UNIVERSITÉ DU BURUNDI
GRENIER DU SAVOIR DU BURUNDI
Dépôt institutionnel officiel
https://repository.ub.edu.bi

Conception et réalisation d'une plate-forme d'enseignement à distance : cas de l'Université des Grands-Lacs

MUTEKANO, Hervé

2021
UB, Faculté des Sciences de l'Ingénieur
https://repository.ub.edu.bi/handle/123456789/1063

Document téléchargé depuis le **Grenier du Savoir du Burundi** — https://repository.ub.edu.bi

République du Burundi
Ministère de l'Education Nationale et de la Recherche Scientifique

Université du Burundi
Faculté des Sciences de l'Ingénieur



Département de TIC
Mastère en Génie Informatique
Année Académique :

2019-2020

MEMOIRE

Présenté à la
Faculté des Sciences de l'Ingénieur (FSI)

En vue de l'obtention du grade de

MASTER

en

Génie Informatique

MUTEKANO Hervé

Conception et réalisation d'une plate-forme d'enseignement à distance : cas de l'Université des Grands-Lacs

Soutenu le 08/09/2021, devant le jury composé de :

Dr	NDAYISABA Longin	Président
Pr	NDIKUMAGENGE Jérémie	Vice-Président
Dr	SAHINGUVU William	Directeur
Dr	NKUNZIMANA Hilaire	Secrétaire

Identification des membres du jury

Dr	NDAYISABA Longin	Président
Pr	NDIKUMAGENGE Jérémie	Vice-Président
Dr	SAHINGUVU William	Directeur
Dr	NKUNZIMANA Hilaire	Secrétaire

Dédicaces

A Dieu Tout Puissant.

A mes Chers Parents.

A mes Chers frères et sœurs.

A mes collègues étudiants.

MUTEKANO Hervé

Remerciements

En premier lieu. nous aimerions remercier Dieu pour nous avoir donné le courage de réaliser ce travail de recherche et de le mener jusqu'au bout.

Je remercie vivement Docteur William SAHINGUVU Directeur de ce travail pour les conseils et les directives pleines d'ardeur et de bonté qu'ils m'ont réservés. Sa disponibilité. sa rigueur scientifique et ses encouragements ont été pour moi d'une importance et utilité sans pareils.

Mes remerciements s'adressent également aux membres du Jury qui ont accepté de lire et évaluer ce travail. Je désire remercier sincèrement mes chers parents, frères et sœurs, pour le soutien constant tant matériel et moral qu'ils m'ont témoignée jusqu'à l'aboutissement de ce travail.

Mes remerciements s'adressent également aux enseignants et au personnel de l'Université du Burundi pour l'encadrement et l'orientation qu'ils m'ont réservé tout au long de cette formation au sein de cette institution. Je tiens à remercier également les personnels d'UGL pour l'accueil chaleureux lors de mes descentes pour la collecte des données.

Mes sentiments de gratitude s'adressent à mes camarades de classe pour le climat d'entraide mutuelle. de soutien et de partage d'expérience tout au long de ma formation à l'Université du Burundi. Mes remerciements s'adressent également à toute personne qui. de près ou de loin a contribué activement dans la réalisation de ce travail.

Résumé

Les Technologies de l'Information et de la Communication (TIC) dans le domaine de la formation a conduit à la création de cette nouvelle réalité appelée e-learning (apprentissage en ligne). Ce dernier est décrit comme l'union du multimédia (son, vidéos, image, texte) et les médias de communication (diffusion en ligne, interactivité). L'enseignement à distance, devenu désormais un atout important pour la formation, passe en particulier par la mise en place d'une plate-forme d'apprentissage à distance. Dans notre mémoire, nous allons concevoir une plate-forme d'enseignement et d'apprentissage à distance de l'Université des Grands-Lacs. L'apprenant est suivi tout au long de son apprentissage. Dans ce système les formateurs et les apprenants communiquent individuellement ou en groupe, l'administrateur installe et assure la maintenance de ce système, gère les accès et les droits des utilisateurs [1].

Mots clés: TIC, e-Learning, EAO, LMS, UML, File d'attente

Abstract

Information and Communication Technologies (ICT) in the field of training has led to the creation of this new reality called e-learning (online learning). The latter is described as the union of multimedia (sound, videos, images, text) and communication media (online broadcasting, interactivity). Distance education, which has now become an important asset for training, requires in particular the establishment of a distance learning platform. In our brief, we are going to design a distance education platform for the University of the Great Lakes. The learner is followed throughout his learning. In this system, trainers and learners communicate individually or in groups, the administrator installs and maintains this system, manages user access and rights.

Keywords: ICT, e-Learning, EAO, LMS, UML, queue theory

Table des matières

Identification des membres du jury i
Dédicaces ii
Remerciements iii
Résumé iv
Abstract iv
Table des matières v
Liste des figures viii
Liste des tableaux ix
Liste des symboles x
AVANT-PROPOS xi
I INTRODUCTION GENERALE 1
I.1. Justification et contextualisation 1
I.2 Objectifs du projet 3
I.2.1 Objectif global 3
I.2.2 Objectifs spécifiques 3
I.3 Problématique 3
I.4 Analyse et critique de l'existant 3
I.5 Solutions proposées 4
I.6. Intérêt du sujet 4
I.7 Délimitation du sujet 5
I.8 Apports scientifiques et technologiques 5
I.9 Domaine d'application 5
I.10 Méthodologie de recherche utilisée 6
I.11 Outils mathématiques utilisés 6
I.12. Résultats attendus 6
CHAPITRE II PRESENTATION DU MILIEU D'ETUDE 8
II.1 Historique 8
II.2 Cahier des charge 9
II.2 .1 De la gouvernance universitaire 9
II.2.2 De la gouvernance administrative 9
II.2.3 De la gouvernance académique 10
II.2.4 Du personnel enseignant 10
II.2.5 De l'enseignement 10
II.2.6 Des infrastructures 11

II.3 Organigramme 11
II.3.1 Introduction 11
II.4 Manuel de procédure 12
II.4.1 Conseil de l'université 12
CHAPITRE III OPTIMISATION D'UNE PLATEFORME D'ENSEIGNEMENT A DISTANCE PAR LA THEORIE DU FILE D'ATTENTE 14
III. 1 Systèmes à temps discret 15
III.2 Notion des files d'attente 15
III.3 Caractéristiques d'un système des files d'attente 16
III.4 Notation de Kendall 16
III.5 Distribution exponentielle 18
III.6 Principaux modèles des files d'attente 21
III.7 Diagramme de transition entre les états 23
III.8 Vérification des performances du serveur de notre application 28
d'apprentissage en ligne 28
CHAPITRE IV EXPRESSION DES BESOINS ET MODELISATION DU SYSTEME D'INFORMATION D'ENSEIGNEMENT A DISTANCE 31
IV.1 Concepts de base et méthodologies 31
IV.1.1 Concepts 31
IV.1.2 Méthodologies 32
IV.2 Présentation de la méthode UML 32
IV.3 Cycles de développement des applications informatiques 35
IV.4 Couche Modèle 37
IV.5 Diagramme des cas d'utilisation 39
IV.6 Description des principaux cas d'utilisation 40
IV.7 Diagramme de classe 44
IV.8 Diagrammes de séquences 46
IV.9 Diagrammes d'activités 47
IV.10 Diagramme de déploiement 50
CHAPITRE V : REALISATION ET DEVELOPPEMENT D'UNE PLATEFORME D'ENSEIGNEMENT A DISTANCE 52
V.1 Introduction 52
V.2 Etude technique 52
V.3 Descriptif d'une plateforme d'enseignement à distance 56
V.4 Architecture 57
V.5 Fonctionnement du nouveau système 58

V.5.1 Interface d'accueil 58
V.5.2 Interface d'authentification 60
V.5.3 Interface pour l'administrateur 60
V.5.4 Interface pour l'instructeur 61
V.5.5 Interface du salon de formation de l'instructeur 62
V.5.6 Interface du salon de formation coté étudiant 62
V.5.7 Apports scientifiques et technologiques 63
CONCLUSION GENERALE ET RECOMMANDATIONS 65
Conclusion générale 65
Recommandations 66
REFERENCES BIBLIOGRAPHIQUES 67

Liste des figures

Figure 1:Histogramme d'enquête 3
Figure 2:Organigramme de l'UGL 12
Figure 3: Structure de base d'une file d'attente 16
Figure 4: Modèle en cascade 36
Figure 5: Modèle en v 36
Figure 6 : Architecture MVC 39
Figure 7 : Diagramme de cas d`utilisation du système 40
Figure 8: Diagramme de classes de notre système 45
Figure 9: Diagramme de séquence pour l`authentification 46
Figure 10 : Diagramme de séquence pour poster un cours 47
Figure 11 : Diagramme de séquence pour la création des formations et sessions 47
Figure 123 : Diagramme de déploiement de notre système 50
Figure 13: Interface d`authentification 60
Figure 14: Interface de l`administrateur 61
Figure 15: Interface de l`instructeur 61
Figure 16:Interface salon coté instructeur 62

Liste des tableaux

Tableau 1:Description de cas d'utilisations « S'authentifier » 40
Tableau 2: Description de cas d'utilisations « Gérer les profils des étudiants » 41
Tableau 3: Description de cas d'utilisations « créer les formations » 42
Tableau 4: Description de cas d'utilisations « enseigner les formations » 43
Tableau 5:Description de cas d'utilisations « Assister les formations » 43
Tableau 6::Description d'une classe 44

Liste des symboles

CPU	: Central Processing Unit
CSS	: Cascading Style Sheets
CU	: Conseil de l'Université
EAO	: Enseignement Assisté par Ordinateur
FSI	: Faculté des Sciences et de l'Ingénieur
HP	: Hewlett-Packard
HTML	: Hypertext Markup Language
HTTP	: Hypertext Transfer Protocol
ICT	: Information and Communication Technology
JS	: JavaScript
LMD	: Langage de Modélisation de Données
LMS	: Learning Management System
MOA	: Maître d'Ouvrage
MOE	: Maître d'Œuvre
MVC	: Modèle Vue Contrôleur
MySQL	: Structured Query Language
NTIC	: Nouvelles Technologies de l'Information et de la Communication
OMT	: Object Modeling Technique
OOSE	: Object Oriented Software Engineering
PCU	: Président du conseil de l'Université
PHP	: Hypertext Preprocessor
PV	: Procès Verbal
RAM	: Random Access Memory
SGBD	: Système de Gestion de Base de Données
SGBDR	: Système de Gestion de Base de Données Relationnelles
TCP/IP	: Transmission Control Protocol/Internet Protocol
TD	: Travaux Dirigés
TIC	: Technologies de l'Information et de la Communication
TNS	: Transparent Network Substrate
TP	: Travaux Pratiques
TPE	: Travaux Pratiques et Evaluations
UGL	: Université des Grands-Lacs
UML	: Unified Modeling Language
WAMP	: Windows Apache MySQL PHP

AVANT-PROPOS

Aujourd'hui, l'informatique est en pleine évolution grâce notamment à l'importance et à la place qu'il crée dans la vie quotidienne. En effet, elle permet d'économiser le temps, de conserver et protéger les données tout en rehaussant la performance et les prestations des institutions concernées.

Mon projet de fin d'études au sein de l'Université des Grands Lacs s'aligne parfaitement sur cette vision. L'objectif dudit projet est de mettre en place une plateforme d'enseignement et apprentissage à distance afin de mieux gérer l'ensemble des informations relatives au système d'enseignement académique. Nous nous sommes particulièrement intéressés aux activités réservées à l'Université des Grands Lacs.

Dans cette optique, nous avons débuté par une étude de l'existant tout en définissant les besoins fonctionnels que l'université pouvait attendre de cette mise en place ainsi que les objectifs de notre projet. Ensuite, nous avons modélisé le système en utilisant le UML enfin d'établir différents diagrammes qui nous ont été utile pour réaliser le travail. Par la suite, la théorie du file d'attente a été utilisée pour mesurer la performance du système. Nous avons entamé la partie relative à l'implémentation du système en utilisant diverses technologies comme le langage de programmation PHP et Java sous l'environnement sublime, NetBeans, JSF, système de gestion de base de données MySQL, etc.

Enfin, pour valider notre proposition, nous avons présenté un prototype de la future plateforme en langage de programmation PHP. Le travail réalisé dans notre projet constitue un moyen efficace et simple d'enseignement et apprentissage en ligne. En effet, il met à la disposition des professeurs un outil technologique de taille susceptible de réduire des difficultés liées à l'intégration des étudiants à participer à des formations et apprentissages conventionnels

I INTRODUCTION GENERALE

I.1. Justification et contextualisation

L'enseignement est une pratique, mise en œuvre par un enseignant, visant à transmettre des compétences à un étudiant, un étudiant ou tout autre public dans le cadre d'une institution éducative. Des nombreux observateurs avertissent que les structures de l'enseignement traditionnel n'ont guère changés depuis le début du XIXème siècle et ce, en dépit des bouleversements sociaux et de l'évolution des moyens de communication.

La tendance à l'amélioration du système sur le plan pédagogique par le recours aux moyens audiovisuels classiques (projections de diapositives, séquences vidéo) n'a pas résolu le problème. En effet, le formateur ou professeur doit à la fois exposer le cours et entreprendre des manipulations techniques assez difficiles. L'enseignement moderne exige des moyens pour faciliter l'apprentissage de l'étudiant tels que le choix de situation, l'explicitation d'objectifs et des critères, les choix de contenus, la mise en œuvre de procédures d'éducation, l'élaboration d'outils pour aider l'étudiant à construire et s'approprier des savoirs et des savoir-faire. Plusieurs solutions d'organisation des formations, à base des NTIC, existent sur le marché. Ces solutions proposent les différentes fonctionnalités de base (communication audio, vidéo, chat, réseaux sociaux...) [19].

Ce mode d'apprentissage est basé sur l'accès des formations en ligne, interactives et parfois personnalisées, diffusées par l'intermédiaire d'un réseau -Internet ou Intranet- ou d'un autre média électronique. Cet accès permet de développer les compétences des apprenants, tout en rendant le processus d'apprentissage indépendant du temps et du lieu. Notre travail se rattache plus particulièrement à la formation à distance en mode synchrone.

Dans cette situation, le formateur et les apprenants sont présents sur le réseau à distance et au même moment, afin que le formateur puisse apporter une aide aux apprenants de façon synchrone dans leur processus. En opposition, le mode asynchrone ne permet pas au formateur d'assister les apprenants en temps réel. Il intervient en différé.

Ce mode d'assistance synchrone est parfois une nécessité. Il est lié aux activités d'apprentissage où un problème peut s'avérer totalement bloquant pour la suite du processus d'apprentissage. L'objectif de notre étude est de mettre en place une plate-forme permettant d'assurer un suivi des activités d'apprentissage des apprenants distants et de pouvoir intervenir instantanément auprès de ces derniers.

Nous souhaitons mettre à la disposition du formateur des outils lui permettant d'assurer les différentes fonctions assignées. Grâce à cette plateforme, il est possible de posséder des outils de communication tel que : le transfert de documents, diffusion de documents, mise ne place et suivi des parcours de formation (forum, conférences...), agenda, glossaire. Sur un plan pratique, on utilise une plate-forme, qui regroupe les outils nécessaires aux principaux types d'utilisateurs : le formateur, l'apprenant et les tuteurs ou responsables [23].

L'administrateur assure les activités du système ; le formateur peut mettre ses cours en ligne, incorporer des ressources pédagogiques multimédias et effectuer un suivi des activités des apprenants ; L'apprenant consulte en ligne ou télécharge les cours, a une vue de l'évolution de son cours, effectue des exercices par leurs performances. Il peut parfois communiquer avec un formateur ou d'autres apprenants [02].

Ce mémoire s'articule autour de cinq chapitres.

En premier lieu, nous mettons l'accent sur le contexte et justification, les objectifs du projet, la problématique, l'analyse et critique de l'existant, l'intérêt du sujet, la délimitation, l'apport scientifique et technologique, le domaine d'application, la méthodologie de recherche utilisée et les outils de recherche utilisés. Le deuxième chapitre est la présentation du milieu d'étude. Dans le troisième chapitre, nous exposons la théorie du file d'attente, le quatrième chapitre est l'expression des besoins et la modélisation du nouveau système avec UML tandis que le dernier chapitre, une étude technique qui est présentée en décrivant l'environnement de développement matériel et logiciel. Nous présentons à cet effet, les différentes fonctionnalités de l'application à travers des captures d'écran.

I.2 Objectifs du projet

I.2.1 Objectif global

L'objectif global est d'implémenter une plateforme d'enseignement et d'apprentissage (LMS) digitale pour assister et accompagner les formations à distance.

I.2.2 Objectifs spécifiques

L'objectifs de ce travail est de :

1) Echanger de ressources en simultanéité ;
2) Gérer et optimiser les ressources ;
3) Créer un espace de discussion.

