

REPUBLIQUE DE COTE D'IVOIRE



UNION – DISCIPLINE - TRAVAIL

Ministère de L'Economie Numérique  
et de la Poste

Ecole Supérieure Africaine des Technologies  
De l'Information et de la Communication



Année académique : 2019/2020

## MÉMOIRE DE PROJET INTERNE

Master 2 Système d'Information et  
Génie Logiciel

# MISE EN PLACE D'UN RELEVÉ DE NOTES NUMÉRIQUE POUR L'ESATIC

Présenté par :

KOFFI YAO JEAN BETHUEL

KONE SIDNEY AZIZ KHADER

KOUADIO STEVENS SEPRAPHIN

JURY :

**Président** : Prof. KEITA Mélalie, Professeur titulaire à UNA

**Rapporteur** : Dr. KAMAGATE BEMAN, Enseignant-chercheur à ESATIC

**Membre** : M. SILUE Mariame Epse COULIBALY, Enseignant à l'ESATIC

### Encadrants académiques :

Prof. KONATE Adama  
Maître de conférences.

M. SILUE Kolo  
Enseignant à ESATIC.

# **DEDICACE**

---

Nous dédions ce mémoire à nos familles respectives !

# **REMERCIEMENTS**

---

Avant tout propos, nous tenons à exprimer notre profonde gratitude à des personnes qui, d'une manière ou d'une autre, ont contribué à la réalisation du présent document. Sans pour autant être exhaustifs, nous aimerons citer :

- Pr KONATE Adama, actuel Directeur Général de l'ESATIC et notre encadrant, pour tous les efforts faits pour notre réussite ;
- M SILUE Kolo, notre encadrant pédagogique, pour tous ses conseils et orientation dans la réalisation du projet ;
- Tout le corps professoral et administratif de l'ESATIC, pour le savoir transmis ;
- Tous ceux qui nous ont soutenus et continuent de nous soutenir par leurs actes, prières et pensées.

# **SOMMAIRE**

---

## **PREMIERE PARTIE : GENERALITES**

### **CHAPITRE I : PRESENTATION DU PROJET**

#### **I. DEFINITION DES NOTIONS CLES**

#### **II. RELEVÉ DE NOTES**

### **CHAPITRE II : ETUDE DE L'EXISTANT**

#### **I. FONCTIONNEMENT DU SYSTEME D'INFORMATION ACTUEL**

#### **II. LIMITES DE L'EXISTANT**

#### **III. PRESENTATION DU CAHIER DE CHARGES**

## **DEUXIEME PARTIE : ANALYSE ET CONCEPTION**

### **CHAPITRE III : ANALYSE DES BESOINS**

#### **I. SPECIFICATION DES BESOINS FONCTIONNELS**

#### **II. SPECIFICATION DES BESOINS NON FONCTIONNELS**

#### **III. ARCHITECTURE DU SYSTEME**

### **CHAPITRE IV : CONCEPTION**

#### **I. METHODE DE CONCEPTION**

#### **II. IDENTIFICATION DES ACTEURS**

#### **II. DIAGRAMMES**

### **CHAPITRE V : PROCESSUS DE SECURITE**

#### **I. BESOINS NECESSAIRES DE SECURITE**

#### **II. TECHNIQUE DE CHRYPTAGE CHOISIE**

#### **III. PROCESSUS DE CHIFFREMENT**

## **TROISIEME PARTIE : REALISATION ET MISE EN PLACE DE LA SOLUTION**

### **CHAPITRE VI : ENVIRONNEMENT DE TRAVAIL**

#### **I. RESSOURCES MATERIELLES ET LOGICIELS**

#### **II. TECHNOLOGIE UTILISER**

### **CHAPITRE VII : REALISATION ET DISCUSSION**

#### **I. PAGE D'AUTHENTIFICATION**

#### **II. ESPACE PROFESSEUR**

#### **III. ESPACE ETUDIANTS**

## **LISTE DES TABLEAUX**

Tableau 1 Ajouter les informations personnelles d'un étudiant.....	20
Tableau 2 : Ajouter les informations personnelles d'un professeur.....	21
Tableau 3 : Ajouter les informations d'une classe .....	22
Tableau 4 : Ajouter les informations personnelles d'un agent de la Direction Pédagogique. .	23
Tableau 5 : Consulter et faire des réclamations par étudiant .....	24
Tableau 6 : Ajouter les notes d'étudiant.....	25
Tableau 7 : Comparaison des techniques cryptographiques .....	33
Tableau 8 : Modifier les informations personnelles d'un étudiant.....	46
Tableau 9 : Supprimer les informations personnelles d'un étudiant. ....	46
Tableau 10 : Consulter les informations personnelles d'un étudiant. ....	46
Tableau 11 : Modifier les informations personnelles d'un professeur. ....	47
Tableau 12 : Supprimer les informations personnelles d'un professeur. ....	47
Tableau 13 : Consulter les informations personnelles d'un professeur .....	48
Tableau 14 : Modifier les informations d'une classe. ....	48
Tableau 15 : Consulter les informations d'une classe.....	48
Tableau 16 : Attribuer une UE et ECUE dans une classe. ....	49
Tableau 17 : Modifier les informations personnelles d'un agent de la Direction Pédagogique. .....	49
Tableau 18 : Supprimer les informations personnelles d'un agent de la Direction Pédagogique .....	50
Tableau 19 : Consulter les informations personnelles d'un agent de la Direction Pédagogique .....	50

## **LISTE DES FIGURES**

Figure 1 : Architecture 3-tiers .....	14
Figure 2 : Diagramme de cas d'utilisation lié aux administrateurs .....	19
Figure 3 : Diagramme de cas d'utilisation pour la gestion des professeurs .....	20
Figure 4 : Diagramme de cas d'utilisation pour la planification des emplois du temps des professeurs et des étudiants .....	21
Figure 5 : Diagramme de cas d'utilisation pour gérer les utilisateurs.....	22
Figure 6 : Diagramme de cas d'utilisation pour les étudiants .....	23
Figure 7 : Diagramme de cas d'utilisation pour les professeurs .....	24
Figure 8 : Diagramme de cas d'utilisation pour l'agent de saisie. ....	25
Figure 9 : Diagramme de cas d'utilisation lié aux agents de vérification.....	25
Figure 10 : Diagramme de cas d'utilisation lié au directeur pédagogique.....	26
Figure 11 : Diagramme des classes de notre système. ....	27
Figure 12 : Diagramme de séquence- Ajout de note par le professeur .....	29
Figure 13 : Diagramme de séquence – Consultation de note par l'étudiant .....	30
Figure 14 : Diagramme de séquence de réclamation de note et réponse à la réclamation .....	31
Figure 15 : Page d'authentification .....	38
Figure 16 : Page des classes du professeur .....	38
Figure 17 : Page des matières du professeur dans une classe. ....	39
Figure 18 : Page d'administration des notes dans une classe.....	39
Figure 19 : Page de calendrier du professeur. ....	40
Figure 20 : Page d'authentification de l'étudiant .....	41
Figure 21 : Page du profil de l'étudiant.....	42
Figure 22 : Relevé de note.....	45

# **INTRODUCTION**

---

Depuis son existence la société fait face à un ensemble de problèmes. L'homme est constamment à la recherche d'une amélioration de son mode de vie et des pratiques sociales. Le secteur de l'éducation occupe une place importante dans le développement d'un pays. Cependant le traitement des informations et l'établissement des documents se font de façon manuelle. Les méthodes manuelles traditionnelles ont montré leurs limites dans la gestion de grands flux d'informations, dans des tâches répétitives.

La nouvelle organisation du travail demande aux établissements d'apprentissage d'utiliser essentiellement l'information comme matière première pour pouvoir être plus efficace, ils doivent s'adapter, si possible anticiper et réagir avec agilité. Ainsi, il nous a été fondamental de voir qu'aujourd'hui les écoles publiques en particulier celle de l'ESATIC auquel nous attacherons notre étude sont des établissements où l'intérêt d'automatiser le processus de traitement, d'archivage et de gestion des notes devient évident. Le défi est de garantir les bonnes gestions des notes des étudiants et de réduire au maximum les risques d'erreurs.

Après une étude approfondie sur le mode de fonctionnement de cette école, utilisant un système informatique, nous avons rencontré quelques difficultés tels que :

- Volume important de notes d'examen traitées par une seule personne, ce qui provoque parfois des erreurs de saisir des notes et dans l'établissement des relevés ;
- Le processus d'obtention du relevé de notes et des résultats est long ;
- Possibilité d'erreur de saisie des différentes notes et moyenne ;
- Possibilité d'erreur dans le calcul des statistiques.

Comment la mise en place d'un système de gestion des notes des étudiants pourrait redynamiser la gestion actuelle de l'ESATIC et quel sera l'impact de cette automatisation du système ?

L'objectif fixé est d'atténuer les problèmes qui se posent lors de l'après notes et examen, notamment lors de la collecte des notes, du traitement des notes, de l'affichage des résultats et l'édition de relevé de notes. Notre travail consiste à informatiser les tâches suivantes : (relevé de note, liste des étudiants, liste des admis, moyenne des notes, classement

des étudiants par ordre). Les avantages souhaités de cette application sont d'avoir un accès rapide à toutes les notes et moyennes par les étudiants et professeurs, assurer un suivi des notes par les étudiants qui pourront faire de réclamations en temps réel sur notre plateforme.

Pour atteindre cet objectif, nous avons porté réflexion sur les études relatives au projet et subdivisé le travail en trois grandes parties. Dans la première partie, nous présenterons les informations générales à la bonne compréhension du projet. Elle fera aussi l'objet d'une analyse préalable qui présentera la collecte des informations sur l'existant, la problématique, l'objectif de notre travail. La deuxième, consistera à faire la spécification des besoins et la conception de notre application. Et la dernière sera consacrée à la présentation des résultats de l'application et sa mise en œuvre.



## **PREMIERE PARTIE : GENERALITES**

## **CHAPITRE I : PRESENTATION DU PROJET**

La présentation d'un projet permet d'avoir une vision claire de celui-ci. C'est pourquoi, dans ce chapitre, nous définissons les notions clés de notre projet.

### **I. DEFINITION DES NOTIONS CLES**

#### **1. Site web**

Un site Web, ou simplement site, est un ensemble de pages web et de ressources reliées par des hyperliens, défini et accessible par une adresse web. Il est composé d'un ensemble de documents structurés, nommés pages web, stockés (hébergés) sur un ordinateur (serveur) connecté au réseau mondial. Il existe deux types de sites web :

- Un site statique : constitué de pages web avec du contenu fixe. Chaque page est codée en HTML (HyperText Markup Language) et affiche les mêmes informations à chaque visiteur. La modification de ces pages se fait par une intervention de l'administrateur du site web. [2] ;
- Un site dynamique : les pages du site qui le compose peuvent être modifiables par les visiteurs. De plus, entre deux visites sur un même site, le contenu de la page peut être différent sans action de l'administrateur du site Internet.

#### **2. Application web**

Une application web est un ensemble de pages qui interagissent avec les utilisateurs, les unes avec les autres, ainsi qu'avec les différentes ressources d'un serveur web, notamment les bases de données [3]. L'architecture d'une application web contient un serveur d'application qui rend possible le traitement d'une logique et d'un état applicatif. Tandis que l'architecture d'un site web contient trois composants principaux :

- Un serveur web ;
- Une connexion réseau ;
- Des navigateurs clients.

En d'autres termes une application web sert à manipuler l'information et elle est soumise à des contrôles d'accès et de rôles, et un site web sert à exposer l'information. [4]

### 3. Le serveur web

Un serveur web est un logiciel qui fournit des pages web en réponse à des requêtes de navigateurs web. Une requête de page est générée lorsqu'un utilisateur clique sur un lien d'une page web ou saisit une URL (localisateur uniforme de ressources) dans le champ adresse du navigateur. [5]

### 4. Hébergement d'une application

L'hébergement est un service visant à rendre un site ou une application web accessible sur Internet. Afin que les pages soient visibles par tout le monde, il faut qu'elles soient stockées sur un ordinateur connecté en permanence à l'internet (serveur). Un hébergeur est comme une société spécialisée dans l'hébergement de sites web sur des serveurs, sa principale activité sur internet consiste à installer des serveurs, à les sécuriser, à les tenir à jour en installant les mises à jour de sécurité pour éviter les attaques malveillantes et à les réparer en cas de panne. [6].

Il existe plusieurs types d'hébergements dont :

- Hébergement gratuit : ce type d'hébergement offre un service basique, avec un espace de stockage assez limité, avec une fiabilité qui n'est pas toujours optimale ;
- Hébergement payant : ce type d'hébergement propose un service de qualité, avec de bonnes capacités de stockage ainsi qu'une grande fiabilité. Le support sera également plus disponible et plus réactif en cas de problème et aidera avec des réponses personnalisées.

### 5. Application mobile

Une application mobile est un logiciel développé spécifiquement pour une plate-forme (l'OS pour Operating System–Système d'exploitation) assurant la distribution des applications. Contrairement à un site web développé en HTML et qui se consulte en ligne au travers d'un navigateur (Internet Explorer, Chrome, Safari, Firefox, ...), une application se télécharge, généralement via un magasin d'applications public, comme l'App Store d'Apple ou Google Play pour Android.[7]

## II. RELEVÉ DE NOTES

Liste récapitulative établie par un établissement d'enseignement ou un organisme scolaire, et indiquant les cours suivis par un élève ainsi que la note obtenue pour chacun d'eux [8]. Le relevé de notes présente tous les renseignements sur votre formation suivie dans un établissement, par cycle d'études, et les activités non contributives à votre programme, et il indique la période d'études pour chacun d'entre eux, il comprend :

- Votre moyenne par session.
- Votre moyenne par cycle.
- Votre moyenne de diplomation, à la fin de votre formation.

Il peut être consulté en ligne ou être obtenu officiellement auprès de l'administration. Notre solution prendra en compte le relevé de notes numérique et officiel.

### 1. Relevé de notes numérique

Disponible en tout temps dans son dossier étudiant sous notre plateforme, le relevé de notes numérique sert uniquement à la consultation. Bien que sa présentation diffère de la version officielle, le contenu qui y figure est le même. Une copie imprimée du relevé de notes électronique n'est pas un document officiel. Elle sera refusée par les organismes et les autres établissements universitaires.

