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

Conception et réalisation d'une application sécurisée de gestion des frais académiques: «Cas de l'ISP /Bukavu»

MUGARUKA BUDUGE, Gulain

2018-02
UB
https://repository.ub.edu.bi/handle/123456789/272

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

République du Burundi
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université du Burundi
Faculté des Sciences de l'Ingénieur



Département de TIC

Mastère en Génie Informatique

Année Académique: 2018-2019

Conception et réalisation d'une application sécurisée de gestion des frais académiques : « Cas de l'ISP/Bukavu »

MEMOIRE

Présenté Par

MUGARUKA BUDUGE Gulain

à la

Faculté des Sciences de l'Ingénieur (FSI)

En vue de l'obtention du grade de

MASTERE

en

Génie Informatique

Soutenu le 28/02/2018, devant le jury composé de :

PA.	NDIKUMAGENGE Jérémie	Président
Dr.	SAHINGUVU William	Vice-Président
Dr.	MUKESHIMANA Michele	Secrétaire
Dr.	NDAYISABA Longin	Directeur
Dr.	NKUNZIMANA Hilaire	Membre

DEDICACE

A mes très chers parents

Honorables, aimables :

Vous représentez pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi.

Votre prière et votre bénédiction m'ont été d'un grand secours pour mener à bien mes études.

Aucune dédicace ne saurait être assez éloquente pour exprimer ce que vous méritez pour tous les sacrifices que vous n'avez cessé de me donner depuis ma naissance, durant mon enfance et même à l'âge adulte.

Vous avez fait plus qu'un parent puisse faire pour que ses enfants suivent le bon chemin dans leur vie et leurs études.

Je vous dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout puissant, vous préserver et vous accorder santé, longue vie et bonheur.

REMERCIEMENTS

Mes remerciements s'adressent tout d'abord au Seigneur Tout puissant qui nous garde nuit et jour.

Ensuite, à mon directeur de mémoire en la personne du Docteur Longin NDAYISABA: J'ai eu le privilège de travailler parmi votre équipe et d'apprécier vos qualités et vos valeurs. Votre sériosité, votre compétence et votre sens du devoir nous ont énormément marqué. Veuillez trouver ici l'expression de ma respectueuse considération et ma profonde admiration pour toutes vos qualités scientifiques et humaines.

Ce travail est pour moi l'occasion de vous témoigner ma profonde gratitude. En plus, mes remerciements s'adressent également à mes familiers et connaissances, pour leur soutien et encouragement qu'ils ne cessent d'apporter à mon égard.

Enfin, que tous ceux qui m'ont aidé matériellement ou moralement trouvent à travers cette phrase mes sincères remerciements.

SIGLES ET ABREVIATIONS

ADMIN. DU BUDGET : Administrateur du Budget

CERUKI : Centre de recherche universitaire du Kivu

CH.B.REL PUBLIQUE : chef de Bureau relation publique

CH.DIV. GAR&SEC : chef de division garde et sécurité

CH.DIV.REL PUBLIQUE : chef de division relation publique

CH.SEC.SC.EX : Chef de section sciences exactes

CH.SEC.SC.H : Chef de section sciences humaines

CH.SEC.SCAI : Chef de section sciences commerciales administratives et informatique

D.C.S.AC : Directeur chef des services académiques

D.C.S.ADMIN : Directeur chef des services administratifs

D.C.S.FIN : Directeur chef des services financiers

D.G : Directeur Général

DIR.PARA.AC : Directeur des services et activités para académiques

DIR.SERV.AC : Directeur des services académiques

DIRCAB : Directeur de cabinet

EDAP : Ecole d'application

HTML: Hyper Text Markup Language

HTTP: Hyper Text Transfer Protocol

ISP : Institut Supérieur Pédagogique

PHP : Personal Home Page

R.AGR : Requête en algèbre relationnelle

R.SQL : Requête en SQL

RDC : République Démocratique du Congo

RSA: Rivest, Shamir et Adelman

SEC.GEN.ACADEMIQUE : Secrétaire général académique

SEC.GEN.ADMINISTRATIF : Secrétaire général Administratif

SGBD : Système de Gestion des Bases de Données

SI: Système d'information

SQL : Structured Query Language (langage d'interrogation des bases de données)

UML: Unified Modeling Language

UP: Unified Process

XML: EXtensible Markup Language

LISTE DES FIGURES

Figure 1.*Diagramme de cas d'utilisation d'un super user* 21
Figure 2.*Diagramme de cas d'utilisation de chef de section* 22
Figure 3. Diagramme de cas d'utilisation du SGAC 22
Figure 4.*Diagramme de cas d'utilisation du Directeur Général* 23
Figure 5.*Diagramme de cas d'utilisation du Percepteur* 23
Figure 6.*Diagramme de cas d'utilisation du Caissier* 24
Figure 7.*Diagramme de cas d'utilisation de l'AB* 24
Figure 8. *Diagramme d'activité de connexion au système* 26
Figure 9. *Diagramme d'activité d'ajout de nouveaux étudiants* 26
Figure 10.*Diagramme d'activité de la situation d'encaissement journalière, mensuelle et annuelle* .. 27
Figure 11.*Diagramme d'activité des relevés périodiques d'encaissement* 27
Figure 12.*Diagramme d'activité de la situation de chaque promotion* 28
Figure 13:diagramme d'activité de la visualisation de corbeille 28
Figure 14. *Diagramme d'activité visualiser la corbeille* 28
Figure 15. *Diagramme d'activité de la liste définitive* 29
Figure 16.*Diagramme d'activité de la perception des frais* 29
Figure 17.*Diagramme de séquence de connexion au système* 30
Figure 18.*Diagramme de séquence d'ajout de nouveaux étudiants* 31
Figure 19.*Diagramme de séquence de la situation d'encaissement journalière, mensuelle et annuelle* 31
Figure 20.*Diagramme de séquence des relevés périodiques d'encaissement* 31
Figure 21.*Diagramme de séquence de la situation de chaque promotion* 32
Figure 22.*Diagramme de séquence de la visualisation de la corbeille* 32
Figure 23.*Diagramme de séquence de la liste définitive* 33
Figure 24.*Diagramme de séquence de la réalisation de la perception* 33
Figure 25.*Diagramme des classes de notre système* 35
Figure 26.*Modèle Physique des données* 36
Figure 27. *Echange d'un message avec le chiffrement symétrique* 50
Figure 28.*Echange d'un message avec le chiffrement asymétrique* 50
Figure 29.*Table ASCII* 51
Figure 30.*Chiffrement avec César* 55
Figure 31.*Déchiffrement avec César* 57
Figure 32. *Interface de chiffrement César* 58
Figure 33.*Interface de génération des nombres premiers* 61
Figure 34.*Interface de chiffrement RSA* 63
Figure 35.*Interface de déchiffrement RSA* 65
Figure 36.*Chiffrement ave RSA* 67
Figure 37.*Déchiffrement avec RSA* 69
Figure 38.*Interface d'accueil WampServer* 71
Figure 39.*Modèle d'une table chiffrée en césar* 73
Figure 40. *Modèle d'une table chiffrée en RSA* 73
Figure 41.*Page d'accueil de notre application* 74
Figure 42.*Interface de connexion* 74
Figure 43.*Espace AB* 74

Figure 44.*Interface de création des nouveaux utilisateurs* 75
Figure 45.*Interface de privilège de connexion* 75
Figure 46.*Interface d'autorisation* 75
Figure 47.*Interface de suppression* 76
Figure 48.*Interface d'Initialisation de la somme à payer* 76
Figure 49.*Interface de modification de compte* 76
Figure 50. *Interface de recherche de la situation journalière* 77
Figure 51.*Situation Journalière en html* 77
Figure 52.*Situation Journalière en PDF* 77
Figure 53.*L'interface de recherche de la situation mensuelle* 78
Figure 54.*Situation mensuelle en html* 78
Figure 55.*Situation mensuelle en pdf* 78
Figure 56.*Interface de recherche de la situation annuelle* 79
Figure 57. *Situation annuelle en format html* 79
Figure 58.*Situation annuelle en format pdf* 79
Figure 59.*Interface de recherche de la situation synthétique journalière* 80
Figure 60.*Situation synthétique html* 80
Figure 61.*Situation synthétique en Excel* 80
Figure 62.*Interface de recherche de la situation par promotion* 81
Figure 63.*Résultat de la situation par promotion en html* 81
Figure 64.*Résultat de la situation par promotion en pdf* 82
Figure 65.*Interface de recherche de la situation de chaque étudiant par année* 82
Figure 66.*Résultat de la situation de l'étudiant en html* 82
Figure 67.*Résultat de la situation de l'étudiant en pdf* 83
Figure 68.*Interface d'affichage des statistiques* 83
Figure 69.*Statistique en Excel* 83
Figure 70 .*Modèle du reçu d'un étudiant* 84

RESUME

Le présent travail porte sur la mise en place d'une application sécurisée de gestion des frais académiques dans un établissement d'enseignement supérieur et universitaire. Dans ce travail nous essayerons de mettre en œuvre une application de gestion financière des étudiants, qui stockera toutes les informations relatives à la paie de ces derniers, le système d'audit et de génération automatique et d'impression des rapports sera mis en place ce qui permettra d'avoir la situation financière en temps réel et au moment opportun. La protection des données contre les accès non autorisés sera mise en place à l'aide de deux crypto systèmes qu'on réalisera (le premier est le crypto système asymétrique RSA codé sur 512 bits et l'autre est le crypto système symétrique César utilisant la table ASCII comme alphabet). Ces crypto systèmes auront pour rôle le cryptage et le décryptage de toutes les données importantes stockées dans notre système de gestion des bases des données. Le premier sera utilisé pour tous les identifiants tandis que le second sera réservé pour tout autre type d'informations (c'est-à-dire tout enregistrement stocké dans le SGBD sauf les identifiants).

Mots clés : Frais académiques, finance, étudiants, crypto système, cryptanalyse

ABSTRACT

The present work is about the implementation of a secure application for academic fees management in a University and Higher Education institution, in which we will try to implement a student financial management application, that will store all the students' payment information, the system of auditing and automatic generation and printing of reports will be set up allowing to have the financial situation in real time and in a timely manner. The protection of data against unauthorized access will be implemented using two crypto systems that will be set up (the first is the asymmetric cryptosystem RSA encoded on 512 bits and the other is the Caesar symmetric crypto system using the ASCII table as alphabet) and whose role will be the encryption and decryption of all important data stored in our database management system. The first will be used for all identifiers whereas the second will be reserved for any other type of information, i.e. any recording stored in the SGBD apart from identifiers.

Key words : Academic fees, finances, students, cryptosystem, cryptanalysis

TABLE DES MATIERES

DEDICACE i
REMERCIEMENTS ii
SIGLES ET ABREVIATIONS iii
LISTE DES FIGURES iv
RESUME vi
ABSTRACT vi
TABLE DES MATIERES vii
CHAPITRE I. INTRODUCTION GENERALE 1
I.1. Justification du contexte 1
I.2. Présentation du milieu d'étude 4
I.2.1. Historique 4
I.2.2. Organigramme 6
I.3. Filières organisées 7
CHAPITRE II. CONCEPTION D'UN SYSTEME D'INFORMATION 8
II.1. Introduction 8
II.2. Processus unifié 8
II.2.1. Principes d'UP 8
II.2.2. Phases du processus unifié et les activités 9
II.2.3. Activités du processus 10
II.3. Planification de projet 11
II.4. Généralités sur le langage UML 18
II.5. Modélisation avec le langage UML 19
II.2.5.1. Diagrammes de cas d'utilisation 19
II.5.2. Diagrammes d'activités 25
II.5.3. Diagrammes de séquence 30
II.5.4. Diagramme des classes 33
II.5.5. Modèle Physiques des données 36
Conclusion partielle 37
CHAPITRE III. FORMALISMES MATHEMATIQUES AVEC ALGEBRE RELATIONNELLE 38
III.1. Introduction 38
III.2. Opérateurs fondamentaux : projection, sélection et jointure 38
III.3. Opérateurs ensemblistes et autres 44
III.4. Illustration de quelques requêtes SQL (de notre projet) en algèbre relationnelle 46

CHAPITRE IV. PROTECTION DES DONNEES CONTRE LES ACCES NON AUTORISES 49
IV.1. Introduction 49
IV.2. Le chiffrement des données informatiques 49
IV.2.1. Types de chiffrement 49
IV.3. Choix des algorithmes de chiffrement 51
IV.3.1. L'algorithme de CESAR (avec 256 caractères) 51
IV.3.2. Algorithme RSA codé sur 512 bits 60
Conclusion partielle 70
CHAPITRE V. REALISATION D'UNE APPLICATION SECURISEE DE GESTION DES FRAIS ACADEMIQUES 71
V.1. Outils de Développement 71
V.2. Présentation de l'application 72
CONCLUSION GENERALE ET RECOMMANDATIONS 85
BIBLIOGRAPHIE 86

CHAPITRE I. INTRODUCTION GENERALE

I.1. Justification du contexte

1. Généralités

L'homme étant pourvu d'intelligence, il cherche à tout prix à rendre son environnement de plus en plus compétitif. C'est ainsi que vers ces dernières décennies, la science s'est vue progresser dans presque tous les domaines, entre autres la médecine, l'informatique et bien d'autres.

De nos jours, l'informatique est devenue présente dans plusieurs secteurs et nul ne songe de s'en passer. Elle est à la base de la croissance de l'entreprise.

Dans l'environnement actuel, la compétitivité des entreprises dépend de plus en plus de leur flexibilité et de leur capacité d'innover, tant dans leur structure organisationnelle, leur mode de production que dans leur mode d'échange avec les différents partenaires.

Bien que cette science de traitement automatique de l'information cherche à satisfaire les besoins de certaines entreprises, les autres restent encore absentes car n'arrivant pas à être à la fine pointe de l'information.

Il sied de soulever que le nombre élevé d'étudiants ne permet pas à l'Institut Supérieur Pédagogique de Bukavu de faire régulièrement le suivi et le contrôle efficace des opérations qui s'y passent. La difficulté de faire ce suivi et ce contrôle cause non seulement à cette Institution mais aussi aux étudiants les problèmes ci-après : la perte des données due à la vétusté des documents, l'omission de certains étudiants sur des listes, etc. Les problèmes ci hauts énumérés sont liés à l'utilisation du système manuel.

2. Objectifs du projet

1) Objectif global du travail

L'objectif de ce travail est de concevoir et de réaliser une application sécurisée et déployée en réseau qui permettra à l'Institut Supérieur Pédagogique de Bukavu de résoudre les problèmes liés à la gestion des frais académiques.

2) Objectifs spécifiques

Les principaux objectifs spécifiques de ce travail sont :

- Concevoir une application sécurisée de gestion déployée en réseau ;
- Générer et imprimer les rapports journaliers, mensuels et annuels de la perception ;
- Imprimer la situation de chaque promotion ;

- Elaborer la liste des créanciers, Concevoir des algorithmes d'upload des listes des étudiants vers notre système (excel-php-mysql);
- Concevoir des fonctions des chiffrages des données stockées dans le système pour une meilleure sécurité.

3. Problématique

Etant donné que l'Institut Supérieur Pédagogique de Bukavu utilise le système manuel, il se heurte aux problèmes suivants:

- La difficulté de produire des rapports fiables en temps réel **;**
- La perte de temps pendant la recherche d'une information **;**
- La perte des données due à la vétusté des documents ;
- L'impossibilité de dégager les statistiques fiables des étudiants en ordre et les erreurs dans le calcul lors de la perception ;
- La difficulté de faire le contrôle et le suivi des percepteurs.

Le constat ci-dessus nous a fort préoccupé et nous a amené à réfléchir autour de la question maîtresse de notre recherche que voici : la conception et la réalisation d'une application sécurisée de gestion des frais académiques dans cette Institution ne serait-elle pas la solution aux problèmes ci-hauts ?

4. Solutions proposées

Etant donné que notre problème, concerne principalement la conception et la réalisation d'une application sécurisée de gestion des frais académiques à l'ISP/Bukavu, nous affirmons que les réponses provisoires (solutions proposées) suivantes sont appropriées à la problématique : partage des données dans toute l'Institution via le réseau, production des rapports en temps réel, génération des rapports journaliers, mensuels et annuels (pour chaque percepteur) au temps voulu, impression des reçus des étudiants, impression des listes fiables des étudiants en ordre, chiffrement des données stockées dans la base des données avec des crypto systèmes forts.

5. Résultats attendus

L'Institution doit disposer de l'intranet et doter d'une application réseau lui permettant d'effectuer des échanges tout en minimisant le temps, les utilisateurs doivent être formés dans l'utilisation de l'application, les reçus de paiement et autres documents doivent être générés automatiquement, les erreurs dans le calcul lors de la perception diminueront, le service de finance doit fournir en temps réel les informations nécessaires.

6. Apports scientifiques et technologiques

La mise en place d'une application sécurisée permettant les échanges dans toute l'Institution, la conception et la réalisation des fonctions de chiffrement des données avec RSA et César pour une très bonne sécurité, réalisation des algorithmes d'upload des données Excel vers Mysql via PHP.

7. Motivation

Nous avons choisi cette thématique en raison de la complexité des domaines informatiques qu'elle aborde: le système d'information, les systèmes de gestion des bases des données, la programmation ainsi que la sécurité des systèmes. Ayant appris les notions relatives à ces derniers, nous nous sommes décidé de les appliquer afin d'acquérir les performances de haut niveau dans les domaines précités.

8. Domaines d'application

Cette application sécurisée sera utilisée dans les établissements de l'enseignement supérieur et universitaire, dans le réseautage informatique, dans l'optimisation des systèmes d'information et systèmes informatiques de gestion, etc.

9. Méthodologie utilisée

Pour réaliser cette recherche, nous avons recouru aux méthodes UP (Unified Process) [basé au langage UML], analytique et aux techniques d'entretien, documentaire et d'interview.

10. Outil mathématique utilisé

L'outil mathématique qui a été sollicité est l'Algèbre relationnelle. L'algèbre relationnelle est un langage de requêtes dans des bases de données relationnelles. L'algèbre relationnelle a été inventée en 1970 par Edgar Frank Codd. Elle a comme objectif :

- Fournir les opérateurs de base pour manipuler les extensions des relations (ensembles de n-uplets ou tables) d'une BD relationnelle.
- Faire un parallèle avec l'arithmétique qui fournit les opérations de base pour manipuler les nombres (addition, soustraction, multiplication et division).

Une requête SQL est traduite, comme on le verra, en un arbre d'opérateurs de l'algèbre relationnelle.

I.2. Présentation du milieu d'étude

I.2.1. Historique

L'ISP Bukavu est une Institution publique. Son histoire peut être découpée en trois moments forts que sont :

1. De 1961 à 1971 : La période de la Régence ;
2. De 1971 à 1986 : La période de Gloire du Révérend Père Dominique Milani ;
3. De 1986 à ce jour : L'après Milani.

L'école de Régence a été ouverte et autorisée à fonctionner à Nyangezi le 9 Octobre 1961 sous la direction du Frère Mariste Césaire et sous l'impulsion du Bureau de l'Enseignement Catholique, B.E.C. en sigle. En 1964, cette école fut transférée à Bukavu au Collège Notre Dame de la Victoire des Pères Jésuites.

L'Ecole de Régence a changé la dénomination en Ecole Supérieure Pédagogique (ESP) par la circulaire n°EDN/ES/RS/02116 du 10 décembre 1964 du Département de l'Education Nationale, avant de devenir l'Ecole Normale Moyenne (ENM).

L'ESP de Bukavu était dirigée par le Père Louis André (1964-1966) et à partir de 1966 par le Révérend Père Dominique Milani (Père Xavérien). Les cours y ont été dispensés à partir d'avril 1966.

La dénomination Institut Supérieur Pédagogique de Bukavu apparaît avec l'ordonnance-loi n°71-075 du 06/08/1971 créant l'université Nationale du Zaïre (UNAZA). Avec la forme de 1971, l'ISP de Bukavu fut intégré dans l'UNAZA.

De 1971 à 1986, l'ISP de Bukavu réussit à se confirmer et à s'imposer comme structure de formation des formateurs sur l'échiquier national et régional.

L'année 1986 correspond à l'année de départ du Père Milani et le début de la gestion de l'Institut par des cadres congolais sous la tutelle entière du Ministère de l'Enseignement Supérieur et Universitaire. Dès lors, cinq Directeurs Généraux se sont succédés. Ce sont : Les professeurs BIKAY Obel Ernest (1986-1993), BISHIKWABO Cubaka (1993-1996), MULYUMBA wa MAMBA ITONGWA Barnabé (1996-2006) et KANINGINI MWENYIMALI Boniface (2006-2017), Mzee Patrick SOMORA (2017 à nos jours).

Si l'ère de Milani a été considérée, à juste titre, comme celle des vaches grasses en dépit de quelques défis, l'on doit reconnaître qu'après ce héros bâtisseur, l'ISP va naviguer à travers plusieurs flots au gré des événements sociopolitiques qui se sont succédés au pays depuis les années 90.

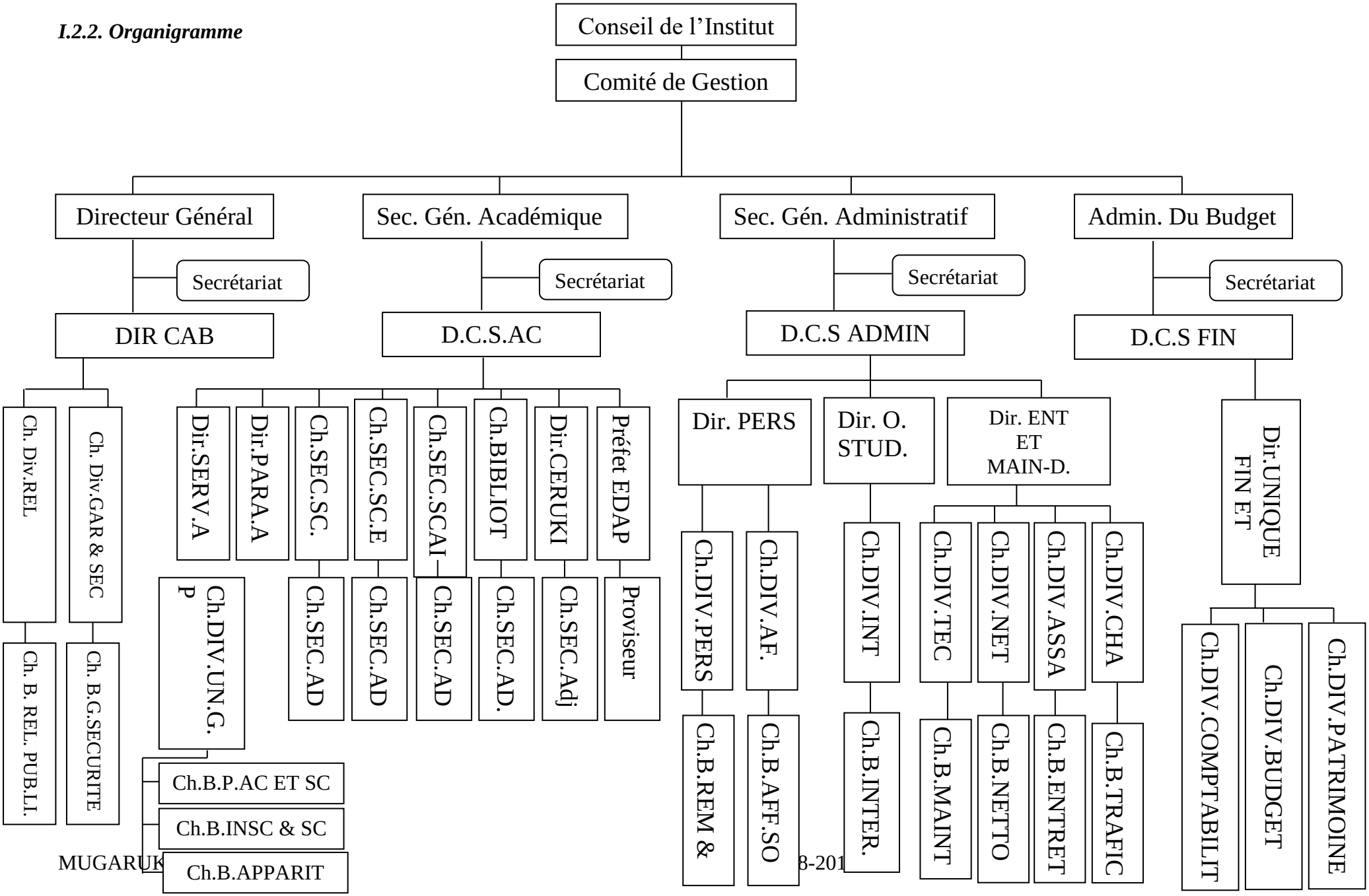
L'Institution qui a fait face à beaucoup de défis, a, malgré tout, survécu jusqu'à jouer le rôle de leadership dans le secteur de l'Enseignement Supérieur et Universitaire au Kivu et dans la région des Grands Lacs africains.

Si l'ISP de Bukavu peut se réjouir aujourd'hui de son rang de grande Institution, il a aussi connu des hauts et des bas.

En effet, sa phase de maturité est à situer entre 2004 et 2011, période pendant laquelle certains signaux sont remarquables. Il s'agit entre autres du nombre croissant de professeurs et du développement des infrastructures. Mais, l'indicateur le plus visible sur le plan académique et organisationnel reste l'ouverture de la nouvelle section des « Sciences Commerciales Administratives et Informatique de Gestion », section considérée comme celle de la modernisation par rapport aux deux autres traditionnellement existantes (Lettres et Sciences Humaines, Sciences Exactes).

A cette grande innovation, il convient d'ajouter le récent arrêté ministériel qui vient d'accorder à l'ISP de Bukavu l'autorisation d'organiser, en son sein, le troisième cycle, ce qui va faire de l'ISP une Institution universitaire certes, il le deviendra de droit, car le prochain cinquantenaire s'annonce prometteur.

I.2.2. Organigramme



I.3. Filières organisées

L'ISP/Bukavu organise cinq sections réparties comme suit :

1. ***Section de sciences exactes***

Cette section est parmi les plus anciennes de l'ISP/BUKAVU.

Elle compte en son sein 6 départements dont Agro-vétérinaire qui débute cette année académique (2019-2020), Biologie chimie, Chimie physique, Géographie et Gestion de l'environnement, Math-Physique et Physique Technologie.

Elle organise le système LMD (Licence Master Doctorat) à partir de l'année 2019-2020

2. ***Section de lettres et sciences humaines***

Elle est aussi ancienne comme la précédente.

Elle compte 4 départements en son sein : Anglais et Cultures Africaines, Français et langues '

3. ***Section des sciences hôtelières et Touristiques***

Cette section compte 2 départements dont Accueil et Tourisme, Hôtellerie et Restauration.

4. ***Section de sciences commerciales, administratives et informatiques***

Elle compte au total 2 départements dont Sciences Commerciales et Administratives et Informatique de Gestion.

5. ***Ecole d'application de l'ISP/Bukavu***

L'école d'application de l'ISP/Bukavu est comptée parmi les laboratoires de cette Institution.

Elle organise 7 sections en plus des classes de $7^{ème}$ et $8^{ème}$ année de l'éducation de base. Ces sections sont: Math-physique, Bio-chimie, Technique-agricole, Technique-vétérinaire, commerciale et gestion, Hôtesse d'accueil, hôtellerie et restauration.

Conclusion partielle

Nous sommes au terme de ce chapitre, qui a été consacré à la présentation de notre sujet de recherche, des aspects méthodologiques ainsi qu'à la localisation de notre milieu et à la présentation de son organisation.

CHAPITRE II. CONCEPTION D'UN SYSTEME D'INFORMATION

II.1. Introduction

Dans cette partie nous allons aborder la notion de gestion de projet et celle relative à la modélisation.

Avec la première, nous montrerons toutes les étapes nécessaires pour réaliser un bon projet d'abord, ensuite décrire les toutes les tâches à réaliser, leurs couts et durée en fin élaborer le diagramme de PERT en spécifiant les marges et le chemin critique.

Dans la deuxième, nous montrerons toutes les phases de développement d'un projet informatique avec la méthode UP et présenter tous les diagrammes UML nécessaires de notre projet. Chacune des parties sera introduite par un bref aperçu des terminologies.

II.2. Processus unifié

Le processus unifié c'est un processus de développement moderne, itératif, efficace sur des projets informatiques de toutes tailles. Très complet, il couvre l'ensemble des activités, depuis la conception du projet jusqu'à la livraison de la solution.

Intégrant une organisation de projet type, une méthodologie utilisant UML et un ensemble de bonnes pratiques cohérentes entre elles, il permet de circonvenir aux problèmes récurrents que rencontrent nombre de réalisations : dérive des coûts et des délais, qualité insuffisante, réponse incomplète aux attentes des utilisateurs.

Un point d'excellence de cette démarche est son adaptabilité : UP peut se décliner en fonction de l'ampleur d'un projet, de l'expérience de l'équipe qui l'assume, de la nature de la solution à construire [1].

II.2.1. Principes d'UP

Le processus de développement UP, associé à UML, met en œuvre les principes suivants :

- processus guidé par les cas d'utilisation,
- processus itératif et incrémental,
- processus centré sur l'architecture,
- processus orienté par la réduction des risques.

Ces principes sont à la base du processus unifié décrit par les auteurs d'UML.

- ✓ **Processus guidé par les cas d'utilisation**

L'orientation forte donnée ici par UP est de montrer que le système à construire se définit d'abord avec les utilisateurs. Les **cas d'utilisation** permettent d'exprimer les interactions du système avec les utilisateurs, donc de capturer les besoins.

- ✓ **Processus itératif et incrémental**

Ce type de démarche étant relativement connu dans l'approche objet, il paraît naturel qu'UP préconise l'utilisation du principe de développement par **itérations** successives. Concrètement, la réalisation de maquette et prototype constitue la réponse pratique à ce principe. Le développement progressif, par **incrément**, est aussi recommandé en s'appuyant sur la décomposition du système en cas d'utilisation.

Les avantages du développement itératif se caractérisent comme suit :

- Les risques sont évalués et traités au fur et à mesure des itérations ;
- Les premières itérations permettent d'avoir un feed-back des utilisateurs ;
- Les tests et l'intégration se font de manière continue,
- Les avancées sont évaluées au fur et à mesure de l'implémentation.

- ✓ **Processus centré sur l'architecture**

Les auteurs d'UP mettent en avant la préoccupation de **l'architecture du système** dès le début des travaux d'analyse et de conception. Il est important de définir le plus tôt possible, même à grandes mailles, l'architecture type qui sera retenue pour le développement, l'implémentation et ensuite le déploiement du système. Le vecteur des cas d'utilisation peut aussi être utilisé pour la description de l'architecture.

- ✓ **Processus orienté par la réduction des risques**

L'analyse des **risques** doit être présente à tous les stades de développement d'un système. Il est important de bien évaluer les risques des développements afin d'aider à la bonne prise de décision. Du fait de l'application du processus itératif, UP contribue à la diminution des risques au fur et à mesure du déroulement des itérations successives.

II.2.2. Phases du processus unifié et les activités

Les phases d'un processus de développement sont des états de celui-ci à un instant t. Le cycle de développement du Processus Unifié organise les tâches et les itérations en quatre phases :

- ✓ **Inception ou (commencement)** : Cette phase correspond à **l'initialisation du projet** où l'on mène une étude d'opportunité et de faisabilité du système à construire. Une évaluation des risques est aussi réalisée dès cette phase [2].

En outre, une identification de principaux cas d'utilisation accompagnée d'une description générale est modélisée dans un diagramme de cas d'utilisation afin de définir le périmètre du projet.

✓ **Élaboration** : Cette phase reprend les résultats de la phase d'inception et élargit l'appréciation de la **faisabilité** sur la quasi-totalité des cas d'utilisation. Ces cas d'utilisation se retrouvent dans le diagramme des cas d'utilisation qui est ainsi complété.

