

Institut Supérieur d'Informatique et de Multimédia de Sfax



Projet

2éme année License en BIG DATA & Analyse de Données

Conception et réalisation d'une application web de gestion d'institut

Réalisé Par :

SIRINE DHOUIB & MARIEM SFAXI

Cours 1 Td 1

ANNEE UNIVERSITAIRE : 2020 - 2021

Liste des figures

FIGURE 1 : EXEMPLE DE REPRESENTATION D'UN ACTEUR	10
FIGURE 2 : EXEMPLE DE REPRESENTATION D'UN CAS D'UTILISATION.....	11
FIGURE 3-4 : DIAGRAMME DE CAS D'UTILISATION D'ACTEUR «ÉTUDIANT»	12
FIGURE 5-6 : DIAGRAMME DE CAS D'UTILISATION D'ACTEUR «ENSEIGNANT»	13
FIGURE 7-8 : DIAGRAMME DE CAS D'UTILISATION D'ACTEUR «ADMINISTRATEUR»	14
FIGURE 9 : DIAGRAMME DE CAS D'UTILISATION «HERITAGE »	15
FIGURE 10 : DIAGRAMME DE CAS D'UTILISATION GLOBAL	15
FIGURE 11 : DIAGRAMME DE CAS D'UTILISATION D'AUTHENTIFICATION.....	16
FIGURE 12 : DIAGRAMME DE CAS D'UTILISATION DE MISE A JOUR.....	16
FIGURE 13 : DIAGRAMME DE CAS D'UTILISATION DE GESTION DES DONNEES	17
FIGURE 14 : DIAGRAMME DE SEQUENCE AUTHENTIFICATION	24
FIGURE 15: DIAGRAMME DE SEQUENCE DE MISE A JOUR INFORMATIONS PERSONNELLES.....	25
FIGURE 16 : DIAGRAMME DE SEQUENCE DE CONSULTATION DE LA LISTE DES ETUDIANTS	25
FIGURE 17 : DIAGRAMME DE SEQUENCE D'INSCRIPTION D'UN ETUDIANT DANS UN COURS DONNE.....	26
FIGURE 18 : DIAGRAMME DE SEQUENCE DE CREATION D'UN UTILISATEUR.....	26
FIGURE 19 : DIAGRAMME DE SEQUENCE DE INSERTION DES NOTES DES ETUDIANTS.....	27
FIGURE 20 : DIAGRAMME DE SEQUENCE DE CONSULTER LES DONNEES.....	27
FIGURE 21 : REPRESENTATION GRAPHIQUE DES NŒUDS D'ACTIVITE.....	28
FIGURE 22 : DIAGRAMME D'ACTIVITES D'AFFECTATION DES NOTES DES ETUDIANTS.....	28
FIGURE 23 : DIAGRAMME DE COLLABORATION DE CREER ETUDIANT.....	29
FIGURE 24 : DIAGRAMME DE COLLABORATION DE DE S'INSCRIRE DANS UN COURS DONNE.....	29
FIGURE 25 : DIAGRAMME DE COLLABORATION D'AUTHENTIFICATION.....	30
FIGURE 26 : DIAGRAMME DE CLASSE GLOBAL.....	30
FIGURE 27 : ARCHITECTURE DU SYSTEME.....	36
FIGURE 28 : PATTERN MVC.....	37
FIGURE 29 : PAGE D'AUTHENTIFICATION « LOGIN »	37
FIGURE 30 : PAGE D'ACCUEIL	38
FIGURE 31 : TABLEAU DE BORD D'ADMINISTRATEUR.....	38
FIGURE 32 : TABLEAU DE BORD D'ETUDIANT.....	39
FIGURE 33 : TABLEAU DE BORD D'ENSEIGNANT.....	39
FIGURE 34 : PAGE D'INSCRIPTION « REGISTER ».....	40

Liste des tableaux

TABLEAU 1 : LIVRABLES DU PROJET	8
TABLEAU 2 : ACTEURS PRIMAIRES	11
TABLEAU 3 : DESCRIPTION TEXTUELLE D'AUTHENTIFICATION	17
TABLEAU 4 : DESCRIPTION TEXTUELLE DU CONTROLE DU LOGIN ET DU MOT DE PASSE.....	17-18
TABLEAU 5 : DESCRIPTION TEXTUELLE DE RECUPERER PRIVILEGES.....	18
TABLEAU 6 : DESCRIPTION TEXTUELLE DE GESTION DES DONNEES	18
TABLEAU 7 : DESCRIPTION TEXTUELLE D'AJOUT DES DONNEES.....	18
TABLEAU 8 : DESCRIPTION TEXTUELLE DE MODIFIER DES DONNEES.....	19
TABLEAU 9 : DESCRIPTION TEXTUELLE DE SUPPRIMER DES DONNEES.....	19
TABLEAU 10 : DESCRIPTION TEXTUELLE DE LISTER DES DONNEES.....	19
TABLEAU 11 : DESCRIPTION TEXTUELLE DE METTRE A JOUR INFORMATIONS PERSONNELLES	20
TABLEAU 12 : DESCRIPTION TEXTUELLE DE S'INSCRIRE DANS UN COURS DONNE.....	20
TABLEAU 13 : DESCRIPTION TEXTUELLE DE CONSULTER LES NOTES DES COURS SUIVIS.....	20
TABLEAU 14 : DESCRIPTION TEXTUELLE D'INSERER LES NOTES DES ETUDIANTS POUR CHAQUE COURS.....	20
TABLEAU 15 : DESCRIPTION TEXTUELLE DE CONSULTER LES DONNEES.....	21
TABLEAU 16 : DESCRIPTION TEXTUELLE DE CONSULTER LA LISTE DES ETUDIANTS.....	21
TABLEAU 17 : DESCRIPTION TEXTUELLE DE CREER ETUDANT.....	21
TABLEAU 18 : DESCRIPTION TEXTUELLE DE CREER ENSEIGNANT.....	22
TABLEAU 19 : DESCRIPTION TEXTUELLE DE CREER COURS.....	22
TABLEAU 20 : DESCRIPTION TEXTUELLE DE CREER DIPLOME.....	22-23
TABLEAU 21 : DESCRIPTION TEXTUELLE DE REALISATION DES STATISTIQUES SUR NOMBRE D'ETUDIANTS.....	23
TABLEAU 22 : DESCRIPTION TEXTUELLE DE REALISATON DES STATISTIQUES SUR TAUX DE REUSSITE.....	23
TABLEAU 23 : DESCRIPTION TEXTUELLE DE REALISATION DES STATISTIQUES SUR LA MOYENNE DES ETUDIANTS.....	24
TABLEAU 24 : PRIORITE DES USE-CASE.....	24
TABLEAU 25 : DICTIONNAIRE DES DONNEES DU TABLE ETUDIANT.....	31
TABLEAU 26 : DICTIONNAIRE DES DONNEES DU TABLE ENSEIGNANT.....	31
TABLEAU 27 : DICTIONNAIRE DES DONNEES DU TABLE UTILISATEUR.....	31
TABLEAU 28 : DICTIONNAIRE DES DONNEES DU TABLE COURS.....	32
TABLEAU 29 : DICTIONNAIRE DES DONNEES DU TABLE DIPLOME.....	32
TABLEAU 30 : DICTIONNAIRE DES DONNEES DU TABLE ETUDIANT.....	32

Table des Matières

Liste des figures.....	2
Liste des tableaux	3
Introduction Générale	5
Chapitre 1 : Cahier de charge et capture des besoins.....	6
1. Introduction.....	7
2. Besoins de nos enseignants	7
3. Besoins fonctionnels.....	7
4. Besoins non fonctionnels.....	8
5. Livrables du projet	8
7. Conclusion	8
Chapitre 2 : Analyse et conception	9
1. Introduction.....	10
2. UML	10
3. Présentation des outils.....	10
4. Identification des acteurs	10
5. Diagrammes de cas d'utilisation	11
6. Description Textuelle.....	17
7. Priorité des use-case.....	24
8. Diagrammes de séquence	24
9. Diagrammes d'activités.....	28
10. Diagrammes de collaboration	29
11. Diagrammes de classe	30
12. Conclusion	32
Chapitre 3 : Réalisation	33
1. Introduction.....	34
2. Environnement et outils de développement.....	34
3. Architecture générale de l'application.....	36
4. Principales interfaces graphiques.....	37
5. Conclusion	40
Conclusion général.....	41
WEBOGRAPHIE.....	42

Introduction Générale

De nos jours, la gestion des données d'une façon automatisée occupe une place privilégiée dans le monde de l'informatique. Grâce à cette technologie, le temps de recherche des données a chuté de manière considérable, ce qui laisse croire qu'elle a encore de beaux jours devant elle.

C'est dans ce contexte que plusieurs établissements essayent de profiter au maximum possible de ces technologies afin d'améliorer leurs productivités et de faire face à quelques problèmes pénibles qui peuvent constituer un obstacle de progression.

Dans ce cadre, Mme « FAIZA GHOZZI », Mr « WAJDI BOUAZIZ » et Mme « HANENE JMAL » nous ont confié la responsabilité de développer une application web permettant de gérer un institut Public , le cas de l'Institut Supérieur D'Informatique et de Multimédia de Sfax.

La naissance de cette idée est due pour répondre à un ensemble des besoins notamment : gestion des étudiants (inscription - réinscription - création des emplois du temps ...); gestion des professeurs (gestion des informations et diplômes - affectation aux classes ...); gestion des niveaux et classes (créations des niveaux avec leurs classes, affecter à chaque niveau ses disciplines ...)

Pour mener à bien le développement de ce projet, le choix de la méthode de développement s'est porté sur la méthode agile "SCRUM". En effet, la méthode SCRUM est une solution de développement logiciel adapté à tous types de projets.

Il faut noter que l'outil Enterprise Architect m'a aidé énormément à élaborer et gérer les différents diagrammes UML.

Pour ce qui concerne le volet technique, nous utiliserons un ensemble d'outils tels que : PHP, Html, Css , bootstrap.

Ayant présenté les outils et les méthodes adoptés, nous allons exposer maintenant le plan du Projet , qui se subdivise en cinq chapitres principaux :

- Chapitre 1 : cahier de charge et capture des besoins.
- Chapitre 2 : analyse et conception.
- Chapitre 3 : réalisation.

Enfin, une conclusion sera donnée enfin de ce rapport.

Chapitre 1 : Cahier de charge et capture des besoins

1. Introduction

Ce chapitre a pour rôle de montrer les besoins de nos enseignants, ainsi que les besoins fonctionnels et non fonctionnels de la future application, de préciser les livrables et les risques les plus critiques du projet.

2. Besoins de nos enseignants

Nos enseignants souhaitent établir la mise en œuvre d'un portail-web sur la gestion de l'Institut Supérieur d'Informatique et de Multimédia qui comporte un ensemble de modules :

- **Gestion des étudiants :**

Ce module contient quatre volets qui sont : l'inscription ainsi que l'exploitation des notes obtenues.

- **L'inscription qui se divise en deux.**

Celle des étudiants qui vont s'inscrire pour la première fois en première année .

Parmi les pièces d'inscription, on trouve les photos d'identité de l'étudiant, la photocopie de la CIN, l'adresse, le Téléphone Mobile et le fixe s'il existe.

Le règlement des frais d'inscription est obligatoire.

- **Gestion des emplois du temps :**

Ce module est divisé en deux, celui des enseignants et celui des étudiants.

Celui des enseignants :

l'administration établit un emploi de temps cohérent répondant aux directives et aux normes des instructions officielles du ministère de l'enseignement supérieur pour chaque niveau et pour chaque discipline, en respectant le nombre d'heures pour tel ou tel niveau concernant telle ou telle matière.

Celui des étudiants :

l'emploi du temps doit être compatible non chargé (maximum 7 heures/jour), il doit comporter les matières autorisées à être enseignées dans tel ou tel niveau.

- **Gestion des enseignants :**

Deux catégories de professeurs sont à mentionner :

La gestion des enseignants consiste à tenir compte de leur dossier professionnel (diplômes, ancienneté,). Elle consiste à tenir compte de leurs absences et retards (justifiés ou non) durant l'année universitaire.

3. Besoins fonctionnels

Avant d'imposer une solution, il faut se tourner vers le demandeur, pour aboutir de manière structurée à la solution. En effet, le but du projet est de satisfaire le besoin. Il faut exprimer clairement les objectifs à atteindre du projet, afin d'éviter toute confusion entre nous et le demandeur.

Le futur système doit permettre à l'utilisateur de gérer un ensemble de modules, dont on trouve :

- ✓ Gestion des étudiants.
- ✓ Gestions des emplois du temps.
- ✓ Gestions des enseignants.

4. Besoins non fonctionnels

Les besoins non fonctionnels décrivent les objectifs liés aux performances du système et aux contraintes de son environnement. Ses exigences techniques sont souvent exprimées sous forme d'objectifs spécifiques que doit atteindre le système.

La maintenabilité : le code doit être compréhensible par simple lecture, notamment en respectant les règles de gestion et les normes de développement.

Exploitabilité : les impacts en termes de performances doivent être pris en compte lors de développement, ainsi que la consommation des ressources (CPU, mémoire, etc.) qui doit être minimisée.

Capacité fonctionnelle et convivialité : les composants développés doivent respecter les spécifications fournies par nos enseignants. Le système doit être facilement utilisable et disposer d'interfaces conviviales, notamment par le respect des règles d'ergonomie de nos enseignants.

Sécurité : Chaque utilisateur, pour accéder à l'application, est obligé de s'authentifier par un nom d'utilisateur et un mot de passe. Il ne pourra accéder qu'aux pages qui lui sont permises par son profil ou les droits d'accès qui lui sont affectés par l'administrateur.

Les mots de passe ne sont enregistrés dans la base qu'après un chiffrement (MD5).

5. Livrables du projet

Phase	Livrable	Responsable
Etude des Besoins	Cahier de Charge	Sirine Dhouib & Mariem Sfaxi
Analyse et Conception	Diagrammes UML	Sirine Dhouib & Mariem Sfaxi
Codage et Test	Application Web	Sirine Dhouib & Mariem Sfaxi
Documentation	Rapport du Projet	Sirine Dhouib & Mariem Sfaxi

Tableau 1 : Livrables du projet

6. Conclusion

A la fin de ce chapitre, on a bien étudié les besoins des enseignants ; on a présenté l'ensemble des fonctionnalités du futur portail de manière organisée dans les différents cycles de l'application soit fonctionnel ou non fonctionnel, les livrables ainsi que les risques du projet.

