

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A/Mira de Bejaïa



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Faculté des Sciences exactes Département d'Informatique

Rapport de fin d'études

Pour l'obtention du diplôme de Master en Informatique

Option génie logiciel

Thème

CONCEPTION ET REALISATION D'UN SITE D'APPRENTISSAGE

Réalisé par:

- M^r ALIM Yanis
- M^r BERROUA Ryad

Présenté devant le jury composé de:

- M^r. SAADI Mustafa (président)
- M^{lle} GASMI Bedrina (Examinatrice)
- M^r. DJEBARI Nabil (Encadreur)

Dédicaces

On dédie notre travail à tous ceux qui nous ont aidés pendant nos études universitaires.

On tient principalement à remercier nos chers parents.

nos frères, nos famille, sans eux, on n'aurais jamais pu accomplir ce travail.

A tous nos amis de promotion de 2^{ème} année Master GL.

A tous ce qui ont participé à notre réussite.

Remerciements

Je tiens également à remercier mes professeurs de nous avoir dispensé un enseignement de qualité durant les 5 années qu'ils ont passées à l'Université Abderrahmane mira de Bejaia.

Je tiens à remercier tout particulièrement M. DJEBARI (Notre Encadrant) pour nous avoir donné une grande part d'autonomie dans ce travail et nous avoir guidés à travers des pistes de réflexion riches et prometteuses.

Je tien aussi exprimer ma gratitude reconnaissance tous les membres de jury d'avoir accepté d'évaluer notre travail.

Je tiens également à remercier toute l'administration et tous ceux qui espèrent qu'un jour notre université sera dans ses meilleures conditions.

Enfin, je tiens à remercier tous nos proches qui nous ont aidé et inspiré dans ce parcours semé d'embûches, et les remercier pour leur aide dans ce travail.

Table des matières

Liste des figures	IV
Liste des tableaux	IV
Liste des abréviations	VIII
Introduction générale	1
Chapitre 1 : Etude préliminaire et analyse des besoins	3
1. Introduction	4
2. Contexte et définition du problème	4
3. Contribution et inspiration	5
4. Les fonctionnalités de quelques sites existants.....	6
5. Les besoins fonctionnels	7
5.1.La gestion de l'application	8
5.2.La gestion des cours	9
5.3.La gestion des tests.....	9
6. Les besoins non fonctionnels	10
7. Conclusion	11
Chapitre 2 : Analyse et conception	18
Introduction	19
1. Identification des acteurs.....	19
2. Identification des cas d'utilisation.....	20
3. Diagramme de cas d'utilisation pour chaque acteur.....	21
3.1Diagramme de cas d'utilisation Administrateur	22
3.2Diagramme de cas d'utilisation Formateur.....	23
3.3 Diagramme de cas d'utilisation élève	24
4. Description textuelle des cas d'utilisations	24
4.1 Description textuelle pour le cas « Authentification ».....	24
4.2 Description textuelle pour le cas « Gestion des cours »	27

4.3 Description textuelle pour le cas « Gestion des tests ».....	28
5. Diagramme de séquence.....	28
5.1 Diagramme de séquence	30
5.2 Diagramme de séquence « Gestion des tests »	31
6. Diagramme de classe.....	32
7. Dictionnaire de données	35
8. Model relationnel	36
Conclusion	37
Chapitre 3 : Réalisation.....	38
Introduction	39
1. Environnement de développement	39
1.1 Visual studio code	39
1.2 WampServer.....	40
1.3 ArgoUml.....	40
2. Adobe Illustrator.....	41
3. Technologie Logicielle.....	41
3.1 Présentation de HTML	42
3.2 Présentation de CSS.....	43
3.3 Présentation de Javascript	43
3.4 Présentation de MYSQL	44
4. Présentation des Frameworks utilisés.....	44
4.1 Définition du Framework.....	44
4.2 Présentation de Laravel	45
4.3 Présentation de Boostrap.....	46
5. Présentation de l'application	46
5.1 Interface« Authentification »	47
5.2 Interface« gestion des cours ».....	48