### 2. Relevé de notes officiel

Le relevé de notes officiel peut être exigé pour déposer une demande d'admission dans un autre établissement universitaire, demander une bourse ou postuler un emploi. Il est imprimé sur du papier comportant des caractéristiques de sécurité, portant une signature et le sceau de l'ESATIC. Seuls les relevés de notes dont la scolarité a été menée entièrement doivent porter le sceau de l'ESATIC.

Dans la première section de cette partie, nous avons vu quelques notions sur les technologies web et mobile, et nous avons présenté le relevé de notes, numériques et officiel. Dans la partie suivante nous ferons une analyse préalable qui consistera à présenter l'existant, identifier la problématique et proposer une solution.

## **CHAPITRE II : ETUDE DE L'EXISTANT**

Ce projet est né de certaines limites constatées dans notre environnement. C'est pourquoi, dans ce chapitre nous présentons le fonctionnement du système actuel de gestion des notes de l'ESATIC dans le but d'en faire une analyse.

### **I. FONCTIONNEMENT DU SYSTEME D'INFORMATION ACTUEL**

La gestion de notes à l'ESATIC se fait à l'aide d'un système d'information. Il se fait selon un calendrier déterminé par le Directeur de la Pédagogie. Il part du lancement d'examens à l'affichage des résultats mais il n'est pas toujours respecté à causes de certaines raisons indépendantes de la volonté des acteurs du système. Cela fait que les étudiants n'ont pas la possibilité de connaître d'avance la date de sortie des résultats. Elle se fait de la façon suivante :

- Les enseignants doivent saisir les moyennes de leurs étudiants par classe sur une plateforme en ligne ;
- Après le lancement des examens et la correction des feuilles de copies, toutes les feuilles sont remises à un agent de saisie de la Direction Pédagogique ;
- L'agent de saisie, ayant un profil différent, doit saisir les notes d'examen sur la même plateforme. Il est la seule personne chargée de le faire, aucun contrôle, jusqu'à la délibération, n'est fait après saisie et validation des notes par l'agent ;
- Toutes les notes et moyennes des étudiants sont disponibles en ligne, par niveau, classe et matière. Cela servira pour la délibération ;
- Des modifications annoncées par le chef de jury peuvent être nécessaires et l'agent de la direction pédagogique doit les appliquer au PV. Finalement les décisions de réussite ou de redoublement sont notées sur le PV et le résultat est affiché sur le tableau d'affichage de l'établissement ;
- L'étudiant fait une demande du relevé de note (exemple : voir en annexe 1) auprès des éducateur, son relevé est imprimé par l'informaticien puis transmis à la secrétaire générale qui le signe ;

## II. LIMITES DE L'EXISTANT

A partir du mode de fonctionnement défini ci-dessus nous parlerons des faiblesses de cette méthode et de la manipulation de notes, en essayant de dégager les problèmes que rencontrent les jurys, les professeurs et les agents du service des examens. Nous avons interrogé la direction de la pédagogie qui nous a cité quelques anomalies, mais pour localiser leur source, nous nous sommes mis en pratique avec elles. Et après une observation, nous avons pu observer les insuffisances suivantes :

- L'étudiant ne sait à quel moment seront disponibles ses résultats, car le temps de saisie des notes et la délibération est parfois indéterminée ;
- L'Edition d'un relevé de note prend assez de temps. En effet la demande passe par processus de formulaire à remplir, une phase de vérification des résultats en ligne et ceux sur le PV signé par les professeurs, une d'impression et de signature ;
- Pas de services de contrôle dans le processus de saisie de notes des examens, une seule personne est chargée de le faire et c'est ce qu'il saisit qui est pris en compte ;
- Non-intégrité des notes, les notes saisies par l'agent peuvent être modifiées par toutes autres personnes ayant accès à la base de données ;
- Les professeurs n'entre pas leurs notes de contrôle continu en ligne. L'intérêt ici sera de permettre au professeur de saisir les notes d'étudiants sur notre plateforme. Ainsi les étudiants pourront avoir la possibilité de voir leur note et moyenne directement en ligne et le professeur n'aura plus besoin de remettre une fiche, sur laquelle est saisie les notes et moyennes, au délégué afin que les autres étudiants puissent faire des vérifications.

Nous constatons que la solution informatique dans le but de redynamiser ou améliorer les services est la plus adéquate puisqu'elle répond mieux aux anomalies rencontrées dans la gestion actuelle. Ainsi nous avons décidé de concevoir une application web et mobile qui va gérer les activités de l'ESATIC et qui va permettre par la suite d'améliorer la rapidité de l'accès aux notes et moyennes par les professeurs et les étudiants.

### III. PRESENTATION DU CAHIER DE CHARGES

#### 1. Contexte général

Pour obtenir son relevé de notes l'étudiant devra faire une demande, en remplissant un formulaire, auprès de l'administration. Certaines demandes durent parfois une semaine tandis que d'autres peuvent durer un ou deux mois. Ainsi plusieurs étudiants se sont plaints de cette situation. Aussi les étudiants utilisent leur relevé de notes pour postuler à des demandes de bourse ou de demande d'admission dans une autre université. Cependant, il y a certains étudiants qui ont la possibilité de falsifier ce relevé et postuler aux offres. L'ère du numérique offre des technologies permettant d'assurer la confiance en un produit notamment la technologie blockchain, ce projet veut rassurer les acteurs (ESATIC et les entreprises) de l'authenticité des relevés de notes.

#### 2. Objectifs

##### *a Objectif général*

L'objectif général de ce projet est de réaliser une application web et mobile pour améliorer la gestion de notes et d'édition de relevé.

##### *b Objectifs spécifiques*

Plusieurs objectifs sont visés à travers ce projet :

- Planifier les emplois de temps des classes et des professeurs ;
- Saisir des notes de contrôle continu par les professeurs ;
- Calculer la moyenne par matière et par module pour chaque étudiant ;
- Calculer systématiquement la moyenne générale du semestre de chaque étudiant ;
- Mise à disposition des résultats en temps réel aux étudiants ;
- Imprimer les relevés de notes de chaque étudiant ;
- Gérer les utilisateurs ;
- Automatiser les tâches qui se traitent manuellement (Saisir de notes de contrôle continu par les professeurs...).

#### 3. Résultat attendu

Les livrables attendus aux vues de ces objectifs sont :

- Une application mobile destinée aux étudiants ;
- Une application web destinée aux professeurs, à la direction pédagogique et aux étudiants ;
- Un manuel d'utilisation pour les utilisateurs finaux.

Tenant compte de l'analyse préalable faite précédemment, il est nécessaire de déterminer concrètement dans la prochaine partie les spécifications des besoins fonctionnels et non fonctionnels et de faire une conception de la solution proposée.



## **DEUXIEME PARTIE : ANALYSE ET CONCEPTION**

## **CHAPITRE III : ANALYSE DES BESOINS**

Comme n'importe quel type de projet, un projet informatique nécessite une phase d'analyse, suivi d'une étape de conception. Dans la phase d'analyse, on cherche d'abord à bien comprendre et à décrire de façon précise les besoins des étudiants et de la direction pédagogique. Que souhaitent-ils faire avec le logiciel ? Quelles fonctionnalités veulent-ils ? Pour quel usage ? Comment l'action devrait-elle fonctionner ? C'est ce qu'on appelle « l'analyse des besoins »

### **I. SPECIFICATION DES BESOINS FONCTIONNELS**

Les besoins fonctionnels ou besoins métiers représentent les actions que le système doit exécuter. Selon notre application, nous avons identifié les besoins fonctionnels :

- Ajouter, modifier, supprimer et consulter les comptes des administrateurs ;
- Ajouter, modifier, supprimer et consulter les classes ;
- Ajouter, modifier et consulter les spécialités ;
- Ajouter, modifier et consulter les UE et ECUE ;
- Ajouter, modifier, supprimer et consulter les professeurs ;
- Ajouter, modifier et consulter les notes de contrôles continu et de travaux pratiques de chaque étudiant ;
- Ajouter et consulter les notes d'examen de chaque étudiant ;
- Calculer les moyennes de contrôles continus, du semestre et générale de chaque étudiant ;
- Faire des réclamations sur les notes de contrôles continus de travaux pratiques ;
- Consulter la statistique des résultats ;
- Imprimer le relevé de notes de chaque étudiant.

### **II. SPECIFICATION DES BESOINS NON FONCTIONNELS**

Les besoins non fonctionnels présentes des exigences internes au système et cachées aux utilisateurs :

- Exigence d'ergonomie
  - Une interface conviviale, lisible et facile à utiliser ;

- Rapidité du temps de réponse car il joue un rôle fondamental. Il doit être court ;
- Utiliser des listes contenant des valeurs correctes afin d'éviter la saisie et par conséquent les erreurs de saisie ;
- Exigences de la sécurité
  - Prévenir contre l'accès direct avec les liens URL et définir un délai de temps pour la fermeture de session non active ;
  - La Traçabilité et la journalisation des transactions et des opérations effectuées ;
  - Le système doit être sécurisé avec l'obligation pour chaque étudiant, professeur, administrateur de saisir un mot de passe et un login ;
  - L'authentification : Elle est assurée par un module d'authentification et d'autorisation, il gère l'accès entre les utilisateurs et l'application par un login et un mot de passe ;
  - La confidentialité et l'intégrité : Elles sont assurées par un chiffrement symétrique des notes et moyennes de chaque étudiant et aussi par un hashage du mot de passe défini par chaque étudiant, professeur, administrateur.

### **III. ARCHITECTURE DU SYSTEME**

Une architecture est un modèle générique et conceptuel qui se rapporte à un sujet et qui représente la fonctionnalité, la structure, le positionnement, l'interrelation des différents types d'éléments (hardware, logiciels, infrastructure) qui la composent.

En règle générale, une application est découpée en 3 niveaux (couches) d'abstraction :

- ❖ La couche présentation : c'est la partie de l'application visible par les utilisateurs (interface utilisateur). Dans notre cas, cette couche est un navigateur web, qui se présente sous forme de pages HTML, composée parfois de formulaire et de bouton ;
- ❖ La couche métier : correspond à la partie fonctionnelle de l'application, celle qui implémente la logique, et qui décrit les opérations que l'application opère sur les données, en fonction des requêtes d'un utilisateur effectuées au travers de la couche présentation ;
- ❖ La couche accès aux données : elle consiste en la partie gérant l'accès à la base de données du système ;

Il existe différentes architectures pour une application web :

- Architecture 1-tiers, Les trois couches (présentation, traitements et les données) s'exécutent sur le même serveur ;
- Architecture 2-tiers, les couches présentation et traitements se trouve chez le client et les données sur un serveur ;
- Architecture 3-tiers.

Pour notre application nous allons utiliser une architecture 3-tiers illustrée dans la figure ci-dessous :

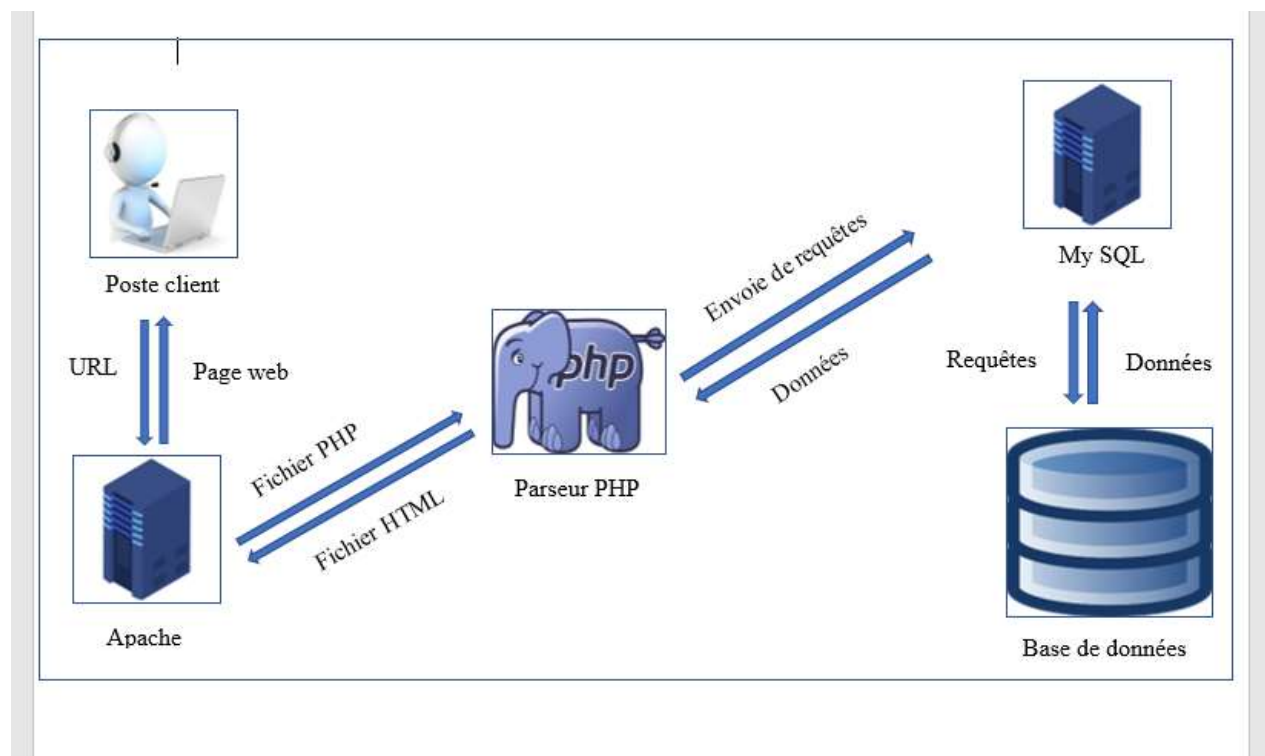


Figure 1 : Architecture 3-tiers

Une application web basée sur un modèle Client - Serveur à une architecture à trois niveaux (appelée architecture 3-tiers), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre :

- Un client, c'est-à-dire l'ordinateur demandeur de ressources, équipée d'une interface utilisateur (généralement un navigateur web) chargée de la présentation ;

- Le serveur d'application (appelé également middleware), chargé de fournir la ressource mais faisant appel à un autre serveur ;
- Le serveur de données, fournissant au serveur d'application les données dont il a besoin.

