

UNIVERSITE LIBRE DES PAYS DES GRANDS LACS
FACULTE DE SCIENCES ET TECHNOLOGIES

DEPARTEMENT DE GENIE INFORMATIQUE



BP. 368 GOMA

www.ulpgl.net

**CONCEPTION ET REALISATION D'UNE
APPLICATION WEB POUR LA FOURNITURE DES
FEEDBACK CONSTRUCTIFS DANS UNE UNITÉ
D'ENSEIGNEMENT**

Par **HEKIMA BAKANGANA Armand**

Travail présenté et défendue en vue de l'obtention du
Diplôme de Bachelor en Sciences de l'Ingénieur

Mention : Génie Informatique

Directeur : CT. IRENGE BAGUMA Raoul

Encadreur : MSc. KAMBALE WA MUHINDO
Abednego

ANNÉE ACADÉMIQUE 2022 - 2023

Epigraphe

« L'éducation est ce qui reste quand on a oublié tout ce qu'on a appris à l'école. »

Albert Einstein

Dédicace

À ma mère Brigitte et à mes chers frères et sœurs. Votre amour et votre soutien inconditionnels ont été ma source d'inspiration constante. Merci d'être mes plus grands soutiens.

Remerciements

Nous exprimons en premier lieu notre gratitude envers Dieu, qui nous a guidé dans la découverte des choses. C'est à la fois un privilège et un honneur que nous apprécions profondément.

A à notre directeur, CT. BAGUMA Raoul IRENGE, ainsi qu'à notre encadreur, MSc. KAMBALE Abednego WA MUHINDO, pour leur suivi attentif, patient et bienveillant tout au long de l'élaboration de ce document. Leur soutien a été inestimable et a grandement contribué à notre travail.

Nous souhaitons également exprimer notre reconnaissance à toutes les personnes qui ont joué un rôle essentiel dans notre bien-être. Nous adressons nos pensées spéciales à notre feu père BAKANGANA Charles SHIRUBWIKO, à notre mère SERUSHAGO Brigitte, à notre oncle BAKANGANA Désire KIHOMBO, ainsi qu'à nos chers frères et sœurs : Flavien BAKANGANA, Carole BAKANGANA, Christelle BAKANGANA, Aubin BAKANGANA, Harold BAKANGANA, Lysette BAKANGANA, Corine BAKANGANA, Christine BAKANGANA, sans oublier notre merveilleux petit frère, Hugues BAKANGANA.

Nous exprimons une reconnaissance toute particulière à Gilbert BAKANGANA, SERUSHAGO Tavi, SEBURIRI Gerard, SERUSHAGO Mariette, BAKANGANA Yvonne, ainsi qu'à tous nos amis, camarades et bienfaiteurs qui ont contribué, de près ou de loin, à notre croissance, notre évolution et notre épanouissement. Leurs encouragements et leur soutien continus ont été une source d'inspiration précieuse.

À tous, nous adressons nos remerciements les plus chaleureux et sincères.

HEKIMA BAKANGANA Armand

Résumé

L'éducation est un pilier fondamental de la société, et l'amélioration continue de l'enseignement est essentielle pour garantir la réussite des étudiants. Fournir des feedbacks constructifs aux étudiants est devenu une priorité dans ce contexte. Cependant, dans les universités de la RDC, la pratique de fournir des feedbacks constructifs est quasi inexistante en raison des classes surchargées et hétérogènes, rendant l'évaluation individuelle chronophage et difficile. Cette situation limite la capacité des étudiants à s'auto-évaluer et à progresser activement dans leur apprentissage. Pour aborder cette problématique, une méthodologie rigoureuse a été adoptée. Elle inclut l'observation participante, des entretiens semi-structurés avec les étudiants et l'analyse documentaire pour recueillir des données sur les besoins spécifiques des enseignants et des étudiants. Des techniques de modélisation UML, telles que les diagrammes de cas d'utilisation, de séquence et de classe, ont été employées pour concevoir l'application en utilisant le logiciel lucidChart pour faire cette modélisation. Ce travail présente une application offrant une plateforme innovante, avec des fonctionnalités distinctes pour les différents types d'utilisateurs. Les administrateurs ont le contrôle total sur les groupes d'utilisateurs, le contenu du système et la gestion des unités d'enseignements, éléments constitutifs et des questions. Les enseignants peuvent gérer leurs éléments constitutifs, superviser les étudiants et leur fournir des commentaires personnalisés. Les étudiants ont accès au contenu des éléments constitutifs et peuvent passer des évaluations. Avec une interface conviviale et des fonctionnalités bien définies, l'application développée en Django(Python) en utilisant l'IDE pycharm simplifie la gestion éducative pour les institutions et améliore l'expérience d'apprentissage pour les utilisateurs. Ainsi ce travail propose une application web visant à améliorer la communication entre les étudiants et professeurs et encourage une meilleure compréhension des matières enseignées. En centralisant et en structurant les retours, l'application contribue à une expérience d'apprentissage plus interactive et enrichissante.

Mots-clés : Feedback constructif, Application web, Système Licence-Master-Doctorat, Unité d'enseignement, Communication

Abstract

Education is a fundamental pillar of society, and the continuous improvement of teaching is essential to ensure an enriched student learning success. Providing constructive feedback to students has become a priority in this context. However, in universities in the Democratic Republic of Congo, the practice of providing constructive feedback is almost non-existent due to overcrowded and heterogeneous classes, making individual evaluation time-consuming and difficult. This situation limits students' ability to self-assess and actively progress in their learning. To address this issue, a rigorous methodology was adopted. This included participant observation, semi-structured interviews with students, and document analysis to gather data on the specific needs of both teachers and students. UML modeling techniques, such as use case diagrams, sequence diagrams, and class diagrams, were employed using the LucidChart software to design the application. This work presents an application offering an innovative platform with distinct functionalities for different types of users. Administrators have full control over user groups, system content, and the management of Ue, Ec and questions. Teachers can manage their EC, supervise students, and provide personalized feedback. Students have access to Ec content and can take assessments. With a user-friendly interface and well-defined features, the application developed in Django (Python) using the PyCharm IDE simplifies educational management for institutions and enhances the learning experience for users. Thus, this work proposes a web application aimed at improving communication between students and professors and encouraging a better understanding of the taught subjects. By centralizing and structuring feedback, the application contributes to a more interactive and enriching learning experience.

Keywords: Constructive Feedback, Web application, Bachelor's-Master's-Doctorate System, Teaching Unit, communication

Table des matières

Epigraphe.....	i
Dédicace.....	ii
Remerciements.....	iii
Résumé.....	iv
Abstract.....	v
Table des matières.....	vi
Listes des sigles et abréviations	viii
Listes des tableaux	ix
Listes des figures.....	x
0 INTRODUCTION GENERALE.....	1
0.1 Contexte	1
0.2 Identification et formulation du probleme	2
0.3 Formulation des hypothèses	2
0.4 Choix et intérêt du sujet	3
0.5 Objectif du travaille.....	4
0.6 Méthodologie et délimitation du travail	5
0.7 Subdivision du travail.....	5
Chapitre 1. GENERALITES	7
1.1 L'effet du feedback sur l'apprentissage des étudiants.....	7
1.2 L'effet de l'évaluation sur l'apprentissage des étudiants	8
1.3 L'impact des outils informatiques sur l'apprentissage.....	9

1.3.1	Le Web.....	10
1.3.2	Les sites Web	12
1.4	Architecture client/serveur [5] :	14
1.4.1	Avantages de l'architecture client/serveur :.....	15
1.5	Conclusion.....	16
Chapitre 2. ANALYSE ET CONCEPTION.....		18
2.1	Méthode d'analyse et de conception :	18
2.1.1	Présentation d'UML :	18
2.1.2	Conception :	34
2.2	Mesure de Sécurité	39
Chapitre 3. REALISATION		42
3.1	Introduction :	42
3.2	Outils de développement :.....	42
3.2.1	L'interface de Django Admin :.....	42
3.2.2	Les serveurs :	43
3.2.3	IDE PyCharm.....	44
3.3	Les langages de programmation utilisés :	45
3.4	Les interfaces de notre application :.....	47
CONCLUSION GENERALE.....		57
Bibliographie.....		59

Listes des sigles et abréviations

EC : Élément constitutif.

HTML : Langage de Balisage Hypertexte.

HTTP : Protocole de Transfert Hypertexte.

IDE : Environnement de Développement Intégré.

JavaScript : Langage de Script Côté Client.

Nginx : Moteur de Serveur Web.

SGBDR : Système de Gestion de Base de Données Relationnelle.

SQLite : Système de Gestion de Base de Données Léger.

UE : Unité d'enseignement.

UML : Langage de Modélisation Unifié.

URL : Adresse de Ressource Uniforme.

Listes des tableaux

Tableau 2.1 : cas d'utilisation du prétendant	22
Tableau 2.2 : cas d'utilisation de l'enseignant.....	22
Tableau 2.3 : cas d'utilisation de l'étudiant.....	22
Tableau 2.4 : cas d'utilisation de l'administrateur.....	23
Tableau 2.5 : Accéder à l'espace administrateur	23
Tableau 2.6 : Gérer les groupes utilisateurs	24
Tableau 2.7 : Confirmation de l'inscription	25
Tableau 2.8 : cas d'utilisation inscription d'un prétendant.....	26
Tableau 2.9 : Gestion des UE	27
Tableau 2.10 : Gérer les EC.....	28
Tableau 2.11 : Gérer les questions.....	29
Tableau 2.12 : Fournir des commentaires aux étudiants	30
Tableau 2.13 : Inscription à un EC	31
Tableau 2.14 : Consulter le contenu des EC.....	32
Tableau 2.15 : Passer des évaluations	33

Listes des figures

Figure 1.1 : Site web statique.....	13
Figure 1.2 : Architecture client server.....	16
Figure 2.1 : Diagramme de cas d'utilisation.....	21
Figure 2.2 : Diagramme de séquence du cas d'utilisation inscription.....	35
Figure 2.3 : Diagramme de séquence du cas d'utilisation confirmer enseignant	36
Figure 2.4 : Diagramme de séquence du cas d'utilisation fournir commentaire.....	37
Figure 2.5 : Diagramme se séquence des cas d'utilisation consulter EC et passer évaluation	38
Figure 2.6 : Diagramme des classes.....	39
Figure 3.1 : interface Django-admin.....	43
Figure 3.2 : Aperçu d'une interface de l'IDE PyCharm.....	45
Figure 3.3 : interface d'authentification	47
Figure 3.4 : interface Dashboard(admin).....	48
Figure 3.7 : interface d'inscription professeur	49
Figure 3.8 : interface Ajouter un EC.....	50
Figure 3.9 : interface liste des EC.....	51
Figure 3.10 : interface ajouter une question	52
Figure 3.11 : Interface liste des mes EC	52
Figure 3.12 : interface passer évaluation	53
Figure 3.13 : interface résultats.....	54
Figure 3.14 : Interface liste EC	55
Figure 3.15 : interface messagerie	55

INTRODUCTION GENERALE

0.1. Contexte

L'éducation est un pilier fondamental de la société, et l'amélioration continue de l'enseignement est essentielle pour garantir la réussite des étudiants. Dans ce contexte, le besoin de fournir des feedbacks constructifs aux étudiants est devenu une priorité. Cependant, de nombreux défis se posent dans le processus d'évaluation et de communication du feedback qui est une information de retour et un processus d'évaluation permettant de connaître l'opinion sur les actions, les comportements ou les résultats d'une personne, d'une équipe ou d'une organisation spécifique [1].

Tout d'abord, le nombre croissant d'étudiants inscrits dans les établissements d'enseignement supérieur rend la tâche des enseignants de plus en plus complexe. Il est difficile de fournir un feedback détaillé et personnalisé à chaque étudiant, en particulier dans des matières variées et surtout dans les classes pléthoriques. De plus, le monde de l'éducation évolue rapidement, avec des besoins pédagogiques changeants et des approches d'enseignement diverses. Les enseignants doivent s'adapter à ces évolutions et trouver des moyens plus efficaces de soutenir l'apprentissage des étudiants. En parallèle, les étudiants sont de plus en plus connectés et technophiles. Ils attendent une expérience d'apprentissage moderne et interactive, avec un accès facile aux informations et aux commentaires sur leur travail.

C'est dans ce contexte que le projet de conception et de développement d'une application de feedback constructif émerge. L'objectif est de combler l'écart entre les besoins des enseignants et des étudiants en offrant une solution technologique permettant une communication efficace du feedback. Cette application vise à simplifier le suivi des performances des étudiants, à fournir des commentaires précis et à encourager l'engagement actif des étudiants dans leur propre apprentissage.

Ainsi, ce projet s'inscrit dans une démarche d'amélioration continue de l'éducation, en exploitant la technologie pour répondre aux besoins changeants des enseignants et des étudiants dans un environnement éducatif en constante évolution.

0.2. Identification et formulation du problème

Au sein des universités de la RDC, la pratique de fournir un feedback constructif aux étudiants au sein d'une unité d'enseignement (UE) est quasi-inexistante. Les enseignants sont souvent confrontés à des classes pléthoriques et hétérogènes. Cependant évaluer et aider chaque étudiant individuellement, est une tâche qui est à la fois chronophage et parfois difficile à réaliser rendant la pratique du feedback inexistante. Par ailleurs, la mise en place d'un tel feedback est essentielle pour permettre aux étudiants de s'autoévaluer et de s'engager activement et de progresser dans leur apprentissage. Au vu de cette problématique, nous orientons nos questions de recherche de la manière suivante :

1. Comment la conception d'un outil informatique peut répondre aux défis auxquels sont butés les enseignants et aux besoins particuliers des étudiants en termes des feedbacks constructif dans une UE ?
2. Quels sont les principaux obstacles et défis que rencontrent les enseignants lorsqu'il s'agit de fournir des feedbacks constructifs à un grand nombre d'étudiants, et comment l'outil peut-il contribuer à les surmonter ?
3. Quels sont les besoins spécifiques des étudiants en ce qui concerne les feedbacks constructifs dans une unité d'enseignement, et comment l'outil peut-il satisfaire ces besoins tout en améliorant leur expérience d'apprentissage ?
4. Comment la technologie peut-elle aider à la fois les enseignants et les étudiants à avoir des feedbacks constructifs pour l'amélioration de l'apprentissage dans nos universités et quelle technologie s'adapterait le mieux pour y arriver ?