6.3 Interface« gestion des cours ».....	49
6.4 Interface« liste des cours ».....	50
6.5 Interface« gestion des tests »	51
7. Implémentation de notre application	52
7.1 Les routes	53
7.2 Les migrations.....	53
7.3 Les contrôleurs	55
7.4 Les model	56
7.5 Les vues.....	56
8. Conclusion	57
9. Conclusion général et perspective	58
10. Références Bibliographiques et webographies	60
11. Annexe.....	65
12. Résumé	67

Liste des figures

Figure2.1 : diagramme de cas d'utilisation Administrateur	12
Figure2.2 : diagramme de cas d'utilisation Formateur	13
Figure2.3 : diagramme de cas d'utilisation Elève	14
Figure2.4 :Diagramme de séquence de cas d'utilisation « Authentification ».....	20
Figure2.5 :Diagramme de séquence de cas d'utilisation « Recherche cours »	21
Figure2.6 :Diagramme de séquence de cas d'utilisation «Recherche test ».....	22
Figure2.7 : Diagramme de séquence de cas d'utilisation « Gestion des cours »	23
Figure2.8 : Diagramme de séquence de cas d'utilisation « Gestion des tests »	24
Figure2.9 :Diagramme de Classe	26
Figure3.1 : Interface Authentification	36
Figure3.2 :Interface gestion de cours	37
Figure3.3 : Interface consulter un cours	38
Figure3.4 :Interface liste des cours	39
Figure3.5 : Interface gestion des tests	40
Figure3.6 : image représentant l'architecture de notre application	41
Figure3.7 :image représentant le fichier web.php.....	42
Figure3.8 : image représentant la migration de la table utilisateurs.....	43
Figure3.9 :image représentant le contrôleur listCours.....	44
Figure3.10 : image représentant le modèle test	45
Figure3.11 : image représentant la vue listecours	45

Liste des tableaux

Tableau2.1 : Les acteurs du system et leur cas d'utilisation.....	10
Tableau2.2 : Description textuelle du cas d'utilisation Authentification	15
Tableau2.3 : Description textuelle du cas d'utilisation ajouter un cours.....	16
Tableau2.4 : Description textuelle du cas d'utilisation modifier un cours	16
Tableau2.5 : Description textuelle du cas d'utilisation supprimer un cours	17
Tableau2.6 : Description textuelle du cas d'utilisation gestion des tests	18
Tableau2.7 : Dictionnaire de données	25
Tableau3.1 : Description textuelle du cas d'utilisation ajouter utilisateur.....	52
Tableau3.2 : Description textuelle du cas d'utilisation supprimer utilisateur.....	52
Tableau3.3 : Description textuelle du cas d'utilisation Modifier accès au utilisateur..	53
Tableau3.4 : Description textuelle du cas d'utilisation passer les recherche cours.....	54
Tableau3.5 :Description textuelle du cas d'utilisation passer les tests.....	54
Tableau3.6 : Description textuelle du cas d'utilisation Contacte	65

Liste des abréviations

UML:UnifiedModelingLanguage : Langage de modélisation unifié.

HTTP: Hypertext Transfer Protocol: Protocole de transferthypertexte.

HTML:HyperText Markup Language.

CSS:Cascading Style Sheets.

MVC:Modèle-Vue-Contrôleur.

ORM:Object-relationalmapping : mapping relationnel objet.

PHP:HypertextPreprocessor.

Introduction générale

L'apprentissage est l'acquisition de savoir-faire, c'est-à-dire le processus d'acquisition de pratiques, de connaissances, compétences, d'attitudes ou de valeurs culturelles, par l'observation, l'imitation, l'essai, la répétition, la présentation. Il s'oppose, tout en le complétant, à l'enseignement dont le but est surtout l'acquisition de savoirs ou de connaissances au moyen d'études, d'exercices et de contrôles des connaissances.

Les nouvelles technologies, qui font partie aujourd'hui du quotidien des familles, constituent une mine d'information pour l'élève, un peu comme le dictionnaire. C'est un moyen pour les amener à s'intéresser aux sujets d'apprentissage et à vouloir aller plus loin. Il ne faut pas oublier que l'élève doit se mettre en situation d'apprentissage.

Sur le chemin menant à la personnalisation de l'apprentissage, la technologie autonomise les élèves en leur permettant de s'impliquer. Elle offre une éducation adaptée à leur vie numérique et prépare leur avenir. Avec des technologies et un accès aux ressources en dehors de la salle de classe, les élèves trouvent l'inspiration pour devenir des résolveurs de problèmes, des penseurs critiques, des collaborateurs et des créateurs. En effet, lorsque la technologie a été intégrée de façon appropriée aux classes, les étudiants développent une passion durable pour l'apprentissage, Alors comment utiliser ces nouvelles technologies afin de faciliter l'apprentissage et améliorer l'enseignement.

