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

Mise en place d’un centre E-Learning à l’Université du Burundi

Niyungeko, Nobel Alfred Sauvy; Sous la direction du : Dr. Mukeshimana Michèle

2023-03
UB, FSI
https://repository.ub.edu.bi/handle/123456789/2436

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



Master en Génie Informatique

Année Académique: 2021-2022

MISE EN PLACE D'UN CENTRE E-LEARNING A L'UNIVERSITE DU BURUNDI

MEMOIRE

Présenté Par

NIYUNGEKO Nobel Alfred Sauvy

à la

Faculté des Sciences de l'Ingénieur (FSI)

En vue de l'obtention du grade de

MASTER
en
Génie Informatique

Soutenu le 03/03/2023, devant le jury composé de :

Dr.	**NDAYISABA Longin**	Président
Dr.	**NIBITANGA Roméo**	Vice-Président
Pr.	**NDIKUMAGENGE Jérémie**	Secrétaire
Dr.	**MUKESHIMANA Michèle**	Directeur

Identification des membres du jury

Dr. NDAYISABA Longin : Président

Dr. NIBITANGA Roméo : Vice-Président

Pr. NDIKUMAGENGE Jérémie : Secrétaire

Dr. MUKESHIMANA Michèle : Directeur

Dédicaces

À mes chers parents ;

À mes sœurs et à mon frère ;

À mes oncles et tantes ;

À mes cousins et cousines ;

À mes camarades de classe ;

À mes amis et connaissances.

NIYUNGEKO Nobel Alfred Sauvy

Remerciements

J'adresse ma profonde gratitude à Dieu le Tout Puissant et Miséricordieux pour m'avoir accordé la force de réaliser ce travail jusqu'au bout.

Mes sincères remerciements s'adressent au Dr MUKESHIMANA Michèle, enseignante à l'Université du Burundi dans la Faculté des Sciences de l'Ingénieur, pour m'avoir guidé dans la réalisation de ce travail malgré ses multiples fonctions. Ses conseils judicieux et sa rigueur scientifique m'ont été d'une très grande utilité.

Je tiens à remercier également toute personne qui, de près ou de loin, a participé à la réalisation de ce travail, à mes parents qui ont eu la curiosité de me guider dans l'accomplissement du présent mémoire et plus particulièrement à tous les enseignants du département de Technologie de l'Information et de la Communication de l'Université du Burundi.

Résumé

Un centre E-Learning ou l'éducation en ligne est une salle de classe sur Internet qui aide les étudiants à étudier indépendamment de leurs localisations géographiques et du temps. Il n'aide pas seulement les étudiants mais aussi les enseignants. Les autres noms de l'éducation en ligne sont l'apprentissage en ligne, les cyber-écoles ou l'apprentissage à distance, qui se distingue de l'enseignement traditionnel en face à face car les étudiants n'ont pas besoin de se rendre dans une salle de classe et d'écouter un instructeur leur parler. C'est un apprentissage autodirigé.

Au Burundi comme ailleurs dans le monde le nombre d'établissements scolaires n'augmente pas proportionnellement aux effectifs des jeunes scolarisés et de plus on remarque un manque d'enseignants pour instruire ces jeunes.

L'éducation académique souffre des mêmes problèmes. De surcroit l'éducation académique n'est pas internationalisée : les étrangères ou étrangers le voulant, sont obligées/obligés d'amorcer des démarches de demande de visa et autres formalités dans l'espoir de devenir membre du parcours académique.

Dans la perspective de trouver des solutions aux problèmes évoqués le présent mémoire propose de mettre en place un centre d'apprentissage en ligne ou E-Learning ouvrant les portes de l'Université du Burundi aux nationaux et internationaux désirant interagir avec ses activités académiques indépendamment de l'espace et du temps.

Pour y aboutir nous nous sommes renseignés sur l'état de l'apprentissage à distance dans les universités francophones issues de l'AUF et par la suite nous avons réalisé la conception du système en utilisant un langage de modélisation dénommé UML. Pour modéliser le système nous avons utilisé l'algèbre relationnelle et avons implémenté le nouveau centre en utilisant les langages de programmation côté client (HTML, CSS et JS) et un langage de programmation côté serveur (PHP).

Le centre E-Learning permettra aux étudiants et enseignants d'interagir en ligne indépendamment de l'espace : un enseignant pourra charger un cours et sa description (textuelle et multimédia) et un étudiant après avoir lu et apprécié la description du cours pourra s'enrôler à ce dernier en ligne.

Mots clés : E-Learning, Université du Burundi, étudiant, enseignant, cours, modélisation UML, algèbre relationnelle.

Abstract

An E-Learning center or online education is an internet-based classroom that helps students study independently of their geographic locations and time. It not only helps students but also teachers. Other names for online education are E-Learning, e-schools or distance learning, which differs from traditional face-to-face education as students do not need to travel in a classroom and listen to an instructor talk to them. It is self-directed learning.

In Burundi, as elsewhere in the world, the number of schools is not increasing in proportion to the number of young people attending school and, moreover, there is a lack of teachers to instruct these young people.

Academic education suffers from the same problems. In addition, academic education is not internationalized: foreigners wishing to do so are obliged to initiate visa application procedures and other formalities in the hope of becoming a member of the academic course.

In the perspective of finding the solutions to the problems mentioned, this thesis proposes to set up an online learning center or E-Learning opening the doors of the University of Burundi to nationals and internationals wishing to interact with its academic activities independently of the space and time.

To achieve this, we learned about the state of distance learning in French-speaking universities and then we designed the system using a modeling language called UML. To model the system we used relational algebra and implemented the new center using client-side programming languages (HTML, CSS, and JS) and a server-side programming language (PHP).

The E-Learning center will allow students and teachers to interact online regardless of space: a teacher can load a course and its description (textual and multimedia) and a student after reading and appreciating the description of the course can enlist to the latter online.

Keywords: E-Learning, University of Burundi, student, teacher, course, UML modeling, relational algebra.

Table des matières

Identification des membres du jury i
Dédicaces ii
Remerciements iii
Résumé iv
Abstract v
Table des matières vi
Liste des figures ix
Liste des tableaux x
Liste des sigles et abréviations xi
Avant-propos xiii
Chapitre I. Généralités et cadre du projet 1
I.1. Généralités 1
I.2. Objectifs du projet 2
I.2.1. Objectif global 2
I.2.2. Objectifs spécifiques 2
I.3. Problématiques 3
I.4. Solutions proposées 4
I.5. Méthodologie de travail utilisée 4
I.6. Outils et moyens utilisés 4
I.7. Résultats attendus 5
I.8. Apports scientifiques et technologiques 5
Chapitre II. Etude de l'existant dans les universités francophones 6
II.1. Introduction 6
II.2. Avantages et freins à la mise en place des services de formation à distance 8
II.3. Progressivité de mise en place de services de formation à distance 9
Chapitre III. Conception du système d'information de la mise en place d'un centre E-Learning 11
III.1. Introduction 11
III.2. Modèles du cycle de vie d'un logiciel 11
III.2.1. Introduction 11
III.2.2. Types des modèles de cycle de vie d'un logiciel 11

III.3. Modélisation du nouveau système avec le langage UML 16
III.3.1. Introduction 16
III.3.2. Diagramme des cas d'utilisation 18
III.3.3. Diagramme d'activités 25
III.3.4. Diagramme de séquence 26
III.3.5. Diagramme de classes 28
III.3.6. Diagramme de déploiement 29
Chapitre IV. Modélisation mathématique avec l'algèbre relationnelle 31
IV.1. Introduction 31
IV.2. Opérateurs relationnels et leurs traductions en SQL 31
IV.2.1. Sélection 31
IV.2.2. Projection 32
IV.2.3. Renommage 34
IV.2.4. Union 34
IV.2.5. Intersection 36
IV.2.6. Différence 36
IV.2.7. Produit cartésien 37
IV.2.8. Jointure 38
Chapitre V. Implémentation et présentation des fonctionnalités du centre 43
V.1. Introduction 43
V.2. Architecture client serveur 43
V.3. Outils d'implémentation utilisés 44
V.3.1. Editeur texte 44
V.3.2. Serveur web 45
V.3.3. Serveur de base de données 45
V.3.4. PHP Data Objects 46
V.3.5. Langage de balisage 47
V.3.6. Langage de programmation des scripts 47
V.3.7. Langage de script côté serveur 48
V.3.8. FPDF 48
V.3.9. WebRTC 48
V.4. Sécurité informatique et son rôle dans E-Learning 48
V.4.1. Introduction 48

V.4.2. Objectifs de protection informatique 49
V.4.3. Stockage des mots de passe 50
V.5. Présentation des fonctionnalités du centre E-Learning 50
V.5.1. Page d'accueil 51
V.5.2. Interface d'enregistrement 51
V.5.3. Interface d'authentification 52
V.5.4. Interface d'un administrateur 53
V.5.5. Interface d'un enseignant 55
V.5.6. Interface d'un étudiant 56
V.5.7. Interface de vidéo conférence 57
Conclusion générale et Recommandations 58
1. Conclusion générale 58
2. Recommandations 58
Références bibliographiques 59

Liste des figures

Figure 1 : Illustration d'un modèle en cascade 12
Figure 2 : Illustration d'un modèle en V 13
Figure 3 : Illustration d'un modèle incrémental 14
Figure 4 : Illustration d'un modèle itératif 15
Figure 5 : Illustration d'un modèle en spirale 16
Figure 6 : Association entre un acteur et un cas d'utilisation 18
Figure 7 : Relation de généralisation entre acteurs 19
Figure 8 : Relation d'inclusion et d'extension entre cas d'utilisation 19
Figure 9 : Relation de généralisation entre cas d'utilisation 20
Figure 10 : Diagramme des cas d'utilisation du centre E-Learning 22
Figure 11: Diagramme d'activités pour le chargement d'un cours 26
Figure 12: Diagramme de séquence « Télécharger un cours » 28
Figure 13: Diagramme de classes du centre 29
Figure 14: Diagramme de déploiement du centre 30
Figure 15: Page d'accueil du centre 51
Figure 16: Interface d'enregistrement 52
Figure 17: Interface d'authentification 52
Figure 18: Interface d'accueil d'un administrateur 53
Figure 19: Interface de définition des catégories des cours par un administrateur 54
Figure 20: Interface de consultation des cours par un administrateur 54
Figure 21 : Interface d'accueil d'un enseignant 55
Figure 22: Interface d'un enseignant pour l'ajout des descriptions d'un cours 55
Figure 23: Interface d'accueil d'un étudiant 56
Figure 24: Interface de téléchargement des cours par un étudiant 57
Figure 25: Interface de vidéo conférence entre un enseignant et des étudiants 57

Liste des tableaux

Tableau 1 : Identification des acteurs et des cas d'utilisation 20
Tableau 2 : Description du cas d'utilisation « S'authentifier » 23
Tableau 3 : Description du cas d'utilisation « Charger un cours en PDF » 23
Tableau 4 : Description du cas d'utilisation « Consulter les descriptions d'un cours » 24
Tableau 5 : Description du cas d'utilisation « Télécharger les cours » 24

Liste des sigles et abréviations

4D	:4th Dimension
A/A	:Année Académique
AES	:Advanced Encryption Standard
AJAX	:Asynchronous JavaScript and XML
API	:Application Programming Interface
AUF	:Agence Universitaire Francophone
CSS	:Cascading Style Sheets
Dr	: Docteur
DSN	: Data Source Name
FPDF	: Free Portable Document Format
HTML	: HyperText Markup Language
HTTP	: HyperText Transfer Protocol
IETF	: Internet Engineering Task Force
IOT	: Internet Of Things
JS	: JavaScript
MD5	: Message Digest 5
MS SQL	: Microsoft SQL Server
P2P	: Peer-to-Peer
PC	: Personal Computer
PDF	: Portable Document Format
PDO	: PHP Data Objects
PECL	: PHP Extension Community Library
PHP	: Hypertext Preprocessor
Pr	: Professeur
SDLC	: Software Development Life Cycle
SQL	: Structured Query Language
TIC	: Technologie de l'Information et de la Communication
UML	: Unified Modeling Language

UNESCO	: United Nations Educational, Scientific and Cultural Organization
W3C	: World Wide Web Consortium
Web RTC	: Web Real Time Communication
XHTML	: Extensible HyperText Markup Language

Avant-propos

Le mémoire intitulé « Mise en place d'un centre E-Learning à l'Université du Burundi » permet à l'Université du Burundi d'ouvrir ses portes tant aux nationaux qu'aux internationaux désirant participer dans ses activités en étant des étudiants ou des enseignants indépendamment de l'espace.

Nous avons organisé le travail en cinq chapitres. Le premier chapitre décrit le cadre du travail en montrant la contextualisation, les objectifs du travail, les problèmes à surmonter et leurs solutions. Nous y discutons aussi les outils employés pour arriver aux résultats obtenus et l'apport tant scientifique que technologique que le travail apporte. Le deuxième chapitre discute de la présence de l'Université virtuelle dans les Universités francophones issues de l'Agence Universitaire Francophone en montrant les bénéfices et défis à relever. Le troisième chapitre décrit les différentes phases de développement d'un logiciel et montre comment nous avons modélisé le travail en utilisant le langage de modélisation unifié. Quant au quatrième chapitre nous décrivons la modélisation mathématique du travail avec l'algèbre relationnelle. Enfin le cinquième chapitre décrit l'architecture logicielle utilisée pour mettre en place le centre et présente quelques fonctionnalités jugées essentielles du centre.

Le choix du thème du mémoire a été motivé par la fermeture de l'Université du Burundi aux étudiants ne pouvant pas assurer leurs présences physiques et dans l'espoir d'augmenter l'effectif des enseignants.

Chapitre I. Généralités et cadre du projet

I.1. Généralités

Pour le développement durable du monde terrestre l'éducation constitue un droit à chacun de nous.

Dès notre plus jeune âge nous fréquentons des écoles en présentiel (physiquement) dans l'optique d'éviter l'analphabétisme et ses conséquences. Cela constitue un handicap à celles ou ceux ne pouvant pas le faire, notamment à cause de leurs handicaps physiques.

Les étudiants poursuivent des études supérieures pour l'amour de l'apprentissage et de la découverte. Pour d'autres, la motivation culturelle et les traditions familiales ont une influence incontestablement positive sur la décision de poursuivre des études supérieures [1].

En France, le nombre d'étudiants n'en finit plus de grimper, le nombre d'étudiants inscrits dans l'enseignement supérieur pour l'année universitaire 2018-2019 a progressé de 2,1% à 2,7 millions selon une étude ministérielle [2]. De même le Burundi est confronté à ce surnombre [3]. De surcroît l'effectif du personnel enseignant résidant au Burundi est limité [4].

Les étudiantes ou étudiants souhaitant un enseignement de qualité dans un domaine particulier dans un pays qui n'est pas leurs pays de naissance se heurtent à l'absence de ce dernier.

C'est dans le but d'éviter les problèmes évoqués que des formations à distance sont dispensées sur le réseau des réseaux qu'est l'internet. C'est ce qu'on appelle « E-Learning ».

En 1999, l'expression « E-Learning » est évoquée pour la première fois dans un contexte professionnel par Elliot Masie lors de la conférence TechLearn à Disneyworld. Cependant, les toutes premières tentatives d'apprentissage en ligne remontent à 1924, lorsque Sidney Pressey, professeur à l'Ohio State University, a créé la toute première machine d'apprentissage électronique, l'Automatic Teacher [5].

Le E-Learning tire son attrait du fait de pouvoir apprendre à son rythme, sur son ordinateur ou autre équipement mobile, des contenus pédagogiques sur des sujets variés.

De plus les universités utilisant le E-Learning peuvent accueillir dans leurs effectifs des étudiants nationaux n'ayant pas le temps d'atteindre les cours en présentiel et aux internationaux.

L'une des bénéfices de la formation à distance provient de l'accessibilité qu'elle offre.

Effectivement, il suffit d'une connexion à l'internet pour être en mesure de suivre une formation à distance, même si celle-ci se déroule dans un autre continent. La formation à distance offre des coûts de formation réduits.

Un autre avantage de la formation à distance réside dans la personnalisation du parcours. En effet, il existe autant de formats que de possibilités d'adapter et d'individualiser les parcours de formation selon les besoins de chacun.

Le E-Learning est de plus en plus utilisé à travers le monde, la part de marché de E-Learning mondialement est aujourd'hui de plus de 190 milliards de dollars. On estime qu'en 2025, il devrait atteindre les 325 milliards de dollars, C'est un marché en pleine expansion.