Nous avons choisi cette architecture car elle nous permet :

- La prise en compte de l'hétérogénéité des plates-formes (serveurs, clients, langages, etc.) ;
- L'amélioration de la sécurité des données, en supprimant le lien entre le client et les données. Le serveur a pour tâche, en plus des traitements purement métiers, de vérifier l'intégrité et la validité des données avant de les envoyer dans la couche d'accès aux données ;
- La rupture du lien de propriété exclusive entre application et données. Dans ce modèle, la base de données peut être plus facilement normalisée et intégrée à un entrepôt de données ;
- Une meilleure répartition de la charge entre différents serveurs d'application [5].

Au cours de ce chapitre nous avons identifié les besoins fonctionnels et non fonctionnels de notre application. Dans le chapitre suivant nous traiterons les méthodologies de conception.

## **CHAPITRE IV : CONCEPTION**

Dans le cycle de vie de notre projet, la conception représente une phase primordiale et déterminante pour produire une application de haute qualité.

Dans ce qui suit nous présenterons la méthode de conception, puis nous passerons à l'identification des acteurs, ainsi nous décrirons les diagrammes de cas d'utilisation, de classes, séquences, nous construirons le modèle relationnel à partir des classes de l'application, en suivant les règles du passage au relationnel.

### **I. METHODE DE CONCEPTION**

Le développement de n'importe quel système d'information nécessite une démarche très importante dans le cycle de vie d'un logiciel, la modélisation en pratique est importante dans les développements des logiciels, dans notre projet nous choisirons le langage de modélisation UML

UML, se définit comme un langage de modélisation graphique et textuel destiné à comprendre et définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. Il véhicule en particulier :

- Les concepts des approches par objets, classes, instance, classification ... ;
- Intégrant d'autres aspects : associations, fonctionnalités, événements, états, séquences....

Ainsi, UML définit 9 types de diagrammes dans deux catégories de vues, les vues statiques et les vues dynamiques.

Vues statiques :

- ❖ Les diagrammes de cas d'utilisation : ils décrivent le comportement et les fonctions d'un système du point de vue de l'utilisateur ;
- ❖ Les diagrammes de classes : ils décrivent la structure statique, les types et les relations des ensembles d'objets ;
- ❖ Les diagrammes d'objets : ils décrivent les objets d'un système et leurs relations ;

- ❖ Les diagrammes de composants : ils décrivent les composants physiques et l'architecture interne d'un logiciel ;
- ❖ Les diagrammes de déploiement : ils décrivent la répartition des programmes exécutables sur les différents matériels.

Vues dynamiques :

- ❖ Les diagrammes de collaboration : ils décrivent les messages entre objets (liens et interactions) ;
- ❖ Les diagrammes d'états-transitions : ils décrivent les différents états d'un objet ;
- ❖ Les diagrammes d'activités : ils décrivent les comportements d'une opération (en termes d'actions) ;
- ❖ Les diagrammes de séquence : ils décrivent de manière temporelle les interactions entre objets et acteur.

Pour la modélisation des besoins et de la conception, nous utilisons les diagrammes UML suivant : diagramme de cas d'utilisation, diagramme des classe, diagramme des séquences, et diagramme de déploiement.

## **II. IDENTIFICATION DES ACTEURS**

Un acteur est une entité externe qui agit sur le système. Les acteurs de notre système sont :

- Les administrateurs ;
- Le directeur pédagogique ;
- Les agents de la direction pédagogique ;
- Les professeurs ;
- Les étudiants.

### **1. Les administrateurs**

On distingue quatre types d'administrateur pour notre système à savoir :

- Un administrateur chargé de la gestion des étudiants. Cet administrateur a pour tâche d'ajouter, modifier, supprimer et consulter la liste des étudiants ;
- Un administrateur chargé de la gestion des professeurs. Cet administrateur a pour tâche d'ajouter, modifier, supprimer et consulter la liste des professeurs ;

- Un administrateur chargé de la planification des UE (unité d'enseignements) et ECUE (élément constitutif d'unité d'enseignements). Cet administrateur a pour tâche d'ajouter, modifier une classe et d'attribuer une UE et/ou ECUE à un professeur dans une classe ;
- Un administrateur chargé de la gestion des utilisateurs. Cet administrateur a pour tâche d'ajouter, modifier, supprimer des comptes des agents de la direction pédagogique et celui du directeur pédagogique.

## 2. Le directeur pédagogique

Le rôle du directeur pédagogique dans notre système est :

- D'avoir une vue sur les notes de contrôles continus, de travaux pratiques et d'examen des étudiants ;
- D'avoir une vue de la statistique de résultat de chaque étudiant par niveau, par classe et par spécialité.

## 3. Les agents de la direction pédagogique

On distingue deux types d'agent de la direction pédagogique dans notre système à savoir :

- Des agents de saisit de notes d'examen. Ces agents ont pour tâche de saisir les notes d'examen des étudiants ;
- Des agents de validation et de vérification des notes saisies. Ces agents ont pour tâche de vérifier et valider les notes saisies.

## 4. Les professeurs

Le rôle des professeurs est de saisir les notes de contrôle continue et de travaux pratiques de leurs étudiants et de calculer les moyennes de contrôles continus de chaque étudiant respectif

## 5. Les étudiants

Les étudiants consultent leurs notes de contrôle continu et de travaux pratique, leurs moyennes de contrôle continu, leurs notes d'examen et leurs moyennes du semestre. Ils peuvent faire des réclamations et consulter leurs relevés de notes.



## II. DIAGRAMMES

### 1. Diagramme de cas d'utilisation

#### a Définition

Les diagrammes de cas d'utilisation décrivent le comportement du système du point de vue de l'utilisateur sous la forme d'actions et de réactions. Un cas d'utilisation indique une fonctionnalité du système déclenché par un acteur externe au système.

#### b Composition

Un diagramme de cas d'utilisation se compose de trois éléments principaux :

- ❖ Un Acteur : c'est l'idéalisation d'un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système. Il se représente par un petit bonhomme avec son nom inscrit en dessous ;
- ❖ Un cas d'utilisation : c'est une unité cohérente représentant une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service. Il représente par une ellipse contenant le nom du cas (un verbe à l'infinitif), et optionnellement, au-dessus du nom, un stéréotype ;
- ❖ Les 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.

#### c Diagramme de cas d'utilisation lié aux administrateurs

##### Gérer étudiant

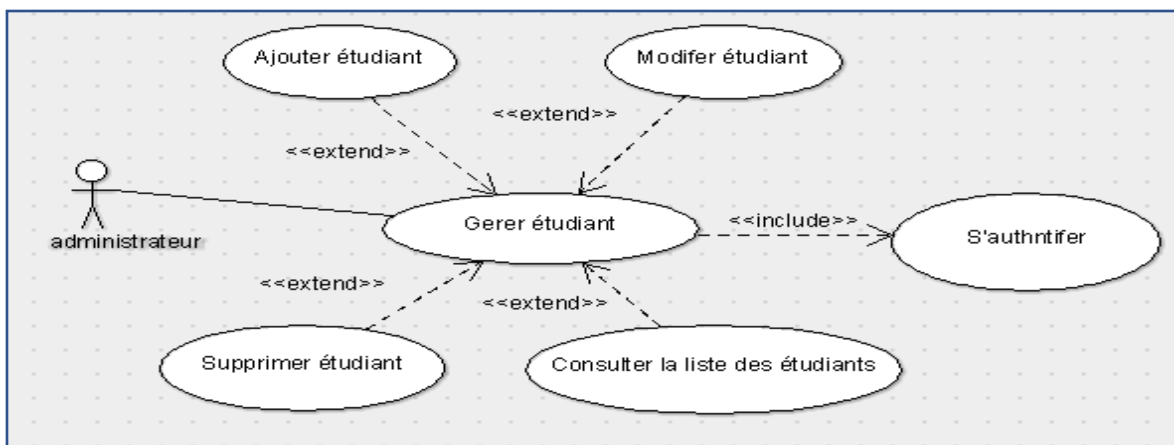


Figure 2 : Diagramme de cas d'utilisation lié aux administrateurs

Tableau 1 Ajouter les informations personnelles d'un étudiant

<b>Acteur</b>	L'administrateur chargé de la gestion des étudiants
<b>Pré condition</b>	-S'authentifier
<b>Post condition</b>	L'ajout de l'étudiant est réussi et enregistrer dans la base de données
<b>Description du scénario</b>	- L'administrateur saisit les informations correspondant au nouvel étudiant et confirme sa demande d'ajout. - Le système ajoute le nouvel étudiant à la base de données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Si l'étudiant existe le système affiche un message d'erreur.

- Description du processus de modification d'informations personnelles d'un étudiant (voir tableau 8, placé en annexe 2) ;
- Description du processus de suppression d'informations personnelles d'un étudiant (voir tableau 9, placé en annexe 2) ;
- Description du processus de consultation d'informations personnelles d'un étudiant (voir tableau 10, placé en annexe 2).

### Gérer professeur

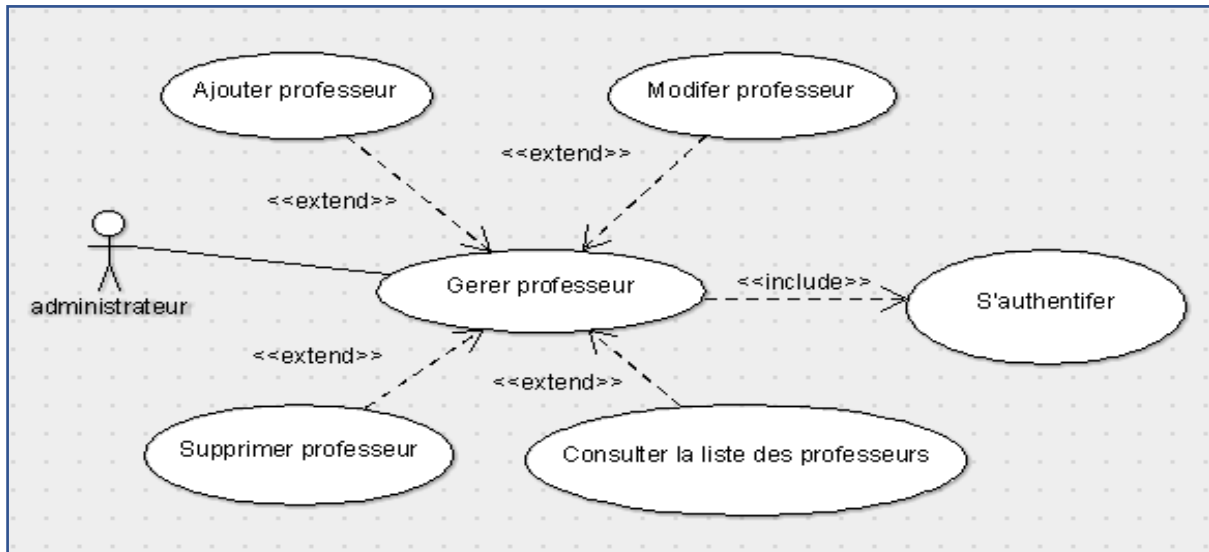


Figure 3 : Diagramme de cas d'utilisation pour la gestion des professeurs

Tableau 2 : Ajouter les informations personnelles d'un professeur

<b>Acteur</b>	L'administrateur chargé de la gestion des professeurs
<b>Pré condition</b>	- S'authentifier - Sélectionner le professeur
<b>Post condition</b>	L'ajout du professeur est réussi et enregistrer dans la base de données
<b>Description du scénario</b>	- L'administrateur saisit les informations correspondant au nouveau professeur et confirme sa demande d'ajout. - Le système ajoute le professeur à la base des données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Si le professeur existe le système affiche un message d'erreur.

- Description du processus de modification d'informations personnelles d'un professeur (voir tableau 11, placé en annexe 2) ;
- Description du processus de suppression d'informations personnelles d'un professeur (voir tableau 12, placé en annexe 2) ;
- Description du processus de consultation d'informations personnelles d'un professeur (voir tableau 13, placé en annexe 2).

## Planification

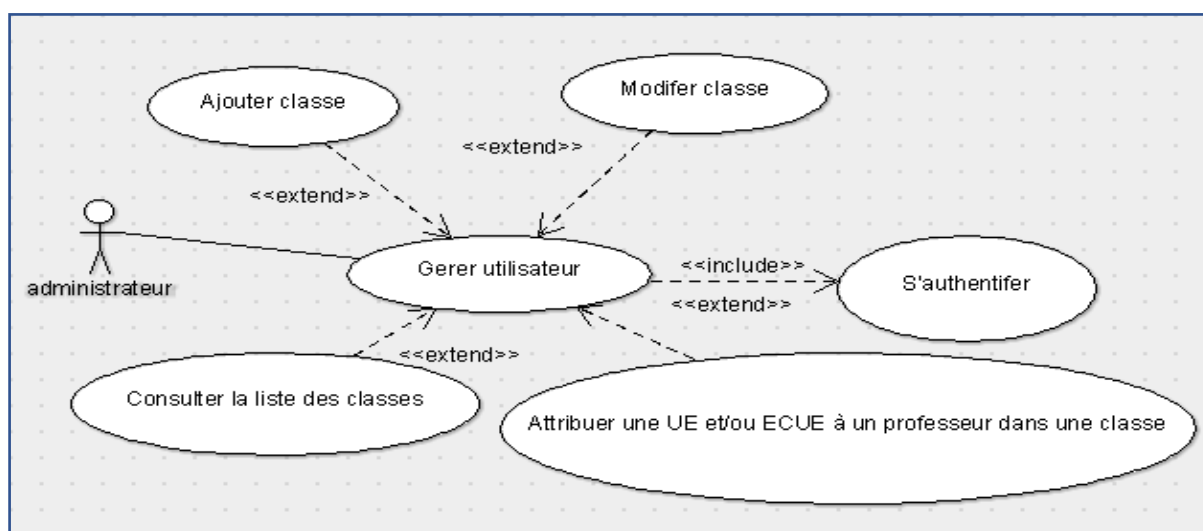


Figure 4 : Diagramme de cas d'utilisation pour la planification des emplois du temps des professeurs et des étudiants

Tableau 3 : Ajouter les informations d'une classe

<b>Acteur</b>	L'administrateur chargé de la planification des UE et ECUE
<b>Pré condition</b>	-S'authentifier
<b>Post condition</b>	L'ajout de la classe est réussi et enregistrer dans la base de données
<b>Description du scénario</b>	<p>- L'administrateur saisit les informations correspondant à l'enregistrement de la classe et confirme sa demande d'ajout.</p> <p>- Le système ajoute la classe à la base des données et envoie un message indiquant le succès de l'opération</p>
<b>Exception</b>	Si la classe existe le système affiche un message d'erreur.

