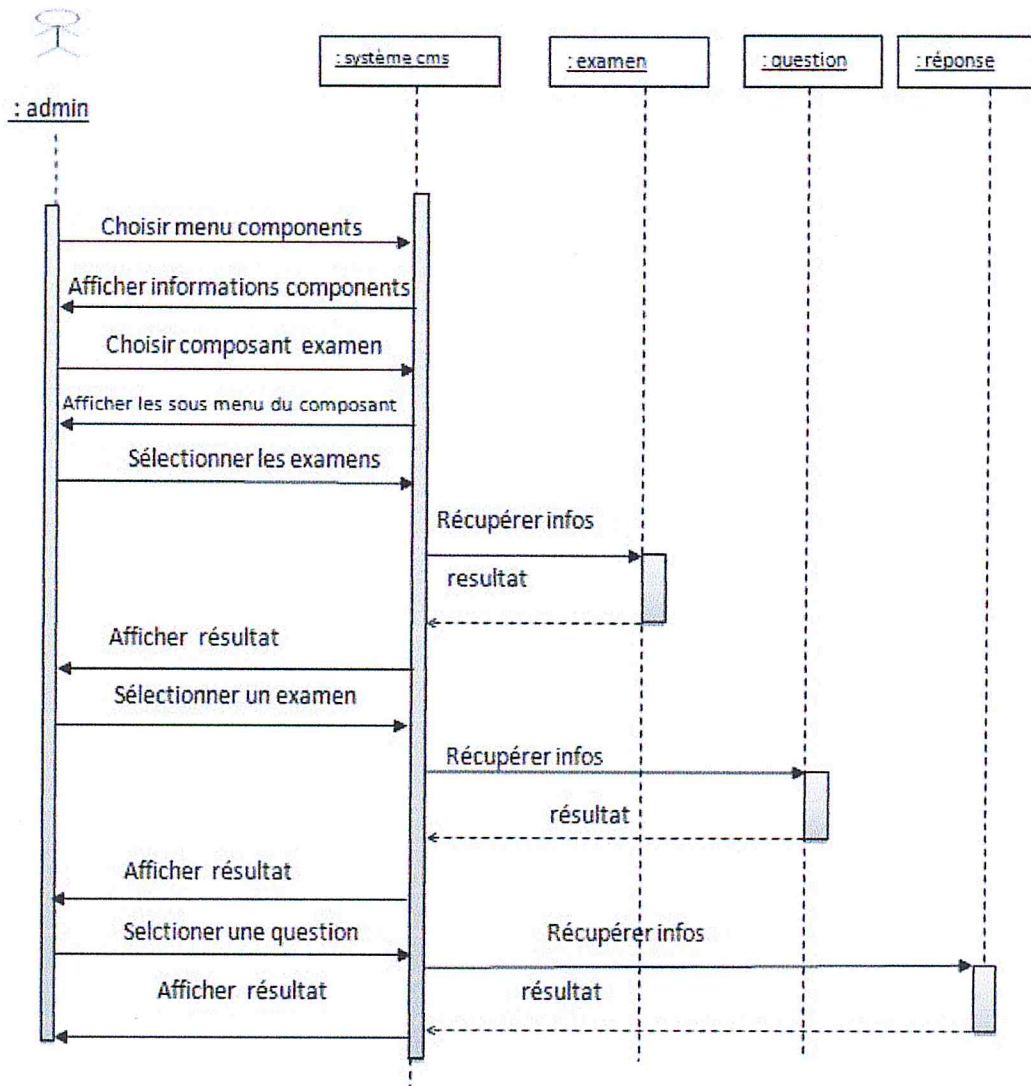


Figure 26 : diagramme de séquence configuration examen

L'administrateur choisit le menu component. Le système affiche les différents composants installés.

L'administrateur sélectionne le composant Examen et le système lui affiche un sous menu de ce composant pour qu'il puisse choisir un examen.

Quand l'administrateur choisit son examen, le système lui renvoi le résultat après avoir vérifier dans la base de données.

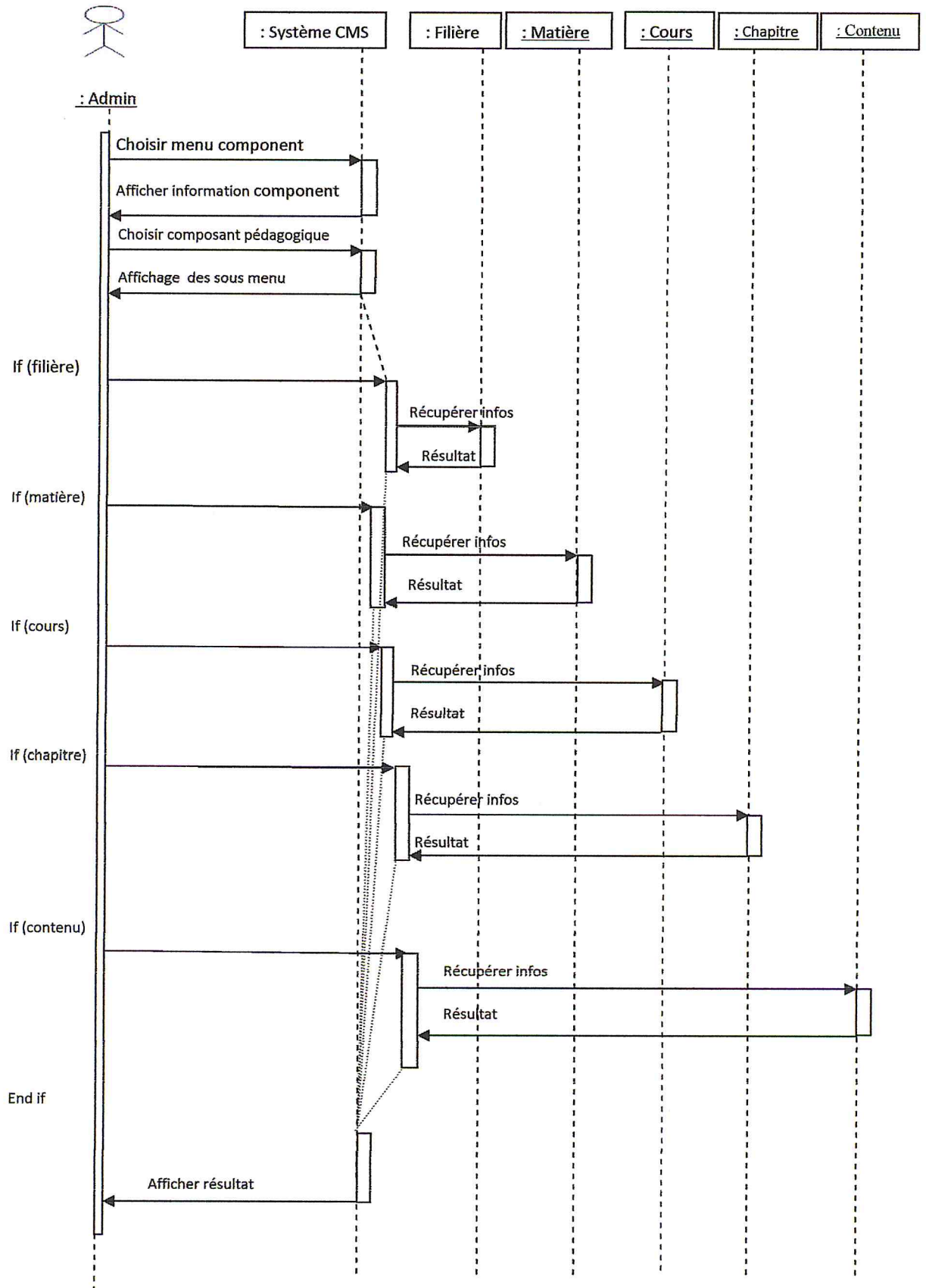


Figure 27 : diagramme de séquence configuration cours

L'administrateur choisit le menu component. Le système affiche les différents composants installés.

L'administrateur sélectionne le composant de source pédagogique, et le système lui affiche un sous menu de ce composant.

L'administrateur choisit le sous menu filière/matière/cours ou chapitre ensuite le système consulte la base de données afin de collecter les informations puis les afficher.