Chapitre 2 : Analyse et conception

1. Introduction

Dans cette partie, on va analyser et modéliser les besoins des enseignants avec le langage UML.

L'activité d'analyse et de conception permet de traduire les besoins fonctionnels et les contraintes issues du cahier des charges et de la spécification des exigences dans un langage plus professionnel et compréhensible par tous les individus intervenants dans la réalisation et l'utilisation de l'application.

2. UML

Choix d'UML :



UML, c'est l'acronyme anglais pour « Unified Modeling Language ». On le traduit par « Langage de modélisation unifié ». La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc.

3. Présentation des outils



Parmi les **fonctionnalités** principales d'**Enterprise Architect** : Gestion des exigences. La modélisation des exigences inclut la personnalisation et la manière dont sont documentées les exigences. Cela permet de lier les exigences à la conception et à l'implémentation permettant la traçabilité des exigences.

4. Identification des acteurs

a. Acteur

Un acteur 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 (figure 1) avec son nom (son rôle) inscrit dessous.

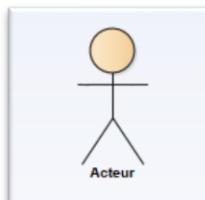


Figure 1 : Exemple de représentation d'un acteur

b. Cas d'utilisation

Un cas d'utilisation est un service rendu à un acteur : c'est une fonctionnalité de son point de vue.
Un cas d'utilisation se représente par une ellipse (figure 2) contenant le nom du cas (un verbe à l'infinitif), et optionnellement, au-dessus du nom, un stéréotype.



Figure 2 : Exemple de représentation d'un cas d'utilisation

c. Acteur Direct

Ce sont les utilisateurs de l'application, qui touchent directement l'application.

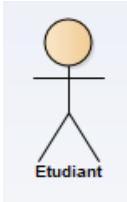
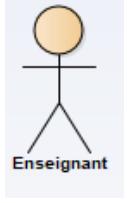
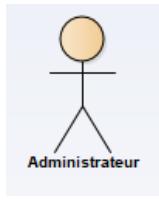
Type D'acteur	Description Fonctionnelle
 Etudiant	L'acteur le plus important, qui aura les fonctionnalités suivantes : <ul style="list-style-type: none">• Consulter ses données.• Modifier les informations Personnelles(adresse mail,Tel...)• S'inscrire dans un cours donné.• Consulter les notes des cours suivis.
 Enseignant	A pour fonctionnalités : <ul style="list-style-type: none">• Mettre à jour ses données.• Insérer les notes des étudiants pour chaque cours.• Afficher la liste de ses étudiants.
 Administrateur	Ce type d'acteur aura les fonctionnalités suivantes : <ul style="list-style-type: none">• Créer un nouvel enseignant ,étudiant ,diplôme et cours.• Consulter etModifier les données d'un étudiant, d'un diplôme et d'un cours.• Réaliser des Statistiques sur le nombre des étudiants inscrits dans chaque diplôme et cours, le taux de réussite par diplôme et par cours et la moyenne des notes des étudiants.

Tableau 2 : Acteurs Primaires

5. Diagramme de cas d'utilisation

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel.

Dans les figures qui suivent, nous présenterons les cas d'utilisation qui mettent en évidence les principales fonctionnalités de chaque acteur dans le système.

a. Diagramme de cas d'utilisation d'acteur «Étudiant»

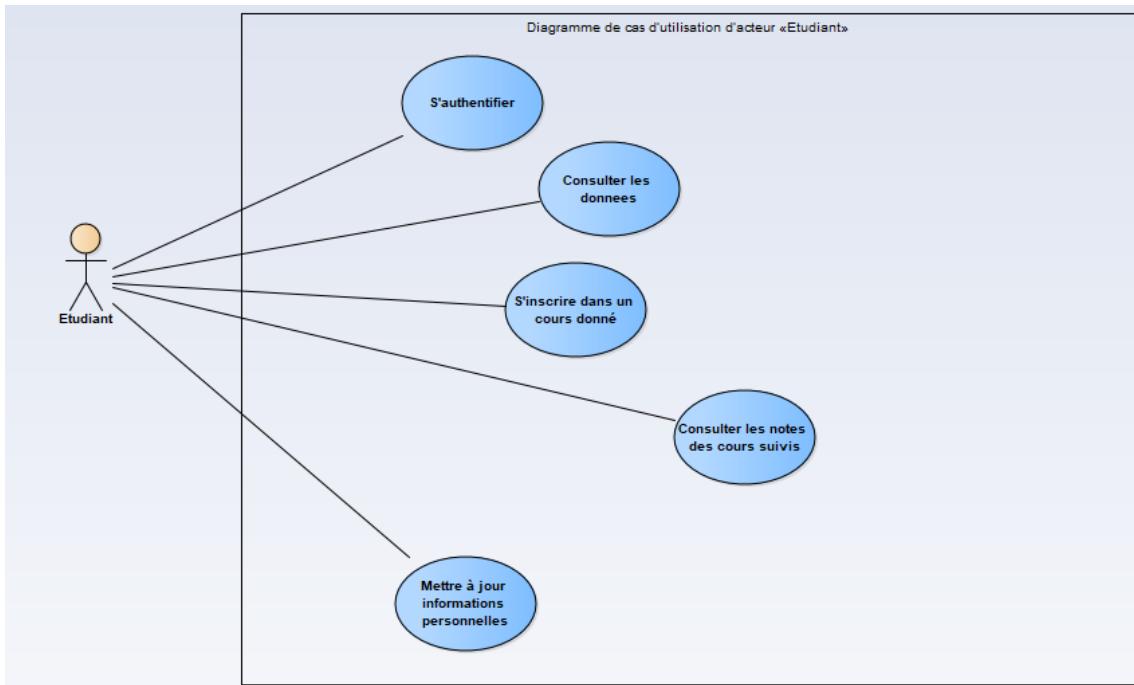


Figure 3 : Diagramme de cas d'utilisation d'acteur «Étudiant»

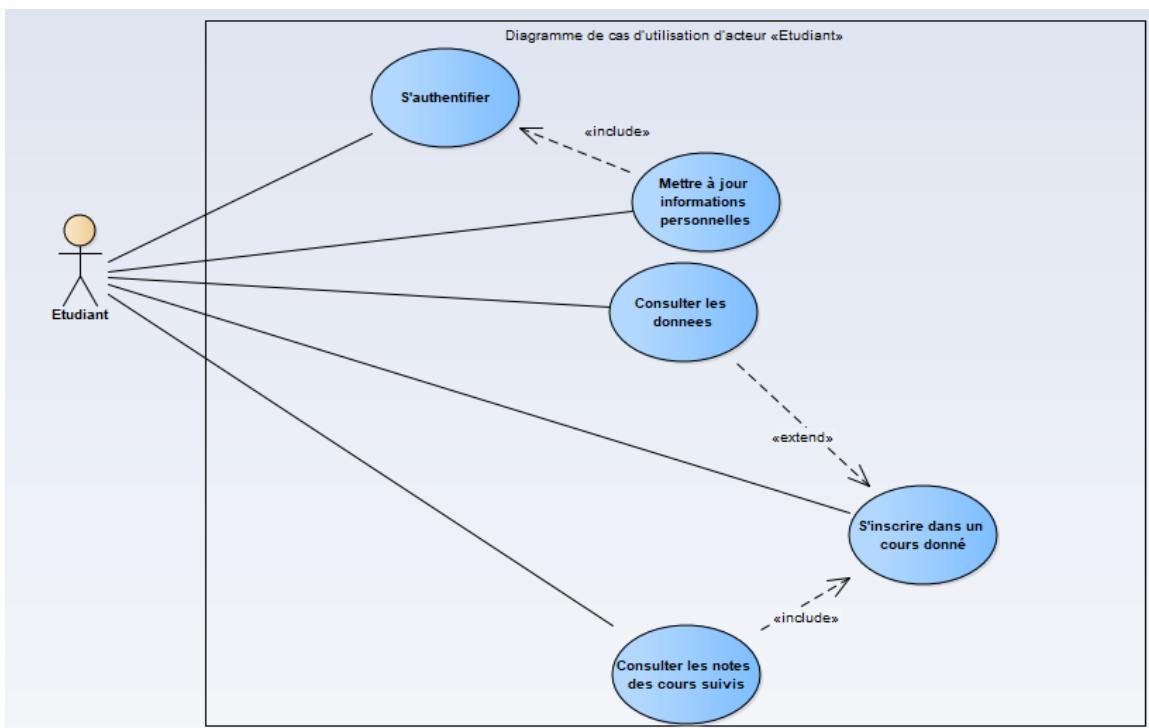


Figure 4 : Diagramme de cas d'utilisation Structuré d'acteur «Étudiant»

b. Diagramme de cas d'utilisation d'acteur «Enseignant»

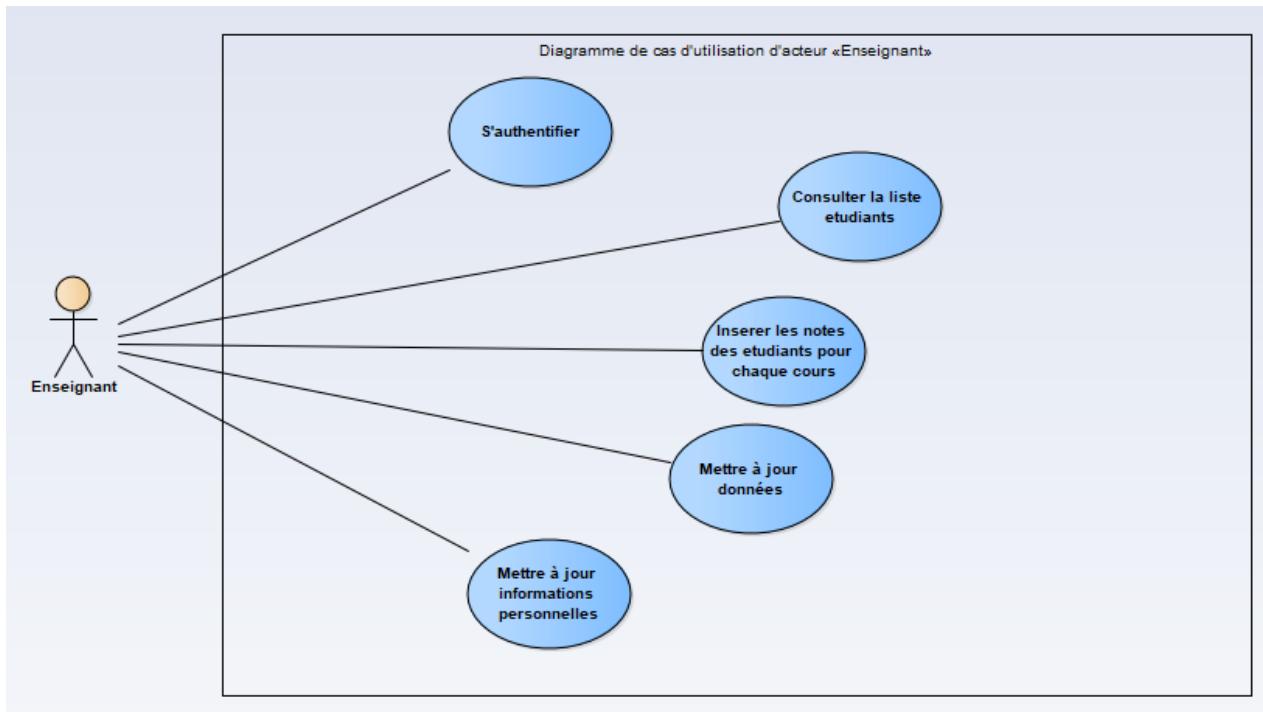


Figure 5 : Diagramme de cas d'utilisation d'acteur «Enseignant»

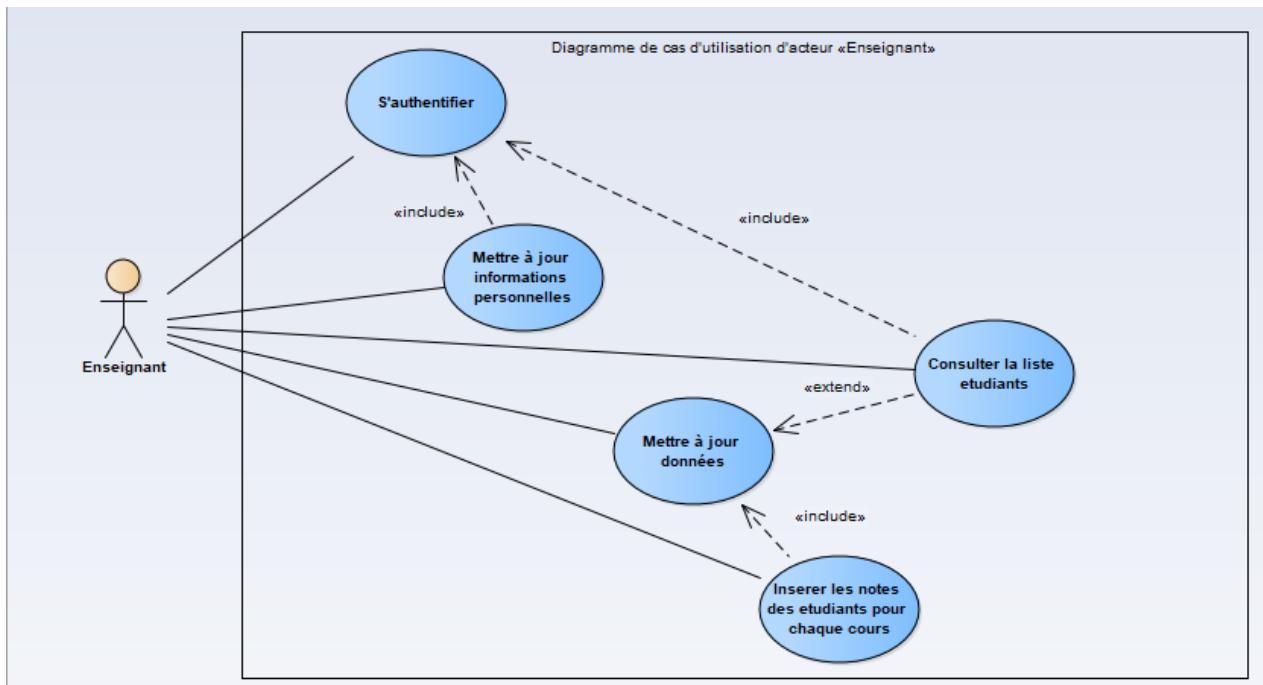


Figure 6 : Diagramme de cas d'utilisation Structuré d'acteur «Enseignant»

c. Diagramme de cas d'utilisation d'acteur «Administrateur»