I.3 Problématique

Intégration des étudiants qui ont des difficultés à participer à des formations et apprentissages conventionnels. Il se heurte aux problèmes suivants :

1) Pas d'échange continu, progressif, interactif et temps réel ;
2) Incapacité d'offrir l'environnement et les ressources appropriées ;
3) Pas de gestion et l'optimisation des ressources ;
4) Manque d'une plateforme

Les constats ci-haut cités nous ont poussé et nous ont amené à formuler la question de notre recherche que voici : la conception et réalisation d'une plateforme d'enseignement et apprentissage ne serait-elle pas une solution aux problèmes ci-dessus ?

I.4 Analyse et critique de l'existant

Nous avons fait l'enquête dont nous présentons ici les résultats. Le problème, pour nous, n'était pas de changer le fonctionnement au sein de l'institut mais de relever les opinions des étudiants, professeurs, les responsables des facultés et instituts, mais de découvrir le schéma de leur organisation, la nature des liaisons les unissant, les modalités de leur acquisition et de leur sélection.

I.5 Solutions proposées

Après l'analyse des statistiques ci-haut et observation, nous proposons une conception et implémentation d'une plateforme d'enseignement regroupant les fonctionnalités d'enseignement et d'apprentissage à distance. Cette dernière englobe à la fois le partage (documents, webcam) et la formation pour faciliter la tâche du formateur et des étudiants [06].

Les solutions proposées sont les suivantes :

1) Créer une plateforme digitale pour faciliter l'enseignement et l'apprentissage ;
2) Créer en temps réel, instantané, multiplexe, broadcast, unicast et multicast un espace commun pour avoir accès aux formations ;

I.6. Intérêt du sujet

Notre travail présente un intérêt scientifique, pour le pays, pour la communauté et personnel.

Quant à la communauté scientifique, le présent projet enrichit la base de thématiques pour l'étude et la recherche scientifique appliquée. Pour le pays, notre travail sera un apport considérable pour le développement socio-économique dans les différents secteurs de la vie du pays y compris l'éducation, la santé, ...

Pour la communauté, elle disposera d'une plateforme d'enseignement et d'apprentissage à distance ce qui permettra aux étudiants d'accéder à l'enseignement via les nouvelles technologies de l'information et de la communication(NTIC) [04].

En fin, pour moi, ce travail me permettra d'améliorer mes connaissances et péser la solidité de mes compétences en la confrontant à un problème de l'enseignement digital. Il va en outre me permettre d'acquérir de l'expérience dans le domaine de la conception des plateformes d'enseignement et donc des applications informatiques orientée éducation et utilisant les architectures mobiles.

I.7 Délimitation du sujet

La digitalisation dans le secteur éducation offre une gamme d'avantages tant aux acteurs qui y sont impliqués directement ainsi qu'à l'université en général.

En ingénierie logicielle, la délimitation du sujet et du domaine d'application est une étape capitale pour la réussite d'un projet de recherches.

Dans l'espace, le présent travail traite des activités de l'UGL ainsi que ses partenaires directement et indirectement impliqués dans la vie de cette dernière.

Dans le domaine, le présent projet traite des aspects d'apprentissage et enseignement de l'UGL.

Dans le temps, notre travail a été réalisé dans le cadre d'un projet de fin d'études post universitaire pour une durée de six mois prévue pour le dernier semestre du cycle de Mastère.

I.8 Apports scientifiques et technologiques

La digitalisation pour la formation en ligne qui répond aux fonctions suivantes :

1) L'accessibilité (facilite la recherche, l'identification, l'accès aux contenus et composants de formation en ligne) ;
2) La réutilisabilité (permet d'utiliser les mêmes contenus et composants à différentes fins, dans différentes applications, dans différents produits, dans différents contextes et via différents modes d'accès) ;
3) La durabilité (permet d'éviter un nouveau développement ou des formats de contenus et des composants dans le cas de changements du support logiciel et technique) ;
4) L'adaptabilité (est rendue possible par la modularisation des contenus et des composants pour mieux répondre aux besoins des utilisateurs) ;
5) L'interopérabilité (permet de faire fonctionner ensemble des composants logiciels grâce à des interfaces communes).

I.9 Domaine d'application

Cette plateforme pourra être utilisée dans les établissements d'enseignement et d'apprentissage, dans l'optimisation des systèmes de formation digitale, etc.

I.10 Méthodologie de recherche utilisée

Ce travail s'inscrit dans le cadre des travaux permettant le renforcement des capacités et l'accessibilité dans le domaine de l'enseignement.

Lors de notre recherche, nous avons fait recours aux différentes techniques de collecte et analyse de données (entretien, l'observation, la documentation), l'analyse et modélisation des systèmes informatiques (modélisation par UML), le cycle de développement des applications informatiques (modèle en cascade ou le modèle en V) et les modèles mathématiques (la théorie du file d'attente).

I.11 Outils mathématiques utilisés

La théorie du file d'attente est une théorie mathématique relevant du domaine des probabilités, qui étudie les solutions optimales de gestion des files d'attente, ou queues. Une queue est nécessaire et se créera d'elle-même si ce n'est pas anticipé, dans tous les cas où l'offre est inférieure à la demande, même temporairement.

Elle peut s'appliquer à différentes situations : gestion des avions au décollage ou à l'atterrissage, attente des clients et des administrés aux guichets, ou bien encore stockage des programmes informatiques avant leur traitement. Ce domaine de recherches étudie notamment les systèmes d'arrivée dans une queue, les différentes priorités de chaque nouvel arrivant, ainsi que la modélisation statistique des temps d'exécution [08].

I.12. Résultats attendus

Dans le secteur de l'éducation, nous proposons d'utiliser une application web client/serveur. Cette application permet aux utilisateurs d'accéder au serveur web et de donner un résultat positif pour l'enseignement et apprentissage à distance en particulier et un avantage pour le pays en général.

Nous montrons que l'application développée puisse générer et apporter une très grande contribution dans le secteur de l'enseignement et d'autres institutions.

Conclusion

Dans ce premier chapitre, nous avons décrit le contexte et les objectifs du projet, évoqué la problématique et les solutions proposées avant de terminer par le domaine d'application de notre application. Le point suivant sera la présentation et fonctionnement de l'UGL.

Dans le chapitre suivant nous passerons à l'analyse et conception du nouveau système d'enseignement et apprentissage digital pour faciliter les étudiants à suivre les formations à distance.

CHAPITRE II PRESENTATION DU MILIEU D'ETUDE

II.1 Historique

L'Université des Grands Lacs (UGL) est le fruit de l'initiative d'un groupe d'intellectuels natifs de la province Bururi réalisée par les collectivités locales, notamment des populations des provinces de Bururi, Makamba et Rutana, des confessions religieuses en l'occurrence, l'Eglise de Pentecôte de Kiremba et le Diocèse Catholique de Bururi. C'est une association sans but lucratif qui a été créée au cours de l'année 2000 dans le souci d'épauler le Gouvernement du Burundi dans la formation de ses futurs cadres. L'UGL a été agréée par les pouvoirs publics par l'ordonnance ministérielle n°530/451 du 20 juin 2000 du Ministre de l'Intérieur et de la Sécurité Publique.

L'Université des Grands Lacs est une université de type communautaire portée par toute la communauté. C'est un projet conçu au sein de la communauté. Les promoteurs du projet ont chaque fois associé tous les partenaires potentiels dans sa conception. Ils n'ont pas élaboré un projet ailleurs qu'ils ont imposé à la communauté. Celle-ci a été associée à tout le processus depuis la conception jusqu'à la mise en place. Ils ont opté pour une démarche participative, la seule qui pouvait garantir un résultat positif.

Elle a obtenu l'autorisation d'ouverture le 24 octobre 2000 par l'ordonnance ministérielle n° 630/853 du Ministre de l'Education Nationale. Ses programmes de formations ont été agréés par ordonnance ministérielle n°610/303 du 7 avril 2004 du Ministre de l'Education Nationale. Sa première rentrée des cours a eu lieu le 8 janvier 2001 à Kiremba en province Bururi où se trouve le siège de l'Université des Grands Lacs. Au début, l'UGL avait un seul campus à KIREMBA, ce campus comptait 57 Etudiants avec deux filières de formation, Administration et Gestion des Affaires et Psychologie et Sciences de l'Education [27].

Les étudiants pouvaient se loger et se nourrir dans l'enceinte de l'Université au campus de Kiremba. Depuis l'or les effectifs n'ont pas cessé d'évoluer et l'UGL a pu ouvrir des autres campus notamment : KIREMBA, BURURI, MAKAMBA, Saint Michel, RUTANA et NGAGARA Q10.

II.2 Cahier des charges

Le cahier des charges s'applique à l'Université des Grands-Lacs exerçant ses activités sur le territoire du Burundi. L'application du présent cahier des charges, est considérée comme université tout établissement qui fédère en son sein la production (recherche), la conservation (publications et bibliothèques) et la transmission (études supérieures) de différents domaines de la connaissance, qui sont centrées sur la seule transmission d'un domaine bien défini de la connaissance.

II.2 .1 De la gouvernance universitaire

L'assemblée générale des membres fondateurs désigne les modalités de fonctionnement de l'Université tant au plan administratif qu'académique. Celles-ci doivent être contenues dans un manuel de procédure dûment validé par le Conseil d'administration. Au sens du présent cahier des charges, le conseil d'administration renvoie des activités au recteur, aux commissaires de comptes de l'université et à la direction académique, finance et comptabilité, assurance qualité et de recherche.

II.2.2 De la gouvernance administrative

Au plan administratif l'université comprend le Conseil d'administration et le Recteur. Le Conseil de direction est l'organe d'orientation et de décision. Il contrôle la gestion de l'université confiée au Recteur et délibère sur le projet de budget de l'université ainsi que sur les comptes administratifs.

Le Secrétaire exécutif coordonne les services d'appui universitaires constitués obligatoirement :

1) D'un Service chargé des finances et de la comptabilité ;
2) D'un Service chargé de la gestion et de la maintenance des infrastructures et des équipements ;
3) D'un Service médico-social ;
4) D'un Service chargé de l'hygiène et de la sécurité.

II.2.3 De la gouvernance académique

Sur le plan académique l'université comprend le Conseil académique et le Recteur. Placé sous la présidence du Recteur, le Conseil académique est composé, des chefs de facultés et instituts, d'un représentant du Ministère chargé de l'Enseignement supérieur, des représentants des Enseignants, des représentants des Etudiants et des représentants du personnel administratif et technique.

Le Secrétaire exécutif de l'Université assiste aux réunions du Conseil académique et en tient le procès-verbal.

Le Conseil délibère sur :

1) Les programmes d'enseignement et de recherche de l'université ;
2) Le régime des examens et des études ;
3) La scolarité, notamment le régime général des inscriptions et le découpage de l'année académique ;
4) Les questions disciplinaires.

II.2.4 Du personnel enseignant

Les enseignants de l'Université sont soumis, pour leur recrutement, aux mêmes conditions académiques que leurs homologues des Universités publiques. Nonobstant les dispositions de l'alinéa précédent, peuvent être recrutés comme enseignants vacataires dans les filières de formations professionnelles :

1) Les professionnels de niveau minimal Master professionnel et justifiant d'une expérience professionnelle de 3 ans au moins ;
2) Les professionnels qui ne justifient pas de ce niveau minimal, mais qui capitalisent au moins 10 ans d'expérience professionnelle.

II.2.5 De l'enseignement

L'UGL doit élaborer et développer un projet de recherche, en partenariat avec les institutions publiques ou privées évoluant dans des domaines similaires. L'agrégation de ces projets sectoriels doit constituer la charpente de la politique globale de recherche de l'Université

II.2.6 Des infrastructures

L'UGL comporte obligatoirement les infrastructures immobilières suivantes :

1) Un bâtiment administratif affecté au Conseil de direction, au Rectorat et à leurs services d'appui ;
2) Des bureaux pour les Responsables des structures composant l'Université ainsi que les Responsables de filière ;
3) Des locaux pour la tenue des Conseils ou assemblées et pour l'organisation des cérémonies universitaires ;
4) Un amphithéâtre ;
5) Au moins une salle de cours et une salle de travaux dirigés pour chaque filière de formation ;
6) Une bibliothèque avec une salle de lecture ;
7) Au moins une salle informatique avec connexion Internet ;
8) Un centre médico-social comportant une salle de consultation, un bureau de médecins, une salle de soins infirmiers, une salle d'observation de 4 lits au moins ;
9) Des bureaux pour les enseignants permanents ;
10) Une salle des professeurs avec connexion Internet ;
11) Une salle destinée aux organisations estudiantines ;
12) Un espace affecté aux activités sportives et aux loisirs ;
13) Des points d'eau potable ;
14) Des toilettes clairement identifiées garçons/filles.

II.3 Organigramme

II.3.1 Introduction

L'Université des Grands Lacs (UGL) est le fruit de l'initiative d'un groupe d'intellectuels natifs de la province Bururi réalisée par les collectivités locales, notamment des populations des provinces de Bururi, Makamba et Rutana, des confessions religieuses en l'occurrence, l'Eglise de Pentecôte de Kiremba et le Diocèse Catholique de Bururi

L'UGL est une structure dont la gestion incombe aux seuls membres adhérents, constituée des personnes physiques et morales qui s'associent et décident librement de mettre en commun tout ou une partie de leurs ressources ainsi que leurs efforts en vue d'exercer des activités de développement dans le secteur de l'éducation. La structure organisationnelle de l'UGL est reprise dans l'organigramme ci-dessous.

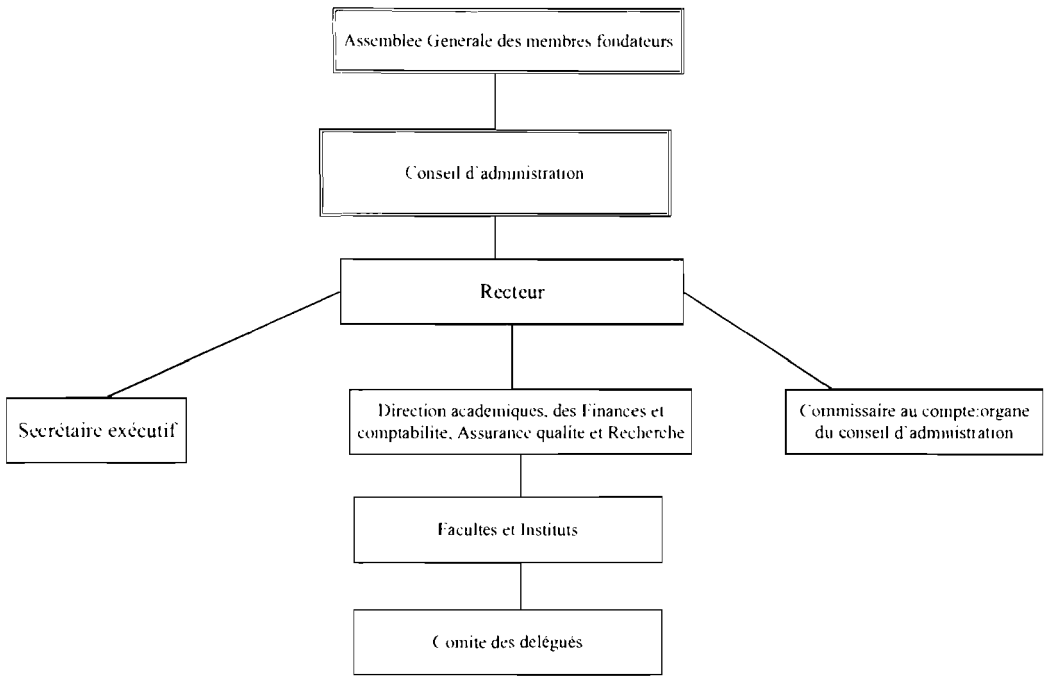


Figure 2:Organigramme de l'UGL

II.4 Manuel de procédure

II.4.1 Conseil de l'université

L'UGL a été agréée par les pouvoirs publics par l'ordonnance ministérielle n°530/451 du 20 juin 2000 du Ministre de l'Intérieur et de la Sécurité Publique.

Elle a obtenu l'autorisation d'ouverture le 24 octobre 2000 par l'ordonnance ministérielle n° 630/853 du Ministre de l'Education Nationale. Ses programmes de formations ont été agréés par ordonnance ministérielle n°610/303 du , avril 2004 du Ministre de l'Education Nationale.

Rappel des attributions du Conseil d'Université.

Le manuel des procédures est un document qui, sous la forme d'instructions claires et précises, contient l'ensemble des opérations courantes de l'institut. Il indique le circuit de traitement de ces opérations en spécifiant notamment les : tâches à faire, niveaux de

responsabilités. différentes étapes de traitement, lieux de réalisation, modalités d'exécution.

Les opérations suivantes sont dans le règlement d'ordre intérieur de l'université : réparation de la réunion, la réunion du CU, l'assemblée de faculté ou d'institut, le fonctionnement de l'Assemblée de Faculté ou d'Institut, le conseil des professeurs, le Conseil interne d'assurance qualité, les missions, les généralités sur les différents concepts, les programmes de formation

Et les enseignements [24].

Conclusion

Dans ce chapitre, il a été question de présenter le cadre de l'étude sur ce, les conditions et processus d'apprentissage et enseignement et d'analyse du système d'enseignement au sein de l'UGL dans lesquelles les recherches ont été menées.