L'apprentissage en ligne est une forme d'éducation moderne qui a émergé avec le développement de la révolution numérique et la diffusion d'internet, tout comme elle a touché divers secteurs, l'éducation a eu une part importante dans ce développement, qui peut être comme une révolution dans les méthodes éducatives et les possibilités de sa prestation.

L'objectif de l'apprentissage en ligne est de remplacer les anciennes méthodes (temps /Place /contenu) de l'apprentissage prédéterminé avec des processus d'apprentissage rapides, ouverts, personnalisés.

C'est dans ce contexte que s'intègre notre projet de fin d'étude qui a pour objectif de mettre en place un site d'apprentissage en ligne.

Ce rapport définit trois chapitres le premier chapitre contiendra à l'étude et spécifications des besoins afin de délimiter notre problématique et les objectifs visés. Nous aurons pour aspect de définir des besoins fonctionnels et non fonctionnels qui sont considérés comme des fonctionnalités du système ou les besoins spécifiant un comportement d'entrée / sortie du système.

Au deuxième chapitre nous montrerons l'analyse et la conception du système qui contiendra différents diagramme et cela avant d'entamer la partie applicative dans le but de bien définir, clarifier les fonctionnalités importantes de l'application, En effet dans ce chapitre on a opté pour la modélisation UML avec le processus UP pour mieux anticiper les résultats du développement de l'application.

Le troisième chapitre se concentrera sur la mise en œuvre de notre application Web, y compris la conception de l'architecture et l'environnement de développement, et nous présenterons en détail les différentes étapes de la mise en œuvre. Ce chapitre est la dernière partie de ce rapport.

Nous terminerons ce rapport par une conclusion générale, dans laquelle nous fournirons également quelques perspectives pour rendre notre travail meilleur, plus ciblé et segmenté pour des groupes de personnes spécifiques.

Chapitre 1
Etude préliminaire
Et
Analyse des besoins

1. Chapitre 1 : Etude préliminaire et analyse des besoins

Introduction :

Dans ce chapitre nous allons décrire le livre dont nous nous sommes inspirés pour réaliser ce projet et nous allons par la suite poser la problématique et les solutions proposées et qui seront facilement adaptés par l'utilisateur.

Nous allons aussi définir les besoins fonctionnels de notre système et son bon fonctionnement seront décrits par la suite comme étant des besoins non fonctionnels.

1.1. Contexte et définition du problème:

L'explosion d'internet et son accès au plus grand nombre permet d'imaginer aujourd'hui des formations accessibles dans de multiples domaines que ce soit d'un point de vue universitaire ou encore dans le cadre de la formation continue, l'utilisateur peut profiter d'un accès pratiquement illimité à des documents éducatifs ou amusants et à des services qui permettent de communiquer avec des personnes d'autres pays et d'autres cultures pour s'enrichir. De plus l'arrivée de l'ordinateur multimédia et des réseaux informatiques a ouvert la porte à de nombreuses possibilités d'apprentissage. On peut citer dans cette catégorie des sites tels Udemy, Skillshare, Lynda...etc. Mais pour un enfant de 6 ans lire et étudier en ligne lui seront difficiles. Cependant comment permettre aux jeunes enfants de naviguer et agréablement, comprendre les cours et effectuer des tests ?

La réponse à ce problème est d'utiliser les outils mis à disposition du développeur par les navigateurs web et manipulables avec des langages de programmation tel que JavaScript afin de guider l'enfant sur le site web. Je citerai des exemples d'outils comme la possibilité d'utiliser du son sur son site web que ce soit des musiques ou bien des voix off aideront l'enfant à se concentrer, l'utilisation des animations, des boutons clignotants, des images SVG.... Permettront de guider l'enfant à naviguer facilement sur le site web.

En faisant quelques recherches sur internet et en s'inspirant des sites web déjà existants nous avons travaillé sur cette application étape par étape afin de concevoir un site d'apprentissage en ligne pour des enfants ayant entre 5 et 8 ans, nous expliquons les étapes suivies dans les chapitres suivants.

