



**FACULTÉ DES SCIENCES INFORMATIQUES**

---

**SUJET : Développement d'une application d'aide à la délibération des  
étudiants de l'Université Nouveaux Horizons**

---

Présenté par : **MUSANZI MAVULA Wilfried**

Option : **Informatique de gestion et administration**

Travail de fin de cycle en vue de l'obtention du diplôme  
de licence en Sciences Informatiques

**ANNÉE ACADÉMIQUE : 2022-2023**



## **FACULTÉS DES SCIENCES INFORMATIQUES.**

---

### **Développement d'une application d'aide à la délibération des étudiants de l'Université Nouveaux Horizons.**

---

Travail de fin d'études présenté en vue de l'obtention du grade de licencié en sciences  
informatiques

Par : **MUSANZI MAVULA Wilfried**

Directeur : **Prof Saint-Jean Djungu**

Encadreur : **Ass. Jonathan Kabemba**

Option : **Informatique de gestion et administration**

**ANNÉE ACADÉMIQUE : 2022-2023**

## ÉPIGRAPHE

---

Le développement logiciel est une activité intellectuelle, créative et complexe qui nécessite une approche rigoureuse, méthodique et itérative pour produire des logiciels de qualité.

---

Steve McConnell, auteur de *Code Complete :  
A Practical Handbook of Software  
Construction*

## DÉDICACE

---

À ma mère, Lydie Kimbassi Mayika.

## RÉMERCIEMENTSSSSS

---

Je tiens tout d'abord à exprimer mes plus sincères remerciements à toutes les personnes qui ont contribué, de près ou de loin, à l'élaboration de ce mémoire. Ce travail n'est pas le fruit d'un effort individuel, mais plutôt l'aboutissement d'une construction collective, à laquelle chacun a apporté une pierre angulaire.

Je voudrais en premier lieu remercier le Très-Haut pour la vie et la grâce qu'Il ne cesse de m'accorder. Ensuite, je tiens à exprimer ma profonde gratitude envers mon directeur de recherche, le Docteur Saint Jean Djungu, pour son accompagnement, son soutien, ses orientations et ses apports inestimables qui ont grandement contribué à la réalisation de ce travail.

Je souhaite également exprimer ma reconnaissance envers mon encadreur, l'Assistant Jonathan Kabemba, pour sa bienveillance, son sérieux et ses multiples apports, qui ont été d'une aide inestimable pour moi.

Je tiens à remercier chaleureusement l'ensemble de ma promotion, ainsi que toutes les personnes qui ont contribué à la réalisation de ce travail en me prodiguant des conseils, des guides, des orientations et surtout des critiques constructives.

Enfin, je tiens à exprimer mes remerciements les plus sincères à ma mère, Yvette Lumbu, pour son amour indéfectible, sa bienveillance et sa confiance inébranlable envers ma personne. Je souhaite également remercier mon père, Francis Simbi, pour le sacrifice consenti et son amour paternel, ainsi que pour sa rigueur qui a porté ses fruits aujourd'hui.

## RÉSUMÉ

---

L'Université Nouveaux Horizons est une institution moderne qui s'efforce constamment d'améliorer ses processus, notamment en ce qui concerne la délibération des étudiants. Dans ce mémoire, nous présentons une solution informatique sous forme d'application pour aider l'université dans ce processus.

La délibération des étudiants est un processus complexe qui implique l'analyse de plusieurs critères, notamment leur performance académique. L'application que nous proposons prend en compte cet aspect en permettant de suivre les données des étudiants, telles que leurs cours en complément et leurs relevés de notes. Elle permet également de calculer les pourcentages et de générer une grille de délibération ainsi que des relevés de notes personnalisés pour chaque étudiant.

L'application facilite la prise de décision du jury en fournissant des informations pertinentes et en simplifiant le processus de délibération. Elle permet également à l'université de gagner en temps et en efficacité dans la gestion des données des étudiants.

Notre solution est le fruit d'une analyse minutieuse des pratiques existantes et des standards en vigueur dans d'autres universités. Elle répond ainsi de manière adéquate aux besoins spécifiques de l'Université Nouveaux Horizons.

Nous avons conçu notre application comme une solution extensible, grâce à une API Rest qui permet une communication aisée avec d'autres services tiers tels que Moodle ou Google Classroom. Cette flexibilité assure une certaine indépendance et une évolutivité de la solution.

En somme, notre application offre une solution efficace et adaptée pour la délibération des étudiants, qui permettra à l'Université Nouveaux Horizons de moderniser ses processus et de gagner en efficacité tout en assurant un suivi personnalisé des étudiants.

## ABSTRACT

---

New Horizons University is a modern institution that is constantly striving to improve its processes, particularly with regard to student deliberation. In this thesis, we present an IT solution in the form of an application to help the university in this process.

Student deliberation is a complex process involving the analysis of several criteria, including academic performance. The application we are proposing takes this into account by tracking student data, such as course completions and transcripts. It also calculates percentages and generates a deliberation grid as well as personalized transcripts for each student.

The application facilitates jury decision-making by providing relevant information and simplifying the deliberation process. It also enables the university to gain time and efficiency in managing student data.

Our solution is the result of a careful analysis of existing practices and standards at other universities. It is therefore ideally suited to the specific needs of New Horizons University.

We have designed our application as an extensible solution, thanks to an API Rest that enables easy communication with other third-party services such as Moodle or Google Classroom. This flexibility ensures the solution's independence and scalability.

All in all, our application offers an effective and adapted solution for student deliberation, which will enable New Horizons University to modernize its processes and gain in efficiency while ensuring personalized student follow-up.

## TABLE DES MATIÈRES

---

épigraphe	2
dépigraphe	3
Rémerciements	4
Résumé	5
Abstract	6
0 Introduction générale	12
0.1 Aperçu générale . . . . .	12
0.2 Contexte et motivation . . . . .	12
0.3 Problématique . . . . .	12
0.4 Méthodes et techniques . . . . .	13
0.5 Etat de la question . . . . .	14
0.6 Hypothèse . . . . .	14
0.7 Subdivision du travail . . . . .	14
1 Généralités	15
1.1 Délibération . . . . .	15
1.2 Logiciel . . . . .	15
1.3 Web . . . . .	15
1.4 API . . . . .	15
1.5 Méthodes de développement logiciel . . . . .	16
1.5.1 Méthodes de développement traditionnelles . . . . .	16
1.5.2 Méthodes de développement agiles . . . . .	16
1.6 La modélisation . . . . .	16
1.7 UML (Unified Modeling Language) . . . . .	17
1.8 Base de données . . . . .	17
2 Cadre du travail	18
2.1 Présentation de l'entreprise . . . . .	18
2.2 Objectifs et vision de l'entreprise . . . . .	18
3 Analyse	19
3.1 Modèles . . . . .	19
3.1.1 Diagramme de classes . . . . .	19
3.1.2 Diagramme d'objets . . . . .	20
3.1.3 Diagramme de cas d'utilisation . . . . .	21
3.1.4 Diagramme de séquence . . . . .	22
4 Implémentation	24
4.1 Choix techniques et motivations . . . . .	24
4.2 Présentation des résultats . . . . .	26
4.2.1 Présentation de l'API Rest . . . . .	26



4.2.2	Présentation de l'interface utilisateur . . . . .	34
	Conclusion	36

## TABLE DES FIGURES

---

Figure 1	Diagramme de classes . . . . .	19
Figure 2	Diagramme d'objets . . . . .	20
Figure 3	Cas d'utilisation : Gestion des ressources . . . . .	21
Figure 4	Cas d'utilisation : Création des PDFs(Relevés) . . . . .	22
Figure 5	Cas d'utilisation : Envoyer le pdf par mail . . . . .	23
Figure 6	Diagramme de séquence . . . . .	23
Figure 7	Page d'authentification . . . . .	34
Figure 8	Tableau de bord . . . . .	35
Figure 9	Page des utilisateurs . . . . .	35
Figure 10	Page des cotes . . . . .	36
Figure 11	Page des relevés . . . . .	36

## LISTE DES TABLEAUX

---

Table 1	Tableau des routes de l'authentification . . . . .	27
Table 2	Tableau des routes de la gestion des facultés . . . . .	27
Table 3	Tableau des routes de la gestion des filières . . . . .	28
Table 4	Tableau des routes de la gestion des cours . . . . .	29
Table 5	Tableau des routes de la gestion des étudiants . . . . .	30
Table 6	Tableau des routes de la gestion des cotes . . . . .	31
Table 7	Tableau des routes de la gestion des délibérations . . . . .	32
Table 8	Tableau des routes de la gestion des utilisateurs . . . . .	33
Table 9	Tableau des routes de la gestion des rôles . . . . .	34

## INTRODUCTION GÉNÉRALE

---

### 0.1 APERÇU GÉNÉRALE

Ce présent travail consiste en la conception et le développement d'une application web d'aide à la délibération des étudiants à l'université Nouveaux Horizons. L'objectif est de faciliter la gestion des données des étudiants, la publication des résultats et de réduire le circuit de circulation des informations.