0.3. Formulation des hypothèses

Aux questions de recherches nous fournissons de réponses anticipatives suivantes :

1. La conception d'un outil informatique adapté aux défis rencontrés par les enseignants et aux besoins spécifiques des étudiants en matière de feedback constructif dans une UE permettrait une amélioration significative de la qualité des interactions enseignant-étudiant et de l'apprentissage en général.
2. Les principaux obstacles et défis auxquels seraient confrontés les enseignants lors des de feedback constructifs à un grand nombre d'étudiants, tels que la charge de travail excessive, pourraient être atténués par l'utilisation de l'outil informatique, améliorant ainsi l'efficacité de ce processus.
3. Les besoins spécifiques des étudiants en matière de feedback constructif, tels que la régularité et la personnalisation des retours, pourraient être satisfaits par l'outil en question, et amélioreraient ainsi leur expérience d'apprentissage, leur motivation et leur performance académique.
4. La mise en œuvre d'un outil technologique combinant des fonctionnalités de suivi des enseignants et d'autoévaluation des étudiants, permettrait d'améliorer le suivi individuel des progrès des étudiants et de les encourager à s'engager activement dans leur propre apprentissage.

0.4. Choix et intérêts du sujet

Le choix de ce sujet découle de plusieurs considérations essentielles :

1. L'importance du feedback constructif : Le feedback constructif revêt une importance capitale dans le processus d'apprentissage, car il permet aux étudiants d'identifier leurs compétences et lacunes, de déterminer les moyens pour améliorer leurs aptitudes et d'avancer dans leurs études. Cependant, l'octroi de feedback constructif à un nombre considérable d'étudiants peut s'avérer complexe pour les enseignants, principalement en raison de contraintes temporelles et de ressources.

2. Nécessité d'une solution technologique : Dans un contexte où tant d'enseignants que d'étudiants se tournent vers l'utilisation de technologies numériques, il est opportun de développer une application web et mobile qui simplifie la prestation de feedback constructif. Une telle application aurait pour vocation de pallier les limitations de temps et de ressources en automatisant certaines tâches et en proposant un environnement interactif pour l'apprentissage.
3. Personnalisation de l'expérience d'apprentissage : Cette application conférerait aux enseignants la capacité de concevoir des programmes, des modules et des évaluations sur mesure, en tenant compte des besoins et des objectifs d'apprentissage spécifiques de chaque étudiant. Elle offrirait la possibilité de personnaliser l'expérience d'apprentissage pour chaque individu, favorisant ainsi un apprentissage plus efficace et stimulant.
4. Suivi de la progression des étudiants : L'application permettrait aux enseignants de surveiller la progression des étudiants, d'évaluer leurs performances et d'identifier les domaines où des améliorations sont requises. Cela simplifierait l'évaluation continue et la prestation de feedback individualisé.

0.5. Objectif du travail

0.5.1. L'objectif général

Notre travail consiste à concevoir et de développer une application qui favorise un environnement d'apprentissage interactif et qui facilite la communication entre les enseignants et les étudiants grâce à des fonctionnalités telles que la création de UE, EC, le suivi des progrès des étudiants et la fourniture de feedback constructif.

0.5.2. Les objectifs spécifiques

Pour parvenir à concevoir et développer une application web et mobile pour le feedback constructif dans une unité d'enseignement nous comptons :

1. Analyser les besoins spécifiques de l'application
2. Concevoir l'architecture de l'application
3. Développer l'application suivant les normes du génie logiciel
4. Effectuer les tests et le déploiement de l'application

0.6. Méthodologie et délimitation du travail

La méthodologie et techniques de recherche qui seront utilisées pour réaliser ce travail incluent :

- L'observation participante
- Les entretiens semi-structurés avec les étudiants
- L'analyse documentaire

Ces techniques permettront de recueillir des données sur les besoins spécifiques des enseignants et des étudiants, les objectifs pédagogiques de l'unité d'enseignement et les outils en ligne existants pour fournir des feedbacks constructifs. Nous utiliserons aussi des techniques de modélisation UML pour la modélisation du système suivant les besoins fonctionnels recueillis.

Le travail sera limité à la conception et au développement d'une application web pour fournir des feedbacks constructifs dans plusieurs unités d'enseignement de l'ULPGL/Goma.

0.7. Subdivision du travail

Excepté l'introduction et la conclusion générale le travail est organisé de la façon suivante :

- Chapitre 1. GENERALITES : Dans cette partie nous allons aborder l'effet de l'évaluation et du feedback sur l'apprentissage des étudiants ainsi que la contribution des outils informatiques sur cet apprentissage.

- Chapitre 2. ANALYSE ET CONCEPTION : Cette étape consiste à recueillir les besoins spécifiques des enseignants et des étudiants en matière de feedback constructif et de modéliser l'application.
- Chapitre 3. REALISATION : ce chapitre présente la réalisation du système, les différents Framework utilisés et la présentation des quelques fonctionnalités du système.

Chapitre 1. GENERALITES

Dans le domaine de l'éducation, l'efficacité de l'évaluation et du feedback sur l'apprentissage des étudiants est un sujet d'intérêt croissant. Ce chapitre examine l'impact de ces pratiques sur l'apprentissage des étudiants et explore comment les outils informatiques peuvent contribuer à améliorer ce processus

1.1 L'effet du feedback sur l'apprentissage des étudiants

Le feedback joue un rôle crucial dans le processus d'apprentissage des étudiants. Selon Hattie et Timperley (2007), le feedback efficace fournit aux étudiants des informations précises sur leur performance par rapport aux objectifs d'apprentissage, ce qui les aide à comprendre leurs forces et leurs faiblesses. Cette compréhension permet aux étudiants de réguler leur apprentissage en identifiant les domaines nécessitant une amélioration et en ajustant leurs stratégies d'étude en conséquence.

Une méta-analyse menée par Kluger et DeNisi (1996) a révélé que le feedback a un effet significatif sur l'apprentissage des étudiants, avec une taille d'effet moyenne de 0,41. Cela suggère que le feedback approprié peut avoir un impact positif sur la performance des étudiants. De plus, lorsque le feedback est fourni de manière opportune et spécifique, il peut améliorer la motivation des étudiants en leur donnant un sentiment d'efficacité et en renforçant leur confiance en leurs capacités (Shute, 2008).

Cependant, il est important de noter que tous les types de feedback ne sont pas également efficaces. Selon Hattie et Timperley (2007), le feedback doit être concret, spécifique et orienté vers la tâche pour être le plus bénéfique. De plus, il doit être fourni de manière régulière et opportune afin que les étudiants puissent l'utiliser pour améliorer leur compréhension et leurs performances.

En conclusion, le feedback joue un rôle essentiel dans l'apprentissage des étudiants en leur fournissant des informations précieuses sur leur performance et en les aidant à réguler leur

apprentissage. Cependant, pour être efficace, le feedback doit être concret, spécifique et opportune.

1.2 L'effet de l'évaluation sur l'apprentissage des étudiants

L'évaluation joue un rôle fondamental dans le processus d'apprentissage des étudiants. Selon Black et William (1998), l'utilisation efficace de l'évaluation formative peut considérablement améliorer les résultats académiques en fournissant aux apprenants des informations précieuses sur leur progression et en identifiant les domaines nécessitant une attention supplémentaire. Contrairement à l'évaluation sommative, qui vise à attribuer une note finale à la fin d'un cycle d'apprentissage, l'évaluation formative se concentre sur le suivi continu de la compréhension des élèves tout au long du processus d'apprentissage, offrant ainsi des opportunités d'intervention et d'amélioration en temps réel.

Une méta-analyse menée par Hattie et Timperley (2007) a révélé que l'évaluation formatrice avait un impact significatif sur l'apprentissage des étudiants, avec une taille d'effet moyenne de 0,90. Cela suggère que les pratiques d'évaluation qui mettent l'accent sur le processus d'apprentissage plutôt que sur le résultat final peuvent avoir un impact positif sur la performance des étudiants. De plus, l'évaluation formative encourage l'auto-régulation chez les étudiants en les incitant à réfléchir sur leur propre apprentissage, à identifier leurs propres lacunes et à élaborer des stratégies pour les surmonter (Nicol & Macfarlane-Dick, 2006).

Cependant, il est important de reconnaître que toutes les formes d'évaluation ne sont pas également bénéfiques pour l'apprentissage des étudiants. Les évaluations qui mettent trop l'accent sur la mémorisation des faits au détriment de la compréhension conceptuelle peuvent décourager l'apprentissage authentique et la pensée critique (Wiggins, 1998). Par conséquent, il est essentiel que les enseignants conçoivent des évaluations qui évaluent non seulement la capacité des élèves à restituer des informations, mais aussi leur capacité à analyser, évaluer et appliquer des connaissances dans des contextes nouveaux et pertinents.

En conclusion, l'évaluation joue un rôle crucial dans l'apprentissage des étudiants en fournissant des informations sur leur progression et en favorisant l'auto-régulation. Cependant, pour être efficace, l'évaluation doit être formative, orientée vers le processus d'apprentissage et centrée sur le développement des compétences et de la compréhension plutôt que sur la simple acquisition de connaissances.

1.3 L'impact des outils informatiques sur l'apprentissage

L'avènement des outils informatiques a profondément transformé la manière dont l'apprentissage est abordé dans les environnements éducatifs. Les technologies telles que les plateformes d'apprentissage en ligne, les logiciels de simulation et les applications mobiles offrent aux étudiants des possibilités uniques d'interactivité et de personnalisation de leur expérience d'apprentissage (Clark & Mayer, 2016).

Une étude menée par Means et al. (2010) a examiné l'impact des technologies de l'information et de la communication (TIC) sur l'apprentissage et a conclu que l'utilisation des TIC pouvait avoir un effet positif sur la performance des étudiants, en particulier lorsque les technologies étaient intégrées de manière efficace dans les pratiques pédagogiques existantes. Par exemple, les plateformes d'apprentissage en ligne peuvent offrir des ressources éducatives riches et variées, permettant aux étudiants d'accéder à du contenu supplémentaire, de participer à des discussions en ligne et de collaborer avec leurs pairs à distance (Garrison & Kanuka, 2004).

De plus, les outils informatiques permettent de collecter des données sur les performances des étudiants, ce qui peut être utilisé pour personnaliser l'enseignement et fournir un feedback adapté (Shute & Zapata-Rivera, 2012). Par exemple, les systèmes de gestion de l'apprentissage peuvent suivre le progrès des élèves dans divers domaines et identifier les domaines où des interventions supplémentaires sont nécessaires. De même, les logiciels de simulation offrent aux étudiants la possibilité d'expérimenter des situations réalistes dans un environnement contrôlé, ce qui peut renforcer leur compréhension des concepts complexes et

améliorer leur capacité à appliquer leurs connaissances dans des contextes réels (Sitzmann, 2011).

Cependant, il est important de reconnaître que l'efficacité des outils informatiques dépend en grande partie de la manière dont ils sont intégrés dans l'enseignement et l'apprentissage. Une utilisation inappropriée ou excessive des technologies peut entraîner des distractions et des perturbations dans le processus d'apprentissage (Kirschner & van Merriënboer, 2013). Par conséquent, il est essentiel que les enseignants soient formés à l'utilisation efficace des outils informatiques et qu'ils veillent à ce que ces outils soient utilisés de manière à enrichir l'expérience d'apprentissage des étudiants, plutôt que de la compromettre.

En conclusion, les outils informatiques ont un impact significatif sur l'apprentissage des étudiants en offrant des possibilités d'interactivité, de personnalisation et de collecte de données. Cependant, pour maximiser les avantages des technologies, il est important de les intégrer de manière réfléchie et efficace dans les pratiques pédagogiques existantes.

1.3.1 Le Web

1.3.1.1 L'historique du Web

Au début des années 2000, le Web a connu une évolution majeure avec l'avènement de technologies permettant la création de contenu dynamique. Cette évolution a été facilitée par l'utilisation de bases de données et de scripts côté serveur comme PHP, ASP, et des méthodes comme CGI, permettant de générer du contenu à la demande. Cette avancée a ouvert la voie à de nouveaux types de sites Web, tels que les boutiques en ligne et les réseaux sociaux, favorisant l'émergence de nombreuses applications web interactives [2].

En 2004, le terme "Web 2.0" a été popularisé par Dale Dougherty et Tim O'Reilly. Cette nouvelle phase du Web a mis l'accent sur le partage de l'information et la collaboration entre les utilisateurs. Les bases de données ouvertes ont permis à des tiers de les exploiter et de créer de nouveaux services, favorisant une véritable dimension sociologique au Web. Selon Joshua Porter, expert en la matière, le Web 2.0 se résume à "partager l'information, fondée sur des bases de données ouvertes que d'autres utilisateurs peuvent utiliser" [2].

Depuis quelques mois, le concept de Web 3.0 gagne en popularité. Cette nouvelle itération du Web promet d'être encore plus humaine et intelligente, en s'appuyant sur des technologies telles que l'intelligence artificielle, la blockchain et le métavers. Le Web 3.0 vise à offrir une expérience personnalisée et immersive aux utilisateurs, tout en garantissant la sécurité et la confidentialité des données [2].