CHAPITRE III OPTIMISATION D'UNE PLATEFORME D'ENSEIGNEMENT A DISTANCE PAR LA THEORIE DU FILE D'ATTENTE

Toutes les sciences naturelles ont en commun une caractéristique de construction des modèles. Avec l'évolution économique et technologique, les mathématiques jouent un grand rôle. Elles sont utilisées dans de nombreux projets de systèmes de gestion pour la prédiction des résultats approchés par des simulations après les calculs sur des ordinateurs. Les mathématiques sont aussi indispensables dans l'innovation lors du design de nouveaux produits industriels qui est souvent réalisé à l'aide des modèles mathématiques et de la simulation numérique.

Notre projet utilise une architecture client-serveur. Le serveur le mieux adapté à notre système est Apache. Les clients se connectent au serveur par des requêtes SQL pour demander des données spécifiques. On doit optimiser le serveur pour qu'il puisse s'adapter à toutes les requêtes issues des clients pour que tout le monde soit servi sans plus de retard.

Il existe deux modes de gestion des requêtes des clients par le serveur Apache : le mode itératif et le mode concurrent.

Pour le mode itératif qui n'est pas fiable aux applications web, le processus serveur traite les requêtes les unes après les autres. Quant au mode concurrent qui est très fiable, le serveur exécute plusieurs requêtes simultanément grâce au schéma veilleur exécutants **[32]**.

Pour prouver la fiabilité du fonctionnement de notre système, nous avons utilisé les notions mathématiques des files d'attente. Notre serveur étant configurable, on peut déterminer le nombre des clients susceptibles d'être accueilli et servi en une unité de temps. On peut donc augmenter dans Apache Config, les paramètres 'Server Limit' qui est le nombre maximal des processus qu'Apache peut lancer simultanément et 'Max Request Workers qui est le nombre maximal que le serveur peut servir en même temps. Ces deux variables portent toujours des valeurs égales et permettent de déterminer le nombre de requêtes SQL qu'Apache peut traiter en même temps **[25]**.

III. 1 Systèmes à temps discret

Un système est dit à temps discret si l'ensemble T est un ensemble discret.

Les systèmes discrets peuvent être synchrones si les variables du système prennent leurs valeurs selon une fréquence prédéfinie. Exemple : T est l'ensemble des nombres pairs.

Ils peuvent être asynchrones si les instants de temps sont distribués aléatoirement.

Ces systèmes sont aussi appelés systèmes à événements discrets.

La simulation à événements discrets est la simulation d'un système dont l'état ne change qu'à des instants temporels distincts.

En langage mathématique, on peut dire que l'ensemble d'instants temporels auxquels le système peut changer est dénombrable.

Un événement est donc la circonstance qui permet au système de changer son état. S'il n'y a pas d'événements, le système reste toujours en état stable **[31].**

III.2 Notion des files d'attente

Les files d'attente constituent un exemple du système classique à événements discrets. Il s'agit d'un modèle qui représente l'accès séquentiel d'un ensemble d'utilisateurs appelés aussi clients à un nombre limité de ressources en provenance du serveur. La théorie des files d'attente et la simulation peuvent prédire les performances du système par des analyses en fonction des paramètres qui sont contrôlable par l'administrateur du système.

L'objectif d'analyse des files d'attente est de minimiser la somme de deux coûts : le coût d'attente des clients et le coût de service. C'est pourquoi lors de la conception des files d'attente, il est indispensable de trouver un bon équilibre entre le nombre de services, l'utilisation des services et la satisfaction des clients.
On appelle file d'attente l'ensemble des clients qui attendent le service à l'exception de celui qui est en train d'être servi. Le système d'attente est donc l'ensemble des clients qui font la queue, y compris celui qui se fait servir à un instant donné **[15].**

III.3 Caractéristiques d'un système des files d'attente

Avec les files d'attente, on peut concevoir plusieurs modèles d'analyse. Le succès de l'analyse des files d'attente repose surtout sur le choix du modèle approprié. Plusieurs caractéristiques sont à prendre en considération : la population, le nombre de serveurs, la capacité du système au service et l'ordre de traitement des clients **[10]**.

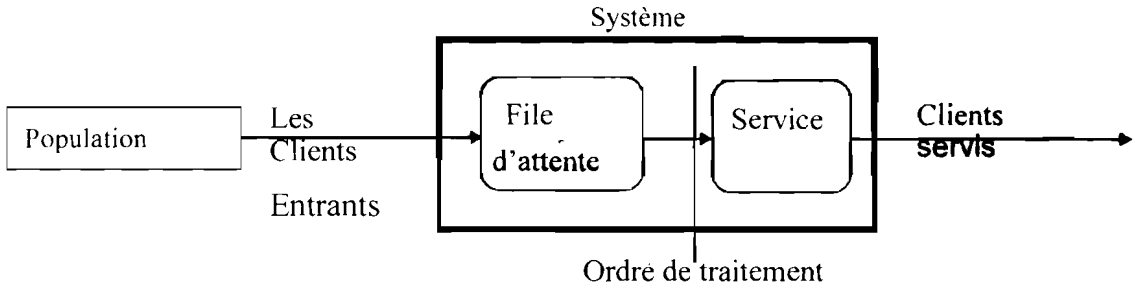


Figure 3: Structure de base d'une file d'attente

1) **Population** : Elle constitue la source de clients potentiels. Elle est caractérisée par son nombre fini ou infini.
2) **File d'attente :** Elle est caractérisée par le nombre permis de clients en attente. Ce nombre peut être fini ou infini.
3) **Clients :** les clients issus de la population se joignent au système avec un taux moyen d'arrivée.
4) **Service :** Le service est assuré par un ou plusieurs serveurs. Le temps qui s'écoule entre le début de service d'un client se dénote le temps de service suivant une loi de probabilité. Le taux de service est une autre caractéristique du système.
5) **Stratégie de service :** La stratégie de service réfère à l'ordre selon lequel les clients sont servis : premier arrivé premier servi (PAPS), au hasard, selon les priorités, ...

III.4 Notation de Kendall

Pour tout système de production de biens ou services, il existe cinq outils de mesure ou indices pour analyser et évaluer les performances du système qui existe déjà ou le système qu'on prévoit concevoir.

Ces mesures sont : le nombre moyen de clients qui attendent dans la file ou dans le système, le temps moyen d'attente dans la file et dans le système, le taux d'utilisation du système, le cout associé au service mis en place et la probabilité qu'un client éventuel attende pour être servi.

Le système de file d'attente se note comme suit :

A/S/C (DS/K/L)

A/S/C/K/L (DS)

A/S/C/K/L/DS

Avec

A : processus d'arrivée
S : processus de sortie
C : nombre de serveurs
K : capacité maximale de la file
L : population de clients
DS : discipline de service

Les cas les plus courants pour les premières positions sont :

G : pour la loi arbitraire ; M : pour symboliser une loi exponentielle ; EK : pour indiquer la loi Gamma d'ordre k ; D : pour la loi déterministe.

Processus d'arrivée de Poisson

Soit $N(t) = \max i : t_i \leq t$ le nombre aléatoire d'arrivées à ou avant l'instant $t \geq 0$, où t_i est l'instant d'arrivée du $i$ème client.

Définition

Le processus stochastique $N(t)$ est appelé un processus de Poisson si

1. Les clients arrivent un à la fois.
2. Le nombre d'arrivées dans l'intervalle $(t, t + s]$, c.-à-d. la variable aléatoire $N(t + s) - N(t)$ est indépendant de $N(u)$, $0 \leq u \leq t$.
3. La distribution de $N(t + s) - N(t)$ est indépendante de t pour tout $t > 0$, $s > 0$.

Théorème de Denis Poisson (1838)

Si N(t) est un processus de Poisson alors le nombre d'arrivée dans chaque intervalle de taille s est une variable aléatoire de Poisson avec paramètre

λs *avec* $\lambda > 0$, *c.-à-d.* Prob $\{N(t + s) - N(t) = k\} = \exp^{-\lambda s}, k = 0, 1, 2, \ldots, t, s \geq 0$

Si N(t) est un processus de Poisson avec un taux λ alors les variables $A_i = t_i - t_{i-1}$ sont distribués selon la loi exponentielle de paramètre λ.

Si X suit une loi de poisson de paramètre λ, soit $X \sim \mathcal{P}(\lambda)$.

Alors, on a par définition que $\mathbb{P}(X = k) = e^{-\lambda}\frac{\lambda^k}{k!}$ et que:

$$\mathbb{E}(X) = \sum_{k=1}^{+\infty} k\,\mathbb{P}(X = k)$$

$$\mathbb{E}(X) = \sum_{k=1}^{+\infty} k\,e^{-\lambda}\frac{\lambda^k}{k!}$$

$$\mathbb{E}(X) = e^{-\lambda}\sum_{k=1}^{+\infty}\frac{\lambda^k}{(k-1)!}$$

$$\mathbb{E}(X) = \lambda e^{-\lambda}\sum_{k=1}^{+\infty}\frac{\lambda^{k-1}}{(k-1)!}$$ (cf. développement en série entière de e^x)

$$\mathbb{E}(X) = \lambda e^{-\lambda} e^{\lambda} = \lambda$$

III.5 Distribution exponentielle

Une variable continue A est distribué selon une loi de probabilité exponentielle de paramètre $\lambda > 0$ ($A \sim \mathbf{E}(\lambda)$) si sa densité de probabilité est

$$pA(a) = \begin{cases} \lambda & \textbf{si } a \geq 0 \\ 0 & \textbf{si } a < 0 \end{cases}$$

La moyenne de A est $1/\lambda$.

La variance de A est $1/\lambda^2$.

Elle satisfait la memoryless

Prob $\{A \geq a_1 + a_2 | A \geq a_1\}$ = Prob $\{A \geq a_2\}$

Cette variable est normalement utilisée pour décrire la durée de vie d'un phénomène sans vieillissement, c'est-à-dire où la durée de vie au-delà de l'instant t est indépendante de l'instant t.

En lien avec cette loi exponentielle on a quelques variables utiles :

λ : Le taux d'arrivée : le nombre moyen d'arrivées par unité de temps.

$\frac{1}{\lambda}$: L'intervalle de temps moyen séparant deux arrivées consécutives.

μ : Le taux de service : le nombre moyen de clients servis par unité de temps.

- : Temps moyen de service d'un client dans le système.

ρ : Taux d'utilisation du système

Dans la théorie des files d'attente l'étude se fait une fois que le système atteint sa situation d'équilibre ; où les états du système sont essentiellement indépendants de l'état initial et du temps déjà écoulé.

La loi de Little

La loi de Little indique au contraire que pour réduire les temps de traversée (améliorer les délais), il faut réduire les en-cours ou augmenter le débit. Or il s'avère qu'un processus de production rencontre toujours un goulet (théorie des contraintes) limitant de fait le débit de production : la seule action efficace à conduire lorsque l'on souhaite réduire les temps de cycle reste donc de réduire les en-cours (en particulier les stocks).

La loi de Little permet également de comprendre la relation entre taille de lot et temps de traversée d'une ligne de production. WIP représente alors la taille du lot et 1/T représente le temps de cycle du processus de production : plus les lots sont importants, plus les délais de traitement de ces lots sont importants et moins l'entreprise est agile pour répondre aux demandes du marché.

$$\bar{N} = \lambda_e \bar{T}$$

Où λ_e est le taux d'entrée dans le système ($\lambda_e = \lambda$ pour une file M/M/1). Puisque $\bar{N} = \overline{N_Q} - \overline{N_S}$ et $\bar{T} = \overline{T_Q} + \bar{T}_S$, on trouve également que

$$\overline{N_Q} = \lambda_e \overline{T_Q} \text{ et } \overline{N_S} = \lambda_e \bar{T}_S$$

La loi de Little s'applique à tous les modèles de file d'attente rencontrés en pratique (pas seulement à la file M/M/1).

Application de la loi de Little

Pour tout système de production de biens ou services, il existe cinq outils de mesure ou indices pour analyser et évaluer les performances du système qui existe déjà ou le système qu'on prévoit concevoir. Ces mesures sont : le nombre moyen de clients qui attendent dans la file ou dans le système, le temps moyen d'attente dans la file et dans le système, le taux d'utilisation du système, le cout associé au service mis en place et la probabilité qu'un client éventuel attende pour être servi.

Exemples : Les lauréats des différentes universités qui terminent leurs études, la file d'attente au parking de Bujumbura ou la croissance démographique dans notre pays.

Utilisation d'un serveur :

$\overline{N_S} = 1 - \pi_0 = \rho$ avec $\overline{N_S}$ =probabilité qu'un serveur soit en cours d'utilisation,

π_0 =probabilité qu'un serveur ne soit pas utilisé en régime permanent

$\overline{N_Q} = \overline{N} - \overline{N_S} = \frac{\rho^2}{1-\rho}$

$\overline{T} = \overline{N} / \lambda \ \frac{\rho}{\lambda(1+\rho)} = \frac{1}{(\mu-\lambda)}$

$\overline{T_S} = 1/\mu$

$\overline{T_Q} = \overline{T} + \overline{T_S} = \frac{\lambda}{\mu(\mu-\lambda)}$

Un seul serveur : $N_Q = \begin{cases} 0 \text{ si } N = 0 \text{ ou } N = 1 \\ N + 1 \text{ si } N > 1 \end{cases}$

$P(NQ = 0) = P(N = 0) + P(N = 1) = \pi_0 + \pi_1 = 1 - \rho + \rho(1 - \rho) = (1 - \rho)(1 + \rho)$.

$P(NQ = k) = P(N = k + 1) = \pi_{k+1} = \rho^k (1 - \rho)$, pour $k > 0$.

Si N est le nombre de clients dans le système à l'équilibre, alors $N + 1 = N_1 \sim$ Géom. $(p = 1 - \rho)$.

Nombre de clients en train d'être servis : $N_S \sim$ Bern(ρ).

Temps total (attente + service) passe dans la file : $T \sim$ Exp $(\mu - \lambda)$.

Temps d'attente TQ (variable mixte) :

$P(TQ = 0) = \pi_0 = 1 - \rho$.

$TQ_i \mid N_Q > 0g \sim Exp\ (\mu - \lambda)$ (comme T).

En situation d'équilibre on note :

P_n : Probabilité qu'il y ait n clients dans le système.

L : Nombre moyen de client dans le système.

L_q : Nombre de clients dans la file d'attente excluant ceux qui sont dans le service.

W : Le temps moyen passé par un client dans le système.

W_q : Le temps moyen passée par un client dans la file (excluant le temps de service)

III.6 Principaux modèles des files d'attente

Il existe plusieurs modèles des files d'attente. Voici quelques exemples :

Modèle M/M/1

1) Les clients se présentent suivant une loi de Poisson ;
2) Le temps de service suit une loi exponentielle de taux μ indépendamment d'un client à l'autre ;
3) La file d'attente peut s'étendre à l'infini ;
4) C'est un modèle à serveur unique.

Modèle M/M/∞

1) Les arrivées suivent une loi de Poisson ;
2) Les temps de service suivent une loi exponentielle ;
3) Le nombre de serveurs est infini. Les clients sont servis simultanément et indépendamment les uns des autres.

Modèle M/G/1

1) Les arrivées suivent une loi de Poisson ;
2) Les temps de service sont arbitrairement distribués.

Pour notre cas, nous aurons un seul serveur et la file est à population infinie. Nous nous intéresserons au premier modèle.

Relations de base

Dans les modèles de file d'attente à population infinie, il existe certaines relations de base qui permettent de déterminer les mesures de performances désirées grâce à quelques valeurs clés. Les principales relations sont présentées ci-dessous :

Notons d'abord que les taux d'arrivée λ et de service μ seront exprimés dans la même unité de mesure.

Le taux d'utilisation du système : c'est le rapport entre la demande qui s'exprime par le taux d'arrivée λ et la capacité de service donnée par μ Multiplié par le nombre de serveurs C. Pour un système à serveur unique (C = 1), le nombre de clients en train d'être servis est donné par : $\rho = \lambda / \mu$ .Le nombre moyen des étudiants en file : L_q est obtenu à partir de la formule appropriée, selon le modèle en question.

Le nombre des étudiants dans le système :

$$L = Lq + \rho$$

Le temps moyen d'attente passé en file : $W_q = \frac{Lq}{\lambda}$

Le temps moyens d'attente dans le système $W = W_q \frac{L}{\mu} = \frac{L}{\lambda}$:

Processus de naissance et de mort

Une file d'attente peut être considéré comme un processus de naissance et de mort du fait que le temps qui s'écoule entre deux arrivées et le temps de service sont distribués exponentiellement.

Naissance = Arrivée du client qui demande du service ;

Mort = départ d'un client dans le système après son service.

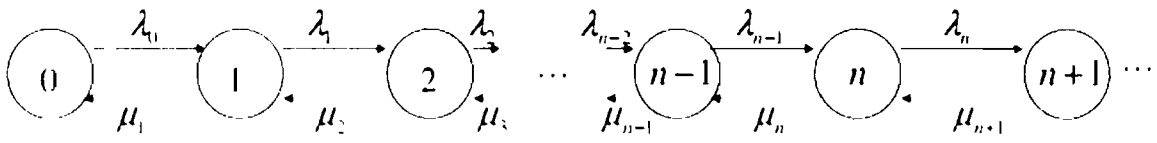
Dans le processus de naissance et de mort :

Les temps s'écoulant entre deux naissances consécutives et entre deux morts consécutifs sont tous les deux distribués exponentiellement.