Cette phase a aussi pour but d'analyser le domaine technique du système à développer afin d'aboutir à une architecture stable. Ainsi, toutes les exigences non recensées dans les cas d'utilisation, comme par exemple les exigences de performances du système, seront prises en compte dans la conception et l'élaboration de l'architecture.

✓ **Construction :** Cette phase correspond à la **production** d'une première version du produit. Elle est donc fortement centrée sur les activités de conception, d'implémentation et de test.

En effet, les composants et fonctionnalités non implémentés dans la phase précédente le sont ici.

✓ **Transition :** Après les opérations de test menées dans la phase précédente, il s'agit dans cette phase de **livrer le produit** pour une exploitation réelle. C'est ainsi que toutes les actions liées au déploiement sont traitées dans cette phase. De plus, des « bêta tests » sont effectués pour valider le nouveau système auprès des utilisateurs.

II.2.3. Activités du processus

Les activités représentent les actions à effectuer au cours d'une phase : une phase passe par l'ensemble des activités. Le temps passé par activité est fonction des phases.

Nous nous limiterons donc à ne donner qu'une brève explication de chaque activité.

UP propose d'appréhender l'**expression des besoins** en se fondant sur une bonne compréhension du domaine concerné pour le système à développer et une modélisation des procédures du système existant.

Ainsi, UP distingue deux types de besoins :

- les besoins fonctionnels qui conduisent à l'élaboration des cas d'utilisation,
- les besoins non fonctionnels (techniques) qui aboutissent à la rédaction d'une matrice des exigences.

1. Analyse

L'analyse permet une formalisation du système à développer en réponse à l'expression des besoins formulée par les utilisateurs. L'analyse se concrétise par l'élaboration de tous les diagrammes donnant une représentation du système tant statique (diagramme de classe principalement), que dynamique (diagramme des cas d'utilisation, de séquence, d'activité, d'état-transition…).

2. **Conception**

La **conception** prend en compte les choix d'architecture technique retenus pour le développement et l'exploitation du système. La conception permet d'étendre la représentation des diagrammes effectuée au niveau de l'analyse en y intégrant les aspects techniques plus proches des préoccupations physiques.

3. **Implémentation**

Cette phase correspond à la **production du logiciel** sous forme des composants, des bibliothèques ou de fichiers.

4. **Test**

Il permet de vérifier :

- La bonne implémentation de toutes les exigences (fonctionnelles et techniques),
- Le fonctionnement correct des interactions entre les objets,
- La bonne intégration de tous les composants dans le logiciel.

II.3. Planification de projet

L'étape de la **planification de projet** est cruciale. Si la planification d'un projet doit être réalisée avec autant de soin, c'est qu'elle va décider de son déroulement par la suite. Il va être découpé en tâches qui vont ensuite être estimées en termes des charges, puis réparties. Cette estimation des charges va permettre d'estimer les besoins en ressources et déterminer la date prévisionnelle de la fin de projet. Le respect du planning projet ainsi constitué va déterminer la réussite ou l'échec du projet.

Parmi les différentes phases d'un projet, la planification est sans aucun doute l'une des plus importantes. Elle consiste à déterminer la liste des tâches à réaliser, à estimer pour chacune d'elle le coût de réalisation et à sélectionner les profils nécessaires et les ressources à prévoir.

Le planning constitué va permettre de réaliser le suivi de projet durant son déroulement. Il sera ainsi possible de déterminer si les objectifs fixés sont atteints, de suivre l'avancement au travers d'un tableau de bord de suivi de projet et enfin de contrôler l'affectation des ressources à la réalisation de différentes tâches.

Une planification des tâches rigoureuse et réaliste sera donc le garant de l'avancement et de la réussite du projet.

II.3.1. Etapes pour bien planifier un projet

Étape 1 : le découpage

La toute première étape de la planification de projet consiste à effectuer un découpage en phases chronologiques. Pour chaque phase, il faut ensuite déterminer la liste des tâches à accomplir, les charges à prévoir et les ressources nécessaires. Les résultats attendus et les livrables du projet sont également détaillés, tout comme le processus de validation employé.

Étape 2 : la hiérarchisation des tâches

Une fois la liste des tâches terminée, il faut décider lesquelles seront prioritaires et devront être réalisées en premier. Il faudra également déterminer les éventuelles interdépendances de façon à anticiper les problèmes. Si des prérequis sont identifiés pour la réalisation d'une tâche, ils devront naturellement être pris en compte et verront leur niveau de priorité augmenter.

Les principaux livrables doivent faire l'objet d'une description complète. Un exemple de livrable d'un projet peut d'ailleurs être fourni afin de servir de modèle. Ces livrables peuvent également être divisés en sous-livrables, et les activités et sous-activités liées à leur réalisation listée. Plus le découpage des livrables en sous-livrables et des activités en sous-activités sera fin, plus la réalisation des étapes suivantes pourra être précise.

Étape 3 : l'ordonnancement des tâches

L'ordonnancement des tâches d'un projet consiste à déterminer dans quel ordre elles devront être réalisées. Il s'agit également à ce stade d'identifier les tâches qui devront être réalisées séquentiellement et celles qui pourront au contraire être parallélisées.

Pour cela, il faut se poser les questions suivantes pour chaque tâche identifiée :

- Cette tâche dépend-elle d'une ou plusieurs autres tâches ? Si c'est le cas, ces prérequis peuvent-ils être traités parallèlement ou bien doivent-ils être traités les uns après les autres ?
- Existe-t-il une marge pour la réalisation de cette tâche ? Un débordement est-il acceptable, et si oui, à combien peut-il se monter sans mettre en danger la suite du projet ?

Il est possible d'élaborer plusieurs scénarios possibles pour l'ordonnancement des tâches. Suivant le déroulement du projet, un scénario particulier pourra être privilégié par rapport à un autre.

Étape 4 : définition du planning

Une fois l'ordonnancement des tâches terminé, l'ordre dans lequel elles doivent être réalisées en fonction de leur priorité est connu. L'objectif est maintenant, à partir de la priorité et de la

charge estimée des tâches, de fixer à chacune des dates de réalisation. A la fin de cette opération, vous aurez le planning projet, qui vous permettra d'identifier les différents jalons qui vous permettront d'atteindre vos objectifs. C'est donc l'indispensable outil de planification de projet. La définition de la planification est une étape particulièrement importante du processus d'élaboration d'un projet, puisqu'elle va notamment déterminer sa date de fin prévisionnelle.

Prenons un exemple de planification d'un projet de développement informatique. Un client souhaite mettre en place une solution personnalisée de gestion de son personnel. Le projet va connaître différentes phases :

1. **Etude préalable :** il s'agit de déterminer le périmètre exact du projet (la « gestion des frais académiques » est une notion très vaste, l'application devra-t-elle comprendre le paiement des documents académiques, des frais de scolarité, des frais de labo, de la paie… ?) et de rédiger un cahier des charges.
2. **Conception :** cette phase correspond à la conception technique de la solution. Des choix qui sont faits à ce stade dépendent de l'estimation de la durée de réalisation de différentes tâches identifiées.
3. **Réalisation :** développement de l'application.
4. **Tests :** tests unitaires, techniques et fonctionnels de l'application.
5. **Recette :** recette avec le client, afin de vérifier que l'application livrée correspond bien au besoin exprimé.
6. **Livraison et mise en production :** installation et mise en service de l'application.

Le processus de planification permet la définition pour chaque tâche de ses dates de début et de fin au plus tôt, et de ses dates de début et de fin au plus tard. Chaque tâche va donc se voir attribuer quatre dates qui tiendront compte de la réalisation de ses prérequis. La durée estimée de réalisation de la tâche est calculée en faisant la différence entre sa date de fin et sa date de début.

Lorsque l'ensemble des dates de début et de fin a été défini, la séquence continue des tâches, permet de déterminer la date de fin prévisionnelle du projet. Il est possible d'établir le chemin critique du projet. Ce chemin, constitué de tâches critiques, est continu du début à la fin du projet. Toute modification de la durée d'une tâche critique a donc une répercussion immédiate sur la durée totale du projet et donc sur sa date de fin. Si une tâche présente dans le chemin critique nécessite une semaine de plus que prévu pour être achevée, alors la date de fin du projet sera automatiquement repoussée d'une semaine, avec toutes les conséquences que cela

pourra entraîner. Pour prendre en compte ce risque, la planification en management de projet va attribuer des marges à chaque tâche. C'est-à-dire que l'on va attribuer à chaque tâche la possibilité de prendre un peu de retard, sans que cela ait un impact sur la durée totale du projet. Les tâches étant sur le chemin critique ont une marge nulle, mais les autres peuvent bénéficier de deux types de marge distincts : [3]

- **La marge libre (ML) :** elle est égale à la différence entre la date de début au plus tôt de la tâche suivante la plus proche et la date de fin au plus tôt de la tâche courante.
- **La marge totale (MT) :** elle est égale à la différence entre la date de début au plus tard de la tâche suivante présentant le plus de contraintes et la date de fin au plus tôt de la tâche courante.

En fonction de ces deux marges, il est possible de déduire l'impact qu'aura un décalage de la réalisation d'une tâche sur l'ensemble du projet.

Maintenant que nous avons réussi à estimer la durée du projet, il va falloir déterminer son budget. Pour cela, il va nous falloir une estimation des charges. Elle va se faire à plusieurs niveaux. Chaque tâche doit faire l'objet d'une estimation. Il s'agit du coût correspondant à la personne qui sera chargée de sa réalisation, et plus généralement du profil nécessaire, car à ce stade, on ne sait pas encore exactement qui lui sera affecté. Une deuxième estimation des charges doit être faite pour chaque phase (étude, conception, recette…) en fonction du découpage du projet, afin de prévoir les ressources nécessaires et leurs affectations. Enfin, tous ces éléments permettent d'estimer la charge complète du projet et d'établir une enveloppe budgétaire.

Toutes ces estimations doivent naturellement tenir compte des charges matérielles incluant les locaux, le matériel informatique, les licences logicielles, les serveurs ou encore les frais de déplacement si nécessaire. Il ne faut surtout pas oublier de tenir compte des délais d'approvisionnement si vous devez faire appel à des fournisseurs. Même si le matériel informatique est fourni par un service interne, il lui faudra sans doute un certain temps avant que tout soit en place et opérationnel. Si des profils particuliers sont nécessaires et qu'ils ne sont pas disponibles en interne, il faut prendre en compte le temps de recherche et de recrutement des ressources humaines ou du choix des prestataires extérieurs.

Étape 5 : l'identification des risques

L'identification des risques a pour objectif l'anticipation de différents problèmes pouvant être rencontrés sur le projet et ayant un impact sur son déroulement. Les facteurs de risques comme la criticité des tâches, la disponibilité des ressources humaines et matérielles, les délais de recrutement ou encore la disponibilité des locaux doivent être répertoriés. Leurs

conséquences potentielles doivent être estimées et classées par ordre d'importance et d'impact sur le projet. Que se passe-t-il s'il manque une ressource (locaux indisponibles ou inaccessibles, membre(s) de l'équipe absent(s), si l'on découvre des coûts cachés, si un fournisseur ne peut fournir du matériel ou une prestation dans les temps, si la technologie sélectionnée à l'origine ne fait finalement pas l'affaire… ? Autant de questions auxquelles il faut apporter une réponse tout au long du projet. Ces risques peuvent être matérialisés sur une matrice des risques, avec en abscisse la gravité du risque et en ordonnée la probabilité d'apparition du problème. Cette matrice permet visuellement d'identifier les risques majeurs et doit être mise à jour tout au long du projet car les risques évoluent avec le temps. Par exemple, l'absence d'un développeur dans l'équipe peut représenter un risque mineur au début du projet, et devenir un risque majeur après quelques semaines et quelques mois lorsque ce développeur aura acquis des connaissances indispensables à l'équipe.

II.3.2. Liste des activités de notre projet

Code	Libellé Activité	Activité Ant.	Durée/ Jour	Coût/Dollar(PT)
a	*Etude de l'existant*	-	*10*	*250*
b	*Proposition de la solution*	*a*	*5*	*100*
c	*conception du système d'information*	*b*	*15*	*250*
d	*Développement*	*b,c*	*20*	*1500*
e	*Proposition et choix du matériel*	*d*	*3*	*100*
f	*commande du matériel et logiciel*	*e*	*10*	*2500*
g	*Configuration du serveur*	*f*	*5*	*500*
h	*Implémentation*	*g*	*2*	*300*
i	*Test*	*h*	*3*	*200*
j	*Formation*	*h, i*	*10*	*800*

a. Matrice d'antériorité

Pour établir cette matrice nous allons créer un tableau à deux entrées identiques : la liste des tâches, suivie d'un tableau comportant des colonnes de niveaux

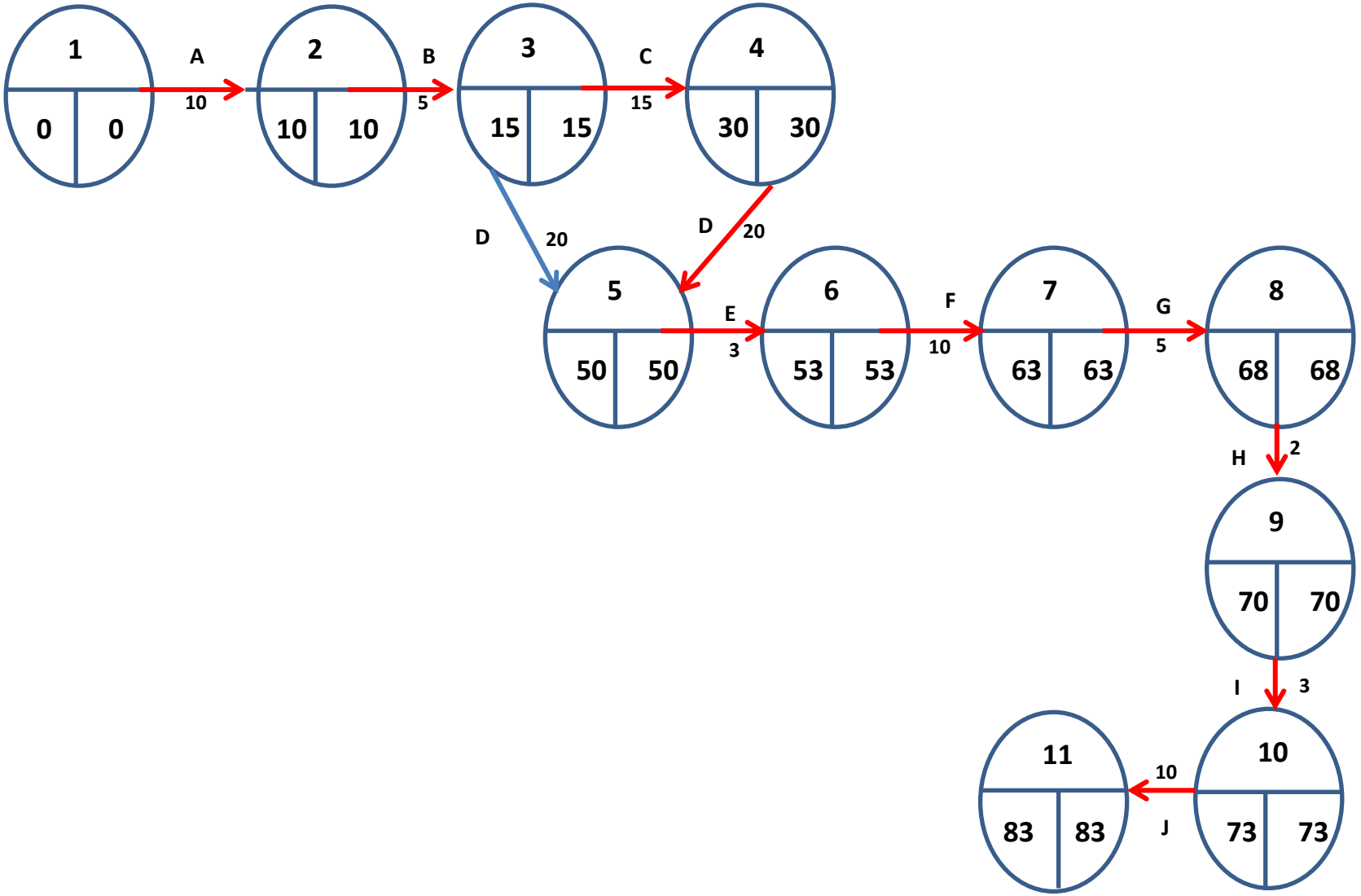
		IL FAUT AVOIR TERMINE										**Niveaux**									
		A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
POUR FAIRE	**A**											**0**									
	B	**1**										**1**	**0**								
	C		**1**									**1**	**1**	**0**							
	D		**1**	**1**								**2**	**2**	**1**	**0**						
	E				**1**							**1**	**1**	**1**	**1**	**0**					
	F					**1**						**1**	**1**	**1**	**1**	**1**	**0**				
	G						**1**					**1**	**1**	**1**	**1**	**1**	**1**	**0**			
	H							**1**				**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**		
	I								**1**			**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	
	J								**1**	**1**		**2**	**2**	**2**	**2**	**2**	**2**	**2**	**2**	**1**	**0**
												A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**

On peut résumer le résultat dans ce tableau :

Niveaux	1	2	3	4	5	6	7	8	9	10
Tâches	A	B	C	D	E	F	G	H	I	J

Le premier tableau représenté ci haut, comprend les tâches que nous devons réaliser dans notre projet et leurs dépendances en les exécutants, tandis que le suivant reprend les tâches et leurs niveaux.

b. Diagramme de PERT



c. Les marges

A: 10-10-0=0

B: 15-5-10=0

C: 30-15-15=0

D: 50-20-30=0

E: 53-3-50=0

F: 63-10-53=0

G: 68-5-63=0

H: 70-2-68=0

I: 73-3-70=0

J: 83-10-73=0

Le chemin critique est A,B,C,D,E,F,G,H,I,J

d. Cout et Durée du Projet

Au regard de ce qui précède, notre projet a besoin de 83 jours et 6500 dollars pour qu'il soit complétement exécuté et implémenté.

II.4. Généralités sur le langage UML

UML (Unified Modeling Language) est un langage formel et normalisé en termes de modélisation objet. Son indépendance par rapport aux langages de programmation, aux domaines de l'application et aux processus, son caractère polyvalent et sa souplesse ont fait de lui un langage universel. En plus UML est essentiellement un support de communication, qui facilite la représentation et la compréhension de solution objet. Sa notation graphique permet d'exprimer visuellement une solution objet, ce qui facilite la comparaison et l'évaluation des solutions. L'aspect de sa notation, limite l'ambigüité et les incompréhensions.

UML fournit un moyen astucieux permettant de représenter diverses projections d'une même représentation grâce aux vues :

Une vue est constituée d'un ou plusieurs diagrammes. On distingue deux types de vues:

La vue statique, permettant de représenter le système physiquement :

- Diagrammes de classes: représentent des collections d'éléments de modélisation statiques (classes, paquetages...), qui montrent la structure d'un modèle.
- Diagrammes d'objets: ces diagrammes montrent des objets (instances classes dans un état particulier) et des liens (relations sémantiques) entre objets.

- Diagrammes de composants: permettent de décrire l'architecture physique statique d'une application en termes de modules : fichiers sources, librairie exécutables, etc.
- Diagrammes de déploiement: montrent la disposition physique du matériel qui compose le système et la répartition des composants sur ce matériel.

La vue dynamique, montrant le fonctionnement du système :

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

II.5. Modélisation avec le langage UML

II.2.5.1. Diagrammes de cas d'utilisation

Les diagrammes de cas d'utilisation (DCU) sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel. Ils sont utiles pour des présentations auprès de la direction ou des acteurs d'un projet, mais pour le développement, les cas d'utilisation sont plus appropriés. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (humain ou machine) et un système. Il est une unité significative de travail. Dans un diagramme de cas d'utilisation, les utilisateurs sont appelés acteurs (actors), ils interagissent avec les cas d'utilisation (use cases) [4]. UML définit une notation graphique pour représenter les cas d'utilisation, cette notation est appelée diagramme de cas d'utilisation. UML ne définit pas de standard pour la forme écrite de ces cas d'utilisation, et en conséquence il est aisé de croire que cette notation graphique suffit à elle seule pour décrire la nature d'un cas d'utilisation. Dans les faits, une notation graphique peut seulement donner une vue générale simplifiée d'un cas ou d'un ensemble de cas d'utilisation.

Les diagrammes de cas d'utilisation sont souvent confondus avec les cas d'utilisation. Bien que ces deux concepts soient reliés, les cas d'utilisation sont bien plus détaillés que les diagrammes de cas d'utilisation.

Cas d'utilisation

Ils permettent de décrire l'interaction entre l'acteur et le système. L'idée forte est de dire que l'utilisateur d'un système logiciel a un objectif quand il utilise le système ! Le cas d'utilisation est une description des interactions qui vont permettre à l'acteur d'atteindre son objectif en utilisant le système. Les cas d'utilisation sont représentés par une ellipse sous-titrée par le nom du cas d'utilisation (éventuellement le nom est placé dans l'ellipse). Un acteur et un cas d'utilisation sont mis en relation par une association représentée par une ligne.

Le plus souvent, le diagramme des cas est établi par la maîtrise d'ouvrage (MOA) d'un projet lors de la rédaction du cahier des charges afin de transmettre les besoins des utilisateurs et les fonctionnalités attendues associées à la maîtrise d'œuvre (MOE).

Acteurs

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

Relations

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

Inclusions

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

Extensions

Les extensions (*extend*) représentent des prolongements logiques de certaines tâches sous certaines conditions. Autrement dit un cas d'utilisation A étend un cas d'utilisation B lorsque le cas d'utilisation A peut être appelé au cours de l'exécution du cas d'utilisation B. Elle est

représentée par une flèche en pointillée avec le terme *extend*. Ce type de relation peut être utile pour traiter des cas particuliers ou fonctions optionnelles, préciser les objectifs, ou encore pour tenir compte de nouvelles exigences au cours de la maintenance du système et de son évolution.

Généralisations

La troisième relation est la relation de généralisation ou spécialisation. Le cas d'utilisation A est une généralisation de B, si B est un cas particulier de A c'est-à-dire lorsque A peut-être substitué par B pour un cas précis. Ces relations sont des traits pleins terminés par une flèche en triangle.

Présentation de nos diagrammes

Vu le nombre important d'acteurs et les cas d'utilisation à représenter, nous nous abstenons de représenter un seul diagramme de cas d'utilisation global, cependant nous optons pour les diagrammes de cas d'utilisation partiels (de chaque acteur).

A. **Le diagramme de cas d'utilisation de l'acteur « Super User »**

La figure suivante montre les actions du « super utilisateur » dans le système.

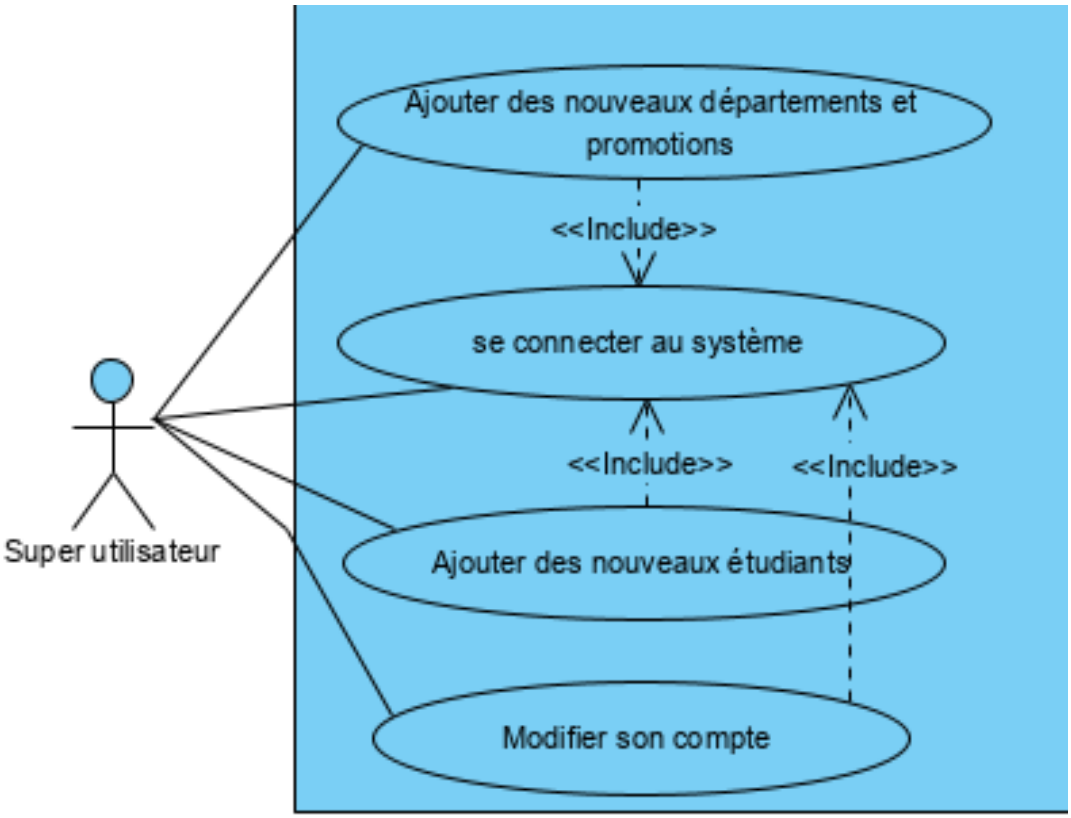


Figure 1.*Diagramme de cas d'utilisation d'un super user*

B. Le diagramme de cas d'utilisation de l'acteur « Sections ou facultés »

La figure ci-dessous montre les tâches de la section ou faculté dans le système.

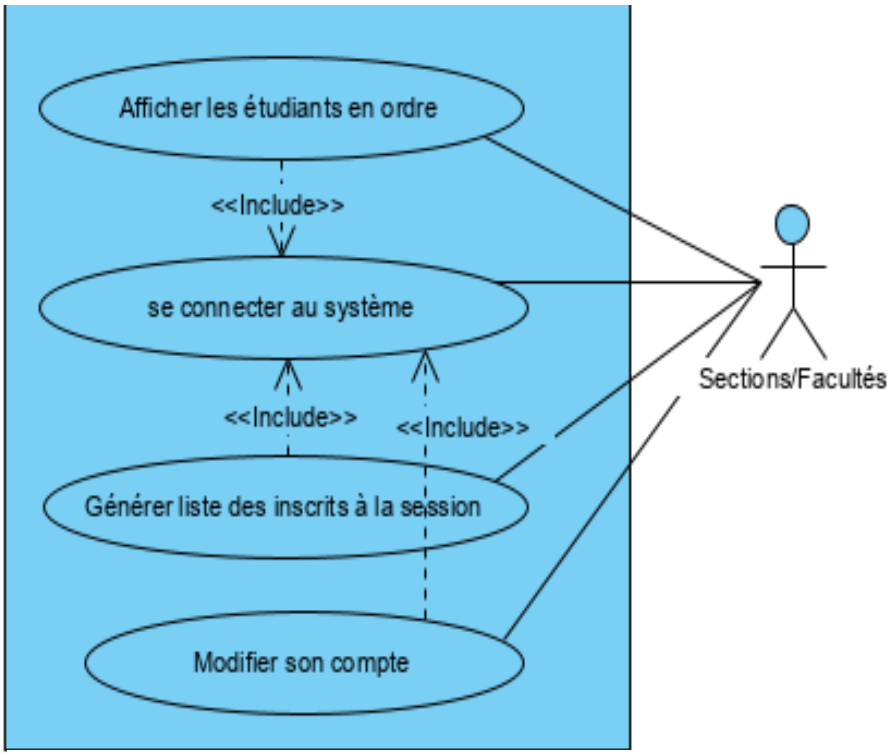


Figure 2.*Diagramme de cas d'utilisation de chef de section*

C. Le diagramme de cas d'utilisation de l'acteur « Secrétaire général académique »

La figure suivante montre les actions de l'acteur « Secrétaire général académique » dans le système.

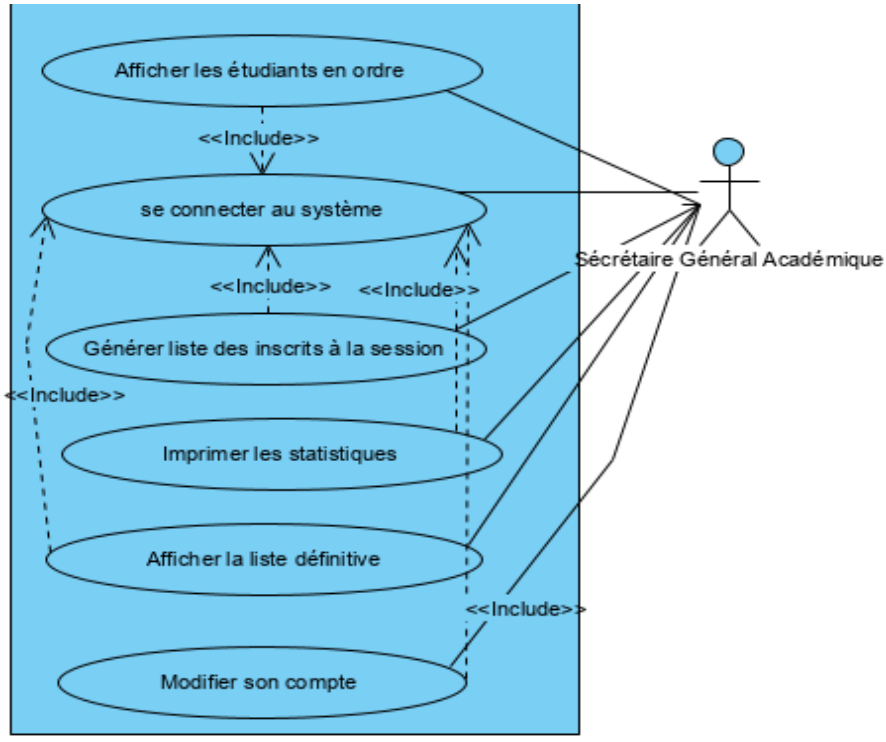


Figure 3. Diagramme de cas d'utilisation du SGAC

D. Le diagramme de cas d'utilisation de l'acteur « Directeur général »

La figure ci-dessous montre les actions du « Directeur général » dans le système.