1.2. Contribution et inspiration :

Afin de créer un site web qui répond au problème posé ci-dessus, nous avons fait des recherches et posé des questions auprès des parents d'élèves afin de comprendre les difficultés auquel font face leurs enfants que ce soit à l'école ou bien à la maison et plus de la moitié étaient d'accord avec la question suivant « pensez-vous qu'un site d'apprentissage en ligne aidera vos enfants à s'améliorer ? ». Nous avons aussi cherché dans le programme de l'éducation nationale dédié aux élèves de première année, ainsi que dans le livre « Pour comprendre les Mathématiques, 2011, Hachette éducation », où nous avons pu trouver des cours, exercices et des idées pour la création de contenu.

Pour ce qui concerne des design web et de la hiérarchie du site web on s'est inspiré des sites déjà existants et proposant des cours pour enfants sous forme de jeux, des quiz animés.. On peut citer des sites tels que LUMNI (www.lumni.fr),

IEDUCATIF (www.jeux.ieducatif.fr) et LOGICIELEDUCATIF (www.logicieleducatif.fr)

1.3. Les fonctionnalités de quelques sites d'apprentissage existants :

1.3.1. Le site «lumni » :

Lumni est un service proposé par l'ensemble des acteurs de la filière audiovisuelle publique (Arte, France Médias Monde, France Télévisions, INA, Radio France et TV5Monde), proposant des services pédagogiques aux étudiants, enseignants et éducateurs.

Ce site offre plus de 12 000 vidéos, audio, jeux, dossiers. Il propose aux élèves de développer leur culture générale, de prolonger leurs cours, de comprendre le monde en apprenant autrement, quant aux enseignants de préparer leur cours, enfin il offre des ressources aux éducateurs pour animer leur atelier.

Permet ces différentes fonctionnalités :

- Connexion/déconnexion.
- Recherche soit vidéo, thème, sujet.
- Consulter les cours
- Partager les cours
- Favoriser les cours
- S'inscrire aux newsletters.

- Publier des cours.
- Télécharger les cours.

1.3.2. Le site « Maxicours » :

Maxicours.com est une plateforme d'accompagnement scolaire en ligne fondée en 2000 par Benjamin Magnard, fils de Roger Magnard créateur des "Cahiers de Vacances".

Fort d'une expérience de 20 ans dans ce domaine, Maxicours est devenu la référence du soutien scolaire pour toutes les classes, avec un service de professeurs en ligne ainsi que des espaces de préparation aux examens.

Les fonctionnalités de Maxicours sont :

- Connexion/déconnexion
- Recherche par catégories, niveaux ou bien matière
- Consulter les cours après avoir payé l'abonnement mensuel
- Consulter des cours sous format multimédia
- Passer des examens sous forme de quiz
- Commenter et évaluer les cours consulté
- Télécharger le contenu des cours sous format PDF
- Commenter et évaluer les cours consulté
- Télécharger le contenu des cours sous format PDF

1.4. Les besoins fonctionnels :

Les besoins fonctionnels représentent les principales fonctionnalités du système. Ces besoins proviennent généralement des utilisateurs du système. Cette application devra permettre :

1.4.1. La gestion de l'application :

L'administrateur après l'authentification peut :

Gérer les actualités de la page (ajouter, modifier ou supprimer le contenu) Gérer les Formateurs (ajouter, modifier ou supprimer un formateur) Consulter l'espace des élèves Consulter l'espace

des Formateurs Gérer les cours et les vidéos relatifs à la formation et enfin Gérer les programmes d'enseignement.

1.4.2. La gestion des cours :

Le formateur après son authentification va pouvoir créer son espace d'apprentissage comme si il possède une classe virtuelle.

Cette partie permet de présenter des cours associés à une matière particulière, ces cours vont être des leçons sous forme de texte suivis d'une vidéo et d'un bloc pour l'entraînement une fois le cours créé il peut le modifier ou bien le supprimer.

1.4.3. La gestion des tests :

Une fois le formateur crée un cours particulier ce dernier contient des chapitres, le formateur doit posséder un espace personnalisé pour créer les tests associés à ces chapitres qui peuvent être soit des questions directes avec des animations ou bien des QCM une fois les tests créés il peut les modifier ou les supprimer.