Nous parlons d'aide à la délibération des étudiants car ce processus est complexe et prend en compte plusieurs aspects tels que la conduite, les notes, la présence, etc. Cependant, dans ce travail, nous ne prenons en compte qu'un seul aspect qui est la performance académique des étudiants.

Pour ce faire, nous avons commencé par analyser la problématique et les besoins de l'université Nouveaux Horizons en matière de délibération des étudiants, afin de proposer une solution adéquate.

Nous avons ensuite effectué une étude comparative des applications existantes et des technologies utilisées pour leur développement. Nous nous sommes inspirés de leurs fonctionnalités et de leurs technologies. Nous nous sommes également intéressés à la manière dont les autres universités gèrent cette problématique.

Après une étude minutieuse des applications existantes, nous avons procédé à la conception de notre application en nous basant sur les besoins de l'université Nouveaux Horizons et en étendant les fonctionnalités des applications existantes pour ne pas réinventer la roue.

Ensuite, nous avons procédé au développement de notre application en utilisant les technologies que nous avons jugées les plus adaptées.

### 0.2 CONTEXTE ET MOTIVATION

Ce projet est axé sur la délibération des étudiants à l'université Nouveaux Horizons. Comme mentionné ci-dessus, la délibération est un processus qui prend en compte plusieurs aspects, et nous nous sommes focalisés sur un seul aspect qui est la performance académique de ces derniers.

Ce travail trouve tout son intérêt dans le fait que l'université Nouveaux Horizons est une institution moderne. Nous voulons donc apporter notre contribution à son essor en lui fournissant un outil qui répondra à ses besoins et qui pourra s'intégrer facilement dans son système d'information.

Nous sommes également motivés par le fait de pouvoir fournir une base qui pourra être étendue pour obtenir un système de gestion des données des étudiants plus complet et plus efficace.

### 0.3 PROBLÉMATIQUE

Une meilleure solution est celle qui répond à un besoin réel. Actuellement, l'université Nouveaux Horizons est confrontée à un problème de gestion des données des étudiants et de publication des résultats.

Voici un bref aperçu du circuit de circulation des informations :

- Après les examens, le professeur ou le chargé de cours envoie soit les notes annuelles, soit les notes de tous les travaux ainsi que l'examen au décanat de l'université, en copie au doyen de la faculté.
- Le décanat, après réception, transmet les notes reçues soit au format pdf, soit Excel (xls), soit manuscrit au président du jury.
- Le jury calcule les moyennes et, après délibération, communique les résultats aux étudiants.

Vous conviendrez avec nous que le circuit de circulation des informations est assez long. De plus, il est difficile de gérer les données des étudiants de manière efficace et la publication des résultats n'est pas aussi évidente qu'elle devrait l'être.

Par ailleurs, il est difficile de suivre l'évolution des étudiants d'une année à l'autre.

Les dites données peuvent être :

- les cours en compléments,
- le(s) relevé(s) de chaque année.
- etc.

## 0.4 MÉTHODES ET TECHNIQUES

Pour la réalisation de ce travail nous avons utilisé plusieurs méthodes et techniques qui nous ont permis de mener à bien notre projet en nous permettant de réunir les informations nécessaires en vue d'en tirer des conclusions et de proposer la solution la mieux adaptée.

### *Méthodes*

1. Méthode analytico-déductive : Nous avons analysé la problématique évoquée ci-dessus en partant des faits concrets pour aboutir à une conclusion générale.
2. Méthode descriptive : Nous avons décrit notre problématique de manière précise et objective.
3. Méthode comparative : Nous avons comparé la manière dont la problématique est gérée ailleurs pour en dégager les similitudes et les différences.

### *Techniques*

Nous avons utilisé la technique documentaire, plus précisément la technique de la recherche bibliographique.

Nous avons consulté des ouvrages, des articles, des documents et des sites web pour obtenir des informations sur les applications existantes et les technologies utilisées pour leur développement. Nous avons également consulté des documents sur les méthodes de conception et de développement d'applications web.

Nous avons également utilisé la technique de l'entretien pour recueillir des informations sur les besoins du corps académique de l'université Nouveaux Horizons et sur les fonctionnalités qu'ils souhaiteraient avoir dans une application d'aide à la délibération. Nous avons également cherché à savoir comment les universités sœurs gèrent cette problématique.

## 0.5 ETAT DE LA QUESTION

Étant dans une université, la délibération est un processus commun. Nous avons pensé que d'autres universités ont forcément été confrontées à la même problématique que nous.

Nous nous sommes donc intéressés à la manière dont les autres universités gèrent la délibération des étudiants et les outils qu'elles utilisent.

Nous avons trouvé des concepts qui nous ont aidé à mieux comprendre la problématique et à mieux cerner les besoins. Nous ne manquerons pas de les mentionner dans la suite de ce travail.

## 0.6 HYPOTHÈSE

Dans le but d'apporter une solution efficace, flexible et solide à la problématique posée, nous avons défini l'objectif de fournir une application web qui :

- permettra de générer une grille de délibération,,
- permettra de faciliter la publication des résultats,
- permettra de faciliter le suivi des données des étudiants,
- pourra s'intégrer facilement dans le système d'information de l'université, ainsi qu'aux autres applications existantes telles que Moodle, Google Classroom, etc.

## 0.7 SUBDIVISION DU TRAVAIL

Ce travail est subdivisé en quatre chapitres. Le premier chapitre est consacré aux généralités, où nous avons défini quelques concepts clés pour une meilleure compréhension du travail. Le deuxième chapitre est consacré au cadre du travail, où nous avons présentons l'Université Nouveaux Horizons. Le troisième chapitre est consacré à l'analyse, où nous présentons les différents diagrammes, enfin, le quatrième chapitre est consacré aux choix techniques et à la présentation des résultats.

## GÉNÉRALITÉS

---

Dans cette partie du travail, nous allons définir quelques termes clés qui seront utilisés tout au long du document afin d'améliorer la compréhension de celui-ci. Cette étape est essentielle pour garantir que les lecteurs puissent comprendre facilement les concepts et les idées présentés dans la thèse. En effet, une bonne définition des termes clés permet de clarifier les concepts importants et de faciliter la lecture et la compréhension du document. Par conséquent, cette partie de la thèse est cruciale pour assurer la clarté et la précision de l'ensemble du travail.

### 1.1 DÉLIBÉRATION

La délibération est une confrontation de vue visant à trancher un problème ou un choix difficile par l'adoption d'un jugement ou d'une décision réfléchie [9][10]. Elle peut être effectuée par un individu seul, mais aussi par un groupe d'individus ou une collectivité. Elle débouche en général sur une décision ou un choix.

### 1.2 LOGICIEL

Selon la définition de l'IEEE, un logiciel est une collection de programmes informatiques, de données et de documentation associée qui fournit des instructions pour l'opération d'un ordinateur ou d'un système informatique, ainsi que pour la résolution de problèmes spécifiques [7]. Les logiciels peuvent être classés en différentes catégories, telles que les systèmes d'exploitation, les logiciels applicatifs, les logiciels standards, les logiciels spécifiques et les logiciels libres [13]. Ces classifications sont utiles pour comprendre les différentes fonctionnalités et caractéristiques des logiciels, ainsi que pour choisir le type de logiciel le plus adapté aux besoins spécifiques d'un projet.

### 1.3 WEB

Le Web, également connu sous le nom de World Wide Web (WWW), est un système d'information en ligne qui permet l'accès à des documents hypertexte reliés entre eux et stockés sur des serveurs à travers le monde [1]. Le Web est accessible via internet à l'aide d'un navigateur web.