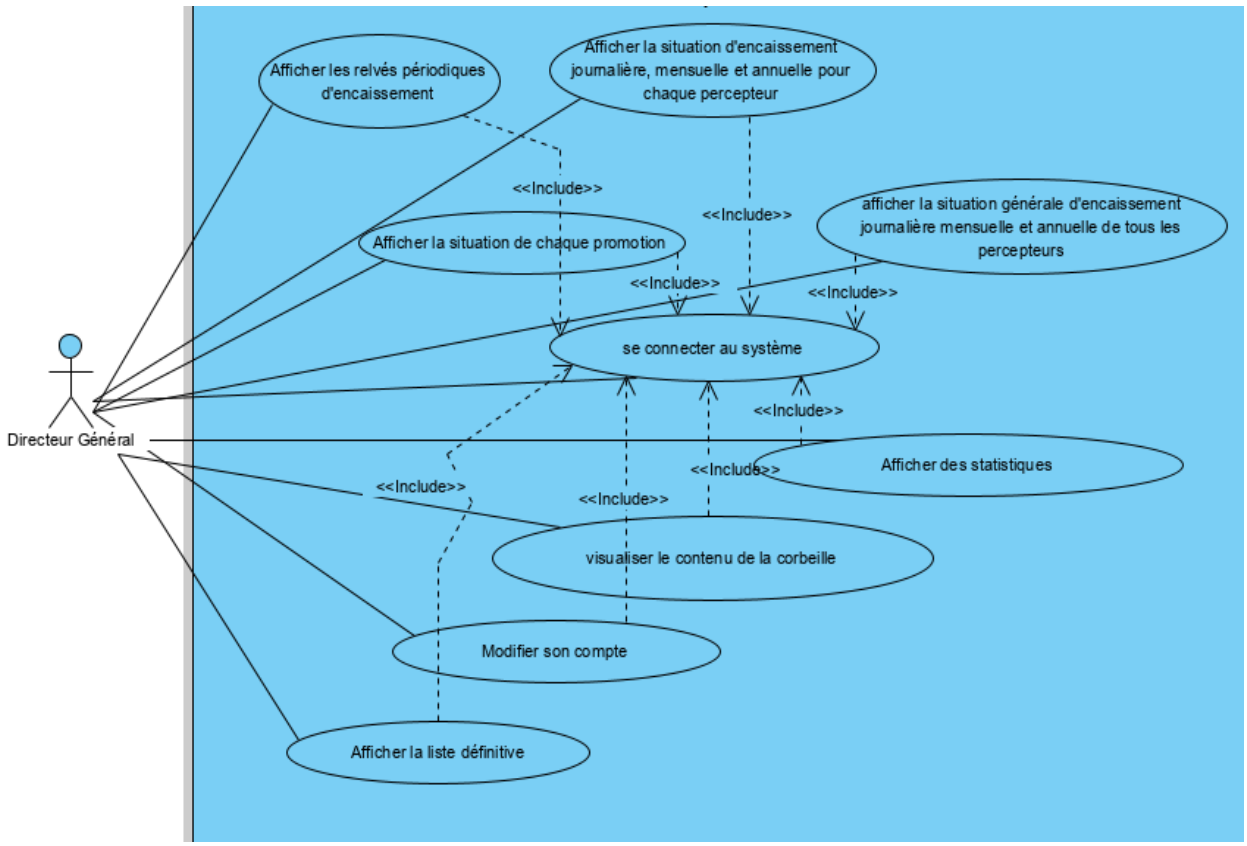


Figure 4.*Diagramme de cas d'utilisation du Directeur Général*

E. Le diagramme de cas d'utilisation de l'acteur « Percepteur »

La figure suivante montre les tâches du « Percepteur » dans le système

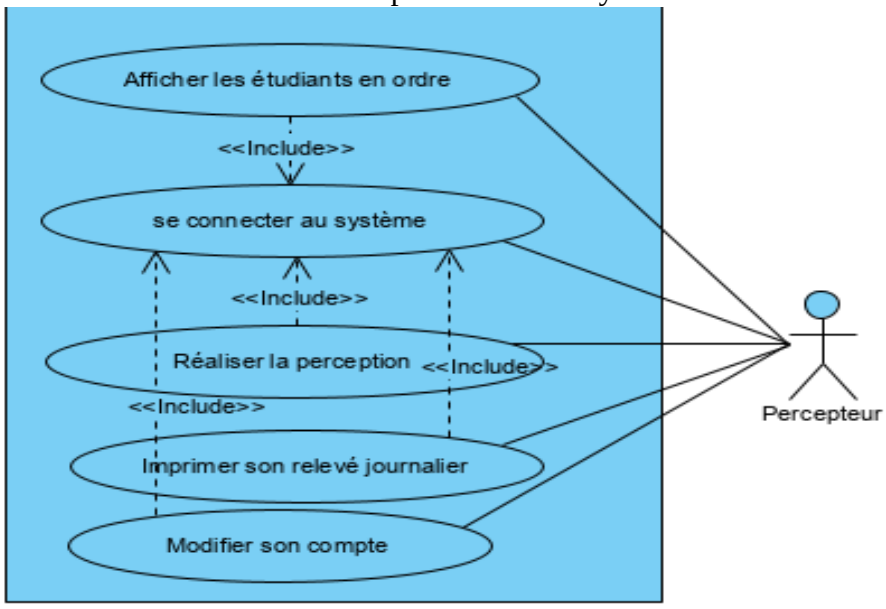


Figure 5.*Diagramme de cas d'utilisation du Percepteur*

F. Le diagramme de cas d'utilisation de l'acteur « Caissier »

La figure ci-dessous montre les actions de l'acteur « Caissier » dans le système.

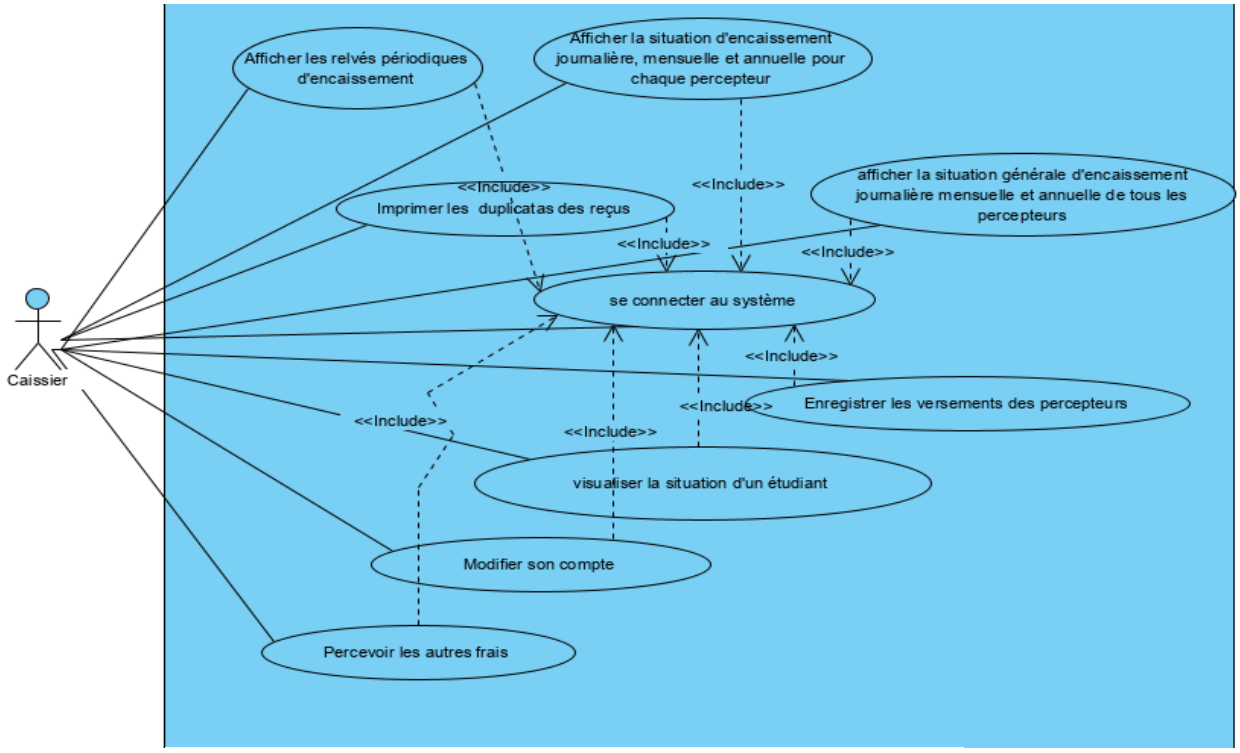


Figure 6.*Diagramme de cas d'utilisation du Caissier*

G. Le diagramme de cas d'utilisation de l'acteur « Administrateur de Budget »

La figure suivante montre les actions de « l'administrateur de Budget » dans le système.

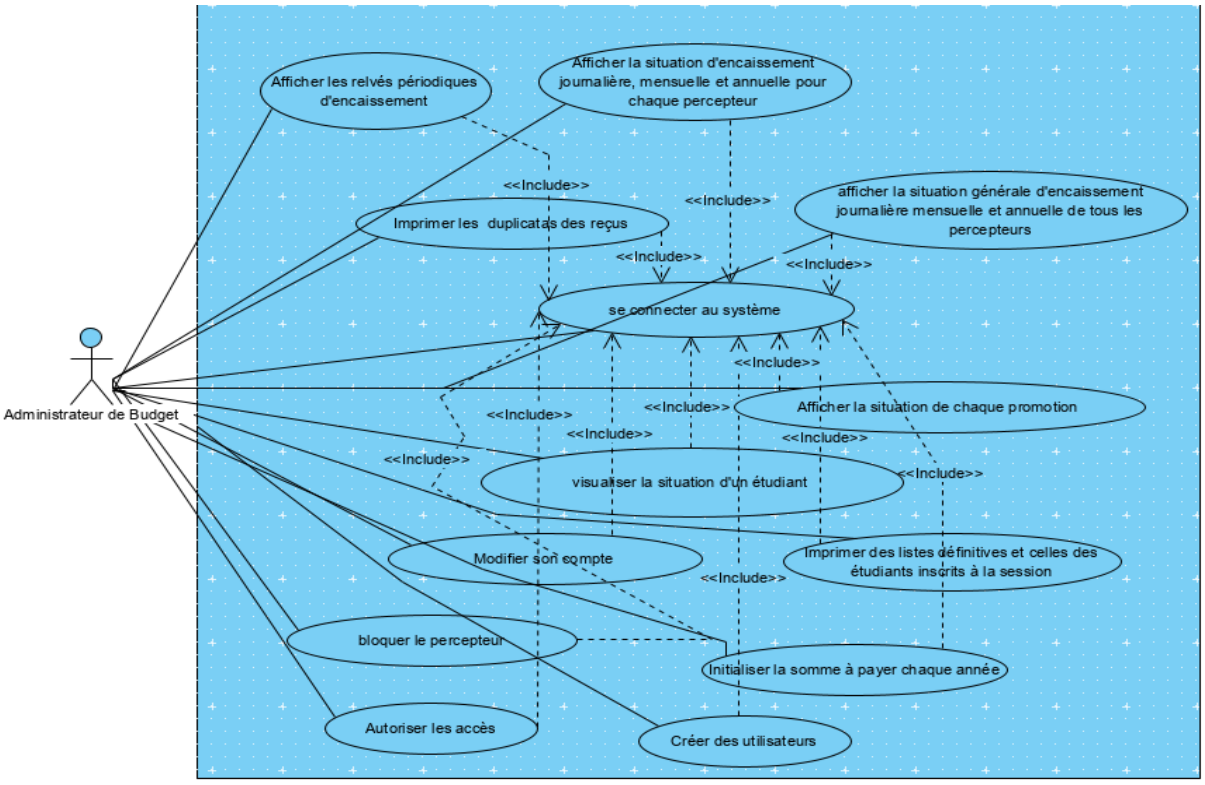


Figure 7.*Diagramme de cas d'utilisation de l'AB*

Eu égard de la complexité de ces diagrammes des cas d'utilisation (partiels) ci hauts, nous nous servons uniquement de certains cas d'utilisation que nous estimons importants pour la représentation des diagrammes d'activités et de séquence.

II.5.2. Diagrammes d'activités

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallélisables (multi-threads ou multi-processus). Le diagramme d'activité est également utilisé pour décrire un flux de travail (workflow). Un diagramme d'activité permet de modéliser un processus interactif, global ou partiel pour un système donné (logiciel, système d'information). Il est recommandable pour exprimer une dimension temporelle sur une partie du modèle, à partir des diagrammes de classes ou de cas d'utilisation, par exemple [5]. Le diagramme d'activité est une représentation proche de l'organigramme ; la description d'un cas d'utilisation par un diagramme d'activité correspond à sa traduction algorithmique. Une activité est l'exécution d'une partie du cas d'utilisation, elle est représentée par un rectangle aux bords arrondis.

Le diagramme d'activité est sémantiquement proche des diagrammes de communication (appelés diagramme de collaboration en UML 1), ou d'état-transitions, ces derniers offrant une vision microscopique des objets du système.

- **Se connecter au système**

La figure suivante montre le processus pour se connecter au système

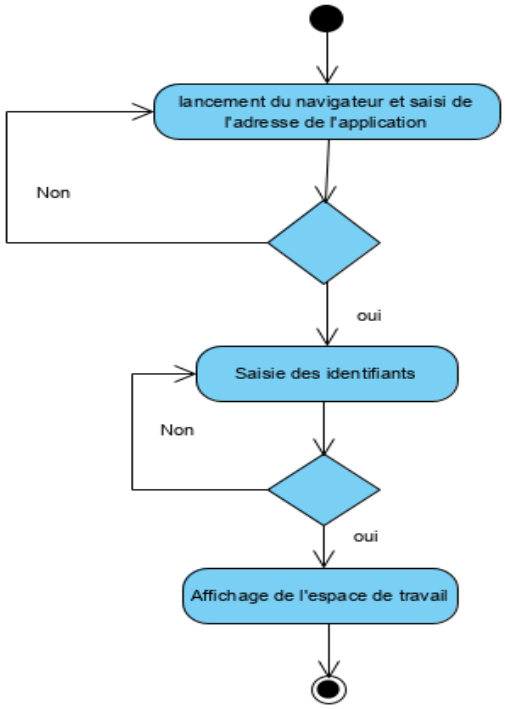


Figure 8. *Diagramme d'activité de connexion au système*

- **Ajouter des nouveaux étudiants**

La figure ci-dessous montre le processus d'ajout d'étudiants.

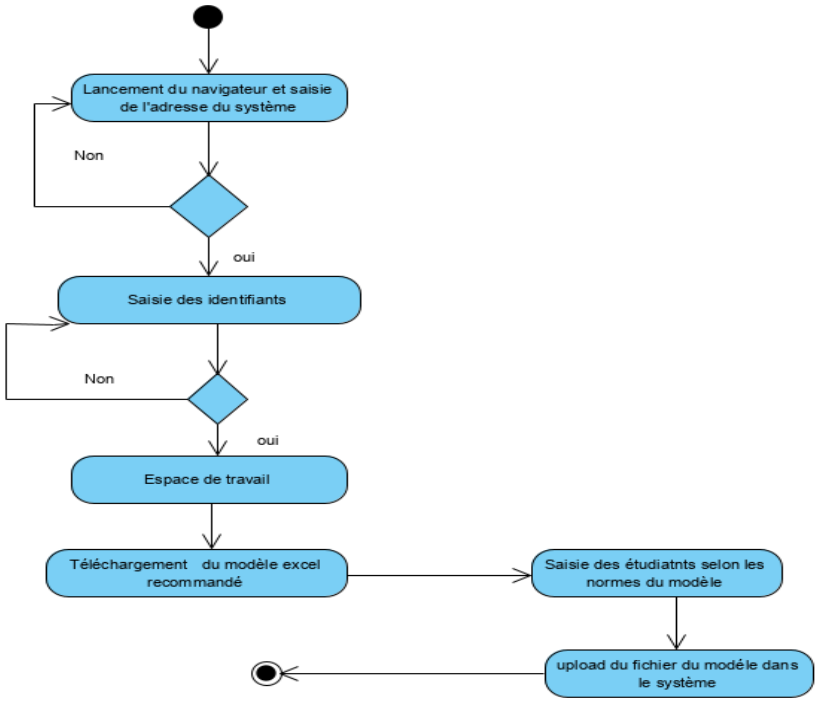


Figure 9. *Diagramme d'activité d'ajout de nouveaux étudiants*

- **Afficher la situation d'encaissement journalière, mensuelle et annuelle pour chaque percepteur**

La figure suivante montre le processus d'affichage de la situation d'encaissement journalière, mensuelle et annuelle.

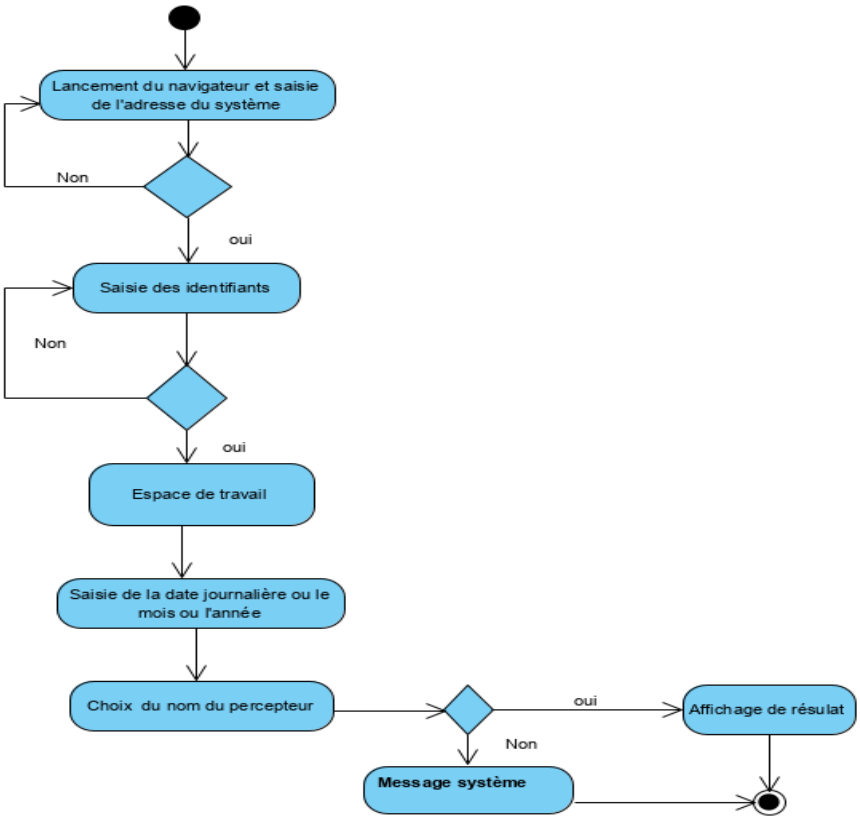


Figure 10.*Diagramme d'activité de la situation d'encaissement journalière, mensuelle et annuelle*

- **Afficher les relevés périodiques d'encaissement**

La figure ci-dessous montre le processus d'affichage des relevés périodiques d'encaissement.

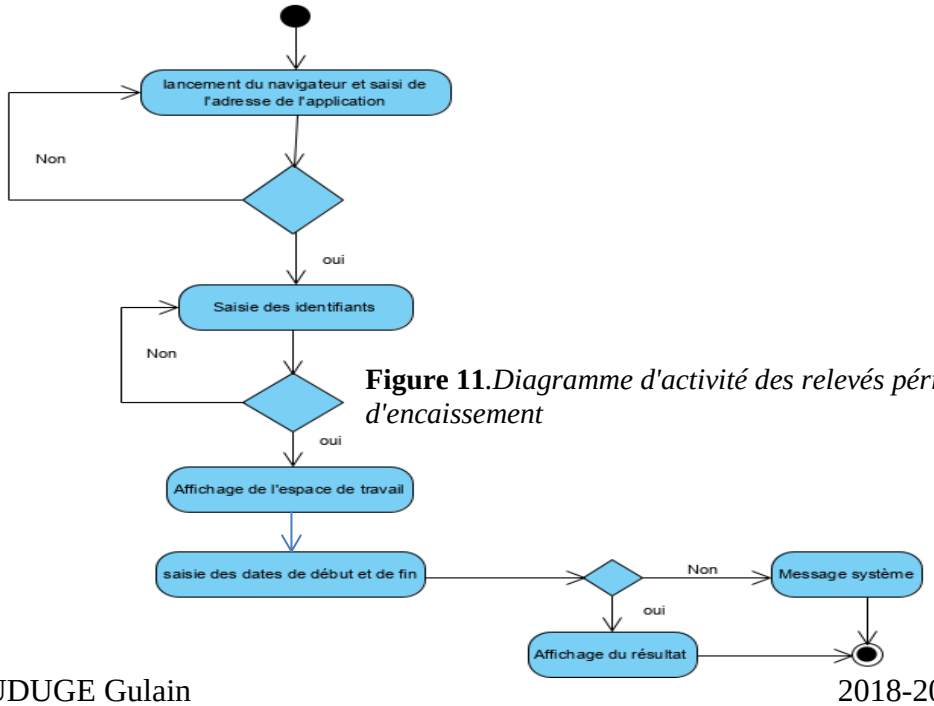


Figure 11.*Diagramme d'activité des relevés périodiques d'encaissement*

- **Afficher la situation de chaque promotion**

La figure suivante montre le processus d'affichage de la situation de chaque promotion.

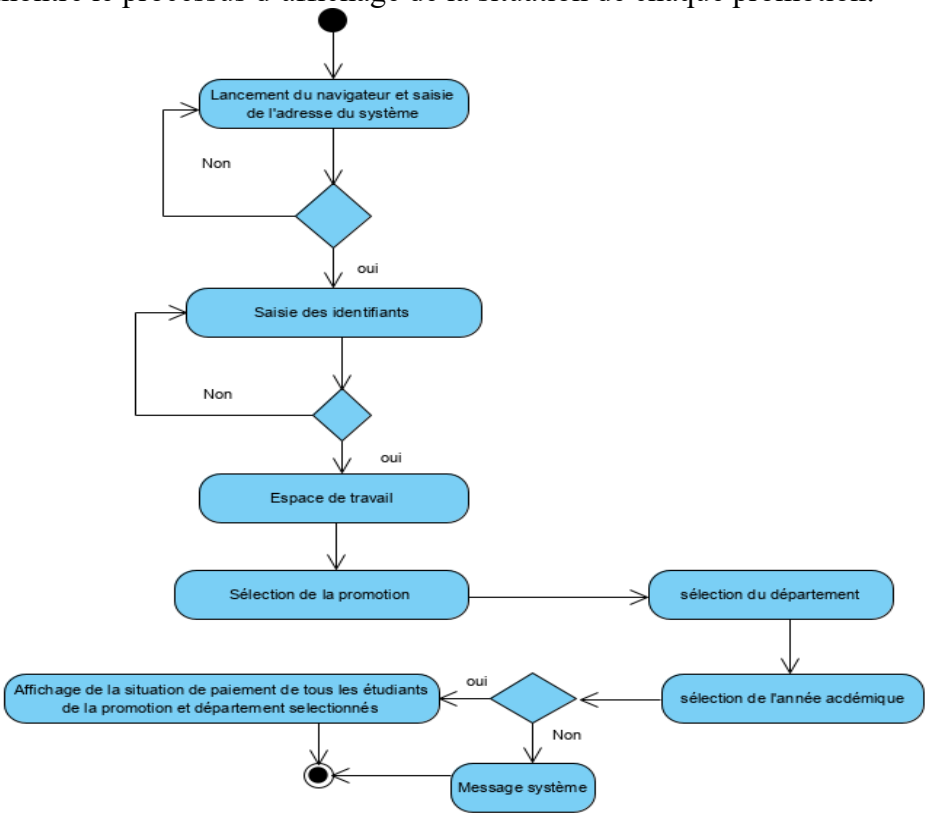


Figure 12.*Diagramme d'activité de la situation de chaque promotion*

- **Visualiser la corbeille**

La figure suivante montre le processus de la visualisation de la corbeille.

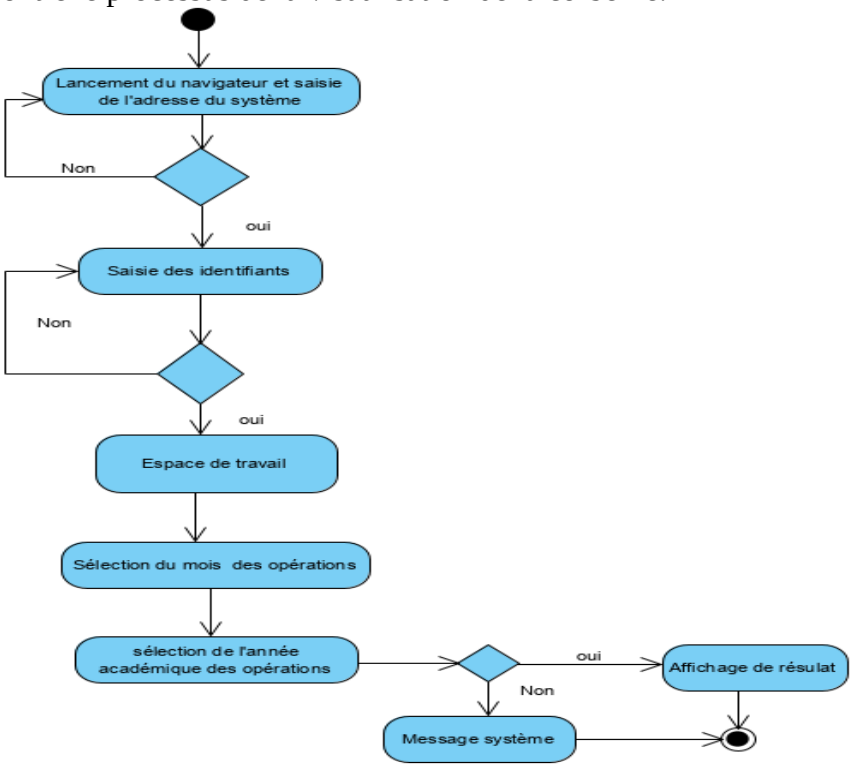


Figure 14. *Diagramme d'activité visualiser la corbeille*

- **Afficher la liste définitive**

La figure suivante montre le processus d'affichage de la liste définitive.

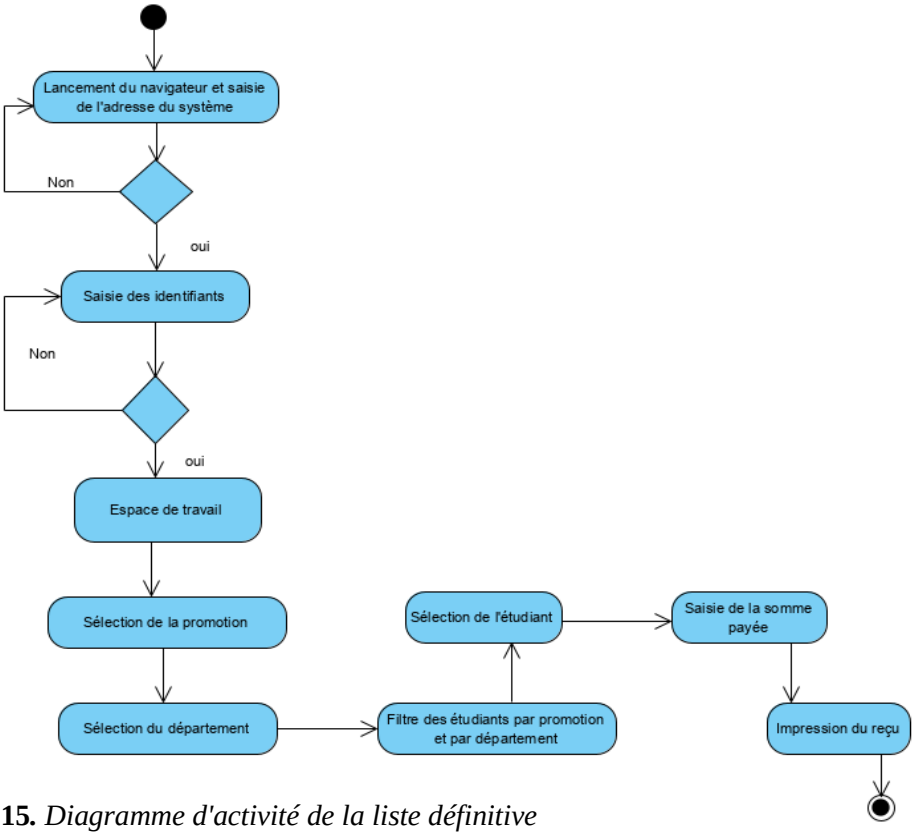


Figure 15. *Diagramme d'activité de la liste définitive*

- **Réaliser la perception**

La figure suivante montre le processus de la réalisation de la perception

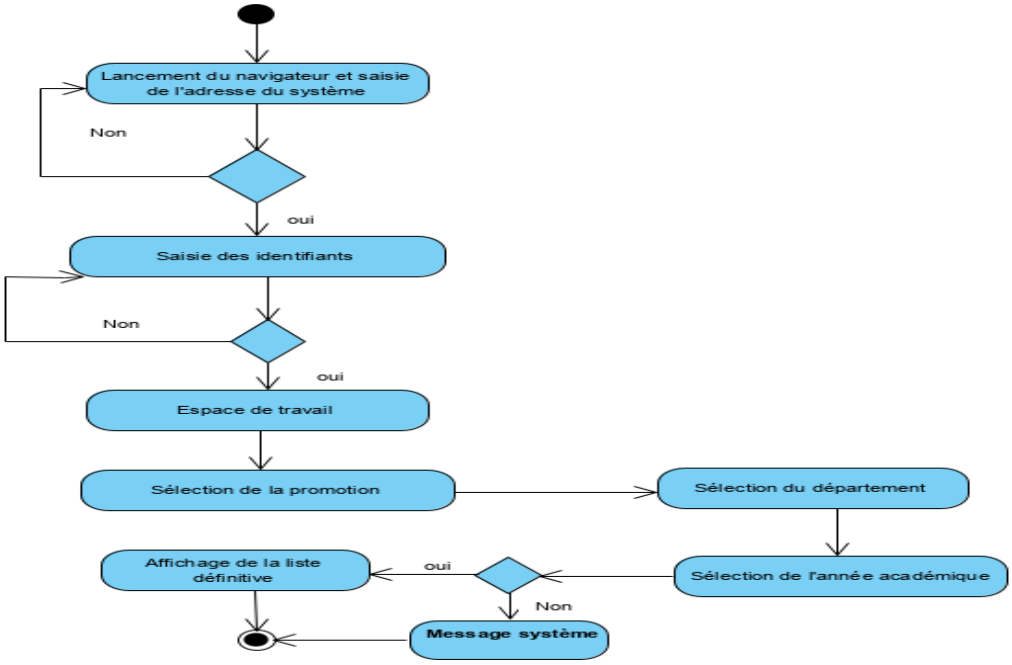


Figure 16.*Diagramme d'activité de la perception des frais*

II.5.3. Diagrammes de séquence

Le diagramme de séquence permet de montrer les interactions d'objets dans le cadre d'un scénario d'un Diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets[6].

La dimension verticale du diagramme représente le temps, permettant de visualiser l'enchaînement des actions dans le temps, et de spécifier la naissance et la mort d'objets. Les périodes d'activité des objets sont symbolisées par des rectangles, et ces objets dialoguent à l'aide de messages.

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

- message simple : le message n'a pas de spécificité particulière d'envoi et de réception.
- message avec durée de vie : l'expéditeur attend une réponse du récepteur pendant un certain temps et reprend ses activités si aucune réponse n'a lieu dans un délai prévu.
- message synchrone : l'expéditeur est bloqué jusqu'au signal de prise en compte par le destinataire. Les messages synchrones sont symbolisés par des flèches barrées.
- message asynchrone : le message est envoyé, l'expéditeur continue son activité que le message soit parvenu ou pris en compte ou non. Les messages asynchrones sont symbolisés par des demi-flèches.
- message dérobant : le message est mis en attente sur une liste d'attente de traitement chez le récepteur.

- **Se connecter au système**

La figure suivante montre l'interaction de l'utilisateur avec le système pendant la connexion.

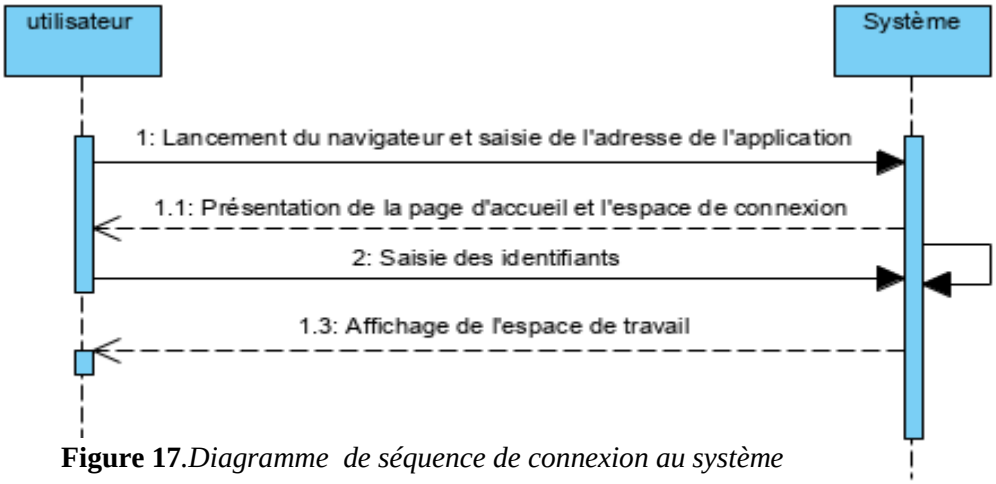


Figure 17.*Diagramme de séquence de connexion au système*

- **Ajouter de nouveaux étudiants**

La figure suivante montre l'interaction de l'utilisateur avec le système lors de l'ajout de nouveaux étudiants.