Après les États-Unis qui représentent presque 40% du marché mondial, l'Asie est le second plus grand marché [6].

Le continent africain rattrape son retard et grimpe dans le classement du marché du E-Learning à une vitesse incroyable, notamment depuis le début de la pandémie Covid-19. Fin 2020, la plateforme gratuite de E-Learning Atingi a été déployée et est disponible sur l'intégralité du continent [7].

Au Burundi de telle plateforme existe comme par exemple l'ecampus de l'Université Lumière de Bujumbura [8]. D'autres universités dont l'Université du Burundi pourraient s'y inspirer.

C'est dans cette optique que le présent projet intitulé « La mise en place d'un centre E-Learning à l'Université du Burundi » fait l'objet de cette étude.

I.2. Objectifs du projet

I.2.1. Objectif global

Le projet a comme objectif global de décentraliser l'éducation académique octroyée par l'Université du Burundi par la création d'un centre E-Learning (apprentissage en ligne) au sein de la dite Université dans le but de dispenser l'éducation au plus grand nombre d'apprenants et dans l'espoir d'augmenter les effectifs des enseignants.

I.2.2. Objectifs spécifiques

Dans la perspective de créer un centre E-Learning à l'Université du Burundi le présent mémoire va satisfaire aux objectifs spécifiques suivants:

1) Mettre en place un centre E-Learning ayant comme utilisateurs les étudiants, les enseignants et un administrateur ;
2) Mettre en place une base de données permettant la gestion du centre ;
3) Mettre en place un centre E-Learning en mode asynchrone où un étudiant et enseignant pourra accéder au centre quand il le souhaite ;
4) Gérer les droits d'accès des utilisateurs du centre ;
5) Donner la possibilité aux enseignants de définir des nouveaux cours et leurs descriptions textuelles et multimédia ;
6) Accorder le droit aux étudiants de consulter les descriptions d'un cours définis par un enseignant avant l'enrôlement à ce cours ;
7) Permettre une vidéo conférence en mode synchrone entre un enseignant et des étudiants ;
8) Permettre la génération des rapports par l'utilisation de la librairie FPDF.

I.3. Problématiques

Notre motivation à mettre en place un centre E-Learning au sein de l'Université du Burundi se repose sur la volonté de résoudre les difficultés rencontrées au sein de la dite Université dont :

1) Le nombre de classes ou d'auditoires pour l'accueil des étudiants est limité dans les enceintes de l'Université du Burundi ;
2) Les effectifs des enseignants sont insuffisants ;
3) La fermeture des portes de l'Université à chaque personne ne pouvant pas assurer leurs présences physiques au sein de l'Université ;
4) L'obligation aux internationaux à chercher des formalités administratives comme le permis de résidence pour être membre du corps estudiantin ou académique de l'Université du Burundi ;
5) Les étudiants et enseignants doivent suivre des horaires de travail définis par l'Université.

I.4. Solutions proposées

A la suite des problèmes évoqués le projet propose la création d'un centre E-Learning à l'Université du Burundi avec comme solutions:

1) La mise en place d'un centre E-Learning ayant comme utilisateurs les étudiants, les enseignants et un administrateur ;
2) La mise en place d'une base de données permettant la gestion du centre ;
3) La mise en place d'un centre E-Learning en mode asynchrone où un étudiant et enseignant pourront accéder au centre quand il le souhaite ;
4) La gestion des droits d'accès des utilisateurs du centre ;
5) L'octroi du droit aux enseignants de définir des nouveaux cours et leurs descriptions textuelles et multimédia ;
6) L'accord du droit aux étudiants de consulter les descriptions d'un cours définis par un enseignant avant l'enrôlement à ce cours ;
7) La possibilité d'amorcer une vidéo conférence entre un enseignant et des étudiants ;
8) La possibilité de générer des rapports par l'utilisation de la librairie FPDF.

I.5. Méthodologie de travail utilisée

L'analyse de l'existant dans les universités francophones issues de l'AUF, la lecture des livres et des articles sur l'Internet ont été utilisés pour aboutir à la mise en place de notre centre E-Learning.

I.6. Outils et moyens utilisés

Pour aboutir au bout de notre travail nous avons utilisé les outils suivants : le langage de modélisation unifié UML comme langage de modélisation, l'algèbre relationnelle comme outil mathématique de modélisation.

Pour l'implémentation du système nous avons fait l'usage des langages de programmation PHP, HTML, CSS, JavaScript.

Nous avons aussi utilisé APACHE comme serveur web et MySQL comme système de gestion de bases de données et FPDF une librairie PHP pour la génération des rapports sous format PDF.

Nous avons utilisé la technologie Web Real-Time Communication (Web RTC), littéralement « communication en temps réel pour le Web » pour la vidéo conférence entre enseignant et étudiants.

I.7. Résultats attendus

La mise en place d'un centre E-Learning à l'Université du Burundi permet que/qu' :

1) Un anonyme indépendamment de sa localisation géographique et de son temps peut consulter le centre ;
2) Chaque acteur du centre peut s'authentifier ;
3) Un enseignant peut définir un cours et sa description tant textuelle que multimédia ;
4) Un étudiant peut consulter les cours et descriptions définis par un enseignant ;
5) Un enseignant peut décider d'enseigner par vidéo conférence ;
6) Un étudiant après s'être enrôlé à un cours peut télécharger le document PDF du cours.

I.8. Apports scientifiques et technologiques

La mise en place d'un centre E-Learning de l'Université du Burundi constitue un apport scientifique avec l'usage de l'algèbre relationnelle et constitue un apport technologique avec l'amorcement de la digitalisation des activités de l'Université du Burundi dans le but d'ouvrir les portes de l'Université au monde entier avec la mise en place des cours et des descriptions des cours par un enseignant. Un étudiant a le droit de s'enrôler à un cours et télécharger un guide PDF du cours après l'appréciation des descriptions de ce dernier.

Dans ce chapitre nous avons montré les objectifs du projet, les problèmes à résoudre et avons montré la solution qui consiste à mettre en place un centre E-Learning au sein de l'Université du Burundi. Dans le chapitre suivant nous allons analyser l'existant dans les Universités de l'Agence Universitaire Francophone.

Chapitre II. Etude de l'existant dans les universités francophones

II.1. Introduction

Lorsqu'en 1993, le vice-président des Etats-Unis, Albert Arnold Gore dit Al Gore, annonce l'ouverture d'un nouveau chantier, les autoroutes de l'information, pouvions-nous imaginer que nous aborderions une révolution technologique touchant tous les secteurs de nos sociétés. Les universités, en particulier celles des pays les plus avancés ont apporté une forte contribution, en termes de recherche, à l'émergence des nouvelles technologies de l'information.

Depuis l'invention du bouquet de protocole IP par Vinton Cerf en 1974, quinze années ont suffi pour créer les fondements d'une intelligence collective. Les années 1985 à 1995 ont vu croître la place des didacticiels dans le télé-enseignement, se substituant peu à peu à l'enseignement autodirigé ou à l'enseignement programmé ayant recours aux supports de cours traditionnels ou audiovisuels. Les années 1995-2000 ont transformé et élargi ce processus d'évolution, en délocalisant apprenants et formateurs, en reconstituant les supports de cours naturels d'enseignement, en faisant transiter sur les réseaux de transmission de données les contenus stables et enrichis, en associant aux contenus magistraux la documentation requise, en permettant enfin, par correspondance électronique, par forum, les échanges entre enseignants et étudiants, en mondialisant le savoir.

Les innovations dans le cursus académiques se multiplient et se diversifient reposant essentiellement sur les nouvelles technologies de l'information. On entend fréquemment parler de mobilité des étudiants, de formations délocalisées ou en alternance et de télé-enseignement. En effet, une pratique existante de l'enseignement, l'université virtuelle, pourrait se développer à l'Université du Burundi. Elle pourrait modifier nos habitudes, sans que nulle part ne soit remise en question la nécessité de l'enseignement traditionnel. Ce qui distingue l'université virtuelle des autres universités, c'est la méthode qu'elle utilise pour diffuser son enseignement. Il s'agit d'une formule souple taillée sur mesure, les plages et les modes d'interaction peuvent être choisis plus librement entre les partenaires (étudiants, assistants et enseignants) utilisant une variété de supports multimédias.

L'université virtuelle conduit à l'abstraction de l'espace. Pour les formations, les étudiants auront la possibilité de suivre une formation se déroulant sur un plan national et international, et ce, en restant physiquement près de leurs lieux de résidence.

Avec l'apprentissage au moyen de la technologie, le professeur n'est plus la source privilégiée de savoir. Autrement dit, il devient un « guide d'appoint » plutôt que « le maître sur la scène ». Au lieu d'être la personne qui connaît toutes les réponses et qui décide des questions, l'enseignant devient « un apprenant expert » qui peut aider les élèves à résoudre des problèmes et à trouver les réponses à leurs questions.

Toutefois l'audace et l'originalité du concept utilisé reste tributaire des technologies de l'information et plus particulièrement de la qualité de retransmissions. Des problèmes de capacité de réseaux de communication restent à résoudre. La saturation d'Internet ne permet ainsi pas toujours d'obtenir un niveau d'interactivité adéquat. Ce Monde Virtuel n'a aucune frontière, aucun terrain géographique [9].

Si historiquement la formation initiale et la création de connaissances nouvelles formaient le cœur de la fonction universitaire, les tensions actuelles indiquent clairement que des évolutions fortes sont nécessaires. La formation tout au long de la vie, objectif proclamé par l'UNESCO ou par la Communauté Européenne, suppose des changements profonds du système éducatif et de ses méthodes.

Si chacun voit les avantages que l'on peut tirer d'une Université Virtuelle dans ce domaine, il s'agit de bien en mesurer les potentialités et les limites intrinsèquement liées.

La place croissante des technologies de l'information et de la communication annonce la fin du modèle traditionnel. Déjà, les jeunes enfants sont démotivés par les méthodes pédagogiques classiques, habitués aux jeux électroniques, leurs démarches intellectuelles procèdent plus de la méthode essai-erreur que du raisonnement scientifique se référant à des connaissances acquises.

En fait, c'est de plus d'autonomie de l'apprenant dont il est question, il ne s'agit plus d'acquérir des connaissances mais de les approprier. Le but de l'école devient alors « apprendre à apprendre » dans un espace autonome. Une expérience intéressante, menée à l'Université de Paris 8, a consisté à mettre en place des formations spécifiques a permis de quantifier, sur sept ans, une amélioration notable du succès et de l'autonomie des étudiants en premier cycle.

<<Apprendre est devenu l'un des grands enjeux de nos sociétés contemporaines », de nombreuses recherches pédagogiques s'intéressent aux didactiques disciplinaires mais ne font qu'effleurer l'utilisation des nouvelles technologies.

Il y a dans le corps enseignant comme une réserve, une pudeur naturelle par rapport aux effets de mode.

Et pourtant, de nombreux rapports attirent notre attention sur les enjeux des nouvelles technologies. Le monde éducatif ne peut pas rester à l'écart de l'utilisation massive dans la vie de tous les jours et dans le monde du travail de ces technologies. « La diffusion d'outils nouveaux peut offrir aussi au système d'éducation et de formation des opportunités et lui permettre de mieux faire face à une demande de formation à la fois plus massive et plus diversifiée» note le Commissariat Général au Plan.

II.2. Avantages et freins à la mise en place des services de formation à distance

Il y a de nombreux avantages, bien connus, ou parfois supposés sans être validés, qui poussent au développement de ces techniques.

Pour les Universités de régions défavorisées, elles permettront de compenser l'absence de spécialistes dans un cursus de formation. Des centres de télé-enseignement peuvent être ouverts, rapprochant les formations des étudiants. Nous ne pensons pas ici aux pays où chaque étudiant est raccordé à l'Internet à domicile et peut se payer son matériel propre et ses communications. La pédagogie est améliorée, plus interactive, au rythme de chaque étudiant.

Les enseignants, eux-mêmes, seront libérés de tâches répétitives, pour se consacrer à des tâches de conception de cours, d'exercices, de démonstrations animées et de dialogue à distance avec les étudiants.

D'un autre côté, il faut analyser quelques-unes des questions auxquelles il faut répondre pour lever les freins qui apparaissent pour la mise en œuvre effective de formations à distance.

Sur le plan administratif se pose la question de l'habilitation des formations à distance, une procédure qui nécessite de s'assurer de la pérennité de l'équipe pédagogique, comportant des enseignants qui appartiennent à plusieurs institutions distantes.

Il n'est pas très clair non plus que les questions économiques soient bien étudiées dans les diverses configurations qui peuvent se présenter.

Il s'agit en particulier des coûts initiaux et récurrents pour les équipements et les communications, pour le recrutement des techniciens qualifiés pour la mise en œuvre et la maintenance de ces systèmes ainsi que les coûts liés à la rémunération des enseignants dispensant la formation à distance.

Sur le plan technique, il faut assurer la confidentialité des sujets d'examens, des transferts de notes.

Il faut aussi tenir compte des décalages horaires si les étudiants sont dispersés dans le monde, authentifier les candidats. Chacun d'entre vous a pu également s'impatienter devant les temps de réponse parfois prohibitifs et décourageant des systèmes de communication.
Enfin, dans les Ecoles d'Ingénieurs, les principes pédagogiques qui passent par les travaux pratiques sont à identifier car tout ne pourra pas se faire par simulation.

II.3. Progressivité de mise en place de services de formation à distance

Compte tenu du besoin réel d'utilisation de la formation à distance et des contraintes que l'on vient d'évoquer, comment peut-on s'y prendre ? Il est possible, pour le Directeur des études, d'avoir une approche pragmatique qui analyse préalablement le plus finement possible les éléments qui ont été cités et de mettre en place des expériences de complexité progressive.
A chaque expérience nouvelle, les questions doivent être traitées : quelles techniques doivent être manipulées, est-ce que les personnes ressources sont en place, faut-il recruter ?
Quelle structuration des équipes pédagogiques faut-il installer ? Quels sont les problèmes administratifs, financiers et académiques nouveaux à régler ? Quelles sont les limites du processus mis en place ?
Une première étape consiste à introduire l'auto-formation comme partie d'un cursus. Les moyens techniques, Internet, toile sont connus. L'organisation des équipes pédagogiques n'est pas changée, toutefois l'organisation de la bibliothèque avec des ouvrages numériques est à revoir. Les limites sont donc au niveau du personnel technique disponible, des personnels de bibliothèque et des enseignants. La mise en œuvre nécessite des gens compétents en Informatique et réseaux, des enseignants aptes à construire des cours multimédias ou à les trouver sur la toile.
Une deuxième étape est d'insérer un cours à distance dans un cursus pour une classe.
Les moyens techniques sont ceux d'auto-formation, ceux de contacts différés entre le professeur et les étudiants, au moins par le courriel électronique ou en programmant une réunion à distance.
Un enseignant doit être disponible à distance pour répondre aux sollicitations des étudiants.

Nous avons montré les enjeux pour l'Université d'entrer de plein pied dans le domaine virtuel. L'intérêt des étudiants et de la société commande.

Tout ne peut pas être traité par les nouveaux médias, mais ils représentent un indéniable facteur de progrès, c'est le rôle de l'Université de se les approprier pour répondre en premier lieu à sa mission traditionnelle mais en l'élargissant à d'autres demandes sociales. Ces outils doivent être créés, améliorés et utilisés par les enseignants. L'enseignant ne doit plus se présenter physiquement face aux élèves devant le tableau noir. Le chapitre suivant montre la conception du centre E-Learning.

Chapitre III. Conception du système d'information de la mise en place d'un centre E-Learning

III.1. Introduction

Les systèmes d'information sont des collections de multiples ressources d'information (par exemple, les logiciels, le matériel, les connexions du système informatique, le logement du système, les utilisateurs du système et les informations du système informatique) pour collecter, traiter, stocker et diffuser des informations.
Des outils tels que les ordinateurs portables, les bases de données, les réseaux et les smartphones sont des exemples de systèmes d'information [10].
Pour concevoir notre système d'information nous avons utilisé le langage de modélisation unifié UML avec l'aide du logiciel Visual Paradigm.

III.2. Modèles du cycle de vie d'un logiciel

III.2.1. Introduction

Les processus logiciels sont un ensemble cohérent d'activités pour spécifier, concevoir, mettre en œuvre et tester des systèmes logiciels. Un modèle de processus logiciel est une représentation abstraite d'un processus qui présente une description d'un processus d'un point de vue particulier. Il existe de nombreux processus logiciels différents, mais tous impliquent la spécification définissant ce que le système doit faire ; la conception et la mise en œuvre définissant l'organisation du système et la mise en œuvre du système ; la validation vérifiant les besoins du client et l'évolution qui s'assure de la modification du système en réponse à l'évolution des besoins des clients.