### 1.4 API

Une interface de programmation d'application ou interface de programmation applicative, souvent désignée par le terme API pour « Application Programming Interface », est un ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels [12][3].

## 1.5 MÉTHODES DE DÉVELOPPEMENT LOGICIEL

Une méthode de développement logiciel est un ensemble de principes, de pratiques et de processus qui sont utilisés pour planifier, concevoir, développer, tester et maintenir des logiciels. Ces méthodes fournissent des approches structurées et organisées pour la gestion de projets de développement de logiciels. Les méthodes de développement logiciel peuvent être utilisées pour améliorer la qualité du logiciel, augmenter la productivité, réduire les coûts et respecter les délais de livraison.

On en retient deux grandes familles : les méthodes de développement traditionnelles et les méthodes de développement agiles.

### 1.5.1 *Méthodes de développement traditionnelles*

Les méthodes de développement de logiciels traditionnelles suivent un processus de développement séquentiel qui se déroule généralement en phases distinctes, comme la planification, la conception, la mise en oeuvre, les tests et la maintenance. Le modèle en cascade (Waterfall) est un exemple de méthode de développement traditionnelle. Dans ce modèle, chaque phase doit être terminée avant que la suivante ne commence [11]. Les méthodes traditionnelles sont souvent utilisées dans des projets à grande échelle avec des exigences et des spécifications clairement définies.

Voici quelques exemples de méthodes de développement traditionnelles :

- Modèle en cascade (Waterfall)
- Modèle en V
- Modèle en spirale
- etc.

### 1.5.2 *Méthodes de développement agiles*

Les méthodes de développement de logiciels agiles sont basées sur des processus itératifs et incrémentaux, où les équipes travaillent de manière collaborative et flexible pour répondre aux besoins changeants du client [8]. Les méthodes agiles mettent l'accent sur la livraison continue de fonctionnalités fonctionnelles, plutôt que sur la planification exhaustive et la documentation. Le développement Agile est souvent utilisé pour des projets avec des spécifications moins clairement définies ou pour des projets nécessitant une grande flexibilité.

Voici quelques exemples de méthodes de développement Agile :

- Scrum
- Kanban
- Extreme Programming (XP)

## 1.6 LA MODÉLISATION

La modélisation est le processus de représentation d'un système, d'un processus, d'un objet ou d'un concept sous forme de modèle [5].



Un modèle est une représentation abstraite et simplifiée d'un système réel qui peut être utilisée pour comprendre, analyser, prédire et concevoir le système.

## 1.7 UML (UNIFIED MODELING LANGUAGE)

Le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (UML), 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 [4][2].

L'UML est une synthèse de langages de modélisation objet antérieurs : Booch, OMT, OOSE. Principalement issu des travaux de Grady Booch, James Rumbaugh et Ivar Jacobson, UML est à présent un standard adopté par l'Object Management Group (OMG). UML 1.0 a été normalisé en janvier 1997 ; UML 2.0 a été adopté par l'OMG en juillet 2005. La dernière version de la spécification validée par l'OMG est UML 2.5.1 (2017).

Voici les principales notations graphiques utilisées dans UML :

## 1.8 BASE DE DONNÉES

Une base de données est un outil utilisé pour stocker et récupérer des données et des informations structurées ou non structurées liées à un thème ou à une activité[6]. Les données peuvent être stockées sous une forme très structurée, comme dans les bases de données relationnelles, ou sous une forme moins structurée, comme dans les bases de données NoSQL. Les bases de données peuvent être centralisées ou réparties sur plusieurs machines. Un système de gestion de base de données (SGBD) est utilisé pour manipuler la base de données et gérer l'accès à son contenu. Les bases de données relationnelles sont organisées en tables, tandis que les bases de données NoSQL s'écartent du modèle relationnel en parlant des collections. Les bases de données NoSQL sont une famille de SGBD qui s'écartent du modèle relationnel traditionnel et sont principalement utilisées pour les systèmes distribués à grande échelle.

## CADRE DU TRAVAIL

---

### 2.1 PRÉSENTATION DE L'ENTREPRISE

Le désir et la volonté d'offrir à la jeunesse une éducation de qualité sont les grandes motivations à l'origine de la création de l'université Nouveaux Horizons. À cette immense volonté viennent se surajouter deux faits du vécu social :

- L'incitation et la suggestion des tiers : Pendant plusieurs années, de nombreuses personnes ont incité les promoteurs de l'université Nouveaux Horizons à créer une institution académique dans le prolongement de l'enseignement de fin d'études à l'École géodésique d'Or. Il est fait mention des parents des élèves de ladite école, des amis, et autres.
- Le regard de l'élève tourné vers l'étranger pour la suite des études après le cycle de l'enseignement secondaire : Le fait est que de nombreux élèves de la République Démocratique du Congo, en fin de leurs études secondaires, n'aspirent qu'à s'expatrier pour aller s'inscrire sous d'autres cieux, pensant que l'ailleurs est meilleur que chez soi, dans des universités supposées de meilleure qualité. Ainsi, il est aisé de comprendre la critique souvent formulée à l'endroit des titres académiques décernés par les universités en République Démocratique du Congo.

Par ailleurs, un autre fait ayant incité les initiateurs à créer l'université Nouveaux Horizons est en quelque sorte ce qu'on pourrait appeler "l'héritage colonial" légué aux universités congolaises dans leur ensemble, à savoir la formation de demandeurs d'emploi plutôt que de créateurs d'emplois.

### 2.2 OBJECTIFS ET VISION DE L'ENTREPRISE

L'université Nouveaux Horizons a pour but de dispenser des enseignements dans différentes filières, en mettant l'accent sur l'esprit d'entreprise (entrepreneuriat), et de former des hommes et des femmes capables de participer efficacement au développement de la communauté, davantage comme des créateurs que des demandeurs d'emplois. L'UNH assure un enseignement de qualité et de niveau international grâce à des compétences requises, judicieusement recrutées pour cette fin.

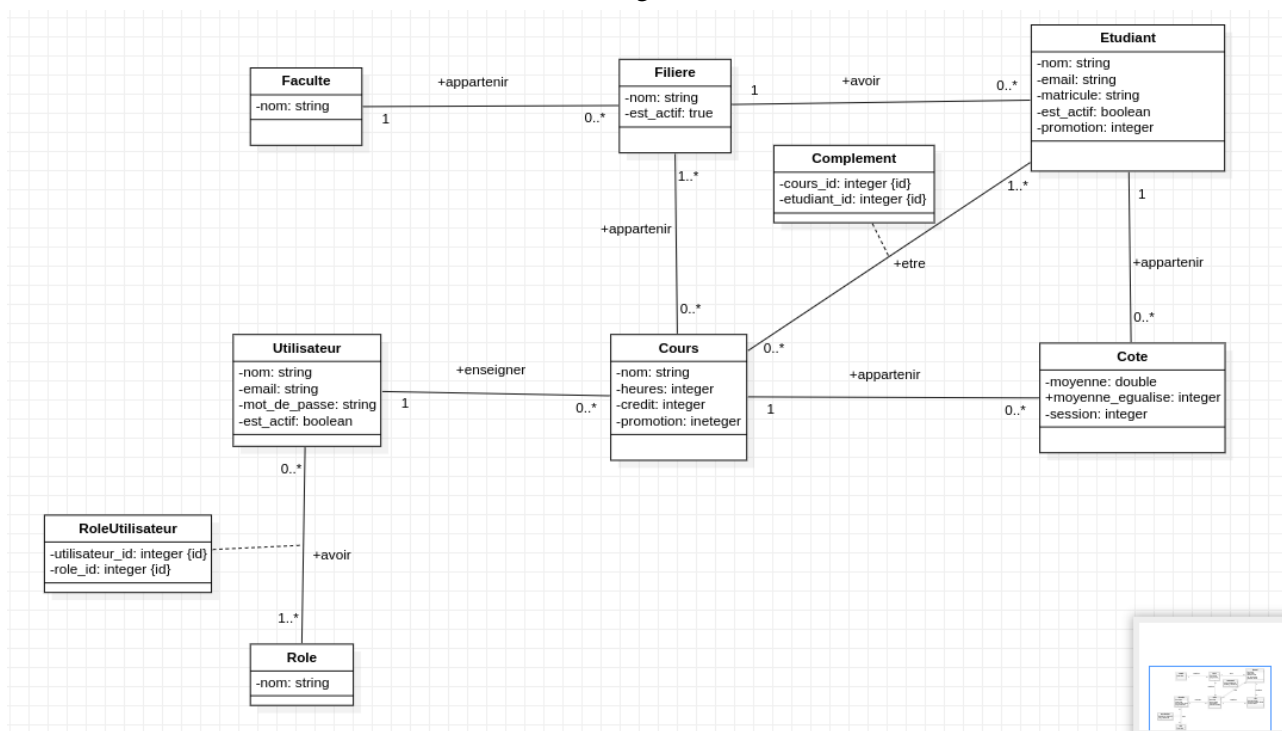
L'université Nouveaux Horizons se veut être une université de standard international, qui contribue à l'avancement du savoir et rayonne dans son milieu et au-delà.