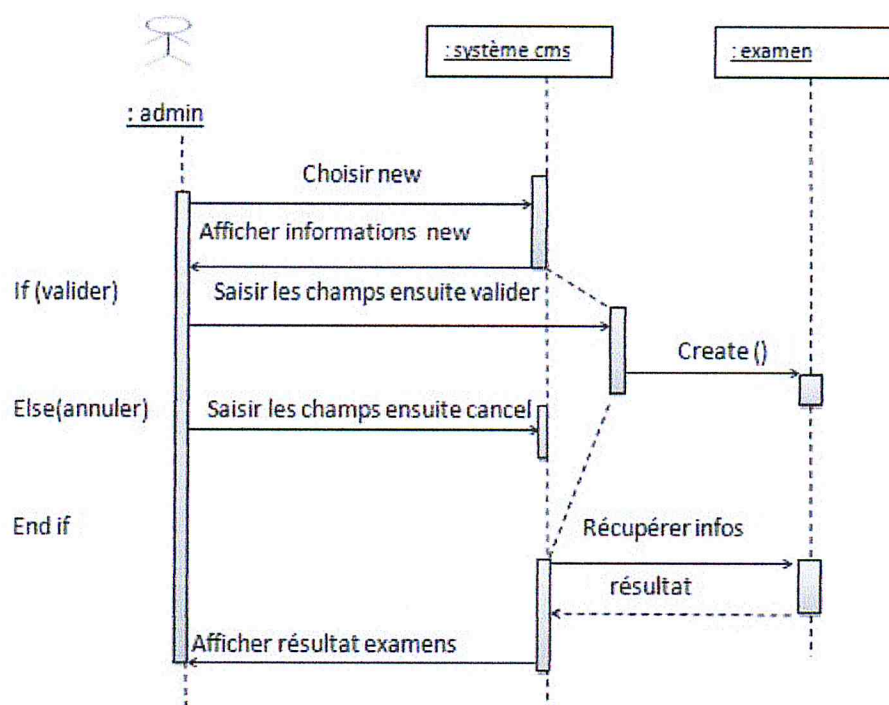


Figure 28 : diagramme de séquence création examen

L'administrateur sélectionne le bouton (New) pour pouvoir créer une nouvelle configuration. Le système affiche une page contenant les informations relatif à une nouvelle configuration, après la saisi de ces informations l'administrateur doit confirmer et sauvegarder cette configuration en sélectionnant Save, il a aussi la possibilité d'annuler la configuration avant de sauvegarder ou quitter l'espace d'une nouvelle configuration avec le bouton cancel.

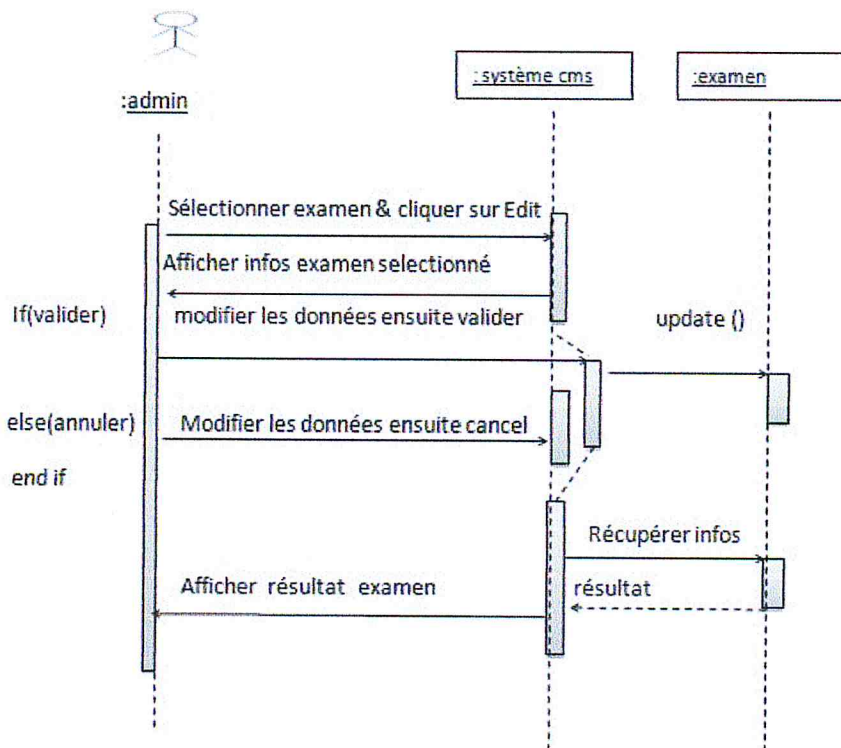


Figure 29 : diagramme de séquence modifier examen

Si l'administrateur veut réaliser une modification relative à un examen, il n'a qu'à sélectionner l'examen ensuite cliquer sur EDIT.

Le système affiche les détails de cet examen, à ce stade l'utilisateur fait ses modifications et clique sur SAVE pour sauvegarder et pour cela le système consulte la base de données pour enregistrer ces changements.

L'administrateur a le droit d'annuler ou même quitter l'espace d'une modification avant de sauvegarder avec l'option CANCEL.

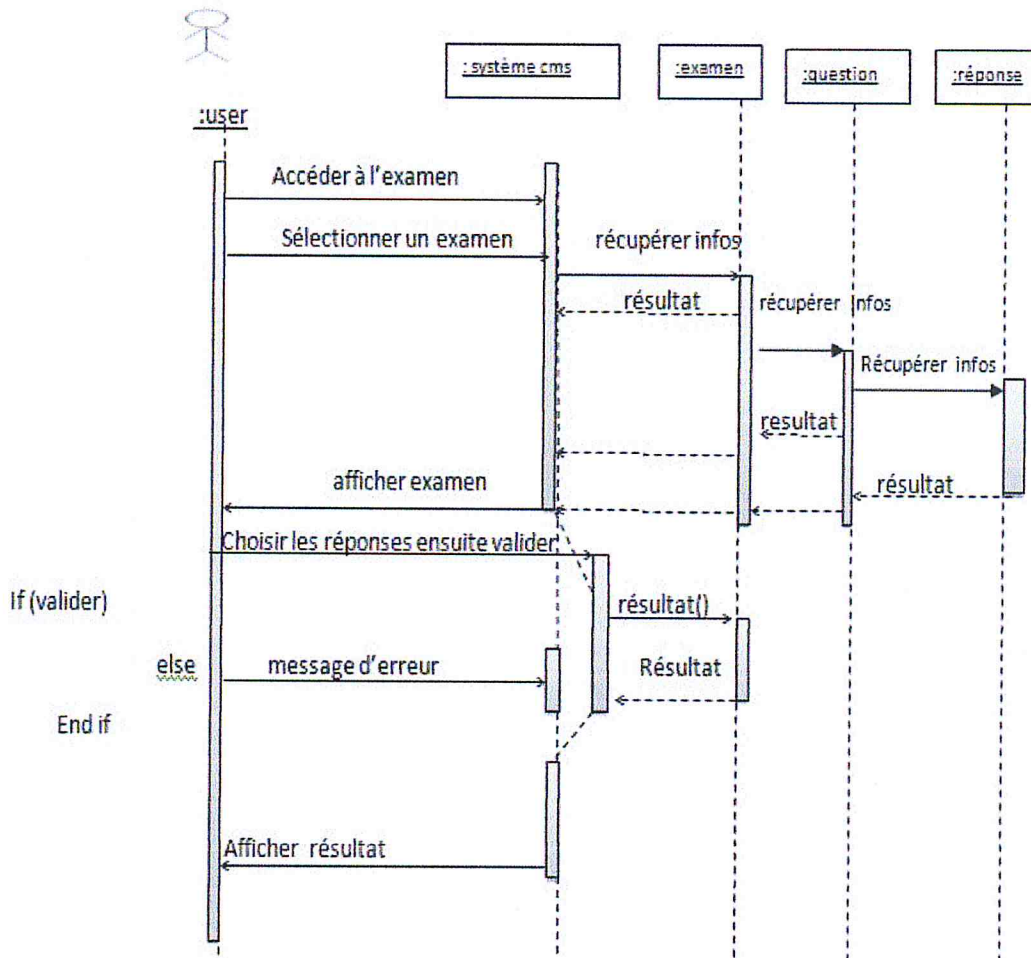


Figure 30 : diagramme de séquence passer examen

L'utilisateur choisit un examen parmi une liste.

Le système demande l'information de la base de données et affiche les questions avec réponses à choix multiples.

L'utilisateur valide une ou plusieurs réponses pour chaque question en cliquant sur le bouton terminer.

Le système lui affiche le résultat de ses réponses en comparant ses choix avec les réponses vraies dans la base de données.