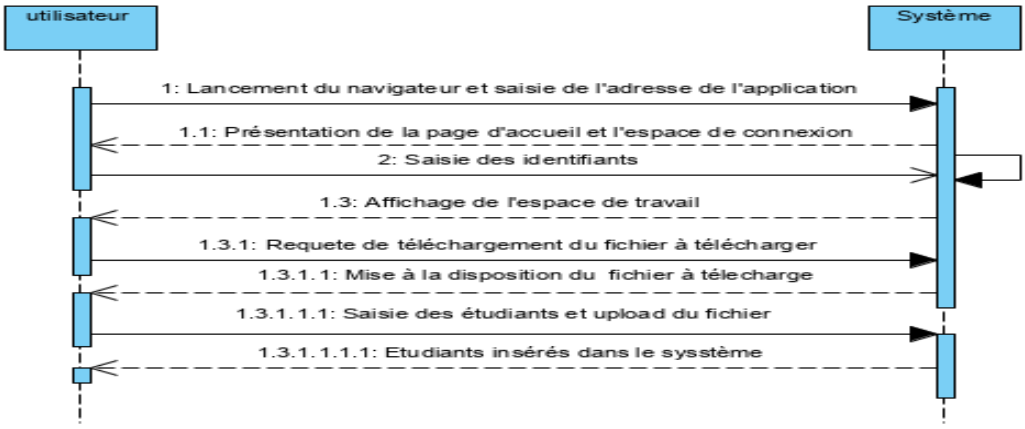


Figure 18.*Diagramme de séquence d'ajout de nouveaux étudiants*

- **Afficher la situation d'encaissement journalière, mensuelle et annuelle pour chaque percepteur**

La figure suivante montre l'interaction de l'utilisateur avec le système pendant l'affichage de la situation d'encaissement, journalière, etc.

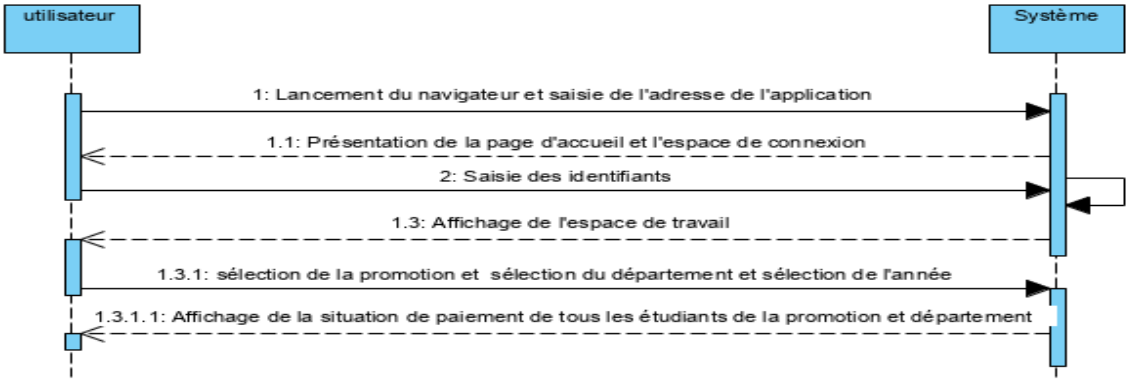


Figure 19.*Diagramme de séquence de la situation d'encaissement journalière, mensuelle et annuelle*

- **Afficher les relevés périodiques d'encaissement**

La figure suivante montre l'interaction de l'utilisateur avec le système pendant l'affichage des relevés périodiques.

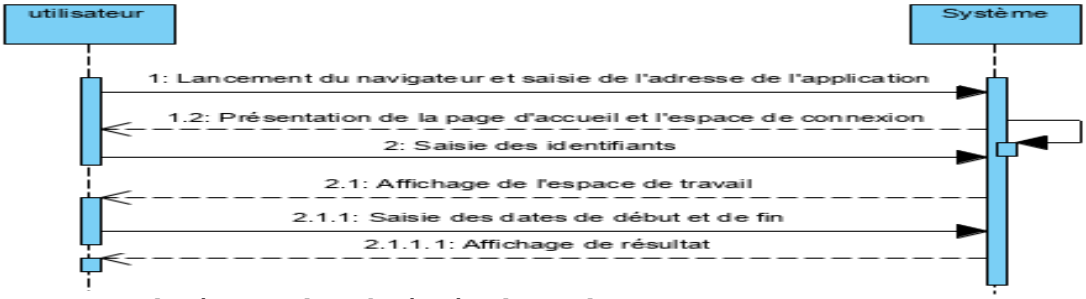


Figure 20.*Diagramme de séquence des relevés périodiques d'encaissement*

- **Afficher la situation de chaque promotion**

La figure suivante montre l'interaction de l'utilisateur avec le système pendant l'affichage de la situation de chaque promotion.

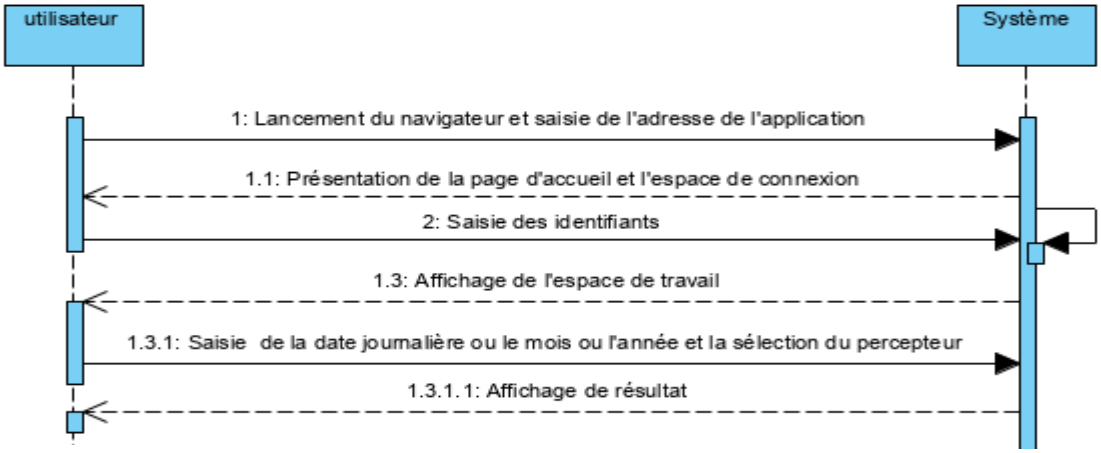


Figure 21.*Diagramme de séquence de la situation de chaque promotion*

- **Visualiser la corbeille**

La figure suivante montre l'interaction de l'utilisateur avec le système pendant la visualisation de la corbeille.

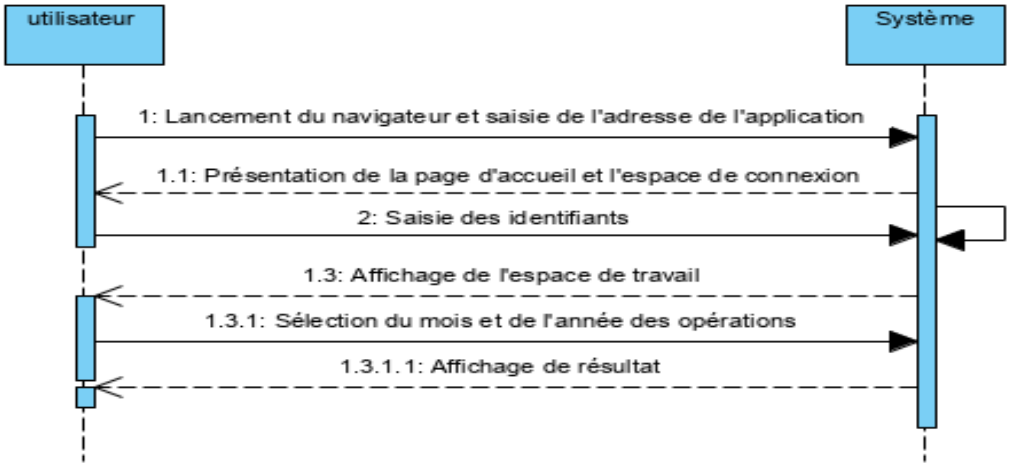


Figure 22.*Diagramme de séquence de la visualisation de la corbeille*

- **Afficher la liste définitive**

La figure suivante montre l'interaction de l'utilisateur avec le système pendant l'affichage de la liste définitive.

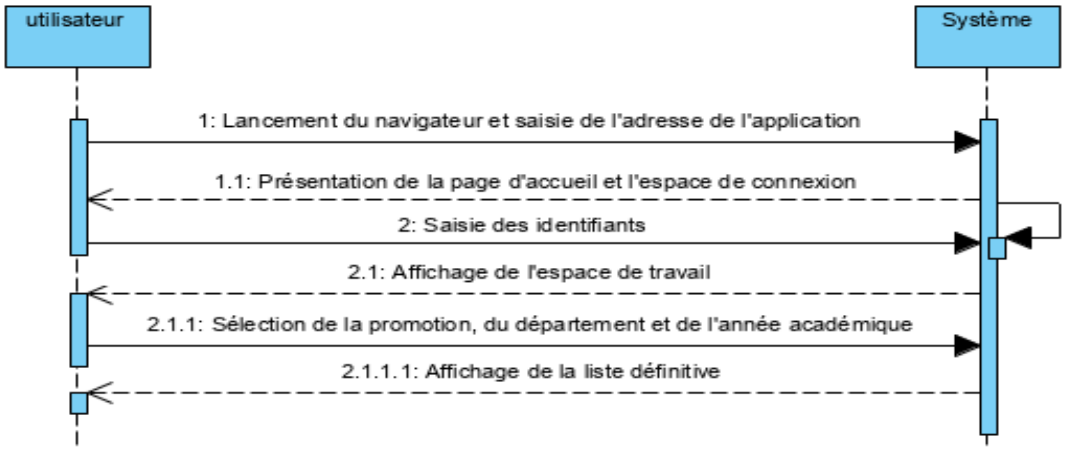


Figure 23.*Diagramme de séquence de la liste définitive*

- **Réaliser la perception**

La figure suivante montre l'interaction de l'utilisateur avec le système pendant la réalisation de la perception.

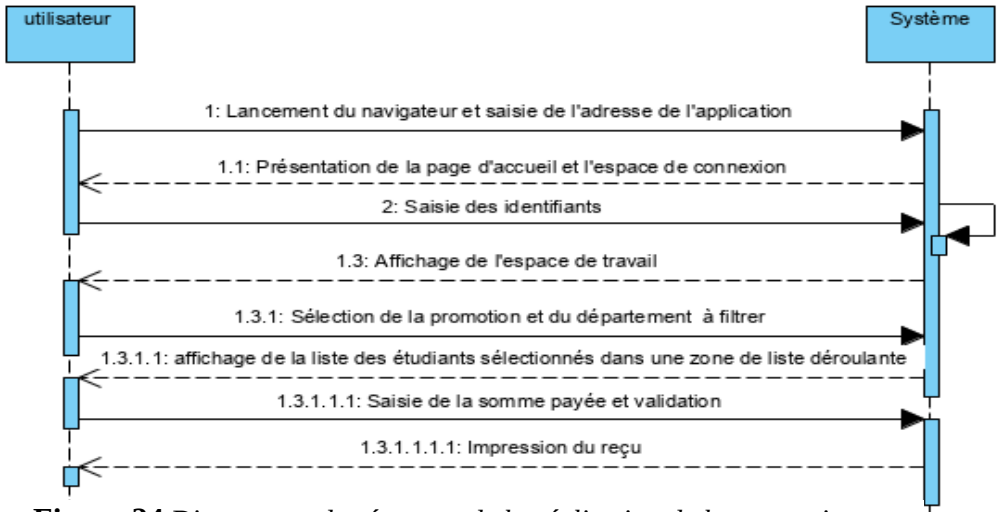


Figure 24.*Diagramme de séquence de la réalisation de la perception*

II.5.4. Diagramme des classes

Le diagramme des classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques. Une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets. Les éléments de cet ensemble sont les instances de la classe. Une classe est un

ensemble de fonctions et de données (attributs) qui sont liées ensemble par un champ sémantique. Les classes sont utilisées dans la programmation orientée objet. Elles permettent de modéliser un programme et ainsi de découper une tâche complexe en plusieurs petits travaux simples. Les classes peuvent être liées entre elles grâce au mécanisme d'héritage qui permet de mettre en évidence des relations de parenté. D'autres relations sont possibles entre des classes, chacune de ces relations est représentée par un arc spécifique dans le diagramme de classes [7].

Une classe est représentée par un rectangle séparé en trois parties :

- la première partie contient le nom de la classe
- la seconde contient les attributs de la classe
- la dernière contient les méthodes de la classe

La seconde et la dernière représentent le comportement de la classe.

La figure suivante représente les classes qui nous ont permis de réaliser notre système d'information.

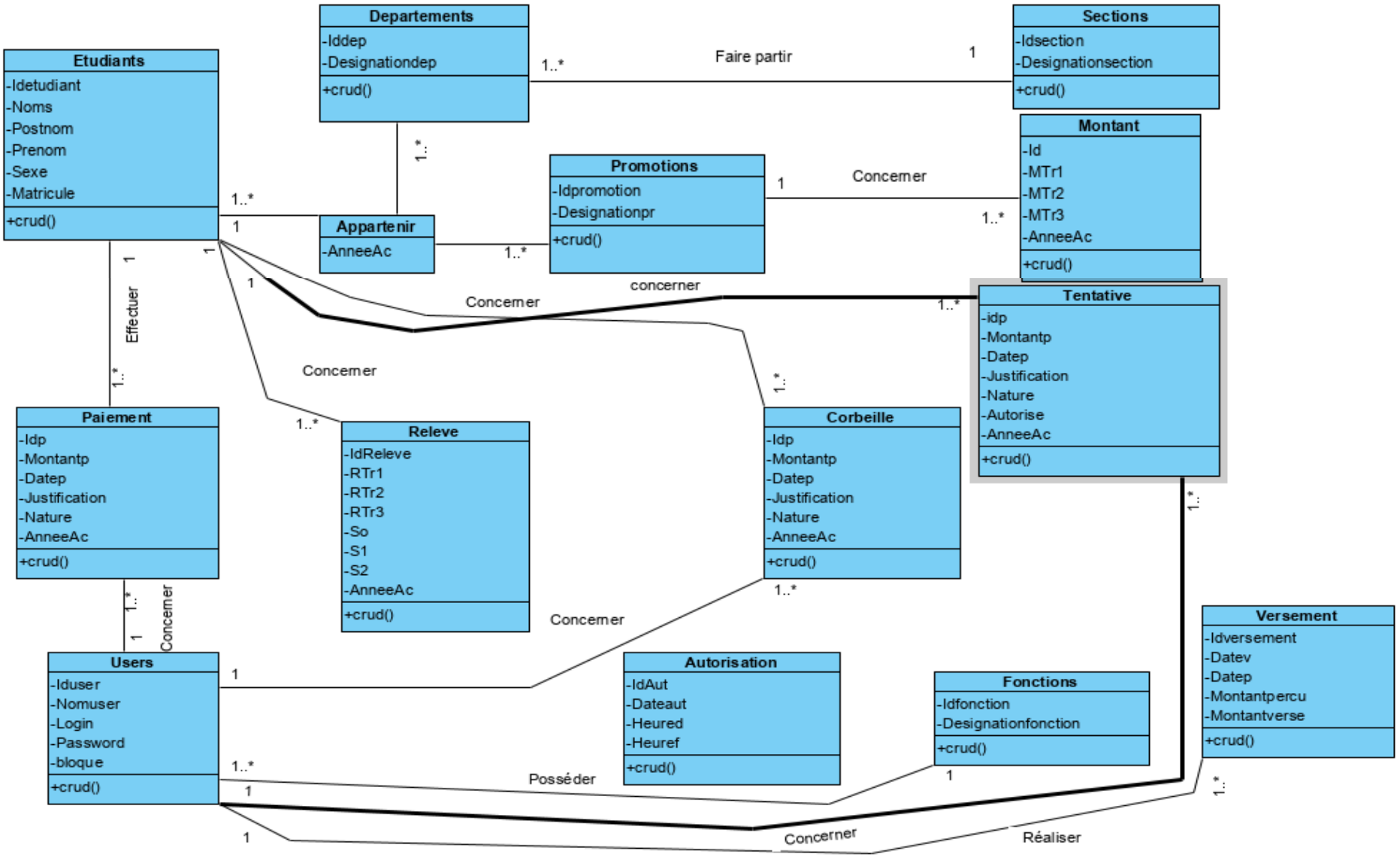


Figure 25.*Diagramme des classes de notre système*

II.5.5. Modèle Physiques des données

La figure suivante décrit textuellement l'implémentation du modèle logique des données dans le système, c'est pourquoi nous nous sommes abstenu de le présenter.

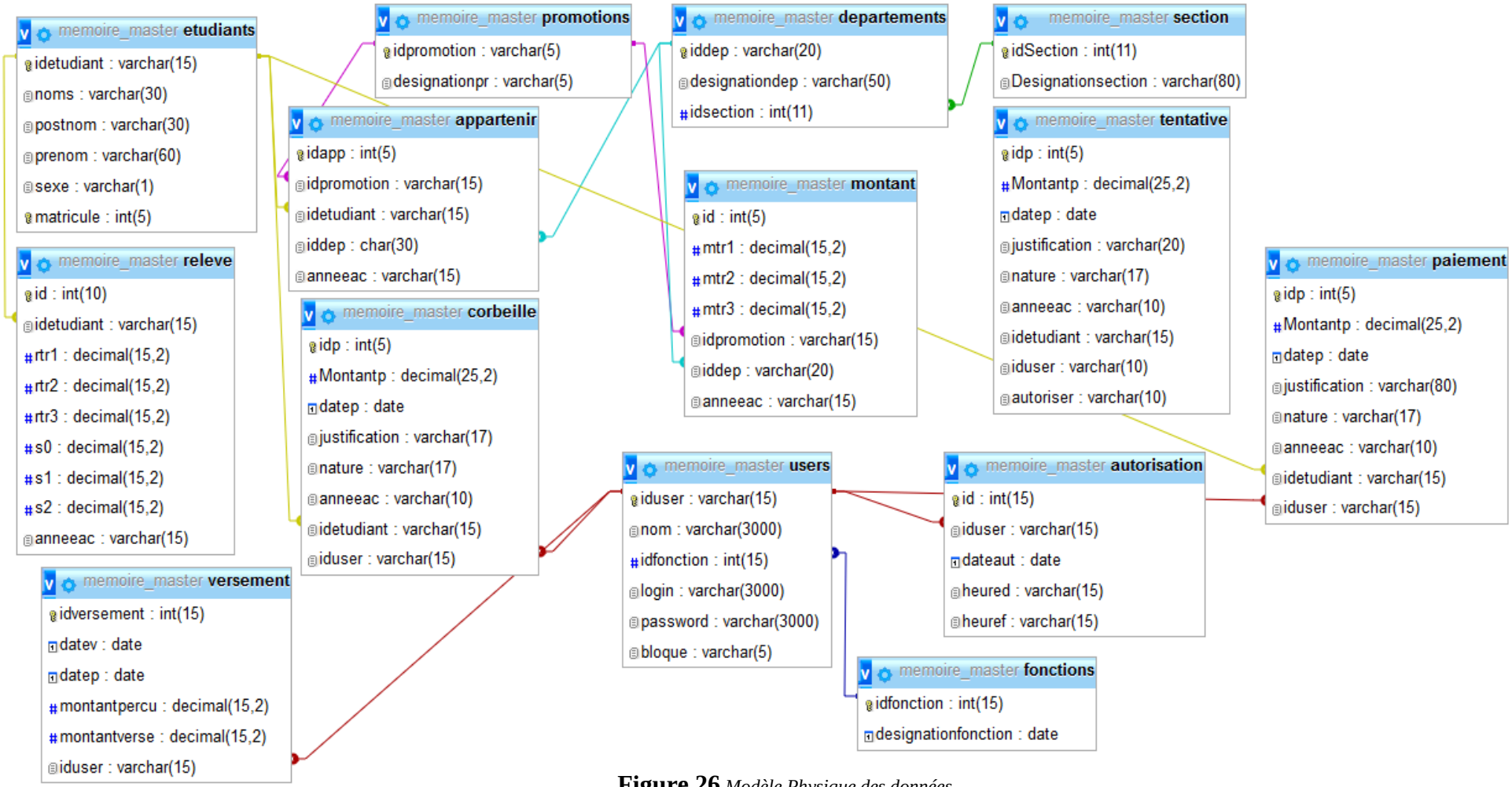


Figure 26.*Modèle Physique des données*

Conclusion partielle

Nous venons d'aborder deux notions essentielles dans ce chapitre, avec la première nous avons parlé de la gestion des projets où nous avons essayé d'épingler les différentes phases d'élaboration de projet et en fin nous avons abordé le point concernant la conception d'un système d'information avec UML.

CHAPITRE III. FORMALISMES MATHEMATIQUES AVEC ALGEBRE RELATIONNELLE

III.1. Introduction

La représentation d'information sous forme relationnelle est intéressante car les fondements mathématiques du relationnel, outre qu'ils permettent une modélisation logique simple et puissante, fournissent également un ensemble de concepts pour manipuler formellement l'information ainsi modélisée. Nous avons opté l'algèbre relationnelle car elle inclut d'autres modèles de représentation et de manipulation des données.

L'algèbre relationnelle est composée par les cinq opérateurs de base et les trois opérateurs additionnels suivants : [8]

Opérateurs de base

- Sélection
- Projection
- Union
- Différence
- Produit cartésien

Opérateurs additionnels

- Intersection
- Jointure

Fondamental: Algèbre relationnelle et SQL

Les questions formulées en algèbre relationnelle sont la base des questions formulées en SQL pour interroger une base de données relationnelle.

III.2. Opérateurs fondamentaux : projection, sélection et jointure

1. Sélection

La sélection est une opération unaire (c'est à dire portant sur une seule relation). La restriction de R1, étant donnée une condition C, produit une relation R2 de même schéma que R1 et dont les tuples sont les tuples de R1 vérifiant la condition C [9].

Exemple

Soit la relation suivante :

1 Etudiants (<u>idetudiant</u>, noms, postnom, prenom, sexe)

Soit les tuples suivants :

(Vu que notre système d'information est chiffré, nous allons nous servir des tables contenant les données déchiffrées)

idetudiant	noms	postnom	prenom	sexe
0007/12-13	MALUBA	Munyololo	Riphine	F
0119/12-13	BATUMIKE	Matabaro		M
0235/12-13	LUKULA	Basomine	Pascal	M
0236/12-13	FURAHA	Kongolo		M
0692/12-13	KABOZA	Murongo		M
0941/12-13	BATUMIKE	Banyanga	Eric	M
1191/12-13	MAKINDU	Ndume	Joseph	M

Soit l'opération suivante :

Affichez les étudiants du sexe « M »

- Requête en Algèbre relationnelle :

  $\sigma_{sexe='M'}(Etudiants)$

- Requête en SQL :

*Select * from etudiants where sexe='M'*

- Résultat:

idetudiant	noms	postnom	prenom	sexe
0119/12-13	BATUMIKE	Matabaro		M
0235/12-13	LUKULA	Basomine	Pascal	M
0236/12-13	FURAHA	Kongolo		M
0692/12-13	KABOZA	Murongo		M
0941/12-13	BATUMIKE	Banyanga	Eric	M
1191/12-13	MAKINDU	Ndume	Joseph	M

2. Projection

Définition : Projection

La projection est une opération unaire (c'est à dire portant sur une seule relation). La projection de R1 sur une partie de ses attributs {A1, A2, ...} produit une relation R2 dont le schéma est restreint aux attributs mentionnés en opérande, comportant les mêmes tuples que R1, et dont les doublons sont éliminés [10].

Remarque : Élimination des doublons

Après suppression d'une partie des attributs du schéma, la relation peut comporter des doublons. Étant donné que l'on ne pourrait plus identifier ces doublons les uns par rapport aux autres, la seule solution sensée est donc de considérer que deux doublons sont équivalents, et donc de n'en garder qu'un seul dans la relation résultante.

Exemple

En utilisant la relation et les tuples du premier exemple, on peut afficher seulement les noms et les post noms des étudiants.

- Requête Algèbre relationnelle :

  $\Pi_{noms,postnom}(Etudiants)$

- Requête SQL

  Select noms,postnom from etudiants

- Résultat :

noms	postnom
MALUBA	Munyololo
BATUMIKE	Matabaro
LUKULA	Basomine
FURAHA	Kongolo
KABOZA	Murongo
BATUMIKE	Banyanga
MAKINDU	Ndume

3. Produit

Définition : Produit cartésien

Le produit cartésien est une opération binaire (c'est à dire portant sur deux relations). Le produit de R1 par R2 (équivalent au produit de R2 par R1) produit une relation R3 ayant pour schéma la juxtaposition de ceux des relations R1 et R2 et pour tuples l'ensemble des combinaisons possibles entre les tuples de R1 et ceux de R2 [11].

Synonymes : Produit

Remarque

Le nombre de tuples résultant du produit de R1 par R2 est égal au nombre de tuples de R1 fois le nombre de tuples de R2.

Remarque

Le nombre de colonnes du produit de R1 par R2 est égal au nombre de colonnes de R1 plus le nombre de colonnes de R2

Exemple

A part la relation précédente, on ajoute une autre appelée paiement :

idp	Montantp	datep	justification	nature	annee	idetudiant	user
1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI

On demande de réaliser le produit de deux relations : Etudiants et paiement

- Requête en Algèbre relationnelle : EtudiantsXpaiement
- Requête SQL : select * from etudiants,paiement ou select * from etudiants cross join paiement
- Résultat :

idetudiant	noms	postnom	prenom	sexe	idp	Montantp	datep	justification	nature	annee	idetudiant	user
0007/12-13	MALUBA	Munyololo	Riphine	F	1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
0007/12-13	MALUBA	Munyololo	Riphine	F	2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
0007/12-13	MALUBA	Munyololo	Riphine	F	3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
0007/12-13	MALUBA	Munyololo	Riphine	F	4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI
0119/12-13	BATUMIKE	Matabaro		M	1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
0119/12-13	BATUMIKE	Matabaro		M	2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
0119/12-13	BATUMIKE	Matabaro		M	3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
0119/12-13	BATUMIKE	Matabaro		M	4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI
0235/12-13	LUKULA	Basomine	Pascal	M	1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
0235/12-13	LUKULA	Basomine	Pascal	M	2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
0235/12-13	LUKULA	Basomine	Pascal	M	3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
0235/12-13	LUKULA	Basomine	Pascal	M	4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI
0236/12-13	FURAHA	Kongolo		M	1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
0236/12-13	FURAHA	Kongolo		M	2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
0236/12-13	FURAHA	Kongolo		M	3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
0236/12-13	FURAHA	Kongolo		M	4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI
0692/12-13	KABOZA	Murongo		M	1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
0692/12-13	KABOZA	Murongo		M	2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
0692/12-13	KABOZA	Murongo		M	3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
0692/12-13	KABOZA	Murongo		M	4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI
0941/12-13	BATUMIKE	Banyanga	Eric	M	1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
0941/12-13	BATUMIKE	Banyanga	Eric	M	2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
0941/12-13	BATUMIKE	Banyanga	Eric	M	3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
0941/12-13	BATUMIKE	Banyanga	Eric	M	4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI
1191/12-13	MAKINDU	Ndume	Joseph	M	1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
1191/12-13	MAKINDU	Ndume	Joseph	M	2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
1191/12-13	MAKINDU	Ndume	Joseph	M	3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
1191/12-13	MAKINDU	Ndume	Joseph	M	4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI

4. Jointure

La jointure est une opération binaire (c'est à dire portant sur deux relations). La jointure de R1 et R2, étant donné une condition C portant sur des attributs de R1 et de R2, de même domaine, produit une relation R3 ayant pour schéma la juxtaposition de ceux des relations R1

et R2 et pour tuples l'ensemble de ceux obtenus par concaténation des tuples de R1 et de R2, et qui vérifient la condition C [12].

a. La jointure naturelle

Exemple

Avec les relations précédentes, on nous demande d'afficher les étudiants qui ont déjà payé

Soit l'opération suivante :[13]

- Requête en algèbre relationnelle :

$\pi_{Etudiants.idetudiant,noms,postnom,datep,montantp}(Etudiants \bowtie_{Etudiants.idetudiant=Paiement.idetudiant} Paiement)$

- Requête sql :

select etudiants.idetudiant,noms,postnom,datep,montantp from etudiants,paiement where etudiants.idetudiant=paiement.idetudiant

- Résultat :

idetudiant	noms	postnom	datep	montantp
0007/12-13	MALUBA	Munyololo	2019-11-08	50.00
0235/12-13	LUKULA	Basomine	2019-11-09	80.00
0235/12-13	LUKULA	Basomine	2019-11-10	80.00
1191/12-13	MAKINDU	Ndume	2019-11-13	111.50

b. Thêta-jointure

Pour illustrer la thêta-jointure, nous insérons une autre relation, appelée « montant » avec la structure suivante :

id	tr1	tr2	tr3	idpromotion	ANNEE	iddep
9	111.50	78.10	155.10	L2	2018-2019	IG
10	111.50	78.10	155.10	G3	2018-2019	IG
11	92.00	55.80	135.10	G1	2018-2019	ANGLAIS
12	89.20	55.80	135.10	G2	2018-2019	ANGLAIS
13	94.80	61.30	140.10	L1	2018-2019	SCA
19	200.00	150.00	100.00	G3	2019-2020	SCA
20	200.00	150.00	100.00	L2	2019-2020	BIOLOGIE
21	250.00	150.00	150.00	G3	2019-2020	HAT
22	250.00	150.00	150.00	L2	2019-2020	HAT

Exemple d'illustration avec la relation paiement :

- *Requête en algèbre relationnelle :*

$\Pi_{montantp,datep,paiement.annee}(Montant \bowtie_{tr1>Montantp} Paiement)$

- *Requête sql:*

  SELECT montantp, datep, paiement.annee FROM paiement, montant WHERE tr1>montantp

- Résultat:

montantp	datep	annee
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
50.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019
80.00	2019-11-08	2018-2019

c. *Equijointure*

Mêmes relations que les précédentes :

- *Requête en algèbre relationnelle :*

$\Pi_{montantp,datep,paiement.annee}(Montant_{tr1=Montantp} \bowtie Paiement)$

- *Requête sql:*

SELECT montantp, datep, paiement.annee FROM paiement, montant WHERE tr1=montantp

- Résultat:

montantp	datep	annee
111.50	2019-11-13	2018-2019
111.50	2019-11-13	2018-2019

III.3. Opérateurs ensemblistes et autres

Pour aborder cette partie, nous augmentons une nouvelle relation « tentative », ayant comme structure :

idp	Montantp	datep	justification	nature	annee	idetudiant	user	autoriser
1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI	Oui
2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI	Oui
5	100.00	2019-11-13	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI	Non

a. Union

L'union de deux relations R1 et R2 de même schéma produit une relation R3 de même schéma constituée de l'ensemble des tuples appartenant à R1 et/ou à R2.

Considérant la relation paiement et tentative, nous pouvons réaliser ces deux requêtes :

- Requête en algèbre relationnelle :

Paiement ∪ Tentative

- Requête en SQL :

```
select idp,montantp,datep,justification,nature,annee,idetudiant,user from paiement union
select idp,montantp,datep,justification,nature,annee,idetudiant,user from tentative
```

Résultat:

idp	montantp	datep	justification	nature	annee	idetudiant	user
1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI
3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI
5	100.00	2019-11-13	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI

b. Différence

La différence entre deux relations R1 et R2 de même schéma produit une relation R3 de même schéma constituée de l'ensemble des tuples de R1 n'appartenant pas à R2. Notons que la différence entre R1 et R2 n'est pas égale à la différence entre R2 et R1.