## ANALYSE

## 3.1 MODÈLES

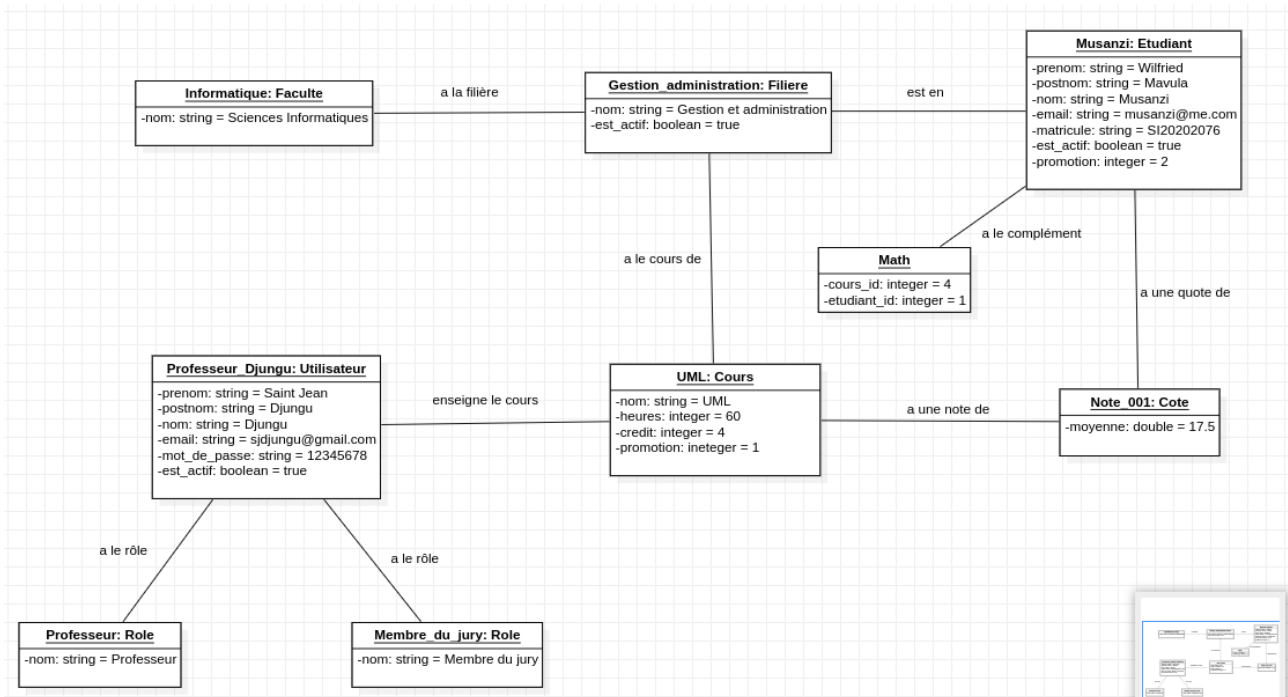
3.1.1 *Diagramme de classes*

FIGURE 1 – Diagramme de classes



## 3.1.2 Diagramme d'objets

FIGURE 2 – Diagramme d'objets



## 3.1.3 Diagramme de cas d'utilisation

FIGURE 3 – Cas d'utilisation : Gestion des ressources

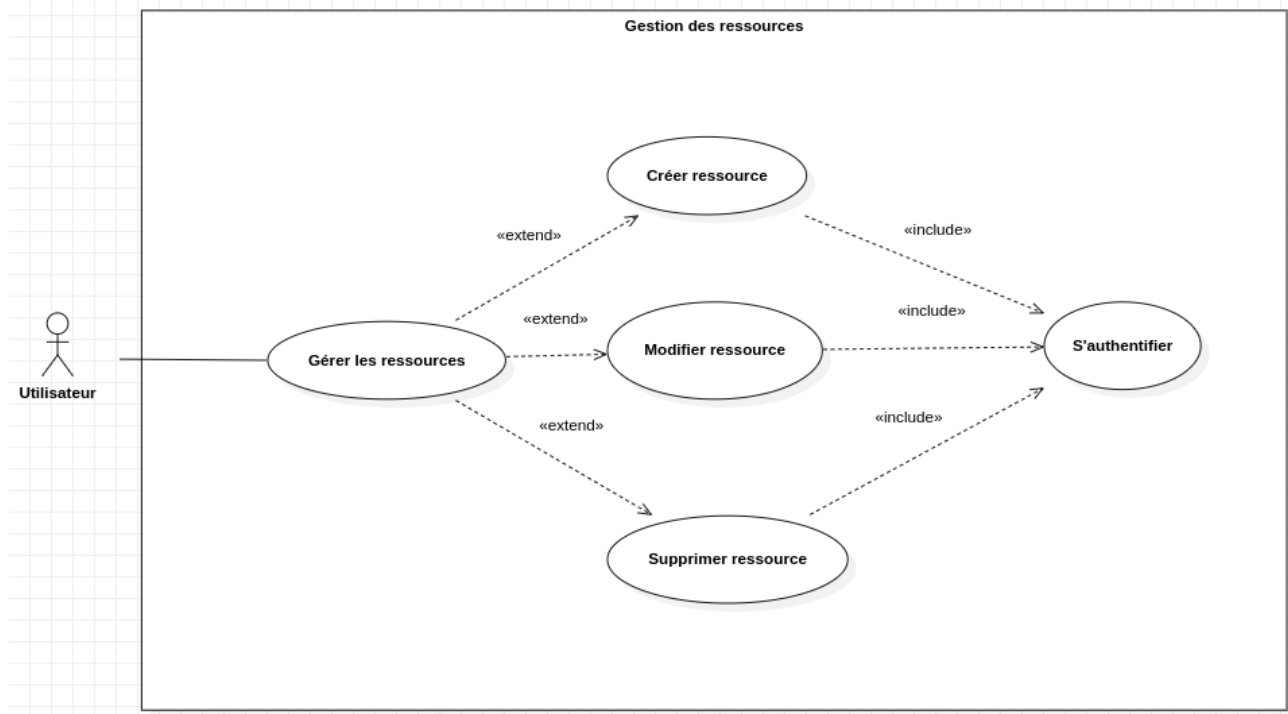
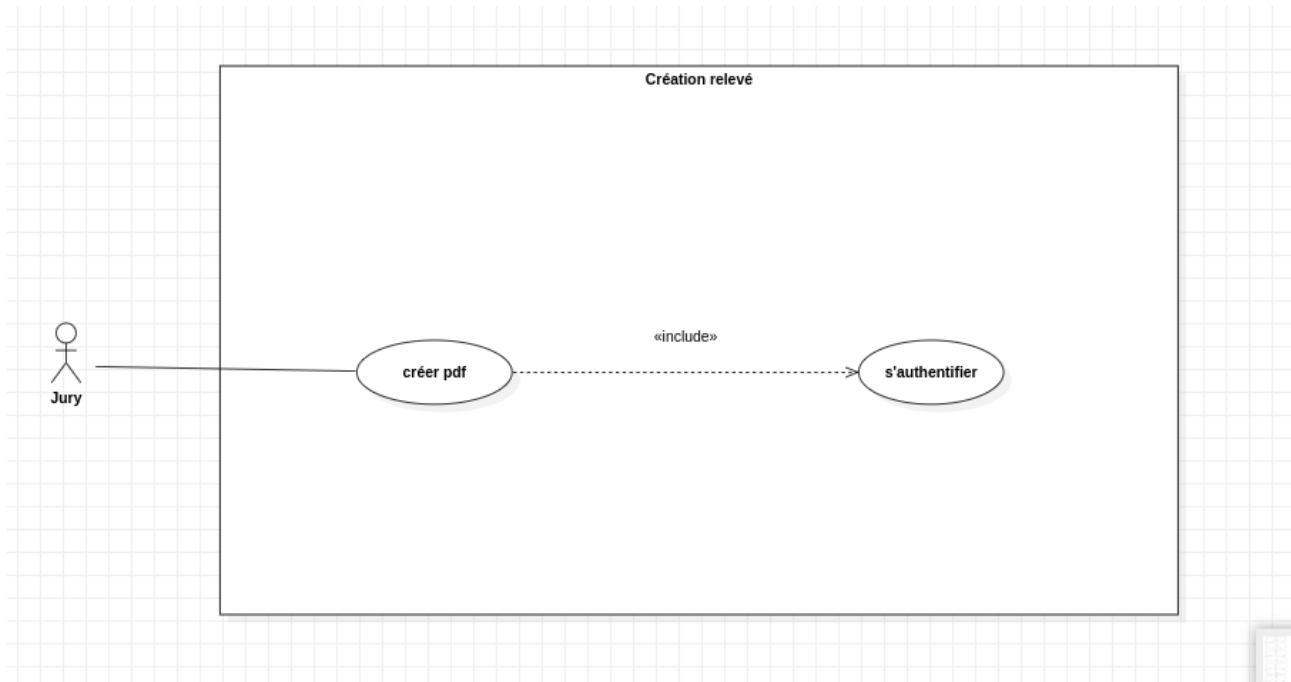