- Description du processus de modification d'informations d'une classe (voir tableau 14, placé en annexe 2) ;
- Description du processus de consultation d'informations d'une classe (voir tableau 15, placé en annexe 2).

### Utilisateurs

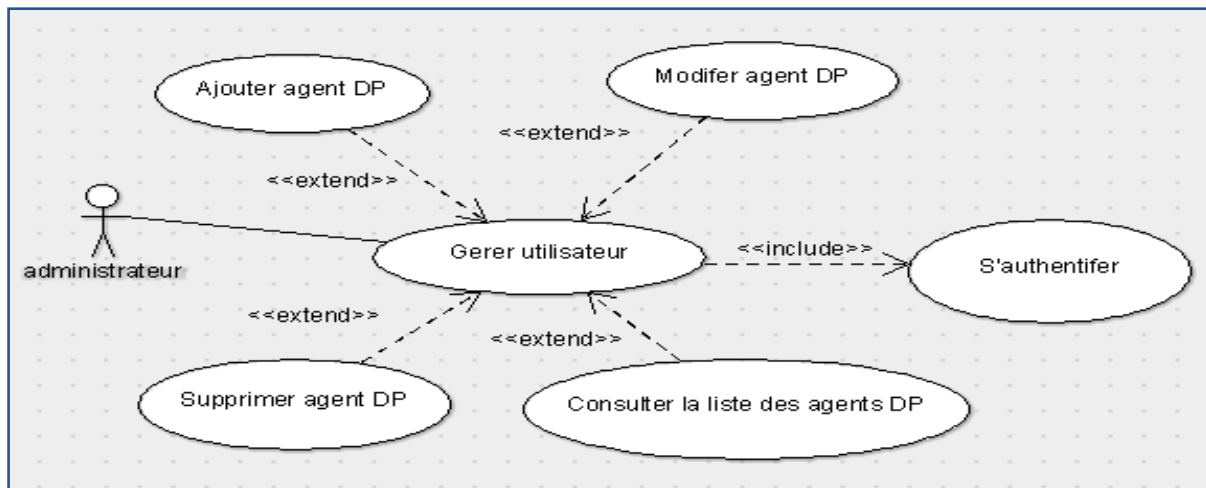


Figure 5 : Diagramme de cas d'utilisation pour gérer les utilisateurs.

Tableau 4 : Ajouter les informations personnelles d'un agent de la Direction Pédagogique.

<b>Acteur</b>	L'administrateur chargé de la gestion des utilisateurs
<b>Pré condition</b>	-S'authentifier
<b>Post condition</b>	L'ajout du compte est réussi et enregistrer dans la base de données
<b>Description du scénario</b>	- L'administrateur saisit les informations correspondant au nouveau compte et confirme sa demande d'ajout. - Le système ajoute le nouveau compte à la base de données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Si le compte existe le système affiche un message d'erreur

- Description du processus de modification d'informations d'un agent de la Direction Pédagogique (voir tableau 17, placé en annexe 2) ;
- Description du processus de suppression d'informations d'un agent de la Direction Pédagogique (voir tableau 18, placé en annexe 2) ;
- Description du processus de consultation d'informations d'un agent de la Direction Pédagogique (voir tableau 19, placé en annexe 2).

*d Diagramme de cas d'utilisation lié aux étudiants*

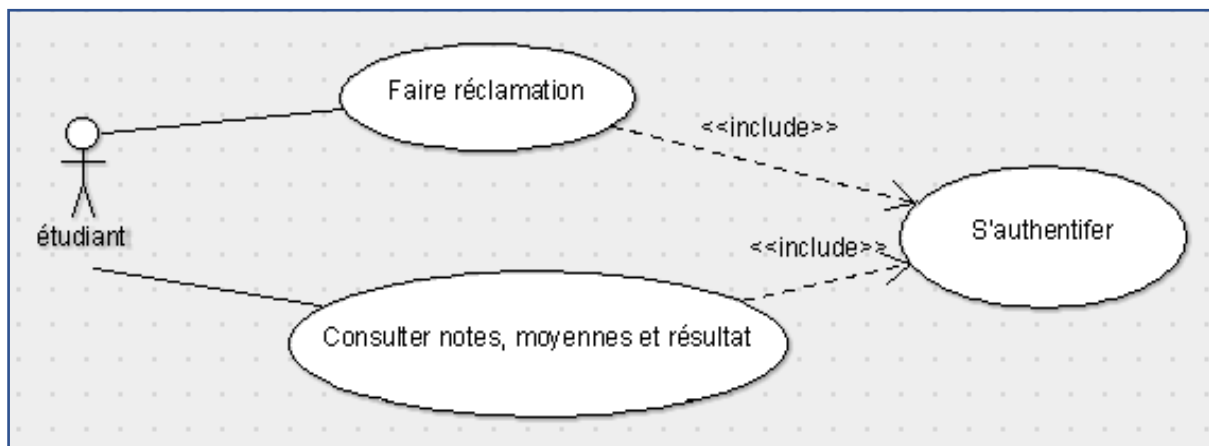


Figure 6 : Diagramme de cas d'utilisation pour les étudiants

Tableau 5 : Consulter et faire des réclamations par étudiant

	Acteur	Pré condition	Post condition	Description du scénario	Exception
<b>Consulter notes, moyennes et résultat</b>	Etudiant	-S'authentifier ; -Aller à l'onglet note ; -Entrer le secret partagé et connu de l'étudiant ; -Sélectionner l'UE ou l'ECUE ; -Liste des notes, moyenne consultée	-Les notes sont consultées  -Le résultat est consulté	- L'étudiant clique sur le bouton voir mes notes ou sur le bouton résultat  - Le système affiche les notes ou le résultat de l'étudiant	Si les notes ou les résultats ne sont pas disponibles le système affiche un message
<b>Faire réclamation</b>	Etudiant	-S'authentifier ; - Aller à l'onglet réclamation ; -Renseigner les infos relatives à la réclamation ; -Sélectionner l'UE ou l'ECUE; -Valider et envoyer au professeur chargé de l'UE/ECUE	- Le message de réclamation est envoyé au professeur concerné	- L'étudiant clique sur le bouton faire une réclamation et saisit les informations correspondant à la réclamation  - Le système envoie le message de réclamation au professeur concerné et envoie un message indiquant le succès de l'opération	Aucune

e Diagramme de cas d'utilisation lié aux professeurs

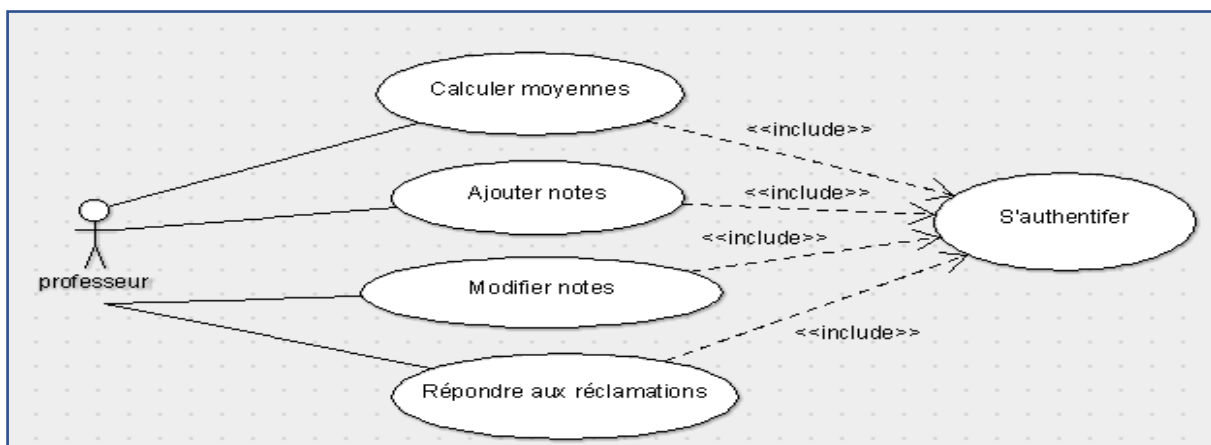


Figure 7 : Diagramme de cas d'utilisation pour les professeurs

Tableau 6 : Ajouter les notes d'étudiant

<b>Acteur</b>	Professeur
<b>Pré condition</b>	-S'authentifier -Sélectionner sa classe
<b>Post condition</b>	L'ajout de la note est réussi et enregistrer dans la base de données
<b>Description du scénario</b>	- Le professeur saisit les notes des étudiants, saisit un secret et confirme sa demande d'ajout. - Le système chiffre les notes, les ajoutent à la base de données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Si le professeur oublie de saisir la note d'un étudiant le système affiche un message d'erreur

*f Diagramme de cas d'utilisation lié aux agents de saisie*

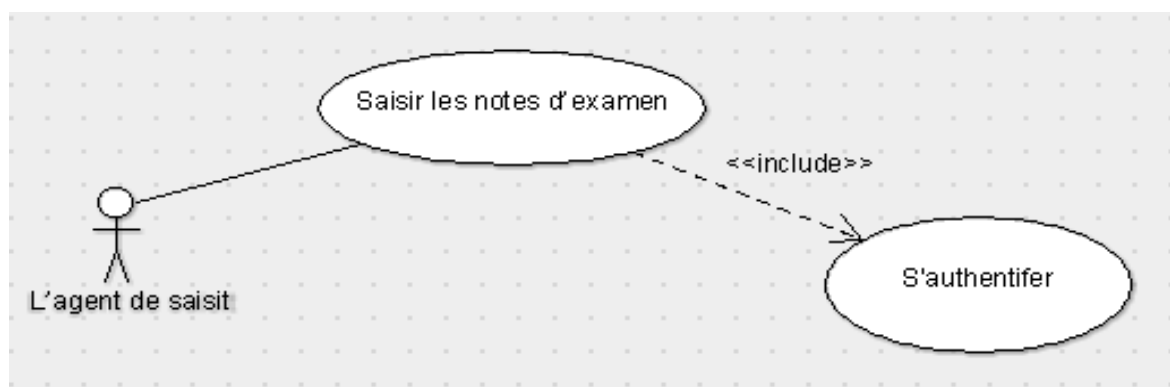


Figure 8 : Diagramme de cas d'utilisation pour l'agent de saisie.

*g Diagramme de cas d'utilisation lié aux agents de vérification*

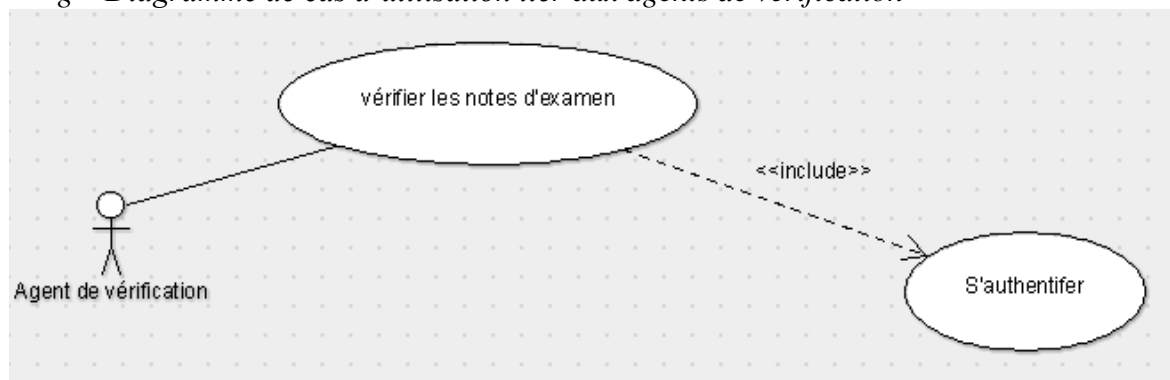


Figure 9 : Diagramme de cas d'utilisation lié aux agents de vérification.

## **h** Diagramme de cas d'utilisation lié au directeur pédagogique

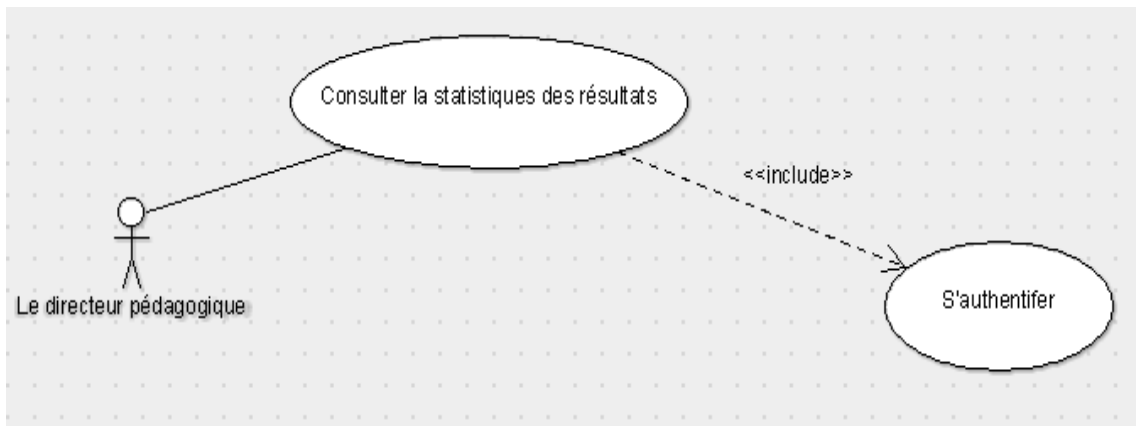


Figure 10 : Diagramme de cas d'utilisation lié au directeur pédagogique.

## 2. Diagramme des classes

### *a* Définition

Le diagramme de 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.