III.2.2. Types des modèles de cycle de vie d'un logiciel

L'une des notions de base du processus de développement d'un logiciel sont les modèles de cycle de vie du développement logiciel (SDLC).
Il existe de nombreux modèles de cycle de vie de développement logiciel qui ont été développés afin d'atteindre différents objectifs requis. Les modèles précisent les différentes étapes du processus et l'ordre dans lequel elles sont réalisées.

Les modèles SDLC les plus utilisés sont : le modèle en cascade, le modèle en V, le modèle incrémental, le modèle itératif et le modèle en spirale.
Détaillons les spécificités de chaque modèle du cycle de vie de développement d'un logiciel.

Modèle en cascade

Considéré comme le modèle, à la fois le plus ancien et le plus simple de tous les autres modèles, le modèle en cascade est une décomposition des activités du projet en phases séquentielles linéaires où chaque phase dépend des livrables de la précédente et correspond à une spécialisation des tâches. Le modèle en cascade est relativement facile à mettre en œuvre et à gérer et suit les mêmes étapes séquentielles (spécification, conception, codage, validation ou test, maintenance) pour chaque projet.

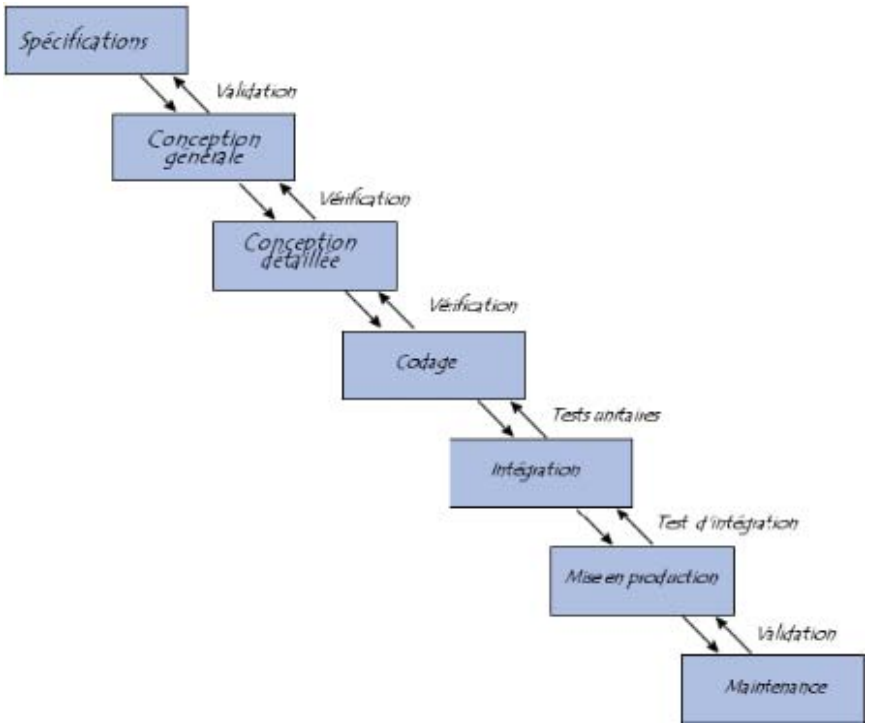


Figure 1 : Illustration d'un modèle en cascade

La figure 1 illustre schématiquement un modèle en cascade. Avec le modèle en cascade il faut faire attention que tous les détails de chaque phase, même les plus petits, soient effectivement complétés. Le cas échéant, l'ensemble du processus sera retardé.

Modèle en V

Le modèle en V représente un processus de développement qui peut être considéré comme une extension du modèle en cascade et est un exemple du modèle en V plus général.
Au lieu de descendre de manière linéaire, les étapes du processus sont courbées vers le haut après la phase de codage, pour former la forme typique en V.

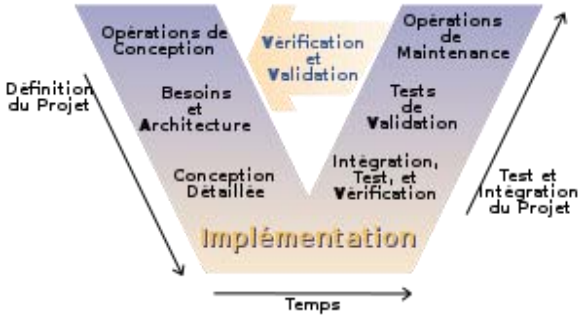


Figure 2 : Illustration d'un modèle en V

La figure 2 illustre un modèle en V. Le modèle en V tient d'avantage compte de la réalité, le processus de développement n'est pas réduit à un enchaînement de tâches séquentielles. Le modèle en V démontre les relations entre chaque phase du cycle de vie du développement et sa phase de test associée.

Modèle incrémental

Le modèle de construction incrémental est une méthode de développement logiciel dans laquelle le modèle est conçu, implémenté et testé de manière incrémentielle (un peu plus est ajouté à chaque fois) jusqu'à ce que le produit soit terminé. Cela implique à la fois le développement et la maintenance. Le produit est défini comme fini lorsqu'il satisfait à toutes ses exigences. Chaque itération passe par les phases d'exigences, de conception, de codage et de test. Et chaque version ultérieure du système ajoute une fonction à la version précédente jusqu'à ce que toutes les fonctionnalités conçues aient été mises en œuvre.

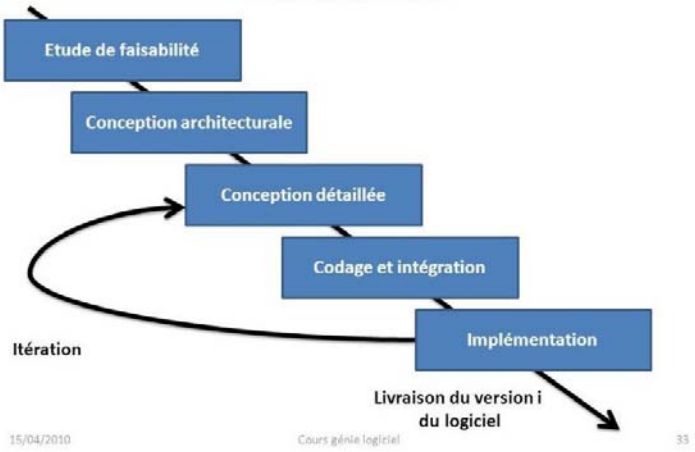


Figure 3 : Illustration d'un modèle incrémental

La figure 3 illustre schématiquement un modèle incrémental. Le modèle incrémental est utilisé lorsque l'équipe logicielle n'est pas très qualifiée ou qu'elle veut premièrement développer les tâches prioritaires et lorsqu'un projet a un long calendrier de développement.

Modèle itératif

Un modèle de cycle de vie itératif ne tente pas de commencer par une spécification complète des exigences, se concentre d'abord sur un ensemble initial de fonctionnalités utilisateur simplifiées, qui gagne ensuite progressivement en complexité et sur un ensemble plus large de fonctionnalités jusqu'à ce que le système ciblé soit complet. Lors de l'adoption de l'approche itérative, la philosophie du développement incrémental sera également souvent utilisée de manière libérale et interchangeable.

En d'autres termes, l'approche itérative commence par la spécification et la mise en œuvre d'une partie seulement du logiciel, qui peut ensuite être examinée et hiérarchisée afin d'identifier d'autres exigences. Ce processus itératif est ensuite répété en livrant une nouvelle version du logiciel à chaque itération. Dans un projet itératif léger, le code peut représenter la principale source de documentation du système ; cependant, dans un projet itératif critique, une spécification logicielle formelle peut également être requise.

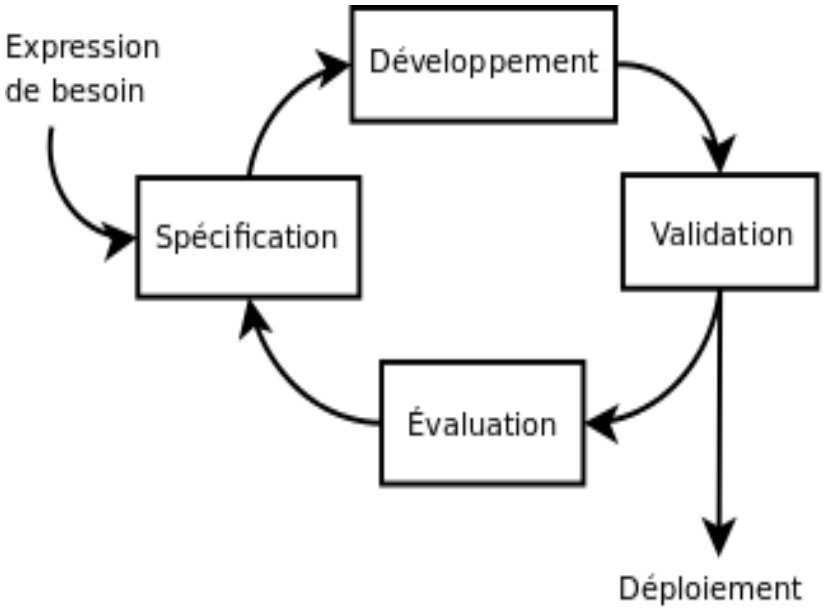


Figure 4 : Illustration d'un modèle itératif

La figure 4 montre une représentation schématique du modèle itératif. Lors de ce modèle les concepteurs développent une version rapide pour ensuite l'améliorer au fur et à mesure des tests. Ce modèle présente un principal inconvénient qu'il peut consommer rapidement des ressources s'il n'est pas contrôlé.

Modèle en spirale

Le modèle en spirale, décrit pour la première fois par Barry Boehm en 1986, est un modèle de processus de développement logiciel axé sur les risques qui a été introduit pour traiter les lacunes du modèle traditionnel en cascade.

Un modèle en spirale ressemble à une spirale avec de nombreuses boucles. Le nombre exact de boucles de la spirale est inconnu et peut varier d'un projet à l'autre. Ce modèle prend en charge la gestion des risques et le projet est livré en boucles. Chaque boucle de la spirale est appelée une phase du processus de développement logiciel.

La phase initiale du modèle en spirale dans les premières étapes du cycle de vie de la cascade qui est nécessaire pour développer un produit logiciel [11].

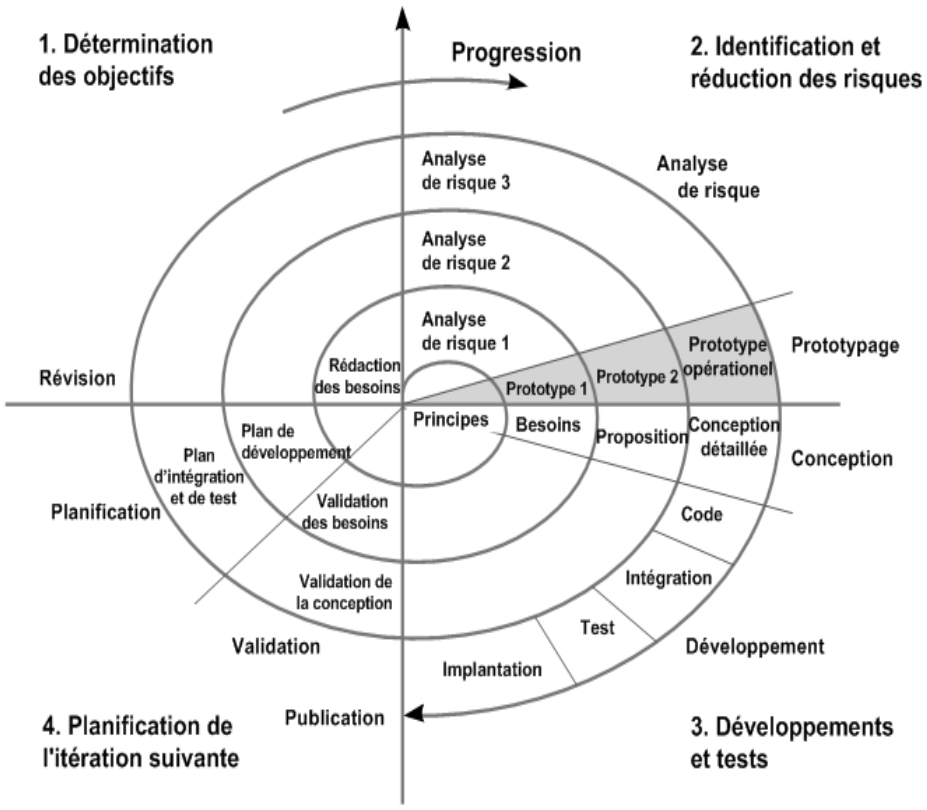


Figure 5 : Illustration d'un modèle en spirale

La figure 5 montre une représentation schématique du modèle en spirale. Avec le modèle en spirale le nombre exact de phases nécessaires au développement du produit peut être modifié par le chef de projet en fonction des risques du projet. Comme le chef de projet détermine dynamiquement le nombre de phases, le chef de projet a un rôle important pour développer un produit en utilisant un modèle en spirale.

Parmi les modèles de développement évoqués ci-haut nous avons utilisé le modèle itératif car le modèle permet de contrôler les spécificités d'un logiciel à plusieurs reprises et le déploiement n'ait effectué que lorsqu'on est en accord avec le fonctionnement du logiciel.

III.3. Modélisation du nouveau système avec le langage UML

III.3.1. Introduction

UML, abréviation de Unified Modeling Language en français Langage de Modélisation Unifié, est un langage de modélisation standardisé composé d'un ensemble intégré de diagrammes,

développé pour aider les développeurs de systèmes et de logiciels à spécifier, visualiser, construire et documenter les artefacts des systèmes logiciels, ainsi que pour la modélisation commerciale et d'autres systèmes non logiciels.
L'UML utilise principalement des notations graphiques pour exprimer la conception de projets logiciels. L'utilisation d'UML aide les équipes de projet à communiquer, à explorer des conceptions potentielles et à valider la conception architecturale du logiciel.

Les principaux objectifs de la conception de l'UML sont résumés par Meilir Page-Jones dans Fundamental Object-Oriented Design in UML comme suit :

1) Fournir aux utilisateurs un langage de modélisation visuel expressif prêt à l'emploi afin qu'ils puissent développer et échanger des modèles significatifs ;
2) Fournir des mécanismes d'extensibilité et de spécialisation pour étendre les concepts de base ;
3) Être indépendant des langages de programmation et des processus de développement particuliers ;
4) Fournir une base formelle pour comprendre le langage de modélisation ;
5) Prendre en charge les concepts de développement de niveau supérieur tels que les collaborations, les frameworks, les modèles et les composants [12].

UML 2.0 définit treize types de diagrammes, répartis en trois catégories : Six types de diagrammes représentent la structure statique de l'application ; trois représentent des types généraux de comportement; et quatre représentent différents aspects des interactions :

1) Diagrammes de structure : montrent la structure statique du système et de ses parties à différents niveaux d'abstraction et de mise en œuvre et comment ils sont liés les uns aux autres. Parmi les diagrammes de structure on a le diagramme de classes, le diagramme d'objets, le diagramme de composants, le diagramme de structure composite, le diagramme de packages et le diagramme de déploiement ;
2) Diagrammes de comportement : montrent le comportement dynamique des objets dans un système, qui peut être décrit comme une série de modifications du système au fil du temps. Parmi les diagrammes de comportement on a le diagramme de cas d'utilisation; diagramme d'activité et diagramme de machine d'état ;

3) Diagrammes d'interaction : tous dérivés du diagramme de comportement plus général, comprennent le diagramme de séquence, le diagramme de communication, le diagramme de temps et le diagramme global d'interaction.

Pour modéliser notre système nous avons utilisé cinq diagrammes dont deux décrivent la représentation statique du système (diagramme de classes et diagramme de déploiement) et trois autres décrivent la représentation dynamique (diagramme des cas d'utilisation, diagramme de séquence et diagramme d'activités).