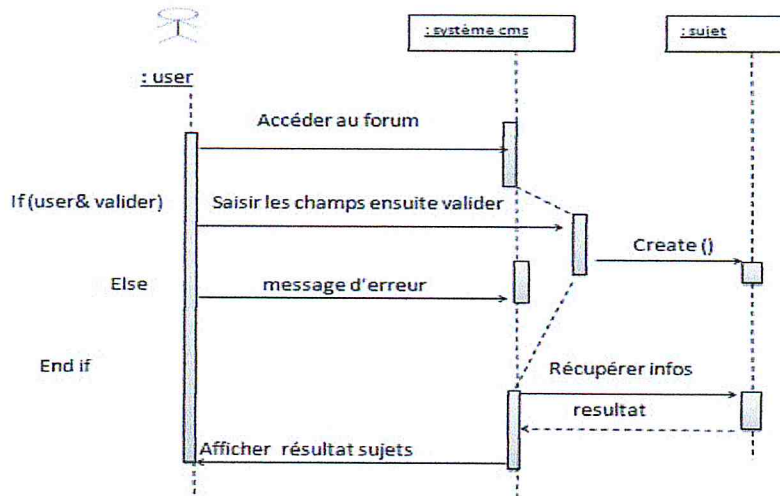


Figure 31 : diagramme de séquence ajouter un sujet

L'utilisateur accède au forum et remplit le formulaire d'un nouveau sujet. Le système l'ajoute à la base de données et l'affiche comme résultat.

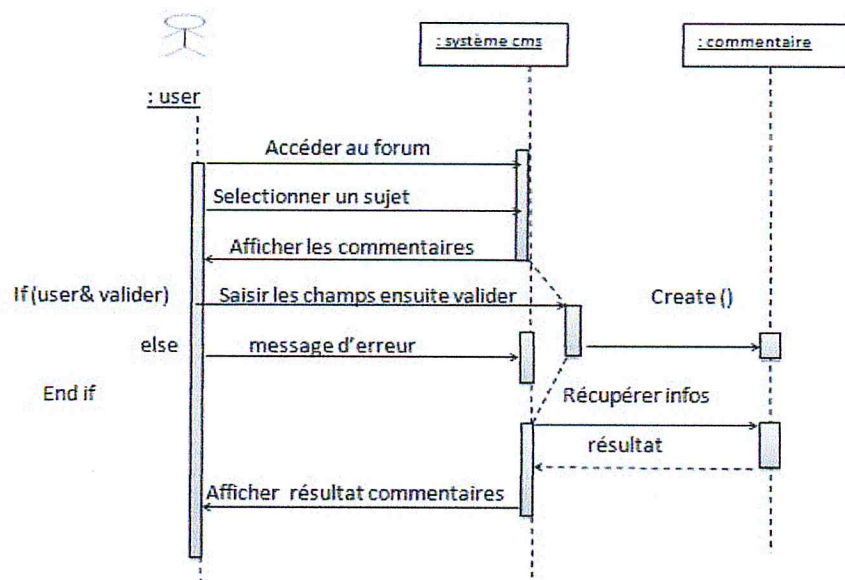


Figure 32 : diagramme de séquence ajouter un commentaire

L'utilisateur accède au forum, sélectionne un sujet et remplit un formulaire pour créer un nouveau commentaire. Le système ajoute ce commentaire à la base de données et l'affiche comme résultat.

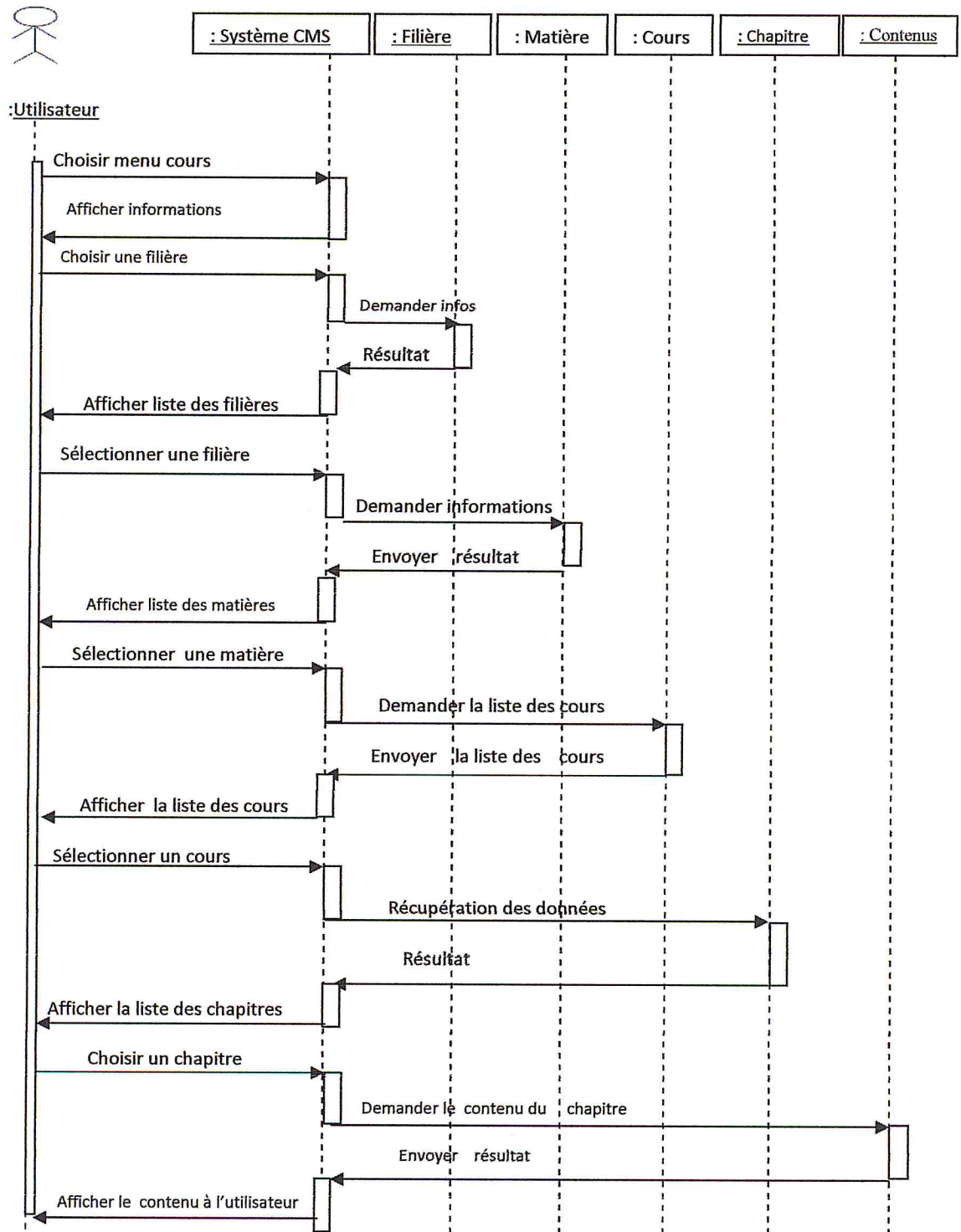


Figure 33 : diagramme de séquence consulter un cours.

L'utilisateur sélectionne une filière parmi une liste de filières.

Le système demande l'information de la base de données et affiche la liste des matières comme résultat.

L'utilisateur choisit une matière.

Le système redemande l'information de la base de données afin d'afficher la liste des cours comme résultat.

L'utilisateur indique son choix de cours parmi la liste des cours.

Le système redemande l'information de la base de données et affiche la liste des chapitres.

L'utilisateur détermine son choix de chapitre et le système affiche le contenu du chapitre comme résultat après avoir demandé l'information de la base de données.

3.3. Diagramme de classes:

Ce diagramme représente les tableaux que nous allons ajouter à la base de données de Joomla.