### *b* Composition

En général un diagramme de classe peut contenir les éléments suivants :

- ❖ **Les classes** : une classe représente la description formelle d'un ensemble d'objets ayant une sémantique et des caractéristiques communes. Elle est représentée en utilisant un rectangle divisé en trois sections. La section supérieure est le nom de la classe, la section centrale définit les propriétés de la classe alors que la section du bas énumère les méthodes de la classe ;
- ❖ **Les associations** : une association est une relation entre deux classes (association binaire) ou plus (association n-aire), qui décrit les connexions structurelles entre leurs instances. Une association indique donc que des liens peuvent exister entre des instances des classes associées ;



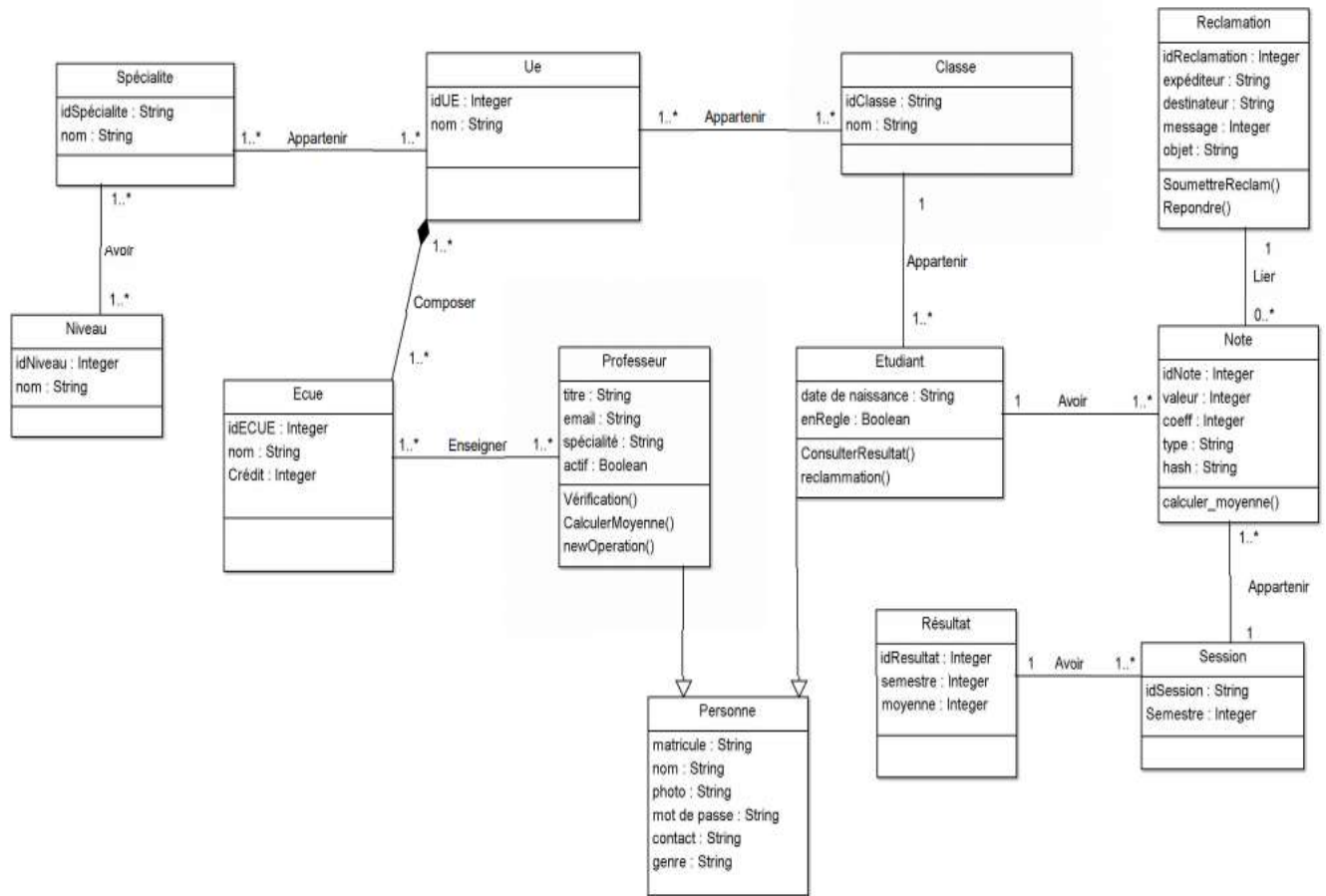


Figure 11 : Diagramme des classes de notre système.

- ❖ **Les attributs** : les attributs représentent les données encapsulées dans les objets des classes. Chacune de ces informations est définie par un nom, un type de données, une visibilité et peut être initialisé. Le nom de l'attribut doit être unique dans la classe.

### 3. Diagramme des séquences

#### a Définition

Un diagramme de séquences est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont. Les diagrammes de séquences sont organisés en fonction du temps qui s'écoule au fur et à mesure que nous parcourons la page. Les objets impliqués dans l'opération sont répertoriés de gauche à droite en fonction du moment où ils prennent part dans la séquence.

#### b Composition

Ce type des diagrammes est composé par les éléments suivants :

- ❖ **Les lignes de vie** : Une ligne verticale qui représente la séquence des événements, produite par un participant, pendant une interaction, alors que le temps progresse en bas de ligne ;
- ❖ **Les messages** : deux types de messages dans le diagramme de séquences, le premier est dit message synchrone utilisé pour représenter des appels de fonction ordinaires dans un programme, le deuxième est appelé message asynchrone, étant utilisé pour représenter la communication entre des threads distincts ou la création d'un nouveau thread ;
- ❖ **Les occurrences d'exécution** : représente la période d'exécution d'une opération ;
- ❖ **Les commentaires** : Un commentaire peut être joint à tout point sur une ligne de vie ;
- ❖ **Les itérations** : représente un message de réponse suite à une question de vérification.

Nous présentons ici les diagrammes de séquence de certains cas d'utilisation. Ceux qui sont importants dans la conception de notre système et qui demandent plus d'explications. Ce sont les cas d'utilisation de :

- Ajout de note (côté professeur) ;
- Consultation de note (côté étudiant) ;
- La réclamation : le faire par les étudiants et la réponse par les enseignants.