FIGURE 4 – Cas d'utilisation : Création des PDFs(Relevés)



#### 3.1.4 Diagramme de séquence

FIGURE 5 – Cas d'utilisation : Envoyer le pdf par mail

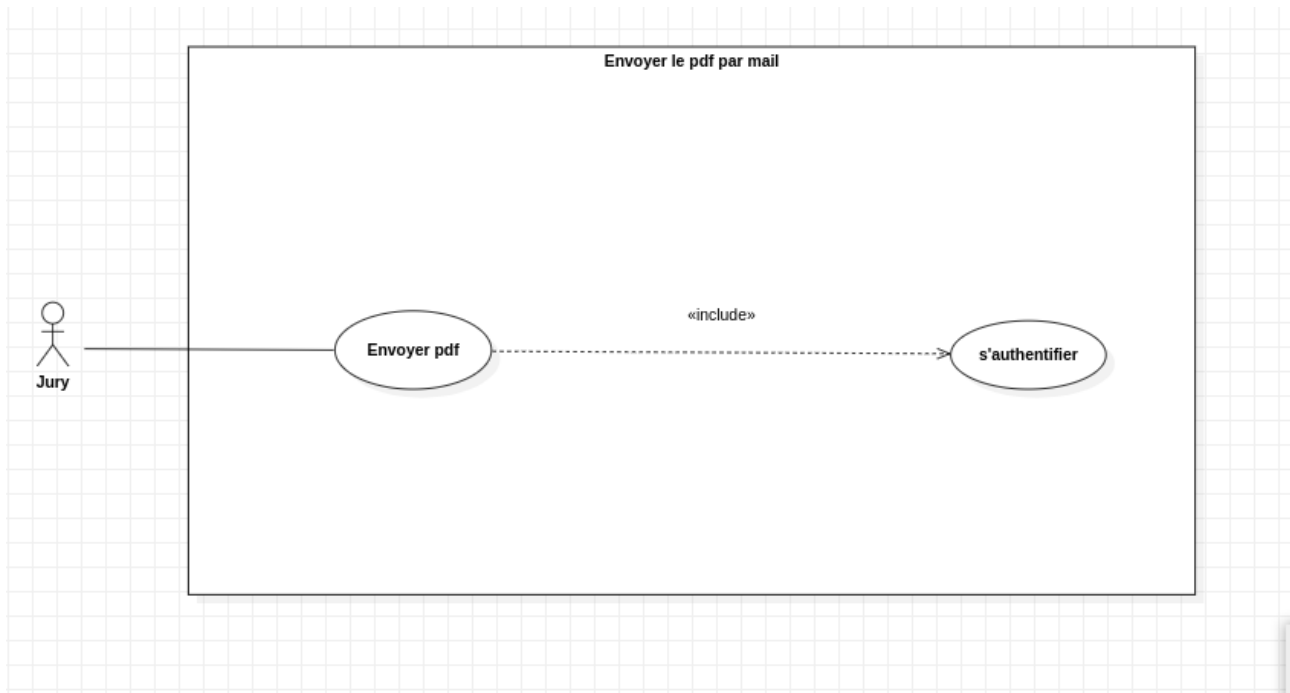
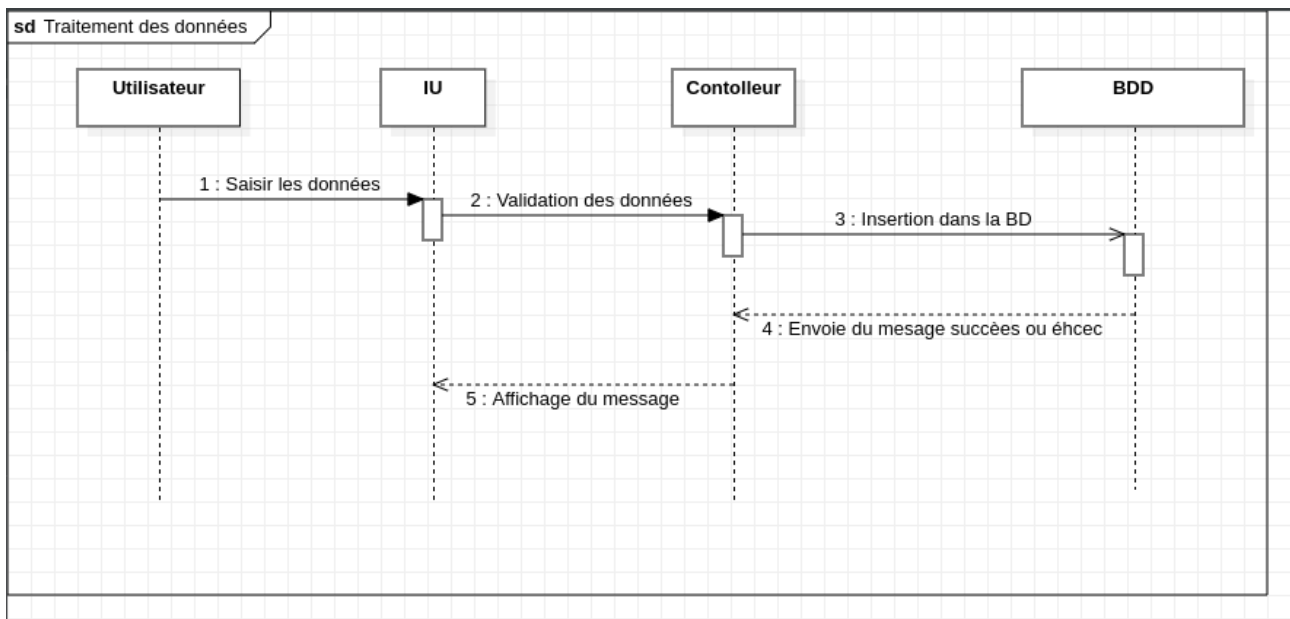


FIGURE 6 – Diagramme de séquence



## IMPLÉMENTATION

---

Dans ce chapitre, nous allons passer en revue les différentes décisions techniques qui ont été prises ainsi que les outils utilisés, et présenter les résultats obtenus.

### 4.1 CHOIX TECHNIQUES ET MOTIVATIONS

#### *MySQL*

Le choix de la base de données est une étape cruciale dans le développement d'une application. Il existe plusieurs types de bases de données, chacune ayant ses avantages et inconvénients propres.

Dans le cadre de ce projet, nous avons opté pour une base de données relationnelle, car elle s'adapte le mieux à notre cas d'utilisation. Elle nous permet de stocker les données de manière structurée et de les manipuler aisément à l'aide de requêtes SQL.