1.5. Les besoins non fonctionnels :

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système, mais plutôt ils identifient des contraintes internes et externes du système.

Les principaux Besoins non fonctionnels de notre site se résument dans les points suivants :

- **Authentification :** Afin d'assurer la sécurité et limiter le nombre des utilisateurs, Le système exige aux utilisateurs de saisir le login et le mot de passe pour accéder au système.
- **Les exigences de qualité :**
 - ❖ Le choix des couleurs doit avoir signification par rapport au domaine ;
 - ❖ La taille et la police des textes doivent être facilement lisibles ;
 - ❖ Les couleurs doivent être harmonieuses et liées.

- La sécurité : Vu que les données manipulées par notre application sont critiques, nous devons garantir une sécurité optimale. Ainsi, les droits d'accès au système doivent être bien attribués, afin d'assurer la sécurité des données.
- La simplicité :
 - ❖ Garantir un certain repérage, aider l'utilisateur à se localiser dans l'application ;
 - ❖ Familiariser rapidement l'utilisateur avec l'application ;
 - ❖ Simplifier la navigation, elle doit être simple et intuitive.
- La rapidité du traitement : En effet, vu le nombre important des transactions quotidiennes, il est possible du temps réel. Impérativement nécessaire que la durée d'exécution des traitements s'approchent le plus.
- La performance : Un site doit être avant tout performant c'est-à-dire à travers ses fonctionnalités, il doit répondre à toutes les exigences des usagers d'une manière optimale.
- Ergonomie : L'application doit offrir une interface simple et facile d'utilisation.
- Fiabilité et possibilité de mise à jour des informations.

Conclusion :

Dans ce chapitre, nous essayons de mener des recherches préliminaires sur notre projet, le but est de faire réflexions sur l'apprentissage des élèves et diverses contraintes de développement.

Par conséquent, la modélisation peut comprendre différentes techniques d'enseignement, et peut identifier les critiques afin qu'elles puissent proposer des solutions en construisant un modèle d'apprentissage et enfin en créant une application.

Afin de réaliser une application d'apprentissage détaillée une analyse et une conception fera l'objet du chapitre suivant.

Chapitre 2
Analyse
Et
Conception

2. Chapitre2 : Analyse et conception

Introduction :

Avant de commencer à coder la partie application, nous nous intéressons à la phase de conception pour bien définir et clarifier les principales fonctions de notre application.

La conception générale et détaillée sera implémentée en UML (UnifiedModelingLanguage). En effet, UML nous permet de mieux concevoir des applications et nous fournit une décomposition et une identification claires des entités et classes nécessaires.

2.1. Identification des acteurs :

- ✓ **L'administrateur** : Il possède le droit d'accéder aux différentes rubriques de l'application afin d'attribuer des niveaux d'accès pour chaque utilisateur au niveau de son profil afin de lui permettre ou pas d'agir sur certaines données au niveau d'un module donné dans un établissement donné.
- ✓ **Le formateur** : le formateur lui-même peut être administrateur il permet de créer des tests pour les élèves afin d'évaluer leurs niveau et de créer des cours spécifique associés à une matière particulière.
- ✓ **L'élève**: utilisateur qui permet de consulter les différents cours disponibles et passer les tests reliés à ces cours.

2.2. Identification des cas d'utilisation :

Le tableau suivant représente les acteurs du système et leurs cas d'utilisation :

N°	Cas d'utilisation		Acteur
1	Authentification		Administrateur, Formateur, élève
2	Gestion des Utilisateurs	Ajouter	Administrateur
		Modifier	
		Supprimer	

3	Gestion des cours	Ajouter	Administrateur, Formateur
		Modifier	
		Supprimer	
4	Gestion des tests	Ajouter	Administrateur
		Modifier	
		Supprimer	
5	Recherche cours		Elève
6	Consulter les cours	Commenter les cours	Elève
7	Passer les tests		Elève
8	Prise de contacte		Elève, Formateur

Table2.1 : Les acteurs du system et leur cas d'utilisation

2.3. Diagramme de cas d'utilisation pour chaque acteur :

Comprendre la fonctionnalité à mettre en œuvre est essentiel pour créer un diagramme de cas d'utilisation pour une application, Dans cette section, nous détaillerons le diagramme de cas d'utilisation pour chaque acteur.