A partir de l'état n, le temps de transition est de type :

$n \rightarrow n+1$ (pour une seule naissance) et/ou $n \rightarrow n-1$ (pour une seule mort).

III.7 Diagramme de transition entre les états



1) λ_n = Taux moyen de naissance lorsque n individus sont dans le système.
2) μ_n = Taux moyen de mort lorsque n individus sont dans le système.

Le processus de naissance et de mort peut être considéré comme une chaîne de Markov en temps continu où les densités de transitions sont spécifiées à l'aide des λ_n et μ_n .
Soient les variables

1) $\pi_j q_j$ = taux auquel le processus passe de j puisque π_j = probabilité (à l'équilibre) que le processus soit dans l'état j
2) q_j = taux de transition de sortir de l'état j étant donné que le processus est dans l'état j
3) $\pi_i q_{ij}$ = taux de passage de l'état i à l'état j puisque
4) q_{ij} = taux de transition de l'état i à l'état j étant donné que le processus est dans l'état i.

$$\sum_{\substack{i=0 \\ i \neq j}}^{M} (\pi_i q_{ij}) \qquad (1)$$

Cette formule ci-dessus exprime le taux de passer à l'**état** j quel que soit l'état i dans lequel se trouve le processus.

Nous ne constatons donc que le **taux de départ de j = taux d'arrivée à j.**

Nous allons donc utiliser les équations de balance suivantes : Equations d'équilibres

$$\pi_j q_j = \sum_{\substack{i=0 \\ i \neq j}}^{M} \pi_i q_{ij} \qquad j \in \{0, \ldots, M\}$$
$$\pi_j q_j = \sum_{j=0}^{M} \pi_j = 1 \tag{2}$$

Intensité de transition

$$q_j = \sum_{\substack{i=0 \\ i \neq j}}^{M} q_{ji} \tag{3}$$

En remplaçant les valeurs des q_j dans les équations d'équilibre, on aura

$$\pi_j q_j = \sum_{\substack{i=0 \\ i \neq j}}^{M} \pi_i q_{ij} \Rightarrow \pi_j \sum_{\substack{i=0 \\ i \neq j}}^{M} q_{ji} = \sum_{\substack{i=0 \\ i \neq j}}^{M} \pi_i q_{ij} \qquad j \in \{0, \ldots, M\} \tag{4}$$

Les équations d'équilibre deviennent

$$\pi_j \sum_{\substack{i=0 \\ i \neq j}}^{M} q_{ji} = \sum_{\substack{i=0 \\ i \neq j}}^{M} \pi_i q_{ij} \qquad j \in \{0, \ldots, M\}$$
$$\sum_{j=0}^{M} \pi_j = 1 \tag{5}$$

On a : **taux de départ de j = taux d'arrivée à j.**

On peut maintenant appliquer les équations d'équilibre pour déterminer les probabilités P_n à l'équilibre.

Les équations de balance deviennent

$$\pi_j \sum_{i \neq j} q_{ji} = \sum_{i \neq j} \pi_i q_{ij} \qquad j = 0,1,2,\ldots$$
$$\sum_j \pi_j = 1 \tag{6}$$

Pour $n = 0$

$$P_0 \lambda_0 = P_1 \mu_1$$

Pour $n = 1,2,\ldots$

$$P_n(\lambda_n + \mu_n) = P_{n-1}\lambda_{n-1} + P_{n+1}\mu_{n+1}$$

Pour les états n

0 $\quad P_1 = \frac{\lambda_0}{\mu_1} P_0$

1 $\quad P_2 = \frac{\lambda_1}{\mu_2} P_1 - \frac{1}{\mu_2}(\mu_1 P_1 - \lambda_0 P_0) = \frac{\lambda_1}{\mu_2} P_1 = \frac{\lambda_1}{\mu_2}\frac{\lambda_0}{\mu_1} P_0$

2 $\quad P_3 = \frac{\lambda_2}{\mu_3} P_2 - \frac{1}{\mu_3}(\mu_2 P_2 - \lambda_1 P_1) = \frac{\lambda_2}{\mu_3} P_2 = \frac{\lambda_2}{\mu_3}\frac{\lambda_1}{\mu_2}\frac{\lambda_0}{\mu_1} P_0$

$\vdots$

n $\quad P_{n+1} = \frac{\lambda_n}{\mu_{n+1}} P_n - \frac{1}{\mu_{n+1}}(\mu_n P_n - \lambda_{n-1} P_{n-1})$

$$= \frac{\lambda_n}{\mu_{n+1}} P_n = \frac{\lambda_n}{\mu_{n+1}} \cdots \frac{\lambda_2}{\mu_3}\frac{\lambda_1}{\mu_2}\frac{\lambda_0}{\mu_1} P_0$$

$\vdots$

Pour simplifier

$$P_n = \frac{\prod_{i=0}^{n-1} \lambda_i}{\prod_{i=1}^{n} \mu_i} P_0 \qquad n = 1,2,\ldots$$

Pour déterminer P_0, on utilise

$$1 = \sum_i P_i = P_0 + \sum_{n\geq 1} \frac{\prod_{i=0}^{n-1} \lambda_i}{\prod_{i=1}^{n} \mu_i} P_0 = P_0 \left[1 + \sum_{n\geq 1} \frac{\prod_{i=0}^{n-1} \lambda_i}{\prod_{i=1}^{n} \mu_i} \right] \Leftrightarrow P_0 = \frac{1}{1 + \sum_{n\geq 1} \frac{\prod_{i=0}^{n-1} \lambda_i}{\prod_{i=1}^{n} \mu_i}} \quad (7)$$

$$\pi_i \sum_{i \neq j} q_{ji} = \sum_{i \neq j} \pi_i q_{ji} \quad j = 0,1,2,\ldots$$

$$\sum_j \pi_j = 1$$

File d'attente infinie à serveur unique (C = 1) : M/M/1

Considérons le modèle de file d'attente où les arrivées et les départs se produisent comme dans l

$\lambda_n \equiv \lambda$ pour $n = 0, 1, 2, \ldots$ et

$\mu_n \equiv \mu$ pour $n = 1, 2, \ldots$ indépendants du nombre de clients dans le système.

Alors

$$P_0 = \frac{1}{1 - \sum_{n\geq 1} \frac{\prod_{i=0}^{n-1} \lambda_i}{\prod_{i=1}^{n} \mu_i}} = \frac{1}{\sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n} \quad (8)$$

Sous l'hypothèse que le taux d'arrivée est plus petit que le taux de service $\lambda < \mu$, on a $\frac{\lambda}{\mu} < 1$, et la progression géométrique

$$\sum_{n=1}^{\infty} \left(\frac{\lambda}{\mu}\right)^n = \frac{1}{1 - \frac{\lambda}{\mu}}$$

Les Caractéristiques à serveur unique :

M : Arrivées poissonniennes, taux λ

M : Service exponentiel, taux μ

1 : serveur

$\rho = \frac{\lambda}{\mu}$ il s'en suit que

$P_n = (1 - \rho)\rho^n$ avec $n = 0, 1, 2, \ldots$

Calcul des caractéristiques de la file d'attente M/M/1

1. Nombre moyen des étudiants dans la file (L)

$$L = \sum_{n=0}^{\infty} nP_n = \sum_{n=0}^{\infty} n(1-\rho)\rho^n = (1-\rho)\rho \sum_{n=0}^{\infty} n\rho^n = (1-\rho)\rho \sum_{n=0}^{\infty} \frac{d}{d\rho}(\rho)^n \quad (9)$$

$$\Rightarrow L = (1-\rho)\rho \frac{d}{d\rho} \sum_{n=0}^{\infty} (\rho)^n$$

$$= (1-\rho)\rho \frac{d}{d\rho}\left(\frac{1}{1-\rho}\right) = (1-\rho)\rho \frac{1}{(1-\rho)^2}$$

$$= \frac{\rho}{1-\rho} = \frac{\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}} = \frac{\lambda}{\mu-\lambda}$$

$$\mathbf{L = \frac{\lambda}{\mu - \lambda}}$$

Le calcul des différentes caractéristiques se base sur les paramètres suivants :

Nombre de serveurs : m ; Capacité de la file : k clients (y compris ceux qui reçoivent du service) ; Distribution des services : Durée moyenne ; Distribution des inter-arrivées : durée moyenne et nombre total de clients N.

2. Nombre moyen des étudiants dans file d'attente (Lq)

$$L_q = \sum_{n=1}^{\infty}(n-1)P_n = \sum_{n=0}^{\infty} nP_n - \sum_{n=1}^{\infty} P_n$$

$$= L - (1 - P_0) = \frac{\lambda}{\mu-\lambda} - \frac{\lambda}{\mu}$$

$$\Rightarrow L_q = \frac{\lambda^2}{\mu(\mu-\lambda)} \qquad (10)$$

3. Temps moyen qu'un étudiant passe dans le système

$$W = \frac{L}{\lambda} = \frac{\lambda}{\mu-\lambda}\frac{1}{\lambda} = \frac{1}{\mu-\lambda} \qquad (11)$$

4. Temps moyen qu'un étudiant passe dans la file d'attente

$$W_q = \frac{L_q}{\lambda} = \frac{\lambda^2}{\mu(\mu-\lambda)}\frac{1}{\lambda} = \frac{\lambda}{\mu(\mu-\lambda)} \qquad (12)$$

III.8 Vérification des performances du serveur de notre application d'apprentissage en ligne

La vérification se fait sur base des données relatives à l'effectif d'internautes de la population africaine d'une part et mondiale d'autre part. Les statistiques prouvent qu'**en 2017, l'**Afrique comptait 362 million de personnes utilisant l'internet sur 1,231**million** de toute la population africaine soit 29%. La planète Terre comptait 7,476 milliards de personnes dont 50 % sont des internautes (3,773 milliards) au mois de janvier.

Pour les réseaux sociaux, on dénombre 2,789 milliard d'utilisateurs, et on 2 ,549 milliards d'utilisateurs de réseaux sociaux sur mobile. Ce nombre d'internautes a atteint 4 milliards en février 2018 soit 53% **[10] [26].** En 2016, Google a totalisé 2000 milliards de requêtes soit 3,8million de requêtes par minute. En 2017, chaque mois, YouTube recevait en moyenne 1,5 milliards d'utilisateurs qui regardaient ses vidéos **[27] [28].**

Exemple

Nous prévoyons recevoir 3800 requêtes SQL par minute en moyenne avec un temps de service d'une seconde pour 150 requêtes simultanément (Server Limit). On va calculer les performances suivantes :

1) Le taux d'utilisation du système
2) Le nombre moyen de requêtes SQL dans la file
3) Temps moyen qu'une requête SQL passe dans la file
4) Temps moyen qu'une requête SQL passe dans le système
5) Temps inactif du serveur en pourcentage

Probabilité qu'il y ait 180 requêtes SQL dans le système et probabilité qu'il n'y ait aucune requête SQL dans le système On estime que le taux d'arriver des requêtes SQL s'accroit de 10 fois. Cherchons une augmentation du Server Limit nécessaire pour supprimer le temps d'inactivité ?

Résultats

- λ = 3800 requêtes minute : le temps de service = 1seconde/150 requêtes = $\frac{1}{\mu}$

  $\mu = \frac{1}{1}\frac{150\ requetes}{seconde} *$ 60 secondes = 9000 requêtes/min

  - $\rho = \frac{\lambda}{C\mu} = \frac{3800}{9000} = 0.42 = 42\%$
  - $L_q = \frac{\lambda^2}{\mu(\mu-\lambda)} = \frac{(3800)^2}{9000(9000-3800)} = \frac{14440000}{46800000} = 0.308 * 150 = 46.2$ requêtes
  - $W_q = \frac{\lambda}{\mu(\mu-\lambda)} = \frac{3800}{9000(9000-3800)} * 150 * 60$ secondes = 0.730 secondes
  - $W = \frac{1}{\mu-\lambda} = \frac{1}{9000-3800} * 150 * 60$ secondes = 1.730 secondes
  - Temps d'inactivité = 1- ρ = 1- 0.42 = 0.58 = 58 %
  - $P_n = (1-\rho)\rho^n$ et $P_0 = 1 - \frac{\lambda}{\mu}$
    - $P_{150} = (1-0.42)0.42^{150} = 0.877 * 10^{-70} \approx 0$

- $P_0 = 1 - \frac{\lambda}{\mu} = 1 - \frac{3800}{9000} = 1 - 0.42 = 0.58$

- $\lambda = 3800 * 10 = 38000$ requêtes minute

Soit l la variable du server Limit

$\mu = l * 60$ secondes pour notre cas

$\rho = \frac{\lambda}{C\mu} = \frac{\lambda}{l*60} = 0.42$

En supprimant le temps d'inactivité, $\rho = 1$ (taux maximal), on a :

$\rho = \frac{\lambda}{C\mu} = \frac{\lambda}{l*60} = 1$

Avec $\lambda = 10\lambda$, on aura :

$l = \frac{\lambda}{1*60} = \frac{38000}{1*60} = 633.3$

A partir des requêtes SQL, nous pouvons en déduire :

1) Le temps nécessaire à une date précise ;
2) Le nombre minimal ou maximal de temps à une telle date ;
3) Le temps moyen utilisé à une date indiquée :
4) Le nombre de liens visités dans un intervalle de temps ;
5) Le nombre minimal ou maximal de liens utilisés ;
6) Le temps moyen qu'il faut pour passer d'un lien à un autre ;
7) La durée pour chaque utilisateur ;
8) La durée minimale ou maximale par utilisateur ;
9) La durée moyenne qu'il faut pour chaque utilisateur.

Si le taux d'arrivées des requêtes SQL s'augmente de 10 fois, on pourra annuler le temps d'inactivité en passant le Server Limit de 150 à 633.3

Conclusion

Notre système est assimilé à un système de files d'attente avec des serveurs parallèles. La file d'attente est constitué par les étudiants qui arrivent sur la plateforme et les serveurs sont les professeurs qui dispensent les cours. Les étudiants arrivent suivant un processus de taux d'arrivée moyen $\boldsymbol{\lambda}$. Le temps de service est $\boldsymbol{\mu}$.

Avec ce modèle, nous avons constaté qu'il faudrait augmenter le nombre de serveur pour ramener l'intensité du trafic à une valeur raisonnable qui est inférieure à 1.

CHAPITRE IV EXPRESSION DES BESOINS ET MODELISATION DU SYSTEME D'INFORMATION D'ENSEIGNEMENT A DISTANCE

IV.1 Concepts de base et méthodologies

IV.1.1 Concepts

L'analyse : C'est un processus consistant à nettoyer, transformer, et modéliser des données. Extraire des informations exploitables pour prendre de meilleures décisions au sein d'une entreprise.

La modélisation : C'est une représentation d'un système réel quelle qu'en soit sa forme physique, graphique, mathématique, verbale ou mentale. Cette représentation intelligible est indispensable pour assurer la compréhension de systèmes naturels complexes.

Le cycle de développement : Il désigne toutes les étapes du développement d'un logiciel, de sa conception à sa réalisation.

UML (Unified Modeling Language) : C'est une représentation simplifiée de tout ou partie d'un système d'information existant ou futur, mettant en évidence certains aspects essentiels.

Les activités d'un cycle de vie d'un logiciel (en anglais *software lifecycle*) : Le cycle de vie suivi pour réaliser une application web, comprend généralement au minima les activités suivantes :

1) Spécification des besoins : elle consiste à définir la finalité du projet et son intégration dans une stratégie globale.
2) Conception générale : dans cette activité, il s'agit de la préparation de l'architecture générale du logiciel.
3) Conception détaillée : elle consiste à définir précisément chaque sous-ensemble du logiciel.
4) Développement : (Implémentation ou programmation) il s'agit d'une traduction des fonctionnalités définies dans la phase de conception en langage de programmation.

5) Tests unitaires : ils permettent de vérifier individuellement que chaque sous-ensemble du logiciel est implémenté conformément aux normes définies dans la conception.
6) Intégration : dite aussi tests systèmes, elle consiste à vérifier que le logiciel correspond exactement au cahier des charges du projet en obtenant enfin un manuelle d'utilisation bien détaillé aux utilisateurs.
7) Validation : c'est-à-dire la validation de conformité du site avec les buts spécifiés à la première étape du cycle de vie.

IV.1.2 Méthodologies

Il existe plusieurs techniques de collecte de données parmi lesquelles on distingue :

Techniques d'analyse des documents et rapports ; Interview et entretien avec le personnel de l'institut ; Questionnaire d'enquête sur un groupe de personne ciblé de l'institut.

IV.2 Présentation de la méthode UML

Les présentes recherches se sont basées sur les démarches de la méthode UML pour aboutir au résultat escompté. UML (Unified Modeling Language) est un langage formel et normalisé en termes de modélisation objet. Son indépendance par rapport aux langages de programmation, aux domaines de l'application et aux processus, son caractère polyvalent et sa souplesse ont fait lui un langage universel. En plus UML est essentiellement un support de communication, qui facilite la représentation et la compréhension de solution objet.