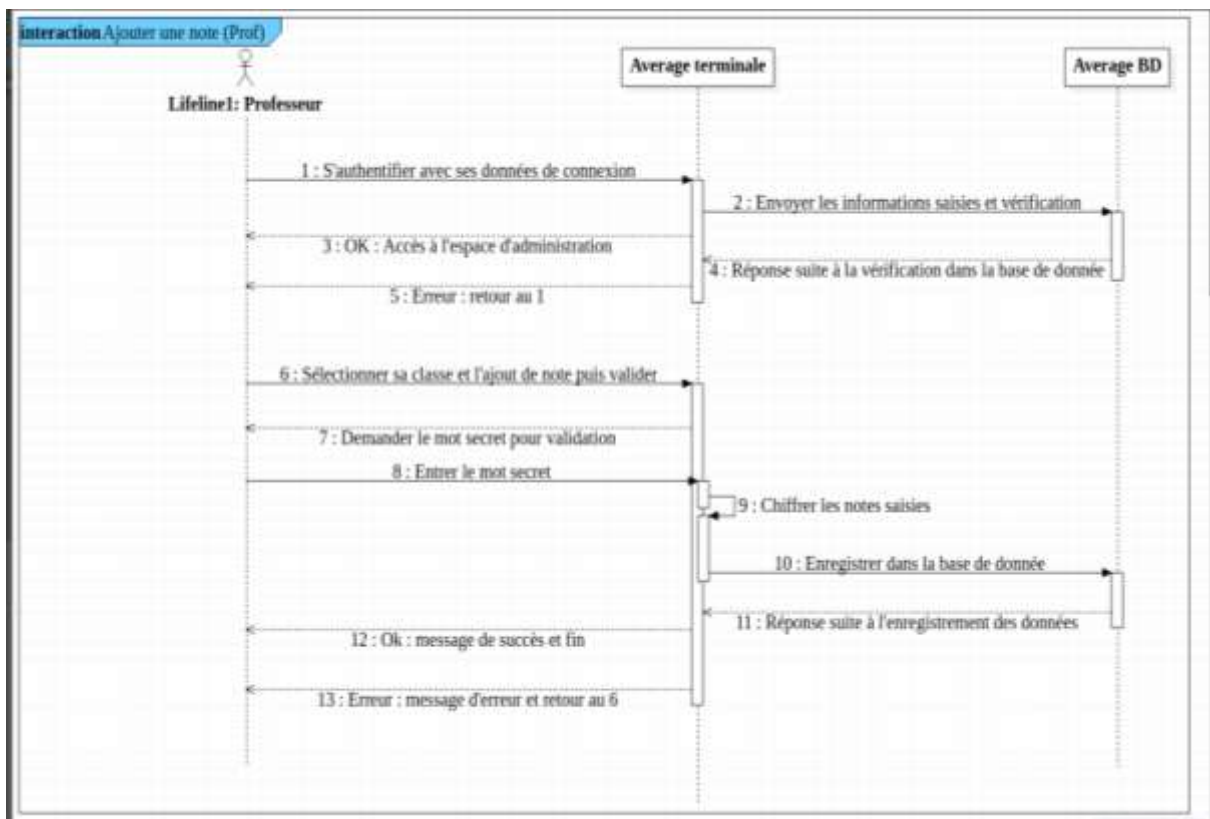


Figure 12 : Diagramme de séquence- Ajout de note par le professeur

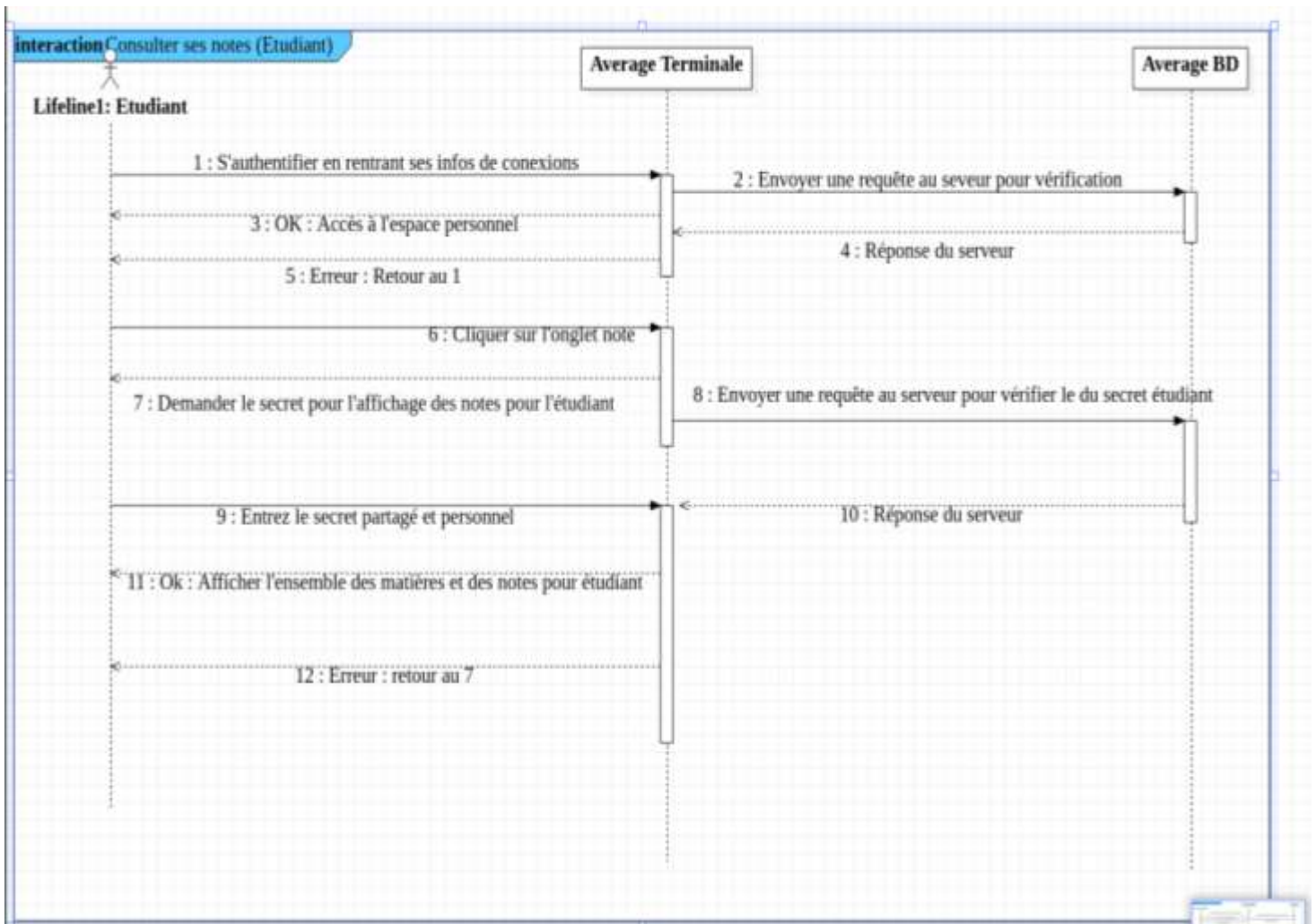


Figure 13 : Diagramme de séquence – Consultation de note par l'étudiant

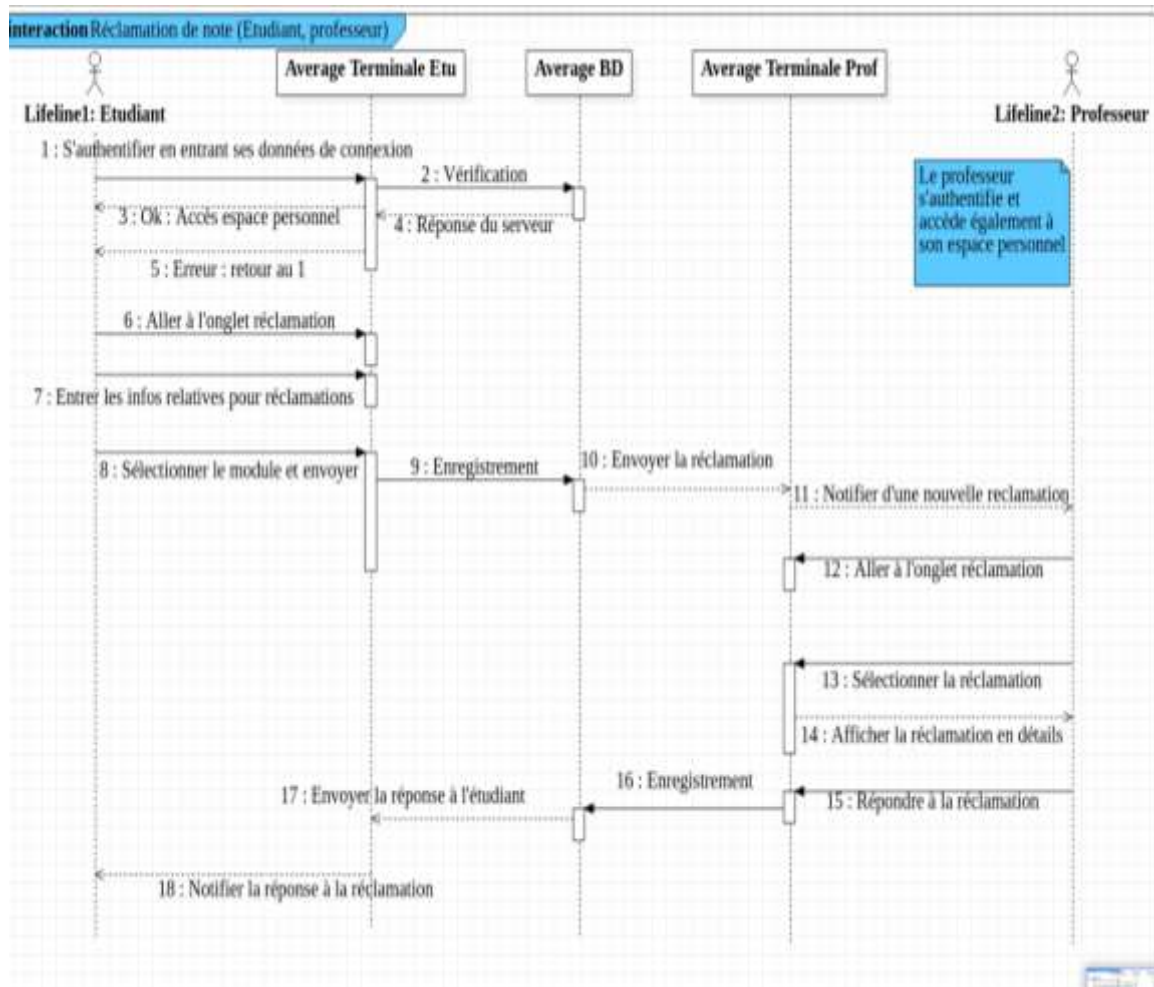


Figure 14 : Diagramme de séquence de réclamation de note et réponse à la réclamation

Dans ce chapitre, nous avons présenté les différents diagrammes définis par UML qui ont permis de bien comprendre les besoins du système à développer ainsi que les différentes interactions entre les objets participant à son fonctionnement, chose qui facilitera la phase d'implémentation et de codage.

## CHAPITRE V : PROCESSUS DE SECURITE

Ce présent chapitre explique nos différents besoins de sécurité pour ledit projet. Nous présentons ici la méthode de chiffrement utilisée et justifions le choix nous expliquons également les différentes procédures de chiffrement.

### **I. BESOINS NECESSAIRES DE SECURITE**

Pour garantir la sécurité de notre système, il faut assurer les besoins essentiels de sécurité que sont : la confidentialité, l'intégrité, la disponibilité et la non-répudiation.

- ❖ **La confidentialité** : Une information est dite confidentielle si elle est accessible uniquement aux entités autorisées à y avoir accès ;
- ❖ **L'intégrité** : Une information est dite intègre si l'on est capable de prouver avec quasi-certitude qu'elle n'a pas été modifiée, soit accidentellement, soit de manière malveillante ;
- ❖ **La disponibilité** : Un système d'information assure de la disponibilité s'il est capable de garantir un fonctionnement correct durant les périodes prévues d'utilisation ;
- ❖ **La non-répudiation** : assure que l'émetteur du message ne pourra pas nier avoir émis le message dans le futur.

Ces différents besoins prennent implicitement en compte le besoin **d'authenticité** et **d'autorisation**. Et pour les garantir, il faut trouver des techniques pour protéger les informations. L'étude de ces techniques est appelée **la cryptologie**, celle-ci comprend la cryptographie et la cryptanalyse. La cryptographie qui porte sur la conception des techniques de protection et la cryptanalyse qui consiste à analyser la sécurité des techniques cryptographiques.

Attardons-nous sur la cryptographie, car il s'agit pour nous de trouver une technique pour protéger les informations (données étudiantes, données enseignantes, les notes, etc.) de notre système.

### **II. TECHNIQUE DE CHRYPTAGE CHOISIE**

En cryptographie, il existe deux grands concepts de chiffrement. Chaque concept confère une technique sinon une méthode de protection des données. Ce sont entre autres : Le chiffrement symétrique et le chiffrement asymétrique.

#### **1. Chiffrement symétrique**

Le chiffrement symétrique, aussi appelé crypto système à clé privée ou conventionnels, il a recours à une même clé secrète unique, pour le chiffrement et le déchiffrement. La clé étant connue que des seuls correspondants.

## 2. Chiffrement asymétrique

Le chiffrement asymétrique utilise deux clés : une clé privée (ou secrète) et une clé publique qui sont liées mathématiquement afin qu'un message chiffré avec l'une puisse être déchiffré avec l'autre. Les clés publique et privée servent respectivement à chiffrer et déchiffrer.

## 3. Comparaison

Tableau 7 : Comparaison des techniques cryptographiques

XXXXXXXXXXXXXXXX	Avantages	Inconvénients
<b>Symétrique</b>	<ul style="list-style-type: none"><li>- Extrêmement rapide</li><li>- Utilise peu de ressources systèmes</li><li>- L'impossibilité de décrypter le message dans le cas de son interception par une personne non autorisée</li></ul>	<ul style="list-style-type: none"><li>- N'assure que la confidentialité des données</li><li>- Une clé symétrique correspond à un échange entre 2 personnes, pour communiquer avec d'autres personnes il faudra une autre clé symétrique.</li></ul>
<b>Asymétrique</b>	<ul style="list-style-type: none"><li>- L'élimination de la problématique de la transmission de clé</li><li>- La possibilité d'utiliser la signature électronique</li><li>- L'impossibilité de décrypter le message dans le cas de son interception par une personne non autorisée</li></ul>	<ul style="list-style-type: none"><li>-Longueur des clés</li><li>-Vitesse d'encryptions</li><li>-Gestion des clés</li><li>-Key Validation</li></ul>

Nous avons opté pour le **chiffrement symétrique** en raison de sa rapidité, de sa facilité d'implémentation dans notre système. Nous savons bien que dans le chiffrement symétrique, le problème majeur est l'utilisation d'une clé unique pour chiffrement et déchiffrement de l'information. Voilà pourquoi nous devons :

- Communiquer la clé de manière sûre à la personne avec laquelle on souhaite dialoguer ;

- Il est nécessaire de garantir la confidentialité de cette clé. Les échanges qui suivront reposent sur celle-ci. En d'autres termes, si une tierce personne accède à la clé, elle pourra lire, modifier, altérer tous les échanges qui s'effectueront entre les 2 protagonistes de départ.

### III. PROCESSUS DE CHIFFREMENT

Après la saisie des notes par le professeur le système définit la concaténation du matricule de l'étudiant à sa note et à la date de saisie ensuite le professeur entre sa clé secrète pour chiffrer le tout. Le système enregistre dans la base de données la note et le chiffré associés à la note. L'étudiant pourra ainsi consulter sa note.

Les notes des examens sont disponibles après la délibération. Cependant leur saisie dans notre base de données se fait selon un procédé de chiffrement symétrique. Les notes d'examen sont concaténées au code UE ou de l'ECUE, le tout est chiffré avec la clé secrète des agents de la direction pédagogique et enregistré dans la base de données. Par conséquent toutes notes entrées dans notre système ne sauraient être modifiées depuis la base de données. Par ailleurs, tout est horodaté et tout utilisateur opérant sur notre système est identifié. Ceci nous permet d'assurer la **non-répudiation** des différentes données ajoutées.



## **TROISIEME PARTIE : REALISATION ET MISE EN PLACE DE LA SOLUTION**

## **CHAPITRE VI : ENVIRONNEMENT DE TRAVAIL**

Ce chapitre constitue le dernier volet de cette partie du rapport. Il a pour objet d'exposer le travail réalisé. Nous présenterons donc l'environnement matériel du projet ainsi que les outils de développement utilisés. Nous montrerons ensuite les différents aperçus d'écran illustrant les fonctionnalités de l'application. Ce chapitre sera clôturé par le chronogramme d'avancement du stage.

### **I. RESSOURCES MATERIELLES ET LOGICIELS**

#### **1. Ressources matérielles**

- ❖ Ordinateur
- ❖ Smartphones

#### **2. Ressources logicielles**

Android Studio un environnement de développement pour développer des applications Android. Son avantage est qu'il simplifie la programmation avec un certain nombre de raccourcis et d'aide à la programmation. Il nous a permis de mettre en place notre application mobile. Il est associé à JAVA et XML.

XAMPP est un ensemble de logiciels permettant de mettre en place un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Cette plateforme de développement web permet de faire fonctionner localement des scripts PHP.

Des éditeurs de texte ouvrant l'accès à un grand nombre de langage. Ils ont été utilisés ici pour coder le code PHP, HTML et JS.

ArgoUML un logiciel libre de création de diagrammes UML. Programmé en Java. Il est multilingue, supporte la génération de code et l'ingénierie inverse. ArgoUML supporte treize (13) types de diagrammes. Nous l'avons utilisé pour réaliser nos diagrammes de classe, cas d'utilisation, et de séquences.

### **II. TECHNOLOGIE UTILISER**

Nous avons eu recours, pour le développement de notre application au langage de programmation JAVA, XML, SQL, HTML, JS et PHP.

- ❖ **Extensible Markup Language (XML)**, en français, est un langage informatique de balisage générique. Il est à l'origine du design de notre application mobile ;
- ❖ **Java** un langage programmation orienté objet créé par James Gosling et Patrick Naughton. Il permet une intégration facile des ressources du système Android, c'est un langage performant en termes fonctionnalités, qualité et résolution. Son choix parmi les outils de langage a donc été fait vu ses différents atouts ;

- ❖ **PHP** signifie **HyperText Préprocesseur**, plus connu sous son sigle PHP, est un langage de scripts multi plateformes, orienté-objet très répandu. Il a été conçu pour permettre la création des applications dynamiques. Il est aussi utilisé pour l'interaction avec la base de données. Dans la réalisation du projet Il nous a permis d'interagir avec la base de données à travers SQL ;
- ❖ **CSS** (Cascading Style Sheets) c'est un langage informatique qui sert à décrire la présentation des documents HTML et XML. CSS est couramment utilisé dans la conception de sites web et bien pris en charge par les navigateurs web. Il à été utilisée pour le design de notre plateforme ;
- ❖ **MYSQL** : est un serveur de base de données qui permet de faire fonctionner localement (sans se connecter à un serveur externe). MySQL est un logiciel libre développé sous licence GNU General Public License. Il a été conçu et est maintenu par une société suédoise **MySQL AB** fondée par deux Suédois et un Finlandais. MySQL fonctionne sur beaucoup de plates-formes différentes, incluant Linux, Mac OS X, Windows et autres.

Les systèmes de gestion de base des données que nous avons analysés ont chacun leur force et leur faiblesse. Si la décision n'est pas déjà prise, ou même si elle l'est, chacune sera prise en considération. C'est ainsi que nous avons conclu que les systèmes de gestion de base des données en accès libre (freeware), telles MySQL et PostgreSQL, sont de très bonnes solutions d'entrée de gamme. Le coût en est très abordable, ils sont gratuits ; ces systèmes de gestion de base des données sont faciles à installer et à maintenir.

Ces deux Systèmes de gestion de base des données reposent sur le modèle du développement de logiciel le plus récent, qui est l'Open Source. Il a le grand avantage de reposer sur une communauté de développement. Il y a normalement un ou des responsables du code en nombre restreint, mais tout l'Internet comme équipe de développement potentielle. Pour sortir du choix embarrassant de MySQL ou de PostgreSQL, nous avons constaté que MySQL est plus supporté par des plates-formes de conception qui sont à notre portée telle que : WAMP, LAMP et XAMP. Ainsi donc, nous avons pu choisir MySQL en fin de compte pour le développement et la gestion de notre base des données. Les bases de données MySQL sont accessibles en utilisant les langages de programmation entre autres le C, C++, Java, Perl, PHP, Python et une API spécifique est disponible pour chacun d'entre eux.

## CHAIPTRE VII : REALISATION ET DISCUSSION

### **I. PAGE D'AUTHENTIFICATION**



Figure 15 : Page d'authentification

L'application se compose de plusieurs interfaces qui guident l'administrateur, l'agent de la DP, le professeur vers les différentes fonctions de l'application après authentification.

Pour s'authentifier, le professeur et l'agent de la DP devront obligatoirement saisir leurs adresses électroniques personnel et leurs mots de passe

### **II. ESPACE PROFESSEUR**

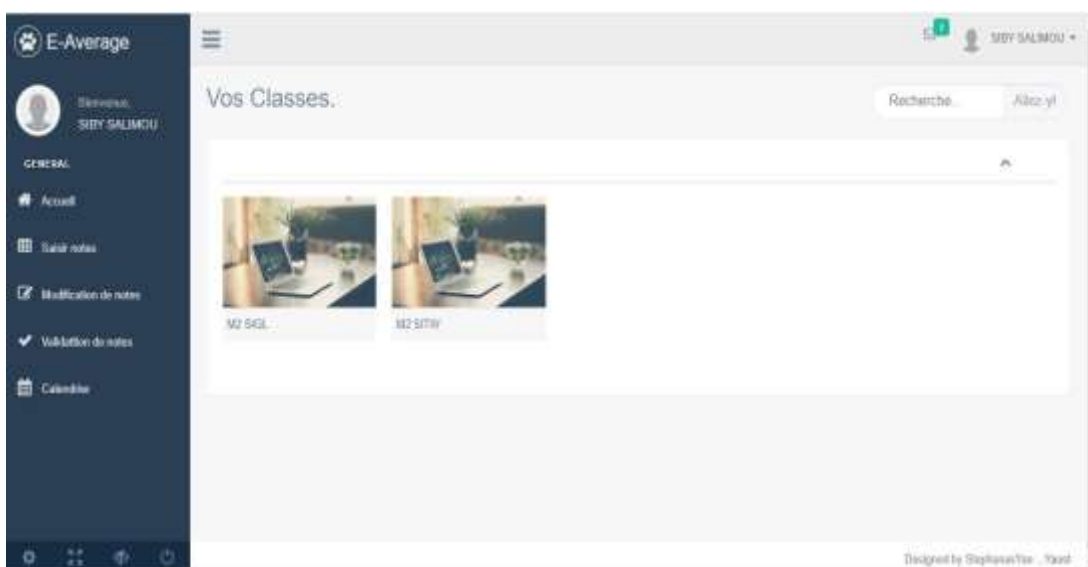


Figure 16 : Page des classes du professeur

Cette interface permet au professeur de voir toutes les classes qu'il enseigne à l'ESATIC. En cliquant sur le nom de la classe, qui est un lien, il sera redirigé vers la liste de ses matières dans cette classe.