Nous avons choisi d'utiliser MySQL, une base de données relationnelle populaire dotée d'une large communauté de développeurs et d'une documentation exhaustive

#### *Architecture de l'application*

Notre application repose sur une architecture client-serveur. Le serveur assure la gestion des données et la logique métier, tandis que le client gère l'interface utilisateur. Pour communiquer avec le serveur, le client utilise une API REST (Representational State Transfer), qui repose sur un ensemble de conventions et de bonnes pratiques pour la conception d'API web.

Cette API REST permet au client d'effectuer des opérations CRUD (Create, Read, Update, Delete) sur les données stockées sur le serveur. Nous avons choisi d'utiliser une API REST pour sa simplicité de mise en œuvre et son utilisation aisée. Elle permet également de séparer la logique métier de l'interface utilisateur, facilitant ainsi la maintenance et l'évolutivité de l'application.

Nous utilisons le format de données JSON (JavaScript Object Notation) pour l'API REST, car il est léger, facile à lire et à écrire.

#### *Le processus unifié*

Nous avons choisi d'utiliser le processus unifié pour le développement de notre application. Le processus unifié est un processus de développement de logiciels itératif et incrémental de la famille des méthodes agiles.

le processus unifié utilise des modèles UML (Unified Modeling Language) pour la conception et la modélisation.



### *Typescript*

Nous avons choisi d'utiliser le langage de programmation Typescript pour le développement de notre application. Typescript est un langage open source développé par Microsoft, conçu pour le développement d'applications JavaScript à grande échelle. Il ajoute des fonctionnalités au JavaScript, comme le typage statique, les classes, les interfaces, les modules, etc. Il est compilé en JavaScript, ce qui permet son utilisation sur n'importe quel navigateur web ou serveur web.

Le principal avantage de Typescript est qu'il est facile à apprendre et à utiliser. Il permet de détecter les erreurs de programmation avant l'exécution du code, ce qui facilite le débogage et améliore la qualité du code. De plus, étant compilé en JavaScript, il permet d'avoir un code client et serveur en Typescript, ce qui évite de changer de langage entre le client et le serveur.

### *NestJS*

Pour le développement de la partie serveur de notre API, nous avons choisi d'utiliser le framework open source NestJS. Ce framework est spécialement conçu pour le développement d'applications Node.js, basé sur Express et utilisant Typescript.

Il permet de créer des applications évolutives et efficaces grâce à son architecture modulaire. NestJS est composé de plusieurs modules, chacun ayant sa propre fonctionnalité.

### *ReactJs*

Nous avons choisi d'utiliser ReactJs pour le développement de notre interface utilisateur (partie client). web. ReactJs est une bibliothèque JavaScript open source pour la création d'interfaces utilisateur. Il est conçu pour faciliter la création d'interfaces utilisateur interactives et réutilisables. Il est utilisé par de nombreuses entreprises, dont Facebook, Instagram, Netflix, Airbnb, etc.

### *Git*

Nous avons adopté Git pour la gestion de version de notre code. Git est un système de contrôle de version distribué open source, conçu pour gérer efficacement les petits et grands projets. Il permet de suivre les modifications apportées au code source et facilite la collaboration entre les développeurs.

Nous avons choisi Git pour sa facilité d'apprentissage et d'utilisation. Il nous permet de suivre les modifications apportées au code source et de revenir à une version antérieure du code si nécessaire.

### *GitHub*

Nous avons décidé d'héberger notre code sur GitHub. GitHub est un service d'hébergement de code source basé sur Git, conçu pour faciliter la collaboration entre les développeurs. Il permet d'héberger des projets open source et privés.

Nous avons choisi GitHub pour son accessibilité et sa facilité d'utilisation. Étant donné que notre projet est open source, nous souhaitons offrir la possibilité à ceux qui le souhaitent de contribuer facilement au projet.

### *Tailwindcss*

Pour le développement de notre interface utilisateur côté client, nous avons opté pour le framework css Tailwindcss. Ce framework adopte l'approche utility-first, qui consiste à privilégier l'utilisation de classes utilitaires pour définir les styles plutôt que de créer des classes spécifiques à chaque élément.

Tailwindcss permet de créer des interfaces utilisateur de manière rapide et efficace, grâce à sa bibliothèque de classes utilitaires prédéfinies

### *Conclusion*

Nous sommes conscients que toutes les décisions prises ne sont pas forcément les meilleures, mais elles ont été prises en fonction des contraintes et des objectifs du projet.

Notre objectif est de résoudre le problème de la délibération tout en fournissant une solution facile à maintenir et à faire évoluer à l'avenir, et également facilement adaptable à d'autres problèmes similaires, avec une grande extensibilité.

Le choix d'une API offre une grande flexibilité et une grande extensibilité, car la solution peut être utilisée par d'autres applications et facilement intégrée à d'autres systèmes. Elle peut également être utilisée pour concevoir d'autres types d'applications, tels que des applications mobiles, web ou desktop.

## 4.2 PRÉSENTATION DES RÉSULTATS

### 4.2.1 *Présentation de l'API Rest*

Notre API comprend un total de 44 routes que nous avons catégorisées en 9 groupes, en fonction de la ressource qu'elles manipulent ou de leur tâche respective

#### *Les routes de l'authentification*

TABLE 1 – Tableau des routes de l'authentification

Méthode	Params	Corps	Réponses
POST /login	-	{ email, mot de passe }	{ jeton }
GET /profile	-	-	utilisateur : { id, nom, email, mot de passe, est_actif }
PATCH /profile	-	{ nom, email }	{ status, message }
PATCH /update-password	-	{ Ancien mot de passe, nouveau mot de passe }	{ status, message }

### *Les routes de la gestion des facultés*

TABLE 2 – Tableau des routes de la gestion des facultés

Méthode	Action	Params	Corps	Réponse
POST /	Créer	-	{ nom }	{ status, message }
GET /	Lire	-	-	facultés[] : { id, nom }
GET /:id	Lire	{ id }	-	facultés : { id, nom }
PATCH /:id	Modifier	{ id }	{ nom }	{ status, message }
DELETE /:id	Supprimer	{ id }	-	{ status, message }

*Les routes de la gestion des filières*

TABLE 3 – Tableau des routes de la gestion des filières

Méthode	Action	Params	Corps	Réponse
POST /	Créer	-	{ nom, faculté }	{ status, message }
GET /	Lire	-	-	filières[] : { id, nom, faculté }
GET /:id	Lire	{ id }	-	filière : { id, nom, faculté }
PATCH /:id	Modifier	{ id }	{ nom, faculté }	{ status, message }
DELETE /:id	Supprimer	{ id }	-	{ status, message }

*Les routes de la gestion des cours*

TABLE 4 – Tableau des routes de la gestion des cours

Méthode	Action	Params	Corps	Réponse
POST /	Créer	-	{ nom, heures, crédits, promotion, enseignant, filière }	{ status, message }
GET /	Lire	-	-	cours[] : { id, nom, heures, crédits, promotion, enseignant, filière }
GET /:id	Lire	{ id }	-	cours : { id, nom, heures, crédits, promotion, enseignant, filière }
PATCH /:id	Modifier	{ id }	{ nom, heures, crédits, promotion, enseignant, filière }	{ status, message }
DELETE /:id	Supprimer	{ id }	-	{ status, message }

*Les routes de la gestion des étudiants*

TABLE 5 – Tableau des routes de la gestion des étudiants

Méthode	Route	Paramètre	Corps	Réponses
GET	/	-	-	{ étudiants[] }
GET	/:id	{ id }	-	étudiant : { id, nom, matricule, email, filière, cours[] }
POST	/	-	{ nom, matricule, email, filière }	{ status, message }
PATCH	/:id	{ id }	{ nom, matricule, email, filière }	{ status, message }
DELETE	/:id	{ id }	-	{ status, message }
GET fields/:fieldId/promotions/-promotionId	Lire	{ fieldId, promotionId }	-	étudiants[] : { id, nom, email, matricule, filière, promotion }
GET cours/:coursID	Lire	{ coursId }	-	étudiants[] : { id, nom, email, matricule, promotion, compléments }