Sa notation graphique permet d'exprimer visuellement une solution objet, ce qui facilite la comparaison et l'évaluation des solutions. L'aspect de sa notation, limite l'ambigüité et les incompréhensions [11].

UML fournit un moyen astucieux permettant de représenter diverses projections d'une même représentation grâce aux vues. Une vue est constituée d'un ou plusieurs diagrammes.

On distingue deux types de vues :

1) La vue statique, permettant de représenter le système physiquement :

Diagrammes de classes : représentent des collections d'éléments de modélisation statiques (classes, paquetages...), qui montrent la structure d'un modèle ; Diagrammes de cas d'utilisation : identifient les utilisateurs du système (acteurs) et leurs interactions avec le système.

2) La vue dynamique, montrant le fonctionnement du système :

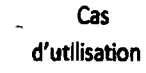
Diagrammes de collaboration : montrent des interactions entre objet (instances de classes et acteurs). Diagrammes de séquence : permettent de représenter des collaborations en objets selon un point de vue temporel, on y met l'accent sur la chronologie (envois de messages). Diagrammes d'états-transitions : permettent de décrire les changements d'états d'un objet ou d'un composant, en réponse aux interactions avec d'autres objets/composants ou avec des acteurs. Diagrammes d'activités : (une variante des diagrammes d'états-transitions) servent à représenter graphiquement le comportement d'une méthode ou déroulement d'un cas d'utilisation.

3) La conception de l'interface a été élaborée en suivant la démarche suivante :

L'élaboration des diagrammes de cas d'utilisation. Cette étape a été réalisée suite à la spécification fonctionnelle de l'application. Recensement des classes candidates et élaboration du diagramme des classes. Dresser les diagrammes de collaboration et de séquences pour mettre en évidence les interactions entre les différents objets du système. Elaborer le diagramme d'états-transitions pour montrer les différents états l'interface.

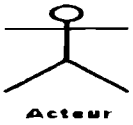
UML définit une notation graphique pour représenter les cas d'utilisation, cette notation est appelée diagramme de cas d'utilisation. UML ne définit pas de standard pour la forme écrite de ces cas d'utilisation, et en conséquence il est aisé de croire que cette notation graphique suffit à elle seule pour décrire la nature d'un cas d'utilisation. Dans les faits, une notation graphique peut seulement donner une vue générale simplifiée d'un cas ou d'un ensemble de cas d'utilisation [12].

4) Cas d'utilisations



Ils permettent de décrire l'interaction entre l'acteur et le système. L'idée forte est de dire que l'utilisateur d'un système logiciel a un objectif quand il utilise le système. Le cas d'utilisation est une description des interactions qui vont permettre à l'acteur d'atteindre son objectif en utilisant le système. Les cas d'utilisation sont représentés par une ellipse sous-titrée par le nom du cas d'utilisation (éventuellement le nom est placé dans l'ellipse). Un acteur et un cas d'utilisation sont mis en relation par une association représentée par une ligne. Le plus souvent, le diagramme des cas est établi par la maîtrise d'ouvrage (MOA) d'un projet lors de la rédaction du cahier des charges afin de transmettre les besoins des utilisateurs et les fonctionnalités attendues associées à la maîtrise d'œuvre (MOE).

5) Acteurs



Ils sont des entités externes qui interagissent avec le système, comme une personne humaine ou un robot. Une même personne (ou robot) peut être plusieurs acteurs pour un système. c'est pourquoi les acteurs doivent surtout être décrits par leur rôle, ce rôle décrit les besoins et les capacités de l'acteur. Un acteur agit sur le système. L'activité du système a pour objectif de satisfaire les besoins de l'acteur. Les acteurs sont représentés par un pictogramme humanoïde (stick man) sous-titré par le nom de l'acteur.

6) Relations

Trois types de relations sont prises en charge par la norme UML et sont graphiquement représentées par des types particuliers de ces relations. Les relations indiquent que le cas d'utilisation source présente les mêmes conditions d'exécution que le cas issu. Une relation simple entre un acteur et une utilisation est un trait simple.

7) Inclusions

Dans ce type d'interaction, le premier cas d'utilisation inclut le second et son issue dépend souvent de la résolution du second. Ce type de description est utile pour extraire un ensemble de sous-comportements communs à plusieurs tâches, comme une macro en programmation. Elle est représentée par une flèche en pointillé et le terme include.

8) Extensions

Les extensions (extend) représentent des prolongements logiques de certaines tâches sous certaines conditions. Autrement dit 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. Elle est représentée par une flèche en pointillée avec le terme extend. Ce type de relation peut être utile pour traiter des cas particuliers ou fonctions optionnelles, préciser les objectifs, ou encore pour tenir compte de nouvelles exigences au cours de la maintenance du système et de son évolution [21].

IV.3 Cycles de développement des applications informatiques

Ces modèles définissent les étapes du développent d'un logiciel. A chaque étape, des livrables sont produits. On ne passe à l'étape suivante que si les résultats sont jugés satisfaisants (MONDJO, 2016). Les modèles de cycles de vie d'un logiciel sont entre autres modèle en cascade, modèle en V, modèle en spirale, modèle incrémental, modèle orienté réutilisation ainsi que le prototypage. Dans la suite, nous présentons deux de ces modèles. Signalons que dans la réalisation de notre travail, nous avons opté pour le modèle en V [13].

1) Modèle en cascade

Le cycle de vie en cascade présente le développement de logiciel comme une suite de phases qui s'enchainent dans un déroulement linéaire, depuis l'analyse des besoins jusqu'à la présentation du cours.

La figure suivante nous montre le modèle de cycle de vie en cascade (AUDIBERT, 2009) :

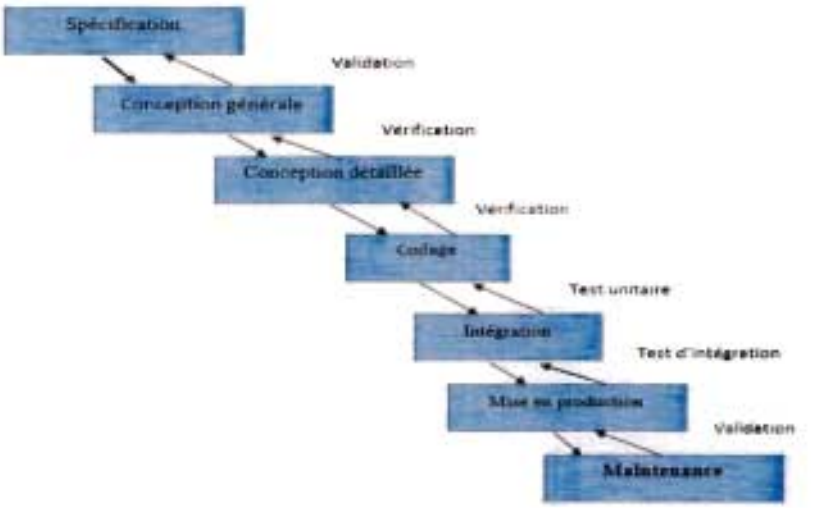


Figure 4: Modèle en cascade

2) Modèle en V

Le modèle en V demeure actuellement le cycle de vie le plus connu et certainement le plus utilisé. Il s'agit d'un modèle en cascade dans lequel le développement des tests et du logiciel sont effectués de manière synchrone (Muller, 1997).

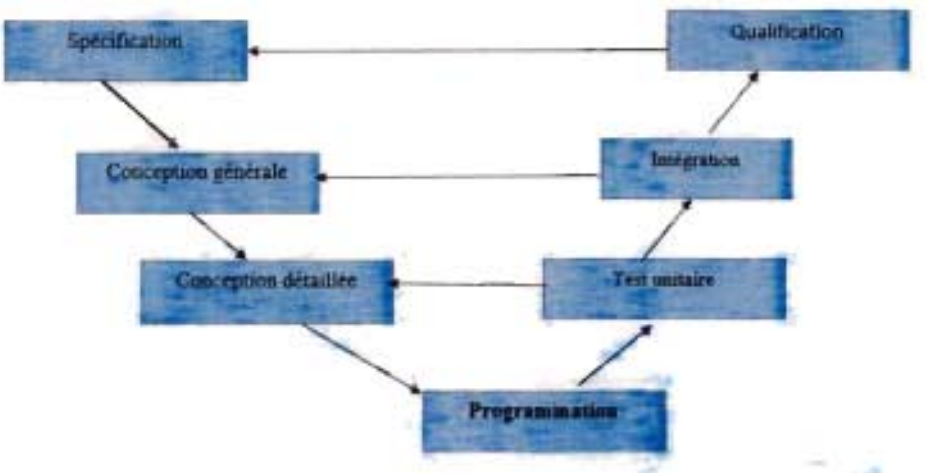


Figure 5: Modèle en v

Le principe de ce modèle est qu'avec toute décomposition doit être décrite la recomposition et que toute description d'un composant est accompagnée de tests qui permettront de s'assurer qu'il correspond à sa description (AUDIBERT, 2009). Ceci rend explicite la préparation des dernières phases (validation-vérification) par les premières - **construction du logiciel, et permet d'éviter d'énoncer une propriété qu'il est im**possible de vérifier objectivement après la réalisation.

IV.4 Couche Modèle

La plupart des plateformes de développement des applications web y compris JEE n'imposent aucun rangement particulier du code. Le problème c'est que si on développe n'importe comment le code va être mal organisé et il devient très vite difficile de retrouver un bout de code ou une fonction qu'on veut modifier. Pour éviter ce problème, les développeurs ont aujourd'hui tendance à utiliser les bonnes pratiques de développement qu'on appelle patrons de conception ou design pattern (en anglais). La réalisation de notre travail va intégrer le design pattern MVC qui est un modèle architectural de conception, une bonne pratique qui décrit comment le code d'une application doit être organisé. Les principaux avantages d'un tel pattern sont évidents :

1) La clarté introduite par un découpage clair et surtout standard des différentes sections d'une application permet une maintenance du code bien plus aisée que si le code ne respectait aucune règle préétablie. C'est cette quasi-universalité du mode de développement qui confère à MVC son intérêt le plus conséquent ;
2) Le découpage, et donc l'isolement des différentes tâches au sein d'une application, permet une meilleure répartition du travail entre les différents profils de développeurs.
3) Permet d'enregistrer les données, de les récupérer, de les lister, de les supprimer, et de les mettre à jour.

Le développement avec le modèle MVC segmente le code en trois parties ou couches, chaque couche ayant une fonction bien précise. Le modèle décrit le comportement de l'application. Il permet de traiter les données et de gérer les interactions avec la base de données. Il met à disposition les méthodes de mise à jour de ces données telles que la création, la suppression, la mise à jour ou la consultation. La vue correspond à l'interface graphique avec laquelle l'utilisateur interagit. Le contrôleur a comme rôle de récupérer les requêtes de l'utilisateur, de les transmettre au modèle, puis de rediriger vers la vue adéquate. (Christelle, 2017). La figure 25 suivante représente des interactions entre le modèle, la vue et le contrôleur dans le cas d'une application web (ATLASSI, 2016)

Couche Vue

La vue correspond à l'interface avec laquelle l'utilisateur interagit. Sa première tâche est de présenter les résultats renvoyés par le modèle. Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (clic de souris, sélection d'une entrée, boutons, etc.). Ces différents événements sont envoyés au contrôleur, elle se contente d'afficher les résultats des traitements effectués par le modèle et d'interagir avec l'utilisateur. Il représente la (ou une) représentation des données du modèle ; il assure la consistance entre la représentation qu'elle donne et l'état du modèle/le contexte de l'application. Cette partie se concentre sur l'affichage. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples. pour afficher par exemple la liste des messages des forums.

Couche Contrôleur

Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données. le contrôleur demande la modification des données au modèle. et ce dernier notifie la vue que les données ont changé pour qu'elle se mette à jour.

Cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le modèle et la vue : le contrôleur va demander au modèle les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue. Le contrôleur contient exclusivement du PHP. C'est notamment lui qui détermine si le visiteur a le droit de voir la page ou non (gestion des droits d'accès).

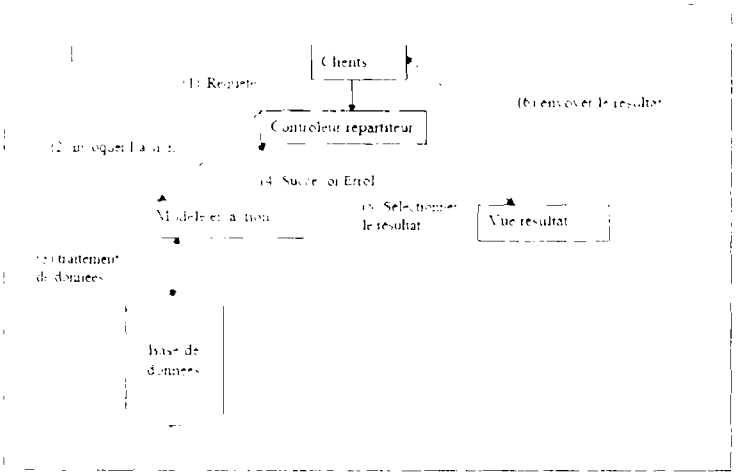


Figure 6 : Architecture MVC

IV.5 Diagramme des cas d'utilisation

Les cas d'utilisation ont été définis initialement par Ivar Jacobson en 1992 dans sa méthode OOSE. Les cas d'utilisation constituent un moyen de recueillir et de décrire les besoins des acteurs du système. Un cas d'utilisation permet de décrire l'interaction entre les acteurs (utilisateurs du cas) et le système. La description de l'interaction est réalisée suivant le point de vue de l'utilisateur. La représentation d'un cas d'utilisation met en jeu trois concepts : l'acteur, le cas d'utilisation et l'interaction entre l'acteur et le cas d'utilisation.

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié. Un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages éventuellement porteurs de données. Une même personne physique peut se comporter en autant d'acteurs différents que le nombre de rôles qu'elle joue vis-à-vis du système.

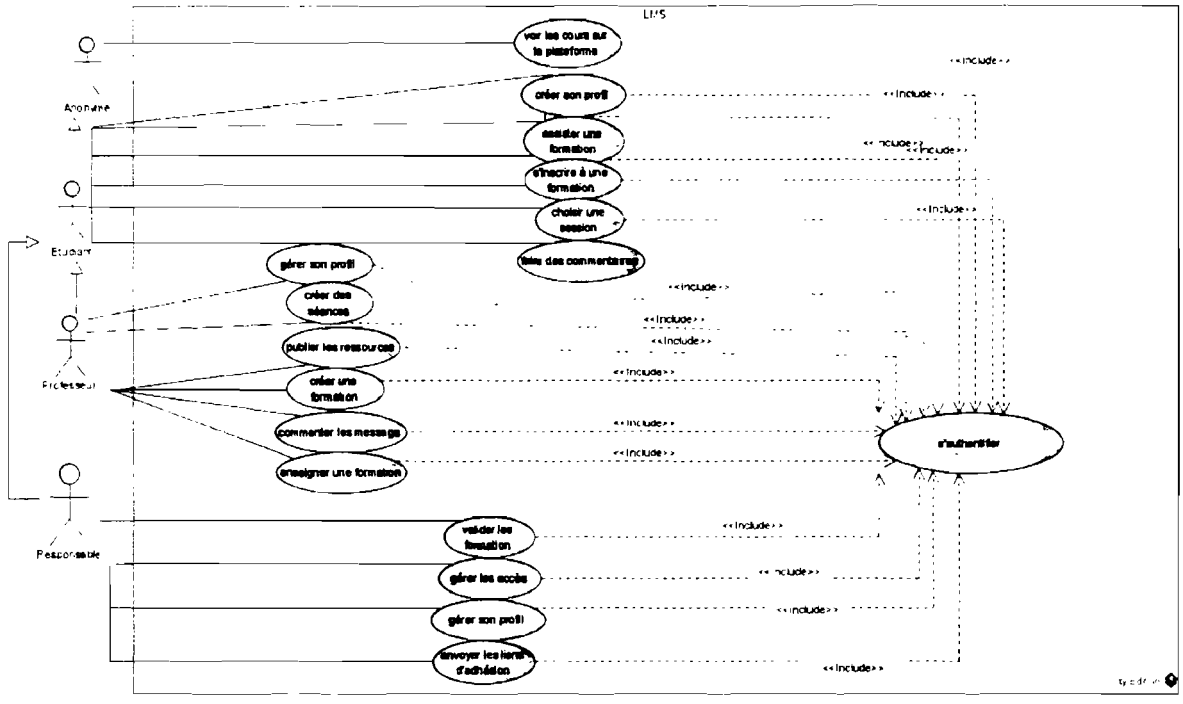


Figure 7 : Diagramme de cas d'utilisation du système

IV.6 Description des principaux cas d'utilisation

Dans le but de mieux comprendre le système et les interactions avec les utilisateurs, dans cette partie on va détailler les scenarios de principaux cas d'utilisation.

1. Cas d'utilisation « S'authentifier »

Sommaire d'identification

Titre : S'authentifier

Objectif : Les utilisateurs s'authentifient avant d'accéder au système ;

Résumé : L'utilisateur lance l'application ; la page d'authentification s'ouvre ; il saisit son nom d'utilisateur et son mot de passe et valide en cliquant sur le bouton « se connecter ».