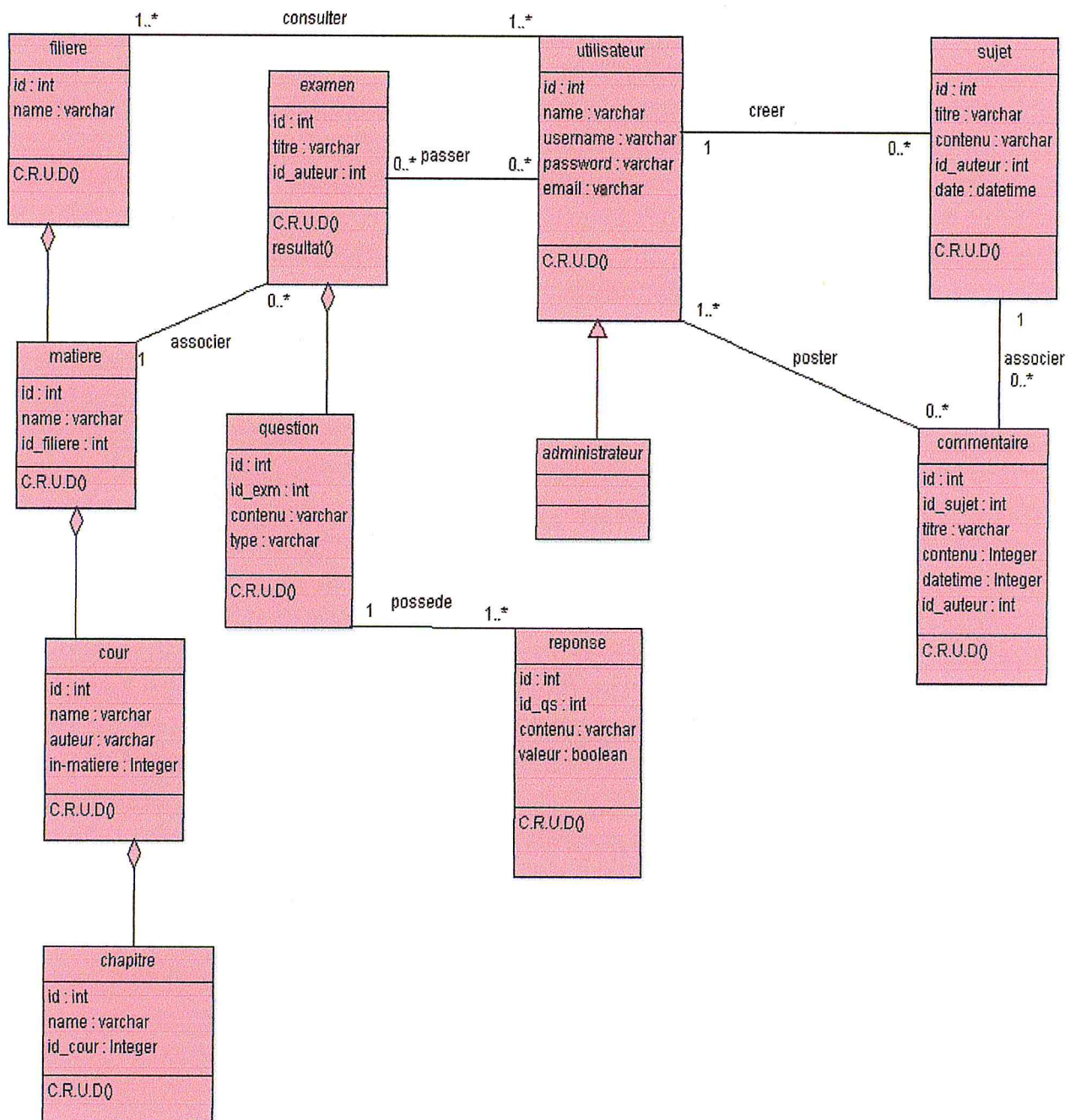


Figure 34 : diagramme de classes (forum/examen/cours)

4. Fonctionnement du composant forum :

Nous allons simplifier par le schéma suivant le déroulement du composant forum, dès que le membre entre dans le site il doit choisir le menu forum pour qu'il puisse voir tous les sujets et leurs commentaires appropriés.

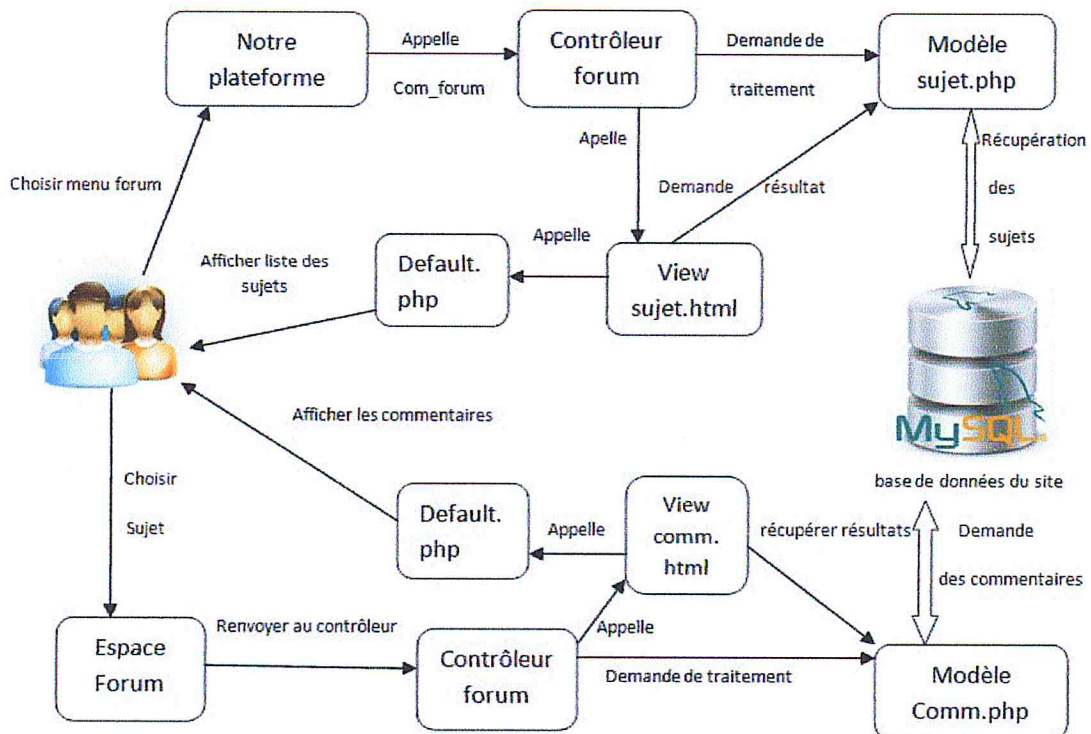


Figure 35 : Le déroulement du composant Forum

Nous allons citer les étapes de déroulement du composant :

- Au début, l'utilisateur doit choisir le menu *cours pédagogiques* qui s'apparaît en haut, cette action va appeler le composant forum installé dans le site.
- Le premier qui va s'exécuter est le contrôleur du composant, et ce dernier va appeler le modèle (sujet) et la vue (sujet).
- Le modèle (sujet.php) est responsable de la sélection des sujets stockés dans la base.
- En suite, la vue (view.html.php) située dans le dossier (view/sujet) va récupérer les sujets à partir du modèle et les retransmet au default.
- A la fin, le default va afficher à l'utilisateur les sujets du forum.
- L'utilisateur choisit un sujet pour le consulter, et maintenant un autre enchaînement va être exécuté en appelant le modèle (commentair.php) et la vue (view.html) du dossier (view/comm) jusqu'à l'affichage des commentaires sur écran.

5. Fonctionnement du composant Cours-pédagogiques et Examen :

Le schéma suivant va nous expliquer comment le composant (cours-pédagogiques) ou (examen) se déroule, sauf qu'ici nous allons utiliser le mot 'spécifique' pour rendre le schéma simple et clair.

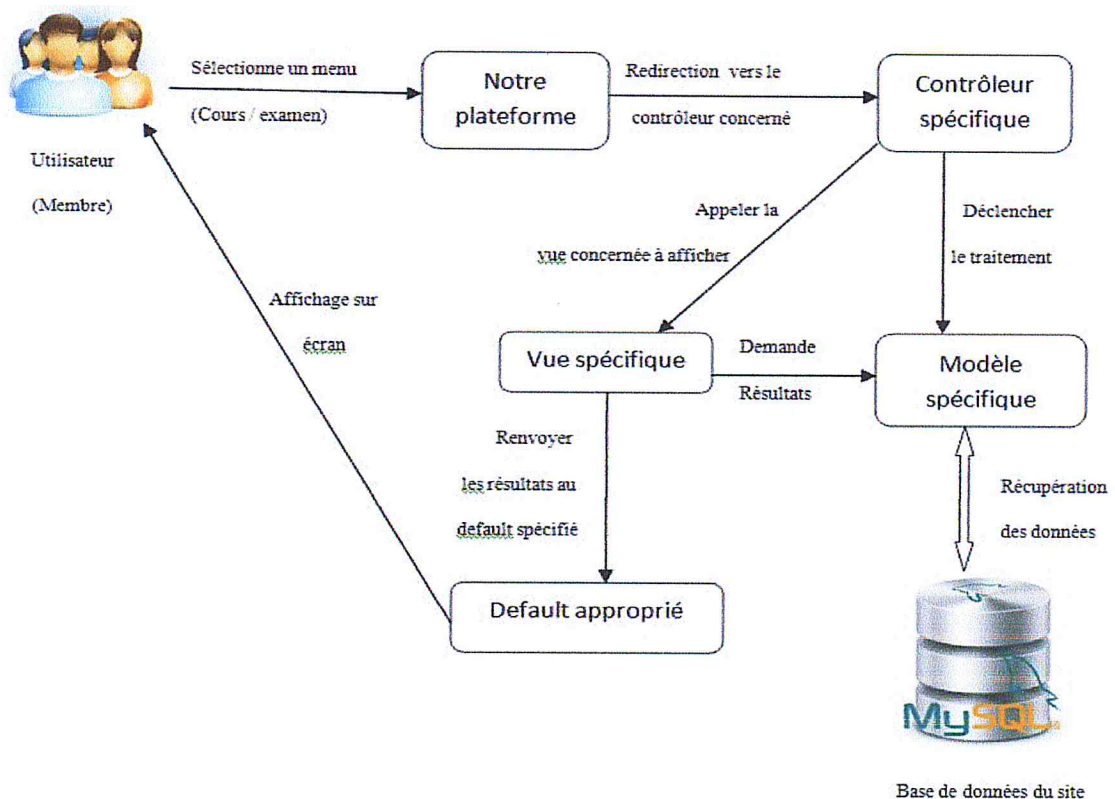


Figure 36: Le déroulement du composant cours-pédagogique et examen

Nous avons présenté le déroulement du composant (cours-pédagogique) et (examen) dans le schéma, nous avons pris en considération l'utilisation du modèle MVC dans Joomla.