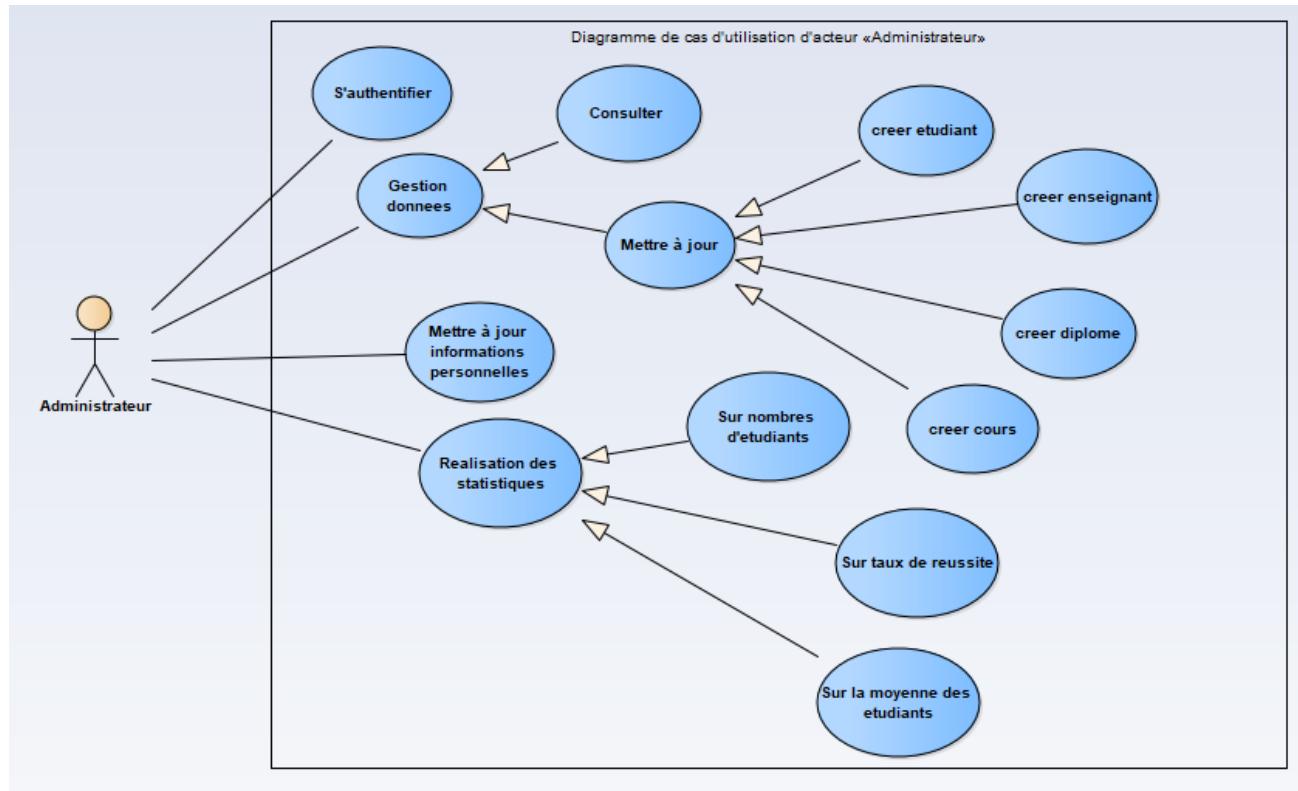


Figure 7 : Diagramme de cas d'utilisation d'acteur «Administrateur»

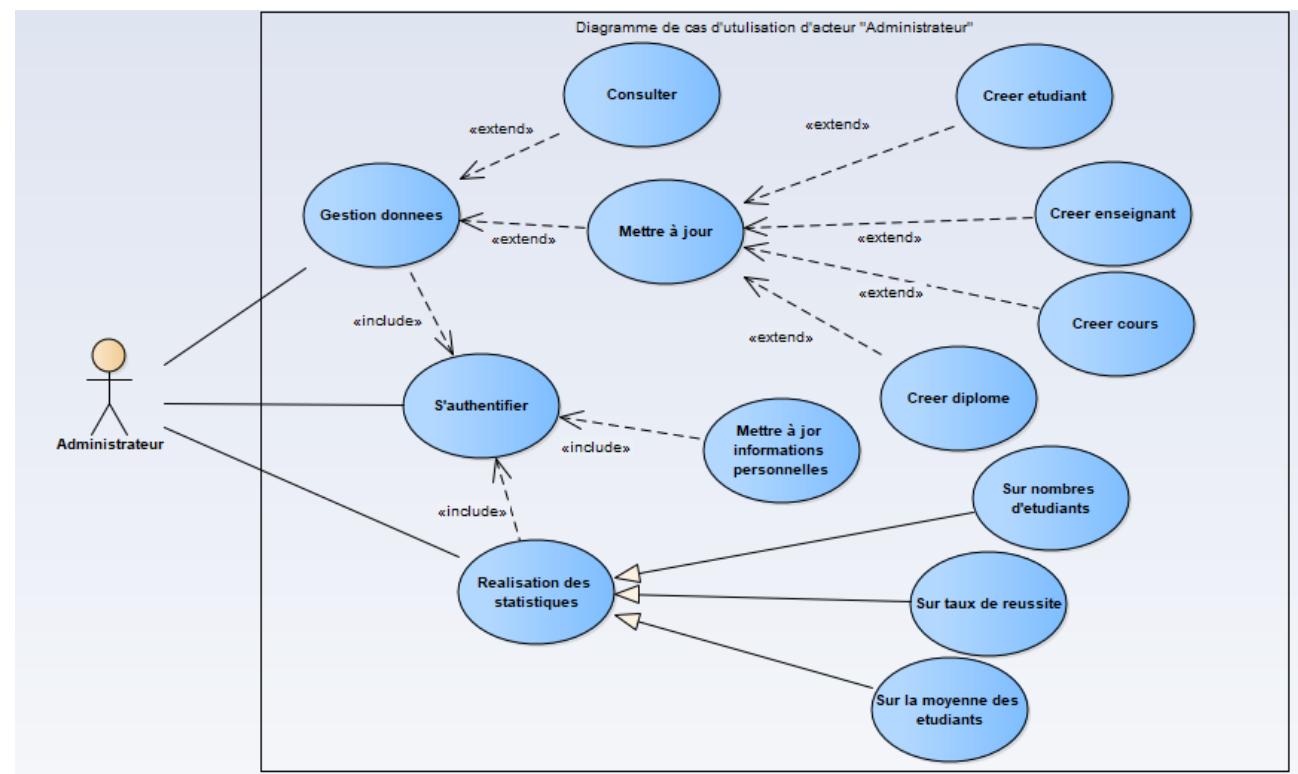


Figure 8 : Diagramme de cas d'utilisation Structuré d'acteur «Administrateur»

Dans notre cas, les trois acteurs (étudiant, Enseignant, Administrateur) possèdent une association commune « consulter le calendrier des évaluations ».

Afin d'éviter de surcharger le diagramme, il est possible de définir une relation d'héritage entre chacun de ces acteurs avec un acteur « utilisateur », qui possède l'association commune.

d. Diagramme de cas d'utilisation « Héritage »

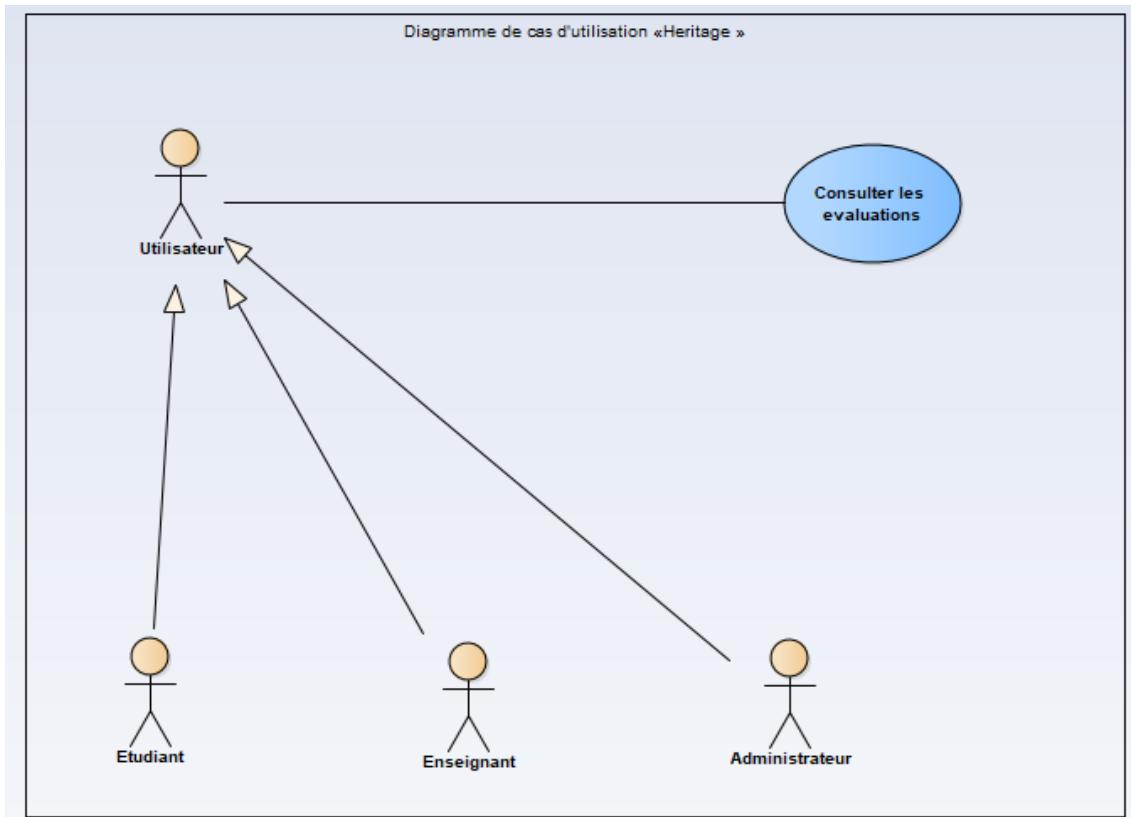


Figure 9 : Diagramme de cas d'utilisation « Héritage »

e. Diagramme de cas d'utilisation « Global »

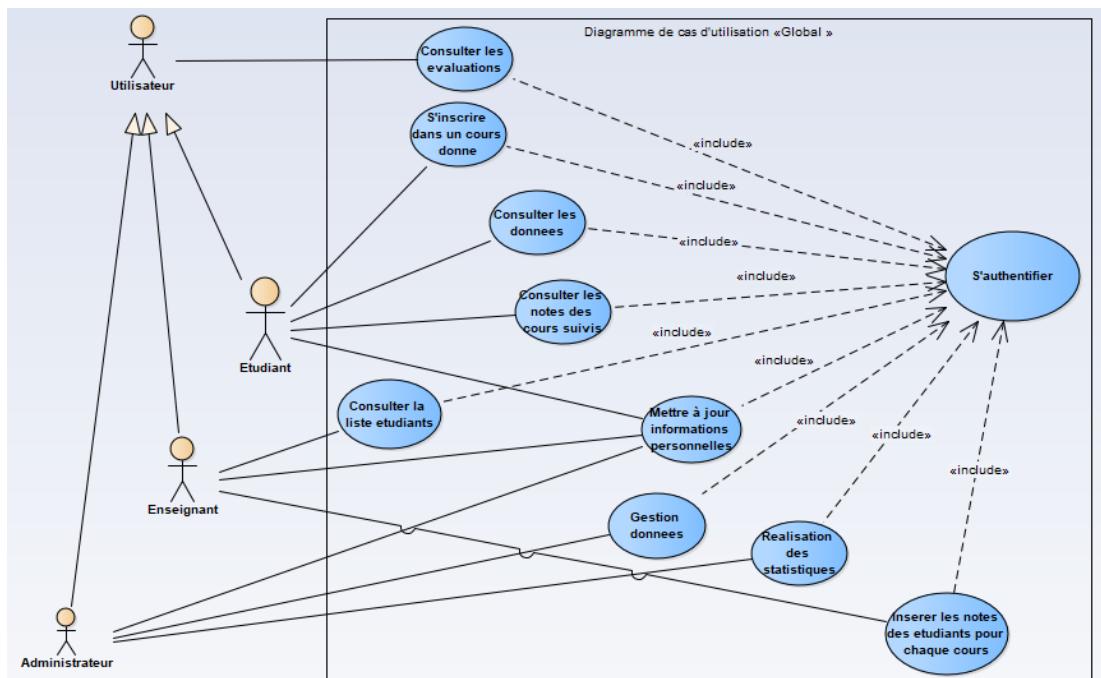


Figure 10 : Diagramme de cas d'utilisation « Global »

f. Diagramme de cas d'utilisation «Authentification»

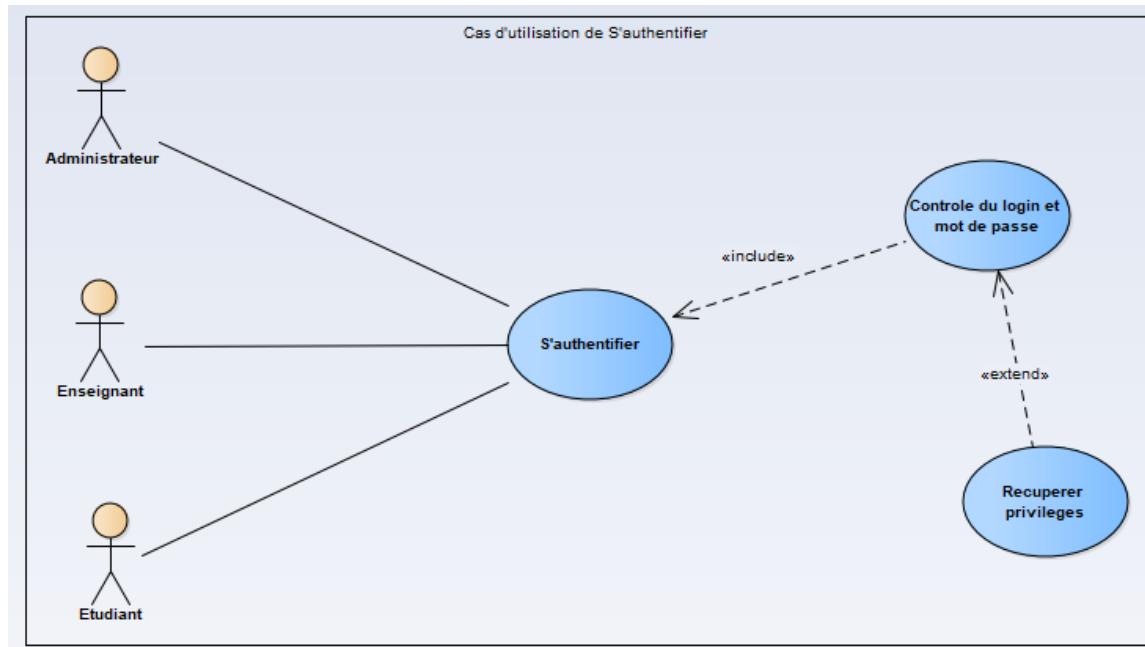


Figure 11 : Diagramme de cas d'utilisation «Authentification»

g. Diagramme de cas d'utilisation «Mise A Jour»