Tableau 1:Description de cas d'utilisations « S'authentifier »

Acteurs concernés	Etudiants, professeur et administrateur
Pré conditions	1. l'utilisateur possède un compte
Post conditions	1. .Le cas démarre après le point 2 de l'enchainement nominal, l'utilisateur s'authentifie.

Scénario nominal	1. Le système invite l'acteur à entrer son login et son mot de passe. 2. L'acteur saisit le login et le mot de passe et choisit son profil. 3. Le système vérifie les paramètres. 4. Le système ouvre l'espace de travail correspondant au profil.
Scénario alternatif	Le login ou le mot de passe est incorrect : ce scénario commence au point 3 du scénario nominal. Le système informe l'acteur que les données saisies sont erronées et le scénario reprend au point 01 du scénario nominal.

2. Cas d'utilisation « inscription à une formation »

Sommaire d'identification

Titre : Gérer les profils d'inscription

Objectif : Pouvoir ajouter sur le panier les étudiants inscrit sur la plateforme, valider leurs inscriptions et enfin visualiser les étudiants déjà inscrit.

Résumé : permet à l'étudiant de remplir un formulaire d'inscription à la formation « Valider l'inscription »

Tableau 2: Description de cas d'utilisations « Gérer les profils des étudiants »

Acteurs concernés	Etudiant
Pré conditions	1. l'étudiant est authentifié
Post conditions	2. le cas démarre après le point 2 de l'enchainement nominal, l'étudiant s'inscrit à une formation
Scénario nominal	3. Le système affiche la liste des formations disponibles. 4. L'acteur choisi une formation. 5. Le système affiche l'interface du payement 6. Le système valide l'inscription à la formation 7. Le système ajoute la formation au profil de l'étudiant 8. Le système notifie l'étudiant du bon déroulement de l'opération
Scénario alternatif	Le payement n'a pas pu être effectué ou a été annulé : ce scénario commence au point 04 du scénario nominal. Le système informe l'acteur que l'inscription à la formation a échoué et le scénario reprend au point 01 du scénario nominal.

3. Cas d'utilisation « créer la formation »

Titre : créer la formation

Objectif : Pouvoir ajouter sur le panier les séances inscrit sur la plateforme, valider et enfin visualiser la séance déjà disponible.

Résumé : permet au formateur de créer sa propre formation

Tableau 3: Description de cas d'utilisations « créer les formations »

Acteurs concernés	Professeur
Pré conditions	1. Le professeur est authentifié
Post conditions	2. le formateur veut ajouter une nouvelle formation
Scénario nominal	3. Le système affiche un formulaire permet de créer une nouvelle formation. 4. Le professeur saisie les coordonnées relatives à la formation. 5. Le système vérifie la disponibilité du salon de formation. 6. Le système affiche l'interface du payement 7. Le système valide la création de la formation 8. Le système ajout la formation au profil du formateur. 9. Le système notifie le formateur.
Scénario alternatif	Indisponibilité de la séance à la date prévue par le professeur : ce scénario commence au point 03 du scénario nominal. Le système informe le professeur que la séance est indisponible et le scénario reprend au point 2 du scénario nominal.

4. Cas d'utilisation « Enseigner une Formation »

Titre : Enseigner la formation

Objectif : Ajouter sur le panier les séances inscrit sur la plateforme, valider et enfin visualiser la séance déjà disponible.

Résumé : permet au formateur de dérouler sa formation.

Tableau 4: Description de cas d'utilisations « enseigner les formations »

Acteurs concernés	Professeur
Pré conditions	1. Le professeur est authentifié
Post conditions	2. Le professeur enseigne une séance
Scénario nominal	3. Le système affiche l'interface du salon de formation 4. Le système affiche les paramètres relatifs au camera et au microphone du professeur 5. Le professeur autorise l'accès à son camera et son microphone.

5. Cas d'utilisation « assister à une Formation »

Titre : Assister la formation

Objectif : voir sur le panier la liste des séances inscrit sur la plateforme, valider et enfin visualiser la séance déjà disponible.

Résumé permet à l'étudiant d'assister à une formation.

Tableau 5:Description de cas d'utilisations « Assister les formations »

Acteurs concernés	Etudiant
Pré conditions	1. l'étudiant est inscrit à une formation
Post conditions	2. l'étudiant assiste à la formation
Scénario nominal	3. Le système affiche l'interface du salon de formation 4. Le système affiche les paramètres relatifs au camera et au microphone du formateur 5. L'étudiant peut autoriser l'accès à sa caméra et son microphone.

IV.7 Diagramme de classe

Le diagramme de classe représente la structure des entités manipulées par les utilisateurs (classes, interfaces, collaborations) et permet de donner la représentation statique du système à développer. La description du diagramme de classe est fondée sur : le concept d'objet, le concept de classe comprenant les attributs et les opérations, les différents types d'association entre classes. Une classe est représentée par un rectangle séparé en trois parties :

Tableau 6::Description d'une classe

Nom class
Attribut1 Attribut2 .....
Opération1() Opération2() ...

Le diagramme de classe représente la structure des entités manipulées par les utilisateurs (classes, interfaces, collaborations).

Permet de donner la représentation statique du système à développer La description du diagramme de classe est fondée sur : le concept d'objet, le concept de classe comprenant les attributs et les opérations, les différents types d'association entre classes.

Le diagramme de classes est le point central dans un développement orienté objet.

En analyse, il a pour objectif de décrire la structure des entités manipulées par les utilisateurs. Il a pour but de représenter les ressources informationnelles pour les processus orientés principalement vers la gestion de données.

En conception, le diagramme de classes représente la structure d'un code orienté objet ou, à un niveau de détail plus important, les modules du langage de développement.

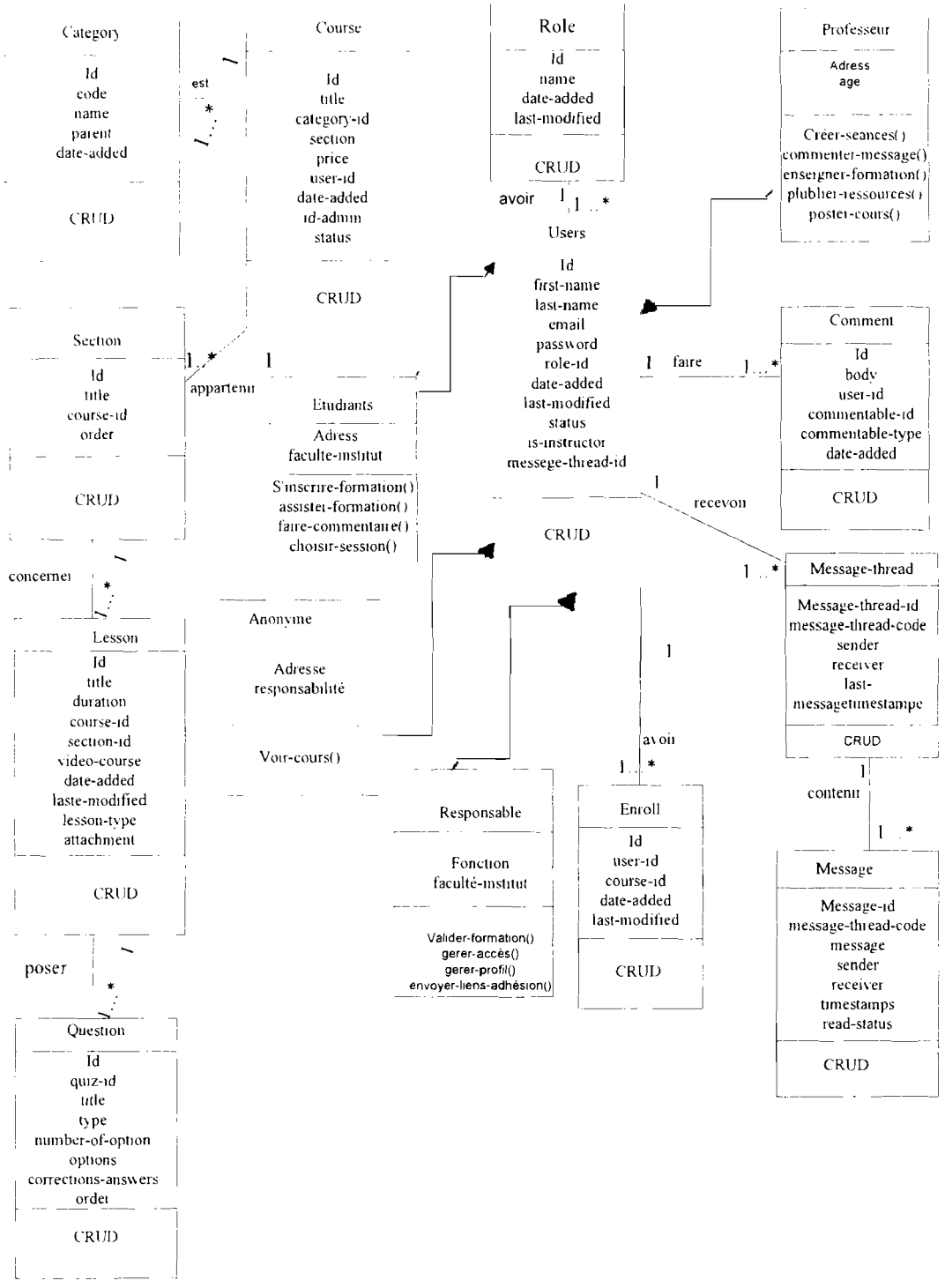


Figure 8: Diagramme de classes de notre système

IV.8 Diagrammes de séquences

Les diagrammes de séquences montrent les interactions à mettre en œuvre entre les classes pour réaliser un résultat, tel qu'un cas d'utilisation. UML étant conçu pour la programmation orientée objet, ces communications entre les classes sont reconnues comme des messages. Le diagramme de séquence énumère des objets horizontalement, et le temps verticalement. Il modélise l'exécution des différents messages en fonction du temps [16].

Plusieurs types de messages (actions) peuvent transiter entre les acteurs et objets.

Message simple : le message n'a pas de spécificité particulière d'envoi et de réception.

Message avec durée de vie : l'expéditeur attend une réponse du récepteur pendant un certain temps et reprend ses activités si aucune réponse n'a lieu dans un délai prévu.

Message synchrone : l'expéditeur est bloqué jusqu'au signal de prise en compte par le destinataire. Les messages synchrones sont symbolisés par des flèches barrées.

Message asynchrone : le message est envoyé, l'expéditeur continue son activité que le message soit parvenu ou pris en compte ou non. Les messages asynchrones sont symbolisés par des demi-flèches.

Message dérobant : le message est mis en attente sur une liste d'attente de traitement chez le récepteur.

1) S'authentifier au système

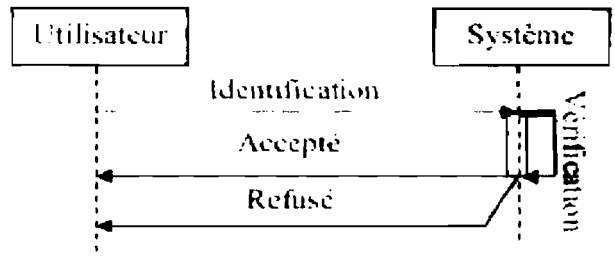


Figure 9: Diagramme de séquence pour l'authentification

2) Poster un cours

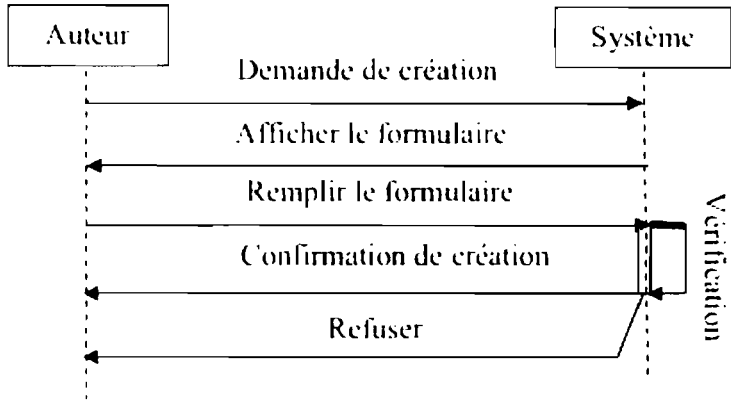


Figure 10 : Diagramme de séquence pour poster un cours

3) Créer les formations et sessions

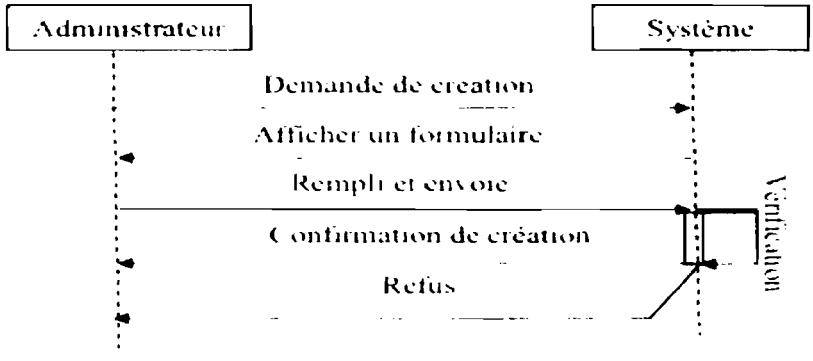


Figure 11 : Diagramme de séquence pour la création des formations et sessions

IV.9 Diagrammes d'activités

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallèles. Le diagramme d'activité est également utilisé pour décrire un flux de travail. Un diagramme d'activité permet de modéliser un processus interactif, global ou partiel pour un système donné (logiciel, système d'information). Il est recommandable pour exprimer une dimension temporelle sur une partie du modèle, à partir des diagrammes de classes ou de cas d'utilisation. Le diagramme d'activité est une représentation proche de l'organigramme ; la description d'un cas d'utilisation par un diagramme d'activité correspond à sa traduction algorithmique. Une activité est l'exécution d'une partie du cas d'utilisation, elle est représentée par un rectangle aux bords arrondis. Le diagramme d'activité est

sémantiquement proche des diagrammes de communication ou d'état-transitions, ces derniers offrant une vision microscopique des objets du système [17].

Les concepts communs ou très proches entre le diagramme d'activité et le diagramme d'état-transition sont : transition, nœud initial (état initial), nœud final (état final), ⊗ nœud de fin flot (état de sortie), ◊ nœud de décision (choix).
Contexte d'utilisation : Description du comportement interne : d'une classe, d'une méthode et d'un cas d'utilisation

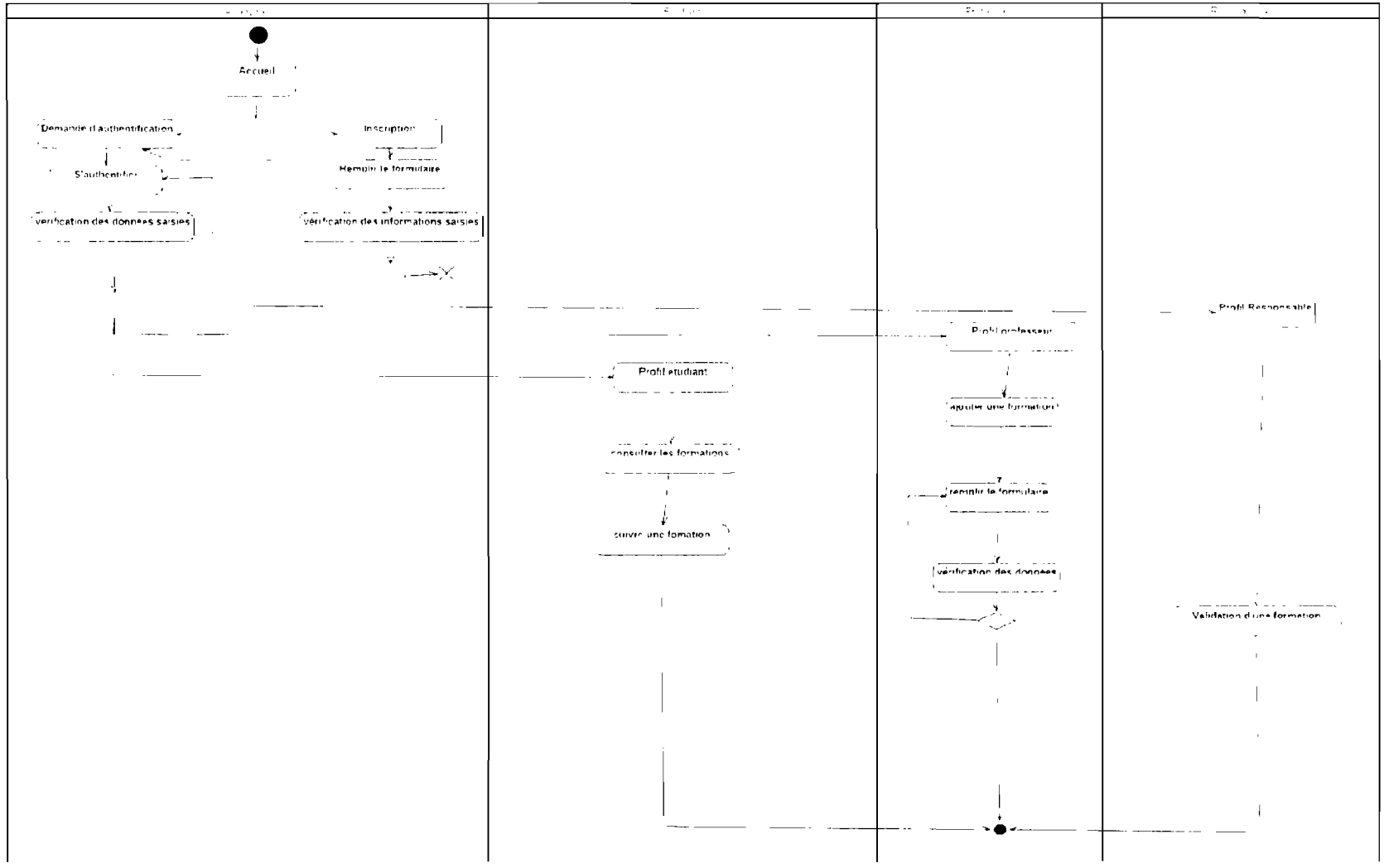


Figure 12:Diagramme d'activité du système

IV.10 Diagramme de déploiement

Le diagramme de déploiement correspond à la fois à la structure du réseau informatique qui prend en charge le système logiciel, et la façon dont les composants d'exploitation y sont installés [15]. Le diagramme de déploiement permet de représenter l'architecture physique supportant l'exploitation du système. Cette architecture comprend des nœuds correspondant aux supports physiques (serveurs, routeurs...) ainsi que la répartition des artefacts logiciels (bibliothèques, exécutables...) sur ces nœuds. C'est un véritable réseau constitué de nœuds et de connexions entre ces nœuds qui modélisent cette architecture. Un nœud correspond à une ressource matérielle de traitement sur laquelle des artefacts seront mis en œuvre pour l'exploitation du système. Les nœuds peuvent être interconnectés pour former un réseau d'éléments physiques.