1.3.1.2 Définition

Le World Wide Web, souvent abrégé en WWW, W3 ou simplement Web, est un service d'information révolutionnaire qui a transformé la manière dont nous accédons à l'information. Signifiant littéralement "toile d'araignée recouvrant le monde", le Web est un vaste ensemble de pages web interconnectées, accessibles via des navigateurs web tels que Google Chrome, Mozilla Firefox ou Safari. Le Web a connu une croissance exponentielle au cours des dernières années, s'imposant comme une source d'information incontournable. Cette croissance s'explique par sa facilité d'utilisation, sa rapidité d'accès et sa richesse en contenus variés. Le Web a révolutionné la communication, l'éducation, le commerce et bien d'autres domaines, devenant un outil essentiel pour la vie quotidienne de millions d'utilisateurs à travers le monde

1.3.1.3 Les concepts du web

- **Une page web** : est un élément fondamental du World Wide Web, conçue pour être consultée par des utilisateurs via un navigateur web. Elle se compose généralement d'un document HTML et d'images, et peut également inclure des scripts, des feuilles de style et d'autres éléments multimédias.
- **Le navigateur web** : est un logiciel client qui permet aux utilisateurs d'explorer et d'interagir avec les pages web. Il interprète le code HTML et les autres éléments d'une page pour afficher son contenu de manière visuelle et interactive. Parmi les navigateurs web populaires, on trouve Google Chrome, Mozilla Firefox, Safari et Microsoft Edge.
- **Le protocole HTTP** (Hyper Text Transfer Protocol) est le principal protocole de communication utilisé sur le Web. Il permet aux navigateurs web de communiquer avec les serveurs web pour demander et recevoir des pages web et d'autres ressources.

Le protocole HTTP est basé sur un système de requêtes et de réponses, où le client envoie une requête au serveur et le serveur répond en envoyant la ressource demandée.

- **Le protocole FTP** (File Transfer Protocol) : est un protocole de communication dédié au transfert de fichiers sur internet. Il permet aux utilisateurs de télécharger et d'uploader des fichiers vers des serveurs FTP, souvent utilisés pour héberger des sites web ou des fichiers volumineux.
- **Un site web**, aussi appelé site internet, est un ensemble de pages web regroupées sous un même nom de domaine et accessibles via une adresse URL unique. Un site web peut avoir plusieurs objectifs, tels que présenter une information, proposer des services en ligne ou vendre des produits.

1.3.2 Les sites Web

1.3.2.1 Définition

Un site web est un ensemble de pages écrites dans un langage normalisé de manière à être compris par tous les navigateurs. L'organisme qui décrit et décide des évolutions de ce langage s'appelle le W3C, et préconise l'utilisation du HTML pour la création de pages web

1.3.2.2 Classification des sites web :

On peut classer les sites web selon le type des pages qu'ils offrent comme suite :

❖ Les sites web statiques :

Un site web statique est composé de pages web dont le contenu est fixe et ne varie pas en fonction de l'utilisateur qui les consulte. Contrairement aux sites dynamiques, les pages statiques ne sont pas générées à la demande, mais préconstruites et stockées sur un serveur.

➤ Avantages des sites web statiques :

Le coût de développement d'un site web statique est généralement moins élevé car la complexité est moindre par rapport à un site dynamique, ce qui le rend plus abordable initialement. De plus, les pages statiques offrent une meilleure performance et une sécurité renforcée, car elles se chargent plus rapidement et sont moins vulnérables aux attaques informatiques grâce à leur nature préconstruite. En outre, la gestion du contenu est simplifiée,

car les mises à jour sont plus faciles à effectuer et ne nécessitent pas de compétences en programmation complexes. Enfin, les pages statiques peuvent être plus facilement optimisées pour le référencement naturel, ce qui peut améliorer la visibilité en ligne du site [3].

➤ Inconvénients des sites web statiques

Les sites web statiques présentent plusieurs limitations. Tout d'abord, leur manque d'interactivité restreint la possibilité d'engager les utilisateurs en temps réel, ce qui peut limiter leur fonctionnalité globale. De plus, la nécessité de mettre à jour manuellement le contenu des pages statiques peut s'avérer fastidieuse, notamment pour les sites volumineux ou sujets à des mises à jour fréquentes. Enfin, ces sites offrent moins de fonctionnalités avancées par rapport aux sites dynamiques, ne permettant pas l'intégration de certaines options telles que les formulaires de contact ou les paniers d'achat.

La Figure 1.1 illustre le fonctionnement d'un site web statique.

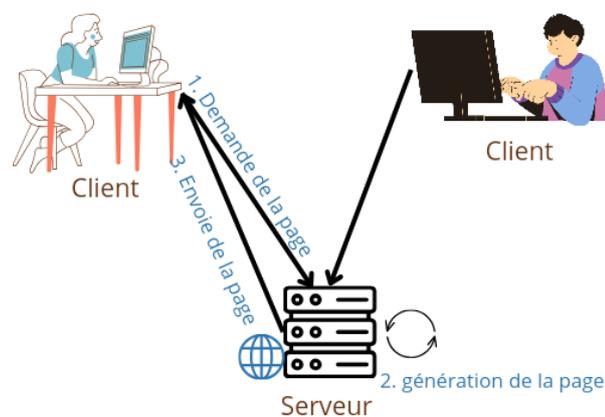


Figure 1.1 : Site web statique

❖ Les sites web dynamiques

Un site web dynamique se distingue par son contenu généré en temps réel, s'adaptant à l'utilisateur qui le consulte ou à d'autres paramètres. Contrairement aux sites statiques, les pages

dynamiques ne sont pas pré-construites, mais créées à la demande à partir d'une base de données et de scripts [4].

Avantages des sites web dynamiques

Les sites dynamiques offrent plusieurs avantages par rapport aux sites statiques. Ils permettent des mises à jour simplifiées via une interface d'administration, offrent une évolutivité pour intégrer de nouvelles fonctionnalités, favorisent l'interaction avec les utilisateurs et proposent des fonctionnalités avancées telles que les boutiques en ligne et les systèmes de gestion de contenu [4].

1.4 Architecture client/serveur [5] :

Le client-serveur est avant tout, une technique de dialogue entre deux processus, l'un, « le client » sous-traitant à l'autre, « le serveur » des fonctions à réaliser.

Le modèle de communication client/serveur est orienté vers la fourniture de services par un processus serveur à un processus client. Un échange consiste donc en la transmission d'une requête à un serveur, qui exécute l'opération demandée et envoie en retour la réponse. Nous définissons ci-dessous plus précisément ces concepts de bases [5].

- **Client :** (client) processus demandant l'exécution d'une opération à un autre processus
Par envoi d'un message contenant le descriptif de l'opération à exécuter et attendant la Réponse à cette opération par un message en retour.
- **Serveur :** (server) processus accomplissant une opération sur demande d'un client, et transmettant la réponse à ce client.
- **Réponse :** (reply) message transmis par un serveur à un client suite à l'exécution d'une Opération contenant les paramètres de retour de l'opération.
- **Requête :** (request) message transmis par un client à un serveur décrivant l'opération à Exécuté pour le compte client.

1.4.1 Avantages de l'architecture client/serveur :

Le modèle client/serveur présente de nombreux avantages pour les réseaux nécessitant un haut niveau de fiabilité :

- Centralisation des ressources : Le serveur centralise des ressources communes à tous les utilisateurs, comme une base de données, ce qui permet d'éviter la redondance et les contradictions dans les données.
- Sécurité accrue : Le nombre de points d'accès aux données étant réduit, la sécurité du réseau est renforcée.
- Administration simplifiée : La gestion du réseau est centralisée au niveau du serveur, ce qui facilite l'administration et la maintenance.
- Évolutivité : L'architecture client/serveur permet d'ajouter ou de supprimer des clients facilement sans perturber le fonctionnement du réseau.

1.4.1.1 Inconvénients de l'architecture client/serveur :

Malgré ses nombreux avantages, l'architecture client/serveur présente quelques inconvénients :

- Coût élevé : Le serveur est une machine puissante et sophistiquée, ce qui implique un coût d'achat et de maintenance élevé.
- Point de défaillance unique : Le serveur est un point de défaillance unique, ce qui signifie que si le serveur tombe en panne, tout le réseau est inutilisable.
- Dépendance au serveur : Les clients dépendent du serveur pour accéder aux ressources et aux services, ce qui peut limiter la flexibilité et l'autonomie des utilisateurs.
- Complexité accrue : La mise en place et la configuration d'une architecture client/serveur peuvent être complexes, surtout pour les réseaux de grande taille.

En conclusion, l'architecture client/serveur est une solution performante et flexible pour les réseaux de taille moyenne et grande. Cependant, il est important de prendre en compte ses avantages et ses inconvénients avant de l'adopter.

1.4.1.2 Fonctionnement d'un système client/serveur :

Un système client/serveur fonctionne suivant le schéma suivant de la Figure 1.2:

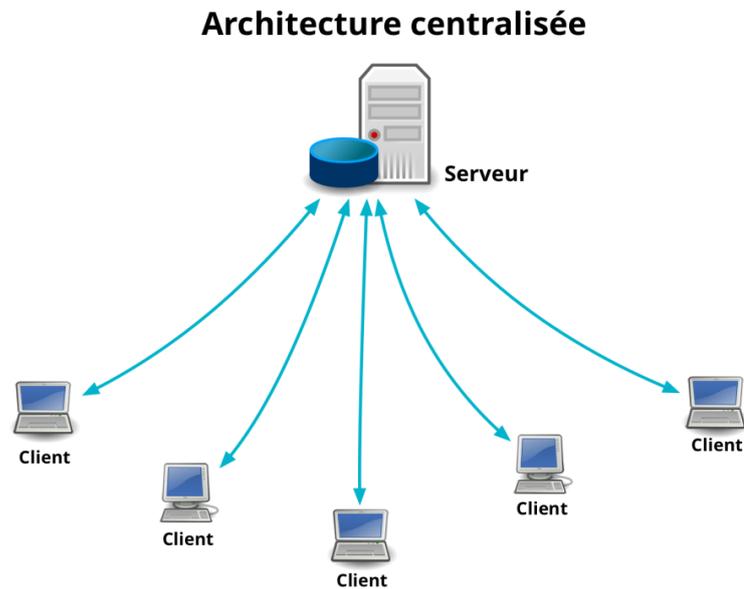


Figure 1.2 : Architecture client server [13]

1. Requête du client : Le client envoie une requête au serveur en précisant son adresse IP et le port correspondant au service demandé.
2. Réception de la requête : Le serveur reçoit la requête du client et l'analyse pour déterminer l'action à effectuer.
3. Traitement de la requête : Le serveur traite la requête en utilisant ses ressources et ses données.
4. Réponse au client : Le serveur envoie une réponse au client, contenant les résultats du traitement ou les informations demandées.
5. Réception de la réponse : Le client reçoit la réponse du serveur et l'affiche à l'utilisateur.

1.5 Conclusion

Ce chapitre souligne l'importance cruciale de l'évaluation, du feedback et des outils informatiques dans l'apprentissage des étudiants. Leur utilisation stratégique peut optimiser

l'expérience d'apprentissage et favoriser le succès académique, nécessitant une intégration réfléchie pour maximiser les avantages tout en minimisant les distractions. En parallèle, il est impératif de reconnaître que ces outils s'inscrivent dans une architecture plus vaste, comparable au modèle client-serveur du Web. Comprendre cette architecture sous-jacente permet de mieux concevoir et mettre en œuvre notre systèmes adaptés aux besoins évolutifs des apprenants.

Chapitre 2. ANALYSE ET CONCEPTION

2.1 Méthode d'analyse et de conception :

Avant de se lancer dans le développement d'une application, il est crucial de suivre une méthodologie rigoureuse pour garantir la précision, l'organisation et la réussite du projet. L'utilisation d'un langage de modélisation comme UML s'avère alors particulièrement utile. Pour cela on a opté pour le langage UML car il offre un ensemble de diagrammes et de notations pour représenter les différents aspects d'une application, du système aux interfaces utilisateur.

2.1.1 Présentation d'UML :

Définition UML :

UML, acronyme d'Unified Modeling Language, est un langage de modélisation graphique standardisé pour les systèmes orientés objet. Il offre un ensemble de notations et de diagrammes pour représenter visuellement les différents aspects d'un système, depuis sa structure interne jusqu'à son comportement et son interaction avec les utilisateurs [6].

2.1.1.1 Quelques définitions de base :

- Acteur : Un acteur représente un rôle joué par une personne ou un élément externe dans un système. Il peut interagir avec un cas d'utilisation et participer à son déroulement.
- Tâche : Une tâche est un ensemble d'actions qu'un acteur spécifique peut effectuer dans le système. Elle représente une unité fonctionnelle du cas d'utilisation.
- Scénario : Un scénario est une séquence d'étapes décrivant le déroulement d'un cas d'utilisation. Il représente un cheminement particulier que l'utilisateur peut emprunter pour atteindre son objectif.

2.1.1.2 Démarche :

a. Identificateur des Acteurs :

- Administrateur : L'administrateur a pour rôle la gestion de l'application, et sa tâche est la gestion des différents groupes utilisateurs, UE, question, en les modifiant, en les supprimant ou bien en ajoutant d'autre UE.
- Enseignant : sa tâche est la gestion des EC, question, étudiant et fournir des commentaires aux étudiants.
- Etudiant : Sa tâche est de s'inscrire au EC , consulter les EC et de passer les évaluations.

b. Les cas d'utilisation global :

Les cas d'utilisation global de l'administrateur :

- S'authentifier
- Gérer les groupes utilisateur
- Confirmer l'inscription des professeurs
- Gérer le contenu du système
- Gérer les UE
- Gérer les questions

Les cas d'utilisation global de Prétendant :

- Inscription

Les cas d'utilisation global de l'Enseignant :

- S'authentifier
- Gérer les EC
- Gérer les étudiants
- Fournir des commentaires aux étudiants

Les cas d'utilisation global de l'Étudiant :

- S'authentifier

- Inscription au EC
- Consulter le contenu des EC
- Passer des évaluations

c. Diagramme de cas d'utilisation global :

Les diagrammes de cas d'utilisation représentent un ensemble de cas d'utilisations, et d'acteurs et les différentes relations entre eux c'est la vue statique des cas d'utilisation d'un système.