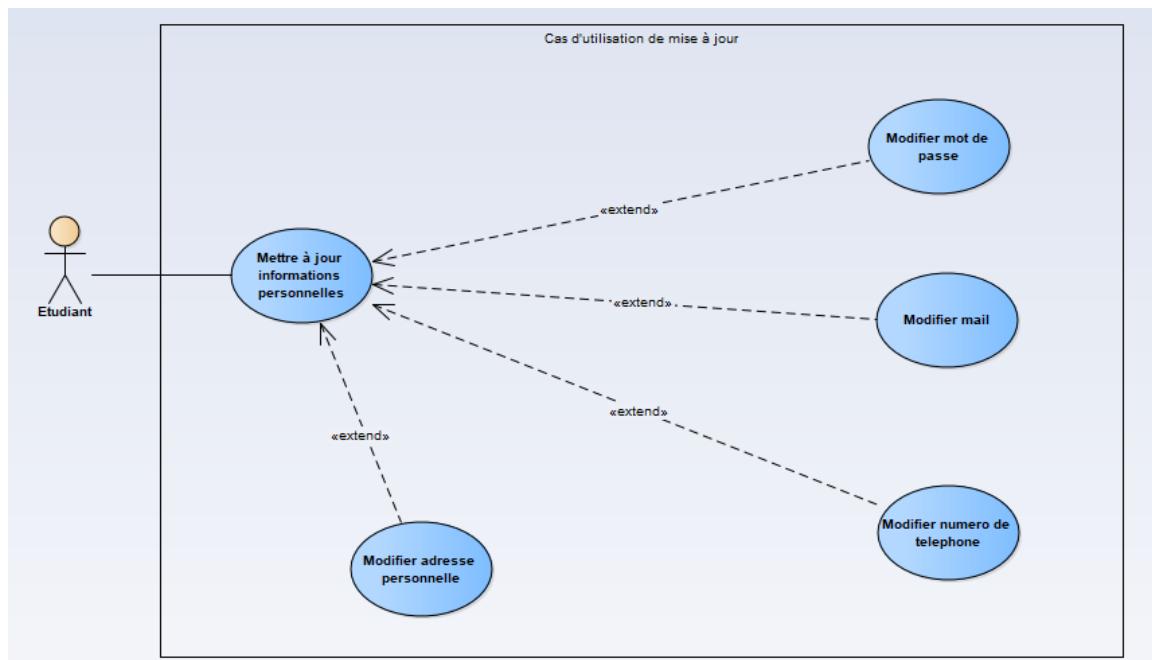


Figure 12 : Diagramme de cas d'utilisation «Mise a jour»

h. Diagramme de cas d'utilisation «Gestion des Données»

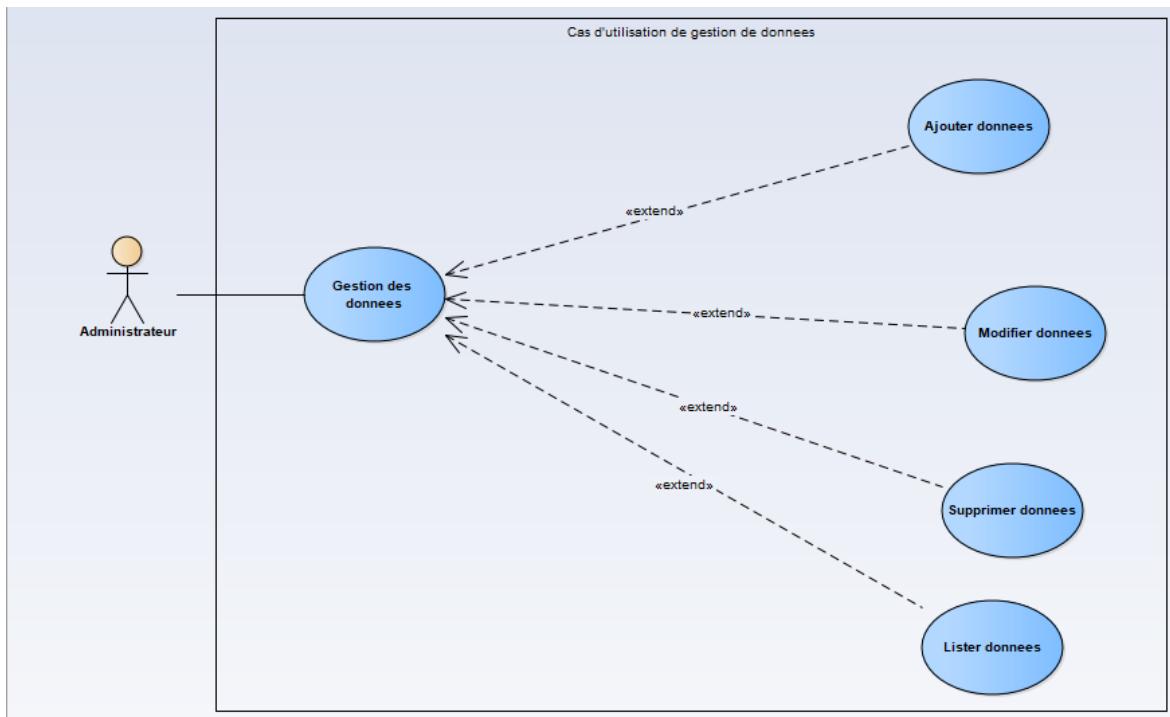


Figure 13 : Diagramme de cas d'utilisation «Gestion des Données»

6. Description Textuelle

a. Description Textuelle «s'authentifier»

Ce cas d'utilisation permet d'identifier de manière unique chaque utilisateur par son login et son mot de passe.

Cas d'utilisation	S'authentifier
Acteurs	Administrateur , Enseignant , Etudiant
Pré-condition	- Une authentification préalable - Base de données crée contenant les données d'authentification
Post-condition	Affiche le formulaire de l'authentification
Scénario nominal	1-L'utilisateur introduit son login et son mot de passe 2- Le système vérifie l'authenticité des infos introduites par l'utilisateur 3- Authentification réussie
Scénario alternatif	-Si le nom d'utilisateur ou le mot de passe erroné, le système affiche un message d'erreur. -Le système le redirige au début du scenario principal.

Tableau 3 : Description Textuelle « s'authentifier »

b. Description Textuelle «Contrôle du login et mot de passe»

Ce cas d'utilisation permet de valider de manière sûre chaque utilisateur qui s'identifie.

Cas d'utilisation	Contrôle du login et mot de passe
Acteurs	Administrateur , Enseignant , Etudiant
Pré-condition	- Une authentification préalable - Base de données crée contenant les données d'authentification
Post-condition	Affiche le formulaire de l'authentification

Scénario nominal	1-L'utilisateur introduit son login et son mot de passe 2- Le système vérifie l'authenticité des infos introduites par l'utilisateur
Scénario alternatif	-Si le nom d'utilisateur ou le mot de passe erroné, le système affiche un message d'erreur. -Le système le redirige au début du scenario principal.

Tableau 4 : Description Textuelle « contrôle du login et du mot de passe »

c. Description Textuelle «Récupérer privilèges»

Ce cas d'utilisation permet d'autoriser de manière unique à chaque utilisateur les privilèges correspondants.

Cas d'utilisation	Récupérer privilèges
Acteurs	Administrateur , Enseignant , Etudiant
Pré-condition	- Une authentification préalable
Post-condition	Affiche le tableau de bord
Scénario nominal	1-L'utilisateur introduit son login et son mot de passe 2- Le système vérifie l'authenticité des infos introduites par l'utilisateur 3- Le système affiche le tableau de bord selon les privilèges donnés à l'utilisateur
Scénario alternatif	Les tableau de bord ne s'affiche pas

Tableau 5 : Description Textuelle « Récupérer Privilèges »

d. Description Textuelle «Gestion des données»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique l'ajout , la modification et la suppression des données.

Cas d'utilisation	Gestion des données
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Des listes des étudiants , enseignants , matières , cours sont affichées sur l'écran
Scénario nominal	1. L'administrateur : ajoute, modifie et supprime des étudiants , enseignants , matières , cours 2. L'administrateur fait des recherches et tries 3. Le système enregistre les modifications et affiche un message de mise à jour réussie avec succès.
Scénario alternatif	- Si un problème apparaît un message d'erreur s'affiche

Tableau 6 : Description Textuelle « Gestion des données »

e. Description Textuelle «Ajouter des données»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique l'ajout des données.

Cas d'utilisation	Ajouter des données
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Des listes des étudiants , enseignants , matières , cours sont affichées sur l'écran
Scénario nominal	1. L'administrateur : ajoute des étudiants , enseignants , matières , cours 2. L'administrateur fait des recherches et tries 3. Le système enregistre les modifications et affiche un message de mise à jour réussie avec succès.
Scénario alternatif	- Si un problème apparaît un message d'erreur s'affiche

Tableau 7 : Description Textuelle « ajouter des données »

f. Description Textuelle «Modifier des données»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique la modification des données.

Cas d'utilisation	Modifier des données
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Des listes des étudiants , enseignants , matières , cours sont affichées sur l'écran
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur : modifie des étudiants , enseignants , matières , cours 2. L'administrateur fait des recherches et tries 3. Le système enregistre les modifications et affiche un message de mise à jour réussie avec succès.
Scénario alternatif	- Si un problème apparaît un message d'erreur s'affiche

Tableau 8 : Description Textuelle « Modifier des données »

g. Description Textuelle «Supprimer des données»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique la suppression des données.

Cas d'utilisation	Supprimer des données
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Des listes des étudiants , enseignants , matières , cours sont affichées sur l'écran
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur : supprime des étudiants , enseignants , matières , cours 2. L'administrateur fait des recherches et tries 3. Le système enregistre les modifications et affiche un message de mise à jour réussie avec succès.
Scénario alternatif	- Si un problème apparaît un message d'erreur s'affiche

Tableau 9 : Description Textuelle « supprimer des données »

h. Description Textuelle «Lister des données»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique l'affichage des données.

Cas d'utilisation	Lister des données
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Des listes des étudiants , enseignants , matières , cours sont affichées sur l'écran
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur : affiche des étudiants , enseignants , matières , cours 2. L'administrateur fait des recherches et tries 3. Le système enregistre les modifications et affiche un message de mise à jour réussie avec succès.
Scénario alternatif	- Si un problème apparaît un message d'erreur s'affiche

Tableau 10 : Description Textuelle « Lister des données »

i. Description Textuelle «Mettre à jour informations personnelles»

Ce cas d'utilisation permet d'autoriser à l'enseignant et à l'étudiant de manière unique la mise à jour des informations personnelles.

Cas d'utilisation	Mettre à jour informations
Acteurs	Enseignant , Etudiant
Pré-condition	Une authentification préalable
Post-condition	Les informations ont été mises à jour
Scénario nominal	1 - L'utilisateur modifie les informations 2 - Le système vérifie les données saisies 3 - Le système enregistre les données et affiche un message de succès
Scénario alternatif	- L'utilisateur saisit des données invalides ou laisse des champs vides - le système affiche un message d'erreur

Tableau 11 : Description Textuelle « Mettre à jour Informations Personnelles »

j. Description Textuelle «S'inscrire dans un cours donné»

Ce cas d'utilisation permet d'autoriser à l'étudiant de manière unique son inscription dans un cours donné.

Cas d'utilisation	S'inscrire dans un cours donné
Acteurs	Etudiant
Pré-condition	Une authentification préalable
Post-condition	L'inscription est mise à jour
Scénario nominal	1. L'étudiant saisit le code d'inscription 2. Le système vérifie le code 3. L'accès au cours est effectué
Scénario alternatif	- L'étudiant saisit un code invalide ou laisse le champ vide - Le système affiche un message d'erreur

Tableau 12 : Description Textuelle « s'inscrire dans un cours donné »

k. Description Textuelle «Consulter les notes des cours suivis»

Ce cas d'utilisation permet d'autoriser à l'étudiant de manière unique sa consultation des notes de ses cours suivis.

Cas d'utilisation	Consulter les notes des cours suivis
Acteurs	Etudiant
Pré-condition	Une inscription effectuée
Post-condition	Les notes des cours sont affichés
Scénario nominal	1. L'étudiant accède au cours dans lequel il est inscrit 2. Le système affiche les notes des cours suivis
Scénario alternatif	- Les notes des cours ne sont pas disponibles

Tableau 13 : Description Textuelle « consulter les notes d'un cours suivi »

l. Description Textuelle «Insérer les notes des étudiants pour chaque cours»

Ce cas d'utilisation permet d'autoriser à l'enseignant d'attribuer des notes pour ses étudiants dans chaque cours.

Cas d'utilisation	Insérer les notes des étudiants pour chaque cours
Acteurs	Enseignant
Pré-condition	Une authentification préalable
Post-condition	Les notes des étudiants pour chaque cours sont mises à jour
Scénario nominal	1. L'enseignant insère des notes pour chaque étudiant 2. Le système enregistre les modifications et affiche un message de mise à jour réussie avec succès.
Scénario alternatif	- Si un problème apparaît un message d'erreur s'affiche

Tableau 14 : Description Textuelle « Insérer les notes des étudiants pour chaque cours »

m. Description Textuelle «consulter les données»

Ce cas d'utilisation permet d'autoriser à l'étudiant de manière unique la consultation de ses données.

Cas d'utilisation	Consulter les données
Acteurs	Etudiant
Pré-condition	Authentification préalable
Post-condition	La liste des données demandées est affichée sur l'écran
Scénario nominal	1-l'étudiant demande l'affichage de la liste des données 2-le système affiche la liste des données demandées
Scénario alternatif	- Aucun résultat - Le système affiche un message de type «aucun résultat trouvé»

Tableau 15 : Description Textuelle « consulter les données »

n. Description Textuelle «consulter la liste des étudiants»

Ce cas d'utilisation permet d'autoriser à l'enseignant de manière unique la consultation de la liste de ses étudiants.