Les étapes sont comme suit :

- Comme il existe une hiérarchie pour arriver à un cours, l'utilisateur doit choisir dans la liste des filières la matière à consulter, après il doit choisir le cours et ainsi de suite. Et c'est la même démarche pour passer un examen.
- Le contrôleur fait le lien entre le modèle et la vue spécifiée.
- Ensuite le modèle concerné fait son traitement et ramène les données de la base du site.
- Après, la vue va récupérer les informations retenus dans le modèle, et les renvoyer tous au default spécifique.
- En dernier, le default va afficher les informations demandées sur l'écran de l'utilisateur.

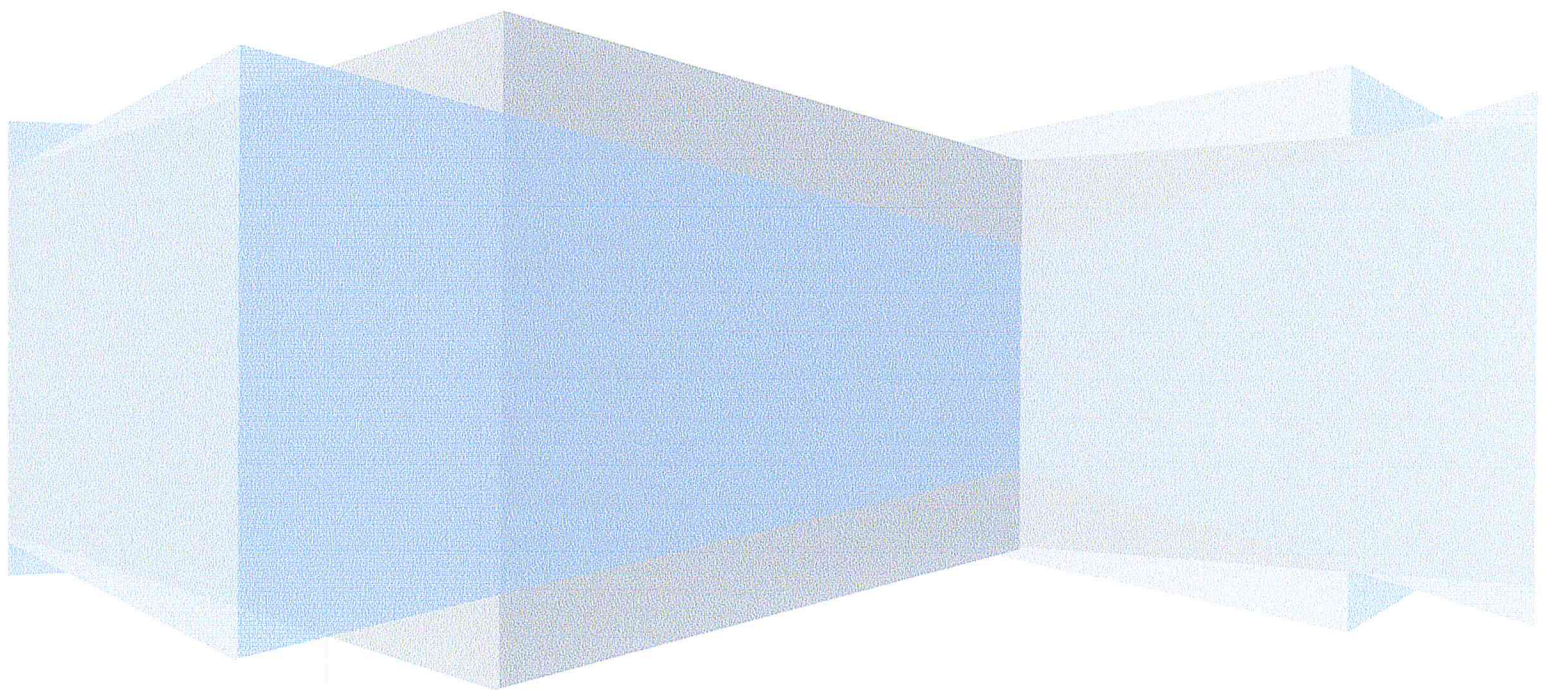
6. Conclusion :

Nous avons illustré les différentes fonctionnalités du système par des diagrammes de cas d'utilisation. Ensuite une étude conceptuelle par des diagrammes de séquences et de classes.

La partie suivante portera essentiellement sur la concrétisation et la mise en œuvre du système étudié.

Chapitre 4

Mise en oeuvre



1. Introduction :

Dans l'étude conceptuelle nous avons présenté notre système avec ses acteurs et cas d'utilisation. Maintenant, nous allons donner le détail d'implémentation des différents modules du système.

En outre, la réalisation concerne aussi le déploiement de l'application dans son environnement réel. Nous allons présenter l'implémentation et la mise en œuvre de cette plateforme par des illustrations.

Nous présenterons en premier le langage et le SGBD utilisés dans la réalisation, puis les interfaces de différents utilisateurs, ainsi que chacune de leurs fonctionnalités.

2. Présentation des outils de développement :

2.1. Le langage PHP :

Le terme PHP est un acronyme récursif de "PHP: Hypertext Preprocessor". Ce langage est principalement utilisé pour produire un site web dynamique. Il est courant que ce langage soit associé à une base de données, tel que MySQL. ^[H]

PHP est un langage impératif disposant depuis la version 5 de fonctionnalités de modèle objet complètes. En raison de la richesse de sa bibliothèque, on désigne parfois PHP comme une plate-forme plus qu'un simple langage. ^[I]

Joomla! Est écrit en PHP. Des milliers de fichiers qui contiennent du PHP ont été écrits au cours de ces dernières années. Joomla! contient plus de 500 000 lignes de code PHP. Cela nécessiterait 244 personnes travaillant à l'année pour le développer! (coût estimé de Joomla!99). Lorsque Joomla! a été fondé, les développeurs utilisaient principalement PHP4.x qui était alors très courant. Cela a été un challenge de réécrire tout le code existant en utilisant aussi bien que possible les caractéristiques du PHP 5.x. ^{[8]p140}

2.2. Programmation orientée objet :

La programmation orientée objet, en anglais Object-oriented programming (OOP) est un paradigme de programmation utilisant des « objets ». Structure de données composées de champs de données et de méthodes ayant chacun leurs interactions.

Pour concevoir des applications et de programmes informatiques. Les techniques de programmation peuvent inclure des fonctionnalités telles que l'abstraction de données, encapsulation, messagerie, modularité, polymorphisme et héritage.

Aujourd'hui, De nombreux langages de programmation modernes supportent OOP.

Le paradigme OOP veut décrire des structures et des rapports entre les objets comme dans le monde réel ! ^{[8]p151}

2.3. Description du serveur de base de données MySQL :

MySQL est un système de gestion de base de données (SGBD). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft SQL Server. ^[1]