La figure 2.1 représente le diagramme de séquence de notre système.

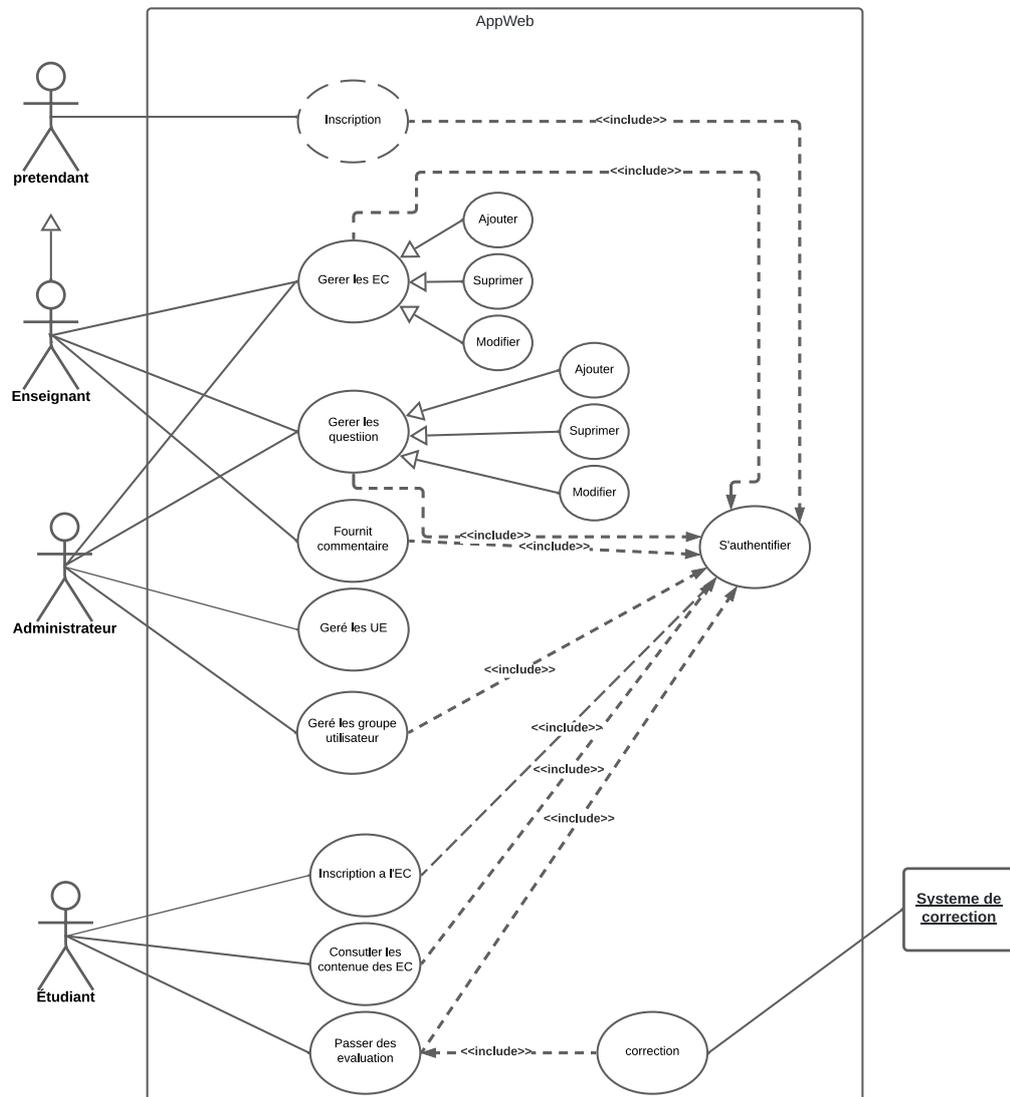


Figure 2.1 : Diagramme de cas d'utilisation

a. Spécification des cas d'utilisation globale :

Les cas d'utilisations d'un prétendant sont représentés dans le tableau 2.1.

Tableau 2.1 : cas d'utilisation du prétendant

Acteur	Cas d'utilisation
Prétendant	C1 : Inscription

Les cas d'utilisations d'un enseignant sont représentés dans le tableau 2.2.

Tableau 2.2 : cas d'utilisation de l'enseignant

Acteur	Cas d'utilisation
Enseignant	C2 : S'authentifier C3 : Gérer les EC C4 : Gérer les étudiants C5 : Fournir des commentaires aux étudiants C6 : Evaluer les performances des étudiants

Les cas d'utilisations d'un étudiant sont représentés dans le tableau 2.3.

Tableau 2.3 : cas d'utilisation de l'étudiant

Acteur	Cas d'utilisation
Etudiant	C7 : Inscription au EC C8 : S'authentifier C9 : consulte le contenu des EC C10 : Passer des évaluations

--	--

Les cas d'utilisations d'un administrateur sont représentés dans le tableau 2.4.

Tableau 2.4 : cas d'utilisation de l'administrateur

Acteur	Cas d'utilisation
Administrateur	C11 : s'authentifier C12 : Gérer les groupes utilisateur C13 : Confirmer l'inscription des professeurs C14 : Gérer le contenu du système C15 : Gérer les UE C15 : Gérer les EC C16 : Gérer les questions

Spécification des scénarios :

La figure 2.5 décrit le processus par lequel l'administrateur du système peut s'authentifier pour accéder à l'espace administrateur.

Tableau 2.5 : Accéder à l'espace administrateur

Cas d'utilisation	Accéder à l'espace administrateur
Acteur(s)	Administrateur, Professeur, Étudiant
Description brève	Ce cas d'utilisation permet à l'administrateur de s'authentifier dans l'espace administrateur du système.
Précondition	L'administrateur doit avoir accès à la page d'authentification.

Post condition	L'administrateur est redirigé vers l'espace administrateur en cas de succès. En cas d'échec, un message d'erreur est affiché.
Scenario de déroulement	
<ol style="list-style-type: none"> 1. L'administrateur accède à la page d'authentification. Ensuite il va voir l'état de toutes les lampes de sa maison et leurs états 2. Le système affiche un formulaire d'authentification. 3. L'administrateur saisit son identifiant et son mot de passe 4. En cas de succès, l'administrateur est redirigé vers l'espace administrateur 5. Le système vérifie les informations d'identification. 6. En cas de succès, l'administrateur est redirigé vers l'espace administrateur. 7. En cas d'échec, un message d'erreur approprié est affiché. 	
Scenario alternatif	
Aucune	

Le tableau 2. 6 décrit le processus par lequel l'administrateur peut gérer les groupes d'utilisateurs dans le système.

Tableau 2.6 : Gérer les groupes utilisateurs

Cas d'utilisation	Gérer les groupes utilisateurs
Acteur(s)	Administrateur

Description brève	Ce cas d'utilisation permet à l'administrateur de gérer les groupes utilisateurs.
Précondition	L'administrateur doit être authentifié dans l'espace administrateur.
Post condition	Les modifications apportées aux groupes utilisateurs sont enregistrées.
Scenario de déroulement	
<ol style="list-style-type: none"> 1. L'administrateur accède à l'espace administrateur. 2. Le système affiche les options de gestion des groupes utilisateurs. 3. L'administrateur choisit d'ajouter, modifier ou supprimer un groupe utilisateur. 4. Le système vérifie les informations d'identification. 5. Les modifications sont effectuées et enregistrées dans la base de données. 	
Scenario alternatif	
Aucune	

Le tableau 2. 7 décrit le processus par lequel l'administrateur peut confirmer ou non l'adhésion d'un prétendant.

Tableau 2.7 : Confirmation de l'inscription

Cas d'utilisation	Confirmation de l'inscription
Acteur(s)	Administrateur

Description brève	Ce cas d'utilisation permet à l'administrateur de confirmer ou refuser l'inscription d'un enseignant.
Précondition	Un professeur a soumis une demande d'inscription
Post condition	L'inscription du professeur est confirmée ou refusée.
Scenario de déroulement	
<ol style="list-style-type: none"> 1. L'administrateur accède à l'espace administrateur. 2. Le système affiche les demandes d'inscription en attente. 3. L'administrateur examine la demande d'inscription d'un prétendant. 4. L'administrateur accepte ou refuse l'inscription. 5. En cas d'acceptation, le prétendant devient enseignant et peut accéder au système. 	
Scenario alternatif	
Aucune	

Le tableau 2. 7 décrit le processus par lequel un prétendant s'inscrit.

Tableau 2.8 : cas d'utilisation inscription d'un prétendant

Cas d'utilisation	Inscription
Acteur(s)	Prétendant
Description brève	Ce cas d'utilisation permet à un prétendant au poste d'enseignant de s'inscrire.

Précondition	Avoir sélectionner l'option inscription.
Post condition	Inscription d'un prétendant.
Scenario de déroulement	
<ol style="list-style-type: none"> 1. Le prétendant accède à l'application web. 2. Le prétendant sélectionne l'option inscription. 3. Le prétendant entre ses information personnel. 4. Le prétendant soumet sa demande d'inscription. 	
Scenario alternatif	
Aucune	

Le tableau 2. 9 décrit le processus par lequel l'administrateur ou l'enseignant peut gérer les UE.

Tableau 2.9: Gestion des UE

Cas d'utilisation	Gérer les UE
Acteur(s)	Administrateur
Description brève	Ce cas d'utilisation permet à l'administrateur de gérer les UE.
Précondition	L'utilisateur doit être authentifié dans l'espace correspondant
Post condition	Les modifications apportées aux UE sont enregistrées.

Scenario de déroulement
<ol style="list-style-type: none"> 1. L'utilisateur accède à l'espace de gestion des UE. 2. Le système affiche les options de gestion des UE. 3. L'utilisateur choisit d'ajouter, modifier ou supprimer une UE. 4. Les modifications sont effectuées et enregistrées dans la base de données.
Scenario alternatif
Aucune

Le tableau 2. 10 décrit le processus par lequel l'administrateur ou l'enseignant peut gérer les EC.

Cas d'utilisation	Gérer les EC
Acteur(s)	Administrateur, Enseignant
Description brève	Ce cas d'utilisation permet à l'administrateur ou à l'enseignant de gérer les EC.
Précondition	L'utilisateur doit être authentifié dans l'espace correspondant
Post condition	Les modifications apportées aux EC sont enregistrées.
Scenario de déroulement	
5. L'utilisateur accède à l'espace de gestion des EC.	

<p>6. Le système affiche les options de gestion des EC.</p> <p>7. L'utilisateur choisit d'ajouter, modifier ou supprimer un EC.</p> <p>8. Les modifications sont effectuées et enregistrées dans la base de données.</p>
Scenari alternatif
Aucune

Le tableau 2. 11 décrit le processus par lequel l'administrateur ou l'enseignant peut gérer les questions.

Tableau 2.10 : Gérer les questions

Cas d'utilisation	Gérer les questions
Acteur(s)	Administrateur, Enseignant
Description brève	Ce cas d'utilisation permet à l'administrateur ou à l'enseignant de gérer les questions.
Précondition	L'utilisateur doit être authentifié dans l'espace correspondant
Post condition	Les modifications apportées aux questions sont enregistrées.
Scenari de déroulement	
1. L'utilisateur accède à l'espace de gestion des questions.	

<ol style="list-style-type: none"> 2. Le système affiche les options de gestion des questions. 3. L'utilisateur choisit d'ajouter, modifier ou supprimer une question. 4. Les modifications sont effectuées et enregistrées dans la base de données.
Scenari alternatif
Aucune

Le tableau 2. 12 décrit le processus par lequel l'enseignant peut fournir des commentaire ou feedback aux étudiants.

Tableau 2.11 : Fournir des commentaires aux étudiants

Cas d'utilisation	Fournir des commentaires aux étudiants
Acteur(s)	Professeur
Description brève	Ce cas d'utilisation permet à l'enseignant de fournir un feedback ou commentaire aux étudiants.
Précondition	L'enseignant doit être authentifié dans l'espace enseignant et avoir sélectionner un EC.
Post condition	Le feedback est envoyé à l'étudiant.
Scenari de déroulement	
<ol style="list-style-type: none"> 1. L'enseignant accède à l'espace feedback. 2. Le système affiche la liste des étudiants. 3. L'enseignant choisit l'étudiant à qui il vas envoyer un feedback. 	

<p>4. L'enseignant saisit le contenu du feedback.</p> <p>5. L'enseignant envoie le feedback.</p>
Scenari alternatif
Aucune

Le tableau 2. 13 décrit le processus par lequel l'étudiant peut s'inscrire à un ou plusieurs EC

Tableau 2.12: Inscription à un EC

Cas d'utilisation	Inscription aux EC
Acteur(s)	Etudiant
Description brève	Ce cas d'utilisation permet à l'enseignant de fournir un feedback ou commentaire aux étudiants.
Précondition	L'étudiant doit être authentifié dans l'espace étudiant et avoir vue la liste des EC.
Post condition	Inscription a un EC
Scenari de déroulement	
<p>1. L'étudiant accède à l'espace EC.</p> <p>2. Le système affiche la liste EC.</p> <p>3. L'étudiant clique sur s'inscrire à cote du nom du EC.</p> <p>4. Le système l'inscrit à cet EC.</p>	

Scenario alternatif
Aucune

Le tableau 2. 14 décrit le processus par lequel l'étudiant peut consulter les contenus d'un EC.