2.3.1. Diagramme de cas d'utilisation Administrateur :

La figure2.1 montre le diagramme de cas d'utilisation de l'administrateur, on voit qu'il possède trois cas d'utilisation tels que la gestion des utilisateurs, gestion des cours et enfin la gestion des tests, ces trois derniers incluent une authentification.

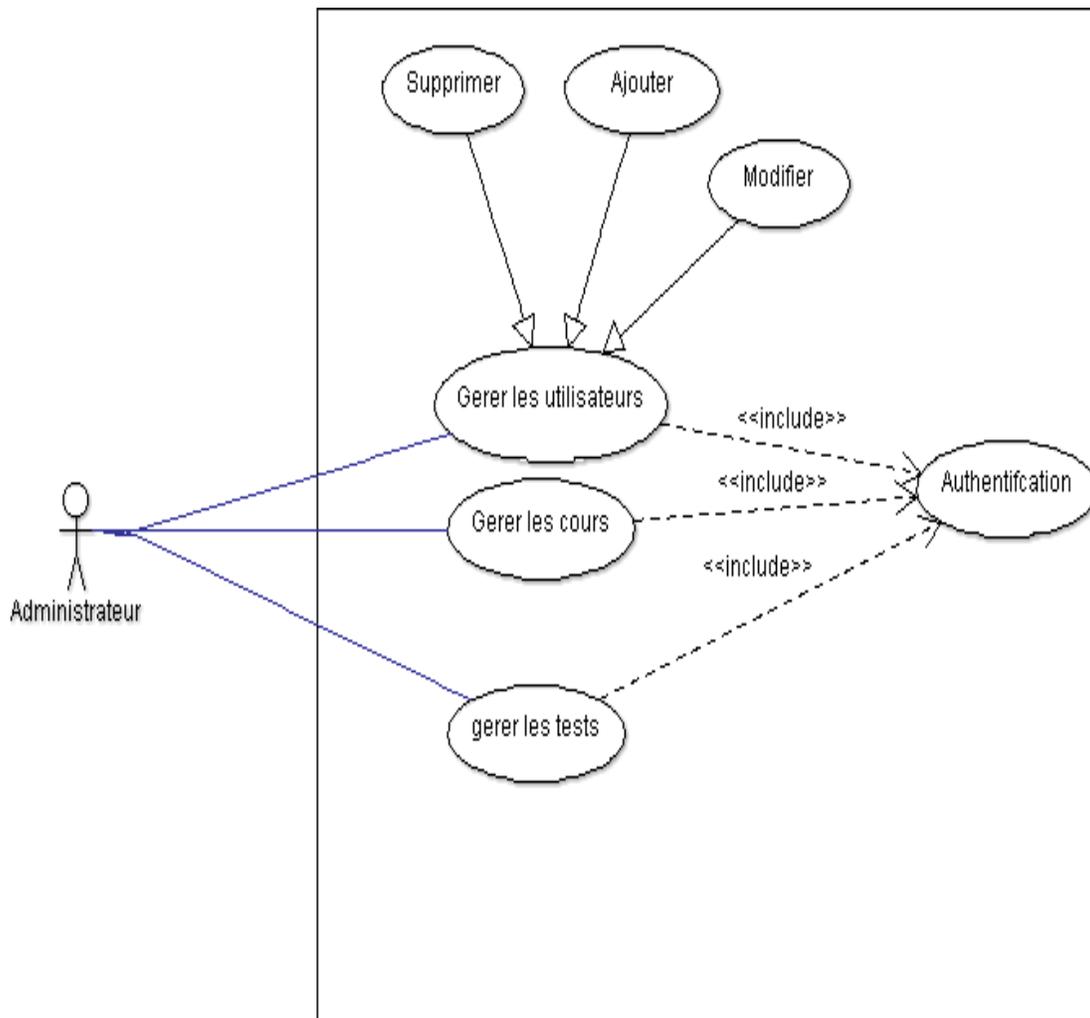


Figure2.1 : diagramme de cas d'utilisation Administrateur

2.3.2 Diagramme de cas d'utilisation Formateur :

La figure 2.2 montre le diagramme de cas d'utilisation du formateur, on voit qu'il possède trois cas d'utilisation tels que la gestion des utilisateurs, gestion des cours et enfin contacte, ces trois dernier inclus une authentification.