Cas d'utilisation	Consulter la liste des étudiants
Acteurs	Enseignant
Pré-condition	Authentification préalable
Post-condition	La liste des étudiants est affichée sur l'écran
Scénario nominal	1. L'enseignant demande l'affichage de la liste des étudiants 2. Le système affiche la liste des étudiants
Scénario alternatif	-Aucun résultat -Le système affiche un message de type « aucun résultat trouvé »

Tableau 16 : Description Textuelle « consulter la liste des étudiants »

o. Description Textuelle «créer étudiant»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique la création d'un nouveau étudiant.

Cas d'utilisation	Créer étudiant
Acteurs	Administrateur, Etudiant
Pré-condition	Authentification préalable de l'administrateur
Post-condition	Un nouveau étudiant est ajouté
Scénario nominal	1. L'utilisateur demande le formulaire d'ajout 2. Le système affiche le formulaire 3. L'utilisateur remplit les champs nécessaires et les valide 4. Le système vérifie les données saisies 5. Le système enregistre le nouvel utilisateur et affiche un message de succès de l'opération
Scénario alternatif	-L'utilisateur saisie des données manquantes ou invalides -Le système affiche un message d'erreur -reprise de l'étape 3 du scénario nominal - L'étudiant existe déjà -Le système demande à l'utilisateur de modifier l'identifiant ou se connecter. -reprise de l'étape 3 du scénario nominal

Tableau 17 : Description Textuelle « créer étudiant »

p. Description Textuelle «créer enseignant»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique la création d'un nouveau enseignant.

Cas d'utilisation	Créer enseignant
Acteurs	Administrateur, Enseignant
Pré-condition	Authentification préalable de l'administrateur
Post-condition	Un nouveau enseignant est ajouté
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur demande le formulaire d'ajout 2. Le système affiche le formulaire 3. L'utilisateur rempli les champs nécessaires et les valide 4. Le système vérifie les données saisies 5. Le système enregistre le nouvel utilisateur et affiche un message de succès de l'opération
Scénario alternatif	<ul style="list-style-type: none"> -L'utilisateur saisie des données manquantes ou invalides -Le système affiche un message d'erreur -reprise de l'étape 3 du scénario nominal - L'enseignant existe déjà <p>-Le système demande à l'utilisateur de modifier l'identifiant ou se connecter.</p> <ul style="list-style-type: none"> -reprise de l'étape 3 du scénario nominal

Tableau 18 : Description Textuelle « créer enseignant »

q. Description Textuelle «créer cours»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique la création d'un nouveau cours.

Cas d'utilisation	Créer cours
Acteurs	Administrateur
Pré-condition	Authentification préalable de l'administrateur
Post-condition	Un cours est ajouté
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur demande le formulaire d'ajout 2. Le système affiche le formulaire 3. L'utilisateur rempli les champs nécessaires et les valide 4. Le système vérifie les données saisies 5. Le système enregistre l'addition du cours et affiche un message de succès de l'opération
Scénario alternatif	<ul style="list-style-type: none"> -L'utilisateur saisie des données manquantes ou invalides -Le système affiche un message d'erreur -reprise de l'étape 3 du scénario nominal - Le cours existe déjà <p>-Le système demande à l'utilisateur de modifier l'identifiant ou se connecter.</p> <ul style="list-style-type: none"> -reprise de l'étape 3 du scénario nominal

Tableau 19 : Description Textuelle « créer cours »

r. Description Textuelle «créer diplôme»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique la création d'un nouveau diplôme.

Cas d'utilisation	Créer diplôme
Acteurs	Administrateur
Pré-condition	Authentification préalable de l'administrateur
Post-condition	Un diplôme est ajouté
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur demande le formulaire d'ajout 2. Le système affiche le formulaire

	<p>3. L'utilisateur rempli les champs nécessaires et les valide</p> <p>4. Le système vérifie les données saisies</p> <p>5. Le système enregistre l'addition du diplôme et affiche un message de succès de l'opération</p>
Scénario alternatif	<ul style="list-style-type: none"> -L'utilisateur saisie des données manquantes ou invalides -Le système affiche un message d'erreur -reprise de l'étape 3 du scénario nominal - Le diplôme existe déjà <p>-Le système demande à l'utilisateur de modifier l'identifiant ou se connecter.</p> <ul style="list-style-type: none"> -reprise de l'étape 3 du scénario nominal

Tableau 20 : Description Textuelle « créer diplome »

s. Description Textuelle «Réalisation des statistiques sur nombre d'étudiants»

Ce cas d'utilisation permet d'autoriser a l'administrateur de manière unique l'évaluation numérique des données et l'élaboration des statistiques sur le nombre d'étudiants.

Cas d'utilisation	Réalisation des statistiques sur nombres d'étudiants
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Les statistiques sont affichés sur l'écran
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur consulte la liste des étudiants 2. L'administrateur compte le nombre des étudiants 3. L'administrateur choisit le diagramme approprié 4. Le système affiche le résultat de la réalisation statistique
Scénario alternatif	<ul style="list-style-type: none"> - Le système affiche un message d'erreur

Tableau 21 : Description Textuelle « réalisation des statistiques sur nombre d'étudiants»

t. Description Textuelle «Réalisation des statistiques sur taux de réussite»

Ce cas d'utilisation permet d'autoriser a l'administrateur de manière unique l'évaluation numérique des données et l'élaboration des statistiques sur le taux de réussite.

Cas d'utilisation	Réalisation des statistiques sur taux de réussite
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Les statistiques sont affichés sur l'écran
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur consulte la liste des étudiants qui réussissent 2. L'administrateur compte le taux de réussite 3. L'administrateur choisit le diagramme approprié 4. Le système affiche le résultat de la réalisation statistique
Scénario alternatif	<ul style="list-style-type: none"> - Le système affiche un message d'erreur

Tableau 22 : Description Textuelle « Réalisation des statistiques sur taux de réussite »

u. Description Textuelle «Réalisation des statistiques sur la moyenne des étudiants»

Ce cas d'utilisation permet d'autoriser à l'administrateur de manière unique l'évaluation numérique des données et l'élaboration des statistiques sur la moyenne des étudiants.

Cas d'utilisation	Réalisation des statistiques sur la moyenne des étudiants
Acteurs	Administrateur
Pré-condition	Une authentification préalable
Post-condition	Les statistiques sont affichés sur l'écran
Scénario nominal	1. L'administrateur consulte la liste des étudiants 2. L'administrateur compte la moyenne des étudiants 3. L'administrateur choisit le diagramme approprié 4. Le système affiche le résultat de la réalisation statistique
Scénario alternatif	- Le système affiche un message d'erreur

Tableau 23 : Description Textuelle « réalisation des statistiques sur la moyenne des étudiants »

7. Priorité des uses-case

	Importance	Critique (Complexité)	Priorité
s'authentifier	2	5	5
consulter les données	5	1	1
s'inscrire dans un cours donné	3	2	2
consulter les notes d'un cours suivi	5	1	1
mettre à jour informations personnelles	3	4	4
consulter la liste des étudiants	5	1	1
insérer les notes des étudiants pour chaque cours	4	3	3
mettre à jour données	3	4	4
gestion données	4	4	4
réalisation des statistiques	1	3	3
consulter les évaluations	5	1	1

Tableau 24 : Prorité des uses-case

8. Diagramme de Séquence

Pour mieux concrétiser les interactions entre les acteurs du système vis-à-vis de l'application et la base de données, nous sommes amenés à traduire nos scénarios en diagrammes de séquence.

Les diagrammes de séquence servent à illustrer les cas d'utilisation. Ils permettent de représenter des collaborations entre les objets selon un point de vue temporel, on y met l'accent sur la chronologie des envois des messages.

a. Diagramme de Séquence «s'authentifier»

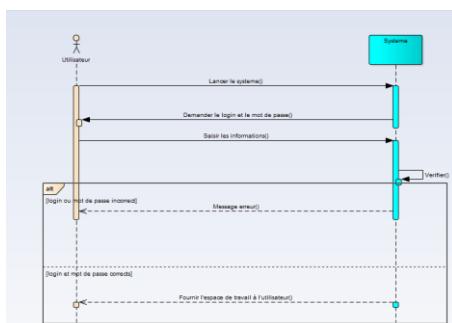


Figure 14 : Diagramme de séquence «S'authentifier»

b. Diagramme de Séquence «mettre à jour informations personnelles»

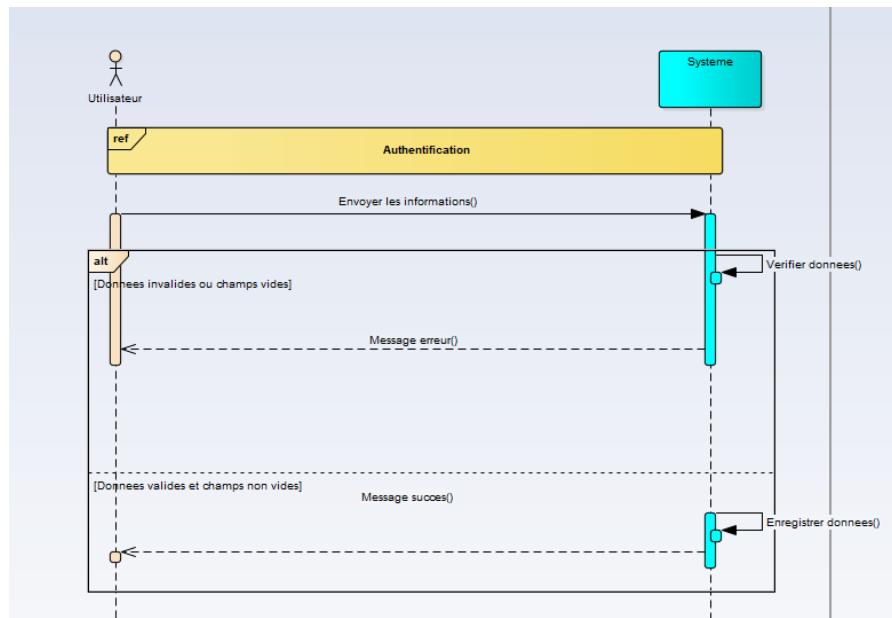


Figure 15 : Diagramme de séquence «Mettre a jour Informations Personnelles»

c. Diagramme de Séquence «consulter la liste des étudiants»

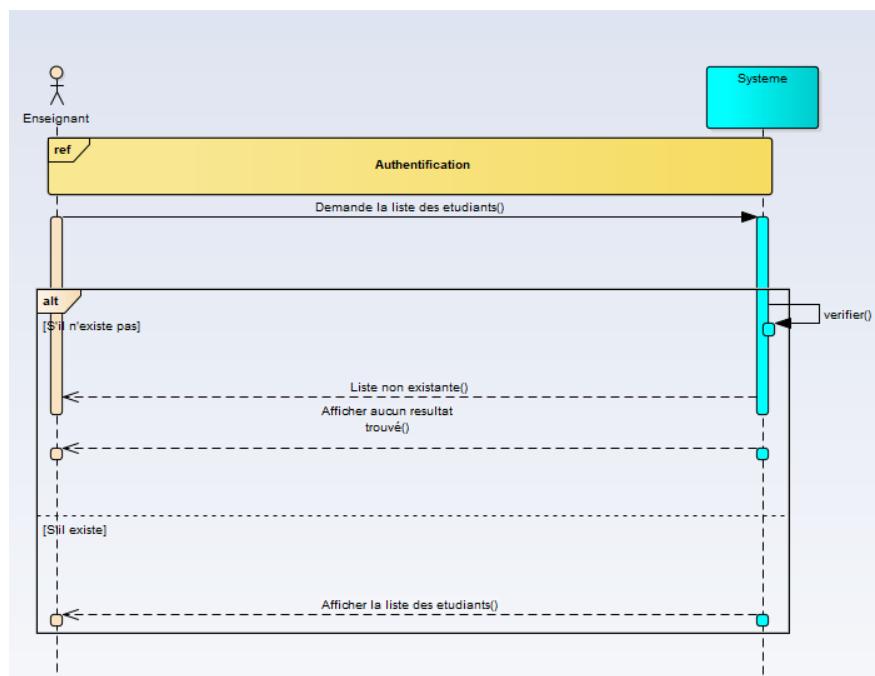


Figure 16 : Diagramme de séquence «Consulter la liste des étudiants»

d. Diagramme de Séquence «s'inscrire dans un cours donné»

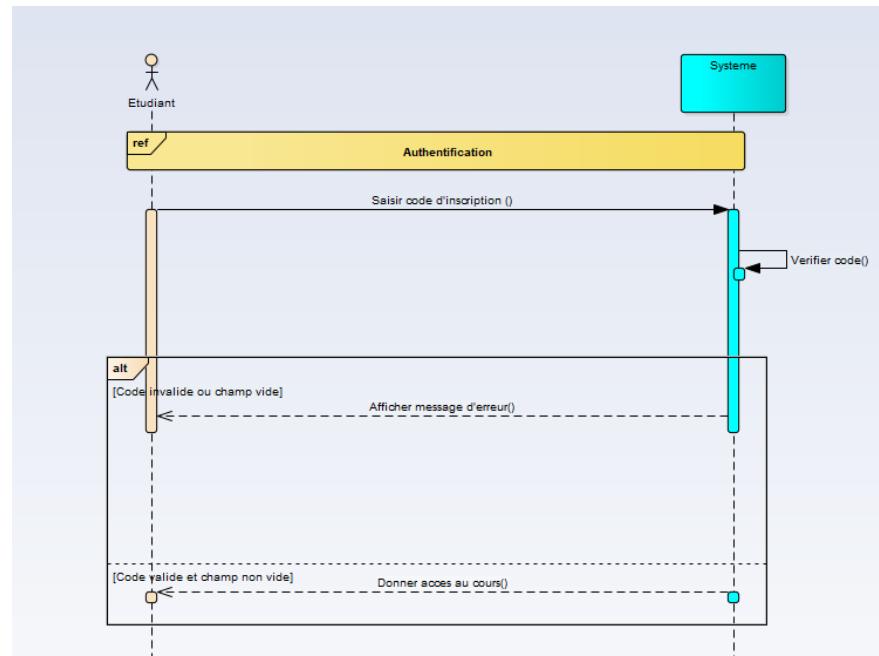


Figure 17 : Diagramme de séquence «S'inscrire dans un cours donné»

e. Diagramme de Séquence «créer utilisateur»