*Les routes de la gestion des cotes*

TABLE 6 – Tableau des routes de la gestion des cotes

Méthode	Route	Params	Corps	Réponses
POST /	Créer	-	{ moyenne, moyenne_eg, session, promo- tion_etudiant, cours, étudiant }	{ status, message }
GET /	Lire	-	-	cotes[] : { id, moyenne, moyenne_eg, session, promotion_etudiant, cours, étudiant }
GET / :id	Lire	{ id }	-	cote : {id, moyenne, moyenne_eg, session, promotion_etudiant, cours, étudiant }
PATCH / :id	Modifier	{ id }	{ id, moyenne, moyenne_eg, session, promo- tion_etudiant, cours, étudiant }	{ status, message }
DELETE / :id	Supprimer	{ id }	-	{ status, message }

*Les routes de la gestion des délibérations*

TABLE 7 – Tableau des routes de la gestion des délibérations

Méthode	Route	Params	Corps	Réponses
GET / :id	Lire	{ id }	-	{ nom, matricule, email, filière, cours[] }
GET generate-reports/ :id	Lire	{ id }	-	{ status, message }
GET send-reports/ :id	Lire	{ id }	-	{ status, message }



*Les routes de la gestion des utilisateurs*

TABLE 8 – Tableau des routes de la gestion des utilisateurs

Méthode	Route	Params	Corps	Réponses
GET /	Lire	-	-	utilisateurs[] : { id, nom, email, mot de passe, est_actif, rôles[] }
GET /:id	Lire	{ id }	-	utilisateur : { id, nom, email, mot de passe, est_actif, rôles[] }
POST /	Créer	-	{ nom, email, rôles[] }	{ status, message }
PATCH /:id	Modifier	{ id }	{ nom, email, rôles[] }	{ status, message }
DELETE /:id	Supprimer	{ id }	-	{ status, message }

### Les routes de la gestion des rôles

TABLE 9 – Tableau des routes de la gestion des rôles

Méthode	Route	Params	Corps	Réponses
GET /	Lire	-	-	rôles[] : { id, nom }
GET /:id	Lire	{ id }	-	rôle : { id, nom }
POST /	Créer	-	{ nom }	{ status, message }
PATCH /:id	Modifier	{ id }	{ nom }	{ status, message }
DELETE /:id	Supprimer	{ id }	-	{ status, message }

#### 4.2.2 Présentation de l'interface utilisateur

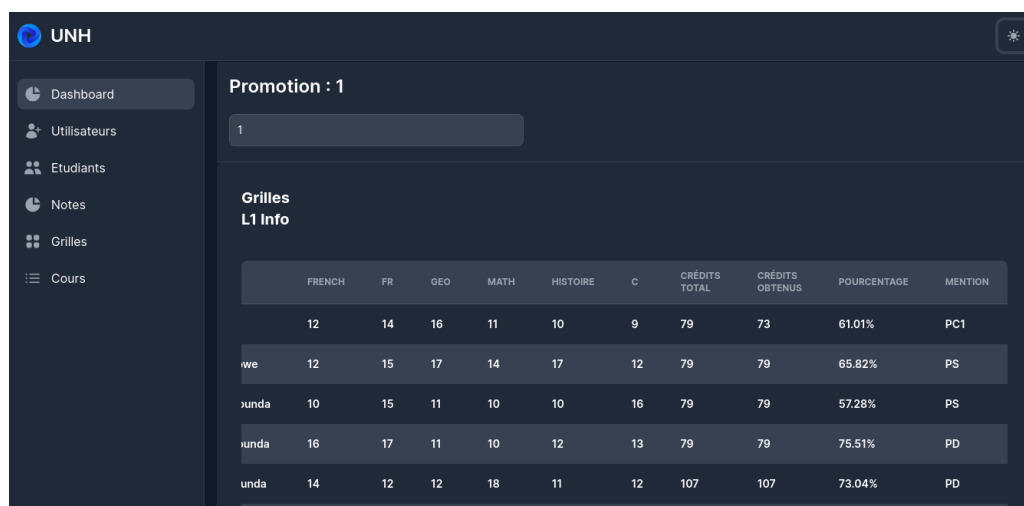
##### Authentification

L'authentification est la première étape de l'utilisation de l'application. Elle permet à l'utilisateur de s'identifier et d'accéder à l'application. L'authentification est gérée par le serveur, qui vérifie les informations d'identification de l'utilisateur et lui renvoie un jeton d'authentification s'il est valide.

FIGURE 7 – Page d'authentification

### Tableau de bord

Le tableau de bord est la page d'accueil de l'application. Il permet à l'utilisateur de voir les statistiques de l'application, telles que le nombre d'étudiants, de cours, de promotions, etc. Il permet également à l'utilisateur de naviguer vers les différentes pages de l'application.



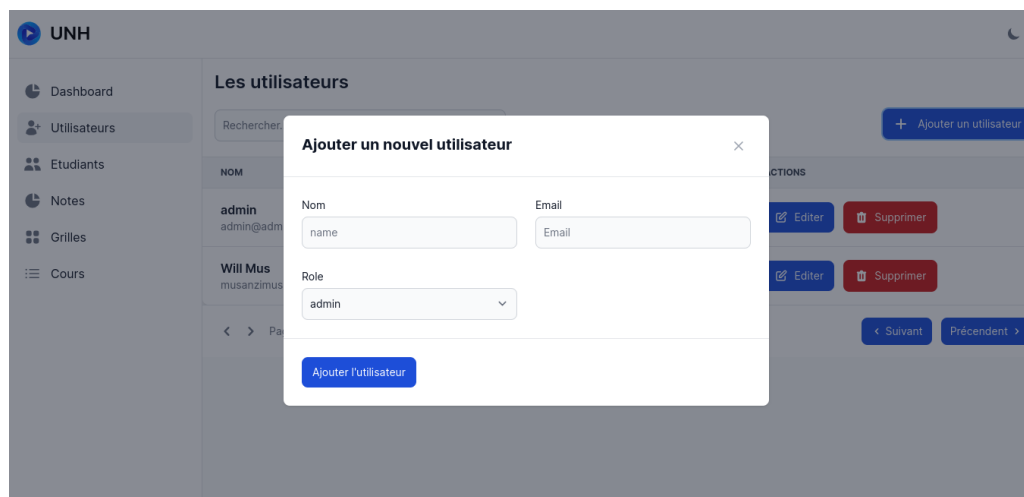
The screenshot shows the UNH dashboard with a sidebar menu and a main content area. The main content area displays 'Promotion : 1' and a table titled 'Grilles L1 Info'.

	FRENCH	FR	GEO	MATH	HISTOIRE	C	CRÉDITS TOTAL	CRÉDITS OBTENUS	POURCENTAGE	MENTION
	12	14	16	11	10	9	79	73	61.01%	PC1
we	12	15	17	14	17	12	79	79	65.82%	PS
xunda	10	15	11	10	10	16	79	79	57.28%	PS
xunda	16	17	11	10	12	13	79	79	75.51%	PD
xunda	14	12	12	18	11	12	107	107	73.04%	PD

FIGURE 8 – Tableau de bord

### Utilisateurs

La page des utilisateurs permet à l'utilisateur de gérer les utilisateurs de l'application. Il peut créer, modifier et supprimer des utilisateurs. Il peut également modifier les rôles des utilisateurs.



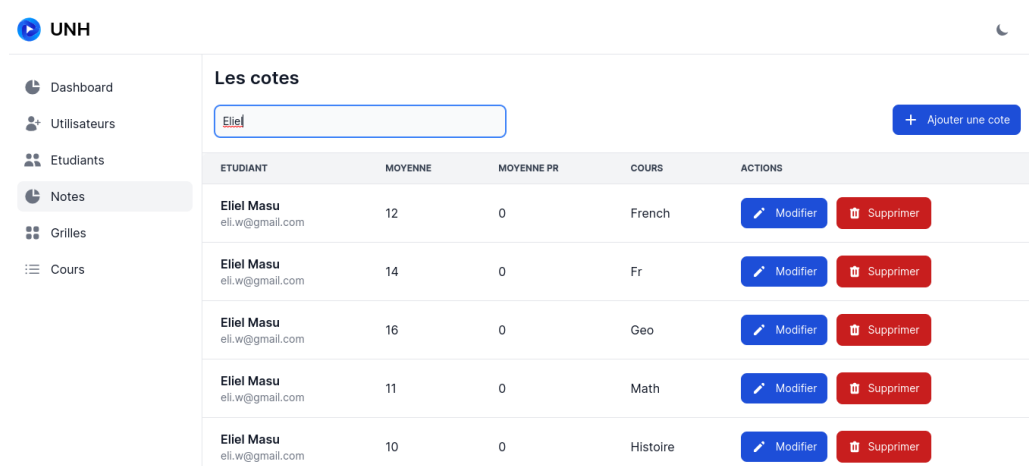
The screenshot shows the 'Les utilisateurs' page in the UNH application. A modal titled 'Ajouter un nouvel utilisateur' is open, allowing the user to add a new user. The modal contains fields for 'Nom' (Name) and 'Email', a 'Role' dropdown menu, and an 'Ajouter l'utilisateur' button.