III.3.2. Diagramme des cas d'utilisation

Le diagramme des cas d'utilisation est un type de diagramme comportementale UML qui vous permet de modéliser les fonctions du système (c'est-à-dire les objectifs) ainsi que les acteurs qui interagissent avec ces fonctions. Les cas d'utilisation vous permettent d'établir un lien entre ce dont vous avez besoin d'un système et la manière dont le système répond à ces besoins.
Un diagramme des cas d'utilisation se compose de (du') :

1) Acteur : est une entité qui joue un rôle dans un système donné. Il peut s'agir d'une personne, d'une organisation ou d'un système externe et est représenté sous la forme d'un individu ;
2) Cas d'utilisation : représente une fonction ou une action au sein du système et est représenté sous la forme d'une sphère ;
3) Système : est utilisé pour définir la portée du cas d'utilisation et dessiné sous la forme d'un rectangle [13].

Il existe cinq types de relations dans un diagramme des cas d'utilisation. Elles sont :

1) Association entre un acteur et un cas d'utilisation : indique une liaison entre un acteur et ce que ce dernier peut accomplir.



Figure 6 : Association entre un acteur et un cas d'utilisation

La figure 6 représente une association entre l'acteur 'Enseignant' et le cas d'utilisation 'Charger un cours en PDF' où un enseignant peut charger un nouveau cours sous format PDF;

2) Relation de généralisation entre acteurs : montre une relation d'héritage entre acteurs et est représentée par une ligne avec comme extrémité un triangle.



Figure 7 : Relation de généralisation entre acteurs

La figure 7 illustre une relation de généralisation ou d'héritage entre les acteurs 'Internaute' et 'Enseignant' où un enseignant est aussi un internaute connu par le centre;

3) Relation d'inclusion et d'extension entre deux cas d'utilisation : La relation d'inclusion montre une association impérative entre cas d'utilisation. Par exemple le cas d'utilisation 'Authentification' contient aussi le comportement décrit dans le cas d'utilisation 'Gérer les étudiants'. Un administrateur va gérer les étudiants après s'être authentifié.

La relation d'extension montre une association facultative entre cas d'utilisation. Par exemple le cas d'utilisation 'Effacer les détails d'un étudiant' ajoute son comportement au cas d'utilisation 'Gérer les étudiants'. Un administrateur peut gérer les étudiants en effaçant les détails de ce dernier, ce n'est pas obligatoire.

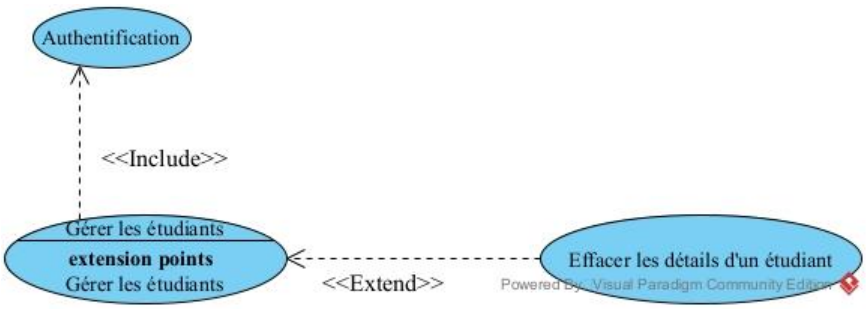


Figure 8 : Relation d'inclusion et d'extension entre cas d'utilisation

La figure 8 montre un exemple de relations d'inclusion et d'extension entre cas d'utilisation. Un administrateur du centre doit s'authentifier avant de gérer les étudiants. Un administrateur s'il le souhaite peut effacer les détails d'un étudiant ;

4) Relation de généralisation entre cas d'utilisation : où le cas d'utilisation enfant est une spécialisation (héritage) du cas d'utilisation parent.

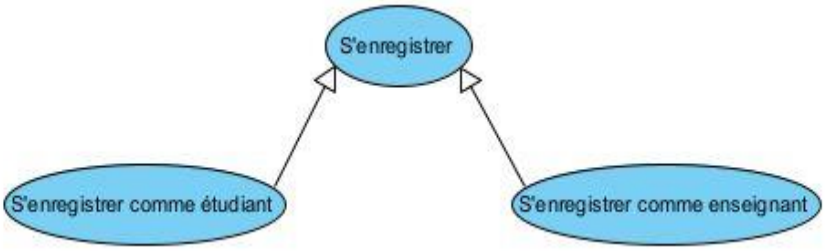


Figure 9 : Relation de généralisation entre cas d'utilisation

La figure 9 montre la relation de généralisation entre cas d'utilisation. Un internaute visitant le centre E-Learning peut s'enregistrer comme un étudiant ou un enseignant.

Identification des acteurs et des cas d'utilisation du centre

Il est temps d'identifier les acteurs et les cas d'utilisation qui vont participer à la mise en place de notre centre. Le tableau 1 illustre le cas.

Tableau 1 : Identification des acteurs et des cas d'utilisation

Acteurs	Cas d'utilisation
Internaute	1) Recherche des cours 2) Consulter les catégories des cours 3) Consulter les cours
Enseignant, Étudiant, Administrateur	Authentification
Enseignant	1) Charger un cours en PDF 2) Charger les descriptions du cours 3) Donner un cours en direct

Étudiant	1) Explorer les cours 2) Consulter les descriptions d'un cours 3) S'enrôler au cours 4) Télécharger les cours
Administrateur	1) Gérer les cours 2) Gérer les catégories des cours 3) Gérer les étudiants 4) Gérer les enseignants

Après l'identification des acteurs et des cas d'utilisation, on peut montrer les fonctionnalités de base dans un diagramme des cas d'utilisation du nouveau système de la mise en place d'un centre E-Learning à l'Université du Burundi.

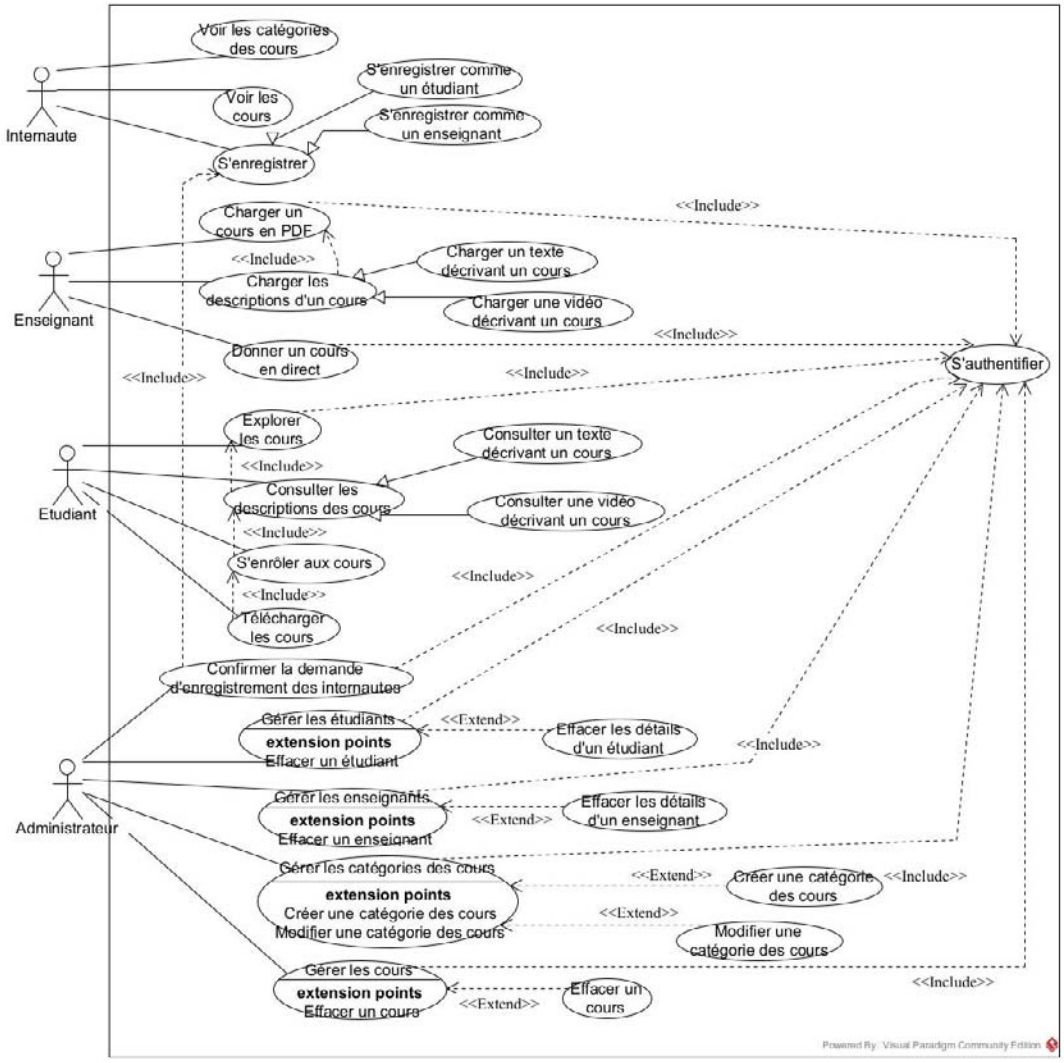


Figure 10 : Diagramme des cas d'utilisation du centre E-Learning

La figure 10 montre le diagramme des cas d'utilisation du centre avec les acteurs participant au fonctionnement du centre et leurs actions potentiels.

Description textuelle des cas d'utilisation

Cette partie montre une description textuelle des cas d'utilisation prioritaires :

1) Cas d'utilisation « S'authentifier » : un étudiant, enseignant et administrateur ont besoin de s'authentifier avant d'accéder à leurs pages dédiées. Le tableau 2 illustre le cas d'utilisation « S'authentifier ».

Tableau 2 : Description du cas d'utilisation « S'authentifier »

Acteurs	Étudiant, enseignant, administrateur
Pré conditions	1) L'affichage de la page d'accueil 2) L'ouverture de la page d'authentification
Post condition	Étudiant ou enseignant ou administrateur sont connectés au centre et peuvent réaliser les tâches qu'ils ont droit
Scénarios normaux	1) Le système affiche un formulaire d'identification 2) L'internaute remplit le formulaire en suivant un format prédéfini 3) Administratif accepte les données du formulaire 4) Acteurs concernés capable de s'authentifier
Scénario d'exception	Le système affiche un message d'erreur si les identifiants d'un acteur sont fausses

2) Cas d'utilisation « Charger un cours en PDF » : après s'être authentifié un enseignant peut charger un cours sous format PDF. Le tableau 3 montre la description du cas d'utilisation « Charger un cours en PDF ».

Tableau 3 : Description du cas d'utilisation « Charger un cours en PDF »

Acteur	Enseignant
Pré conditions	1) Un cours n'apparaît pas à la liste des cours (depuis la page d'accueil) 2) Un enseignant a accès à son compte
Post conditions	1) Affichage de la liste des cours dispensés par l'enseignant 2) La modification des cours par l'enseignant
Scénarios normaux	1) Le système affiche la page d'accueil 2) Authentification d'un enseignant 3) Vérification si le cours n'a pas déjà été chargé
Scénario d'exception	Le système affiche un message d'erreur si un enseignant charge un cours au format autre que PDF

3) Cas d'utilisation « Consulter les descriptions d'un cours » : un étudiant a le droit de consulter les descriptions tant textuelles que multimédias d'un cours avant de s'enrôler. Le tableau 4 montre le cas d'utilisation « Consulter les descriptions d'un cours »

Tableau 4 : Description du cas d'utilisation « Consulter les descriptions d'un cours »

Acteur	Étudiant
Pré conditions	1) Un enseignant doit avoir uploadé le cours et sa description 2) Authentification d'un étudiant
Post condition	Un étudiant s'enrôle ou pas
Scénarios normaux	1) Un internaute ouvre la page d'accueil du centre 2) Un étudiant s'authentifie 3) Un étudiant regarde une liste des cours et sélectionne un cours qu'il/elle souhaite

4) Cas d'utilisation « Télécharger les cours » : un étudiant après s'être enrôlé à un cours peut télécharger le cours. Le cinquième tableau illustre la description du cas d'utilisation « Télécharger les cours »

Tableau 5 : Description du cas d'utilisation « Télécharger les cours »

Acteur	Étudiant
Pré condition	Authentification d'un étudiant
Post condition	La lecture du livre du cours
Scénarios normaux	1) L'authentification d'un étudiant 2) L'étudiant consulte la description d'un cours 3) L'étudiant s'enrôle au cours

III.3.3. Diagramme d'activités

Le diagramme d'activités est essentiellement un organigramme pour représenter le flux d'une activité à une autre activité.

L'activité peut être décrite comme une opération du système. L'objectif fondamental des diagrammes d'activité est de capturer le comportement dynamique du système. Il est également appelé organigramme orienté objet.

Ce diagramme UML se concentre sur l'exécution et le flux du comportement d'un système au lieu de l'implémentation. Les diagrammes d'activités consistent en des activités composées d'actions qui s'appliquent à la technologie de modélisation comportementale [14].

Diagramme d'activités pour le chargement d'un nouveau cours

Montrons un diagramme d'activités décrivant comment un enseignant peut charger un cours.

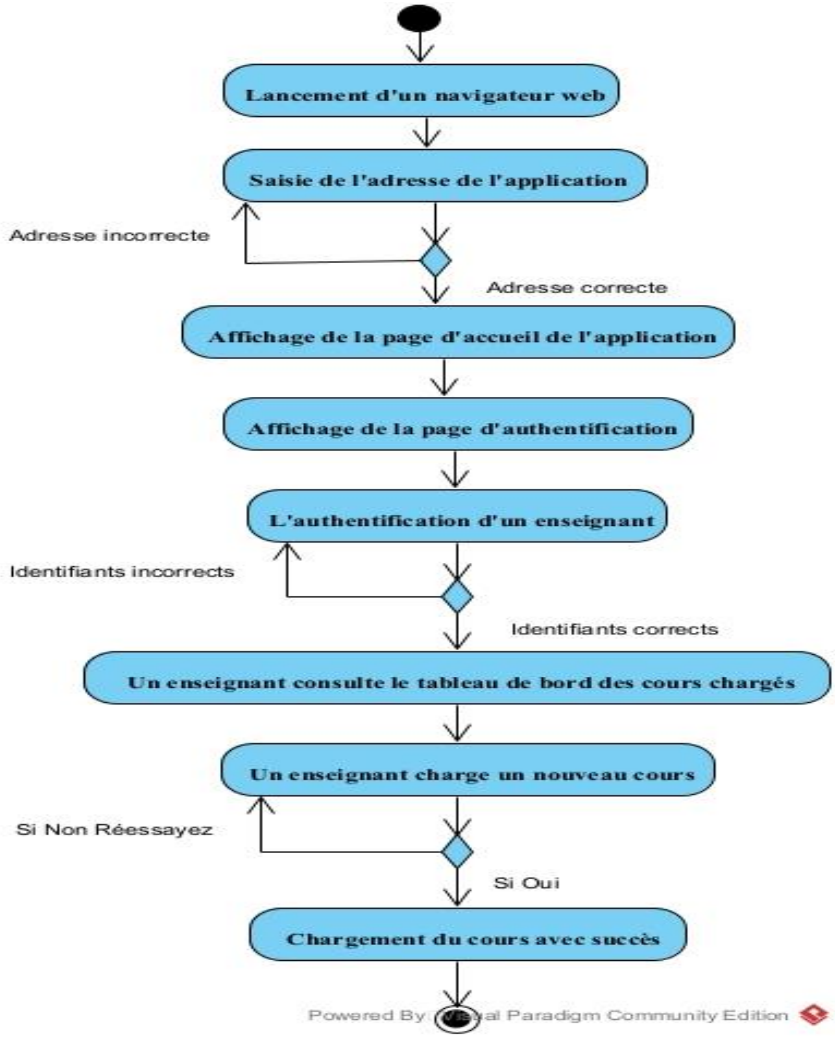


Figure 11: Diagramme d'activités pour le chargement d'un cours

La figure 11 illustre un diagramme d'activités pour le chargement d'un cours par un enseignant. Un enseignant après s'être authentifié a la possibilité de charger un nouveau cours ou regarder la liste des cours déjà chargés.

III.3.4. Diagramme de séquence

Le diagramme de séquence représente l'interaction entre les objets dans un ordre séquentiel. Le but d'un diagramme de séquence en UML est de visualiser la séquence d'un flux de messages dans le système. Le diagramme de séquence montre l'interaction entre deux lignes de vie sous la forme d'une séquence d'événements ordonnée dans le temps.

Un diagramme de séquence montre une implémentation d'un scénario dans le système. Les lignes de vie du système participent à l'exécution d'un système.

En résumé le diagramme de séquence est un diagramme d'interaction qui détaille comment les opérations sont effectuées - quels messages sont envoyés et quand.