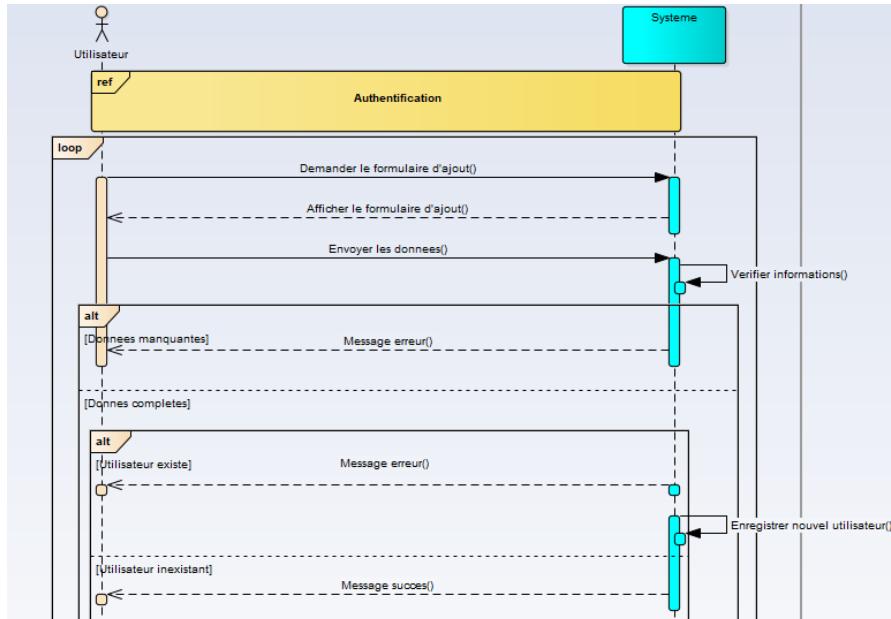


Figure 18 : Diagramme de séquence «Créer Utilisateur»

f. Diagramme de Séquence «Insérer les notes des étudiants»

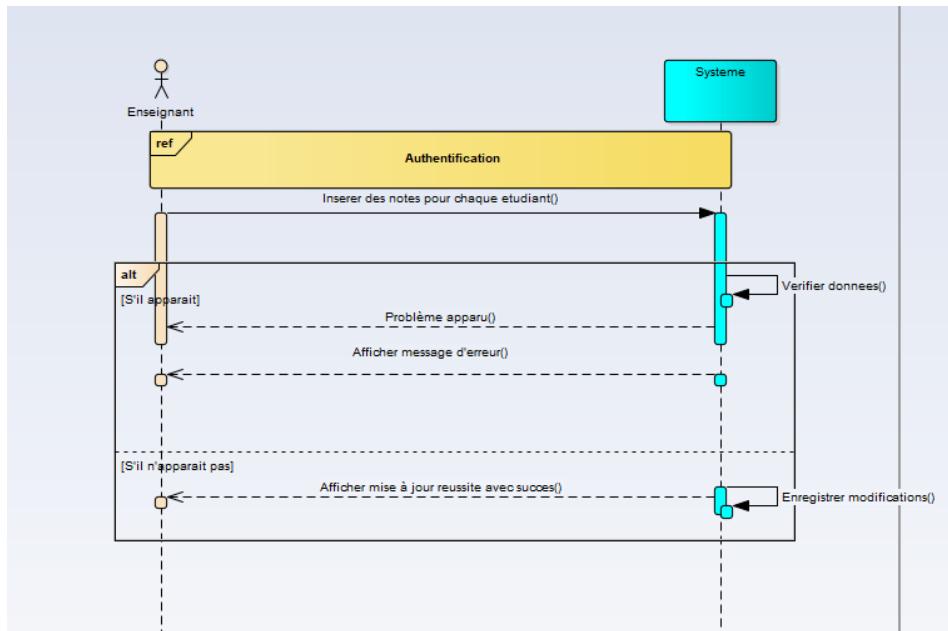


Figure 19 : Diagramme de séquence «Insérer les notes des étudiants»

g. Diagramme de Séquence «consulter les données»

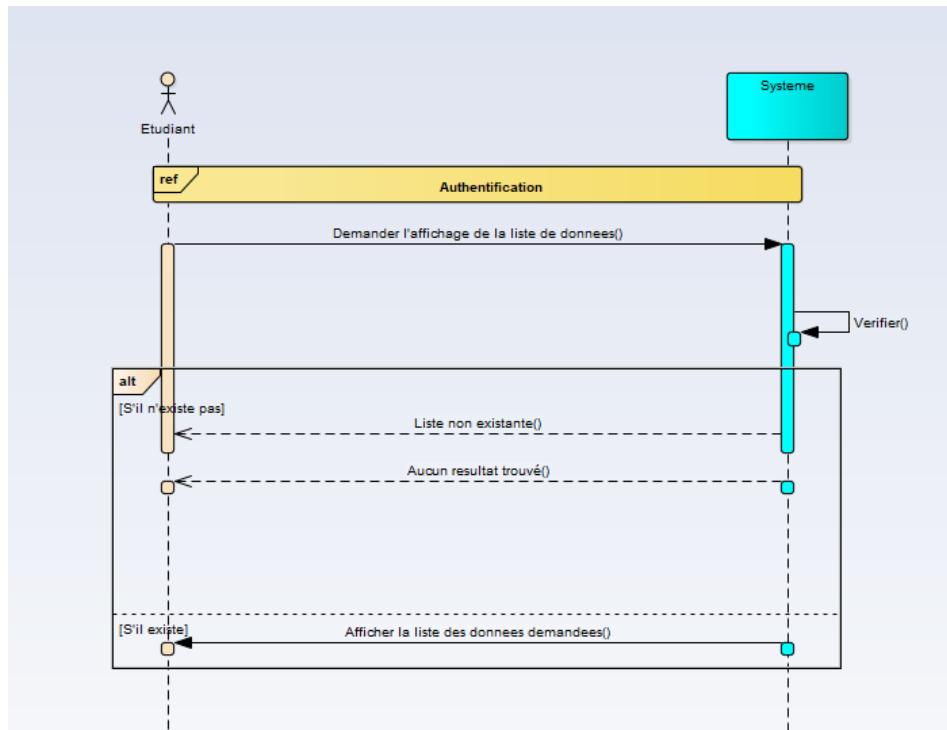


Figure 20 : Diagramme de séquence «Consulter les données»

9. Diagramme d'activités

Dans la phase de conception, les diagrammes d'activités sont particulièrement adaptés à la description des cas d'utilisation. Plus précisément, ils viennent illustrer et consolider la description textuelle des cas d'utilisation. De plus, leur représentation sous forme d'organigrammes les rend facilement intelligibles et beaucoup plus accessibles que les diagrammes d'états-transitions. On se concentre ici sur les activités telles que les voient les acteurs qui collaborent avec le système dans le cadre d'un processus métier.

- **Nœuds d'activités :**

De la gauche vers la droite, on trouve : le nœud représentant une action, qui est une variété de nœud exécutable, un nœud objet, un nœud de décision ou de fusion, un nœud de bifurcation ou d'union, un nœud initial, un nœud final et un nœud final de flux.

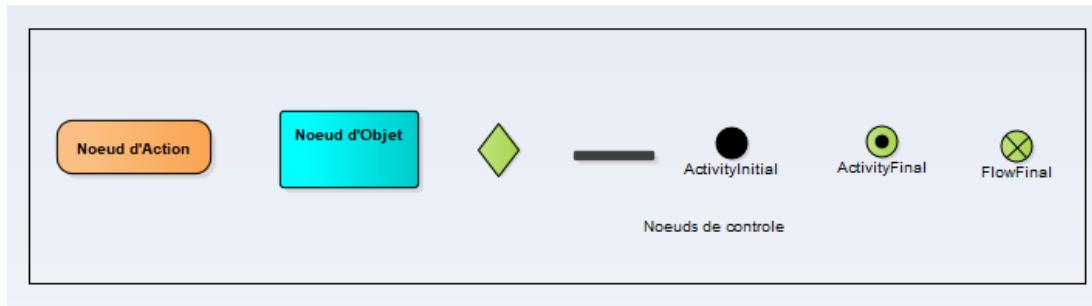


Figure 21 : Représentation Graphique des nœuds d'activités

- **Diagramme d'activité «Affectation des notes des étudiants»**

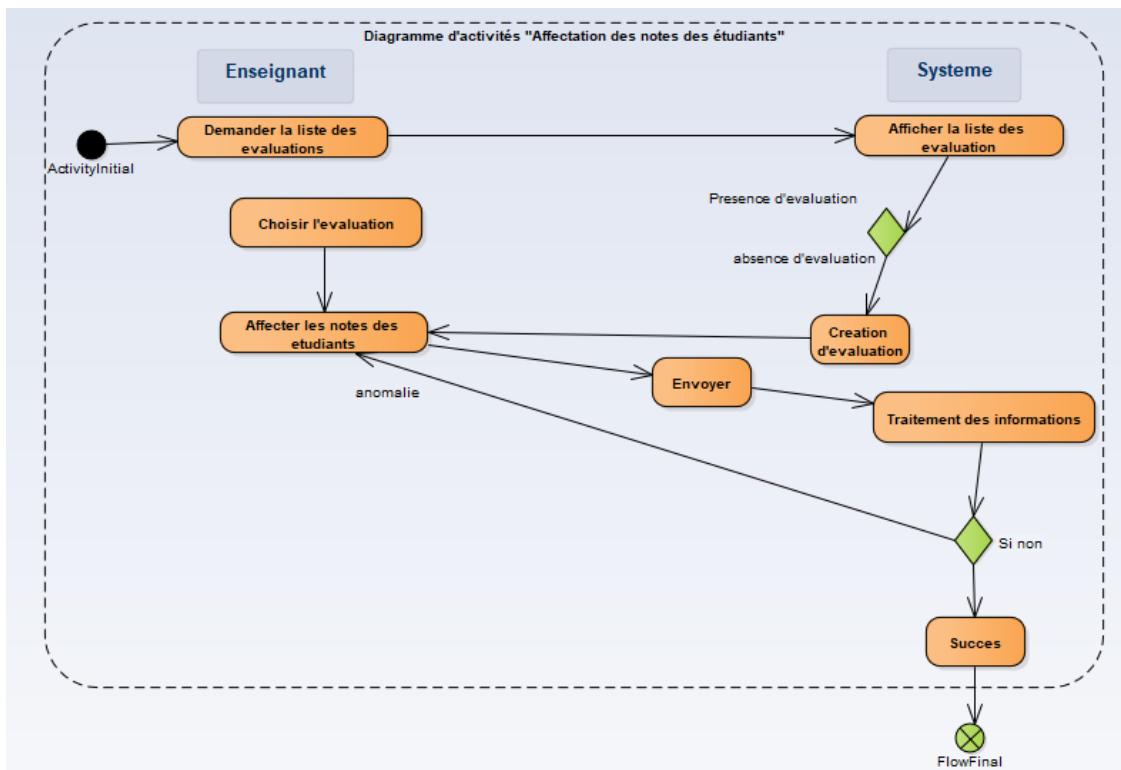


Figure 22 : Diagramme d'activité « Affectation des notes des étudiants »

10. Diagramme de Collaboration

Un diagramme de collaboration est un diagramme d'interactions, représentation simplifiée d'un diagramme de séquence se concentrant sur les échanges de messages entre les objets, et où la chronologie n'intervient que de façon annexe.

Cela consiste en un graphe dont les nœuds sont des objets et les arcs (numérotés selon la chronologie) et les échanges entre ces objets.

a. Diagramme de Collaboration «Créer Etudiant»

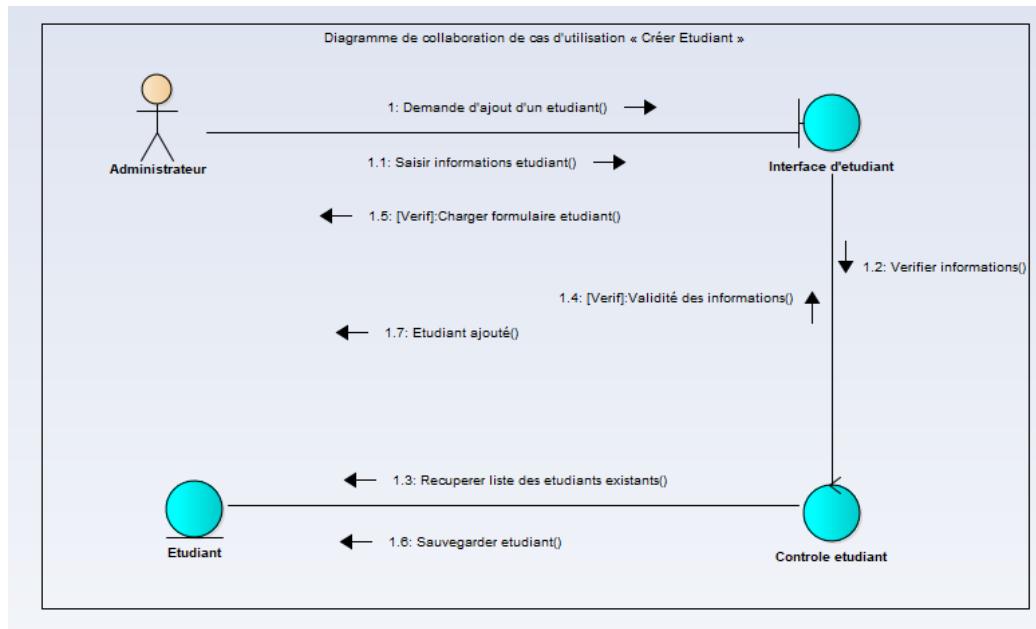


Figure 23: Diagramme de Collaboration «Créer Etudiant»

b. Diagramme de Collaboration «S'inscrire dans un cours donné»