Background page details: The page title is 'Les utilisateurs'. It includes a search bar, a list of users (e.g., 'admin', 'Will Mus'), and buttons for 'Ajouter un utilisateur', 'Modifier', and 'Supprimer'.

FIGURE 9 – Page des utilisateurs

### Cotes

La page des cotes permet à l'utilisateur de gérer les cotes des étudiants. Il peut créer, modifier et supprimer des cotes.



**UNH**

Dashboard  
Utilisateurs  
Etudiants  
Notes  
Grilles  
Cours

**Les cotes**

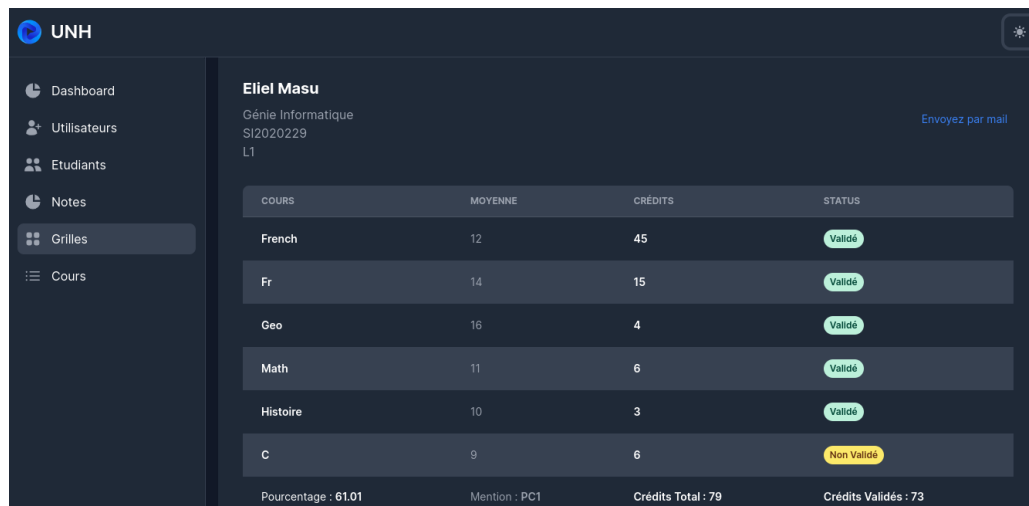
Eliel + Ajouter une cote

ETUDIANT	MOYENNE	MOYENNE PR	COURS	ACTIONS
Eliel Masu eli.w@gmail.com	12	0	French	<a href="#">Modifier</a> <a href="#">Supprimer</a>
Eliel Masu eli.w@gmail.com	14	0	Fr	<a href="#">Modifier</a> <a href="#">Supprimer</a>
Eliel Masu eli.w@gmail.com	16	0	Geo	<a href="#">Modifier</a> <a href="#">Supprimer</a>
Eliel Masu eli.w@gmail.com	11	0	Math	<a href="#">Modifier</a> <a href="#">Supprimer</a>
Eliel Masu eli.w@gmail.com	10	0	Histoire	<a href="#">Modifier</a> <a href="#">Supprimer</a>

FIGURE 10 – Page des cotes

### Relvés

La page des relevés permet à l'utilisateur de générer et d'envoyer des relevés aux étudiants.



**UNH**

Dashboard  
Utilisateurs  
Etudiants  
Notes  
Grilles  
Cours

**Eliel Masu**  
Génie Informatique  
SI2020229  
L1 Envoyez par mail

COURS	MOYENNE	CRÉDITS	STATUS
French	12	45	Validé
Fr	14	15	Validé
Geo	16	4	Validé
Math	11	6	Validé
Histoire	10	3	Validé
C	9	6	Non Validé

Pourcentage : 61.01    Mention : PC1    Crédits Total : 79    Crédits Validés : 73

FIGURE 11 – Page des relevés

## CONCLUSION

---

Face aux difficultés rencontrées dans la délibération des étudiants, de la récolte des notes à la publication des résultats et au suivi des compléments des étudiants, nous avons proposé une application pour résoudre ce problème. Nous avons ainsi contribué à l'essor technologique de l'Université Nouveaux Horizons, tout en mettant en pratique les leçons apprises tout au long de notre cursus académique. Notre objectif était également de fournir une base qui pourrait être étendue à un système plus complet.

La solution que nous avons proposée est une application client-serveur composée d'une API Rest d'un côté pour offrir une solution facilement extensible et indépendante, et d'une interface web de l'autre.

Notre travail se limite à fournir une solution que nous appelons ici "module", qui peut certes être utilisée, mais qui est loin de ce que nous aurions voulu proposer. Elle se situe entre un système d'apprentissage et un système de gestion des étudiants.

Ce travail est loin d'être fini et peut être amélioré en permanence. Le code source est disponible sur GitHub. Quelques pistes pour des développements ultérieurs pourraient être l'extension à un système d'apprentissage complet avec la gestion des travaux, la possibilité pour les étudiants de se connecter, le suivi des performances, etc. et également à un système de gestion des étudiants, avec la gestion des paiements, des inscriptions, etc.

En conclusion, ce travail fournit une base pouvant être étendue, une solution à un problème et surtout contribue à l'essor technologique de notre université. Elle permet de calculer le pourcentage, de proposer des mentions, de gérer les étudiants et leurs compléments, de générer une proposition de relevé pouvant être modifiée, ce qui allège la tâche du jury et lui permet de se concentrer sur d'autres aspects de la délibération.

## BIBLIOGRAPHIE

---

- [1] T. BERNERS-LEE et al. « World-wide web : The information universe ». In : *Electronic Networking : Research, Applications and Policy* 1.2 (1992), p. 52-58. DOI : [10.1016/0968-5227\(92\)90001-F](https://doi.org/10.1016/0968-5227(92)90001-F).
- [2] Grady BOOCH, James RUMBAUGH et Ivar JACOBSON. *The Unified Modeling Language User Guide*. 1st. Reading, MA : Addison-Wesley, 1999. ISBN : 978-0201571684.
- [3] R.T. FIELDING. « Architectural styles and the design of network-based software architectures ». In : *PhD thesis* (2000).
- [4] Martin FOWLER et Kendall SCOTT. *UML Distilled : A Brief Guide to the Standard Object Modeling Language*. 3rd. Boston, MA : Addison-Wesley, 2003. ISBN : 978-0321193681.
- [5] Sanford FRIEDENTHAL, Alan MOORE et Rick STEINER. *A Practical Guide to SysML : The Systems Modeling Language*. 3rd. Waltham, MA : Morgan Kaufmann, 2014. ISBN : 978-0123852069.
- [6] Hector GARCIA-MOLINA. « Database Systems : The Complete Book ». In : *ACM Computing Reviews* 60.3 (2019), p. 186-188. DOI : [10.1145/3310730](https://doi.org/10.1145/3310730).
- [7] IEEE. *IEEE Standard Glossary of Software Engineering Terminology*. IEEE Std 610.12-1990, 1990.
- [8] R. C. MARTIN. *Clean Code : A Handbook of Agile Software Craftsmanship*. Prentice Hall, 2008.
- [9] Michael B PAULSEN. « Student academic achievement and retention ». In : *New Directions for Institutional Research* 1984.41 (1984), p. 5-23.
- [10] Michael B PAULSEN. « Student academic achievement and retention ». In : *New Directions for Institutional Research* 1984.41 (1984), p. 5-23.
- [11] R. PRESSMAN. *Software Engineering : A Practitioner's Approach*. McGraw-Hill Education, 2010.
- [12] L. RICHARDSON et S. RUBY. *RESTful Web Services*. O'Reilly Media, 2007. ISBN : 9780596529260.
- [13] Ian SOMMERVILLE. *Software engineering*. Pearson Education Limited, 2016.