Les diagrammes de séquence sont organisés en fonction du temps. Le temps avance au fur et à mesure que vous descendez la page [15]. Les objets impliqués dans l'opération sont listés de gauche à droite selon le moment où ils participent à la séquence de messages.

Dans un diagramme de séquence, une ligne de vie est représentée par une barre verticale. Un flux de messages entre deux ou plusieurs objets est représenté à l'aide d'une ligne pointillée verticale qui s'étend au bas de la page.

Dans un diagramme de séquence, différents types de messages et d'opérateurs sont utilisés. L'itération et le branchement sont également utilisés [16]. Les différents types de messages dans un diagramme de séquence sont :

1) Message synchrone : est utilisé lorsque l'expéditeur attend que le destinataire traite le message et répond avant de poursuivre avec un autre message ;
2) Message asynchrone : est utilisé lorsque l'appelant du message (l'expéditeur) n'attend pas que le destinataire traite le message et répond avant d'envoyer d'autres messages à d'autres objets du système ;
3) Message de retour : est utilisé pour indiquer que le destinataire du message a fini de traiter le message et rend le contrôle à l'appelant du message. Les messages de retour sont des éléments de notation facultatifs pour un message asynchrone mais une barre d'activation déclenchée par un message synchrone implique toujours un message de retour ;
4) Message de création de participant : Les objets ne vivent pas nécessairement pendant toute la durée de la séquence d'événements. Des objets ou des participants peuvent être créés en fonction du message envoyé. La notation de la boîte de participant supprimé peut être utilisée lorsque vous devez montrer que le participant particulier n'existait pas jusqu'à ce que l'appel de création ait été envoyé. Si le participant créé fait quelque chose immédiatement après sa création, vous devez ajouter une case d'activation juste en dessous de la case du participant ;
5) Message de destruction de participant : De même, lorsqu'ils ne sont plus nécessaires, les participants peuvent également être supprimés d'un diagramme de séquence. Cela se fait en ajoutant un « X » à la fin de la ligne de vie dudit participant ;

6) Message réflexif : Lorsqu'un objet s'envoie un message à lui-même, on parle de message réflexif. Il est indiqué par une flèche de message qui commence et se termine à la même ligne de vie [17].

Diagramme de séquence du cas d'utilisation « Télécharger les cours »

Après la lecture de la description textuelle et la vision d'une vidéo décrivant un cours, un étudiant peut s'enrôler au cours et télécharger un livre du cours. Montrons un diagramme de séquence décrivant comment un étudiant peut télécharger un cours.

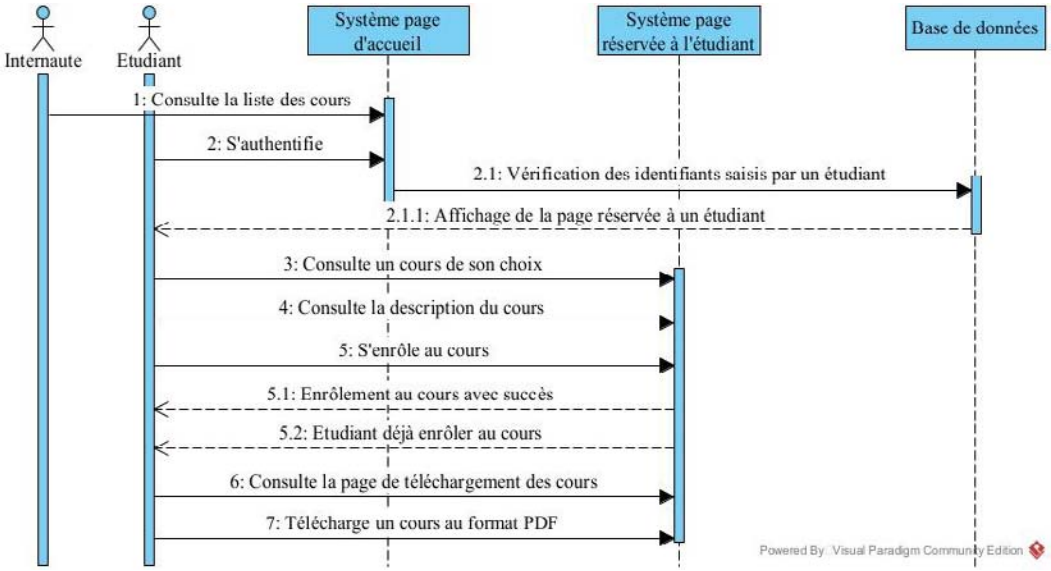


Figure 12: Diagramme de séquence « Télécharger un cours »

La figure 12 illustre un diagramme de séquence montrant les séquences prises par un étudiant pour télécharger un cours au format PDF.

III.3.5. Diagramme de classes

Un diagramme de classes est une structure statique qui donne un aperçu d'un système logiciel en affichant des classes, des attributs, des opérations et leurs relations entre eux.

Ce diagramme inclut le nom de la classe, les attributs et les opérations dans des compartiments désignés séparés. Le diagramme de classes aide à construire le code pour le développement d'applications logicielles.

Le diagramme de classes définit les types d'objets dans le système et les différents types de relations qui existent entre eux. Il donne une vue de haut niveau d'une application [18].

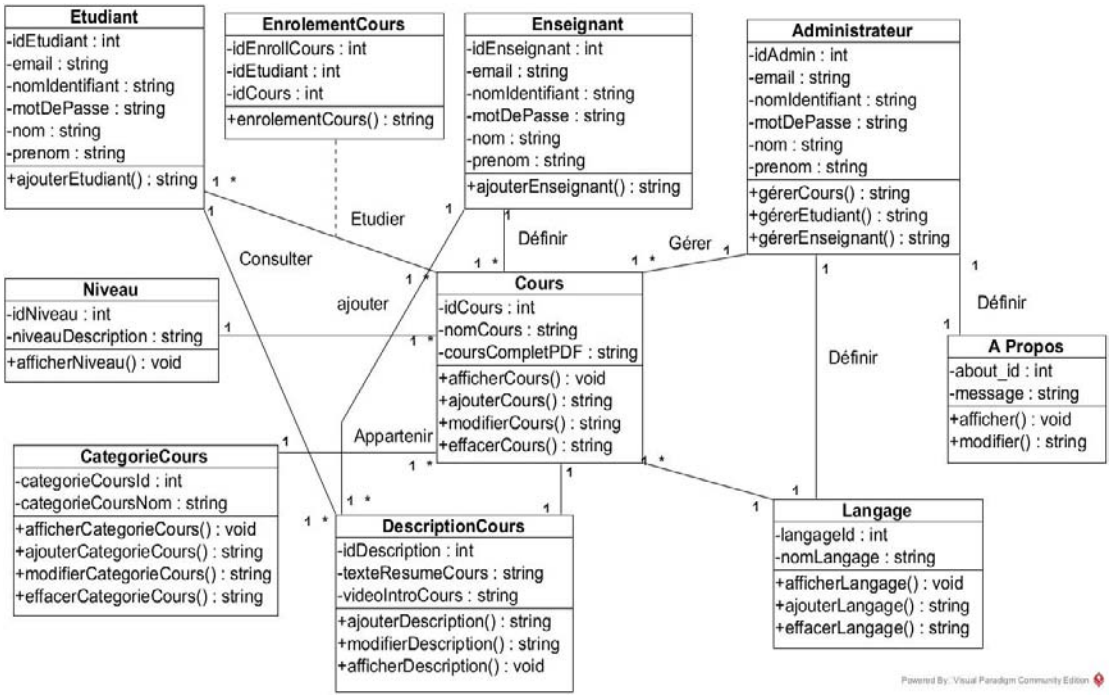


Figure 13: Diagramme de classes du centre

La figure 13 montre les classes (par exemple Etudiant, Enseignant, Niveau) en gras et leurs associations (par exemple Ajouter, Etudier, Consulter) dans le diagramme de classes de notre centre.

III.3.6. Diagramme de déploiement

Un diagramme de déploiement est un type de diagramme statique qui décrit la disposition physique des ressources matérielles qui composent le système et montre la répartition des composants logiciels sur ces matériels.

Le diagramme de déploiement précise comment les composants logiciels sont répartis sur les nœuds et quelles sont les connexions entre les composants ou les nœuds.

Les éléments du diagramme de déploiement sont :

1) Un nœud : représente une ressource matérielle sous forme d'un cube ;
2) Artefact : représente un composant logiciel avec la forme d'un rectangle ;
3) Environnement d'exécution : représente un nœud jouant le rôle d'un composant logiciel ;

4) Association de déploiement : représente une association entre un artefact et un nœud [19].

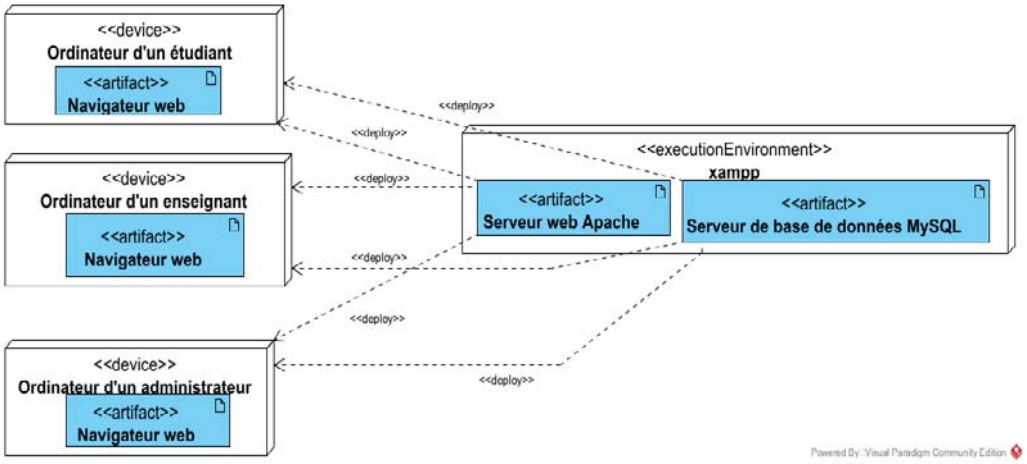


Figure 14: Diagramme de déploiement du centre

Le diagramme de déploiement de notre système est montré dans la figure 14. La figure décrit les équipements logiciels et les équipements physiques requis pour le fonctionnement du centre.

À l'issu du chapitre nous avons discuté à propos des modèles de cycle de vie d'un logiciel et avec l'aide du langage de modélisation unifié UML nous avons montré la conception du centre E-Learning. Le chapitre suivant détaille le modèle mathématique utilisé qu'est l'algèbre relationnelle.

Chapitre IV. Modélisation mathématique avec l'algèbre relationnelle

IV.1. Introduction

L'algèbre relationnelle, défini par Edgar Frank « Ted » Codd, est un langage de requête mathématique procédural pour les relations. Il collecte des instances de relations en entrée et donne des occurrences de relations en sortie. Il utilise diverses opérations pour effectuer cette action. Les opérations de requête d'algèbre relationnelle SQL sont effectuées de manière récursive sur une relation. La sortie de ces opérations est une nouvelle relation, qui peut être formée à partir d'une ou plusieurs relations d'entrée [20].

IV.2. Opérateurs relationnels et leurs traductions en SQL

Les opérateurs relationnels sont compartimentés en trois groupes.
Les opérateurs relationnels unaires qui acceptent une relation comme opérande et produisent une autre relation en sortie. Ces opérateurs sont la sélection ou la restriction, la projection et le renommage.
Les opérateurs relationnelles binaires acceptant deux relations ayant le même schéma comme opérandes et produisent une nouvelle relation en sortie. Ces opérateurs sont l'union, l'intersection, la différence et le produit cartésien
Les opérations relationnelles n-aires qui acceptent deux ou plusieurs relations comme opérandes et produisent une nouvelle relation en sortie. Ces opérateurs sont les types de jointure.
Explorons en détail le rôle de chaque opération évoquée ci-haut.

IV.2.1. Sélection

L'opération de sélection ou de restriction est utilisée pour sélectionner un sous-ensemble des tuples d'une relation R satisfaisant une condition de sélection donnée.
Il est représenté par $\sigma_p R$ où p représente la condition ou le critère de sélection et R représente une relation.

Exemple 1 : Soit la relation enseignant

idEnseignant	nom	prenom	email	username	grade
1	MUKESHIMANA	Michèle	michele@gmail.com	michele	Docteur
5	Ndikumagenge	Jérémie	jeremie@gmail.com	jeremie	Professeur
7	Jyoti	Kumar	jyoti@gmail.com	jyotiku	Docteur

Exprimons la requête qui montre les enseignants de grade de Docteur.

Requête en algèbre relationnelle $\sigma_{grade='Docteur'}enseignant$

Requête SQL : SELECT * FROM enseignant WHERE grade= 'Docteur' ;

Résultat de la requête :

idEnseignant	nom	prenom	email	username	grade
1	MUKESHIMANA	Michèle	michele@gmail.com	michele	Docteur
7	Jyoti	Kumar	jyoti@gmail.com	jyotiku	Docteur

Exemple 2 : Montrons un enseignant de grade de Docteur qui porte le nom de Jyoti.

Requête en algèbre relationnelle : $\sigma_{grade='Docteur' and\ nom='Jyoti'}enseignant$

Requête SQL : SELECT * FROM enseignant WHERE grade='Docteur' and nom='Jyoti';

Résultat de la requête :

idEnseignant	nom	prenom	email	username	grade
7	Jyoti	Kumar	jyoti@gmail.com	jyotiku	Docteur

IV.2.2. Projection

La projection $\pi_{A1,A2,\ldots,Ak}R$ s'applique à une relation R et ne garde que les attributs A_1, A_2, …, A_k de la relation R. Contrairement à la restriction, on ne supprime pas les tuples mais des attributs. Cela aide à extraire les valeurs des attributs spécifiés pour éliminer les valeurs en double. La projection est symbolisée par π.

Exemple 1: Soit la relation etudiant

idEtudiant	nom	prenom	email	username
1	Niyonzima	Lary	laryniyo@gmail.com	lary123
2	Arakaza	Axel	arakaza@gmail.com	arakaza
3	Bayubahe	Edouard	bayubahe@gmail.com	bayubahe
4	Kakunze	Gérard	kakunze@gmail.com	kakunze
5	dusabe	guillaume	guillaume@gmail.com	dusabegui

On veut afficher les noms et prénoms des étudiants.

Requête en algèbre relationnelle : $\pi_{nom,prenom} etudiant$

Requête SQL : SELECT DISTINCT nom, prenom FROM etudiant ;

Résultat de la requête :

nom	prenom
Niyonzima	Lary
Arakaza	Axel
Bayubahe	Edouard
Kakunze	Gérard
dusabe	guillaume

Exemple 2 : Affichons le nom et prénom d'un étudiant dont l'adresse électronique est laryniyo@gmail.com

Requête en algèbre relationnelle : $\pi_{nom,prenom}(\sigma_{email='laryniyo@gmail.com'} etudiant)$

Requête SQL: SELECT DISTINCT nom, prenom, email FROM etudiant where email='laryniyo@gmail.com';

Résultat de la requête :

nom	prenom	email
Niyonzima	Lary	laryniyo@gmail.com

IV.2.3. Renommage

Le renommage est une opération unaire utilisée pour renommer les attributs d'une relation. Il est représenté par le signe ρ. $\rho_{a/b}R$ renomme l'attribut 'a' de la relation R par 'b'.

Exemple : Avec la relation cours renommons l'attribut coursCompletPDF par coursPDF

nom_cours	coursCompletPDF
Gestion des projets	GESTION DE PROJECTS 2019-20.pdf
Méthodologie de recherche	Methodologie de recherche.pdf
Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...

Requête en algèbre relationnelle : $\rho_{coursCompletPDF/coursPDF}cours$

Requête SQL : ALTER TABLE cours CHANGE coursCompletPDF coursPDF varchar(255);

Résultat de la requête :

nom_cours	coursPDF
Gestion des projets	GESTION DE PROJETS 2019-20.pdf
Méthodologie de recherche	Methodologie de recherche.pdf
Problem Solving Programming Concepts	Problem Solving, Programming concepts 9th ed(intro...

IV.2.4. Union

L'union de deux relations A et B définit une nouvelle relation incluant tous les tuples qui se trouvent dans les deux relations en éliminant les tuples en double. Il est symbolisé par le symbole ∪.

Pour qu'une opération d'union soit valide, les conditions suivantes doivent être remplies :

1) Les deux relations A et B doivent avoir le même schéma, c'est à dire le même nombre d'attributs, mêmes noms et mêmes types.
2) Les tuples en double doivent être supprimés.