Figure 17 : Page des matières du professeur dans une classe.

Le professeur a accès à toutes ses matières dans une classe donnée. En cliquant sur le nom de la matière, il sera redirigé vers la liste des étudiants de la classe pour administrer les notes et E-Average se chargera de calculer systématiquement les moyennes de chaque étudiant en fonction de la formule de calcul fourni par le professeur.

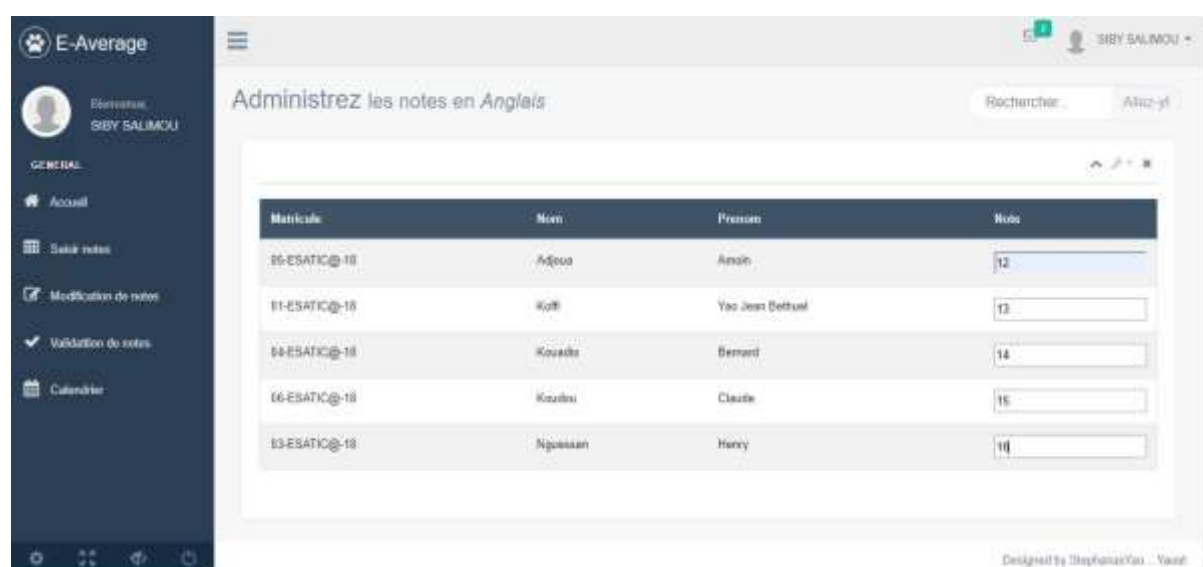


Figure 18 : Page d'administration des notes dans une classe.

Interface de saisir de notes des étudiants. Le professeur devra préciser le coefficient de la note, le type de la note (Devoir, Interrogation...) et le semestre. Toute case vide sera considérée comme la note zéro (0) attribué à l'étudiant.

NB : Le professeur aura toujours la possibilité de modifier les notes qu'il a saisi avant l'étape de validation. Une fois l'étape de validation passée, il ne pourra plus modifier les notes.

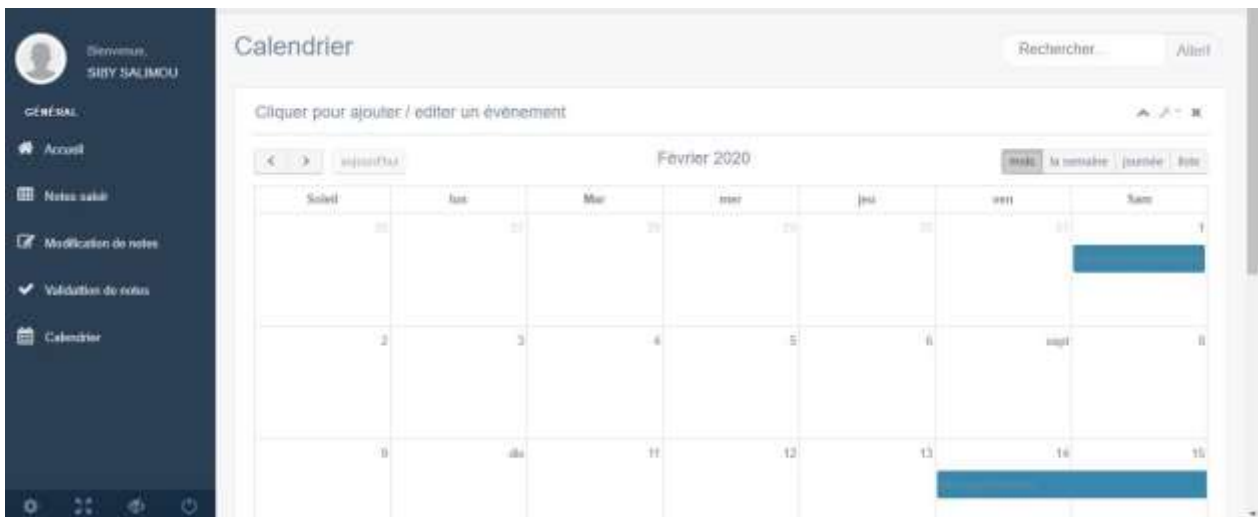


Figure 19 : Page de calendrier du professeur.

Cette interface contient un calendrier qui permettra au professeur de planifier toutes ses interrogations ou devoir. Il a aussi la possibilité de planifier d'autres évènements.

### III. ESPACE ETUDIANTS

L'étudiant aura à son actif un espace mobile

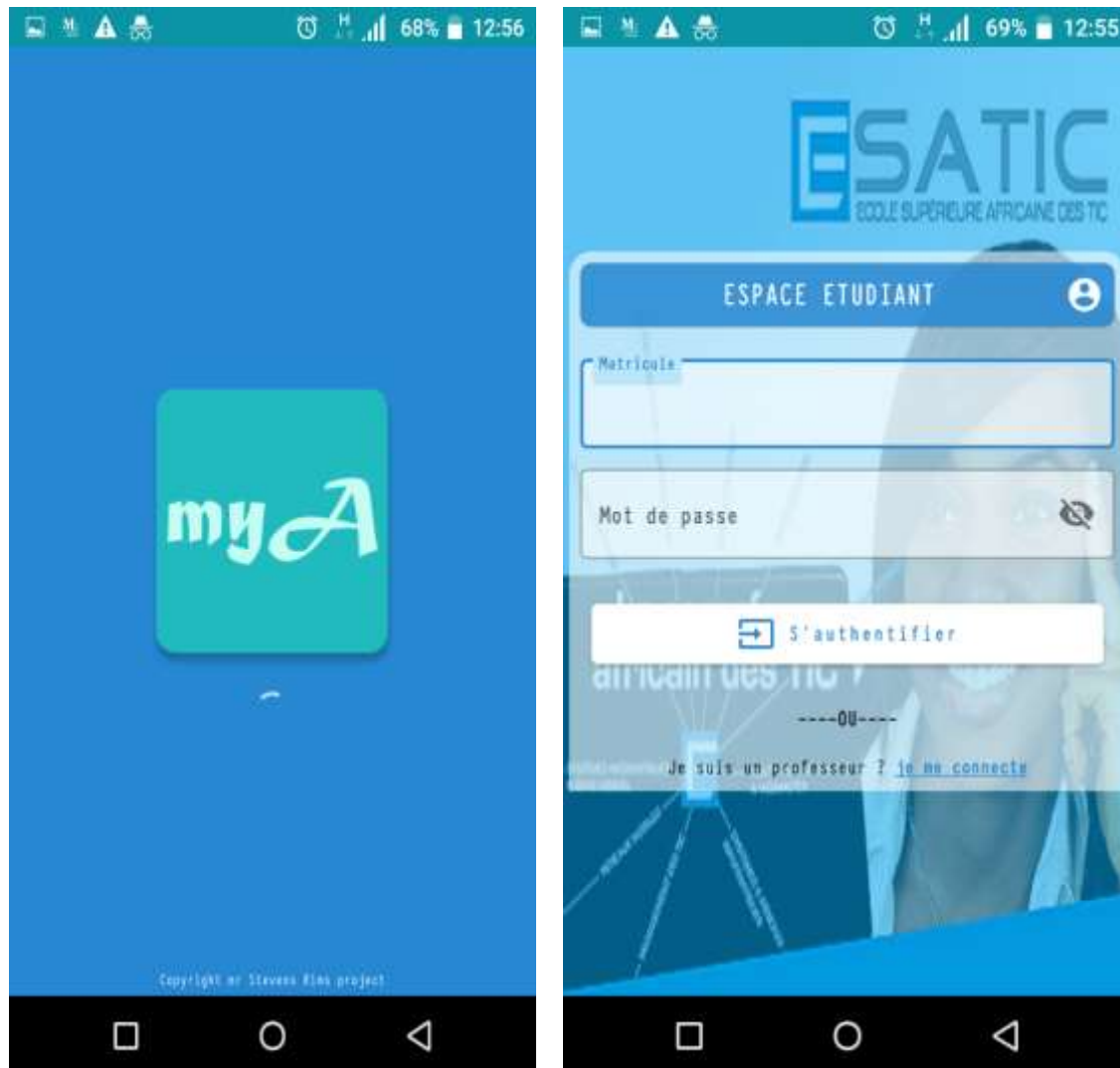


Figure 20 : Page d'authentification de l'étudiant

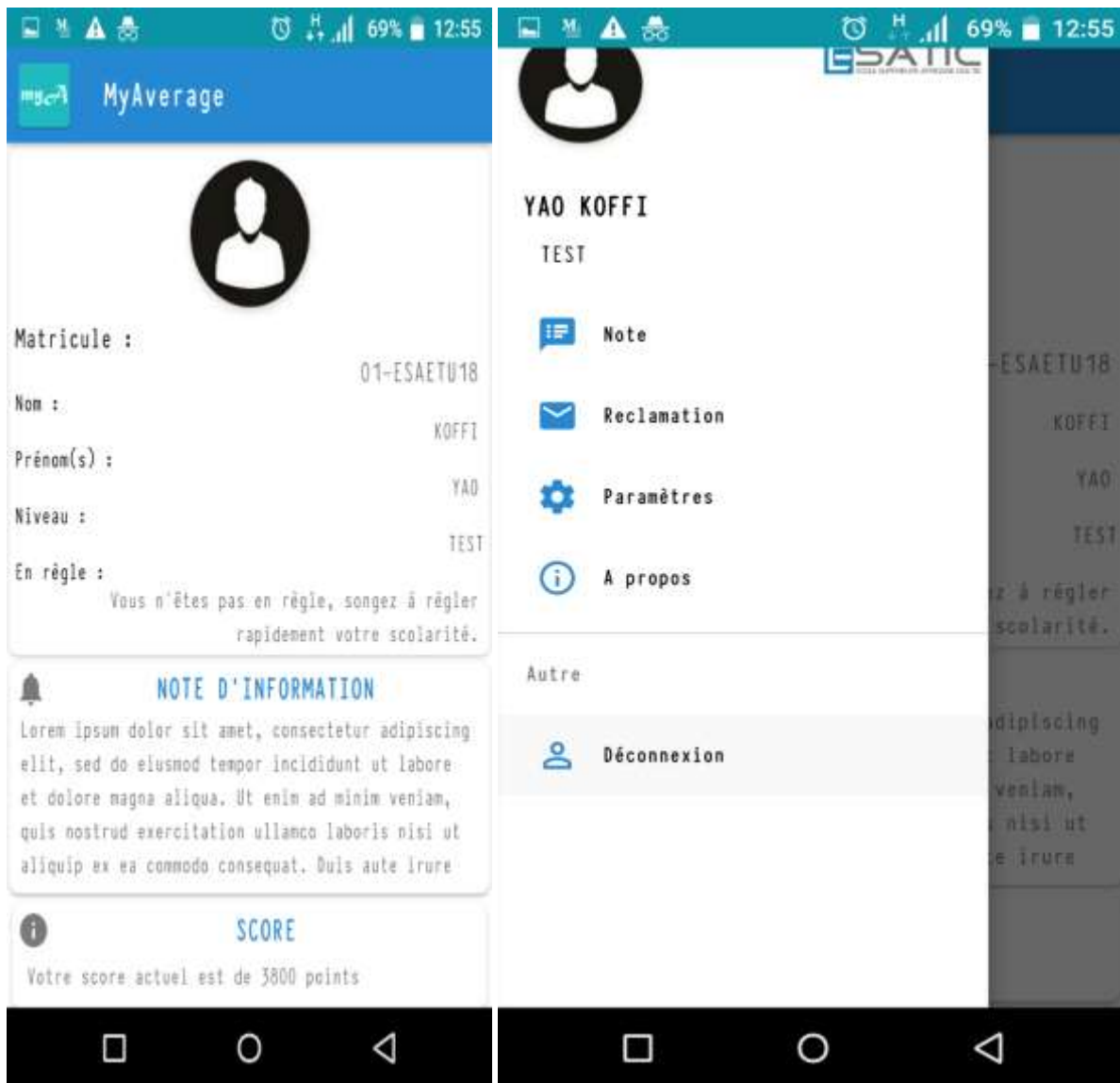


Figure 21 : Page du profil de l'étudiant



## **CONCLUSION**

---

Cette étude avait pour but de vérifier la possibilité de mettre en place une application web et mobile pour l'édition d'un relevé de notes numériques pour l'ESATIC à l'aide du langage de modélisation UML, du langage PHP et du langage JAVA. Pour ce faire, il a fallu passer en revue tout le processus et les bonnes pratiques qui sont lié à une telle mise en place. Certes, nous avons rencontré quelques difficultés liées à la mise en place, mais cela nous a permis d'améliorer nos connaissances en informatique, d'avoir un esprit critique et d'apprendre à gérer plusieurs situations.