3. Les interfaces :

3.1. Partie front-end (site) :

- **La page d'accueil :**

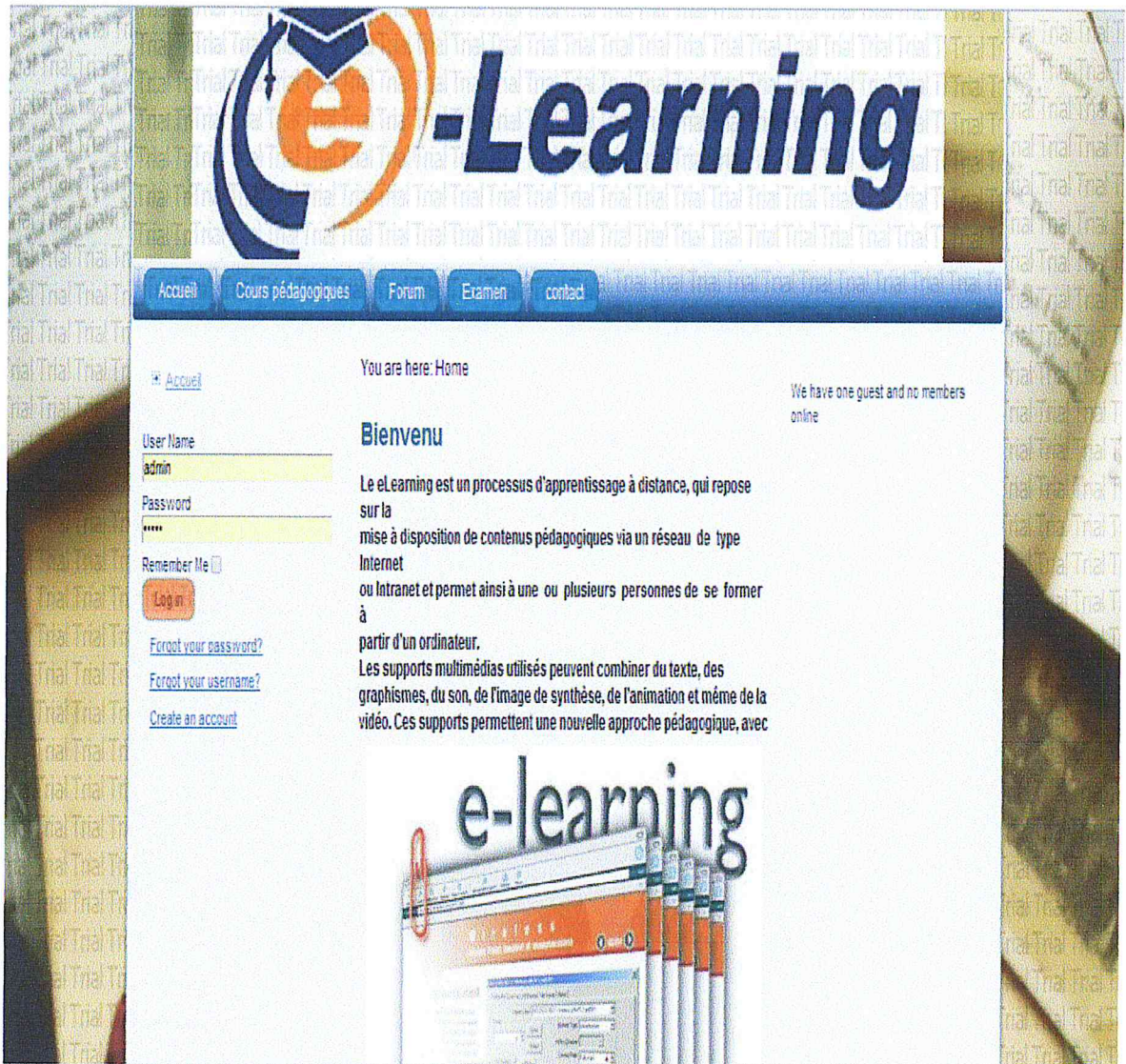


Figure 37 : La page d'accueil du site.

Sur cette figure nous avons un aperçus sur la page d'accueil de notre plateforme. Un visiteur peut s'inscrire pour devenir membre ou s'authentifier s'il est déjà inscrit.

- **Le Menu Cours-pédagogiques :**

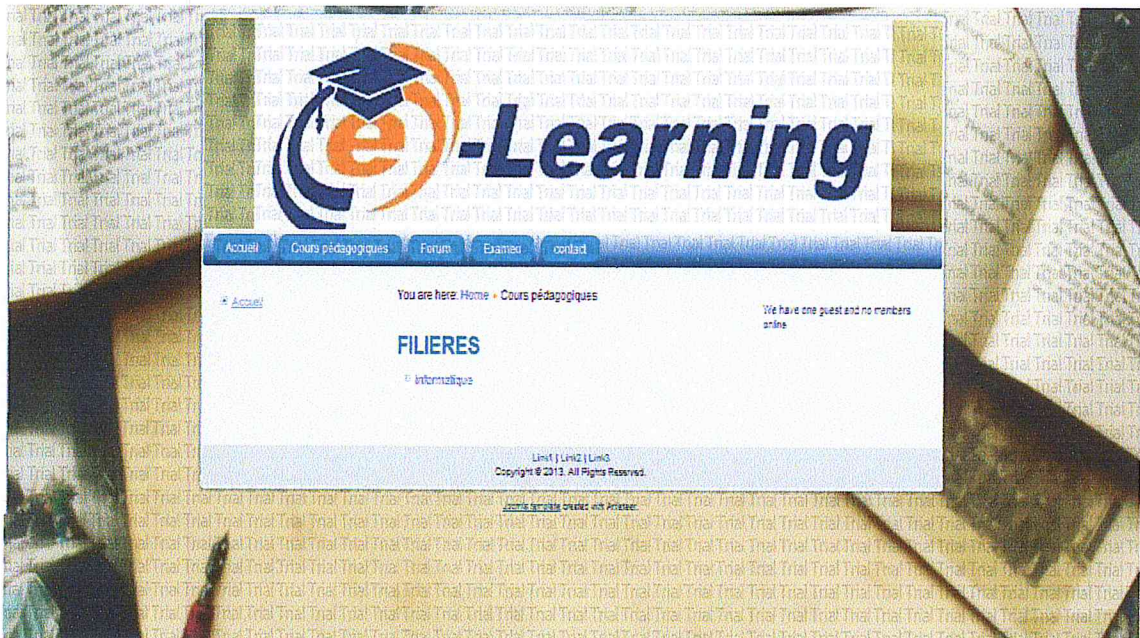


Figure 38: Menu cours pédagogiques.

Après la sélection du menu cours pédagogiques, une liste de filières apparaît.

En choisissant une filière, la liste des matières va s'apparaître, et après un choix de matières on peut consulter la listes des cours et après la liste des chapitres.

- **Le contenu d'un chapitre :**



Figure 39 : Le contenu d'un chapitre.

- **Espace forum :**

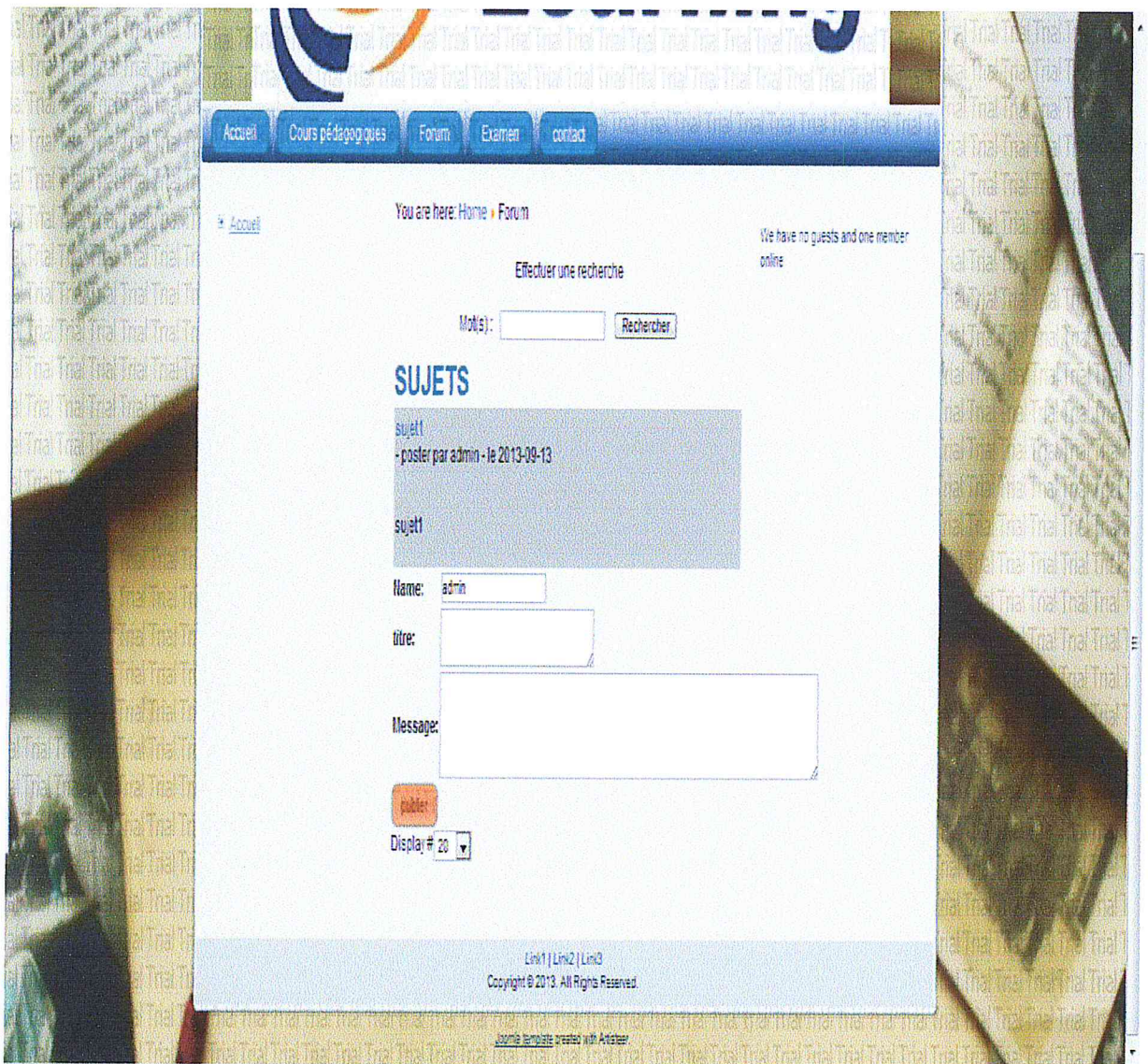


Figure 40 : Espace forum.