- Requête en algèbre relationnelle :

Paiement - Tentative

- Requête en SQL :

```
SELECT paiement.idp, paiement.montantp, paiement.datep, paiement.justification,
paiement.nature, paiement.annee, paiement.idetudiant, paiement.user FROM paiement
LEFT JOIN tentative ON paiement.idp = tentative.idp WHERE tentative.idp IS NULL
```

Résultat :

idp	montantp	datep	justification	nature	annee	idetudiant	user
3	80.00	2019-11-08	Solde tanche1	Paiement	2018-2019	0235/12-13	ABELI
4	111.50	2019-11-13	Solde tanche1	Paiement	2018-2019	1191/12-13	ABELI

c. Intersection

L'intersection de deux relations R1 et R2 de même schéma produit une relation R3 de même schéma constituée de l'ensemble des tuples appartenant à la fois à R1 et à R2. Notons que l'intersection n'est pas une opération de base, car elle est équivalent à deux opérations des différences successives.

- Requête en Algèbre relationnelle :

Paiement ∩ Tentative

- Requête en SQL :

SELECT paiement.idp, paiement.montantp, paiement.datep, paiement.justification, paiement.nature, paiement.annee, paiement.idetudiant, paiement.user FROM paiement INNER JOIN tentative ON paiement.idp = tentative.idp

Résultat :

idp	montantp	datep	justification	nature	annee	idetudiant	user
1	50.00	2019-11-08	Acompte Tranche1	Paiement	2018-2019	0007/12-13	ABELI
2	80.00	2019-11-08	Complément Tranche1	Paiement	2018-2019	0235/12-13	ABELI

d. Le renommage

Utilisons l'exemple de la jointure naturelle

Requête en Algèbre relationnelle

$\pi\rho_{Matricule \leftarrow Etudiants.idetudiant,noms,postnom,montantp}(Etudiants \bowtie_{Etudiants.idetudiant=Paiement.idetudiant} Paiement)$

- Requête en SQL :

SELECT etudiants.idetudiant AS Matricule, noms, postnom, montantp FROM etudiants, paiement WHERE etudiants.idetudiant = paiement.idetudiant

- Résultat

Matricule	noms	postnom	montantp
0007/12-13	MALUBA	Munyololo	50.00
0235/12-13	LUKULA	Basomine	80.00
0235/12-13	LUKULA	Basomine	80.00
1191/12-13	MAKINDU	Ndume	111.50

e. Les agrégats

Les principales fonctions sont les suivantes [14]:

- AVG() pour calculer la moyenne sur un ensemble d'enregistrements
- COUNT() pour compter le nombre d'enregistrements sur une table ou une colonne particulière ;
- MAX() pour récupérer la valeur maximum d'une colonne sur un ensemble de lignes. Cela s'applique à la fois pour des données numériques ou alphanumériques ;
- MIN() pour récupérer la valeur minimum de la même manière que MAX();
- SUM() pour calculer la somme sur un ensemble d'enregistrement.

Utilisons l'exemple de la jointure naturelle

Requête en Algèbre relationnelle :

$$\pi\rho_{Matricule \leftarrow Etudiants.idetudiant,noms,postnom,\ (mon}\gamma_{f)}(Etudiants \bowtie_{Etudiants.idetudiant=Paiement.idetudiant} Paiement)$$

- Requête en SQL :

SELECT etudiants.idetudiant AS Matricule, noms, postnom, sum(montantp),datep FROM etudiants, paiement WHERE etudiants.idetudiant = paiement.idetudiant group by datep

- Résultat :

Matricule	noms	postnom	sum(montantp)	datep
0007/12-13	MALUBA	Munyololo	210.00	2019-11-08
1191/12-13	MAKINDU	Ndume	111.50	2019-11-13

III.4. Illustration de quelques requêtes SQL (de notre projet) en algèbre relationnelle

Compte tenu d'un nombre suffisant des requêtes que nous avons utilisées, nous présentons seulement ceci à titre d'exemple :

- vérification des identifiants saisis par l'utilisateur et ceux du système

- Requête en SQL :

SELECT * FROM users WHERE login='budu' AND password='budu17'

- R.AGR :

$$\sigma_{login=\ 'budu'\ \mathbf{et}\ password='budu17'}(Users)$$

- Affichage de la situation de paiement des étudiants de la G1 IG ,année 2018-2019

- Requête en SQL :

SELECT noms,appartenir.idpromotion,tr1,tr2,tr3,releve.annee,postnom,prenom from releve,etudiants,appartenir,departements,promotions where appartenir.idetudiant=etudiants.idetudiant and etudiants.idetudiant=releve.idetudiant and

promotions.idpromotion=appartenir.idpromotion and departements.iddep=appartenir.iddep and appartenir.annee=releve.annee and appartenir.idpromotion= "G1" and releve.annee="2018-2019" and departements.iddep="IG"

- R.AGR :

$\Pi[_{noms,appartenir.idrpromotion,tr1,tr2,tr3,releve.annee,postnom,prenom}(Etudiants \bowtie_{Etudiants.idetudiant=Appartenir.idetudiant} Appartenir \text{ et } Promotions \bowtie_{promotions.idpromotion=appartenir.idpromotion} Appartenir \text{ et } Etudiants \bowtie_{etudiants.idetudiant=releve.idetudiant} Releve \text{ et } Departements \bowtie_{departemens.iddep=appartenir.iddep} Appartenir)]$et $\sigma[(_{appartenir.idpromotion= "G1"} Appartenir)et (_{releve.annee="2018-2019"} Releve)et (_{departements.iddep="IG"} Departement)]$

- Affichage de la somme journalière perçue à la date du 15/10/2019 en utilisant le renommage de l'id étudiant en matricule)

- Requête en SQL :

SELECT sum(Montantp) FROM paiement where datep="2019/10/15"

- R.AGR :

$\pi [\gamma_{f montantp} Paiement)] \ \sigma[(_{datep= "2019/10/15" Paiement)}]$

- Affichage de la situation des étudiants de G1 IG qui ont déjà payé au minimum 100 $ à la première tranche de l'année académique 2019-2020

- Requête en SQL :

SELECT noms,appartenir.idpromotion,tr1,tr2,tr3,releve.annee,postnom,prenom from releve,etudiants,appartenir,departements,promotions where appartenir.idetudiant=etudiants.idetudiant and etudiants.idetudiant=releve.idetudiant and promotions.idpromotion=appartenir.idpromotion and departements.iddep=appartenir.iddep and appartenir.annee=releve.annee and appartenir.idpromotion=''G1" and releve.annee"2019-2020" and departements.iddep="IG" and tr1 >= 100

- R.AGR :

$\Pi[_{noms,appartenir.idrpromotion,tr1,tr2,tr3,releve.annee,postnom,prenom}(Etudiants \bowtie_{Etudiants.idetudiant=Appartenir.idetudiant} Appartenir \text{ et } Promotions \bowtie_{promotions.idpromotion=appartenir.idpromotion} Appartenir \text{ et } Etudiants \bowtie_{etudiants.idetudiant=releve.idetudiant} Releve \text{ et } Departements \bowtie_{departemens.iddep=appartenir.iddep} Appartenir)]$et $\sigma[(_{appartenir.idpromotion= "G1"} Appartenir)et (_{releve.annee="2018-2019"} Releve)et (_{departements.iddep="IG"} Departement) et (_{Releve.tr1>=100} Releve)]$

- Affichage de la situation de paiement de l'étudiant « AMANI KAZINGUVU Gustave » pour l'année académique 2019-2020

- Requête en SQL :

SELECTnoms,appartenir.idpromotion,montantp,datep,justification,nature,paiement.annee,postnom,prenom,idp from paiement,etudiants,appartenir,promotions,departements where appartenir.idetudiant=etudiants.idetudiant and etudiants.idetudiant=paiement.idetudiant and paiement.annee=appartenir.annee and noms="AMANI " and postnom="KAZINGUFU" and prenom="Gustave" and paiement.annee="2019-2020" and promotions.idpromotion=appartenir.idpromotion and departements.iddep=appartenir.

- R.AGR :

$\Pi[_{noms,appartenir.idrpromotion,montantp,datep,justification,nature,paiement.annee,postnom,prenom,idp}$ (*Etudiants* $\bowtie_{Etudiants.idetudiant=Appartenir.idetudiant}$ *Appartenir et* *Promotions* $\bowtie_{promotions.idpromotion=appartenir.idpromotion}$ *Appartenir et* *Etudiants* $\bowtie_{etudiants.idetudiant=paiement.idetudiant}$ *Paiement et* *Departements* $\bowtie_{departemens.iddep=appartenir.iddep}$ *Appartenir*)]et $\sigma[($*Etudiants*$_{noms= "AMANI"})$et ($_{postnom="KAZINGUFU"}$*Etudiants*)et ($_{prenom="Gustave}$ *Etudiants*) et ($_{paiement.annee= ''2019-2020 ''}$*Paiement*)]

Conclusion partielle

Cette partie de notre travail s'est intéressée sur l'outil mathématique que nous avons utilisé, à la justification de son choix, à la démonstration de ses opérations et à la mise en œuvre de ces dernières dans notre système.

CHAPITRE IV. PROTECTION DES DONNEES CONTRE LES ACCES NON AUTORISES

IV.1. Introduction

Dans ce chapitre, nous allons développer et montrer les mécanismes de sécurité qui nous ont permis de chiffrer les informations stockées dans le système. Il existe deux types notamment : le chiffrement symétrique et asymétrique. A chaque chiffrement y est associé des algorithmes appropriés comme par exemple les algorithmes de Hill, vigenère et César pour le chiffrement symétrique, RSA pour le chiffrement asymétrique. De ce fait, nous avons choisi les algorithmes de César codé sur 256 caractères et RSA codé sur 512 bits. Le chiffrement RSA est utilisé pour chiffrer les mots de passe tandis que le chiffrement César est mis en œuvre pour chiffrer les autres données du système(les noms des étudiants, leur paiement, etc.) . Avec ces deux chiffrements nous sommes arrivé à implémenter un crypto système qui a le rôle de générer les nombres premiers, de chiffrer les données avec RSA et César. Ensuite nous les avons introduits dans notre application de gestion sécurisée des frais académiques.

En implémentant ces mécanismes de de sécurité dans notre système, nous avons mis en évidence les cinq principes génériques de la sécurité informatique qui sont :

- L'intégrité
- L'authentification
- La non répudiation
- La confidentialité
- La disponibilité

IV.2. Le chiffrement des données informatiques

Le chiffrement (ou cryptage) est un procédé de cryptographie grâce auquel on souhaite rendre la compréhension d'un document impossible à toute personne qui n'a pas la clé de (dé)chiffrement. Ce principe est généralement lié au principe d'accès conditionnel [15]. Le principal objectif du chiffrement consiste à garantir la confidentialité des données numériques stockées sur des systèmes informatiques ou transmises via Internet ou d'autres réseaux[16].

IV.2.1. Types de chiffrement

Il existe deux types de chiffrement en informatique, qui sont :

- Chiffrement symétrique
- Chiffrement asymétrique

- ✓ ***Le chiffrement symétrique***

Dans le chiffrement symétrique, une même clé est partagée entre l'émetteur et le récepteur. Cette clé dite symétrique est utilisée par l'émetteur pour chiffrer le message et par le récepteur pour le déchiffrer en utilisant un algorithme de chiffrement symétrique [17].

La figure suivante indique l'échange du message crypté avec le chiffrement symétrique.

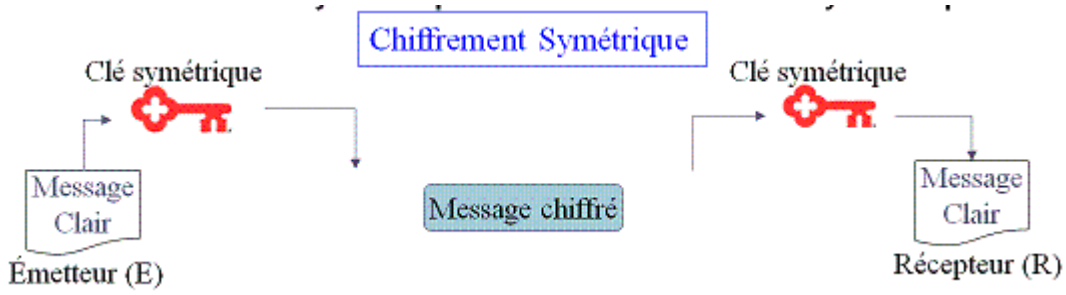


Figure 27. *Echange d'un message avec le chiffrement symétrique*

- ✓ ***Le chiffrement asymétrique***

Dans un système asymétrique, le récepteur génère une paire de clés asymétrique : une clé publique qui est diffusée à tout le monde et une clé privée maintenue secrète chez le récepteur. La particularité de cette paire de clé est que tout message chiffré avec la clé publique ne peut être déchiffré qu'avec la clé privée correspondante [18]. D'où la confidentialité des messages chiffrés avec la clé publique d'un récepteur. Bien évidemment la clé privée correspondante ne peut être calculée à partir de la clé publique correspondante.

La figure suivante indique l'échange du message crypté avec le chiffrement asymétrique.



Figure 28.*Echange d'un message avec le chiffrement asymétrique*

IV.3. Choix des algorithmes de chiffrement

Nous optons pour le chiffrement RSA parce qu'il est parmi le plus puissant algorithme car difficile à casser mais son implémentation exige des moyens matériels importants. Par contre le chiffrement de César est léger et cassable facilement si on utilise l'alphabet français de 26 lettres mais son implémentation ne demande pas d'énormes moyens matériels. Son temps exécution n'est pas constaté par l'utilisateur, c'est-à-dire que ça ne demande pas beaucoup de temps pour être exécuté.

IV.3.1. L'algorithme de CESAR (avec 256 caractères)

Ce système de chiffrement est très simple à mettre en œuvre, cependant étant totalement symétrique, il suffit de faire une soustraction pour connaître le message initial. Une méthode primaire est d'essayer les 26 combinaisons possibles et voir si l'on peut obtenir un message compréhensible (l'inefficacité de cet algorithme). Etant conscient de ce problème, nous avons implémenté certaines informations de notre système avec cet algorithme mais codé sur 256 caractères avec possibilité d'avoir de longues clés. Ce qui va permettre la sécurité de ces dernières chiffrées avec CESAR.

- ✓ ***Fonctionnement de l'algorithme de CESAR***

On remplace chaque lettre ou caractère du texte à chiffrer par la lettre qui est située à n place plus loin dans le code ASCII. Par exemple, si l'on pose n = 50, A deviendra s, B deviendra t etc……

La figure suivante indique le tableau du code ascii avec 256 caractères :

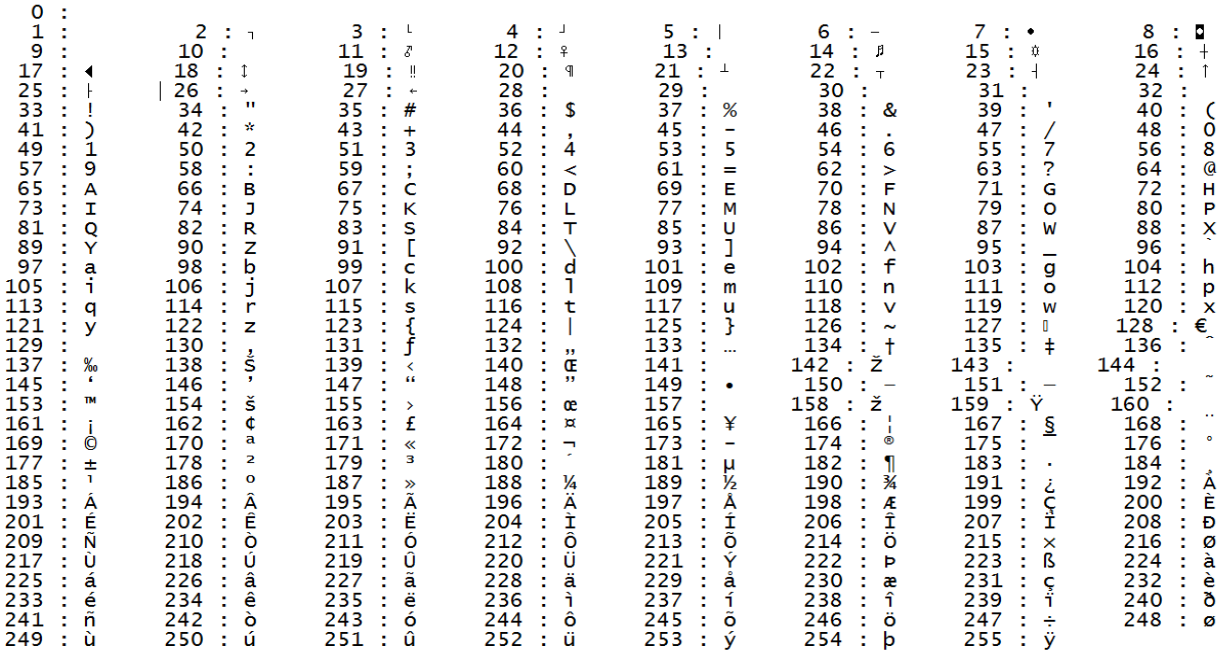


Figure 29.*Table ASCII*

L'algorithme est donc le suivant :

Soit n = la clé, c'est-à-dire le décalage

Soit k = la place de la lettre à crypter dans le code ASCII

Soit c = la place du caractère crypté dans le code ASCII

c = [(n + k) modulo 256]

Expliquons le script : Basiquement on pourrait penser que le chiffre de César équivaut à additionner n et k et ainsi à obtenir la position du nouveau caractère crypté. Seulement il est possible que n + k > 256. Donc, dans ce cas-là, il est nécessaire d'appliquer le modulo, qui est en fait le reste de la division euclidienne.

Donc si n + k < 256, [(n + k) modulo 256] sera égal à n + k.

Et si n + k >= 256, [(n + k) modulo 256], dans tous les cas, [(n + k) modulo 256] sera compris entre 0 et 255.

- ✓ ***Implémentation de l'algorithme de César dans notre projet***
  - ***Introduction***

Pour implémenter cet algorithme, nous nous sommes servi de deux fonctions prédéfinies en php telles que :

- La fonction ord () : Convertit le premier octet d'une chaîne en une valeur entre 0 et 255
- La fonction chr () : Génère une chaîne d'un octet à partir d'un nombre

Après étude et analyse, nous avons remarqué que la fonction **ord ()** ne renvoie pas la vraie valeur ASCII lorsque le résultat est supérieur à 127 et inférieur à 256.

Exemple :

```
<?php
        $a='Ù';
        echo ord($a);
?>
```

Le code Ascii renvoyé après exécution de ce petit code est 195, mais en regardant bien ce tableau ci-dessus on trouve que la valeur correspondante à ce caractère est 217 et le caractère associé au code ascii 195 est Ã. La fonction ord n'intègre pas l'encodage utf_8.

Vu ce problème qui risque de nous donner de mauvais résultats pendant le décryptage des données, nous avons essayé de le contourner en utilisant nos deux fonctions améliorées que voici :

```
<?php
        function ord_utf8($s)
            {
                return(int)($s=unpack('C*',$s[0].$s[1].$s[2].$s[3]))&&$s[1]<(1
                <<7)?$s[1]:($s[1]>239&&$s[2]>127&&$s[3]>127&&$s[4]>12
                7?(7&$s[1])<<18|(63&$s[2])<<12|(63&$s[3])<<6|63&$s[4]:($s
                [1]>223&&$s[2]>127&&$s[3]>127?(15&$s[1])<<12|(63&$s[2
                ])<<6|63&$s[3]:($s[1]>193&&$s[2]>127?(31&$s[1])<<6|63&$
                s[2]:0)));
            }

        function chr_utf8($n,$f='C*')
            {
                return
            $n<(1<<7)?chr($n):($n<1<<11?pack($f,192|$n>>6,1<<7|191&$n):
            ($n<(1<<16)?pack($f,224|$n>>12,1<<7|63&$n>>6,1<<7|63&$n):
            ($n<(1<<20|1<<16)?pack($f,240|$n>>18,1<<7|63&$n>>12,1<<7|63&$
            n>>6,1<<7|63&$n):'')));
            }
?>
```

Ce sont ces deux fonctions qui vont nous permettre de réaliser notre crypto système.

- ***Algorithme pour le chiffrement et déchiffrement du texte avec César***

- ***Le chiffrement***

- Début ;

- Entrées :

Fichier contenant les deux fonctions implémentées ;

-Traitement

Longueur du texte à crypter=fonction de récupération de caractère se trouvant dans un texte(variable à chiffrer)

Pour i=0 et i<=longueur du texte à crypter moins un

Valeur ascii=fonction retournant le code ASCII(fonction retournant la position du mot(variable à chiffrer, i))

Valeur ascii=Valeur ascii+cle

Si valeur ascii >=256 ou valeur ascii < 0 alors

Valeur ascii=Valeur ascii%256

Fin si

Chaine extraite= fonction qui convertit le code ascii en char(Valeur ascii)

Chaine cryptée= Chaine cryptée+ Chaine extraite

Fin pour

- Sortie

Chaine cryptée

- Fin.

La figure ci-dessous indique le bloc-schéma correspondant au chiffrement de César.

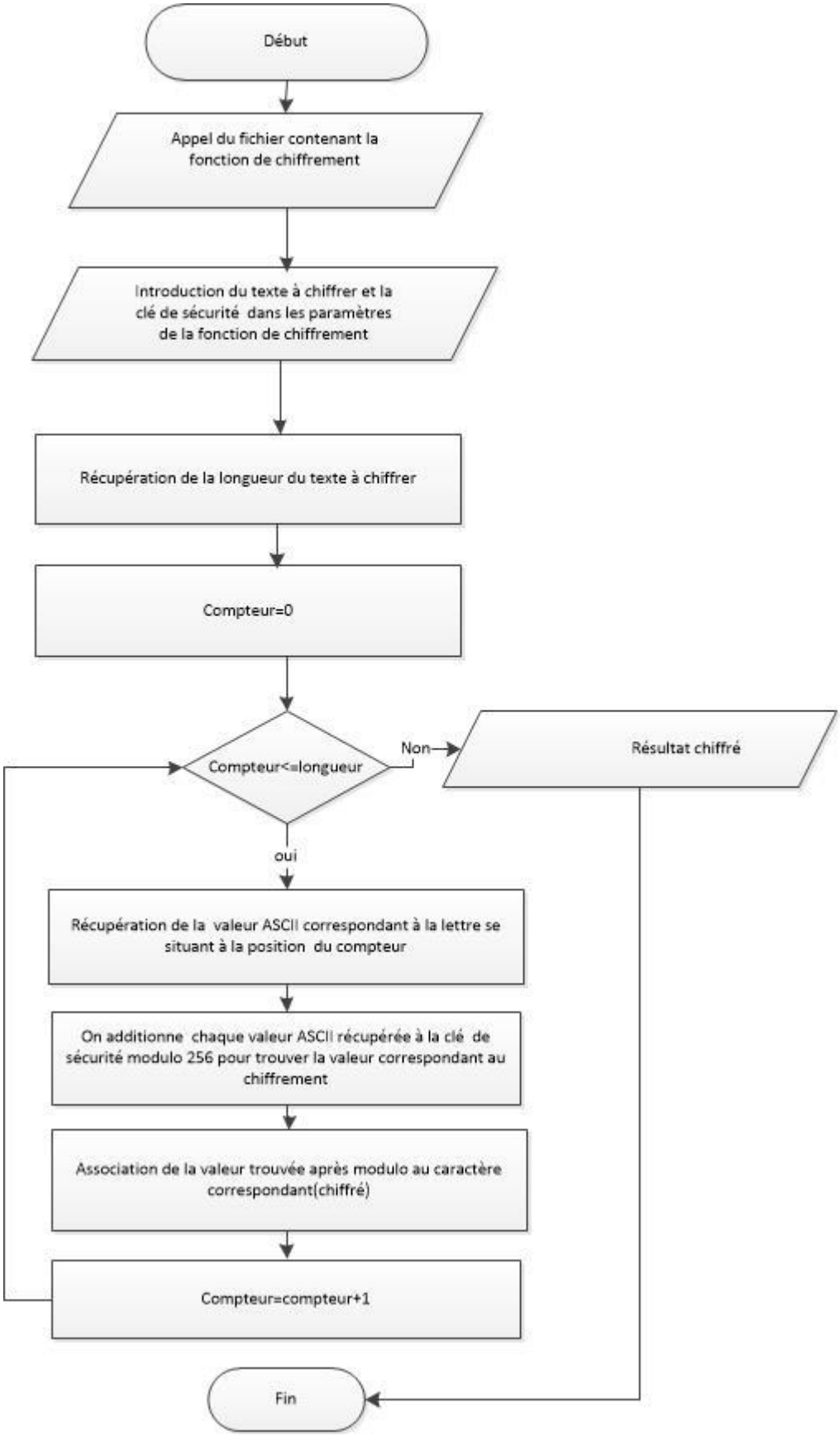


Figure 30.*Chiffrement avec César*

- ***Le déchiffrement***

- Début ;

- Entrées :

Fichier contenant les deux fonctions implémentées ;

-Traitement

Longueur du texte à décrypter=fonction de récupération de caractère se trouvant dans un texte(variable à déchiffrer)

Pour i=0 et i<=longueur du texte à décrypter moins un

Valeur ascii=fonction retournant le code ASCII(fonction retournant la position du mot(variable à déchiffrer, i))

Valeur ascii=Valeur ascii-cle

Si valeur ascii >=256 ou valeur ascii < 0 alors

Valeur ascii=Valeur ascii%256

Fin si

Chaine extraite= fonction qui convertit le code ascii en char(Valeur ascii)

Chaine décryptée= Chaine décryptée+ Chaine extraite

Fin pour

- Sortie

Chaine décryptée

- Fin.

La figure ci-dessous indique le bloc-schéma correspondant au déchiffrement de César.

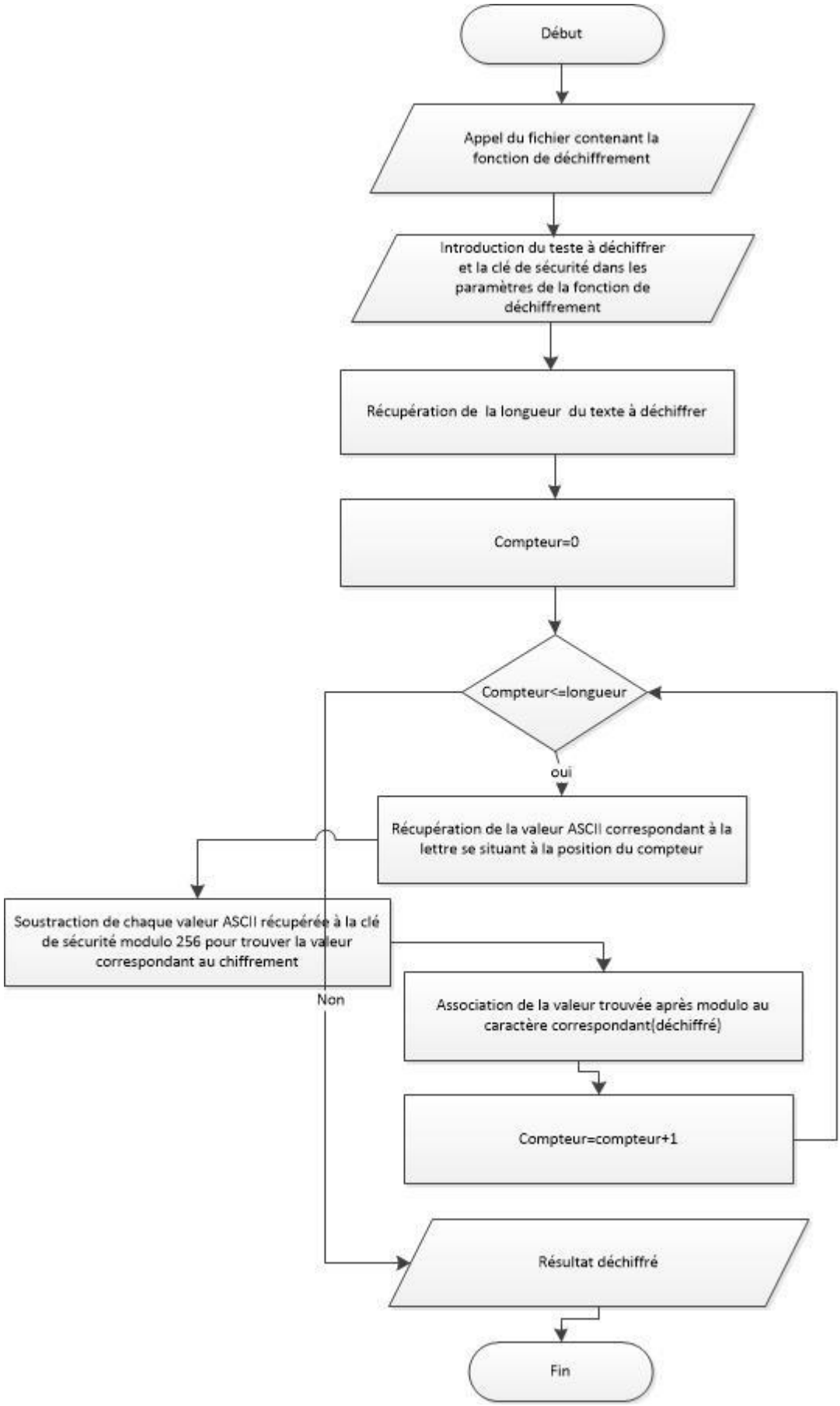


Figure 31.*Déchiffrement avec César*

Ces blocs-schémas traduisent fidèlement ce que nos algorithmes ci-dessus viennent de démontrer. Outre ces deux algorithmes qui sont implémentés dans notre projet de recherche, nous avons essayé de mettre à la disponibilité de tout le monde un outil (c'est un outil supplémentaire) permettant de faire le chiffrement et déchiffrement des fichiers textes avec CESAR et RSA, dont les images suivantes montrent le fonctionnement de cet outil (crypto système CESAR) :

La figure ci-dessous la manière dont il faut crypter un fichier texte avec le chiffrement de césar à partir de cet interface.

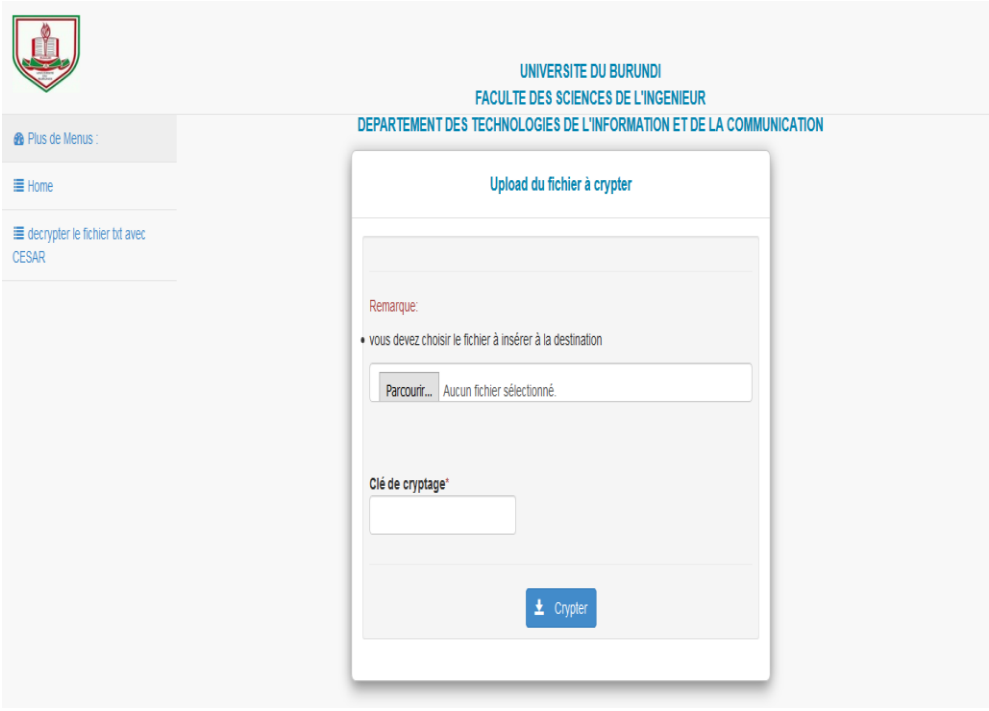


Figure 32. *Interface de chiffrement César*

Après avoir démarré tous les services, l'interface qui va nous servir à faire le chiffrement de nos données avec césar apparaît . Deux conditions sont exigées pour réaliser le chiffrement : la première est d'avoir sur un disque (interne ou externe) le fichier à chiffrer (du type .txt) et la deuxième est de posséder la clé de chiffrement. Une fois ces deux conditions réunies, on peut uploader (de son emplacement) ce fichier et insérer la clé de sécurité. Enfin, on clique sur le bouton crypter pour générer un autre fichier du type .txt mais chiffré. Le résultat chiffré se stocke sur le serveur dans un fichier nommé :' ***fichiercrypt.txt***' qu'on peut récupérer et utiliser pour des fins utiles ; on s'en sert également pour le déchiffrement. Et le fichier déchiffré est nommé :***' defichiercrypt.txt'*** contenant le texte en clair.