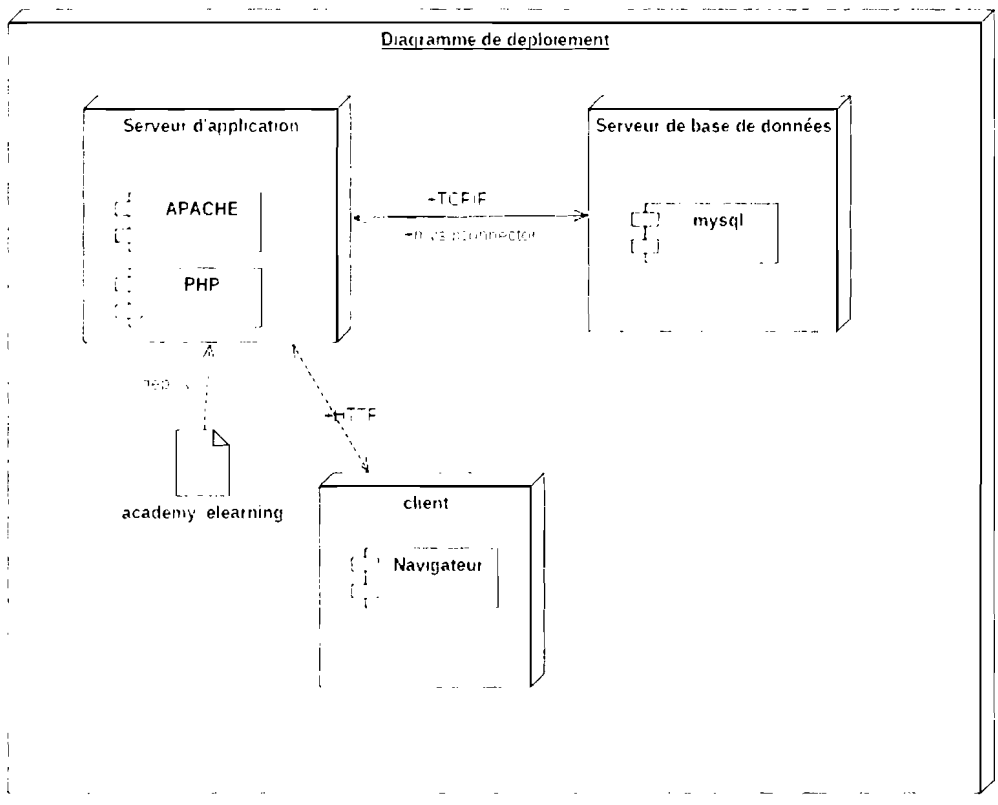


Figure 123 : Diagramme de déploiement de notre système

Conclusion

Ce chapitre a porté sur l'analyse et la conception du système d'information. Grâce au langage de modélisation UML, la mise en place de différents diagrammes a permis de comprendre petit à petit le système informatique à mettre en place. Il s'agit du diagramme des cas d'utilisation qui représente les actions réalisées par les acteurs, du diagramme de classes qui représente les différentes entités manipulées par les acteurs et du diagramme de déploiement qui décrit la disposition physique des différents matériels qui entrent dans la composition du système et la répartition des programmes exécutables sur ces matériels.

CHAPITRE V : REALISATION ET DEVELOPPEMENT D'UNE PLATEFORME D'ENSEIGNEMENT A DISTANCE

V.1 Introduction

Après l'analyse approfondie et la conception du nouveau système, il était temps de passer à sa réalisation à l'aide des outils adéquats.

Cette partie a pour objectif majeur de réaliser et de présenter le produit final de notre application. C'est un point qui est composé par trois petites parties : la première partie présente les ressources matérielles, logicielles et langages utilisés pendant la réalisation de ce projet, la seconde partie concerne l'implémentation des données et des traitements et enfin la troisième partie consiste à présenter brièvement l'utilisation et la manipulation du logiciel à travers quelques interfaces [22].

V.2 Etude technique

L'étude technique est une phase d'adaptation de conception à l'architecture technique. Elle a pour objectif de décrire au plan fonctionnel la solution à réaliser d'une manière détaillée ainsi que la description des traitements. Cette étude, qui suit l'étude détaillée, constitue le complément de spécification informatique nécessaire pour assurer la réalisation du futur système. Cette étude permet également de déterminer [05] :

La structure informatique de la base de données ; l'architecture des programmes ; la structure de chaque programme et l'accès aux données.

Environnement de réalisation

Pour réaliser notre application, nous avons fait recours aux différents outils matériels et logiciels : Matériel de base : l'ordinateur de marque HP, Windows 10 professionnel, processeur : Intel(R) Core (T M) i5-2410M CPU @ 2.30GHZ, RAM de 8 Go, Système d'exploitation de 64 bits.

Choix de langages de développement et SGBD

Le langage de programmation de script côté serveur est **PHP**. Il permet, de décrire dans une page web, un affichage dynamique d'information, c'est-à- dire que le texte affiché peut dépendre de variables. Les instructions PHP sont généralement

contenues dans des fichiers d'extension PHP. Ces fichiers peuvent contenir du HTML. entremêlé avec le code PHP. Quand un navigateur demande un tel fichier. le serveur Apache exécute les instructions PHP. qui produisent une page HTML. Une fois la page HTML générée. le serveur la renvoie au navigateur, qui ne voit qu'une page HTML.

Pour gérer les données. nous avons choisi MySQL qui est un système de gestion de base de données. Son rôle est de stocker les données. sous forme de tables. et de permettre la manipulation de ces données à travers le langage de requête SQL.
PHP est capable de passer à MySQL de telles requêtes à travers les fonctions de préfixe MySQL. PHP possède également des fonctions pour dialoguer avec d'autres systèmes de gestion de base de données. C'est une raison du succès du couple PHP + MySQL dans la mise en place des applications web [16].

JavaScript un langage de script utilisé côté client. Son rôle principal est de manipuler les données contenues dans les pages web, comme par exemple les animations et autres effets dû à certains événements. Dans notre projet, ce langage a été utilisé pour réaliser certaines animations, la gestion de la taille des écrans pour l'affichage et pour gérer d'autres types d'effets [29].

Cascading Style Sheets (CSS) : est un langage de style permettant d'organiser les informations contenues dans les pages HTML ou des documents XML. Ce langage CSS permettant de préciser les caractéristiques visuelles et sonores de présentation d'une page Web a été utilisé pour organiser le contenu des pages de notre application [07].

Hypertext Markup (Language HTML), né en 1994, est le langage du Web par excellence. C'est lui qui indique au navigateur de quoi est constituée la page web qu'il est en train de consulter, tant au niveau de la langue de prononciation, de son titre et contenu, mais aussi et surtout de la structure et de la fonction de l'ensemble des éléments qui la composent : paragraphes, citations, titres, listes, liens hypertextes, images [17].

SQL (Structured Query Language, ou Langage de requêtes structuré en français) est un langage de définition de données (LDD, ou en anglais DDL Data Définition Language), un langage de manipulation de données (LMD, ou en anglais DML, Data Manipulation Language), et un langage de contrôle de données (LCD, ou en anglais DCL, Data Control Language), pour les bases de données relationnelles. Ce langage a été utilisé pour gérer les données stockées dans la base de données de notre application à l'aide de ses requêtes [32].

StarUML et Visual Paradigm sont des outils de modélisation logicielle qui prennent en charge UML. Ils fournissent différents types de diagrammes. StarUML accepte la notation UML 2.0. Il supporte activement l'approche MDA (Model Driven Architecture) en supportant le concept de profil UML et en permettant de générer du code pour plusieurs langues. Nous avons utilisé ces outils pour modéliser notre application.

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 (X Apache 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 fonctionne sur les systèmes d'exploitation les plus répandus. Ce logiciel nous a été très utile pour ce projet parce qu'il contient plus d'outils qui ont été utilisé pour concevoir la base de données, pour simuler notre application sur le serveur local, etc. Il nous a permis de mettre en place un serveur local qui nous a aidés à développer tout en testant notre application.
Pour le serveur web nous avons utilisé Apache. Son rôle est d'écouter les requêtes émises par les navigateurs (qui demandent des pages web), de chercher la page demandée, de la générer et de la renvoyer au client [33].

Microsoft Office Word 2016 est un logiciel de traitement de texte couvre deux notions, assez différentes en pratique : Un éditeur de textes interactif et un compilateur pour un langage de mise en forme de textes.

Microsoft Office Power Point 2016 est un créateur de présentations (succession de diapositives). Il est utilisé pour créer des présentations avec du texte, avec des images, sons, vidéos et autres objets.

Pour éditer les pages lors du développement de cette application, nous avons utilisé un environnement de travail appelé Notepad++.

Bootstrap est un framework qui peut rendre plus facile pour créer l'architecture d'une page web. Mais Bootstrap va bien plus loin qu'offrir du code CSS déjà bien organisé et structuré. Il offre aussi des plugins jQuery de qualité pour enrichir les pages web.

Il existe des frameworks côté serveur (désignés *backend* en anglais), et d'autres côté client (désignés *frontend* en anglais). Bootstrap fait partie de cette deuxième catégorie.

Les frameworks CSS sont spécialisés, comme leur nom l'indique, dans les CSS.

C'est-à-dire qu'ils nous aident à mettre en forme les pages web : organisation, aspect, animation.

V.3 Descriptif d'une plateforme d'enseignement à distance

Ce projet est composé de trois parties principales :

Partie Gestion des formations :

1) Permet la gestion des utilisateurs (création des comptes pour les formateurs et les étudiants).
2) Permet l'organisation des formations : donne la possibilité à un formateur de créer sa propre formation ainsi qu'elle offre la possibilité aux étudiants de s'inscrire aux différentes formations proposées.

Partie médiathèque : c'est la partie permettant la gestion des fichiers dont :

1) Uploader un fichier
2) Télécharger un fichier

Partie Web Conférence : cette partie représente le salon de formation qui offre les fonctionnalités suivantes :

1) Activer audio/vidéo.
2) Discussion avec les formateurs et les étudiants [29].
3) Pour la partie Web Conférence, on a eu recours aux technologies web RTC qui est un système de vidéo conférence gratuit présenté par Google. Il permet à ces utilisateurs de discuter simultanément en mode audio et vidéo. En plus, du point de vue matériel, « Web RTC » est également accessible via smartphone et tablette qui correspond beaucoup au besoin de ce projet.

Pour la partie Web Conférence, on a eu recours aux technologies web RTC qui est un système de vidéo conférence gratuit présenté par Google. Il permet à ces utilisateurs de discuter simultanément en mode audio et vidéo. En plus, du point de vue matériel, « Web RTC » est également accessible via smartphone et tablette qui correspond beaucoup au besoin de ce projet.

V.4 Architecture

L'architecture de cette application est de type client-serveur, où un ordinateur interagit avec d'autres sur Internet. Comme on a mentionné précédemment, notre application est composée de trois parties :

1) La partie pour la gestion des formations et des étudiants développés en PHP/HTML/CSS/JavaScript.
2) De même pour la partie gestion des fichiers.
3) La partie pour le salon de formation qui est une adaptation du système développé en PHP/HTML/CSS/Javascrit

Nous déduisons qu'on a besoins :

1) D'un serveur d'application(ApacheTomcat) : utilisé comme un serveur d'application pour la partie gestion des formations et des utilisateurs.
2) D'un serveur Google : qui se charge de diffuser l'ensemble des informations Multimédia (vidéo, micro. ...) à l'ensemble des participants.
3) D'un serveur base de données MySQL : se charge au stockage des données.
4) D'un client léger : un navigateur web (Firefox, Google Chrome, ...).

Dans la réalisation du projet, nous avons opté pour une architecture MVC afin de garantir une assurance de la maintenabilité, la modularité de l'application et la rapidité de développement.

MVC littéralement Modèle Vue Contrôleur est une architecture qui organise l'interface Homme-Machine d'une manière à ce que le développement puisse se faire en couches indépendantes. Il impose la séparation entre les données, la présentation et les traitements, ce qui donne trois parties fondamentales dans l'application finale : le modèle de données, le contrôleur et la vue [35].

V.5 Fonctionnement du nouveau système

Dans cette partie, nous présentons les différentes captures d'écran des interfaces graphiques de notre système d'E-learning Management System.

En saisissant l'adresse de la page d'accueil dans la barre d'adresse de votre navigateur https://217d-154-117-216-220.ngrok.io/academy_Herve/, vous affichez la page d'accueil de la plate-forme, en cliquant sur le lien « Inscription ». Un formulaire simple et apparu du login et du mot de passe personnel qui serviront à vous connecter sur la plateforme lors de notre prochaine visite. Lors de cette phase d'inscription, veillez à cocher la case « Créer des formations », en tant que créateur des formations (formateur), ou cocher la case « m'inscrire à des formations », en tant que suivant des formations (apprenant).

Le formateur doit être inscrit sur la plate-forme puis l'administrateur crée la formation et associé le formateur souhaité. La figure ci-dessous montre que la formation a été bien crée avec une indication de son formateur.

Après la création d'une formation, le formateur peut créer le contenue de cours tests, et les autres ressources (documents, liens, glossaires...etc.).

Depuis notre espace de formation, un clic suffit pour accéder à la page d'accueil d'une formation en tant que formateur-administrateur.

Depuis notre espace de formation, un clic suffit pour accéder à la page d'accueil d'une formation en tant qu'apprenant.

Un cours d'apprentissage est une séquence d'apprentissage découpée en chapitres eux-mêmes découpés en étapes. Il peut être organisé en fonction d'un contenu, il constituera alors une sorte de table des matières, ou bien en fonction d'activités.

V.5.1 Interface d'accueil

Après les phases d'étude de l'existant, la conception et la modélisation fonctionnelle et organisationnelle on a développé les interfaces de l'application. La page accueil présente la page d'accueil de l'application, à partir de cette interface, si l'internaute est un nouvel utilisateur, le site lui propose de rejoindre la communauté, donc de créer son compte. Les interfaces du site sont présentées sous forme de lien. Elles présentent quatre liens

principaux se différenciant selon l'utilisateur qu'il soit connecté ou non. Lorsque l'utilisateur n'est pas connecté : Accueil : qui amène l'utilisateur à la page d'accueil du site. Inscription : qui amène l'utilisateur vers la page d'inscription. Connexion : qui amène l'utilisateur sur la page de connexion du site. Espace admin : c'est un espace spécifique à l'administrateur. Lorsque l'utilisateur est connecté : Accueil : lien vers la page d'accueil du site. Liste des utilisateurs : comporte deux sous-menus : Liste des étudiants : permet de lister tous les étudiants inscrits au site. Liste des formateurs : permet de lister les formateurs. Apprendre : composé de deux sous-menus :

Médiathèques : lien qui amène à la médiathèque du site. Formations : permet de lister toutes les formations proposées par les formateurs. Contacts : qui amène l'utilisateur à la page de communication avec l'administrateur du site.



Figure 14: Interface d'accueil

V.5.2 Interface d'authentification

A travers de cette interface tout utilisateur ayant accès au système doit fournir son nom d'utilisateur ainsi que son mot de passe. Après vérification, le système lui affiche la page d'accueil de notre système ainsi que l'ensemble des fonctionnalités dont il a droit.