Ici nous sommes dans l'espace forum, on peut voir la liste des sujets ainsi que l'auteur et la date de l'envoi de chaque sujet.

En cliquant sur le titre du sujet, une liste de messages concernant ce sujet apparaît.

Un visiteur a le droit de consulter le forum mais sans la possibilité d'ajouter un sujet ou répondre à un sujet, pour le faire il doit s'authentifier.

3.2. Partie back-end (Administration) :

- **Menu components :**

Après une installation avec succès de nos trois composants (forum ,examen et cours pédagogiques) nous devons les apercevoir sur le menu components et ils seront prêt à être utilisé.

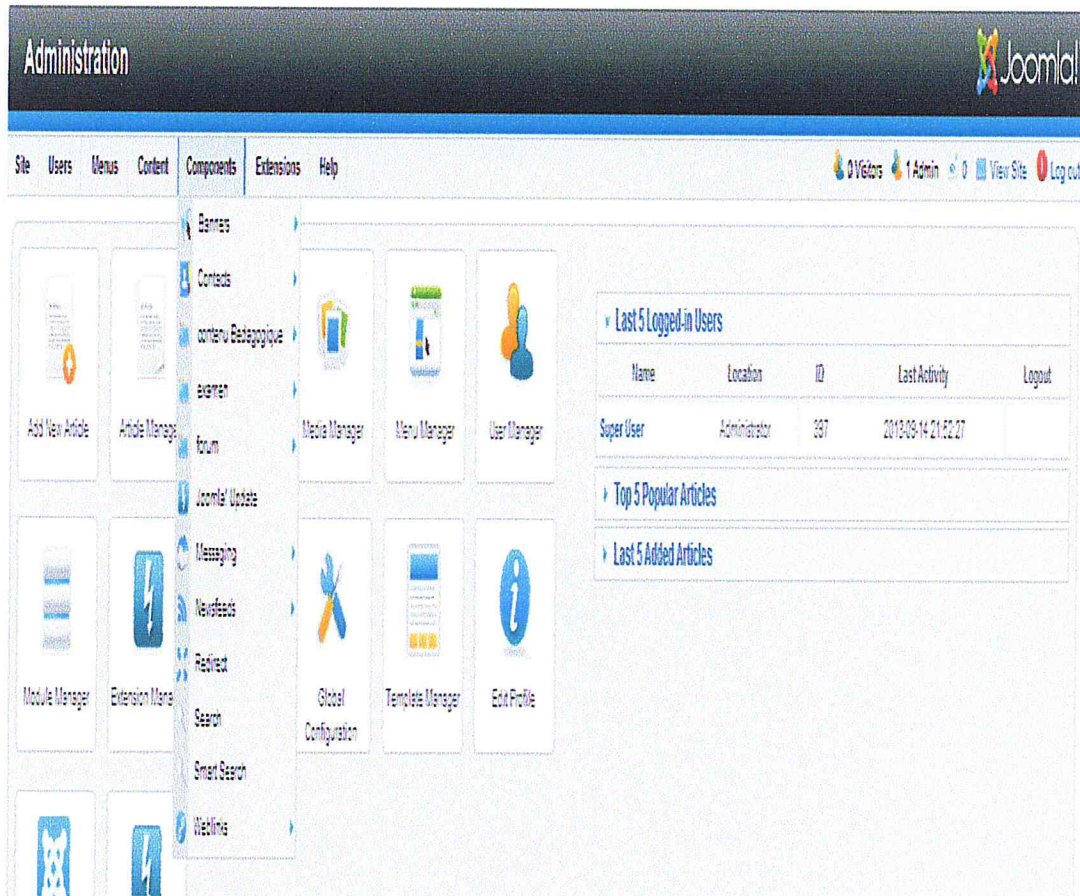


Figure 41 : installation réussite des composants.

Dans le sous menu du composant cours on peut distinguer les différentes fonctionnalités :

- Filières : récupère tous les informations concernant la table filières.
- Matières : récupère tous les informations concernant la table matières.
- Cours : récupère tous les informations concernant la table cours.
- Chapitres : récupère tous les informations concernant la table chapitres.

- **Sous-menu du composant cours :**

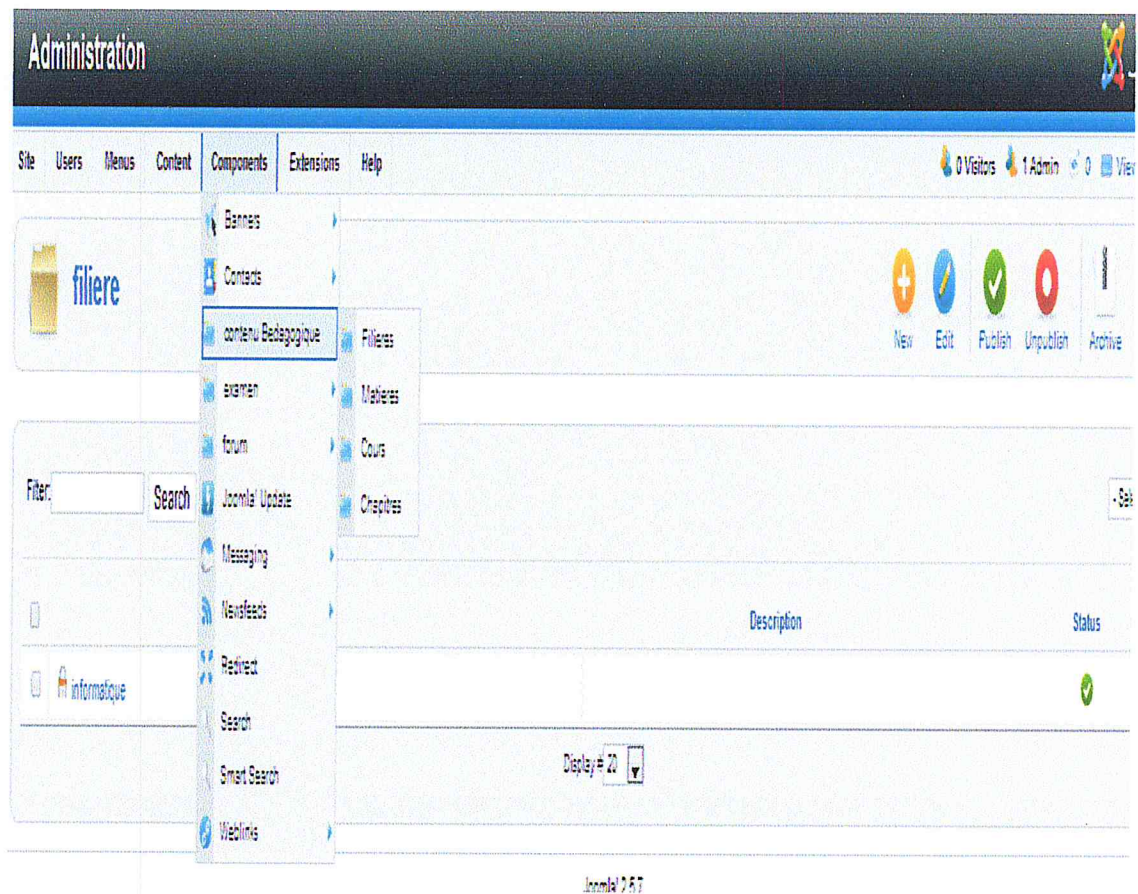


Figure 42: Sous menu cours pédagogique.

- **Modifier un chapitre :**

The screenshot shows a web interface for editing a chapter. At the top, there's a header with a book icon and the text 'Edit chapitre'. On the right side of the header, there are four icons: a checkmark (Save), a floppy disk (Save & Close), a floppy disk with a plus sign (Save & New), and a red 'X' (Cancel).

The main content area is titled 'chapitre' and contains several form fields:

- Name:** A text input field containing 'introduction'.
- Alias:** A text input field containing 'introduction'.
- Teaser:** A large text area containing 'Espace de données'.
- Status:** A dropdown menu with 'Publié' selected.
- Description:** A rich text editor with a toolbar (bold, italic, underline, link, unlink, list, list, link, unlink, undo, redo, help) and a text area containing the following text:

Une base de données informatique est un **ensemble de données** qui ont été stockées sur un support informatique, et **organisées et structurées** de manière à pouvoir facilement consulter et modifier leur contenu.