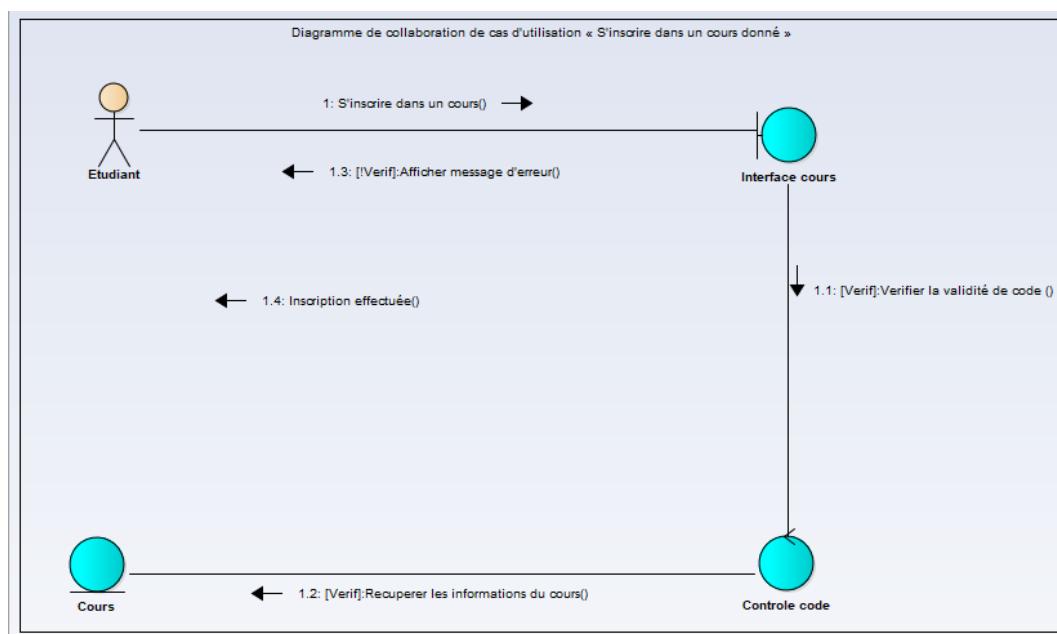


Figure 24 : Diagramme de Collaboration «S'inscrire dans un cours donné»

c. Diagramme de Collaboration «S'authentifier»

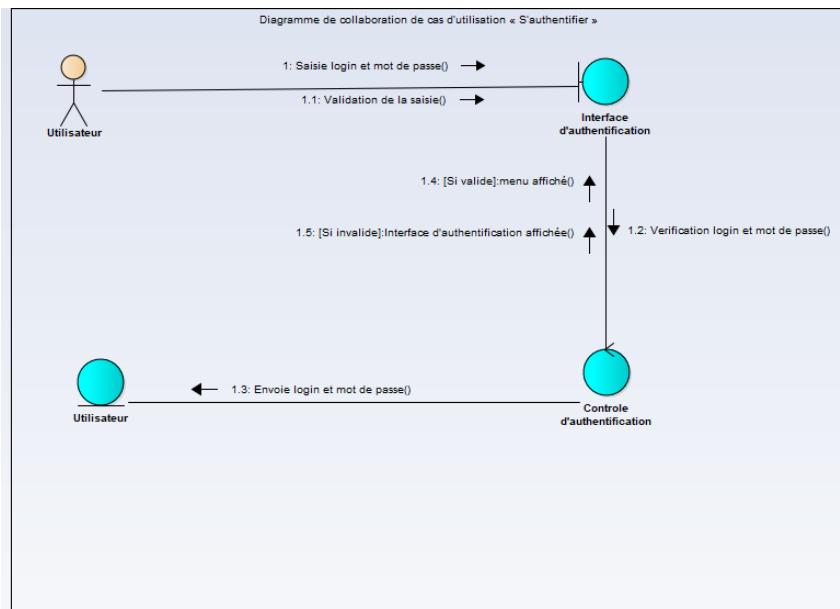


Figure 25 : Diagramme de Collaboration «S'authentifier»

11. Diagramme de Classe

Le diagramme de classe est une description statique du système focalisé sur le concept de classe et d'association. Une classe représente un ensemble d'objets qui possèdent des propriétés similaires et des comportements communs décrivant en terme d'attributs et d'opérations.

Une association consiste à présenter les liens entre les instances de classe. Durant cette section, nous allons présenter les diagrammes de classes entités à notre application.

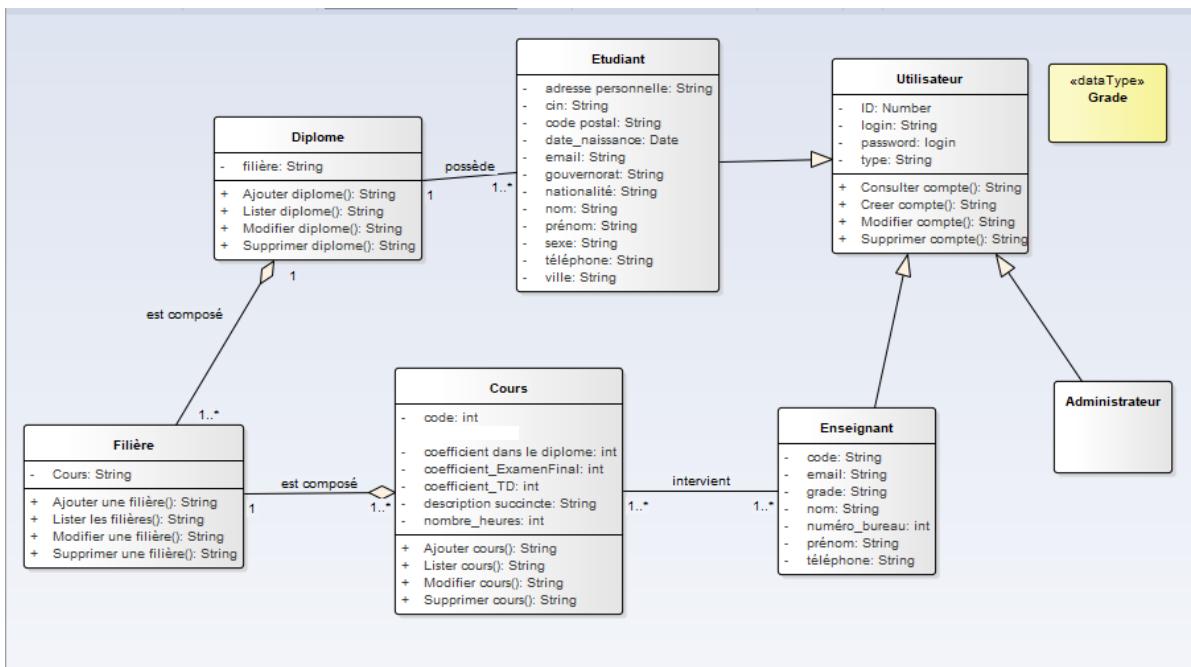


Figure 26 : Diagramme de Classe

Le dictionnaire des données de tables présentées dans le diagramme des classes est donné par le tableaux suivants :

Etudiant

Attribut	Description	Type
Adresse Personnelle	Adresse de l'étudiant	String
CIN	Numéro de carte d'identité nationale de l'étudiant	String
Code postal	Code postal de l'étudiant	String
Date Naissance	Date de naissance de l'étudiant	Date
Email	Adresse email de l'étudiant	String
Gouvernorat	Gouvernorat de l'étudiant	String
Nationalité	nationalité	String
Nom	Nom de l'étudiant	String
Prénom	Prénom de l'étudiant	String
Sexe	Sexe de l'étudiant	String
Téléphone	Téléphone de l'étudiant	String
Ville	Ville de l'étudiant	String

Tableau 25 : dictionnaire de données du table « étudiant »

Enseignant

Attribut	Description	Type
Code	Identifiant de l'enseignant	String
Email	Adresse email de l'enseignant	String
Grade	Grade de l'enseignant	String
Nom	Nom de l'enseignant	String
Prénom	Prénom de l'enseignant	String
Numéro de Bureau	Numéro de Bureau de l'enseignant	Integer
Téléphone	Téléphone de l'enseignant	String

Tableau 26 : dictionnaire de données du table « Enseignant »

Utilisateur

Attribut	Description	Type
ID	Identifiant de l'utilisateur	Number
Login	Nom d'utilisateur	String
Password	Mot de passe de l'utilisateur	Password
Type	Type de l'utilisateur(Enseignant ,Etudiant ,Admin)	String

Tableau 27 : dictionnaire de données du table « Utilisateur »

Cours

Attribut	Description	Type
Code	Identifiant du cours	Integer
Coefficient TD	Coefficient du td du cours	Integer
Coefficient dans le diplôme	Coefficient du cours dans le diplôme	Integer
Coefficient examen final	Coefficient examen final du cours	Integer
Description succincte	Description brève du cours	String
Nombre d'heures	Nombre d'heures pour le cours	Integer

Tableau 28 : dictionnaire de données du table « Cours »

Diplôme

Attribut	Description	Type
Filière	Filière du diplôme	String

Tableau 29 : dictionnaire de données du table « Diplôme »

Filière

Attribut	Description	Type
Cours	Cours de la filière	String

Tableau 30 : dictionnaire de données du table « Filière »

12. Conclusion

Dans ce chapitre, nous avons conçu et documenté le code que nous devons produire. Dans cette phase, toutes les questions concernant la manière de réaliser le système à développer ont été élucidées. Le produit obtenu est un modèle graphique (ensemble de diagrammes) prêt à être codé. Dans le chapitre suivant nous allons étudier en détails les outils et les langages utilisés durant la phase de construction.

Chapitre 3 : Réalisation

1. Introduction

Après avoir achevé l'étape d'analyse et conception de l'application, on va entamer dans ce chapitre la partie réalisation et implémentation dans laquelle on s'assure que le système est prêt pour être exploité par les utilisateurs finaux.

2. Environnement et outils de développement

a. outils de développement

Les principaux outils de développement utilisés pour l'élaboration de notre application sont comme suit :

- **PHPMyAdmin :**

phpMyAdmin est un outil qui fournit une interface facile d'utilisation pour gérer une base MySQL. On peut effectuer à peu près toutes les commandes, cela dépend des droits de l'utilisateur. L'utilisation de phpMyAdmin implique que vous êtes l'administrateur de la base de donnée.



- **Laragon :**

Laragon est un environnement de développement web rapide, flexible, intuitif, productif et puissant qui s'adresse à tous. **Laragon** vous offre tout ce dont vous avez besoin pour créer des applications web modernes.



- **Enterprise Architect :**

Enterprise Architect est un logiciel de **modélisation** et de **conception UML**. Couvrant, par ses fonctionnalités, l'ensemble des étapes du cycle de conception d'application, il est l'un des logiciels de conception et de modélisation les plus reconnus.



- **Adobe Illustrator :**

Adobe Illustrator est un logiciel de **création graphique vectorielle**. Il fait partie de la gamme **Adobe**, peut être utilisé indépendamment ou en complément de **Photoshop**, et offre des outils de dessin vectoriel puissants.



- **Microsoft Word :**

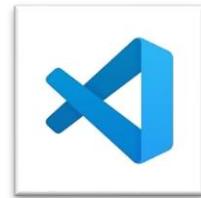
Microsoft Word est un logiciel phare de la suite Bureautique Microsoft Office. Word permet de mettre en page vos documents, de générer des sommaires, de numérotier automatiquement des pages, corriger la grammaire et l'orthographe, de créer des schémas, de faire du publipostage.



- **Visual Studio Code :**

Visual Studio Code est un éditeur de **code** extensible développé par Microsoft.

Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du **code**, les snippets, la refactorisation du **code** et Git intégré.



b. Langages de développement

Les principaux langages de développement utilisés pour l'élaboration de notre application sont comme suit :

- **PHP :**

Le **PHP**, pour **Hypertext Preprocessor**, désigne un langage informatique, ou un langage de script, utilisé principalement pour la conception de sites web dynamiques. Il s'agit d'un langage de programmation sous licence libre qui peut donc être utilisé par n'importe qui de façon totalement gratuite.



- **HTML :**

Le **HyperText Markup Language**, généralement abrégé **HTML** ou dans sa dernière version **HTML5**, est le **langage de balisage** conçu pour représenter les **pages web**.



- **Système de gestion de base de données « MYSQL » :**

MySQL est un système de gestion de bases de données relationnelles (SGBDR) fonctionnant sous Windows et Linux. Il fait partie des logiciels de gestion de base des données les plus utilisées au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft SQL Server.



- **UML :**

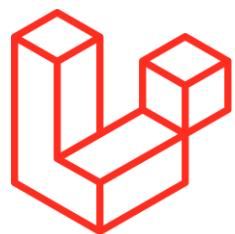
Le **Langage de Modélisation Unifié** est un **langage** de modélisation graphique à base de **pictogrammes** conçu comme une méthode normalisée de visualisation dans les domaines du **développement logiciel** et en **conception orientée objet**.



c. Technologies et Framework adoptées

- **Laravel :**

Laravel est un **Framework web open-source** écrit en **PHP** respectant le principe **modèle-vue-contrôleur** et entièrement développé en **programmation orientée objet**. Il fournit des fonctionnalités en termes de routage de requête, de **mapping objet-relationnel** (un système baptisé Eloquent implémentant **Active Record**), d'authentification, de vue (avec Blade), de **migration de base de données**, de **gestion des exceptions** et de **test unitaire**.



- **Bootstrap :**

Bootstrap est une **collection d'outils** utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de **sites** et **d'applications web**. C'est un ensemble qui contient des codes **HTML** et **CSS**, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions **JavaScript** en option.



3. Architecture Générale de l'application

a. Architecture logiciel du système

Dans ce projet, nous avons fait appel au Framework Laravel. L'un des intérêts de l'intégration de ce Framework est de permettre la mise en place d'une architecture rigoureuse, de manière à garantir la maintenabilité, l'évolutivité et l'exploitabilité de l'application.

La figure ci-dessous montre l'architecture qui a été mise en place dans le cadre de ce projet, cette architecture est largement admise comme la plus efficace et généralisable à n'importe quel projet Web.

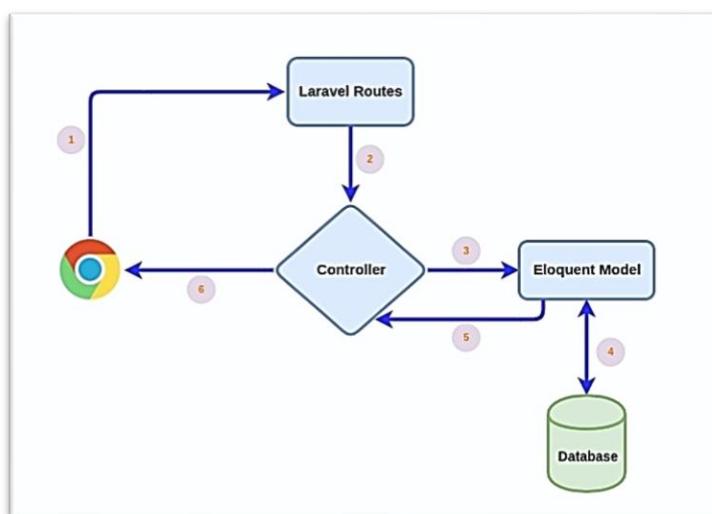


Figure 27 : Architecture du Système

b. Le modèle MVC

Le Framework Laravel fournit une architecture MVC (Model-View-Controller) et des composants prêts à utiliser pour développer des applications Web flexibles et peu couplées. Le modèle MVC permet de séparer les différents aspects de l'application (logique d'entrée, logique métier et logique UI), tout en fournissant un couplage en vrac entre ces éléments.