C'est la première fenêtre qui s'affiche si on exécute l'application. Toute personne qui veut bénéficier des services de l'application doit d'abord s'authentifier. Après authentification, une fenêtre principale s'affiche selon l'individu authentifié. Voici la fenêtre ci- dessous pour "Authentification".



V.5.3 Interface pour l'administrateur

Cette interface est contrôlée par l'administrateur, ses responsabilités sont en autres gérer les cours, les étudiants, les instructeurs et configurer son compte.

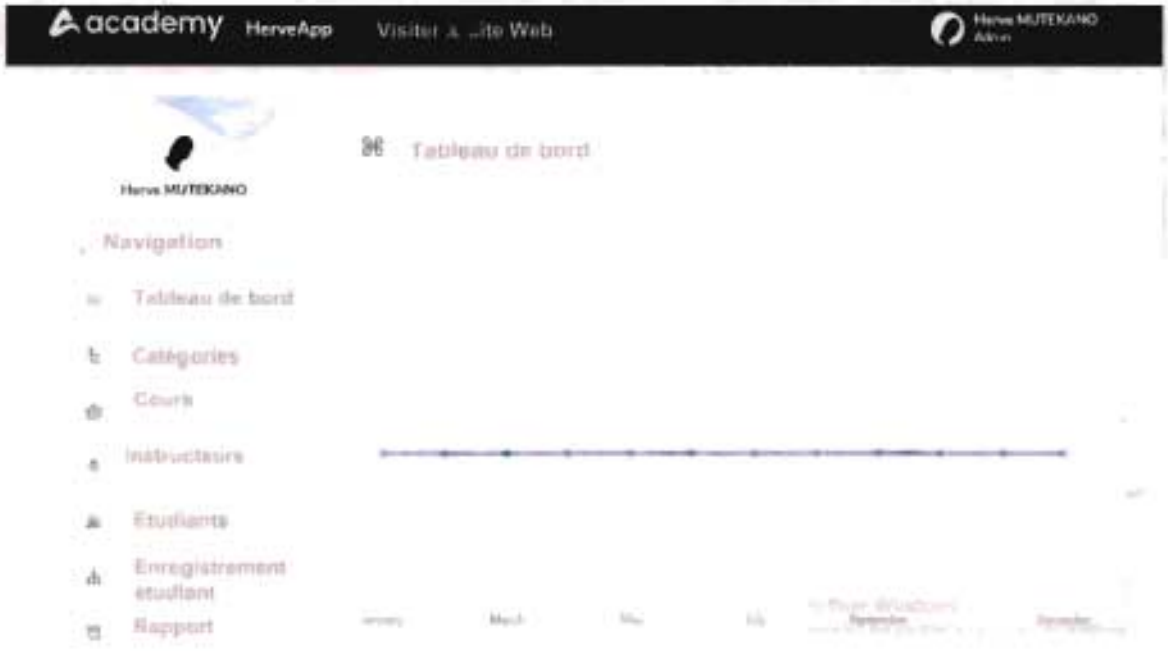


Figure 14: Interface de l'administrateur

V.5.4 Interface pour l'instructeur

Ce formulaire permet à l'instructeur de poster les cours et gérer son profile.



Figure 15: **Interface de l'instructeur**

V.5.5 Interface du salon de formation de l'instructeur

La figure ci-dessous représente l'interface du salon de formation après authentification de l'instructeur (respectivement étudiant). Pour animer la formation il faut que l'instructeur active son camera et son microphone. De même, pour assister à la formation, l'étudiant doit activer son camera et son microphone et peut discuter avec le formateur et ses collègues.



Figure 16:Interface salon coté instructeur

V.5.6 Interface du salon de formation coté étudiant

L'étudiant s'inscrit au rôle dans le cours, active son caméra et microphone pour suivre la formation et discuter avec son formateur.



Figure 19: Interface salon coté étudiant

V.5.7 Apports scientifiques et technologiques

L'utilisation des TIC est en ces jours devenue impérative dans la gestion des entreprises. De nombreuses institutions intègrent aujourd'hui l'utilisation des logiciels dans différents niveaux d'organisation. Ce qui est plus rationnel, vu les facilités qu'ils offrent. Les présentes recherches proposent de renforcer des fonctionnalités des logiciels utilisés dans les universités par le développement d'une plateforme d'enseignement permettant l'identification et l'organisation des enseignements au sein des universités en République du Burundi. Il permet d'amorcer une recherche qui pourra servir d'ébauche à des recherches futures.

Ce projet de fin d'études nous a permis de se familiariser avec un certain nombre de concepts tout en se basant sur les connaissances acquises au cours de notre formation.

Ainsi, le fait d'avoir travaillé avec mon directeur, des personnels de l'institut nous a mis sur la voie professionnelle. Ce projet de fin d'étude a été une occasion, pour développer et exercer nos capacités d'observation, d'analyse, de conception, de développement et de rédaction [34].

Ce projet de fin d'études nous a permis de :

1) Mettre en œuvre les notions et les connaissances acquises sur la « Méthodologie de conception des bases de données » ;
2) Améliorer mes connaissances théoriques concernant la communication client/serveur ;
3) Etudier les architectures multicouches et réaliser l'importance du travail sur ce type d'application dans le monde de programmation ;
4) Apprendre à gérer un projet.
5) Au niveau de la conception et du développement, cette étude m'a permis de :
6) Mener une conception orientée objet représentée avec le langage UML ;
7) Apprendre à maitriser la méthodologie de conception et de développement des applications clientes ;
8) Maitriser les étapes de développement et de conception d'une application Web.

Conclusion

Ce chapitre a présenté de façon détaillée le développement de la plateforme mise en place. Il est de ce fait partie de la présentation de l'environnement de développement et du langage de programmation utilisés jusqu'à l'illustration de quelques codes source. Pour terminer, il a été présenté dans le chapitre une documentation de la plateforme qui facilite la consommation des services offerts, ainsi qu'un espace d'échange entre le formateur et étudiants.

CONCLUSION GENERALE ET RECOMMANDATIONS

Conclusion générale

Les présentes recherches ont abouti à la mise en place d'une plateforme digitale permettant aux gestionnaires de l'institution d'identifier le besoin à grande échelle dans le domaine de l'enseignement sollicité.

Pour atteindre ce résultat, la méthodologie adoptée a reposé sur les techniques de l'observation, la documentation, l'entretien libre et l'enquête pour la collecte des données. Il a été nécessaire de comprendre le domaine d'étude, à savoir le système de l'enseignement présentiel, pour mettre en place un système informatique tenant compte des besoins de l'institution dans la gestion de leurs étudiants.

La modélisation, basée sur le langage UML, a donc porté sur l'identification des besoins des utilisateurs futurs du plateforme mise en place. Le langage UML a permis d'élaborer le diagramme de cas d'utilisation pour décrire les fonctionnalités du système tel que vues par les utilisateurs, le diagramme de classe pour représenter les objets manipulés par les utilisateurs et les relations entre eux ainsi que le diagramme de déploiement pour voir comment est-ce que le la plateforme a été mise en place.

L'implémentation, à partir du langage de programmation PHP, a été de produire une plateforme que l'UGL peut utiliser pour la formation et apprentissage de leurs étudiants.

La plate-forme réalisée permet aux formateurs de poster ses cours, ses tests et suivre les apprenants ; aux apprenants de s'inscrire à la plate-forme, lire ses cours, faire des tests, communiquer avec les autres apprenants de la plate-forme (chat, forum, messagerie interne/externe), comme il peut communiquer avec les formateurs via les classes virtuelles ; A l'administrateur de gérer la plate-forme (l'activation des comptes, le suivi et l'évaluation des autres utilisateurs…).

L'analyse de l'enquête s'est limitée, dans cette étude, à l'enseignement et apprentissage dans le secteur de l'éducation. Cette étude peut donc être étendue dans d'autres domaines proposés par l'UGL.

Recommandations

Après avoir implémenté l'E-learning Management System, nous adressons des recommandations à l'UGL ce qui suit :

1) Acquérir tous les matériels nécessaires pour le déploiement du système ;
2) Utiliser tous les aspects du système ;
3) Former tous les utilisateurs d'abord et ensuite à l'utilisation du système ;
4) Engager les moyens nécessaires pour l'information des autres services de l'institution.

A l'Université du Burundi de :

1) De soutenir les projets de recherche dans le domaine de l'éducation et assurer leurs financements ;
2) Appuyer les centres de recherches dans l'exécution des gros projets de développement national.

Le travail que nous avons réalisé peut-être amélioré et enrichi afin d'en faire un système comprenant d'autres fonctionnalités. Parmi les perspectives à prendre en compte pour améliorer le fonctionnement du système, nous citons notamment :

1) Intégration de la langue kirundi dans la plate-forme, le chronomètre au niveau des tests et les droits d'accès au système ;
2) Faire des captures vidéo via la webcam pendant le parcours des cours afin de permet au formateur de connaitre l'état de l'apprenant ;
3) Diagramme d'état de lieu de l'apprenant
4) Le partage de tout type de support de cours de formation (PPT, PDF, Word, Excel, ...) avec animation.
5) L'adaptation de l'application pour une utilisation de machines virtuelle d'une manière qui permet à un étudiant cloud e-learning et un formateur de travailler sur des machines virtuelles, c'est-à-dire avant que la formation débute l'administrateur doit créer au part avant des machines virtuelles équipées par des outils appartenant à la formation. Par exemple, dans le cas d'une formation sur l'outil Microsoft Office Word les machines virtuelles doivent être équipé par cet outil.
6) La mise en place d'un forum : un espace de discussion publique formé de catégories et de sous-forums permettant aux utilisateurs de discuter sur des sujets variés et classés par catégories.

REFERENCES BIBLIOGRAPHIQUES

Les ouvrages généraux

[01]Anne –Marie Husson. « Quel Modèle Qualité Pour la e-formation ? ». Année 2004.

[02]Atmani Hocine, Sefsafi Soumia. « Découverte et sélection des web services dans le cadre du e-Learning selon le profil utilisateur ». Mémoire de fin d'études. Année 2009.

[03] B. Kouninef, M. Djelti. S.M. Rerbal. « conception et réalisation d'une plateforme e-learning avec migration au e-learning », Institut Des télécommunications. Année 2006.

[04] Bernard Blandin. « Normes, standards, labels, chartes et démarches qualité pour la e-formation ». Centre Inffo. Année 2004.

[05] Michel Arnaud. « Problématique de la normalisation pour la formation en ligne ». Lyon. France. Année 2004.

[06] Sans auteur « Gérer les établissements d'enseignement ». Partenaires Microsoft Éducation. Année 2007.

[07] Pierre-Léonard Harvey. « les plates-formes d'apprentissage en ligne ». Université du Québec à Montréal. Année 2003. [25]. S.P.E. « La Formation dans les collectivités territoriales ». Secteur public. Année 2007.

[08] Medjek Faiza, Aouaouche El-Maouhab, Adjerad Radia and Ghehria Nahla. « Editeur SCORM en web services conforme à la norme IMS GWS « Instructional Management Systems General Web Services ». Centre de Recherche sur l'Information Scientifique et Technique (CERIST). Année 2004.

[09] Ail Mlouk Addi : Conception et réalisation d'une application de gestion intégrée au sein de la société Eone Group basée sur OpenERP. FSSM. Facultés des sciences Semlalia de Marrakech. Ingénierie des systèmes d'information. 2012- 2013.

[10] Moulay Hachemi Rahma Yasmina. Files d'Attentes et applications. Master. Universite Aboubakr Belkaid-Tlemcen Facultes des Sciences, 2014/2015,72p.

STRAUB, INES. « Le nombre d'internautes dans le monde atteint les 4 milliards ». @INESSTB. [En ligne]. (Février 2018).

[11] Etienne Anken. « Spécifications et standards e-learning », Institut SYSIN de l'EIVD. Année 2004. [11]. F.E.U.Q. « Avis sur la formation à distance », Fédération étudiante universitaire du Québec. Année 2009.

[12] Kazi Aouel Bassim et Rostane Zakaria : Suivie es enseignements du LMD par application de la méthode 2TUP. Université Abou Bekr Belkaid de Tlemcen, Faculté des Sciences de l'Ingénieur. Département d'Informatique. Projet de Fin d'Etudes pour l'obtention du Diplôme d'Ingénieur d'Etat en Informatique. Novembre 2007.

[13] Bontempi. G. : Modélisation et simulation. Département d'informatique, Boulevard de Triomphe CP-212, 54 p.

[14] François Larrey. « e-learning ses fondements et son utilisation dans le secteur bancaire ». Genève Place financière. Année 2004.

[15] Pierre-Henri Amalric. « Le B-A-BA de SCORM », France. Année 2007.

Les monographies et articles

[1] Bernard Blandin. « L'histoire de l'enseignement à distance et de la FOAD ». Centre Inffo. Année 2004.
[2] P Fulgence NAHAYO, « Modèles et Systèmes d'Aide à la Décision ». Faculté des Sciences et de l'Ingénieur(FSI). Département de TIC. Cours. 06 Décembre 2020
[3] Archive de l'UGL

[4] Dr William SAHINGUVU, « Streaming Technology ». Faculté des Sciences et de l'Ingénieur(FSI), Département de TIC, Cours, 14 Mars 2021.
[5] NTAYAGABIRI Jean Pierre, « Conception et développement d'un système automatisé de gestion d'un circuit de distribution des produits agro-pastoraux au Burundi : « Cas de la Coopérative de Traitement de Miel « COTRA-Miel » » », Faculté des Sciences et de l'Ingénieur(FSI). Département de TIC. mémoire, 23 Avril 2020
[6] IRAMBONA Alphonsine, « Mise en place d'un système Automatise De Gestion De Production Et De Suivi Des Produits Agropastoraux Au Burundi : « Cas De La Cape ». Faculté des Sciences et de l'Ingénieur(FSI). Département de TIC, mémoire, 03 Juillet 2020
[7] Mahmoud Baklouti. « E-learning : Présentation. aspects. enjeux et avenir », Mémoire de mastère. UNIVERSITE DE SFAX, Année 2003.

Les webographies

[1]
http://www.allaboutelearning.lu/cms/elearning/content.nsf/id/ApprenantElearningEnPratique-Communication. visité en février 2010.
[2]
https://docs.google.com/forms/d/1dGBIny3CHtZLSk1hvIxoYLa5AzzxcXCeZ2o954O8WMc/edit#responses
[3] https://httpd.apache.org/docs/2.4/mod/mpm_ common.html [Page consultée le 14 Septembre 2018].
[4] https://siecledigital.fr/2018/02/14/le-nombre-dinternautes-dans-le-monde-atteint-les-4-milliards/ [Page consultée le 22 mars 2018].
[5] CHEDJOU. KAMDEM. « Etat des lieux 2017 d'internet et des réseaux Sociaux en Afrique ». @INESSTB (Janvier 2017) http://cmdafrique.net/2017/01/25/etat-lieux-2017-dinternet-reseaux-sociaux- afrique/ [Page consultée le 22 mars 2018].
[6] https://www.webrankinfo.com/dossiers/google/chiffres-cles [Page consultée le 14 mars 2018].

[7] https://www.webrankinfo.com/dossiers/google/chiffres-cles [Page consultée le 14 mars 2018].

[8] https://fr.wikipedia.org/wiki/File_d'attente, visité le 05 janvier 2018

[9] https://www.alsacreations.com/article/lire/1205-outils-integration-web-debuter.html

[10] https://www.supinfo.com/articles/single/3210-cycle-vie-logiciel

[11] https://openclassrooms.com/courses/creez-des-applications-de-qualiteavec-ledesign-pattern-mvc/le-genie-logiciel-gl, visité le 10 Avril 2017

[12] https://www.supinfo.com/articles/single/2519-architecture-client-serveur

[13] https://www.supinfo.com/articles/single/3408-presentation-pattern-mvc

Annexe

Internet	1																			
Travail	4																			
Vie dans la communauté	6																			
Information nouvelle	7																			
Plutôt rassurée(e)	10																			
Plutôt angoissée(e)																				
Plutôt en colère	11																			
Autre(s) sentiment(s)	12																			
Manque d'autonomie des étudiants			1																	
Confiance dans la stratégie pour les travaux en ligne			4																	
Confiance concernant l'organisation des examens			3																	
Confiance dans la stratégie qui va permettre de développer le numérique			2																	
Confiance dans la capacité d'adaptation des enseignants			5																	
Ordinateur portable				6																
Téléphone avec internet				2																
Ordinateur fixe				13																
Message électronique					4															
Vidéos ou audioconférence					5															
Réseaux sociaux					7															
Plateforme de formation					1															
Partage des ressources par les étudiants						9														
Test d'évaluation en ligne						12														
Conception des ressources avec les étudiants						8														
Ressources proposées par des sites académiques							1													
Outil de travail collaboratif							3													
Sites éducatifs ou culturels							4													
Echange direct avec les étudiants								2												
Envoie et réception des devoirs par messagerie								11												
Dépôt de cours sur les plateformes																				
Travail en groupe en salle informatique									6											