Preions l'exemple d'un site web avec un système de news et de membres. On va utiliser une base de données MySQL pour stocker toutes les données du site : les news (avec la date de publication, le titre, le contenu, éventuellement l'auteur,...) et les membres (leurs noms, leurs emails,...).

Tout ceci va constituer notre base de données pour le site. Mais il ne suffit pas que la base de données existe. Il faut aussi pouvoir la gérer, interagir avec cette base. Il faut pouvoir envoyer des message à MySQL (messages qu'on appellera "requêtes"), afin de pouvoir ajouter des news, modifier des membres, supprimer, et tout simplement afficher

At the bottom of the description field, there are buttons for 'Article', 'Image', 'Page Break', 'Read More', and 'Toggle editor'.

On the right side, there is a sidebar with two sections:

- Details:** Contains a dropdown for 'cour' with 'Les BD et les SGBD' selected, and a dropdown for 'Access' with 'Public' selected.
- Optional:** A section that is currently empty.

Figure 43 : Modifier un chapitre.

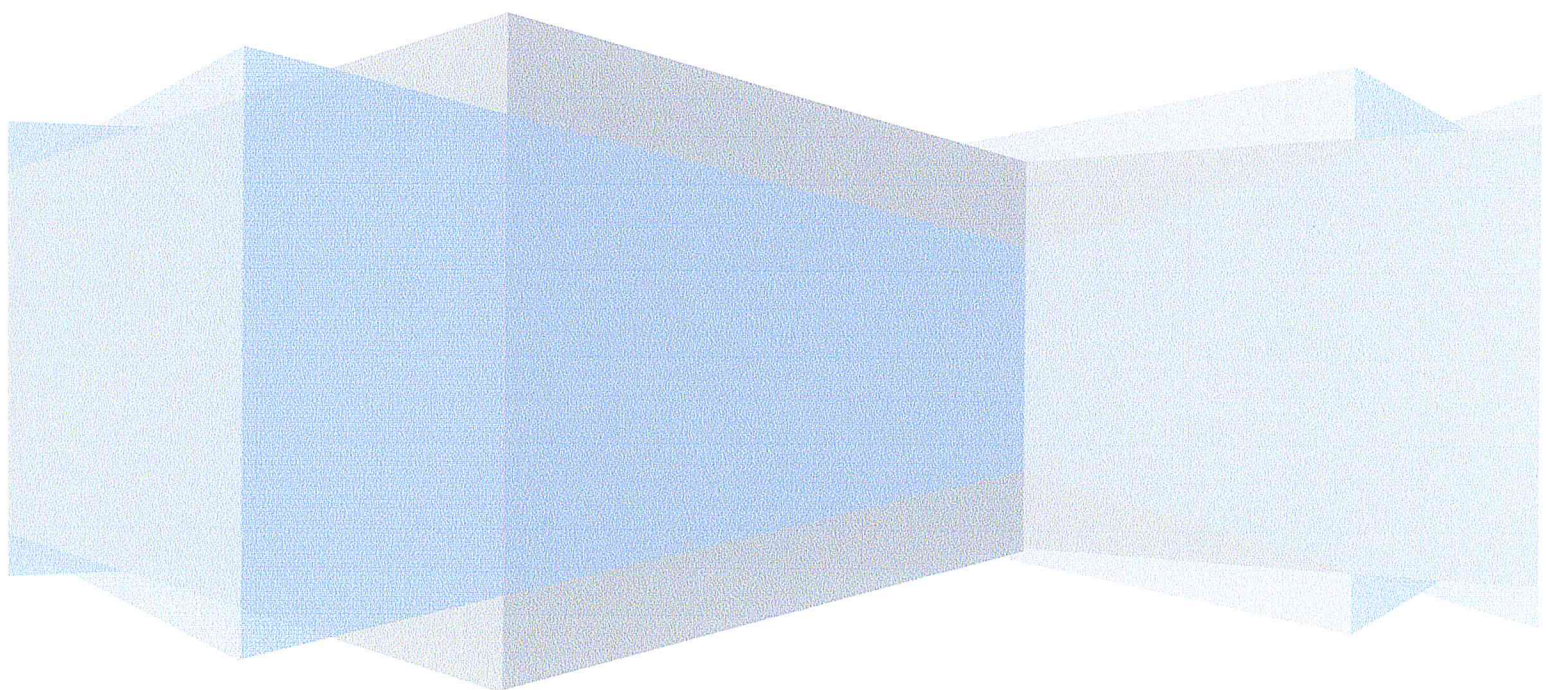
4. Conclusion:

Joomla offre une capacité majeure à l'administrateur de gérer son site , les utilisateurs et les cours, ainsi que les outils de communication. Mais le processus de développement d'un composant sous Joomla est un processus difficile à réaliser, il nécessite beaucoup de connaissances dans le domaine.

Des tests et des essais ont été effectués sur notre plate-forme afin de vérifier sa fiabilité et sa stabilité.

Toutefois, nous devons reconnaître que notre plate-forme ne montrera ses qualités et son efficacité, que dans un environnement réel, c'est-à-dire déployée dans le réseau ou sur le Web, avec de vrais acteurs.

Conclusion générale



Dans ce projet nous avons implémenté une solution de E-Learning en y implémentant un ensemble d'outils parmi plusieurs. Pour cela, nous avons réalisé un forum de discussions et des examens en ligne (questions à choix multiples) ainsi que des cours. Ces outils peuvent être greffés dans n'importe quel CMS basé sur Joomla. Ainsi les outils développés suivent la politique de Joomla et donc bénéficient des avantages de ce dernier,

L'optique de notre travail a été, en premier lieu, de parler de e-Learning en générale, et puis de présenter le CMS Joomla, son principe de fonctionnement et son importance dans le domaine web, qui satisfait les besoins spécifiques de chaque utilisateur.

Aussi, de présenter le concept du modèle MVC et son principe.

Par la suite, nous avons fait une étude conceptuelle de notre système où nous avons mis en évidence les besoins des différents acteurs et le déroulement des différents processus ainsi que l'aspect technique du système.

A la fin, nous sommes parvenus à réaliser des extensions qui satisfont nos besoins.

Cela nous a permis d'activer nos connaissances acquises durant les cinq années universitaires.

- 1- Walid Kassem, Ahmad Mounajed, Nadia Saadoun, **Etat de l'Art du E-Learning**,Projet soutenu le 16/02/2004, université paris 2,p11,p11.
- 2- ILMAR Slaidins ,**Guide de bonnes pratiques à l'usage des acteurs du e-learning**, November 2004, p4.
- 3- LIVRE BLANC, **Comment réussir la mise en place d'un projet eLearning ?**,ED La Défence, 25/2/2005, p5,p9.
- 4- B. Kouninef, M. Djelti, S.M. Rerbal, **Conception et realisation d'une plate forme e-learning avec migration au m-learning**, Institut des télécommunications d'Oran,p1,p2,p2,p3.
- 5- Ivan GAUTREAU, **Joomla! pour les Nuls**, 23 avril 2007,p5.
- 6- HAGEN Graf et autres,**Joomla! 2.5 le Guide pour debutant**, p15.
- 7- Stéphan CAMMARATA,**Document d'accompagnement de la formation Joomla !**, Académie de Strasbourg, Version 2012,p4.
- 8- HAGEN Graf et autres, **Developpement Joomla! le Guide pour debutant**, p20, p140, p151.
- 9- NEMRI Samir, **Conception et réalisation d'un forum WEB pour l'Ecole Nationale des Impots**, promotion 2010/2011,p14.
- 10- S. Graine, **UML2 Pour une modélisation orientée objet**, Ed l'Abeille, 2009, p15, p90.



SITES WEB:

- A-** <http://doc.automne-cms.org/web/Documentation/464-c-est-quoi-un-cms-.php>

- B-** http://www.joomla.fr/news-proposees/nouvelles-fonctionnalites-de-joomla-25?utm_campaign=Feed%3A+joomlafr+%28Joomla.fr%29&utm_medium=feed&utm_source=feedburner

- C-** <http://www.systrio.fr/blog-systrio/item/27-joomla-16-acl-comment-personnaliser-la-gestion-des-droits-utilisateurs->

- D-** [http://docs.joomla.org/Extension_types_\(general_definitions\)](http://docs.joomla.org/Extension_types_(general_definitions))

- E-** <http://julien-pauli.developpez.com/tutoriels/php/mvc-controleur/>

- F-** <http://dev.joomla.fr/tutoriels/item/56-creation-dun-composant-mvc-hello-world>

- G-** http://fr.wikipedia.org/wiki/Diagramme_de_s%C3%A9quence

- H-** <http://glossaire.infowebmaster.fr/php/>

- I-** <http://www.webmestremedia.com/formulaire-en-ligne-repentigny-montreal.php>