Tableau 2.13 : Consulter le contenu des EC

Cas d'utilisation	Consulter le contenu des EC
Acteur(s)	Étudiant
Description brève	Ce cas d'utilisation permet à l'étudiant de consulter le contenu des EC.
Précondition	L'étudiant doit être authentifié dans l'espace étudiant.
Post condition	L'étudiant a consulté le contenu du EC.
Scenario de déroulement	
<ol style="list-style-type: none"> 1. L'étudiant accède à l'espace des EC. 2. Le système affiche la liste des EC disponibles. 3. L'étudiant choisit un EC à consulter. 4. Le système affiche les questions du EC sélectionné. 	
Scenario alternatif	
Aucune	

Le tableau 2. 15 décrit le processus par lequel l'étudiant passe des évaluations.

Tableau 2.14 : Passer des évaluations

Cas d'utilisation	Passer des évaluations
Acteur(s)	Étudiant
Description brève	Ce cas d'utilisation permet à l'étudiant de passer des évaluations.
Précondition	L'étudiant doit être authentifié dans l'espace étudiant et avoir sélectionné un EC.
Post condition	Les réponses de l'étudiant sont enregistrées.
Scenario de déroulement	
<ol style="list-style-type: none"> 1. L'étudiant accède à l'espace des évaluations. 2. Le système affiche la liste des évaluations disponibles. 3. L'étudiant choisit une évaluation à passer. 4. L'étudiant répond aux questions de l'évaluation. 5. L'étudiant soumet ses réponses. 	
Scenario alternatif	
Aucune	

2.1.2 Conception :

Au cours de cette phase, une nouvelle perspective du modèle émerge. Cette perspective met en avant les modules et les éléments exécutables physiques, sans pour autant se focaliser sur la concrétisation du système. Elle repose sur :

- **Les diagrammes de séquence** : ils présentent la vue dynamique du système. L'objectif du diagramme de séquence est de représenter les interactions entre les objets en indiquant la chronologie des échanges. Cette représentation se réalise par cas d'utilisation.
- **Les diagrammes de classes** : représentent la vue statique des objets. Leur intérêt majeur est de modéliser les entités d'un système. Autrement dit, ils expriment les relations existantes entre les pages client et serveur. Dans ce qui suit on représentera quelques diagrammes de séquence et de classe globale, correspondant aux cas d'utilisation déjà décrits.

A. Diagrammes de séquences :

La figure 2.2 représente le diagramme de séquence du cas d'utilisation inscription.

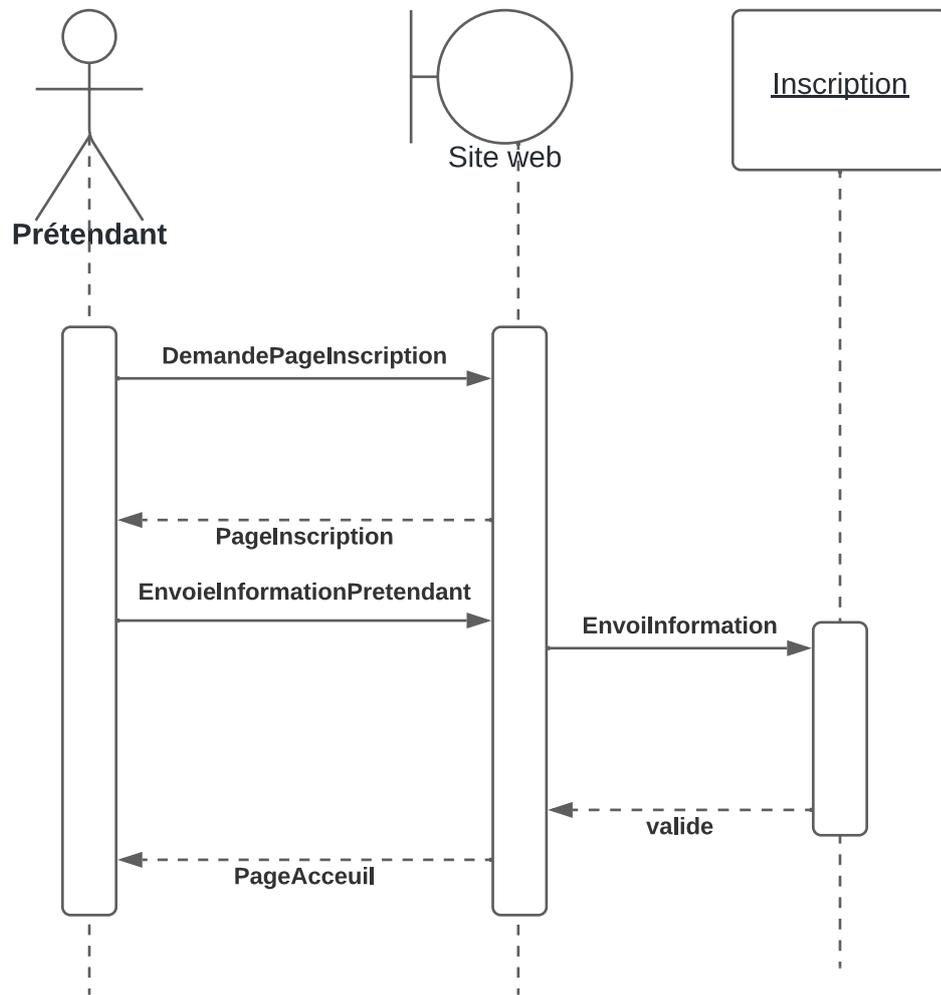


Figure 2.2 : Diagramme de séquence du cas d'utilisation inscription

La figure 2.3 représente le diagramme de séquence du cas d'utilisation confirmer inscription.

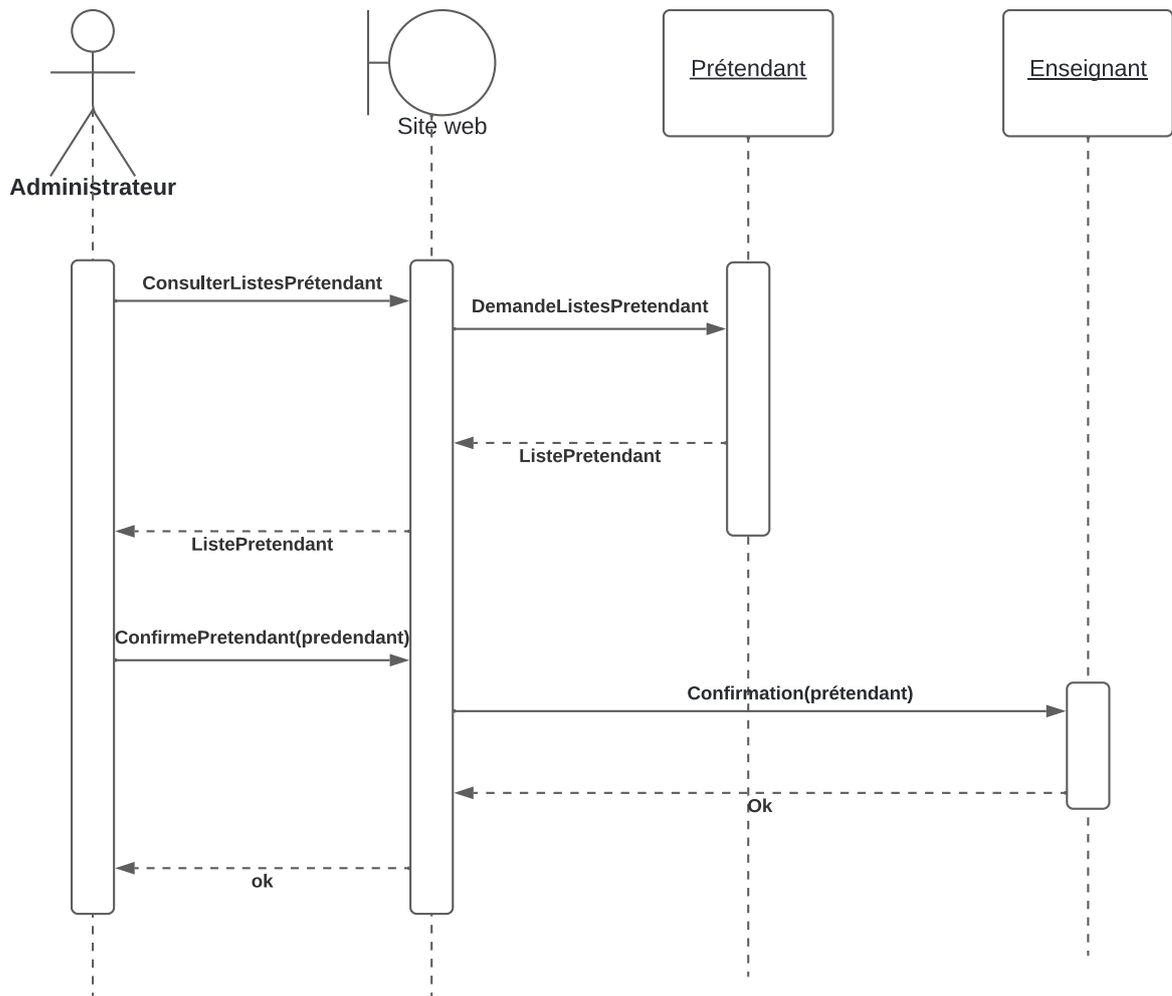


Figure 2.3 : Diagramme de séquence du cas d'utilisation confirmer enseignant

La figure 2.4 représente le diagramme de séquence du cas d'utilisation fournir commentaire ou feedback.

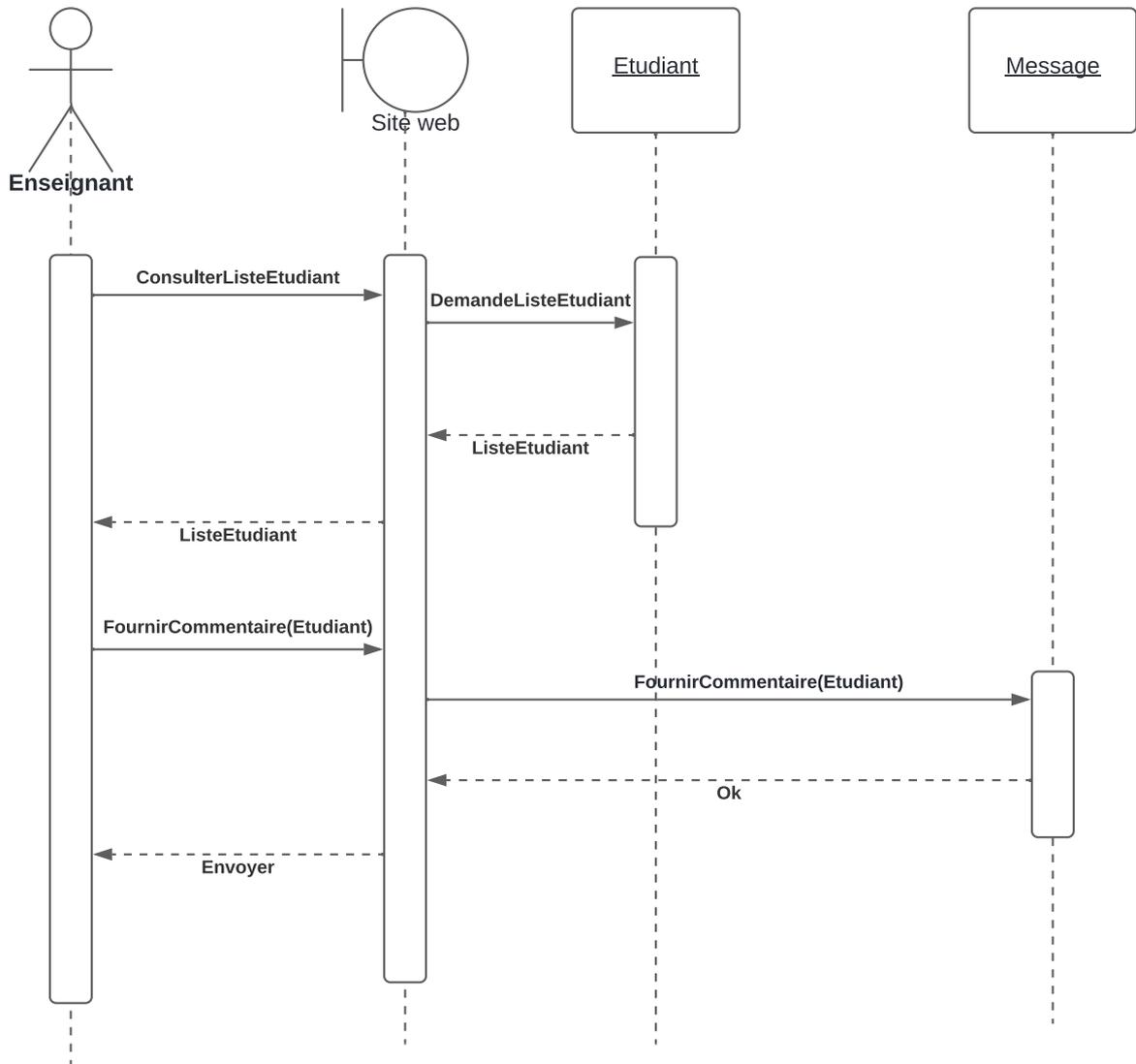


Figure 2.4 : Diagramme de séquence du cas d'utilisation fournir commentaire

La figure 2.5 représente le diagramme de séquence du cas d'utilisation consulter contenu EC et passer évaluation.