A titre d'exemple, voici du texte à chiffrer en ayant 2500 comme clé de sécurité :

Après avoir démarré tous les services, interface qui va nous servir à faire le chiffrement de nos données avec césar apparait :

deux conditions sont exigées pour réaliser le chiffrement, la première :il faut avoir sur un disque (interne ou externe) le fichier à chiffrer (du type .txt)tandis que la deuxième est de posséder la clé de chiffrement.

Une fois ces deux conditions réunies on peut uploader (de son emplacement) ce fichier et insérer la clé de sécurité, afin on clique sur le bouton crypter pour générer un autre fichier du type .txt mais chiffré.

Et le résultat de ce chiffrement donne :

46¬7ä%:3-6ä(-1%66-ä8397ä0)7ä7)6:-')7ðä-28)6*%')ä59-
ä:%ä2397ä7)6:-6ä¤ää*%-6)ä0)ä',-
*****6)1)28ä()ä237ä(322-)7ä%:)'ä'-7%6ä%44%6%-8äþÑÎ()9<ä'32(-8-***
327ä7328ä)<-+-)7ä4396ä6-%0-7)6ä0)ä',-**6)1)28ðä0%ä46)1-¬6)äþ-
0ä*%98ä%:3-6ä796ä92ä(-759)äì-28)62)ä39ä)<8)62)íä0)ä*-',-)6ä¤ä',-
*****6)6äì(9ä8=4)äò8<8í8%2(-7ä59)ä0%ä()9<-***
¬1)ä)78ä()ä4377-()6ä0%ä'0-ä()ä',-**6)1)28òäÑÎä2)ä*3-
7ä')7ä()9<ä'32(-8-327ä6-92-
)7ä32ä4)98ä9403%()6äì()ä732ä)140%')1)28íä')ä*-',-)6ä)8ä-
27-6)6ä0%ä'0-ä()ä7-'96-8-ðä%*-2ä32ä'0-
59)ä796ä0)ä&39832ä'6=48)6ä4396ä+-2-6)6ä92ä%986)ä*-',-
)6ä(9ä8=4)äò8<8ä1%-7ä',-**6-ò

Signalons que le résultat de déchiffrement marche aussi impeccablement, le texte initial nous retourné :

Après avoir démarré tous les services, interface qui va nous servir à faire le chiffrement de nos données avec césar apparait :

deux conditions sont exigées pour réaliser le chiffrement, la première :il faut avoir sur un disque (interne ou externe) le fichier à chiffrer (du type .txt)tandis que la deuxième est de posséder la clé de chiffrement.

Une fois ces deux conditions réunies on peut uploader (de son emplacement) ce fichier et insérer la clé de sécurité, afin on clique sur

le bouton crypter pour générer un autre fichier du type .txt mais chiffré.

IV.3.2. Algorithme RSA codé sur 512 bits

Le crypto système RSA (du nom de ses inventeurs Rivest, Shamir et Adelman) est une méthode de codage complexe mais très efficace, encore très largement utilisée aujourd'hui, notamment dans la protection des dossiers hautement confidentiels.

- ✓ ***Fonctionnement de l'algorithme RSA***

Le RSA repose sur un codage quelque peu complexe. En effet, une personne choisit deux nombres premiers que nous appellerons communément *p* et *q*. Le produit de ces deux nombres est noté *n*. La personne procède ensuite à un nouveau produit, cette fois de $r = (p - 1)*(q - 1)$ et choisit un nombre premier noté *e*, également premier avec le produit *r*. La clé (RSA *n* ; *e*) appelée clé publique est ensuite publiée librement sur Internet ou autre annuaire. La clé privée (qui ne sera pas révélée et servira au décodage) résulte de $d = (r + 1)/(e)$ et servira au décodage du message [19]

. Etant donné que le crypto système RSA utilise les nombres premiers, nous avons essayé de mettre en place un outil capable de générer les nombres premiers codés sur 512 bits. (C'est le deuxième outil supplémentaire que nous avons développé)

- ✓ ***Générations des nombres premiers***

Les nombres premiers sont une perpétuelle source de recherche pour les savants. Bien que définis de manière simple, ils sont l'objet d'analyse théorique forte, complexe et depuis la mise en place d'algorithmes nécessitant des nombres premiers en cryptographie, ils représentent un réel enjeu actuel. Et c'est à tel point que de mystérieux mécènes sont prêts à débourser des sommes considérables pour stimuler la recherche dans ce domaine et obtenir des résultats exploitables [20]. Plus nous avons des nombres premiers codés sur plus des bits, plus la sécurité de nos données devient forte. C'est ainsi que nous avons pensé au développement de cet outil qui est capable de générer les nombres premiers codés sur 512 bits, soit 155 chiffres voire plus. En image l'outil de génération de ces nombres:

Cette figure montre comment on génère les nombres premiers codés sur 512 bits

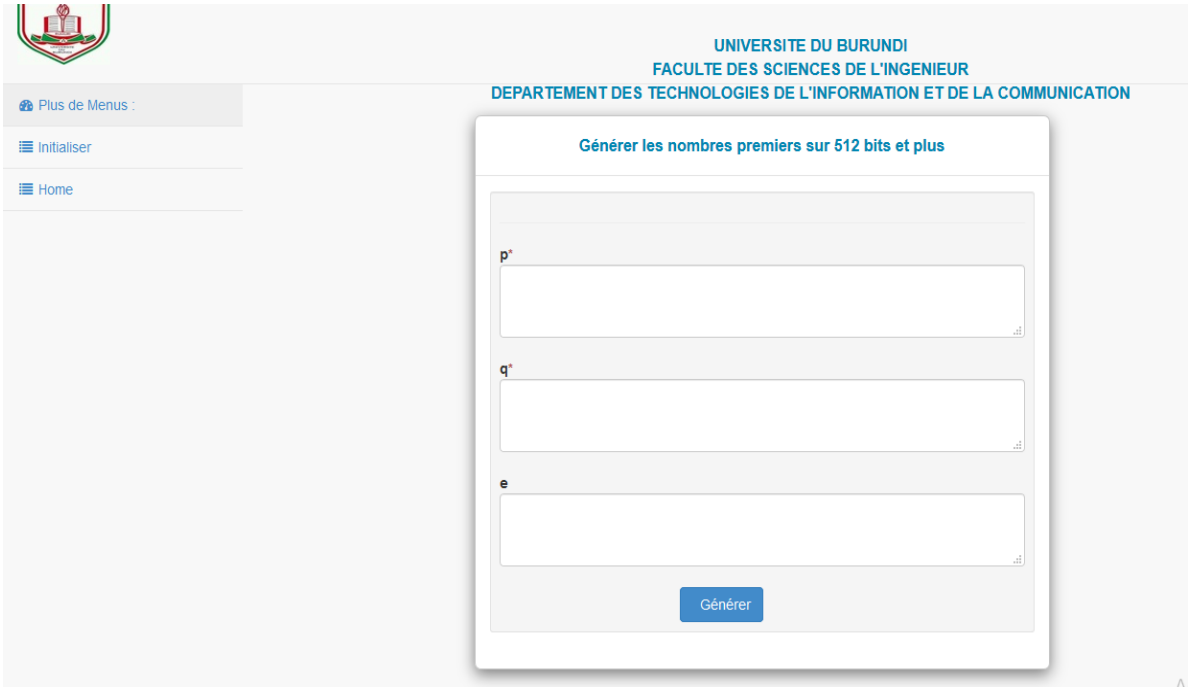


Figure 33.*Interface de génération des nombres premiers*

Cet outil génère automatiquement trois nombres premiers codés sur 512 bits (soit 155 chiffres). Voici quelques nombres premiers générés par cet outil :

1)
596721879206948255956517722222391612576547124577030263036256374835636003050630233930207666727646422524763576327763248280816952531492576451641112311507 34319

2)
504904769736044483586598200666953948480234706501778356718783586283103783804 174865153522250204767692397231174684135751921259721876247952983649439620954 14433

3)
293185244785254653722377967404437448445646281140220403077880230652046092476 419692480032507605680677444020818528973111730357380719808966320163279570559 06471

4)
453001569670613897625228834430419398513348646620426550949795142628851584602 164565504664231289071083584747585979070756687532295735594470169063485726951 58911

5)
70153974760750044699803906664014599788455681139733460306854186438895919180160591645864914217791756596308958326807720951606562778902708781834218869037788511

6)
70272200276495188049507955859640429257538566864221506188800084998920117616124572049261856025944986483271203597584971047909063153795869889688420790016965869

7)
20197218138942838158255145104627317416974982388748773230455891290544826823598244490616336939561440144848439254559230776241179392553741094384520355765367481

8)
46323769925493614072308143445109220780294469567969361847233425789727521073467983584807438986351975000873282338719373901359517378903815484342687340630266169

9)
75670512382747609471015931553577997092420049965450112804017060630110571775784342961084372772761738106672182021810448970043988563132111579935969948608524273

10)
31634092347761892048887779802099494879746618247792861815933495689287784731247312078774147555879735695089231607882232606154513196928530303405879941892948879

Bien que le travail de retrouver ou de générer ces nombres est complexe, nous y sommes arrivé, et notre outil est capable d'en générer autant que nous voulons et ce qui prouve le dynamisme de notre système du fait qu'on peut changer des clés quand on le voudra.

Exemple de chiffrement d'un message 1500 avec comme valeurs :

P=
75670512382747609471015931553577997092420049965450112804017060630110571775784342961084372772761738106672182021810448970043988563132111579935969948608524273

q =
3163409234776189204888777980209949487974661824779286181593349568928778473124731207877414755587973569508923160788222326061545131969285303034058799418929487 9

e =
4632376992549361407230814344510922078029446956796936184723342578972752107346798358480743898635197500087328233871937390135951737890381548434268734063026616 9

La figure suivante montre comment on crypte des données avec le chiffrement RSA.

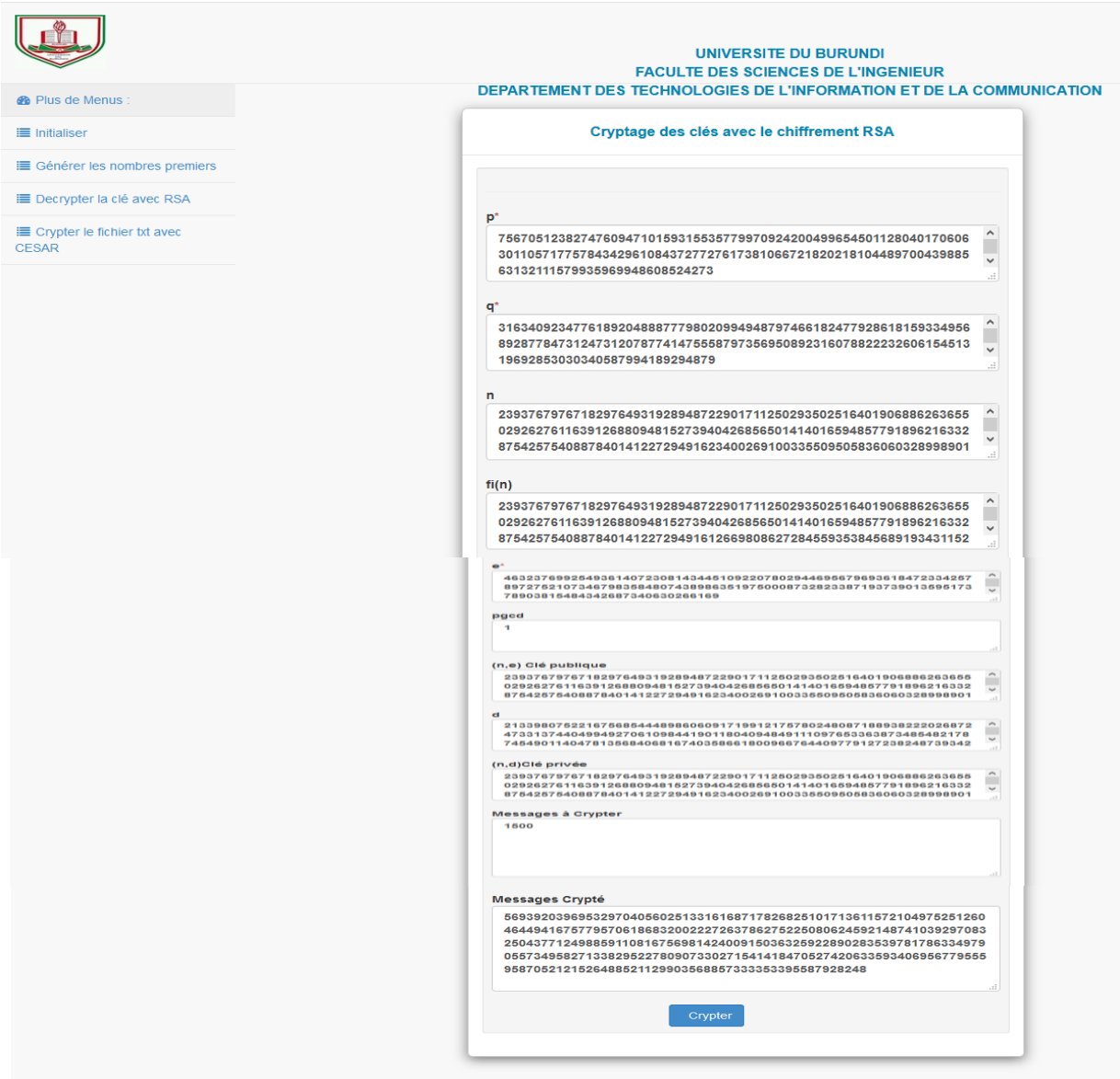


Figure 34.*Interface de chiffrement RSA*

n =

23937679767182976493192894872290171125029350251640190688626365502926276116391268809481527394042685650141401659485779189621633287542575408878401412272949162340026910033550950583606032899890157647046376292114791416513906188685137714658201851237838484843471729111703529836521230737659891559358226108651826260979 67

$\varphi(n)$ =

23937679767182976493192894872290171125029350251640190688626365502926276116391268809481527394042685650141401659485779189621633287542575408878401412272949162669808627284559353845689193431152379253796941596850452170083429947015726443416514526526351984287337114975672329097989241432610633097121233430723982827881 6

d =

21339807522167568544489860609171991217578024808718893822026872473313744049949270610984419011804094849111097653363873485482178745490114047813568406816740358661800966764409779127238248739342604465827908634047006643013104260082927515595852585023220122914141313086482929329865470390952366670436637299212941176164 1

Message crypté :

56939203969532970405602513316168717826825101713611572104975251260464494167577957061868320022272637862752250806245921487410392970832504377124988591108167569814240091503632592289028353978178633497905573495827133829522780907330271541418470527420633593406956779555958705212152648852112990356885733335339558792824 8

Le déchiffrement se fera également à l'aide de cet outil, avec comme condition la détention de la clé privée (n, d) et le message à déchiffrer. La figure suivante montre comment se fait déchiffrement avec RSA.

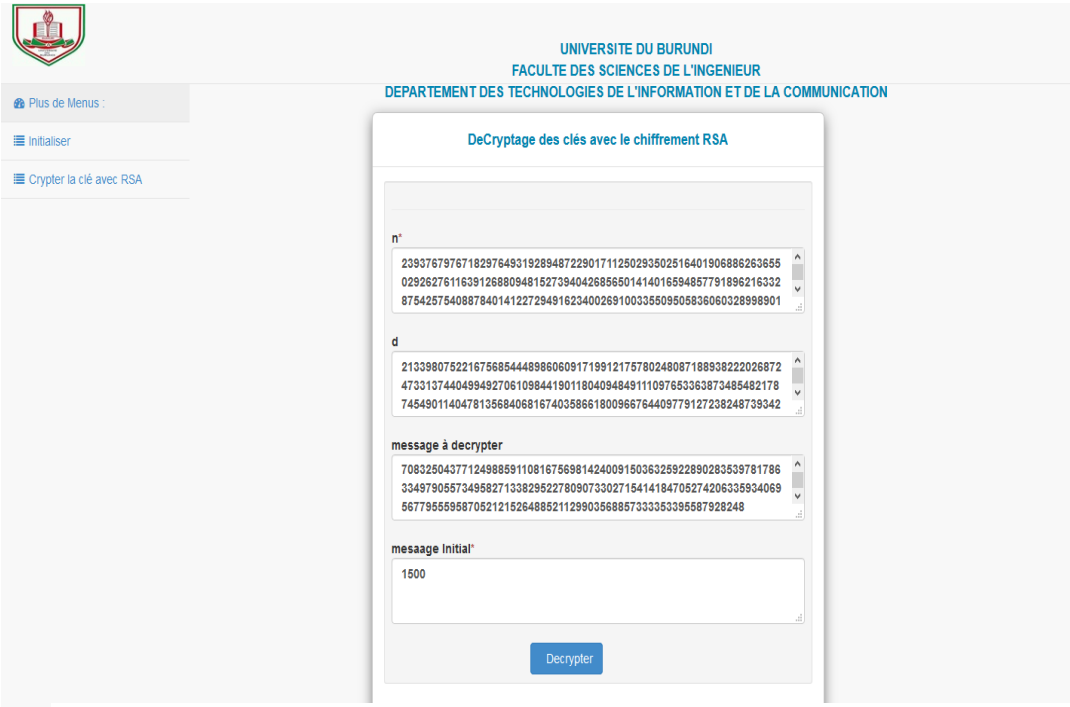


Figure 35.*Interface de déchiffrement RSA*

- ✓ ***Algorithme pour le chiffrement et déchiffrement du texte avec RSA***
  - ***Le chiffrement***

Avec cet algorithme, nous essayerons de chiffrer tous les identifiants qui seront stocqués dans notre système. Il repose sur la théorie des nombres premiers (codés sur 512 bits).

- Début ;

- Entrées :

La bibliothèque MathBiginteger se trouvant dans notre application

Fichier contenant les deux fonctions implémentées ;

-Traitement

Longueur du texte à crypter=fonction de récupération de caractère se trouvant dans un texte(variable à chiffrer)

Pour i=0 et i<=longueur du texte à crypter moins un

Valeur ascii=fonction retournant le code ASCII(fonction retournant la position du mot(variable à chiffrer, i))

Si Valeur ascii = 1 chiffre alors

Valeur ascii=00+Valeur ascii

Si non

Si Valeur ascii = 2 chiffre alors

Valeur ascii=0+Valeur ascii

Fin si

Fin si

Valeur Ascii reçue= Valeur Ascii reçue +Valeur ascii

Résultat chiffré= Valeur Ascii reçue exposant **e** modulo **n**

Fin pour

- Sortie

Résultat chiffré

- Fin.

La figure ci-dessous indique le bloc-schéma correspondant au chiffrement RSA.

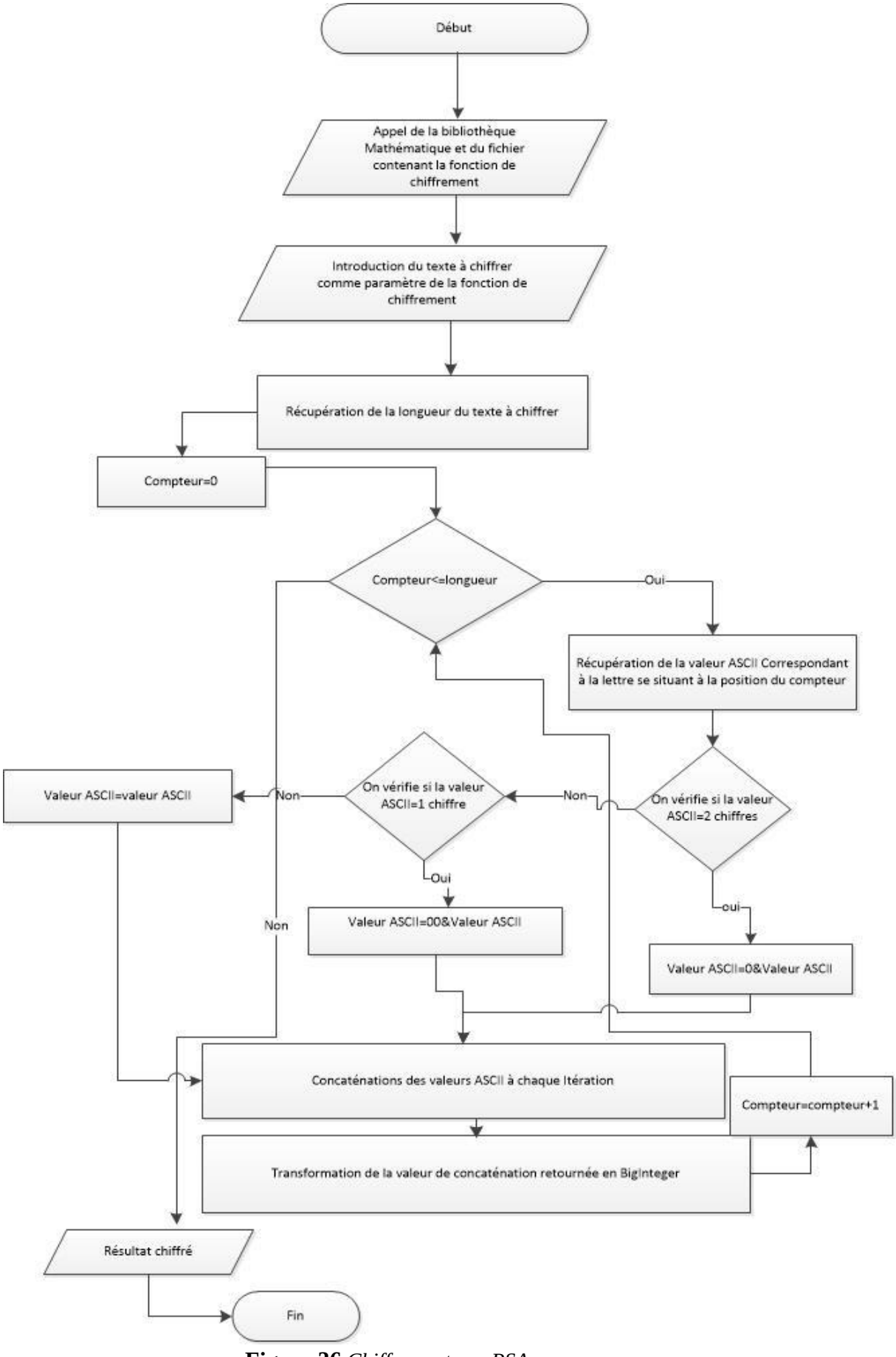


Figure 36.*Chiffrement ave RSA*

- ***Le déchiffrement***

Avec celui-ci nous allons mettre en place le mécanisme de déchiffrement des informations traitées par l'algorithme précédent. Il utilise les mêmes techniques que la précédente.

- Début ;

- Entrées :

La bibliothèque MathBiginteger se trouvant dans notre application

Fichier contenant les deux fonctions implémentées ;

-**Traitement**

Texte à décrypter= texte chiffré exposant **d** modulo **n**

Longueur du texte à décrypter=fonction de récupération de caractère se trouvant dans un texte(Texte à décrypter)

Pour i=0 et i<=longueur du texte à décrypter /3

Valeur ascii=fonction retournant le code ASCII(fonction retournant la position du mot(variable à déchiffrer, i))

chaine reçue= fonction transformant le code ASCII en chaine(Valeur Ascii)

Résultat déchiffré= Résultat déchiffré+ chaine reçue

Fin pour

- Sortie

Résultat déchiffré

- Fin.

La figure ci-dessous indique le bloc-schéma correspondant au déchiffrement RSA.

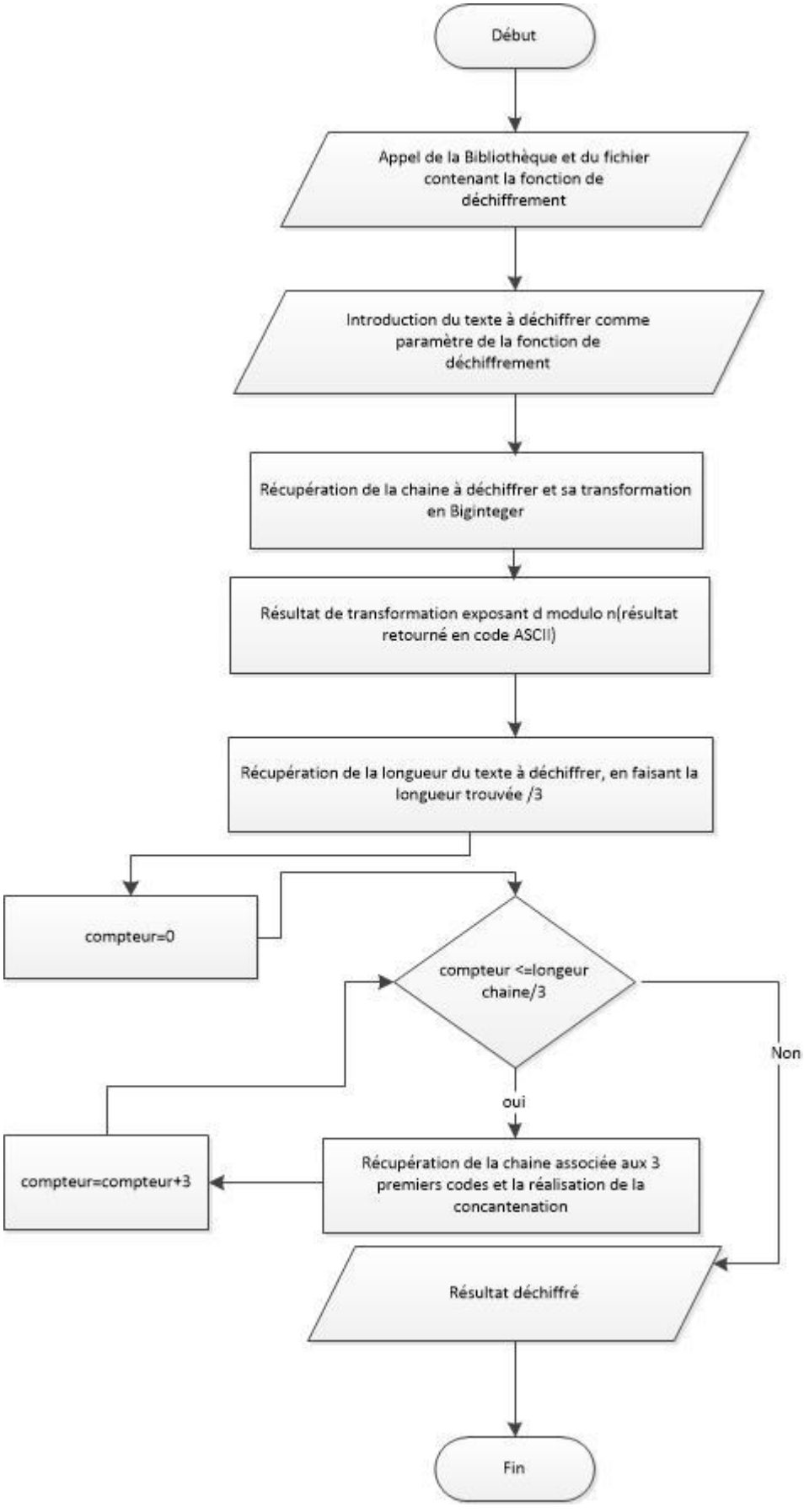


Figure 37.*Déchiffrement avec RSA*

Conclusion partielle

Dans ce chapitre nous avons développé les mécanismes de sécurité de notre système. Nous avons développé les crypto systèmes asymétrique et symétrique. Pour le crypto système asymétrique, nous avons développé et implémenté un algorithme de chiffrement et de déchiffrement RSA, celui-ci est utilisé pour chiffrer les mots de passe par contre pour le crypto système symétrique, nous avons fait recours au chiffrement César pour crypter les autres types des données.

CHAPITRE V. REALISATION D'UNE APPLICATION SECURISEE DE GESTION DES FRAIS ACADEMIQUES

V.1. Outils de Développement

1. WampServer

WampServer est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi qu'une administration PhpMyAdmin.

Il dispose d'une interface d'administration permettant de gérer et d'administrer ses serveurs au travers d'un tray-icon (icône près de l'horloge de Windows).

La figure suivante montre la page d'accueil de wampserver.



Figure 38.*Interface d'accueil WampServer*

2. PHP

PHP est un langage de scripts libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale, en exécutant les programmes en ligne de commande. PHP est un langage impératif disposant depuis la version 5 des fonctionnalités de modèle objet complètes. En raison de la richesse de sa bibliothèque, on désigne parfois PHP comme une plate-forme plus qu'un simple langage.

3. MYSQL

MySQL est un système de gestion de bases des données (SGBD). Selon le type d'application, sa licence est libre ou propriétaire. Il fait partie des logiciels de gestion des base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels.

4. Bootstrap

Bootstrap est une compilation de plusieurs éléments et fonctions web-design personnalisables, le tout emballé dans un seul et même outil. Les développeurs qui utilisent Bootstrap pour la création de leur site web choisissent les éléments qu'ils veulent utiliser. Les éléments personnalisables compilés dans Bootstrap sont une combinaison de HTML, CSS et JavaScript. Et grâce à la magie de l'open-source, Bootstrap s'améliore en permanence : de nouvelles fonctions absolument géniales ont été ajoutées comme le 100% mobile responsive ou la très large sélection de plugins jQuery.

5. SQL DESIGNER

Cet outil nous a permis de générer le code SQL automatiquement. Nous dessinions seulement les tables, et le code SQL se générait automatiquement.

6. MathBigInteger

C'est une bibliothèque mathématique que nous avons utilisée pendant le chiffrement, en utilisant les nombres premiers codés sur 512 bits.

7. PHP Excel

Cette bibliothèque nous a été utile pendant l'upload et la génération automatique des fichiers Excel.

8. FPDF

Celle-ci nous a permis de générer les rapports en format PDF.

9.Mirosoft visio

Cet outil nous a été utile pour la présentation des blocs schèmas.

10. Microsoft visual paragdim

Celui-ci nous a permis de dessiner certains diagrammes UML

11. Power AMC

Comme le précédent, il nous a été utile pour les diagrammes UML

V.2. Présentation de l'application

Avant de passer à la présentation proprement dite de notre application, nous avons le devoir de signaler que nous avons développé un autre outil qui nous permettra ainsi qu'aux futurs chercheurs de générer les nombres premiers aléatoires codés sur 512 bits et de chiffrer/déchiffrer les informations avec ces derniers.