Exemple : Avec la relation etudiant

id	nom	prenom	email	username
1	Niyonzima	Lary	laryniyo@gmail.com	lary123
2	Arakaza	Axel	arakaza@gmail.com	arakaza
3	Bayubahe	Edouard	bayubahe@gmail.com	bayubahe
4	Kakunze	Gérard	kakunze@gmail.com	kakunze
5	Dusabe	Guillaume	guillaume@gmail.com	dusabegui

Et enseignant

id	nom	prenom	email	username
1	MUKESHIMANA	Michèle	michele@gmail.com	michele
2	Ndikumagenge	Jérémie	jeremie@gmail.com	jeremie
3	Jyoti	Kumar	jyoti@gmail.com	jyotiku

Affichons l'union des deux relations.

Requête en algèbre relationnelle : $etudiant \cup enseignant$

Requête SQL: SELECT * FROM etudiant union select * from enseignant;

Résultat de la requête :

id	nom	prenom	email	username
1	Niyonzima	Lary	laryniyo@gmail.com	lary123
2	Arakaza	Axel	arakaza@gmail.com	arakaza
3	Bayubahe	Edouard	bayubahe@gmail.com	bayubahe
4	Kakunze	Gérard	kakunze@gmail.com	kakunze
5	Dusabe	Guillaume	guillaume@gmail.com	dusabegui
1	MUKESHIMANA	Michèle	michele@gmail.com	michele
2	Ndikumagenge	Jérémie	jeremie@gmail.com	jeremie
3	Jyoti	Kumar	jyoti@gmail.com	jyotiku

IV.2.5. Intersection

L'intersection de deux relations A et B définit une nouvelle relation consistant en un ensemble de tous les tuples qui sont à la fois dans les relations A et B. Les conditions définies dans l'union sont valides pour l'intersection. Il est symbolisé par ∩.

Exemple : avec les relations etudiant et enseignant affichons l'intersection des deux relations.

Requête en algèbre relationnelle **:** $enseignant \cap etudiant$

Requête SQL: SELECT * FROM enseignant WHERE (id,nom,prenom,email,username) IN (SELECT * FROM etudiant);

Résultat de la requête :

id	nom	prenom	email	username

Le résultat de la requête est nul car il n'y a pas de tuples en commun entre les relations enseignant et etudiant.

IV.2.6. Différence

La différence de deux relations A et B définit une nouvelle relation qui inclut tous les tuples qui sont dans la relation A mais pas dans la relation B. Les conditions définies dans l'union sont valides pour la différence. La différence est symbolisée par le signe −

Exemple : avec les relations etudiant et enseignant affichons les occurrences qui sont dans la relation enseignant qui ne sont pas dans la relation etudiant.

Requête en algèbre relationnelle : $enseignant - etudiant$

Requête SQL: SELECT * FROM enseignant WHERE (id,nom,prenom,email,username) NOT IN (SELECT * FROM etudiant);

Résultat de la requête :

id	nom	prenom	email	username
1	MUKESHIMANA	Michèle	michele@gmail.com	michele
2	Ndikumagenge	Jérémie	jeremie@gmail.com	jeremie
3	Jyoti	Kumar	jyoti@gmail.com	jyotiku

IV.2.7. Produit cartésien

Le produit cartésien entre deux relations R1 et R2 définit une nouvelle relation qui est composée de toutes les combinaisons possibles entre les tuples de R1 et ceux de R2 et ayant pour schéma la juxtaposition de ceux des relations R1 et R2. Le produit cartésien est représenté par le signe × [21].

Exemple : Avec la relation cat (pour catégorie)

cat_id	cat_name
1	Projets technologiques
2	Connaissances générales

Et la relation cours

id_cours	nom_cours	coursCompletPDF
2	Gestion des projets	GESTION DE PROJECTS 2019-20.pdf
4	Méthodologie de recherche	Methodologie de recherche.pdf
5	Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...

Requête en algèbre relationnelle : $cat \times cours$

Requête SQL: SELECT * FROM cat CROSS JOIN cours;

Résultat de la requête :

cat_id	cat_name	id_cours	nom_cours	coursCompletPDF
1	Projets technologiques	1	Gestion des projets	GESTION DE PROJETS 2019-20.pdf
2	Connaissances générales	1	Gestion des projets	GESTION DE PROJETS 2019-20.pdf
1	Projets technologiques	2	Méthodologie de recherche	Methodologie de recherche.pdf
2	Connaissances générales	2	Méthodologie de recherche	Methodologie de recherche.pdf
1	Projets technologiques	3	Problem Solving Programming Concepts	Problem Solving, Programming concepts 9th ed(intro...
2	Connaissances générales	3	Problem Solving Programming Concepts	Problem Solving, Programming concepts 9th ed(intro...

Le produit cartésien est rarement utilisé à cause de son inefficacité lorsque nous avons deux relations ayant un nombre élevé de tuples. Le problème est résolu avec l'opération de jointure.

IV.2.8. Jointure

L'opération de jointure permet de joindre des tuples de deux ou plusieurs relations satisfaisant une condition. L'opération de jointure est essentiellement un produit cartésien suivi d'une sélection.

Il existe plusieurs types de jointure et elles sont regroupées en deux catégories : jointures internes et jointures externes.

Explorons en détail les différents types de jointure.

Jointure interne

Dans une jointure interne seules les tuples qui satisfont un critère de sélection sont inclus dans la relation résultante, les autres sont exclus. Les opérations de la jointure interne sont la thêta-jointure, l'équijointure et la jointure naturelle.

Explorons en détail les types de jointure interne.

1) Une thêta-jointure est une jointure qui utilise plusieurs types de conditions dans le critère de sélection. Elle est dénotée par $\sigma_E(A \times B)$ ou par $A_E \bowtie B$, où E représente la condition à satisfaire.

Exemple : avec les relations cat (pour catégorie) et cours affichons les noms des catégories des cours, du matériel PDF des cours si l'identifiant de la relation cat cat_id est inférieur à la clé étrangère cat_id de la relation cours.

Requête en algèbre relationnelle : $\sigma_{cat.catid<cours.catid} cat \times cours$

Requête SQL: SELECT cat_name, nom_cours, coursCompletPDF FROM cat, cours WHERE cat.cat_id< cours.cat_id;

Résultat de la requête :

cat_name	nom_cours	coursCompletPDF
Projets technologiques	Méthodologie de recherche	Methodologie de recherche.pdf
Projets technologiques	Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...

2) L'équijointure est une thêta-jointure qui utilise uniquement une condition d'égalité.

Exemple : Avec les relations cat (pour catégorie) et cours on veut afficher les noms des catégories, des cours, du matériel PDF des cours si l'identifiant de la relation cat cat_id est égal à la clé étrangère cat_id de la relation cours.

Requête en algèbre relationnelle : $\sigma_{cat.catid=cours.catid} cat \times cours$

Requête SQL: SELECT cat_name, nom_cours, coursCompletPDF FROM cat, cours WHERE cat.cat_id= cours.cat_id;

Résultat de la requête :

cat_name	nom_cours	coursCompletPDF
Projets technologiques	Gestion des projets	GESTION DE PROJECTS 2019-20.pdf
Connaissances générales	Méthodologie de recherche	Methodologie de recherche.pdf
Connaissances générales	Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...

3) La jointure naturelle ne peut être effectuée que s'il existe un attribut commun entre les relations à joindre. Le nom et le type de l'attribut doivent être identiques.

Exemple 1 : avec les relations cat (pour catégorie) et cours

Requête en algèbre relationnelle : $cat \bowtie cours$

Requête SQL : SELECT * FROM cat natural join cours;

Requête SQL équivalente : SELECT * FROM cat JOIN cours ON cat.cat_id=cours.cat_id;

Résultat de la requête :

cat_id	cat_name	id_cours	nom_cours	coursCompletPDF
1	Projets technologiques	2	Gestion des projets	GESTION DE PROJECTS 2019-20.pdf
2	Connaissances générales	4	Méthodologie de recherche	Methodologie de recherche.pdf
2	Connaissances générales	5	Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...

Exemple 2 : Combinaison de la jointure naturelle avec une condition.

Requête en algèbre relationnelle : $cat \bowtie_{cat.catid<2} cours$

Requête SQL: SELECT cat_name,nom_cours,coursCompletPDF FROM cat JOIN cours ON cat.cat_id=cours.cat_id WHERE cat.cat_id<2;

Résultat de la requête:

cat_name	nom_cours	coursCompletPDF
Projets technologiques	Gestion des projets	GESTION DE PROJECTS 2019-20.pdf

Jointure externe

Dans une jointure externe, en plus des tuples qui satisfont le critère de sélection, nous incluons également certains ou tous les tuples qui ne correspondent pas au critère. Les opérations de la jointure externe sont la jointure externe entière, la jointure externe gauche et la jointure externe droite.

Explorons en détail les types de jointure externe.

1) La jointure externe gauche est utilisée pour joindre deux ou plusieurs tables et renvoie toutes les lignes de la table de gauche et uniquement les lignes de l'autre table qui ont rempli la condition de jointure. Si aucun tuple correspondant n'est trouvé dans la relation de droite, les attributs de la relation de droite dans le résultat de la jointure sont remplis avec des valeurs nulles. La jointure externe gauche entre deux relations A et B est symbolisée par A⟕B

Exemple : avec les relations cat (pour catégorie) et cours

Requête en algèbre relationnelle : cat⟕cours

Requête SQL: SELECT * FROM cat LEFT JOIN cours ON cat.cat_id=cours.cat_id;

Résultat de la requête :

cat_id	cat_name	id_cours	nom_cours	coursCompletPDF	cat_id
1	Projets technologiques	2	Gestion des projets	GESTION DE PROJECTS 2019-20.pdf	1
2	Connaissances générales	4	Méthodologie de recherche	Methodologie de recherche.pdf	2
2	Connaissances générales	5	Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...	2
3	Réseaux avancés	NULL	NULL	NULL	NULL

2) La jointure externe droite se comporte comme la jointure externe gauche mais le rôle des relations est interchangé. La jointure externe droite est utilisée pour joindre deux ou plusieurs tables et renvoie toutes les lignes de la table de droite et uniquement les résultats de l'autre table qui ont rempli la condition de jointure. Si aucun tuple correspondant n'est trouvé dans la relation de gauche, les attributs de la relation de gauche dans le résultat de la jointure sont remplis avec des valeurs nulles. La jointure externe droite entre deux relations A et B est symbolisée par A⋈B

Requête en algèbre relationnelle : cat⋈cours

Requête SQL: SELECT * FROM cat RIGHT JOIN cours ON cat.cat_id=cours.cat_id;

Résultat de la requête:

cat_id	cat_name	id_cours	nom_cours	coursCompletPDF	cat_id
1	Projets technologiques	2	Gestion des projets	GESTION DE PROJECTS 2019-20.pdf	1
2	Connaissances générales	4	Méthodologie de recherche	Methodologie de recherche.pdf	2
2	Connaissances générales	5	Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...	2

3) Dans une jointure externe entière, tous les tuples des relations sont inclus dans le résultat de jointure. La jointure externe entière est l'union de la jointure externe gauche et la jointure externe droite. La jointure externe entière est symbolisée par A⟗B

Requête en algèbre relationnelle : cat⟗cours

Requête SQL: SELECT cat.cat_id,cat_name,id_cours,nom_cours,coursCompletPDF FROM cat LEFT JOIN cours ON cat.cat_id=cours.cat_id UNION SELECT cat.cat_id,cat_name,id_cours,nom_cours,coursCompletPDF FROM cat RIGHT JOIN cours ON cat.cat_id=cours.cat_id;

Résultat de la requête :

cat_id	cat_name	id_cours	nom_cours	coursCompletPDF
1	Projets technologiques	2	Gestion des projets	GESTION DE PROJECTS 2019-20.pdf
2	Connaissances générales	4	Méthodologie de recherche	Methodologie de recherche.pdf
2	Connaissances générales	5	Problem Solving Programming Concepts	Problem Solving, Programming Concepts 9th ed (intr...
3	Réseaux avancés	NULL	NULL	NULL

Dans ce chapitre nous avons montré l'applicabilité du modèle mathématique de l'algèbre relationnelle dans la gestion de notre centre E-Learning.

Dans le chapitre suivant nous montrons à la base de la conception de notre système expliquée dans le troisième chapitre l'implémentation de notre centre.

Chapitre V. Implémentation et présentation des fonctionnalités du centre

V.1. Introduction

En se basant sur les résultats obtenus lors des chapitres précédents dans ce chapitre nous montrons l'architecture logicielle utilisée et les outils employés pour mettre en place notre système. Nous présentons aussi les aspects de la sécurité de notre système et les interfaces graphiques montrant les grandes fonctionnalités du centre E-Learning de l'Université du Burundi.

V.2. Architecture client serveur

Dans le contexte informatique, le client est un ordinateur ou tout autre appareil, également appelé hôte, qui utilise réellement le service ou accepte les informations. Les appareils clients comprennent les ordinateurs portables, les stations de travail, les appareils IoT (Internet of Things) et les appareils similaires compatibles avec le réseau.

Un serveur est un ordinateur distant qui donne accès aux données et aux services. Les serveurs sont généralement des appareils physiques tels que des serveurs en rack, bien que l'essor du cloud computing ait introduit les serveurs virtuels dans l'équation. Le serveur gère des processus tels que le courrier électronique, l'hébergement d'applications, les connexions Internet, l'impression, etc.

L'architecture client-serveur, également appelée modèle client-serveur, est une application réseau qui répartit les tâches et les charges de travail entre les clients et les serveurs qui résident sur le même système ou sont reliés par un réseau informatique.

L'architecture client-serveur comprend généralement des postes de travail, des PC ou d'autres appareils de plusieurs utilisateurs, connectés à un serveur central via une connexion Internet ou un autre réseau. Le client envoie une demande de données, et le serveur accepte et prend en charge la demande, renvoyant les paquets de données à l'utilisateur qui en a besoin.

Les réseaux Peer-to-Peer, également appelés réseaux P2P, consistent en des groupes d'ordinateurs (également appelés nœuds ou pairs) reliés entre eux dans un réseau, où les pairs agissent à la fois comme client et comme serveur.

Les pairs ont des responsabilités et des autorisations égales pour travailler avec les données. Cette configuration diffère radicalement du modèle client-serveur, qui a des groupes d'utilisateurs et de serveurs bien définis.

Les principales différences entre les deux modèles de réseau sont:

1) Les réseaux client-serveur ont besoin d'un serveur de fichiers central et coûtent par conséquent plus cher à mettre en œuvre. Peer-to-Peer n'a pas ce serveur ;
2) Les réseaux client-serveur délimitent les utilisateurs et les fournisseurs ; les pairs agissent à la fois comme consommateurs et fournisseurs ;
3) Les réseaux client-serveur offrent plus de niveaux de sécurité, ce qui les rend plus sûrs. Les utilisateurs finaux sont responsables de la sécurité du réseau Peer-to-Peer ;
4) Les réseaux client-serveur offrent une meilleure stabilité et évolutivité. La plage idéale pour les réseaux P2P est de deux à huit utilisateurs ;
5) Les utilisateurs Peer-to-Peer peuvent partager des fichiers plus rapidement et plus facilement que sur un réseau client-serveur ;
6) Si un serveur de réseau client-serveur tombe en panne, tout s'arrête, mais si un seul nœud d'un réseau P2P tombe en panne, le reste est opérationnel [22].

V.3. Outils d'implémentation utilisés

Dans cette partie nous expliquons le rôle des différents outils d'implémentation utilisés pour aboutir à nos objectifs.

V.3.1. Editeur texte

Un éditeur texte est un programme informatique qui permet aux utilisateurs de créer, de modifier ou d'éditer des fichiers en texte brut. Il peut être utilisé pour créer des programmes informatiques, modifier le code source des langages de programmation, modifier le langage de balisage hypertexte (HTML) et de créer des pages web ou des modèles de conception web [23] .Notepad++ est un éditeur de texte amélioré (format txt), qui intègre aussi l'écriture de code source. Il est basé sur le composant Scintilla, et il est disponible sous l'environnement Microsoft Windows.

Cet outil gratuit et open source est un incontournable auprès des développeurs débutants comme expérimentés. Le plus est qu'il a une interface graphique simple et classique pour gagner en productivité sur vos projets techniques [24].