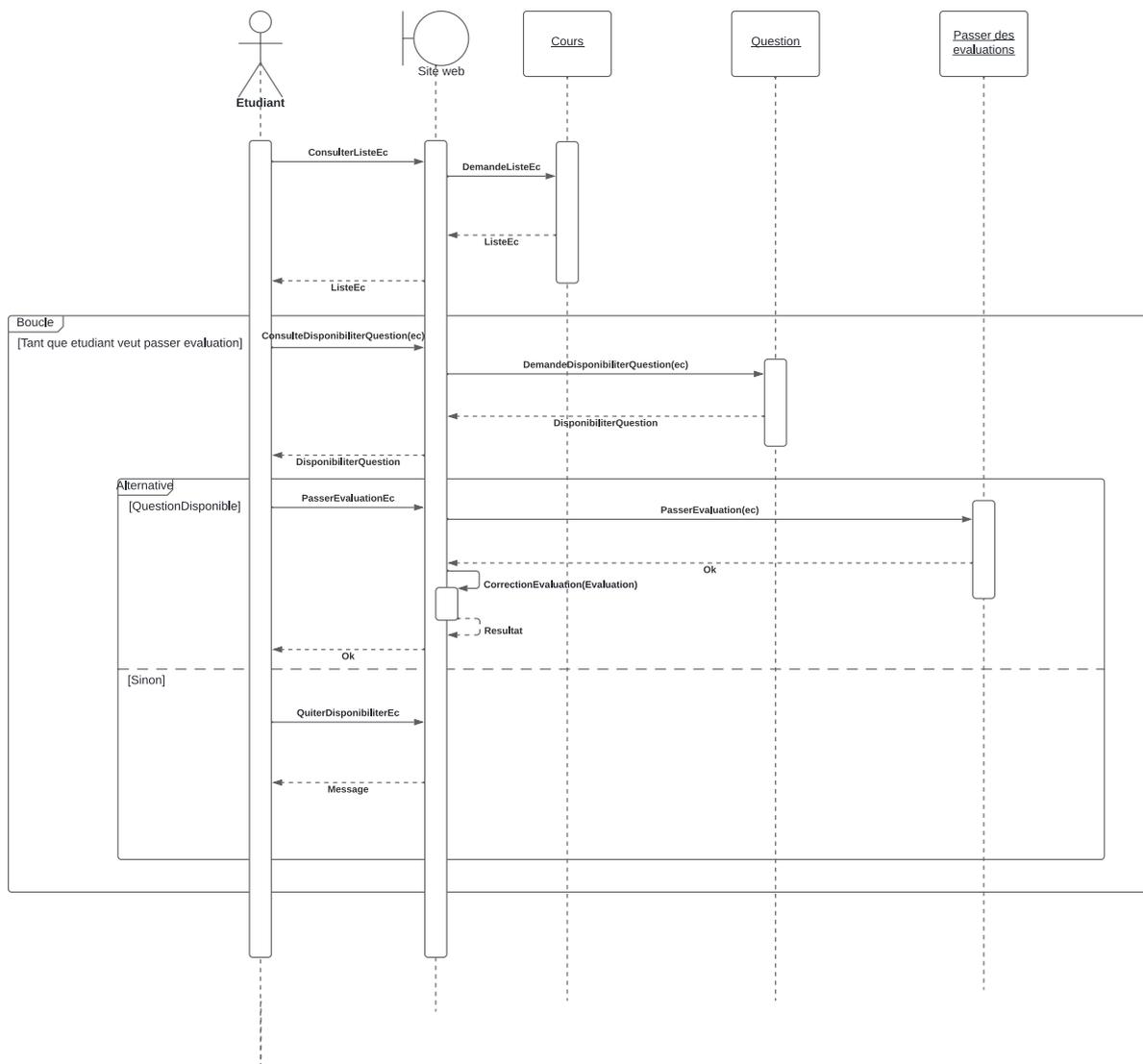


Figure 2.5 : Diagramme se séquence des cas d'utilisation consulter EC et passer évaluation

B. Conception de la base de données :

Dans cette section du chapitre, nous commencerons par décrire la méthode de conception de la structure de la base de données à travers un modèle conceptuel : le diagramme de classe (ou classe entité). Ensuite, nous procéderons à la transformation de ce modèle en un modèle relationnel optimisant l'implémentation de la base de données.

Le diagramme de classe globale du schéma conceptuel :

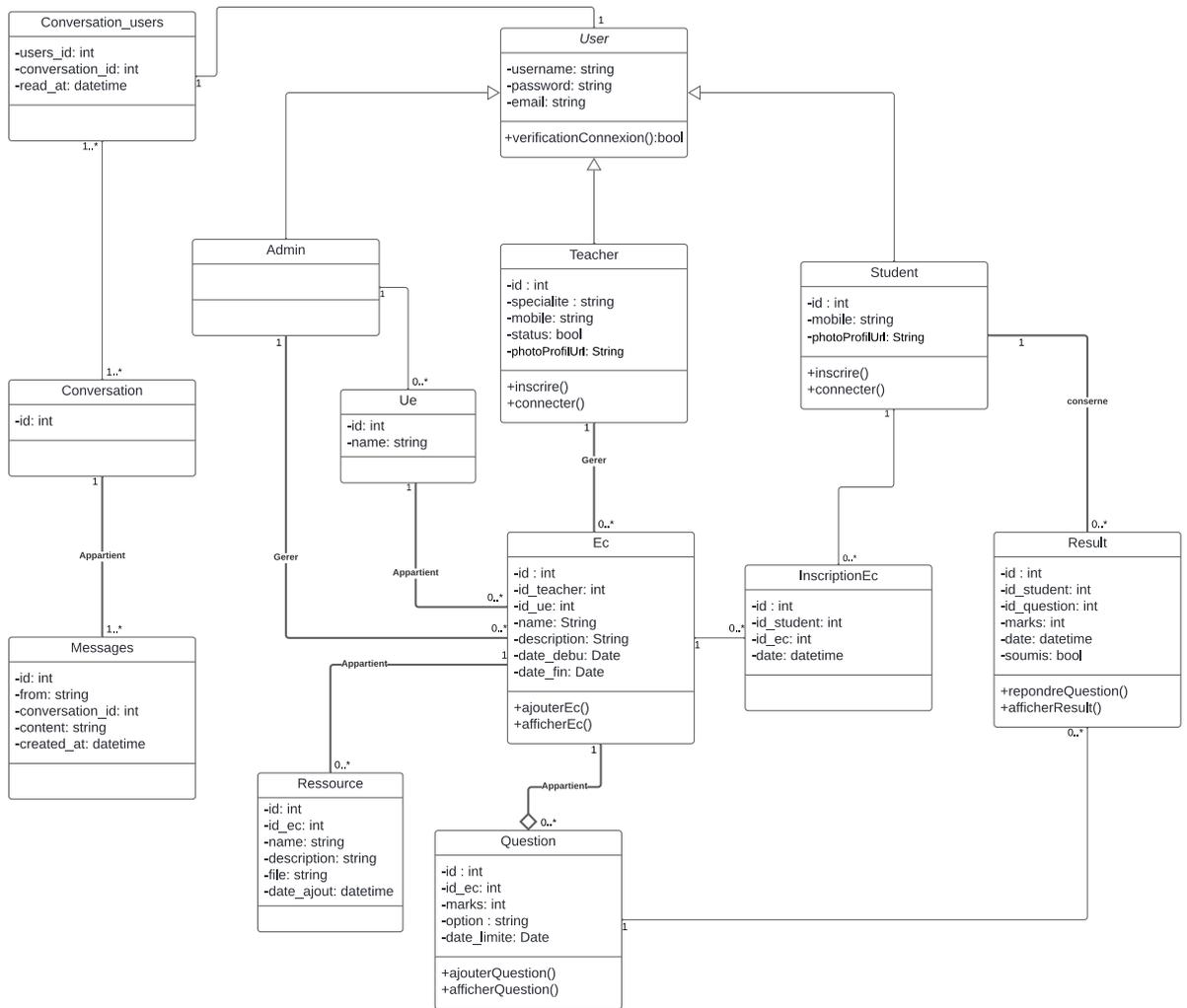


Figure 2.6 : Diagramme des classes

2.2 Mesure de Sécurité

La sécurité est un enjeu majeur pour la conception de notre application web destinée à la fourniture de feedback constructif dans une unité d'enseignement. Elle garantit la protection des données utilisateurs et prévient toute tentative d'intrusion ou de manipulation malveillante. Plusieurs stratégies sont mises en place pour sécuriser l'application et protéger les informations sensibles.

1. Authentification et gestion des utilisateurs

Pour garantir que seuls les utilisateurs autorisés (enseignants, étudiants, administrateurs) accèdent à l'application, un système d'authentification fort est utilisé. Ce système comprend une connexion par email et mot de passe, combinée à des mesures telles que la validation des comptes via email et des limites sur les tentatives de connexion pour prévenir les attaques par force brute.

2. Chiffrement des données et communication sécurisée

Le chiffrement est essentiel pour protéger les données sensibles lors de leur transfert et de leur stockage. Le protocole HTTPS est mis en place avec des certificats SSL/TLS pour assurer une communication sécurisée entre le serveur et les utilisateurs. De plus, les données stockées dans la base de données, telles que les feedbacks et les informations des utilisateurs, sont chiffrées pour éviter leur exposition en cas d'accès non autorisé.

3. Protection contre les attaques courantes

Des mesures spécifiques sont prises pour protéger l'application contre les attaques les plus fréquentes, comme les injections SQL et les failles XSS (Cross-Site Scripting). Cela inclut la validation stricte de toutes les entrées utilisateurs pour empêcher l'injection de code malveillant, ainsi que l'utilisation d'ORM (Object-Relational Mapping) pour interagir avec la base de données en toute sécurité. Les requêtes et formulaires sont également protégés contre les attaques CSRF (Cross-Site Request Forgery) en utilisant des jetons CSRF.

4. Gestion des permissions et des rôles utilisateurs

La gestion des permissions permet de contrôler l'accès aux différentes sections et fonctionnalités de l'application. Les utilisateurs sont classés par rôles (étudiant, enseignant, administrateur) et chaque rôle dispose d'autorisations spécifiques. Cela empêche les utilisateurs non autorisés d'accéder aux informations sensibles ou de modifier les feedbacks d'autres personnes.

5. Sauvegarde des données et audit des actions

Afin d'assurer la continuité des services et la sécurité des données, des sauvegardes régulières des données sont effectuées. En cas de perte de données, il sera possible de restaurer rapidement l'application. Par ailleurs, un système d'audit des activités enregistre les actions des utilisateurs (modifications, suppressions, accès aux données) afin de détecter d'éventuelles anomalies ou activités suspectes.

2.2. Conclusion

Dans ce chapitre, nous avons exploré la méthodologie d'analyse et de conception nécessaire pour développer notre application avec succès. En utilisant le langage de modélisation UML (Unified Modeling Language), nous avons pu représenter visuellement les différents aspects du système, de sa structure interne à son interaction avec les utilisateurs. Pour la phase d'analyse, nous avons conceptualisé les divers scénarios d'utilisation, puis nous les avons interprétés en élaborant des diagrammes de séquence et de classe lors de la phase de conception. Nous avons aussi abordé les mesures de sécurité pris en charge pour la protection de notre application. Le prochain chapitre se concentrera sur la mise en œuvre pratique de notre application, mettant ainsi en pratique les concepts élaborés jusqu'à présent.

Chapitre 3. REALISATION

3.1 Introduction :

Après avoir exploré la conception de notre application dans le chapitre précédent, nous nous concentrons désormais sur sa réalisation dans ce chapitre. Nous décrirons la création et la mise en œuvre des différents programmes ainsi que les interfaces principales par lesquelles les utilisateurs (administrateur, professeur et étudiant) interagissent avec le système et sa base de données. Ces éléments sont essentiels à la construction de notre application et garantissent son bon fonctionnement.

3.2 Outils de développement :

3.2.1 L'interface de Django Admin :

L'interface de Django Admin est une interface intégrée fournie par le framework Django pour l'administration des applications web. Cette interface conviviale permet aux administrateurs de gérer efficacement les données et les configurations de leurs applications Django via un navigateur web [7].

Voici quelques-unes des principales fonctionnalités de l'interface Django Admin :

- Gestion des modèles de données : création, modification et suppression des instances de modèles.
- Personnalisation des vues d'administration : possibilité d'ajouter, de supprimer ou de modifier des champs dans les formulaires d'édition.
- Gestion des utilisateurs et des groupes : ajout, modification et suppression des utilisateurs et affectation des permissions.
- Support multilingue : possibilité de gérer des applications dans différentes langues.
- Intégration avec l'authentification et l'autorisation Django pour un contrôle d'accès sécurisé.

- Génération automatique de l'interface d'administration à partir des modèles de données Django, réduisant ainsi le besoin de développement personnalisé.
- Cette interface robuste simplifie grandement la tâche des administrateurs en leur offrant un accès facile à toutes les fonctionnalités nécessaires à la gestion efficace de leurs applications Django.



Figure 3.1 : interface Django-admin

3.2.2 Les serveurs :

- **Serveur web Django :**

Pour héberger une application web Django, un serveur web est nécessaire pour gérer les requêtes des clients, servir les pages HTML aux navigateurs web et exécuter les programmes sur le serveur. Bien que Django puisse être déployé sur divers serveurs web, un choix populaire est le serveur web intégré fourni par Django lui-même lors du développement, appelé "Django Development Server".