Fonctionnement du système

Nous rappelons avant toute chose que nos données sont stockées dans une base de données mysql et toutes sont chiffrées/cryptées, à titre d'exemple voici certaines tables de notre système ayant les données chiffrées :

a. Table Etudiants (chiffrée avec César)

idetudiant	noms	postnom	prenom	sexe
6803/18-19	/$16	"=>EG=	)=PDEH@A	"
6140/18-19	5#%.3!	=>QJC=	.KI=EJ	)
5974/18-19	+*!	)QCAG=	J@NA	)
6595/18-19	1(*#(%.!	*@AGK	)ENE=I	"
5864/18-19	5!*	=ODKIAG=	+I>AJE	"
6426/18-19	$1'	=D=C=NDA		)
6155/18-19	%'3*%*!	=>SEJA		"
6350/18-19	1'	ENEJC=JEJA	,=NB=EP	)
6831/18-19	%!1)!.%	"=FEJC=	)KEOA	)
6466/18-19	"1.$%/$	)QH=J@Q	EIAA	"
5901/18-19	$!.%0%!.	!VA?GE=	&A=J	)
5957/18-19	%)*%	)>KJAGQ>A	DNEOLEJ	)
5775/18-19	%/*	3KJS=	,DEHELLA	)
5701/18-19	&-1!	)=OANQG=	0DKI=O	)
6060/18-19	"*/%(!)+	)=OQI>QGK	NEOPKPA	)
6834/18-19	'%0+'+	/=HQIQ	#N=PEAJ	)
5739/18-19	'513	)QPKJK		)
6362/18-19	)01)1%.$%	)QOD=C=HQO=	&KOQA	)
6055/18-19	)%(%6%	)QHQJCQH=		"
6965/18-19	)1 !'!.!6	)QJUANAJG=J=	.KOEJA	"
5807/18-19	)1#%/$+	=PQIQGA	&QHEAJ	)
6189/18-19	)1/$#(1/	QDAJ@S=		)
5994/18-19	)1/%/%	"=OD=>=C=JUK	&A=J =LPEOPA	)
6315/18-19	)3!*!*01	QH=I>K	!NE?G	)
6654/18-19	*!!)	)QHQJ@Q	!OPDAN	"
6168/18-19	*#+)+.	"=RQV=	5RAPPA	"
6438/18-19	*/$++(!	)=NKUE	*=PD=HEA	"
6565/18-19	*/$+'*+	=OEJUEVA	%JJK?AJP	)
6203/18-19	,.+#.))!	U=IKJE	HKREO	)
5867/18-19	.1(%)	@NEAJ		)

Figure 39.*Modèle d'une table chiffrée en césar*

b. Table users (chiffrée avec RSA)

iduser	nom	fonction	login	password
31	3709944898037054663273975568984394054290111890366...	3951367291296183500159475746064664744996319064763 2...	3709944898037054663273975568984394054290111890366...	3709944898037054663273975568984394054290111890366...
32	1314403302216178646866810819530868897912269177566550...	1503786478238125545986450695919489389432582783715 6...	1314403302216178646866810819530868897912269177566550...	1314403302216178646866810819530868897912269177566550...
35	2111824180864460484240089921897833848734423482941 9...	3328029939039094547754827958431304107658131836664 2...	2888178362229971960360527096494493555160729242654...	2238973160622148073773312204282634690395542606209 9...
39	2149666135035031520847762949668790240223973675025 8...	3328029939039094547754827958431304107658131836664 2...	2149666135035031520847762949668790240223973675025 8...	2149666135035031520847762949668790240223973675025 8...
40	9501761272646793498763726016929344113651933354147 1...	3328029939039094547754827958431304107658131836664 2...	6879550896067502463926928399389990544156044265426...	4337746931387242007558078897748093589709457783156 9...
41	2807054555520729135114648119869708945474947968959 4...	3328029939039094547754827958431304107658131836664 2...	3601869395994871288487984216792683664061505747869 7...	3601869395994871288487984216792683564061505747869 7...
42	1297448050937406912325786776083749992037823526544 1...	3328029939039094547754827958431304107658131836664 2...	1297448050937406912325786776083749992037823526544 1...	2327194801126859857837986428845339502307897015254 2...
43	2508872752281568670375443791133241668099977195496 8...	1486503915338649869047128607729388603038427285954169...	3070096734773605686940586301400235806972430988630 0...	2508872752281568570375443791133241668099977195496 8...
44	4810268108526019356224834037711470747226053636005 9...	1486503915338649869047128607729388603038427285954169...	4810268108526019356224834037711470747226053636005 9...	4810268108526019356224834037711470747226053636005 9...
45	3868763704007893281762915042888877409928320384497 1...	3868763704007893281762915042888877409928320384497 1...	3550959078414840644992873391178081671258987668902 0...	3550959078414840644992873391178081671258987668902 0...
46	9802929318287783666950221046478625187416025215500 5...	3011612413113757593611384442023696614008866503316 9...	9802929318287783666950221046478625187416025215500 5...	9802929318287783666950221046478625187416025215500 5...
47	2738817664914797331798029981067931721454038096498 0...	3883848343222943936816748015262098896194240868161 1...	7792208754281857619100938210377432183285079501998 8...	7792208754281857619100938210377432183285079501998 8...

Figure 40. *Modèle d'une table chiffrée en RSA*

Eu égard ce qui précède, Il est à noter qu'à chaque requête de l'utilisateur, le système fait automatiquement le déchiffrement pendant l'affichage.

Après ce rappel, voici alors les étapes à suivre pour se connecter à notre système :

L'utilisateur doit lancer l'adresse de notre application se trouvant sur le serveur dans la barre d'adresse de navigateur, et la page d'accueil s'affiche de la manière suivante **[Figure40]:**

La figure suivante montre la page d'accueil de notre système.

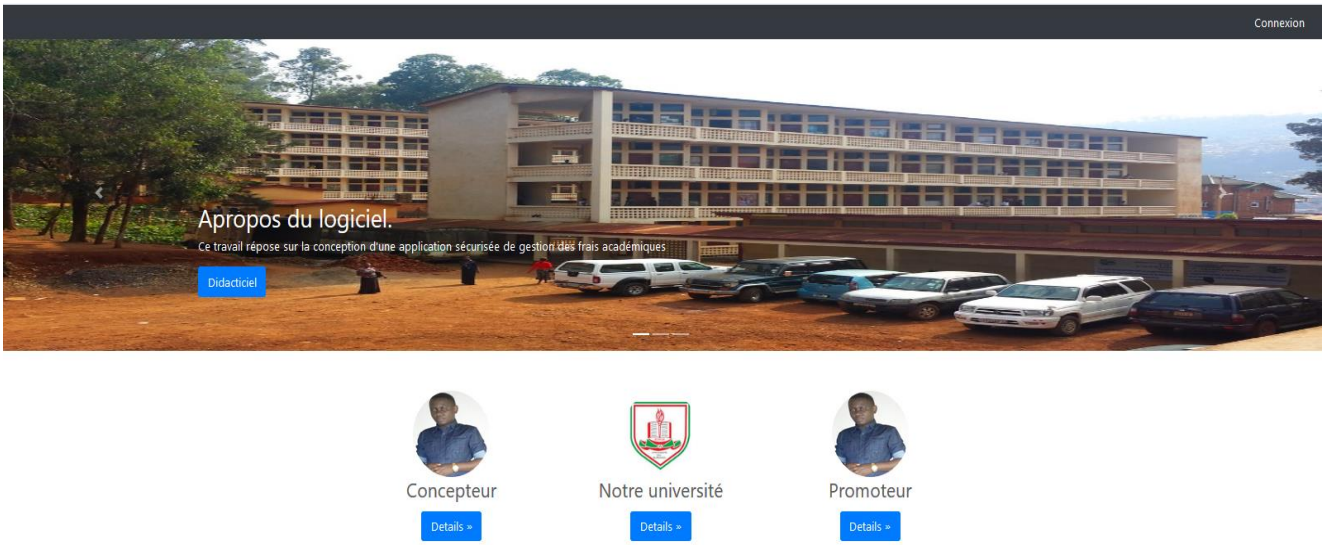


Figure 41.*Page d'accueil de notre application*

En étant à la page d'accueil, l'utilisateur doit cliquer sur connexion pour lancer le système.

La figure suivante montre l'interface de connexion

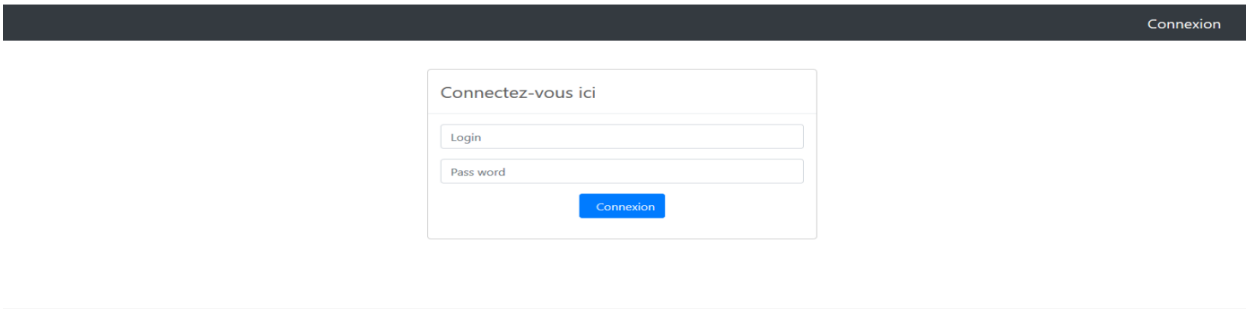


Figure 42.*Interface de connexion*

Si les cordonnées sont correctes et la fonction est « Administrateur de Budget », le système le dirige vers la page d'administrateur de budget. La figure suivante montre l'espace de l'administrateur de Budget.

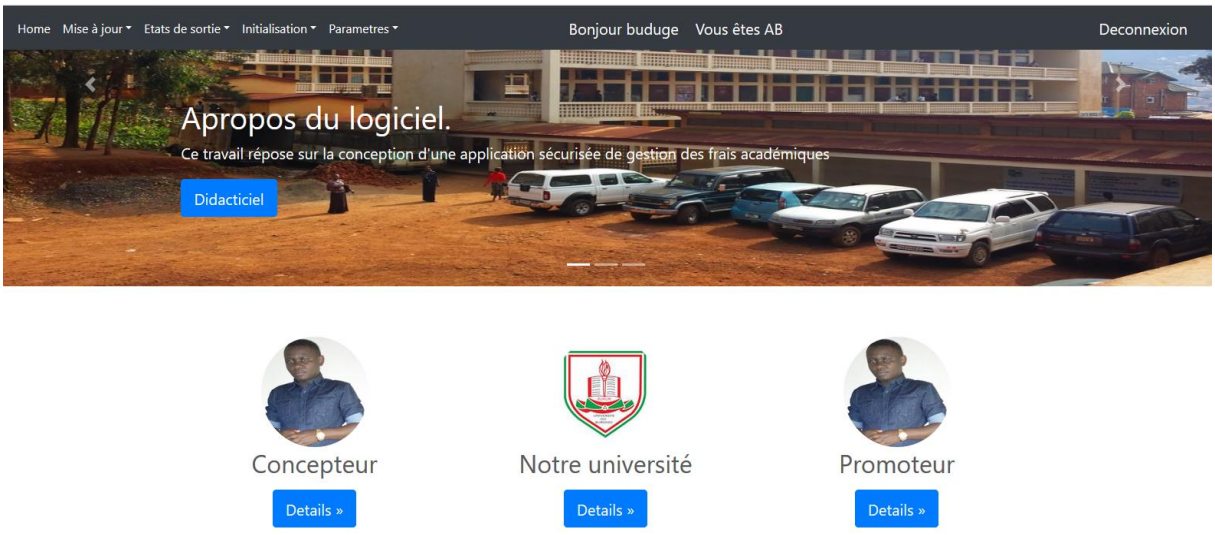


Figure 43.*Espace AB*

Comme expliqué dans le chapitre précédent, l'administrateur de Budget est l'acteur principal du système. Ses attributions sont aussi détaillées dans le diagramme de cas d'utilisation, toutefois en voici quelques-unes en mode graphique, il peut :

- Créer des nouveaux utilisateurs

La figure suivante montre l'interface de la création des nouveaux utilisateurs.

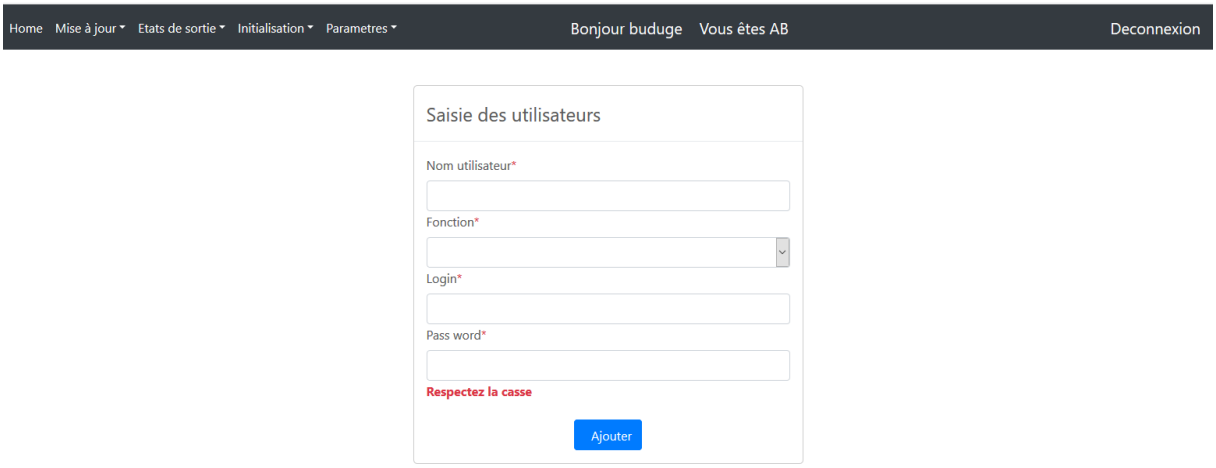


Figure 44.*Interface de création des nouveaux utilisateurs*

- Bloquer les autres utilisateurs

La figure ci-dessous montre l'interface pour bloquer les autres utilisateurs.

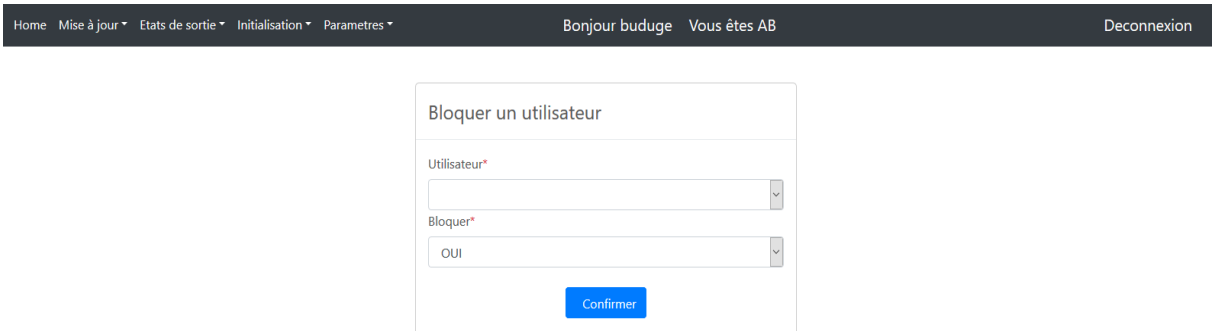


Figure 45.*Interface de privilège de connexion*

- Autoriser les accès

La figure suivante montre l'interface d'autorisation des accès

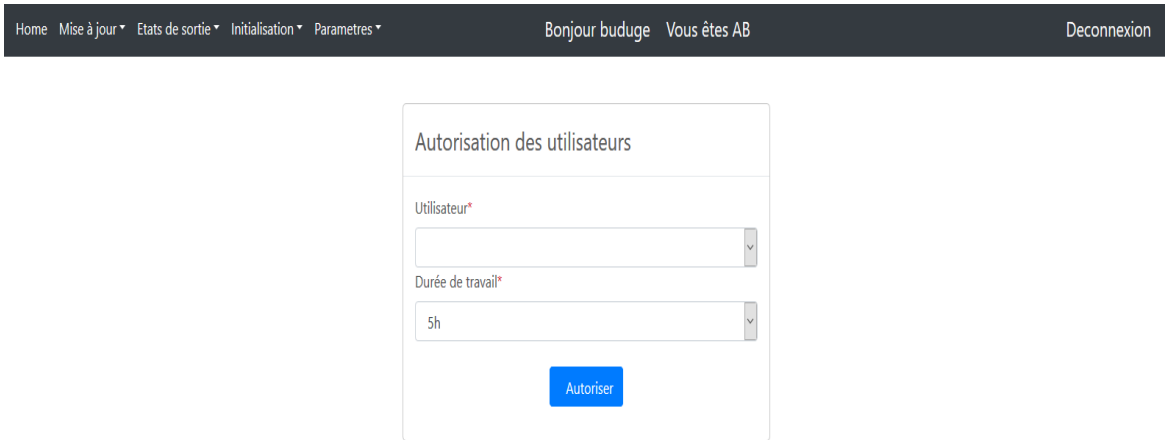


Figure 46.*Interface d'autorisation*

- Supprimer une opération

La figure suivante montre l'interface de suppression d'opérations.

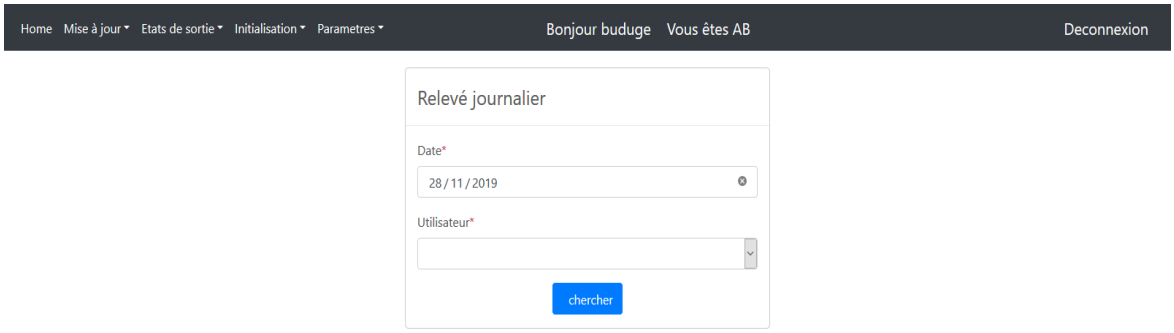


Figure 47.*Interface de suppression*

- Initialiser la somme à payer chaque année

La figure suivante montre l'interface d'initialisation de la somme à payer

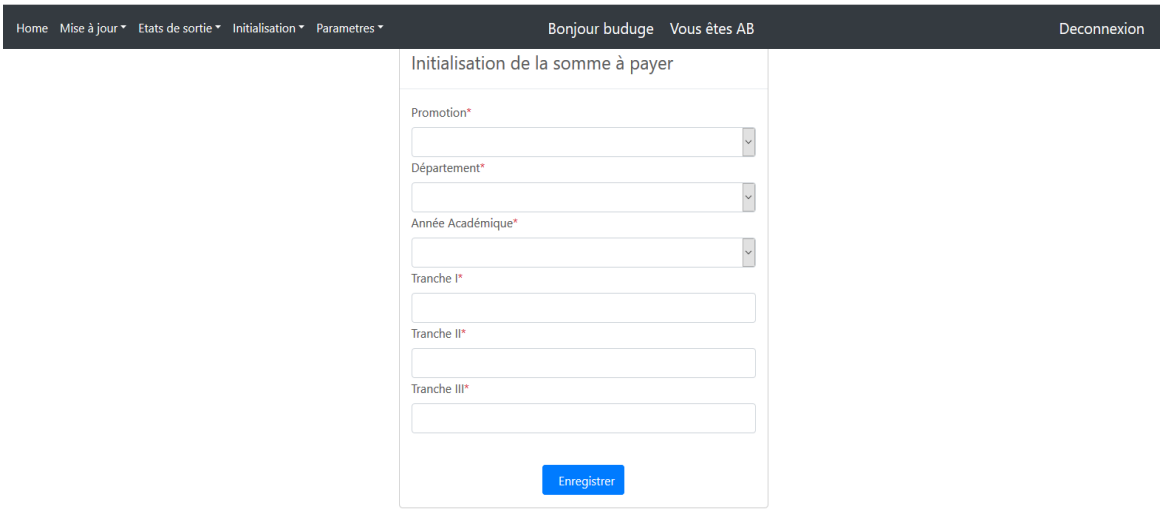


Figure 48.*Interface d'Initialisation de la somme à payer*

- Modifier son compte

La figure suivante montre l'interface de la modification de compte

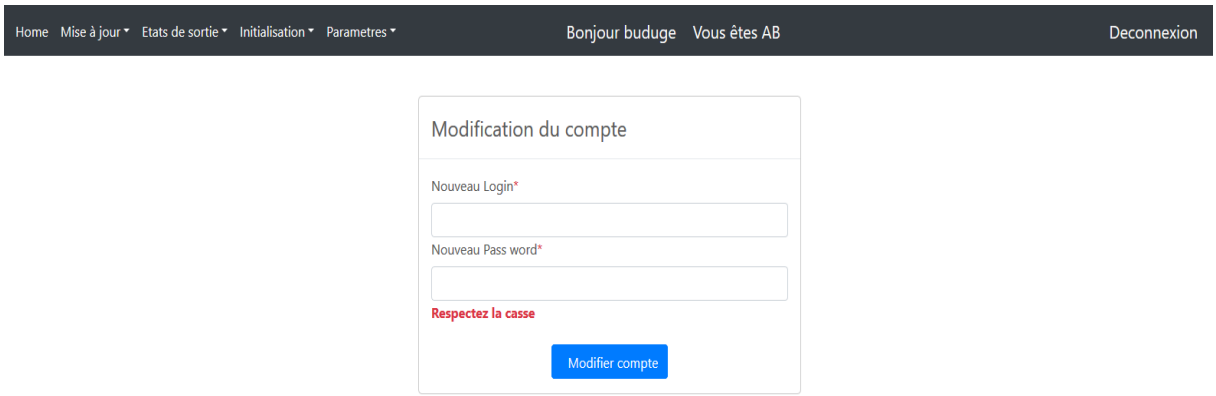


Figure 49.*Interface de modification de compte*

A part ces tâches qu'il réalise directement dans le système, il imprime les rapports suivants en PDF et en Excel :

- Le relevé de perception journalière en pdf

L'administrateur de budget sélectionne la date du jour et l'utilisateur, ensuite il clique sur le bouton chercher.

La figure suivante montre l'interface de recherche de la perception journalière

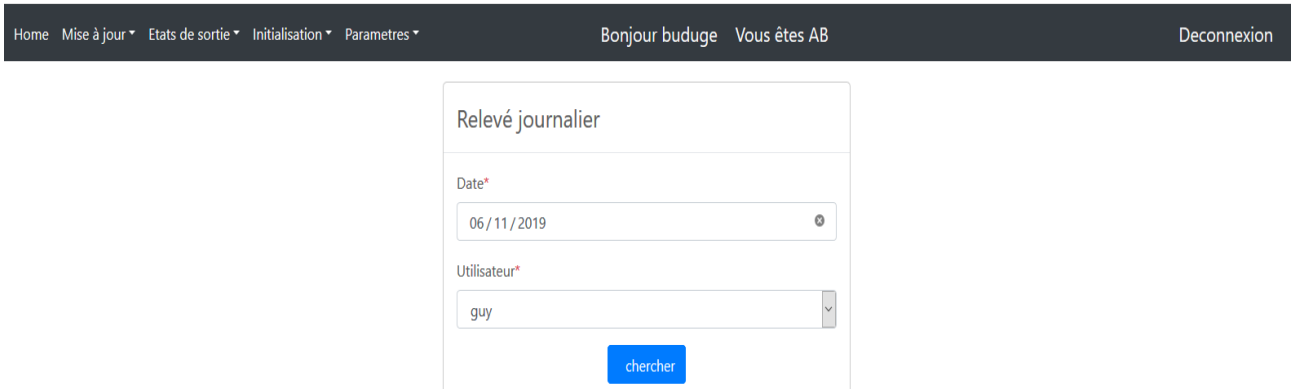
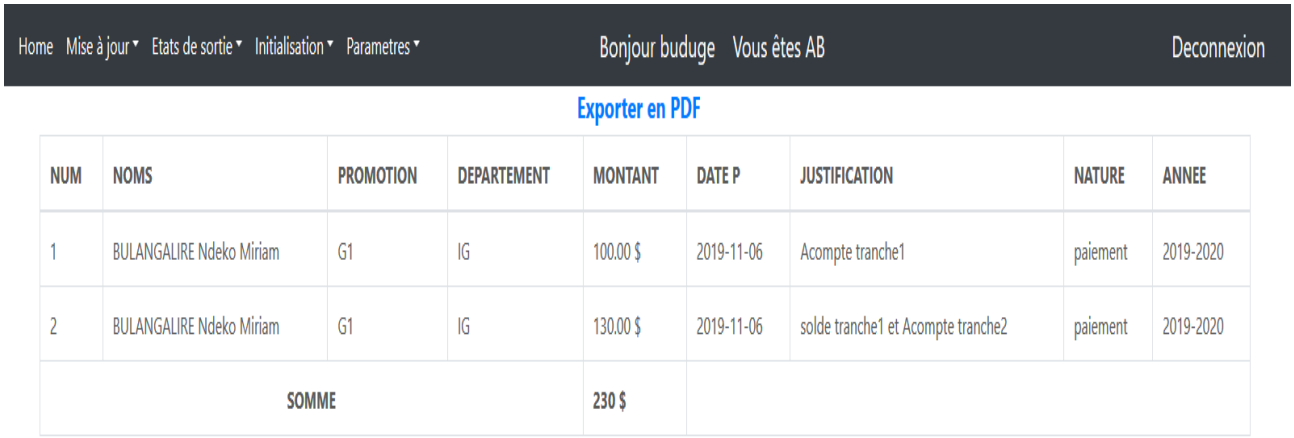


Figure 50. *Interface de recherche de la situation journalière*

Si la requête retourne le résultat, voici comment ce dernier s'afficherait dans le tableau html :

La figure suivante montre le résultat de la recherche précédente en html



NUM	NOMS	PROMOTION	DEPARTEMENT	MONTANT	DATE P	JUSTIFICATION	NATURE	ANNEE
1	BULANGALIRE Ndeko Miriam	G1	IG	100.00 $	2019-11-06	Acompte tranche1	paiement	2019-2020
2	BULANGALIRE Ndeko Miriam	G1	IG	130.00 $	2019-11-06	solde tranche1 et Acompte tranche2	paiement	2019-2020
SOMME				230 $				

Figure 51.*Situation Journalière en html*

S'il veut imprimer ce rapport en pdf, il clique sur exporter en pdf. Voici la présentation du rapport en pdf. La figure suivante montre le résultat de la recherche précédente en pdf.

ETABLISSEMENT PUBLIC
INSTITUT SUPERIEUR PEDAGOGIQUE
SERVICE DE PERCEPTION DES FRAIS ACADEMIQUES
ANNEE ACADEMIQUE: 2019-2020
RELEVE JOURNALIER DE guy DU 2019-11-06

RECU	NOMS	POST NOM	PRENOM	PROM.	DEP.	MONTANT	LIBELLES
79	BULANGALIRE	Ndeko	Miriam	G1	IG	100.00	Acompte tranche1
80	BULANGALIRE	Ndeko	Miriam	G1	IG	130.00	solde tranche1 et Acompte tranche2

TOTAL MONTANT PERCU : 230 $

SERVICE DE PERCEPTION

Figure 52.*Situation Journalière en PDF*

- Le relevé de perception mensuelle en pdf

L'administrateur de budget sélectionne le mois, l'année académique et l'utilisateur, en suite il clique sur le bouton chercher. La figure suivante montre l'interface de recherche de la perception mensuelle.

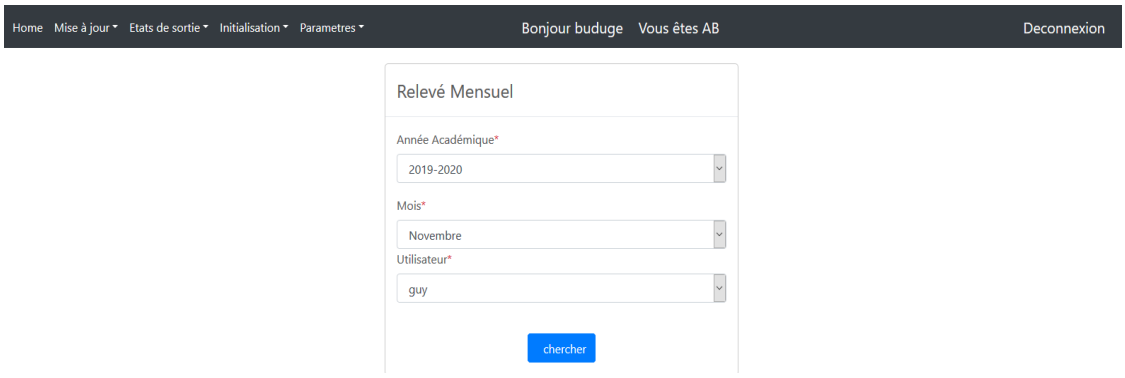
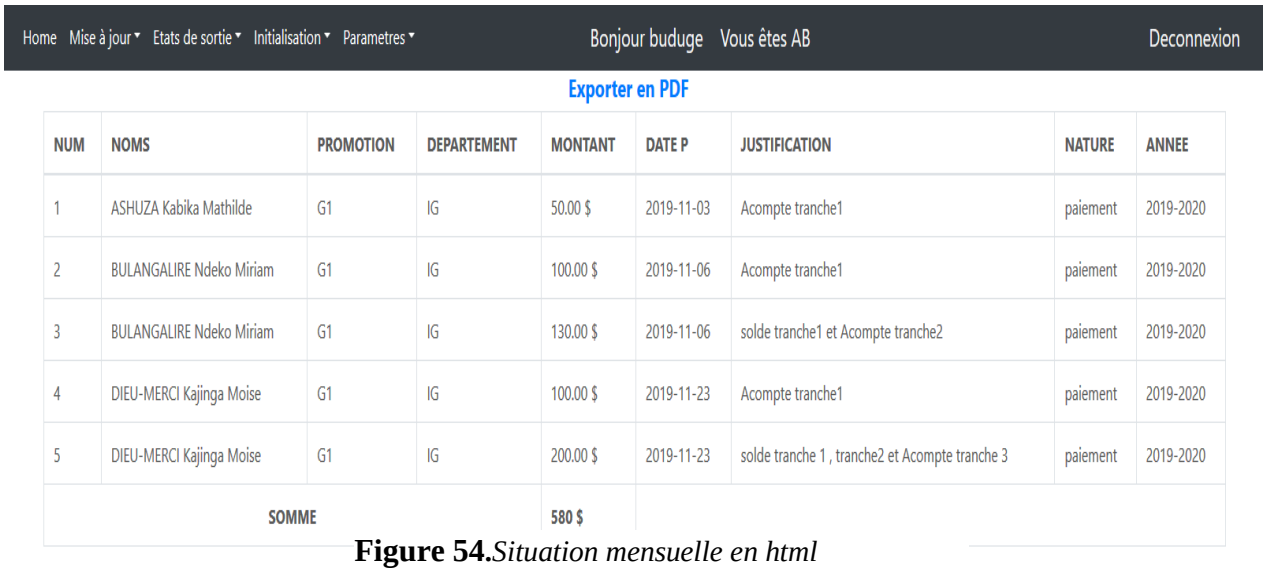


Figure 53.*L'interface de recherche de la situation mensuelle*

Lorsque la requête s'exécute avec succès, voici le résultat au format html et téléchargeable en pdf. La figure suivante montre le résultat de la requête précédente en html.