V.3.2. Serveur web

Le serveur web est un logiciel géré par votre fournisseur d'hébergement de site web afin que les visiteurs puissent voir les pages web de votre site.
Le logiciel joue un rôle similaire à celui d'un hôte de restaurant.
Lorsque vous arrivez dans un restaurant, l'hôte vous accueille, vérifie vos informations de réservation et vous emmène à votre table. De la même manière, le serveur web recherche la page web que vous avez demandée.
Un serveur web fait plus. Il agit également comme un serveur de restaurant lorsqu'il récupère la page et la sert pour votre plus grand plaisir.
Et il joue le rôle de maître d'hôtel du restaurant en gérant les communications avec le site web, en traitant vos demandes et en s'assurant que d'autres modules sont prêts à vous servir.
Enfin, un serveur web nettoie la mémoire, le cache, les modules et les efface pour les nouveaux visiteurs du site web [25].
Le serveur web employé est Apache, un logiciel de serveur web gratuit et open source. Le nom officiel est Serveur Apache HTTP et il est maintenu et développé par Apache Software Foundation [26].

V.3.3. Serveur de base de données

Un serveur de base de données sert à stocker, à extraire et à gérer les données dans une base de données. Il permet également de gérer la mise à jour des données. Il donne un accès simultané à cette base à plusieurs serveurs Web et utilisateurs.
Enfin, il assure la sécurité et l'intégrité des données. Et quand on parle de données, on entend des millions d'éléments simultanément accessibles à des milliers d'utilisateurs.
Nous avons opté pour MySQL qui est la base de données la plus populaire en hébergement Web. Il s'agit d'une option puissante, tout en étant gratuite (source libre), conçue pour fonctionner avec le langage de programmation PHP [27].

V.3.4. PHP Data Objects

PHP Data Objects, aussi connu sous l'acronyme PDO, est une interface pour l'accès aux bases de données en PHP sans avoir à écrire un script propre à une base de données spécifique. Plutôt que d'appeler directement les fonctions mysql_, mysqli_ et pg_ (pour PostgreSQL), les développeurs peuvent utiliser l'interface PDO, simplifiant le portage des applications vers d'autres bases de données.

L'extension PHP Data Objects est actuellement incluse par défaut dans les installations de PHP 5.1. Il est disponible pour les utilisateurs de PHP 5.0 via PECL (PHP Extension Community Library).

PDO utilise des fonctionnalités de PHP initialement introduites dans PHP 5.0. Il n'est donc pas disponible pour les utilisateurs de PHP 4.x et inférieur.

PHP Data Objects présente un certain nombre de différences significatives par rapport à l'interface MySQL utilisée par la plupart des applications PHP sur PHP 4.x et inférieur.

Ces différences sont :

1) Orientation objet : alors que l'extension mysql utilisait un certain nombre d'appels de fonctions qui fonctionnaient sur une poignée de connexion et des séries de résultat, l'extension PDO a une interface orientée objet ;
2) Indépendance de la base de données : l'extension PDO, contrairement à l'extension mysql, est conçue pour être compatible avec plusieurs bases de données avec peu d'effort de la part de l'utilisateur, à condition que le SQL soit utilisé dans toutes les requêtes ;
3) Les connexions à la base de données sont établies avec un nom de source de données (de l'anglais Data Source Name ou DSN). Un DSN est une chaîne qui contient toutes les informations nécessaires pour se connecter à une base de données, telles que le nom du système de gestion de base de données ainsi que le nom d'une base de données 'mysql:dbname=test_db' [28].

Les systèmes prenant en charge PDO sont MS SQL, MySQL, Oracle, PostgreSQL, SQLite, CUBRID, Firebird, Informix, 4D [29].

V.3.5. Langage de balisage

Un langage de balisage est un langage servant à définir et à structurer les informations contenues dans un document.

Les langages de balisages sont aussi appelés langages de description car ils ont pour but de décrire la structure des documents décrits.

On parle par exemple de langage de description de page lorsqu'il s'agit de décrire la structure d'une page d'un site Internet (titre, paragraphes, sauts de ligne, passages en gras ou en italique, etc.).

Le langage de description de page le plus connu et lequel nous avons utilisé est le HTML qui est amené à être remplacé par le XHTML.

Pour créer des pages web à l'apparence soignée nous avons utilisé les feuilles de style en cascade, de l'anglais Cascading Style Sheets (CSS) [30].

V.3.6. Langage de programmation des scripts

JavaScript (souvent abrégé en « JS ») inventé par Brendan Eich en 1995 est un langage de script léger, orienté objet.

Il est principalement connu comme le langage de script des pages web. Avec les langages HTML et CSS, JavaScript est au cœur des langages utilisés par les développeurs web [31].

Nous avons aussi utilisé une bibliothèque JavaScript, dénommée jQuery, open-source inventée par John Resig en 2006. Insérer du JavaScript pur dans des projets web peut changer le comportement des pages. Ce langage vient compléter le HTML et le CSS de votre site, qui déterminent le contenu et l'apparence de votre page, en y ajoutant une couche d'interactivité.

JQuery (la bibliothèque JavaScript open-source) nous permet d'exploiter la puissance de JavaScript pour accomplir une myriade de choses sur votre page web. Vous pouvez par exemple ajouter, supprimer ou modifier des éléments HTML au sein de votre page ; changer les styles des éléments de la page en modifiant le CSS qui leur est associé ; animer des éléments de votre page ; envoyer et recevoir des données depuis un serveur grâce à AJAX (Asynchronous JavaScript and XML, c'est-à-dire JavaScript et XML asynchrones) pour ne plus avoir besoin de recharger vos pages après validation d'un formulaire, et bien plus encore [32].

V.3.7. Langage de script côté serveur

Un langage de script côté serveur est un langage de programmation mis en œuvre sur un serveur http pour produire une page Web dynamique. Les langages de script côté serveur sont nombreux. On peut en citer PHP, Java, Python, C#.

Nous avons utilisé PHP car PHP est un langage de scripts généraliste et Open Source, spécialement conçu pour le développement d'applications web. Il peut être intégré facilement au HTML [33].

V.3.8. FPDF

FPDF est une classe PHP gratuite qui permet de générer des fichiers PDF directement depuis PHP sans dépendre d'une librairie extérieure. Comme FPDF est gratuite nous pouvons l'utiliser pour tout type d'utilisation et le modifier selon nos besoins [34].

V.3.9. WebRTC

Le WebRTC, qui signifie Web Real-Time Communications (Communications Web en temps réel) est un projet open source présenté par Google, Mozilla et d'autres qui permet des communications en temps réel sans plugin via une API JavaScript [35].

Avec WebRTC vous pouvez ajouter des capacités de communication en temps réel à votre application. Il prend en charge la vidéo, la voix et les données à envoyer entre pairs, permettant aux développeurs de créer de puissantes solutions de communication vocale et vidéo. La technologie est disponible sur tous les navigateurs web [36].

V.4. Sécurité informatique et son rôle dans E-Learning

V.4.1. Introduction

La sécurité informatique est la protection de l'information et plus particulièrement le traitement de l'information. La sécurité informatique vise à empêcher la manipulation des données et des systèmes par des tiers non autorisés.

En cryptographie, les attaques sont de deux types : les attaques passives et les attaques actives. Les attaques passives sont celles qui récupèrent les informations du système sans affecter les ressources du système, tandis que les attaques actives sont celles qui récupèrent les informations du système et apportent des modifications aux ressources du système et à leurs opérations.

V.4.2. Objectifs de protection informatique

La sécurité des informations est définie par les trois objectifs de protection informatique que sont la disponibilité, l'intégrité et la confidentialité. Ceux-ci doivent être entretenus. En outre, d'autres éléments peuvent être ajoutés : authenticité et non-répudiation.

La confidentialité de la sécurité informatique signifie que les données ne sont accessibles qu'à des personnes autorisées. En d'autres termes, la protection d'accès doit être définie.

Un autre point central de la confidentialité de l'information est le transport des données. Celui-ci doit toujours être crypté, symétriquement ou asymétriquement. Cela signifie que les personnes non autorisées ne peuvent pas accéder au contenu [37].

Pour garantir le transport des données nous avons utilisé l'algorithme de chiffrement symétrique dénommé Advanced Encryption Standard ou AES. AES est un chiffrement par bloc. La taille de la clé peut être de 128/192/256 bits, AES-128 crypte les données en blocs de 128 bits chacun. Cela signifie qu'il prend 128 bits en entrée et produit 128 bits de texte chiffré en sortie. AES repose sur le principe du réseau de substitution-permutation, ce qui signifie qu'il est effectué à l'aide d'une série d'opérations liées qui impliquent le remplacement et le mélange des données d'entrée [38].

L'intégrité donne l'assurance que les informations reçues sont exactes. Si le contenu du message est modifié après que l'expéditeur l'a envoyé mais avant d'atteindre le destinataire prévu, on dit que l'intégrité du message est perdue.

La disponibilité garantit que les personnes disposant de l'autorisation appropriée peuvent utiliser les systèmes et récupérer des données de manière fiable et rapide. Cela garantit que les systèmes d'information sont fiables et accessibles.

L'authentification est le mécanisme permettant d'identifier l'utilisateur, le système ou l'entité. Il garantit l'identité de la personne qui tente d'accéder à l'information. L'authentification est principalement sécurisée à l'aide d'un nom d'utilisateur et d'un mot de passe. La personne autorisée dont l'identité est préenregistrée peut prouver son identité et accéder aux informations sensibles.

La non-répudiation est un mécanisme qui empêche le refus du contenu d'un message envoyé via un réseau. Dans certains cas, l'expéditeur envoie le message et le nie plus tard.

Mais la non-répudiation ne permet pas à l'expéditeur de refuser l'envoi d'un message au destinataire [39]. La non-répudiation garantit qu'aucune partie ne peut nier avoir envoyé ou reçu un message.

V.4.3. Stockage des mots de passe

Avec l'augmentation des cas de vol d'informations d'identification et de vidage de données sur le Dark Web, le stockage sécurisé d'informations d'identification telles que les mots de passe est devenu beaucoup plus important.

En cas de violation de la sécurité, il est important de s'assurer que les attaquants soient incapables de décoder les mots de passe et de causer des dommages supplémentaires. Il existe plusieurs manières de stocker des mots de passe dans une base de données en toute sécurité, en utilisant un algorithme de hachage.

L'algorithme de hachage le plus courant est MD5 et c'est le plus rapide. Cependant, il est facile et rapide de craquer. Vous pouvez utiliser la fonction password_hash au lieu de MD5, car elle est plus sécurisée. Cette fonction génère un hachage de soixante caractères basé sur BCRYPT (algorithme CRYPT_BLOWFISH). Il accepte trois paramètres: le mot de passe, l'algorithme de hachage (utilise par défaut BCRYPT) et une valeur facultative telle que le coût. Vous pouvez vérifier le mot de passe haché en utilisant la fonction password_verify [40].

Un exemple d'une relation où les mots de passe sont hachées avec la fonction password_hash et les noms des utilisateurs cryptées avec l'algorithme de chiffrement symétrique AES est montré ci-dessous.

username	password
Szc0QUxlWDNlZUgyUDBmS0tqeERzUT09OjoFFDPBolwiMQQeJg...	$2y$10$TesO3oFzc3nukrf7y3DSy.xlWRQg.w/hQgQsGwcGbE6...
RFk3Y1BYUFI4R1RzM2ZwOHR5dW1xUT09Ojr4AcnWbYaFw0zaSw...	$2y$10$GEazrz8pVT1ThHrpzYkMRu7RtqclO1eSAiE7kT5aicf...

V.5. Présentation des fonctionnalités du centre E-Learning

Dans cette section nous allons présenter les grandes fonctionnalités du centre E-Learning de l'Université du Burundi.

V.5.1. Page d'accueil

La page d'accueil présente une vue synthétique des fonctionnalités du système et est accessible à n'importe qui le veut.

Elle montre les différentes catégories des cours et les cours. Un anonyme peut chercher les cours en écrivant des mots clés dans le formulaire de recherche. Avec cette recherche un visiteur a accès aux noms des cours et des enseignants des cours.

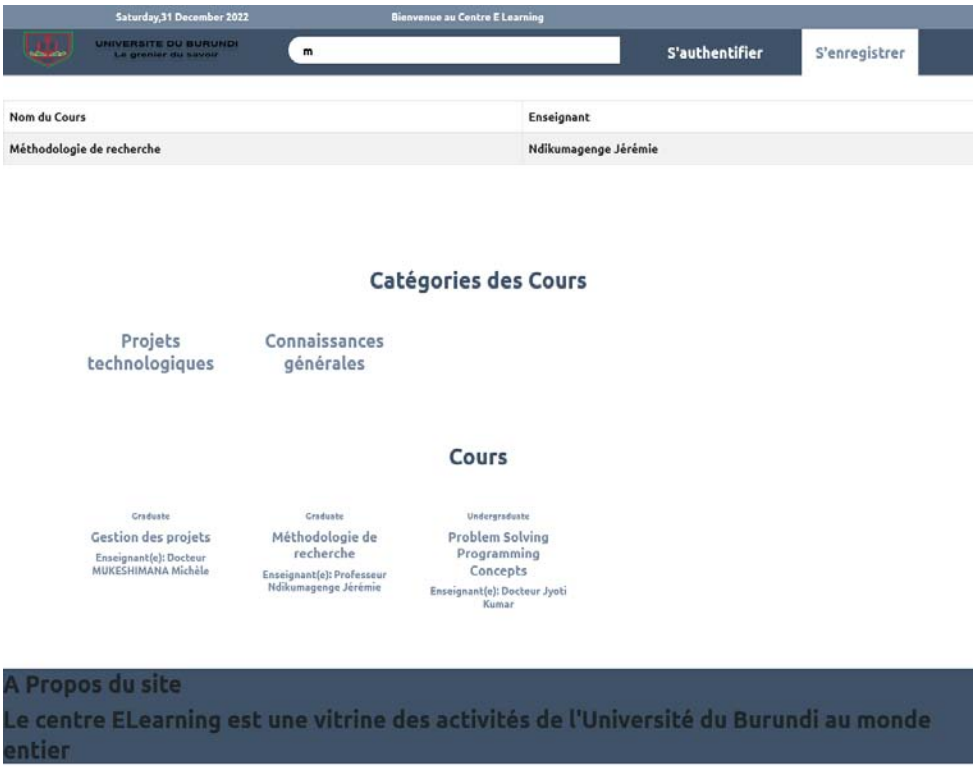


Figure 15: Page d'accueil du centre

La figure 15 illustre la page d'accueil du centre E-Learning et le champ où saisir les mots clés pour la recherche. Avec comme exemple en saisissant le mot clé 'm' dans le formulaire de recherche le centre nous octroie le nom du cours commençant par le mot clé et son enseignant.

V.5.2. Interface d'enregistrement

L'interface d'enregistrement montre un formulaire à remplir par un internaute dans le but de demander qu'il soit un étudiant ou un enseignant.



Figure 16: Interface d'enregistrement

La figure 16 illustre l'interface d'enregistrement du centre E-Learning accessible à tout un chacun. Si l'internaute veut s'enregistrer comme enseignant un champ dédié nommé grade apparait. L'administrateur du système valide ou refuse la demande.

V.5.3. Interface d'authentification

L'interface d'authentification montre un formulaire dans lequel un étudiant, un enseignant ou un administrateur peut remplir pour accéder à leurs propres interfaces.



Figure 17: Interface d'authentification

La figure 17 illustre l'interface d'authentification où un étudiant, enseignant ou administrateur est invité à remplir le formulaire avec leurs noms utilisateurs et mots de passe.

V.5.4. Interface d'un administrateur

L'interface d'un administrateur présente les activités accomplies par l'administrateur du centre. Parmi ses activités nous pouvons citer :

1) Une liste de demande des internautes souhaitant être un étudiant ou un enseignant ;
2) Une liste des catégories des cours dans laquelle l'administrateur peut ajouter, modifier, supprimer une catégorie des cours ;
3) Une liste des cours dans laquelle l'administrateur peut consulter ou effacer les cours;
4) Une liste des descriptions des cours dans laquelle l'administrateur peut consulter ou effacer les descriptions des cours ;
5) Une liste des enseignants dans laquelle l'administrateur peut consulter ou effacer les détails d'un enseignant ;
6) Une liste des étudiants dans laquelle l'administrateur peut consulter et effacer les détails d'un étudiant ;
7) Un message explicitant le pourquoi du système.