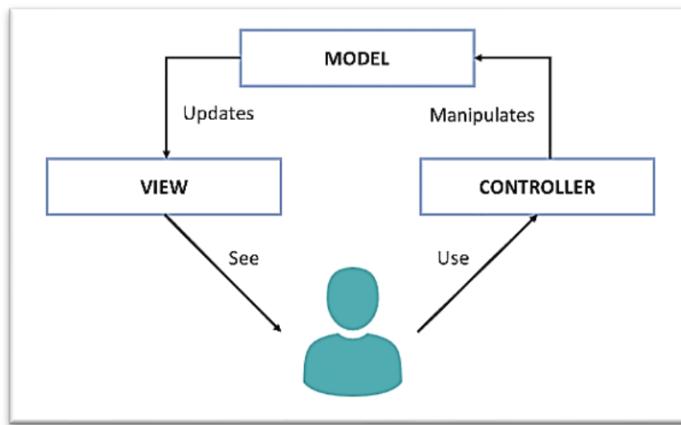


Figure 28 : Pattern MVC

4. Principales interfaces graphiques

La conception des interfaces de l'application est une étape très importante puisque toutes les interactions avec le cœur de l'application passent à travers ces interfaces, on doit alors guider l'utilisateur avec les messages d'erreurs et de notification si besoin, ainsi présenter un système complet. Dans cette partie, nous allons présenter quelques interfaces de l'application, répondant aux recommandations ergonomiques de compatibilité, de guidage, de clarté, d'homogénéité et de souplesse.

a. Authentification

Pour l'authentification on a opté de créer une page d'authentification qui permet aux utilisateurs de l'application de s'identifier par leurs login et leurs mots de passe.

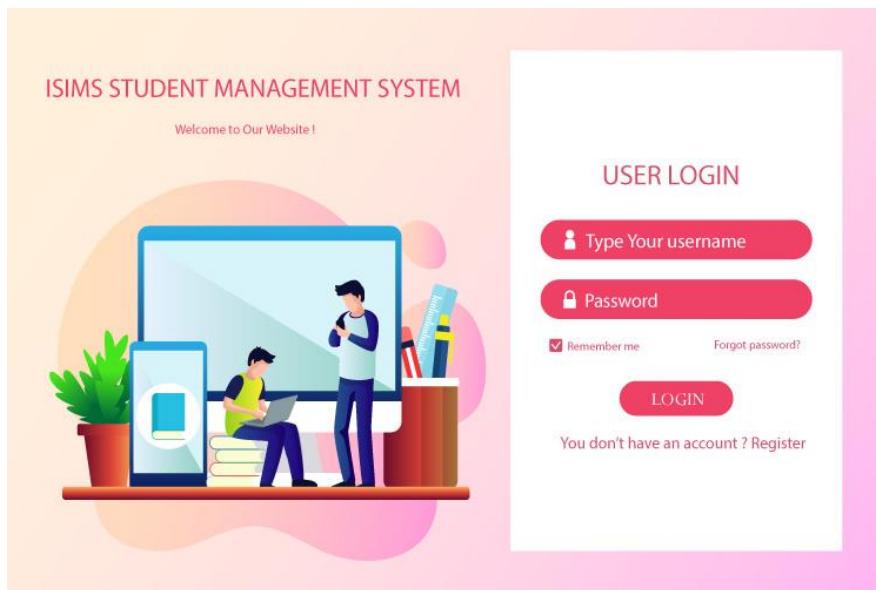


Figure 29 : Page d'authentification

Si les coordonnées de l'utilisateur sont erronées, le système affiche un message d'erreur et l'invite à resaisir ses coordonnées.

Sinon l'utilisateur est redirigé vers la page d'accueil puis son tableau de bord dans lequel on trouve un menu de tous les modules de l'application et chacun de ces modules contient un ensemble des fonctionnalités sous forme des sous menus relatifs à ce module.

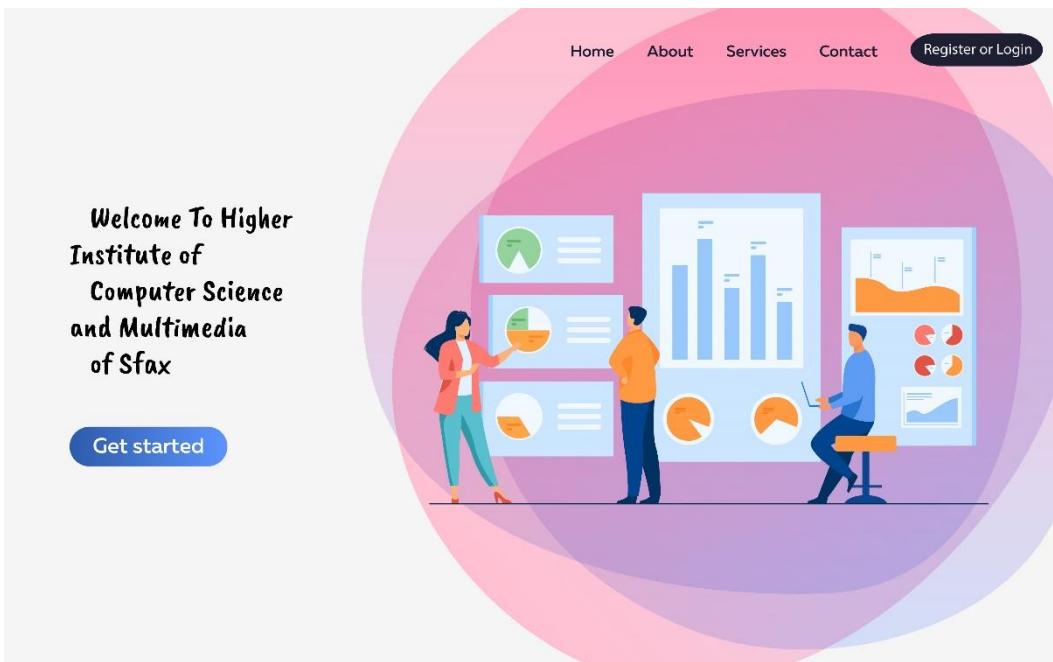


Figure 30 : Page d'accueil



Figure 31 : Tableau de Bord de l'administrateur



Figure 32 : Tableau de Bord de l'étudiant

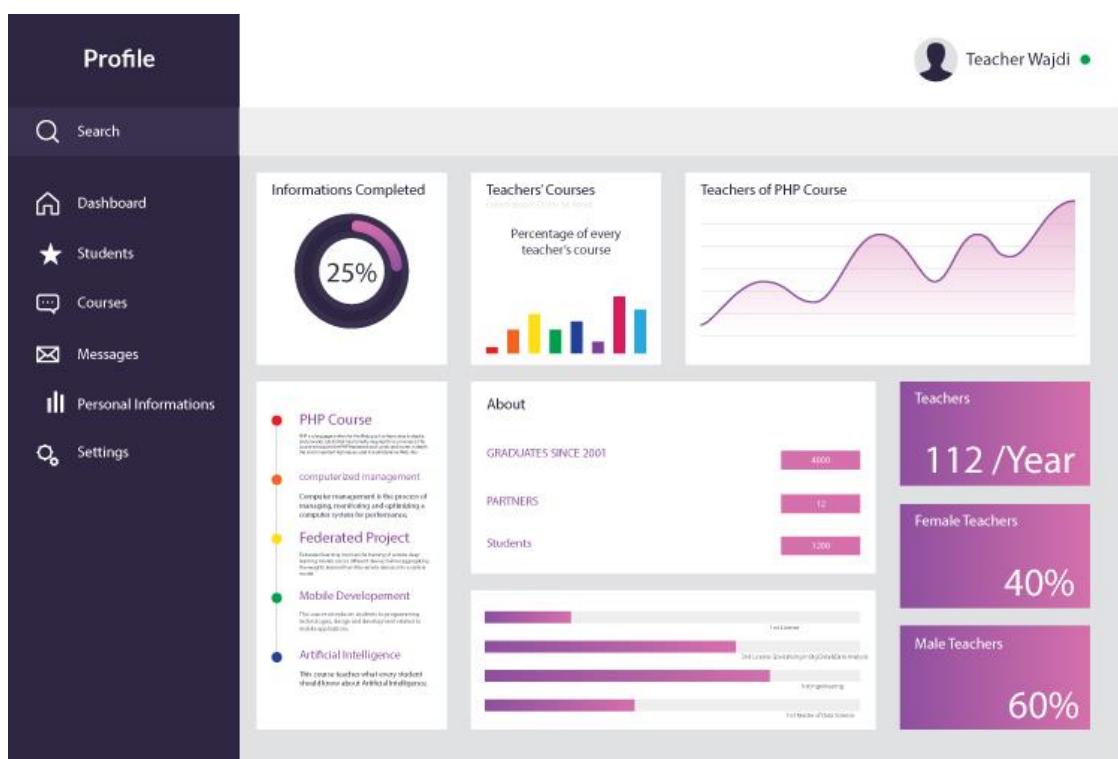


Figure 33 : Tableau de Bord de l'enseignant

b. Inscription

La figure 33 est visualisée lorsque le choix de l'utilisateur, a porté sur le sous menu « Inscription ».

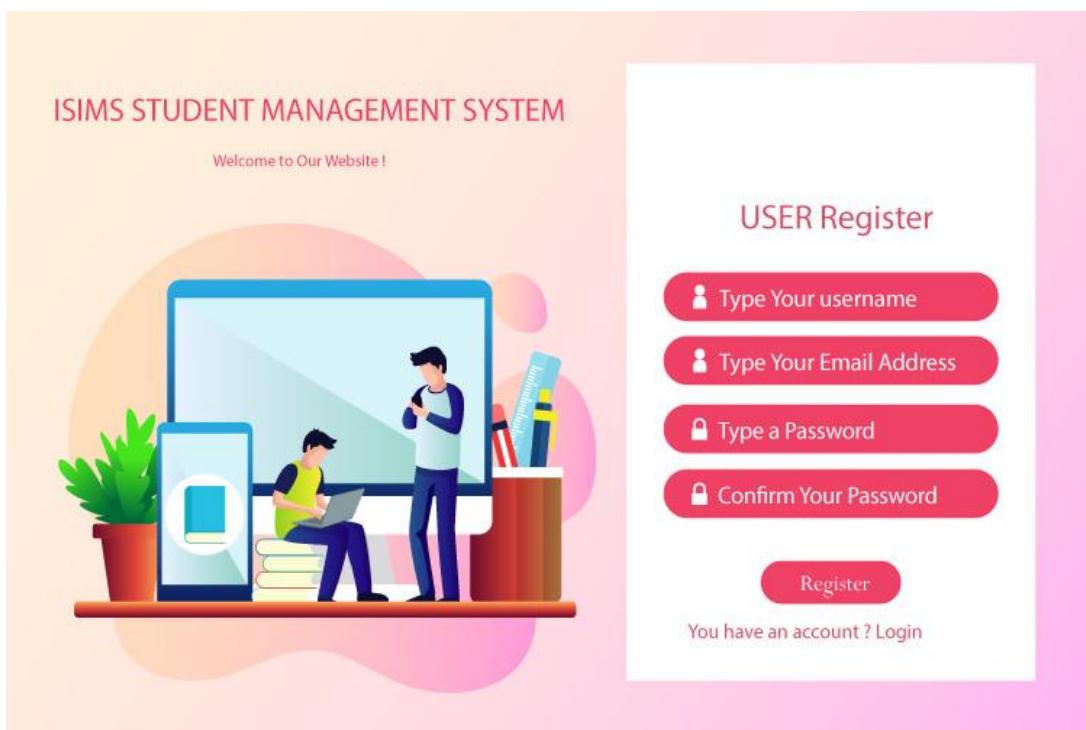


Figure 34 : Page d'inscription

5. Conclusion

A travers ce chapitre, nous avons présenté la réalisation de l'application en justifiant nos choix technologiques, en représentant quelques interfaces graphiques que nous avons jugées les plus importantes

Conclusion Générale

L'objectif de notre projet était de concevoir et une application de gestion administrative des cours dans notre institut.

Le point de départ de la réalisation de ce projet était une récolte d'informations nécessaires pour dresser un état de l'existant, présenter un aperçu sur la problématique ainsi que l'architecture utilisée au sein des réseaux des établissements.

Par la suite, nous nous sommes intéressés à l'analyse et la spécification des besoins qui nous a permis de distinguer les différents acteurs interagissant avec l'application visée.

L'objectif de la partie suivante était la conception détaillée, dans laquelle nous avons fixé la structure globale de l'application.

Le dernier volet de notre projet était la partie réalisation qui a été consacrée à la présentation des outils du travail et les interfaces les plus significatives de notre application.

L'apport de ce travail a été d'une importance très considérable. En effet, il nous a permis : de suivre une méthodologie de travail bien étudiée, d'approfondir nos connaissances dans le monde de développement des applications et de bien nous exercer sur le Framework Laravel.

La réalisation d'un tel projet, nous a permis d'apprendre et de toucher du doigt une partie des divers aspects du métier de développeur et de celui du concepteur.

Webographie

Comment créer une application web/api avec Laravel 8

<https://kholima.com/comment-creer-une-application-web-api-avec-laravel-8/#:~:text=Pour%20installer%20Laravel%20vous%20avez,Comment%20installer%20et%20utiliser%20XAMPP.>

Évoluer vers une architecture MVC en PHP

<https://bpesquet.developpez.com/tutoriels/php/evoluer-architecture-mvc/>

Laravel File Upload: How To Upload File In Laravel

<https://appdividend.com/2018/08/15/laravel-file-upload-example/>

Authentication - Laravel - The PHP Framework For Web Artisans

<https://laravel.com/docs/5.7/authentication>

How to Install Bootstrap 4 in Laravel 7/6 Tutorial and Example

<https://www.techiediaries.com/laravel/how-to-install-bootstrap-in-laravel-6-7-by-example/>

The basic architecture of Laravel applications

https://subscription.packtpub.com/book/web_development/9781788833912/1/ch01lvl1sec11/the-basic-architecture-of-laravel-applications