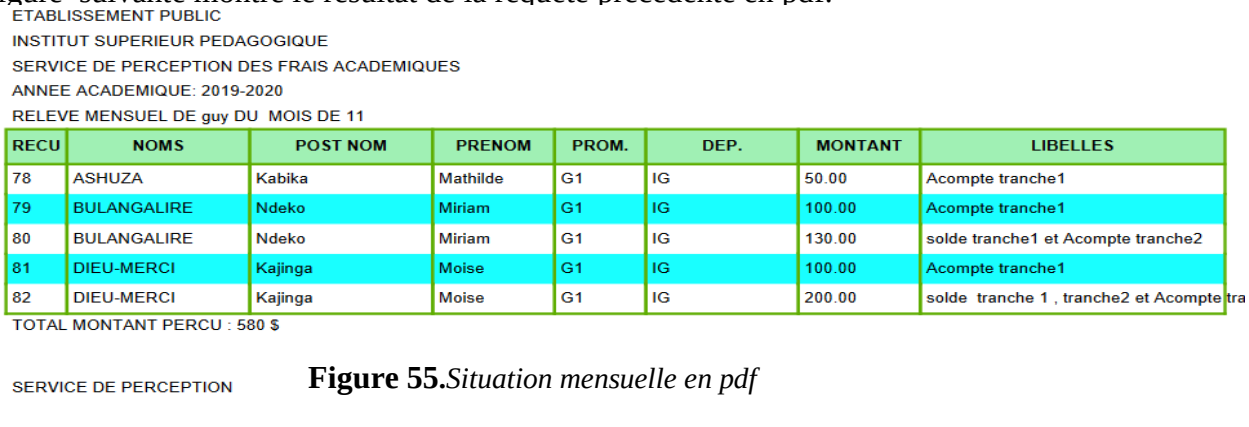


NUM	NOMS	PROMOTION	DEPARTEMENT	MONTANT	DATE P	JUSTIFICATION	NATURE	ANNEE
1	ASHUZA Kabika Mathilde	G1	IG	50.00 $	2019-11-03	Acompte tranche1	paiement	2019-2020
2	BULANGALIRE Ndeko Miriam	G1	IG	100.00 $	2019-11-06	Acompte tranche1	paiement	2019-2020
3	BULANGALIRE Ndeko Miriam	G1	IG	130.00 $	2019-11-06	solde tranche1 et Acompte tranche2	paiement	2019-2020
4	DIEU-MERCI Kajinga Moise	G1	IG	100.00 $	2019-11-23	Acompte tranche1	paiement	2019-2020
5	DIEU-MERCI Kajinga Moise	G1	IG	200.00 $	2019-11-23	solde tranche 1 , tranche2 et Acompte tranche 3	paiement	2019-2020
		SOMME		580 $				

Figure 54.*Situation mensuelle en html*

Le rapport en pdf se présente de cette façon:

La figure suivante montre le résultat de la requête précédente en pdf.



ETABLISSEMENT PUBLIC
INSTITUT SUPERIEUR PEDAGOGIQUE
SERVICE DE PERCEPTION DES FRAIS ACADEMIQUES
ANNEE ACADEMIQUE: 2019-2020
RELEVE MENSUEL DE guy DU MOIS DE 11

RECU	NOMS	POST NOM	PRENOM	PROM.	DEP.	MONTANT	LIBELLES
78	ASHUZA	Kabika	Mathilde	G1	IG	50.00	Acompte tranche1
79	BULANGALIRE	Ndeko	Miriam	G1	IG	100.00	Acompte tranche1
80	BULANGALIRE	Ndeko	Miriam	G1	IG	130.00	solde tranche1 et Acompte tranche2
81	DIEU-MERCI	Kajinga	Moise	G1	IG	100.00	Acompte tranche1
82	DIEU-MERCI	Kajinga	Moise	G1	IG	200.00	solde tranche 1 , tranche2 et Acompte tra

TOTAL MONTANT PERCU : 580 $

SERVICE DE PERCEPTION

Figure 55.*Situation mensuelle en pdf*

- Le relevé de perception annuelle en pdf

L'administrateur de budget sélectionne l'année académique et l'utilisateur, ensuite il clique sur le bouton chercher. La figure suivante montre l'interface de recherche de la perception annuelle.

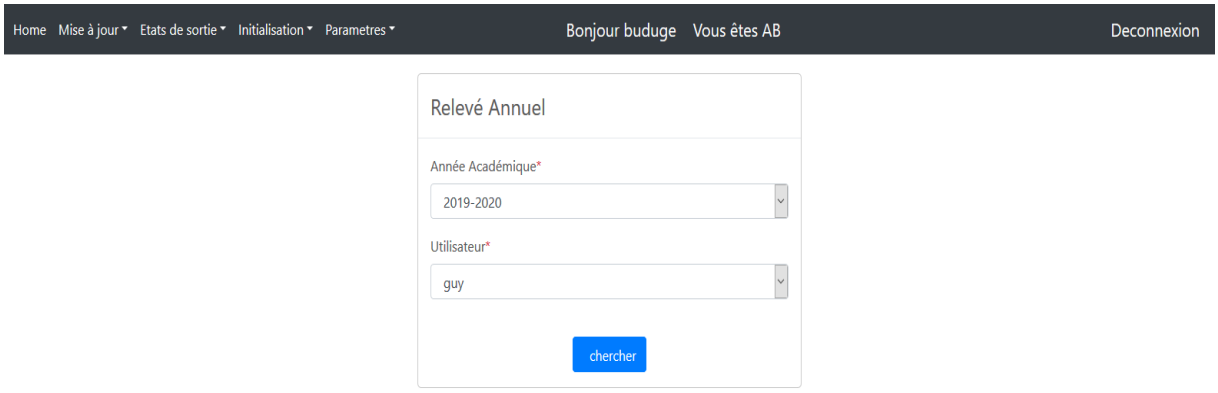
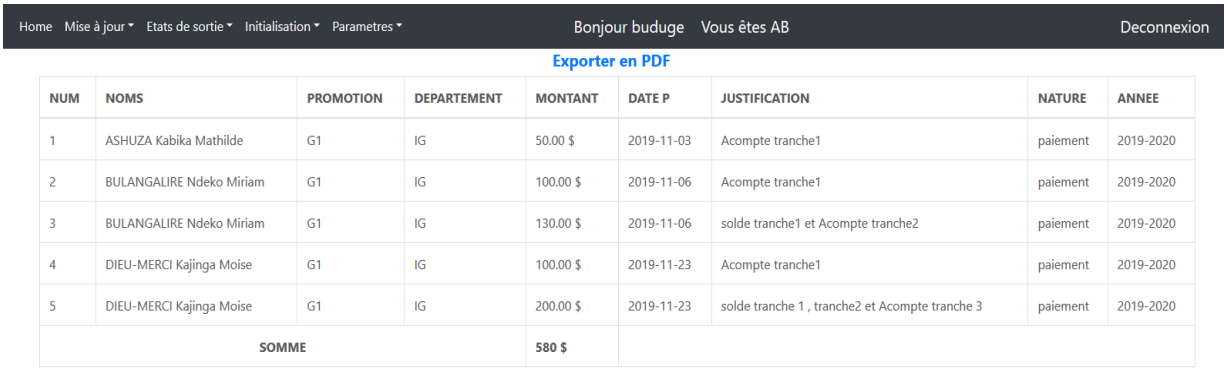


Figure 56.*Interface de recherche de la situation annuelle*

Le résultat html exportable en pdf se présente comme suit:

La figure suivante affiche le résultat de la requête précédente en html.



NUM	NOMS	PROMOTION	DEPARTEMENT	MONTANT	DATE P	JUSTIFICATION	NATURE	ANNEE
1	ASHUZA Kabika Mathilde	G1	IG	50.00 $	2019-11-03	Acompte tranche1	paiement	2019-2020
2	BULANGALIRE Ndeko Miriam	G1	IG	100.00 $	2019-11-06	Acompte tranche1	paiement	2019-2020
3	BULANGALIRE Ndeko Miriam	G1	IG	130.00 $	2019-11-06	solde tranche1 et Acompte tranche2	paiement	2019-2020
4	DIEU-MERCI Kajinga Moise	G1	IG	100.00 $	2019-11-23	Acompte tranche1	paiement	2019-2020
5	DIEU-MERCI Kajinga Moise	G1	IG	200.00 $	2019-11-23	solde tranche 1 , tranche2 et Acompte tranche 3	paiement	2019-2020
SOMME				580 $				

Figure 57. *Situation annuelle en format html*

La figure suivante affiche le résultat de la requête précédente en html

ETABLISSEMENT PUBLIC
INSTITUT SUPERIEUR PEDAGOGIQUE
SERVICE DE PERCEPTION DES FRAIS ACADEMIQUES
ANNEE ACADEMIQUE: 2019-2020
RELEVE ANNUEL DE guy

RECU	NOMS	POST NOM	PRENOM	PROM.	DEP.	MONTANT	LIBELLES
78	ASHUZA	Kabika	Mathilde	G1	IG	50.00	Acompte tranche1
79	BULANGALIRE	Ndeko	Miriam	G1	IG	100.00	Acompte tranche1
80	BULANGALIRE	Ndeko	Miriam	G1	IG	130.00	solde tranche1 et Acompte tranche2
81	DIEU-MERCI	Kajinga	Moise	G1	IG	100.00	Acompte tranche1
82	DIEU-MERCI	Kajinga	Moise	G1	IG	200.00	solde tranche 1 , tranche2 et Acompte tr

TOTAL MONTANT PERCU : 580 $

SERVICE DE PERCEPTION

Figure 58.*Situation annuelle en format pdf*

- Le relevé journalier synthétique en Excel

L'administrateur de budget a besoin de sélectionner la date du jour pour afficher le résultat en html.

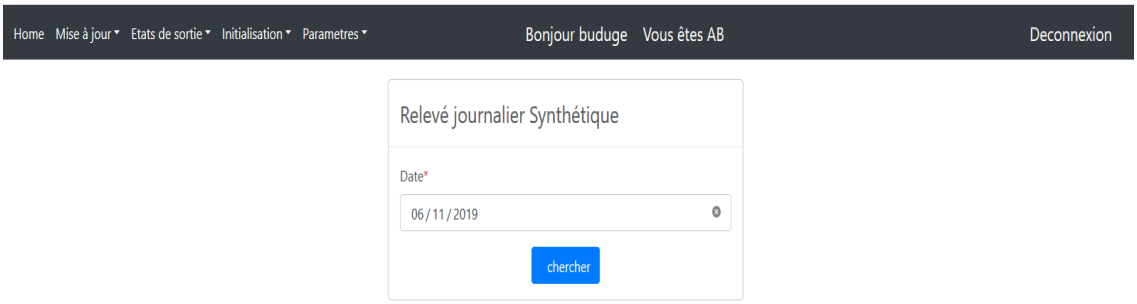


Figure 59.*Interface de recherche de la situation synthétique journalière*

Le résultat html se présente comme suit:

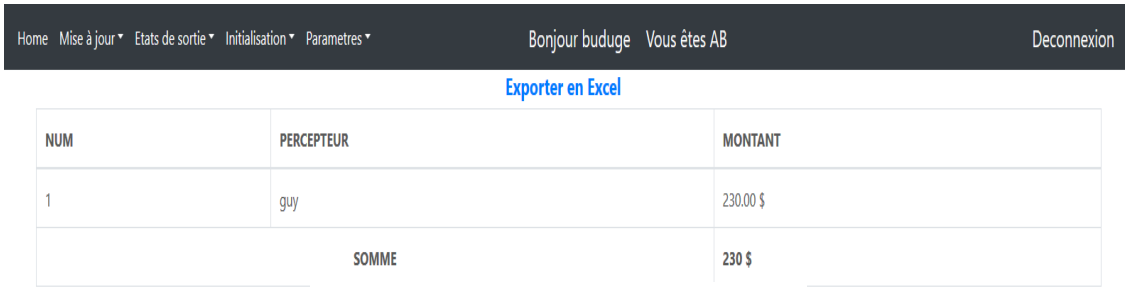


Figure 60.*Situation synthétique html*

S'il choisit de cliquer sur exporter le résultat en format Excel, voici comment le rapport se présente :

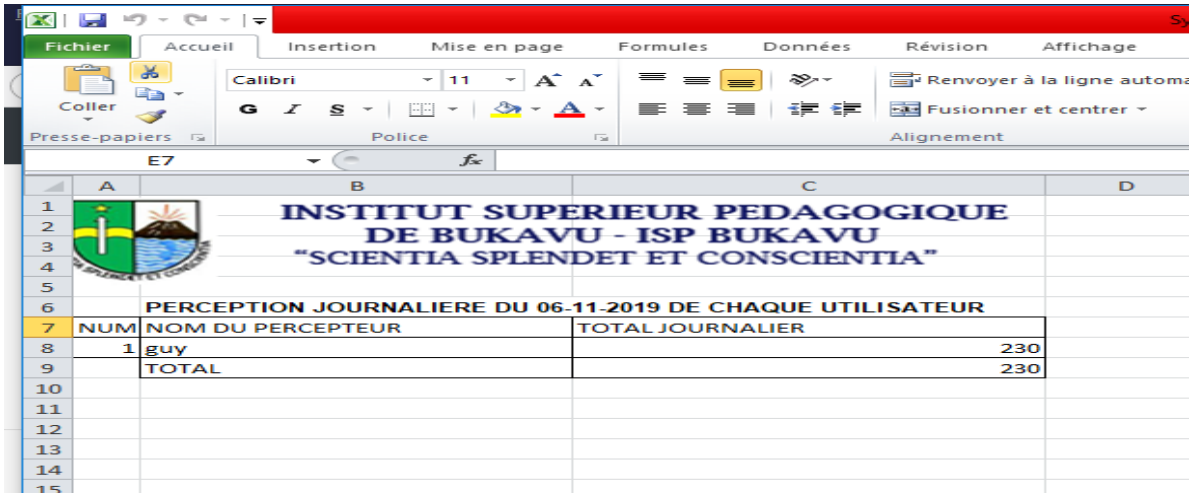


Figure 61.*Situation synthétique en Excel*

- Le relevé de chaque promotion en pdf

Pour afficher la situation des étudiants en rapport avec leur promotion, l'administrateur de budget sélectionne la promotion, le département ainsi que l'année, ensuite il clique sur chercher.

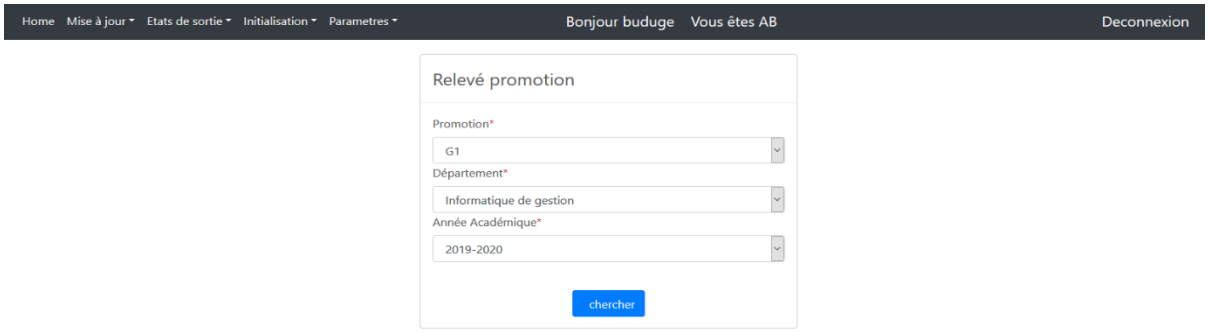
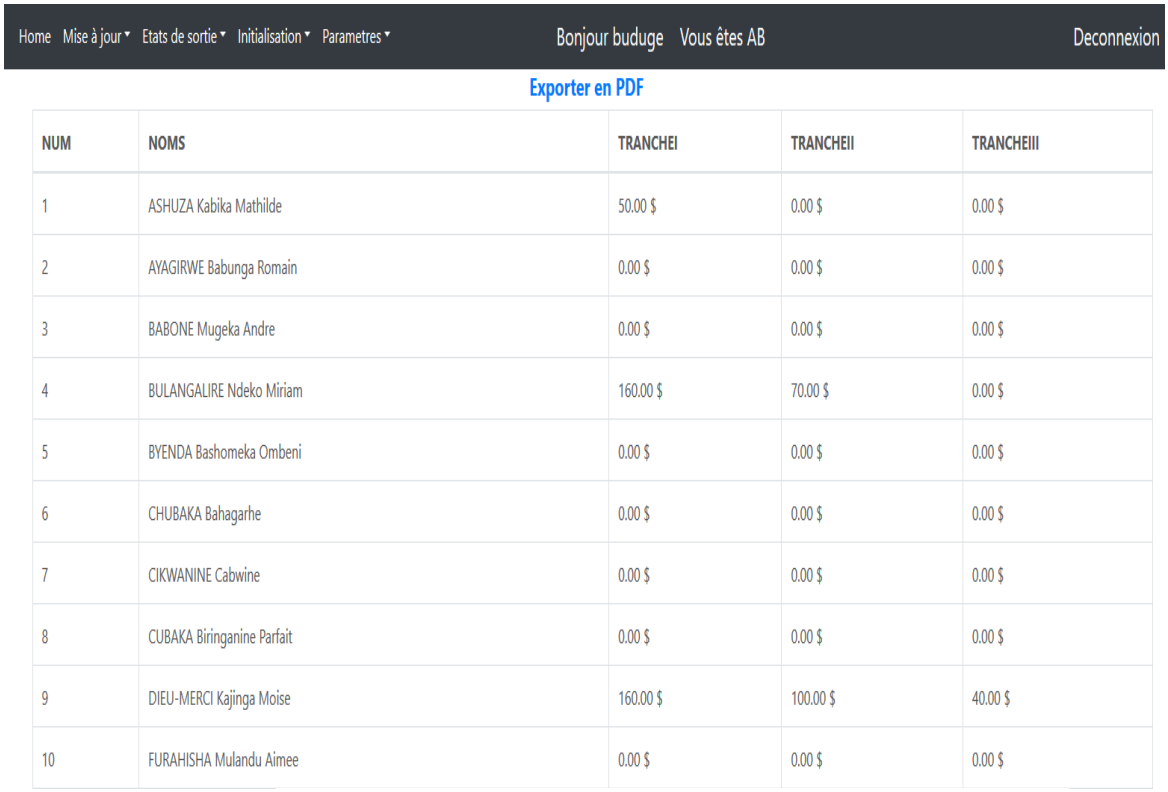


Figure 62.*Interface de recherche de la situation par promotion*

En cas de succès de la requête, le résultat en html se présente de cette manière **[Figure62]:**



NUM	NOMS	TRANCHEI	TRANCHEII	TRANCHEIII
1	ASHUZA Kabika Mathilde	50.00 $	0.00 $	0.00 $
2	AYAGIRWE Babunga Romain	0.00 $	0.00 $	0.00 $
3	BABONE Mugeka Andre	0.00 $	0.00 $	0.00 $
4	BULANGALIRE Ndeko Miriam	160.00 $	70.00 $	0.00 $
5	BYENDA Bashomeka Ombeni	0.00 $	0.00 $	0.00 $
6	CHUBAKA Bahagarhe	0.00 $	0.00 $	0.00 $
7	CIKWANINE Cabwine	0.00 $	0.00 $	0.00 $
8	CUBAKA Biringanine Parfait	0.00 $	0.00 $	0.00 $
9	DIEU-MERCI Kajinga Moise	160.00 $	100.00 $	40.00 $
10	FURAHISHA Mulandu Aimee	0.00 $	0.00 $	0.00 $

Figure 63.*Résultat de la situation par promotion en html*

Le rapport de ce résultat en pdf est:

ETABLISSEMENT PUBLIC
INSTITUT SUPERIEUR PEDAGOGIQUE
SERVICE DE PERCEPTION DES FRAIS ACADEMIQUES
RELEVE DE PAIEMENT DE LA G1 -IG- 2019-2020

NUM	NOM	POST NOM	PRENOM	TRANCHE I	TRANCHE II	TRANCHE III
1	ASHUZA	Kabika	Mathilde	50.00	0.00	0.00
2	AYAGIRWE	Babunga	Romain	0.00	0.00	0.00
3	BABONE	Mugeka	Andre	0.00	0.00	0.00
4	BULANGALIRE	Ndeko	Miriam	160.00	70.00	0.00
5	BYENDA	Bashomeka	Ombeni	0.00	0.00	0.00
6	CHUBAKA	Bahagarhe		0.00	0.00	0.00
7	CIKWANINE	Cabwine		0.00	0.00	0.00
8	CUBAKA	Biringanine	Parfait	0.00	0.00	0.00
9	DIEU-MERCI	Kajinga	Moise	160.00	100.00	40.00
10	FURAHISHA	Mulandu	Aimee	0.00	0.00	0.00
11	HERITIER	Ezeckia	Jean	0.00	0.00	0.00
12	IMANI	Mbonekube	Chrispin	0.00	0.00	0.00
13	ISANDA	Wonwa	Philippe	0.00	0.00	0.00
14	JACQUES	Maseruka	Thomas	0.00	0.00	0.00
15	KANSILEMBO	Masumbuko	Aristote	0.00	0.00	0.00
16	KITOKO	Salumu	Gratien	0.00	0.00	0.00
17	KYUBWA	Mutono		0.00	0.00	0.00

Figure 64.*Résultat de la situation par promotion en pdf*

- Le relevé étudiant en pdf

L'administrateur de budget saisit les noms de l'étudiant et sélectionne l'année académique concernée, ensuite il clique sur le bouton chercher.



Figure 65.*Interface de recherche de la situation de chaque étudiant par année*

Le résultat en format html se présente comme suit:



Figure 66.*Résultat de la situation de l'étudiant en html*

Et voici le rapport correspondant en pdf:

ETABLISSEMENT PUBLIC

INSTITUT SUPERIEUR PEDAGOGIQUE

SERVICE DE PERCEPTION DES FRAIS ACADEMIQUES

SITUATION DE L'ETUDIANT BULANGALIRE Ndeko Miriam G1 IG 2019-2020

NUM	MONTANT	DATE PAIEMENT	LIBELLES	PERCEPTEUR
1	100.00	2019-11-06	Acompte tranche1	guy
1	130.00	2019-11-06	solde tranche1 et Acompte tranche2	guy

TOTAL MONTANT PERCU : 230 $

SERVICE DE PERCEPTION

Figure 67.*Résultat de la situation de l'étudiant en pdf*

- La statistique en Excel

Avec la statistique, le système affiche la situation chiffrée des étudiants en ordre pour une tranche sélectionnée, par promotion, département et par année **[Figure67]**

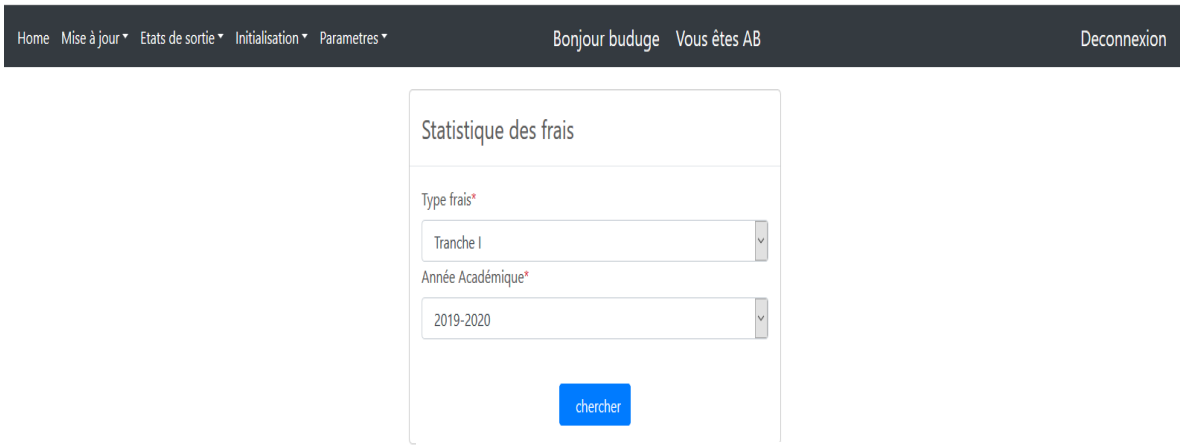


Figure 68.*Interface d'affichage des statistiques*

Le système génère automatiquement le rapport en Excel que voici:



INSTITUT SUPERIEUR PEDAGOGIQUE DE BUKAVU
SERVICE DES FINANCES

STATISTIQUE DES ETUDIANTS QUI ONT DEJA PAYE LA PREMIERE TRANCHE
2019-2020

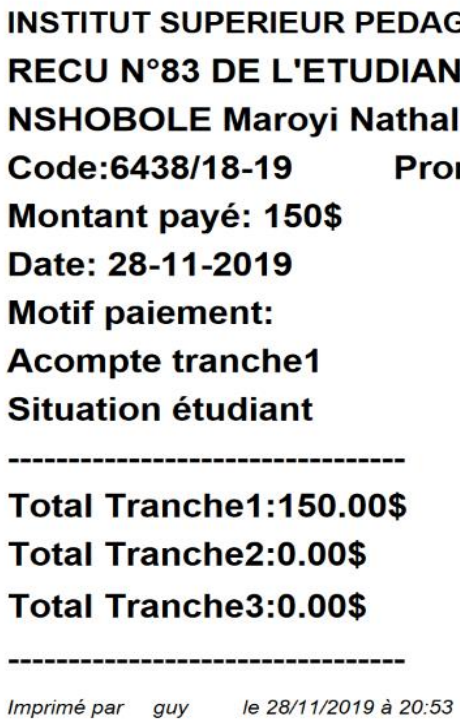
N°	DEPARTEMENTS	G1	G2	G3	L1	L2	TOTAL
1	Anglais culture africaine	0	0	0	0	0	0
2	Français-langues africaines	0	0	0	0	0	0
3	Histoire sciences sociales	0	0	0	0	0	0
4	Hotellerie, accueil et tourisme	0	H:0 T:0	H:0 T:0	H:0 T:0	H:0 T:0	0
5	Biologie-Chimie	0	0	0	0	0	0
6	Chimie-Physique	0	0	0	0	0	0
7	Géographie et gestion de l'environnement	0	0	0	0	0	0
8	Mathématique-Physique	0	0	0	0	0	0
9	Physique-Technologie	0	0	0	0	0	0
10	Informatique de Gestion	2	0	0	0	0	2
11	Sciences Commerciales et Administratives	0	0	0	0	0	0
	TOTAL	2	0	0	0	0	2

H:Hotellerie
T:Tourisme

Imprimé le 28-11-2019 à 19:53

Figure 69.*Statistique en Excel*

Voici comment se présente le reçu de paiement d'un étudiant en pdf :


INSTITUT SUPERIEUR PEDAGOGIQUE
RECU N°83 DE L'ETUDIANT:
NSHOBOLE Maroyi Nathalie
Code:6438/18-19 Promotion:G1 IG
Montant payé: 150$
Date: 28-11-2019
Motif paiement:
Acompte tranche1
Situation étudiant

Total Tranche1:150.00$
Total Tranche2:0.00$
Total Tranche3:0.00$

Imprimé par guy le 28/11/2019 à 20:53

Figure 70 .*Modèle du reçu d'un étudiant*

Conclusion partielle

Ce chapitre a été destiné à la présentation graphique des interfaces qui sont implémenté dans notre système.

Vous remarquerez que pour un seul utilisateur, on présente au moins 10 pages et pourtant nous en avons beaucoup, c'est pourquoi nous nous abstenons de présenter les actions des autres utilisateurs considérant que l'administrateur de budget est l'acteur principal.

CONCLUSION GENERALE ET RECOMMANDATIONS

Conclusion

Ce travail avait pour objectif principal la conception d'une application sécurisée de gestion des frais académiques à l'Institut Supérieur Pédagogique de Bukavu.

Il a fallu dans un premier temps concevoir des crypto systèmes afin de les implémenter dans notre application de gestion des frais académiques avec comme objectif la garantie de la sécurité des données. Le développement de ces mécanismes de sécurité et la conception d'un système informatisé de gestion des frais académiques vise à résoudre les problèmes tels que la perte des données due à la vétusté des documents, la difficulté de produire des rapports fiables en temps réel, limiter les accès non autorisés.

Tenant compte de ces problèmes, nous nous sommes fixé les objectifs spécifiques de concevoir une application sécurisée de gestion déployée en réseau de générer et d'imprimer les rapports journaliers, mensuels et annuels de la perception, d'élaborer la liste des créanciers, de concevoir des algorithmes d'upload des listes des étudiants vers notre système (excel-php-mysql) et de concevoir des fonctions des chiffrages des données stockées dans le système pour une meilleure sécurité.

Ainsi donc, nous sommes parvenu à la réalisation d'une application informatique qui a témoigné notre démarche tout au long de notre rédaction, le résultat est sans doute celui de la conception et de la réalisation d'une application sécurisée de gestion des frais académiques à l'Institut Supérieur Pédagogique de Bukavu.

Cette recherche ouvre une nouvelle voie à nos successeurs c'est pourquoi nous invitons d'autres chercheurs à aborder les autres aspects liés au cursus académique des étudiants (délibération, enseignement, stage, logement, etc.).

Recommandations

Après avoir implémenté le système de gestion sécurisée des frais académiques, nous recommandons à l'Institut Supérieur Pédagogique de Bukavu ce qui suit :

- Acquérir tous les matériels nécessaires pour le déploiement du système
- Utiliser tous les aspects du système
- Former tous les utilisateurs en Excel d'abord et ensuite à l'utilisation du système
- Engager les moyens nécessaires pour l'information des autres services de l'institution.

BIBLIOGRAPHIE

[1] Conallen J. , *Concevoir des applications Web avec UML*, Eyrolles, 2000

[2] D'Souza D.F. et Wills A.C., *Objects, Components and Frameworks with UML,* Addison-Wesley, 1999

[3] van Wijk, C. Gilles*, Théorie des projets*, Paris, Editions Ellipses*,* février 2020

[4] N. Kettani, D. Mignet, P. Paré, C. Rosenthal-Sabroux, *de merise à uml***,** ISBN 2-212-08997-X, Editions Eyrolles

[5] Gamma E., E. Helm, R. Johnson et J. Vlissides2, *Design Patterns, Elements of Reusable Object-Oriented Software,* Addison-Wesley, 1995

[6] Muller P.A. et N. Gaertner, *Modélisation objet avec UML*, Eyrolles, 2000

[7] ROQUES, P. et VALLEE, F., *UML 2 en action de l'analyse des besoins à la conception,* Ed. EYROLLES, Paris, 2005.

[8] Andreas Meier, *Introduction pratique aux bases dedonnées relationnelles*, Collection IRIS Springer, 2006(ISBN 2-287-25205-3)

[9] Codd, E.F, *A Relational Model of Data for Large SharedData Banks,* CACM 13 N6 juin 1970

[10] Laurent AUDIBERT, *Bases de données de la modélisationau SQL,* Ellipses, 2009 (ISBN 978-2-7298-5120-0

[11] Christian SOUTOU, *Programmer avec MySQL*, Ed.Eyrolle, 2015 5eme édition (ISBN XXXX)

[12] Jérôme GABILLAUD, *SQL et Algèbre Relationnelle*, 3emeédition ENI, 2010 (ISBN 978-2-7460-5472-1)

[13] Addison-Wesley , *The Logical Level*, Inc., 1995, 685 p.

[14] *Apprendre les bases de données et SQL,* sur Developpez.com (consulté le 18 Décembre 2019)

[15] CHEY COBB, *Sécurité réseaux pour les nuls,* First Interactive, New York, 2003, p.287 - 288.

[16] CLAUDE SERVIN, *Réseaux & télécoms,* Edition Dunod, Paris, 2003, 2006.

[17]*https://moodle.utc.fr/pluginfile.php/16777/mod_resource/content/0/SupportIntroSecu/res/chiffrement-sym.png*, url valide le 28 Mai 2019 à 21h 15'

[18]*https://moodle.utc.fr/pluginfile.php/16777/mod_resource/content/0/SupportIntroSecu/res/chiffr-asym.png*, url valide le 28 Mai 2019 à 21h 35'

[19] Stallings W. *Network Security Essentials,* 2nd edition, Prentice Hall, 2003.

[20] FONTAINE Phillipe et RAFFEGEAU Romain, *Qui sont les pirates du Net ?,* Science & Vie Junior, n°303, Décembre 2014, pages 28-37