Cette application a permis en premier lieu de répondre aux besoins des utilisateurs par la résolution des problèmes de la gestion des notes. En deuxième lieu, elle a permis également, d'orienter le personnel vers une application unique.

Enfin, pour répondre à la problématique qui s'est imposée à nous, il ressort que les langages PHP pour le web et JAVA pour le mobile sont adaptés pour tout système de ce type. Toutefois, une bonne connaissance de ces langages est requise afin de mener à bien un projet.

Toute réalisation humaine n'étant pas parfaite, l'application implémentée à partir de cette analyse peut présenter quelques insuffisances.

## **BIBLIOGRAPHIE / WEBOGRAPHIE**

---

P. ROQUES. – UML 2 : modéliser une application web. N° 12389, 4e édition, 2008, 246 pages.

X. BLANC, I. MOUNIER. – UML 2 pour les développeurs. N° 12029, 2006, 202 pages.

P. ROQUES, F. VALLÉE. – UML 2 en action. N° 11213, 2e édition, 2002, 400 pages.

[2] <http://apcpedagogie.com/les-sites-web-statiques/>

[3] <https://www.ideematic.com/dictionnaire-digital/application-web/>, consulté le 20/02/2020.

[4] David LACERTE. Applications web et mobiles. TAURON, 25 septembre 2013.

[5] <https://www.syloe.com/glossaire/serveur-informatique/>, consulté le 18/02/2020.

[6] <https://www.tinycluster.com/blog/quest-ce-quest-un-hebergement-web/>, consulté le 17/02/2020.

[7] [https://www.fdc.com.coop/export/sites/default/fr/federations/les-scopcommunication/\\_media/docs-fede-communication/Etudes/963-GuideConcevoirApplicationMobile.pdf](https://www.fdc.com.coop/export/sites/default/fr/federations/les-scopcommunication/_media/docs-fede-communication/Etudes/963-GuideConcevoirApplicationMobile.pdf), consulté le 18/02/2020.

[8] <http://www.thesaurus.gouv.qc.ca/tag/accueil.do>, consulté le 17/02/2020.

[9] <https://blog.devensys.com/chiffrement-symetrique/>, consulté le 15/02/2020

[9] <http://www.ordinateur.cc/r%C3%A9seaux/s%C3%A9curit%C3%A9-desr%C3%A9seaux/75792.html>, consulté le 13/02/2020



**Annexe 2 : Liste complète des tableaux de description du diagramme de cas d'utilisation lié aux administrateurs.**

Tableau 8 : Modifier les informations personnelles d'un étudiant

<b>Acteur</b>	L'administrateur chargé de la gestion des étudiants
<b>Pré condition</b>	- S'authentifier - Sélectionner l'étudiant
<b>Post condition</b>	La modification de l'étudiant déjà inscrit est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	- L'administrateur saisit les informations correspondant à l'étudiant déjà inscrit et confirme sa demande de modification. - Le système modifie l'étudiant dans la base de données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Aucune

Tableau 9 : Supprimer les informations personnelles d'un étudiant.

<b>Acteur</b>	L'administrateur chargé de la gestion des étudiants
<b>Pré condition</b>	- S'authentifier - Sélectionner l'étudiant
<b>Post condition</b>	La suppression de l'étudiant qui existe est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	- L'administrateur supprime l'étudiant déjà enregistré et confirme la suppression. - Le système supprime l'étudiant de la base de données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Aucune

Tableau 10 : Consulter les informations personnelles d'un étudiant.

<b>Acteur</b>	L'administrateur chargé de la gestion des étudiants
<b>Pré condition</b>	- S'authentifier
<b>Post condition</b>	La consultation de la liste des étudiants est réussie
<b>Description du scénario</b>	- L'administrateur clique sur gestion des étudiants. - Le système affiche toute la liste des étudiants de la base des données.
<b>Exception</b>	Aucune

Tableau 11 : Modifier les informations personnelles d'un professeur.

<b>Acteur</b>	L'administrateur chargé de la gestion des professeurs
<b>Pré condition</b>	<ul style="list-style-type: none"> <li>- S'authentifier</li> <li>- Sélectionner le professeur</li> </ul>
<b>Post condition</b>	La modification du professeur déjà enregistré est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	<ul style="list-style-type: none"> <li>- L'administrateur saisit les informations correspondant au professeur déjà enregistré et confirme sa demande de modification.</li> <li>- Le système modifie le professeur dans la base de données et envoie un message indiquant le succès de l'opération</li> </ul>
<b>Exception</b>	Aucune

Tableau 12 : Supprimer les informations personnelles d'un professeur.

<b>Acteur</b>	L'administrateur chargé de la gestion des professeurs
<b>Pré condition</b>	<ul style="list-style-type: none"> <li>- S'authentifier</li> <li>- Sélectionner le professeur</li> </ul>
<b>Post condition</b>	La suppression du professeur qui existe est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	<ul style="list-style-type: none"> <li>- L'administrateur supprime le professeur déjà enregistré et confirme la suppression.</li> <li>- Le système supprime le professeur de la base de données et envoie un message indiquant le succès de l'opération</li> </ul>
<b>Exception</b>	Aucune

Tableau 13 : Consulter les informations personnelles d'un professeur

<b>Acteur</b>	L'administrateur chargé de la gestion des professeurs
<b>Pré condition</b>	-S'authentifier
<b>Post condition</b>	La consultation de la liste des professeurs est réussie
<b>Description du scénario</b>	- L'administrateur clique sur gestion des étudiants. - Le système affiche toute la liste des étudiants de la base des données.
<b>Exception</b>	Aucune

Tableau 14 : Modifier les informations d'une classe.

<b>Acteur</b>	L'administrateur chargé de la planification des UE et ECUE
<b>Pré condition</b>	- S'authentifier - Sélectionner la classe
<b>Post condition</b>	La modification de la classe déjà enregistré est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	- L'administrateur saisit les informations correspondant à la classe déjà enregistré et confirme sa demande de modification.  - Le système modifie la classe dans la base de données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Aucune

Tableau 15 : Consulter les informations d'une classe.

<b>Acteur</b>	L'administrateur chargé de la planification des UE et ECUE
<b>Pré condition</b>	- S'authentifier - Sélectionner la classe
<b>Post condition</b>	La consultation des classes est réussie
<b>Description du scénario</b>	- L'administrateur clique sur consulter les classes - Le système affiche la liste de toute les classes
<b>Exception</b>	Aucune

Tableau 16 : Attribuer une UE et ECUE dans une classe.

<b>Acteur</b>	L'administrateur chargé de la planification des UE et ECUE
<b>Pré condition</b>	
<b>Post condition</b>	L'attribution est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	<ul style="list-style-type: none"> <li>- L'administrateur clique sur une classe et saisit les informations correspondant à l'attribution</li> <li>- Le système enregistre dans la base de données et envoie un message indiquant le succès de l'opération</li> </ul>
<b>Exception</b>	Aucune

Tableau 17 : Modifier les informations personnelles d'un agent de la Direction Pédagogique.

<b>Acteur</b>	L'administrateur chargé de la gestion des utilisateurs
<b>Pré condition</b>	<ul style="list-style-type: none"> <li>-S'authentifier</li> <li>-Sélectionner le compte</li> </ul>
<b>Post condition</b>	La modification du compte existant est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	<ul style="list-style-type: none"> <li>- L'administrateur saisit les informations correspondant au compte existant et confirme sa demande de modification.</li> <li>- Le système modifie le compte dans la base de données et envoie un message indiquant le succès de l'opération</li> </ul>
<b>Exception</b>	Aucune

Tableau 18 : Supprimer les informations personnelles d'un agent de la Direction Pédagogique

<b>Acteur</b>	L'administrateur chargé de la gestion des utilisateurs
<b>Pré condition</b>	- S'authentifier -Sélectionner le compte
<b>Post condition</b>	La suppression du compte existant est réussie et enregistrer dans la base de données
<b>Description du scénario</b>	- L'administrateur supprime le compte déjà enregistré et confirme la suppression. - Le système supprime le compte de la base de données et envoie un message indiquant le succès de l'opération
<b>Exception</b>	Aucune

Tableau 19 : Consulter les informations personnelles d'un agent de la Direction Pédagogique

<b>Acteur</b>	L'administrateur chargé de la gestion des utilisateurs
<b>Pré condition</b>	-S'authentifier
<b>Post condition</b>	La consultation de la liste des comptes est réussie
<b>Description du scénario</b>	-L'administrateur clique sur le bouton direction pédagogique.  -Le système affiche toute la liste des comptes de la base des données.
<b>Exception</b>	Aucune



# TABLE DES MATIERES

---

DEDICACE.....	I
REMERCIEMENTS .....	II
SOMMAIRE .....	III
LISTE DES TABLEAUX.....	IV
LISTE DES FIGURES .....	V
INTRODUCTION.....	1
PREMIERE PARTIE : GENERALITES .....	3
CHAPITRE I : PRESENTATION DU PROJET .....	4
I. DEFINITION DES NOTIONS CLES.....	4
1. Site web .....	4
2. Application web.....	4
3. Le serveur web.....	5
4. Hébergement d'une application.....	5
5. Application mobile .....	5
II. RELEVÉ DE NOTES .....	6
1. Relevé de notes numérique.....	6
2. Relevé de notes officiel .....	6
CHAPITRE II : ETUDE DE L'EXISTANT .....	7
I. FONCTIONNEMENT DU SYSTEME D'INFORMATION ACTUEL .....	7
II. LIMITES DE L'EXISTANT .....	8
III. PRESENTATION DU CAHIER DE CHARGES .....	9
1. Contexte général .....	9
2. Objectifs.....	9
<i>a Objectif général .....</i>	9
<i>b Objectifs spécifiques .....</i>	9
3. Résultat attendu .....	9
DEUXIEME PARTIE : ANALYSE ET CONCEPTION .....	11
CHAPITRE III : ANALYSE DES BESOINS.....	12
I. SPECIFICATION DES BESOINS FONCTIONNELS .....	12
II. SPECIFICATION DES BESOINS NON FONCTIONNELS .....	12
III. ARCHITECTURE DU SYSTEME .....	13
CHAPITRE IV : CONCEPTION .....	16
I. METHODE DE CONCEPTION .....	16

II. IDENTIFICATION DES ACTEURS .....	17
1. Les administrateurs .....	17
2. Le directeur pédagogique .....	18
3. Les agents de la direction pédagogique .....	18
4. Les professeurs .....	18
5. Les étudiants .....	18
II. DIAGRAMMES.....	19
1. Diagramme de cas d'utilisation .....	19
<i>a Définition .....</i>	<i>19</i>
<i>b Composition .....</i>	<i>19</i>
<i>c Diagramme de cas d'utilisation lier aux administrateurs .....</i>	<i>19</i>
<i>d Diagramme de cas d'utilisation lier aux étudiants .....</i>	<i>23</i>
<i>e Diagramme de cas d'utilisation lier aux professeurs .....</i>	<i>24</i>
<i>f Diagramme de cas d'utilisation lier aux agents de saisit .....</i>	<i>25</i>
<i>g Diagramme de cas d'utilisation lier aux agents de vérification .....</i>	<i>25</i>
<i>h Diagramme de cas d'utilisation lier au directeur pédagogique .....</i>	<i>26</i>
2. Diagramme des classes .....	26
<i>a Définition .....</i>	<i>26</i>
<i>b Composition .....</i>	<i>26</i>
3. Diagramme des séquences.....	27
<i>a Définition .....</i>	<i>27</i>
<i>b Composition .....</i>	<i>27</i>
CHAPITRE V : PROCESSUS DE SECURITE.....	32
I. BESOINS NECESSAIRES DE SECURITE.....	32
II. TECHNIQUE DE CHRYPTAGE CHOISIE.....	32
1. Chiffrement symétrique .....	32
2. Chiffrement asymétrique .....	33
3. Comparaison .....	33
III. PROCESSUS DE CHIFREMENT .....	34
TROISIEME PARTIE : REALISATION ET MISE EN PLACE DE LA SOLUTION .....	35
CHAPITRE VI : ENVIRONNEMENT DE TRAVAIL.....	36
I. RESSOURCES MATERIELLES ET LOGICIELS .....	36
1. Ressources matérielles.....	36
2. Ressources logicielles.....	36
II. TECHNOLOGIE UTILISER .....	36

CHAIPTRE VII : REALISATION ET DISCUSSION .....	38
I. PAGE D'AUTHENTIFICATION.....	38
II. ESPACE PROFESSEUR .....	38
III. ESPACE ETUDIANTS .....	41
CONCLUSION .....	43
BIBLIOGRAPHIE / WEBOGRAPHIE .....	44
ANNEXES .....	45
TABLE DES MATIERES .....	51
RESUME.....	44
ABSTRACT .....	44

## RESUME

---

Ce mémoire traite des étapes à respecter dans la conception d'une plateforme web pour les professeurs et mobile pour l'étudiant pour l'édition d'un relevé de notes numériques et de son intégration dans le système informatique actuel de l'ESATIC. La mise en place de cette application s'est faite à l'aide des langages PHP et JAVA. Cela suppose de disposer, pour chacune des librairies et bibliothèques de ces langages, d'un minimum de connaissances sur leurs contenus et fonctionnement respectifs. Une première étape du processus fut le choix de la méthode de développement, un choix qui s'est porté sur la méthode SCRUM des méthodes AGILE. Celle-ci a permis, dans un premier temps de recenser et spécifier les besoins et dans un second temps de dégager les priorités jusqu'à la fin du projet. Une seconde étape, inspirée des bonnes étapes de conception, à l'aide du langage de modélisation UML, permettant la réalisation avec les langages PHP et JAVA.

## ABSTRACT

---

This thesis deals with the steps to follow in the design of a web platform for teachers and mobile for the student for editing a digital transcript and its integration into the current IT system of ESATIC. The implementation of this application was done using the PHP and JAVA languages. This presupposes having, for each of the libraries and libraries of these languages, a minimum of knowledge about their content and their respective functioning. A first step in the process was the choice of the development method, a choice that focused on the SCRUM method of the AGILE methods. This made it possible, on the one hand to identify and specify the needs and on the other hand to identify the priorities until the end of the project. A second step, inspired by good design steps, using the UML modeling language, allowing realization with the PHP and JAVA languages.