- **Le serveur web Django est un serveur de développement léger** : conçu pour faciliter le processus de développement d'applications web Django. Voici quelques points à considérer à propos du serveur web Django :
- **Serveur de développement intégré** : Le serveur web Django est inclus avec le Framework Django, ce qui permet aux développeurs de commencer rapidement à

développer et tester leurs applications localement sans avoir besoin de configurer un serveur web tiers

- **Facilité d'utilisation :** Le serveur web Django est facile à utiliser et à configurer, ne nécessitant généralement qu'une simple commande pour démarrer le serveur.
- **Fonctionnalités de développement :** Bien qu'il soit principalement destiné à une utilisation en environnement de développement, le serveur web Django offre des fonctionnalités de débogage pratiques telles que la prise en charge du rechargement automatique lors de la modification du code source de l'application.
- **Limitations de performances :** Bien qu'adapté au développement local, le serveur web Django peut présenter des limitations en termes de performances et de capacités par rapport à des serveurs web plus robustes comme Apache ou Nginx. Il n'est donc généralement pas recommandé pour une utilisation en production à grande échelle.

3.2.3 IDE PyCharm

PyCharm est un environnement de développement intégré (IDE) créé par JetBrains, conçu spécifiquement pour le développement d'applications Python. Cet IDE offre une gamme complète d'outils pour la création, l'édition, le débogage et le déploiement d'applications Python [8].

L'IDE PyCharm nous a servi à structurer, organiser, implémenter et développer entièrement notre projet (application web).

La figure 3.2 montrer une image de l'IDE PyCharm.

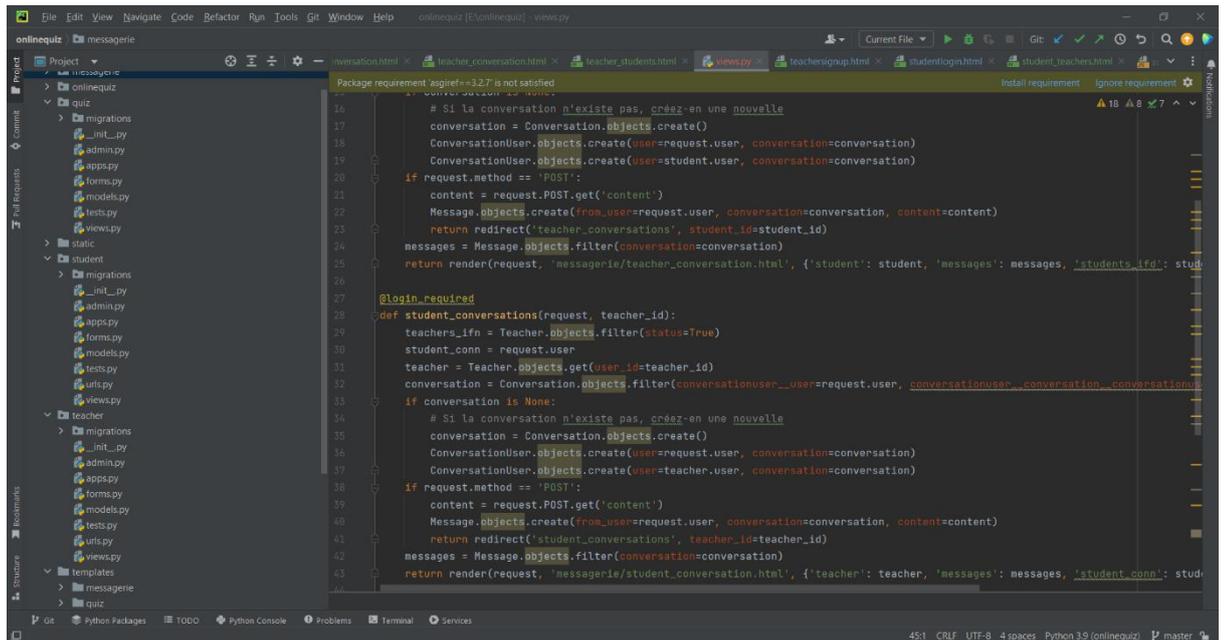


Figure 3.2: Aperçu d'une interface de l'IDE PyCharm

3.3 Les langages de programmation utilisés :

3.3.1. HTML (HyperText Markup Language) :

HTML est un langage de balisage dont la fonction est de structurer et de formater les pages web, en utilisant un serveur web et en définissant la présentation d'un document à l'aide de balises. Ces balises définissent la manière dont un document doit être affiché et les liens qu'il doit établir avec d'autres documents [9].

Ce langage permet la visualisation des documents sur internet à partir de différentes machines, grâce au protocole HTTP, qui permet l'accès à des documents via des réseaux en utilisant une adresse unique appelée URL.

Le langage HTML nous a servi à implémenter et à coder nos pages qui seront affichées par le navigateur aux utilisateurs.

3.3.2. Python :

Python est un langage de programmation interprété, de haut niveau et polyvalent, largement utilisé dans le développement logiciel, la science des données, l'intelligence artificielle, le développement web, et bien d'autres domaines. Conçu pour être simple et lisible, Python privilégie la lisibilité du code, ce qui en fait un choix populaire parmi les programmeurs débutants et expérimentés. Il offre une large gamme de fonctionnalités intégrées et de bibliothèques standard, ce qui facilite le développement d'applications complexes avec un minimum d'effort [10].

3.3.3. PostgreSQL :

PostgreSQL est un système de gestion de base de données relationnelle (SGBDR) robuste, puissant et extensible, souvent utilisé dans les applications web, les environnements d'entreprise et les projets nécessitant une gestion de données avancée. Contrairement à SQLite, PostgreSQL fonctionne en tant que serveur et offre des fonctionnalités étendues telles que la gestion avancée des utilisateurs, la réplication, la montée en charge et la prise en charge de données géospatiales. Bien que son déploiement nécessite une configuration initiale, PostgreSQL offre une grande flexibilité et des performances élevées, en faisant un choix privilégié pour les applications nécessitant une gestion de données robuste et évolutive [11].

3.3.4. JavaScript :

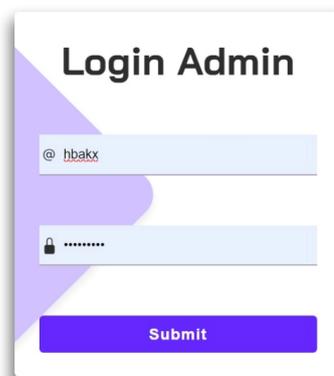
JavaScript, développé par Netscape en 1995, est le premier langage de script conçu pour le web. Intégré dans un document HTML, ce langage de programmation offre des fonctionnalités permettant d'enrichir les pages web en exécutant des instructions côté client, directement dans le navigateur plutôt que sur le serveur web. JavaScript facilite la validation des formulaires, la manipulation des fenêtres, ainsi que la gestion des éléments graphiques, offrant ainsi une expérience utilisateur interactive et dynamique [12].

3.4 Les interfaces de notre application :

Compte tenu du nombre important d'interfaces présentes dans notre système, nous allons ici présenter seulement quelques-unes des interfaces les plus pertinentes et représentatives.

- **Interface d'authentification (Administrateur) :**

Dans cette interface représentée par la figure 3.3, l'administrateur doit saisir son identifiant et son mot de passe afin d'accéder à son espace réservé. Cette méthode permet de contrôler l'accès à cette interface spécifique.



The image shows a web form titled "Login Admin". It features two input fields: the first is for an email address, containing "@ hbakx", and the second is for a password, masked with "*****". Below the fields is a blue "Submit" button. The form is set against a white background with a purple decorative shape on the left side.

Merci de nous avoir fait confiance et continuer d'utiliser notre site
Développé et créé par le groupe SoftMas-Arm, Report Abuse - Term of Service - Private Policy

SoftMas

Figure 3.3: interface d'authentification

- **Interface Dashboard(admin) :**

C'est dans cette interface représentée à la figure 3.4 et figure 3.5 que l'administrateur est redirigé après s'être authentifié avec succès.

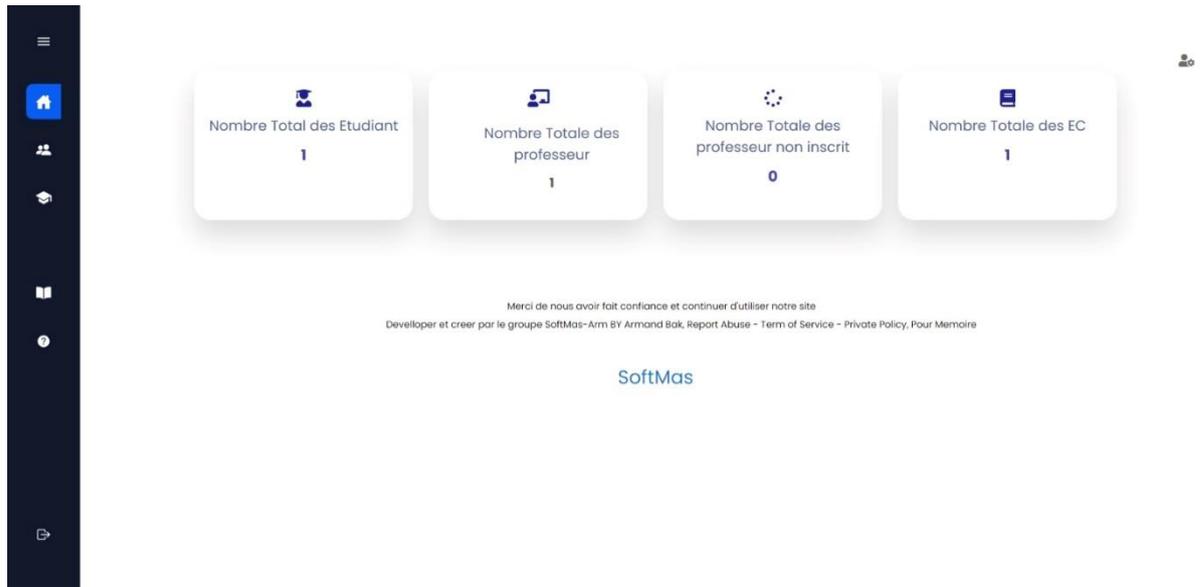


Figure 3.4: interface Dashboard(admin)

- **Interface Ajouter UE (administrateur) :**

C'est dans cette interface représentée à la figure 3.6 que l'administrateur va pouvoir ajouter une UE.

The form titled "Ajouter une Unité d'enseignement" contains the following fields:

- Section: "Information de l'unité d'enseignement"
- Field: "UE Name" with the value "GIN322" entered.
- Action: "Add" button.

The footer includes the same text as Figure 3.4: "Merci de nous avoir fait confiance et continuer d'utiliser notre site. Developeur et creer par le groupe SoftMas-Arm BY Armand Bak, Report Abuse - Term of Service - Private Policy, Pour Memoire". The SoftMas logo is centered at the bottom.

Figure 3.5: interface UE(professeur)

- **Interface inscription (professeur) :**

C'est dans cette interface représentée à la figure 3.7 que le professeur s'inscrit s'il n'a pas encore de compte et attend l'approbation de l'administrateur pour que son inscription soit validée.

Inscription

Information de l'enseignant

Username Firstname

Lastname Password

Mobile Specialiter

Picture Aucun fichier n'a été sélectionné

Merci de nous avoir fait confiance et continuer d'utiliser notre site
Developper et creer par le groupe SoftMas-Arm BY Armand Bak, Report Abuse - Term of Service - Private Policy, Pour Memoire

SoftMas

Figure 3.6: interface d'inscription professeur

- **Interface Ajouter un EC (Professeur) :**

C'est dans cette interface représentée par la figure 3.8 que le professeur peut ajouter un EC avant d'être redirigé vers l'interface de liste des EC.

Ajouter un Element constitutif

Information du EC

EC Name: Travaux de programmation

UE Name: GIN322

description: (dans ce cours nous allons apprendre a developper des logiciels)

Date du debut: 08/06/2024

Date de fin: 27/06/2024

Nombre Totale des point: 30

Add

Merci de nous avoir fait confiance et continuer d'utiliser notre site
Developper et creer par le groupe SoftMas-Arm BY Armand Bak, Report Abuse - Term of Service - Private Policy, Pour Memoire

Figure 3.7 : interface Ajouter un EC

- **Interface liste des EC (professeur) :**

C'est dans cette interface représentée par la figure 3.9 que le professeur peut voir la liste des EC qu'il a déjà créés.



Figure 3.8: interface liste des EC

- **Interface ajouter une question (professeur) :**

C'est dans cette interface représentée par la figure 3.10 que le professeur ajoute ou crée une question qui sera poser aux étudiants en indiquant aussi la réponse à cette question.

Figure 3.9: interface ajouter une question

- **Interface liste de mes EC déjà inscrit (Etudiant) :**

C'est dans cette interface représentée par la figure 3.11 que l'étudiant va voir la liste des EC auxquels il s'est déjà inscrit.

Figure 3.10: Interface liste des mes EC

- **Interface passer évaluation (Etudiant) :**

C'est dans cette interface représentée à la figure 3.12 que l'étudiant va pouvoir passer son évaluation et répondre aux questions.