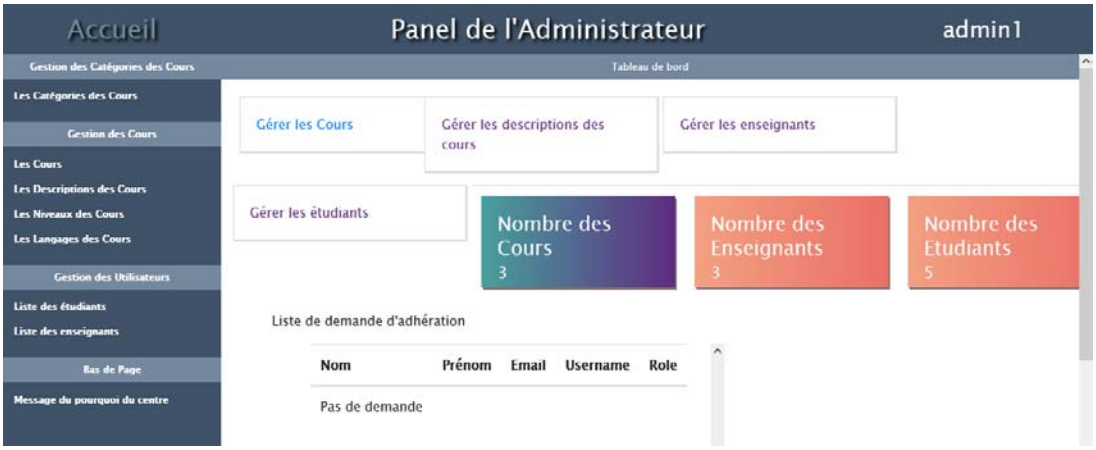


Figure 18: Interface d'accueil d'un administrateur

La figure 18 montre la page d'accueil de l'administrateur du centre.

Depuis la page d'accueil l'administrateur du centre a connaissance des demandes d'adhésion des internautes, du nombre total des cours et descriptions des cours chargés par les enseignants, du nombre total des enseignants et étudiants.

Un administrateur a le droit de définir et effacer les catégories des cours. Il peut s'il le veut générer un rapport sous format PDF avec l'usage de la librairie FPDF.



Figure 19: Interface de définition des catégories des cours par un administrateur

La figure 19 montre l'interface de définition des catégories des cours par un administrateur et ces catégories des cours sont accessibles à tout internaute sur la page d'accueil du centre.

Un administrateur peut consulter les cours définis par les enseignants et peut d'effacer un cours s'il le juge nécessaire. Il a aussi le droit de générer un rapport sous format PDF.

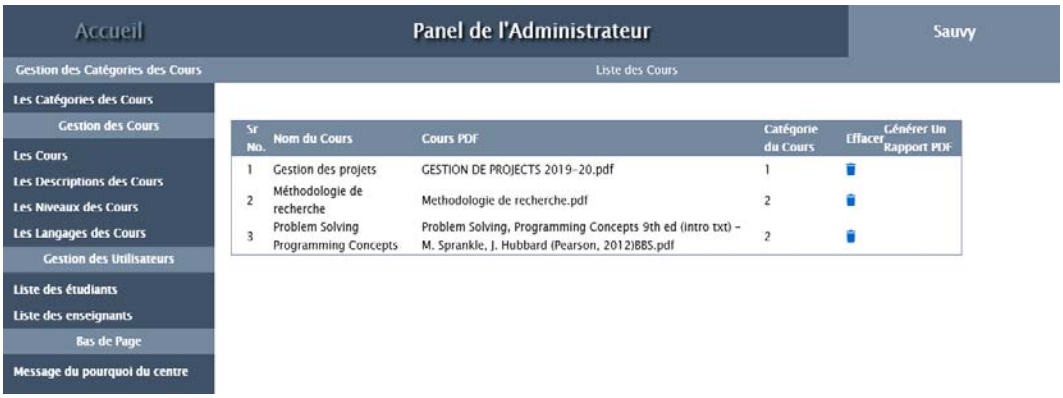


Figure 20: Interface de consultation des cours par un administrateur

La figure 20 montre l'interface de consultation des cours par un administrateur où ce dernier peut connaitre les noms des cours, les noms des livres et les noms des catégories des cours.

V.5.5. Interface d'un enseignant

Un enseignant a le rôle d'ajouter et de modifier un nouveau cours.



Figure 21 : Interface d'accueil d'un enseignant

La figure 21 montre l'interface d'accueil d'un enseignant où un enseignant consulte la liste des cours qu'il/elle enseigne et peut ajouter un nouveau cours. Par l'ajout d'un nouveau cours un enseignant a besoin de spécifier une catégorie des cours, un nom du cours, choisir un livre du cours à ajouter, le niveau du nouveau cours et la langue du livre utilisée dans le nouveau cours.

Un enseignant a aussi le droit de définir la description d'un cours tant textuelle que visuelle dans le but de montrer le pourquoi d'apprendre son cours.



Figure 22: Interface d'un enseignant pour l'ajout des descriptions d'un cours

Après la définition d'un nouveau cours un enseignant est invité à définir la description de ce dernier. La figure 22 montre l'interface de l'ajout des descriptions textuelles et multimédia d'un cours par un enseignant.

V.5.6. Interface d'un étudiant

Un étudiant a le rôle d'explorer les cours et leurs descriptions définis par un enseignant avant de s'enrôler au cours. Un étudiant du centre peut aussi faire des recherches des cours à sa volonté.
Avec cette recherche un étudiant a accès au nom d'un cours, au nom d'un enseignant et du support PDF du cours.

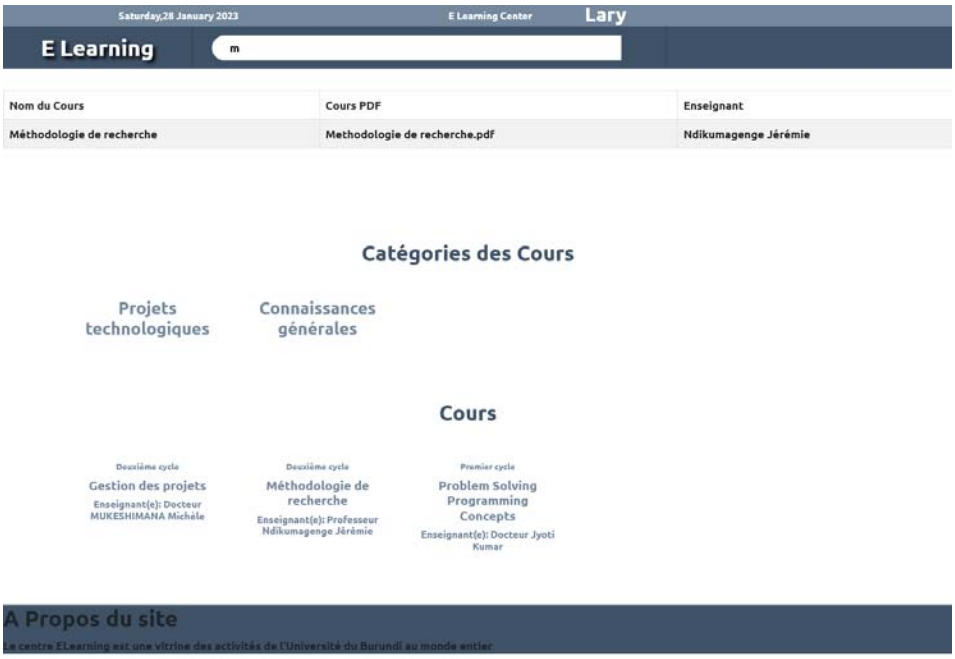


Figure 23: Interface d'accueil d'un étudiant

La figure 23 montre l'interface d'accueil d'un étudiant et le champ où saisir les mots clés pour la recherche par un étudiant. Avec comme exemple en saisissant le mot clé 'm' dans le formulaire de recherche un étudiant a accès au nom du cours commençant par le mot clé, au nom du livre du cours et son enseignant.

Un étudiant peut télécharger un livre PDF du cours.



Figure 24: Interface de téléchargement des cours par un étudiant

La figure 24 montre l'interface de téléchargement d'un cours par un étudiant. Un étudiant après avoir apprécié les descriptions du cours et s'être enrôlé au cours peut télécharger un livre du cours disponible sous format PDF.

V.5.7. Interface de vidéo conférence

Un enseignant a le rôle d'amorcer une vidéo conférence avec les étudiants.

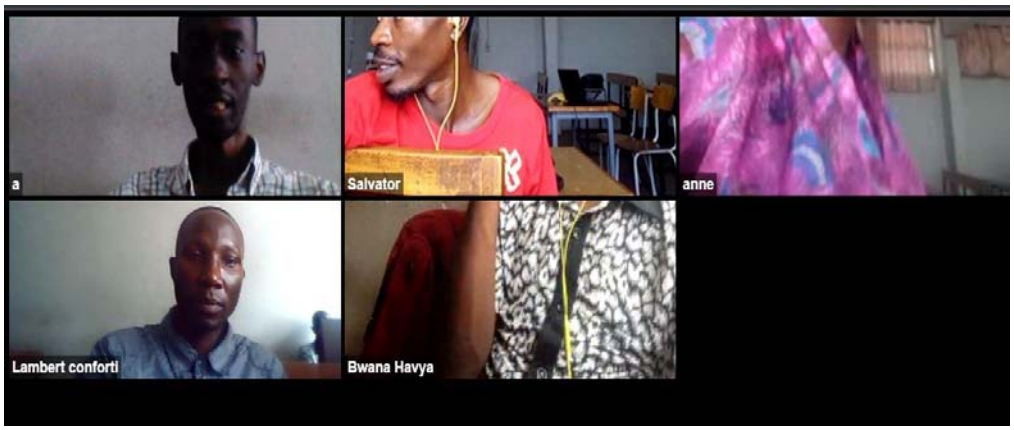


Figure 25: Interface de vidéo conférence entre un enseignant et des étudiants

La figure 25 montre l'interface de vidéo conférence où un enseignant est l'initiateur et invite les étudiants à y participer.

Ce chapitre nous a permis de présenter les outils et les langages de développement utilisés dans l'implémentation de notre système ainsi que certaines interfaces de notre centre.

Conclusion générale et Recommandations

1. Conclusion générale

Le présent travail s'est porté sur la mise en place d'un centre E-Learning à l'Université du Burundi avec la nécessité d'ouvrir les portes de la dite Université à n'importe qui voulant bénéficier de ses formations sans avoir à s'y présenter physiquement.

Le travail était subdivisé en cinq chapitres. Le premier chapitre a montré le pourquoi et le cadre du travail. Le second chapitre a étudié l'existant dans les Universités francophones issues de l'AUF. La conception et la modélisation du système avec UML comme outils de modélisation a eu lieu au troisième chapitre. Le quatrième chapitre a montré l'application de l'algèbre relationnelle comme modèle mathématique dans la mise en place du système. Quant au cinquième chapitre nous avons expliqué les outils de développement utilisés et avons montré quelques interfaces graphiques du système.

A l'issu du travail les utilisateurs du système peuvent consulter le centre indépendamment de leurs localisations géographiques et de leurs temps. Un enseignant peut définir un cours et sa description, un étudiant peut consulter les cours et descriptions définis par un enseignant. Un étudiant après s'être enrôlé à un cours peut télécharger le livre du cours et un enseignant peut décider d'amorcer une vidéo conférence avec les étudiants. De cela nous avons résolu les problèmes évoqués dans le premier chapitre.

2. Recommandations

Après l'implémentation du centre de mise en place d'un centre E-Learning à l'Université du Burundi, nous recommandons à l'Université du Burundi d'utiliser le centre et de proposer d'autres fonctionnalités pouvant être intégrés dans le système, aux futurs chercheurs d'améliorer le centre en améliorant les fonctionnalités existantes ou en y intégrant d'autres fonctionnalités pouvant aider l'approche de l'Université à s'ouvrir plus aux étrangers comme le paiement en ligne et la consultation des curriculums des cours.

Références bibliographiques

[1] https://www.etudesuniversitaires.ca/planifier-ses-etudes/pourquoi-poursuivre-des-etudes-superieures/, Consulté le 29 Août 2022

[2] https://www.lesechos.fr/ politique-societe/societe/le-nombre-detudiants-nen-finit-plus-de-grimper-en-france-1168900, Consulté le 29 Août 2022

[3] https://www.aa.com.tr/fr/education/burundi-faute-d-auditoires-toute-une-promotion-d-%C3%A9tudiants-est-pri%C3%A9e-d-attendre-encore-une-ann%C3%A9e/91988#, Consulté le 30 Août 2022

[4] Samuel MBADINGA, *Le Renforcement des capacités dans les institutions de formation des enseignants en Afrique subsaharienne : Burundi*, 2003.

[5] https://e-student.org/history-of-e-learning/, Consulté le 26 Août 2022

[6] https://www.callimedia.fr/e-learning-chiffres-et-tendances-du-marche/, Consulté le 29 Août 2022

[7] https://www.callimedia.fr/la-formation-a-distance-dans-le-monde/, Consulté le 29 Août 2022

[8] http://www.ecampus.ulbu.bi, Consulté le 30 Août 2022

[9] Universités francophones, *Former les ingénieurs par l'université virtuelle*, 2000.

[10] https://www.simplilearn.com/types-of-information-systems-and-applications-article, Consulté le 16 Août 2022

[11] https://www.visual-paradigm.com/guide/software-development-process/what-is-a-software-process-model/, Consulté le 16 Août 2022

[12] https://www.visual-paradigm.com/guide/, Consulté le 16 Août 2022

[13] https://www.uml.org/what-is-uml.htm, Consulté le 4 Octobre 2022

[14] https://www.guru99.com/uml-activity-diagram.html, Consulté le 4 Octobre 2022

[15] https://www.guru99.com/interaction-collaboration-sequence-diagrams-examples.html, Consulté le 4 Octobre 2022

[16] https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/, Consulté le 4 Novembre 2022

[17] https://creately.com/blog/diagrams/sequence-diagram-tutorial/, Consulté le 4 Novembre 2022

[18] https://www.guru99.com/uml-class-diagram.html, Consulté le 4 Octobre 2022

[19] https://laurent-audibert.developpez.com/Cours-UML/?page=diagrammes-composants-deploiement, Consulté le 10 Novembre 2022

[20] https://www.guru99.com/relational-algebra-dbms.html, Consulté le 14 Novemnre 2022

[21] https://openclassrooms.com/fr/courses/4449026-initiez-vous-a-lalgebre-relationnelle-avec-le-langage-sql/4568761-effectuez-un-produit-cartesien, Consulté le 15 Novembre 2022

[22] https://www.simplilearn.com/what-is-client-server-architecture-article, Consulté le 16 Novembre 2022

[23] https://www.easytechjunkie.com/what-is-a-text-editor.htm, Consulté le 18 Novembre 2022

[24] https://www.blogdumoderateur.com/tools/notepad/, Consulté le 18 Novembre 2022

[25] https://www.wpbeginner.com/glossary/apache/, Consulté le 19 Novembre 2022

[26] https://www.hostinger.fr/tutoriels/quest-ce-quapache-serveur-web-apache, Consulté le 19 Novembre 2022

[27] https://iweb.com/fr/blog/introduction-aux-serveurs-de-base-de-donnees, Consulté le 19 Novembre 2022

[28] https://en.wikibooks.org/wiki/PHP_Programming/PHP_Data_Objects, Consulté le 19 Novembre 2022

[29] https://www.codementor.io/php/tutorial/building-modern-secure-php-applications, Consulté le 20 Novembre 2022

[30] http://www.dicodunet.com/definitions/normes/langage-de-balisage.htm, Consulté le 20 Novembre 2022

[31] https://developer.mozilla.org/fr/docs/Web/JavaScript, Consulté le 21 Novembre 2022

[32] https://openclassrooms.com/fr/courses/3504441-introduction-a-jquery/3639626-javascript-jquery-cest-quoi-la-difference, Consulté le 23 Janvier 2023

[33] https://www.php.net/manual/fr/intro-whatis.php, Consulté le 4 Octobre 2022

[34] http://www.fpdf.org, Consulté le 22 Novembre 2022

[35] https://www.3cx.fr/voip-sip/quest-ce-que-webrtc/, Consulté le 28 Novembre 2022

[36] https://webrtc.org, Consulté le 23 Janvier 2023

[37] https://www.hornetsecurity.com/en/knowledge-base/it-security/, Consulté le 23 Novembre 2022

[38] https://www.geeksforgeeks.org/advanced-encryption-standard-aes/, Consulté le 18 Janvier 2023

[39] https://www.geeksforgeeks.org/cryptography-and-network-security-principles/, Consulté le 5 Décembre 2022

[40] https://www.getastra.com/blog/php-security/comment-securiser-le-site-php/, Consulté le 5 Décembre 2022