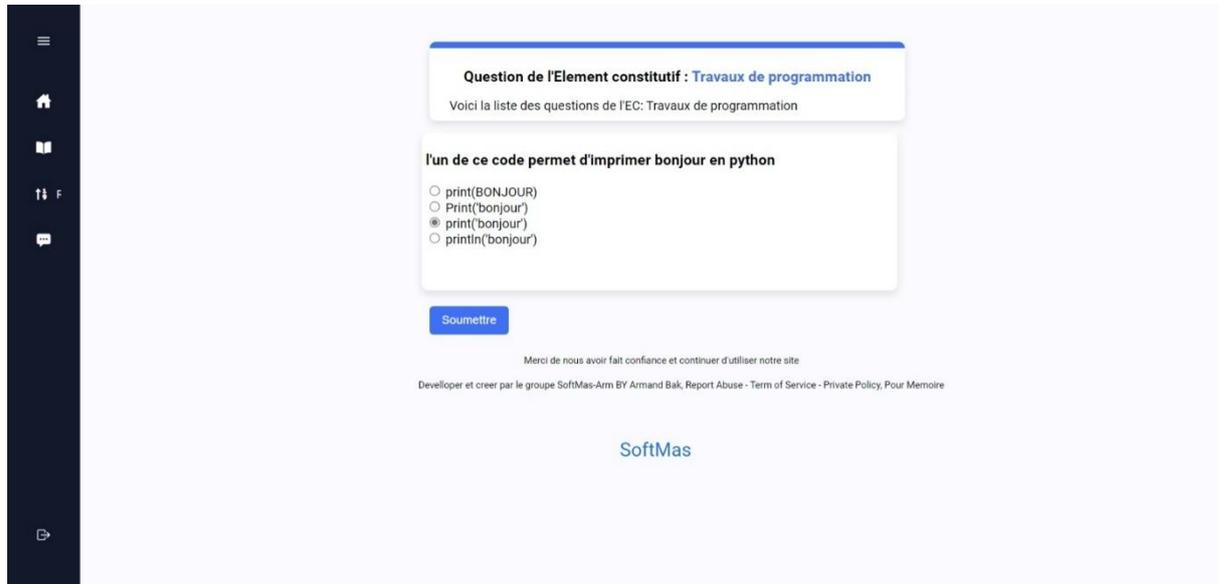


Figure 3.11: interface passer évaluation

- **Interface Résultat (Etudiant) :**

C'est dans cette interface représentée par la figure 3.13 que l'étudiant va voir les résultats de ses évaluations (points obtenus).

Exam Name ↑	Total Marks ↑	Attempt ↑	Exam Date ↑
Travaux de programmation	12	Attemp 1	June 6, 2024, 1:54 p.m.

Figure 3.12: interface résultats

- **Interface Liste EC (Professeur) :**

C'est dans cette interface représenté par la figure 3.14 que le professeur va pouvoir choisir un EC pour lequel il veut fournir un feedback avant d'être rediriger sur une liste des étudiant qui sont déjà inscrit au EC.

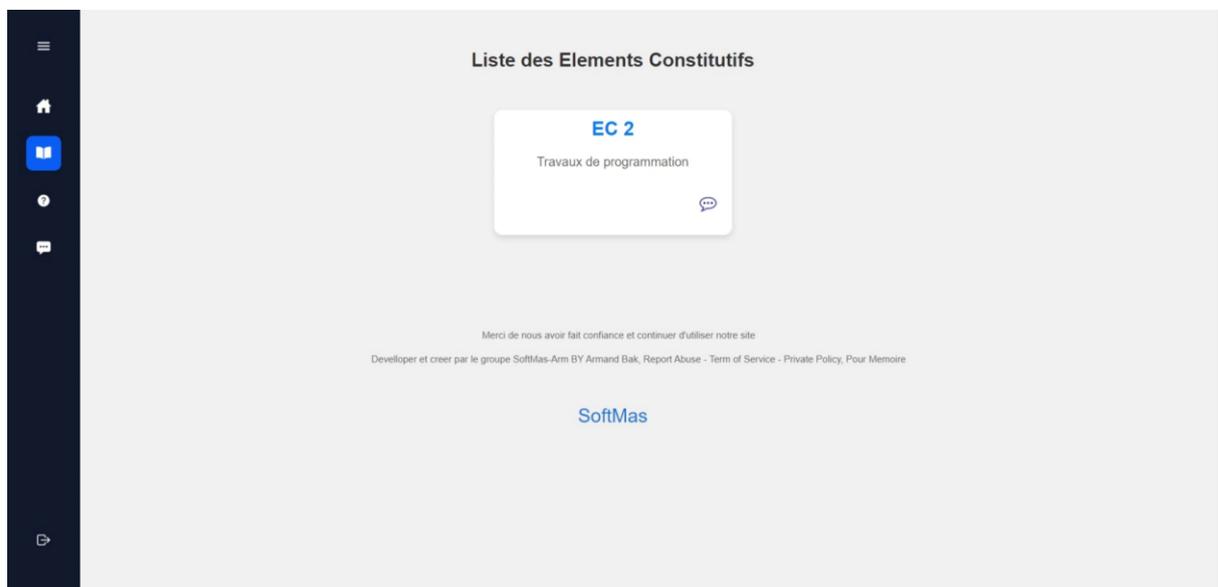


Figure 3.13: Interface liste EC

- **Interface Messagerie (Professeur) :**

C'est dans cette interface représentée par la figure 3.15 que le professeur va pouvoir envoyer un feedback à l'étudiant concernant son EC.

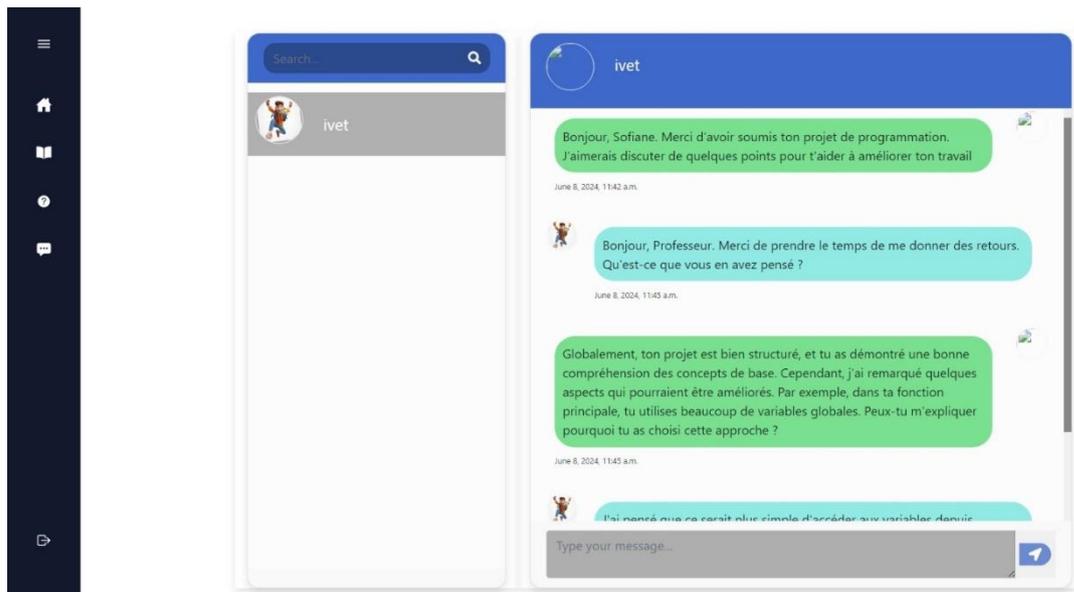


Figure 3.14: interface messagerie

Conclusion :

Dans ce chapitre, nous avons détaillé la réalisation de notre application web destinée à améliorer la fourniture de feedback constructif aux étudiants. En utilisant des technologies robustes telles que Django, Python, HTML, JavaScript, et PostgreSQL, nous avons développé une application intuitive et fonctionnelle. Les différentes interfaces pour les administrateurs, professeurs, et étudiants ont été conçues pour faciliter la gestion des UE, EC, la passation des évaluations, et la communication des feedbacks. Les tests effectués ont montré que notre application répond efficacement aux besoins identifiés, en simplifiant la tâche des enseignants et en enrichissant l'expérience d'apprentissage des étudiants. Ces résultats préliminaires

confirment la pertinence de notre approche et ouvrent la voie à une mise en œuvre plus large et à des améliorations futures.

CONCLUSION GENERALE

Au terme de ce travail intitulé « Conception et développement d'une application web pour la fourniture de feedback constructif dans une unité d'enseignement », nous avons élaboré une application qui facilite la gestion des interactions entre administrateurs, professeurs et étudiants, en mettant l'accent sur la fourniture de feedback constructif. Cette application, construite à l'aide du framework Django, offre une interface web conviviale pour la gestion des utilisateurs, des UE, EC, des évaluations et de la communication interne (feedback).

Bien que le système proposé ne soit pas une solution exhaustive intégrant toutes les fonctionnalités possibles de ce type de plateforme, nous avons néanmoins élaboré une application fonctionnelle. Cette application permet d'enregistrer les utilisateurs, de gérer les UE, EC et les évaluations, et de faciliter la fourniture et la réception de feedback constructif entre les différentes parties prenantes de l'UE.

Tout travail scientifique visant à résoudre un problème précis, l'application que nous avons conçue s'inscrit dans une démarche de simplification aidant les enseignants à fournir des feedbacks aux étudiants et à les évaluer. Ainsi, cette application permet une administration efficace et une interaction fluide entre les administrateurs, les professeurs et les étudiants, tout en mettant l'accent sur l'amélioration continue par le biais de feedback constructif.

Pour mettre en place ce système, nous avons d'abord étudié les technologies et frameworks web modernes, afin de maîtriser leurs fonctionnalités et de les utiliser de manière optimale. Ensuite, nous avons réalisé une conception et une analyse approfondies qui nous ont permis de développer le système tel que présenté dans ce travail. Les méthodes de développement logiciel que nous avons appliquées nous ont aidé à décrire entièrement le cycle de développement de notre système. Cette application web, intégrant des caractéristiques interactives et dynamiques, a bénéficié de l'ingénierie logicielle pour sa description graphique et textuelle.

Grâce à l'utilisation de Django Admin pour la gestion des données, de PyCharm comme IDE de développement, et des langages comme HTML, CSS, JavaScript et Python, nous avons créé

une application web capable de gérer diverses opérations éducatives et de stocker les informations dans une base de données relationnelle PostgreSQL. Cette approche permet aux utilisateurs de l'application d'interagir de manière sécurisée et efficace avec l'application, de gérer les UE, EC, les évaluations, de fournir et de recevoir des feedbacks constructifs, et d'obtenir les résultats souhaités.

Conscients que ce travail n'épuise pas toute la matière gravitant autour du sujet, nous ne prétendons pas à une performance finale. Puisque l'objectif global se réalise souvent par une succession d'amélioration, ce travail peut être élargi dans l'avenir. Les passionnés de ce domaine pourront améliorer cette application en y ajoutant des fonctionnalités pour le rendre plus attrayant, conviviale et conforme aux normes de standardisation comme l'intégration de l'intelligence artificiel, une analyse de donner plus pousser et bien plus encore.

Bibliographie

- [1] [En ligne]. Available: <https://ratingcaptain.com/fr/blog/quest-ce-que-le-feedback-et-comment-le-donner-correctement>. [Accès le 15 mai 2024].
- [2] A. Arnaud. e. Francois. Houste, 2017: Dunod.
- [3] S. Stormacq, La révolution des sites statiques, Eyrolles, 2020.
- [4] S. Mosser, Architecture des applications web : Concepts, techniques et outils, Dunod, 2021.
- [5] George. e. Olivier. GARDARIN, Le client Serveur, Eyrolles, 1996.
- [6] C. Morley. & J. Hugues. & B. Leblanc, UML pour l'analyse d'un système, Dunod, 2003.
- [7] «Django documantation | Django,» 13 05 2024. [En ligne]. Available: <https://docs.djangoproject.com/en/5.0/>.
- [8] 23 avril 2025. [En ligne]. Available: www.jetbrains.com.
- [9] J-P. Vincent. e. Mathieu. Nebra, HTML5 et CSS3 - Maîtrisez les standards de la création de sites web, Eyrolles, 2013.
- [10] G. Swinnen, Apprendre à programmer avec Python, Eyrolles, 2019.
- [11] T. Bruwn. e. H.-Jurgen. Schönig, PostgreSQL - Au coeur de l'administration, ENI, 2019.
- [12] O. Hondemarck, JavaScript - Les fondamentaux du langage, ENI, 2017.
- [13] [En ligne]. Available: [architectureCentralisee.png \(907×735\) \(coopernet.fr\)](http://architectureCentralisee.png).
- [14] R. GRIN, "Introduction aux bases de données, modèle relationnel", université de nice, 1999.
- [15] «site-web-statique,» 24 mai 2024. [En ligne]. Available: <https://www.hostinger.fr/tutoriels/site-web-statique>.
- [16] «comprendre-le-fonctionnement-dun-site-web-dynamique,» 24 mai 2024. [En ligne]. Available: <https://com-maker.fr/comprendre-le-fonctionnement-dun-site-web-dynamique/>.
- [17] «web-1-0-web-2-0-web-3-0-definition-et-differences,» 24 mai 2024. [En ligne]. Available: <https://easypartner.fr/blog/web-1-0-web-2-0-web-3-0-definition-et-differences/>.
- [18] Sloan. R. Weitzel, Un feedback constructif: comment présenter et faire passer votre message, Center for Creative Leadership, 2008.

- [19] D. Crockford, JavaScript : Les bons éléments, Pearson, 2013.
- [20] J. Hattie. e. Shirley. Clarke, Visible Learning : Feedback, Routledge, 2019.
- [21] J.-F. Nogier, Jean-François Nogier, Thierry Bouillot, Jules Leclerc, Dunod, 2013.
- [22] Julien. e. A. Augustin. Maupetit, Django : Développez des sites web python comme un pro, Eyrolles, 2018.
- [23] Maxwell. B. Chaterle, Maîtriser Django:Le guide complet du développement web, Amazon Digital Services LLC - Kdp, 2023.
- [24] P. Kruchten, Architecture logicielle : des processus et des hommes, Dunot, 2007.
- [25] J.-Philippe. Accart, L'histoire du Web : 25 ans de révolution, Eyrolles, 2015.
- [26] C. Grange. e. Fabrice. Havet, Conception et développement d'applications web : Architecture, performance et sécurité, Eyrolles, 2019.