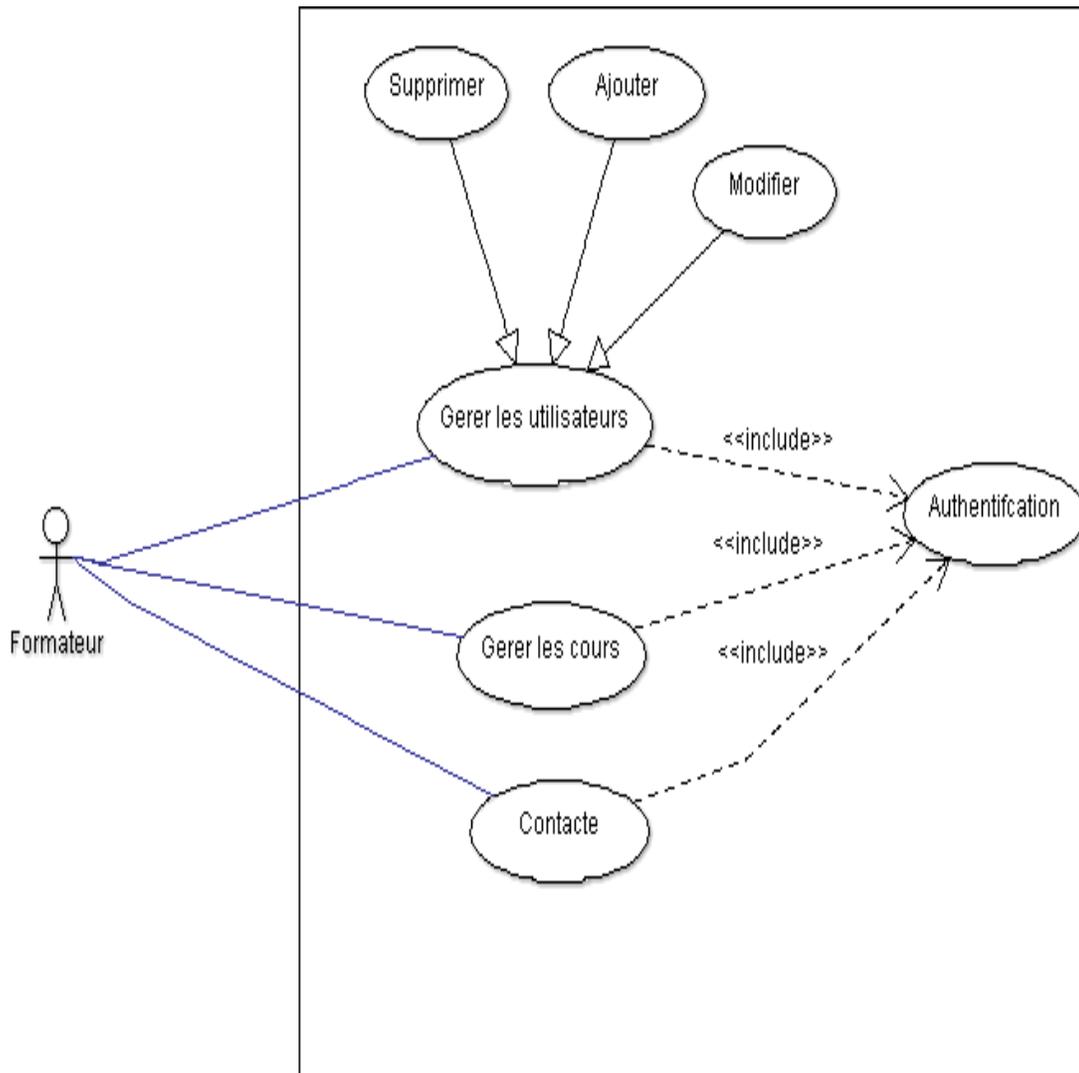


Figure 2.2 : diagramme de cas d'utilisation Formateur

2.3.3 Diagramme de cas d'utilisation Elève :

la figure 3 si dessous montre le diagramme de cas d'utilisation de l'élève, on voit qu'il possède quatre cas d'utilisation tels que recherche cours, commenter les cours étend du cas consulter les cours et enfin contacte des tests, ces quatre dernier inclus une authentification.

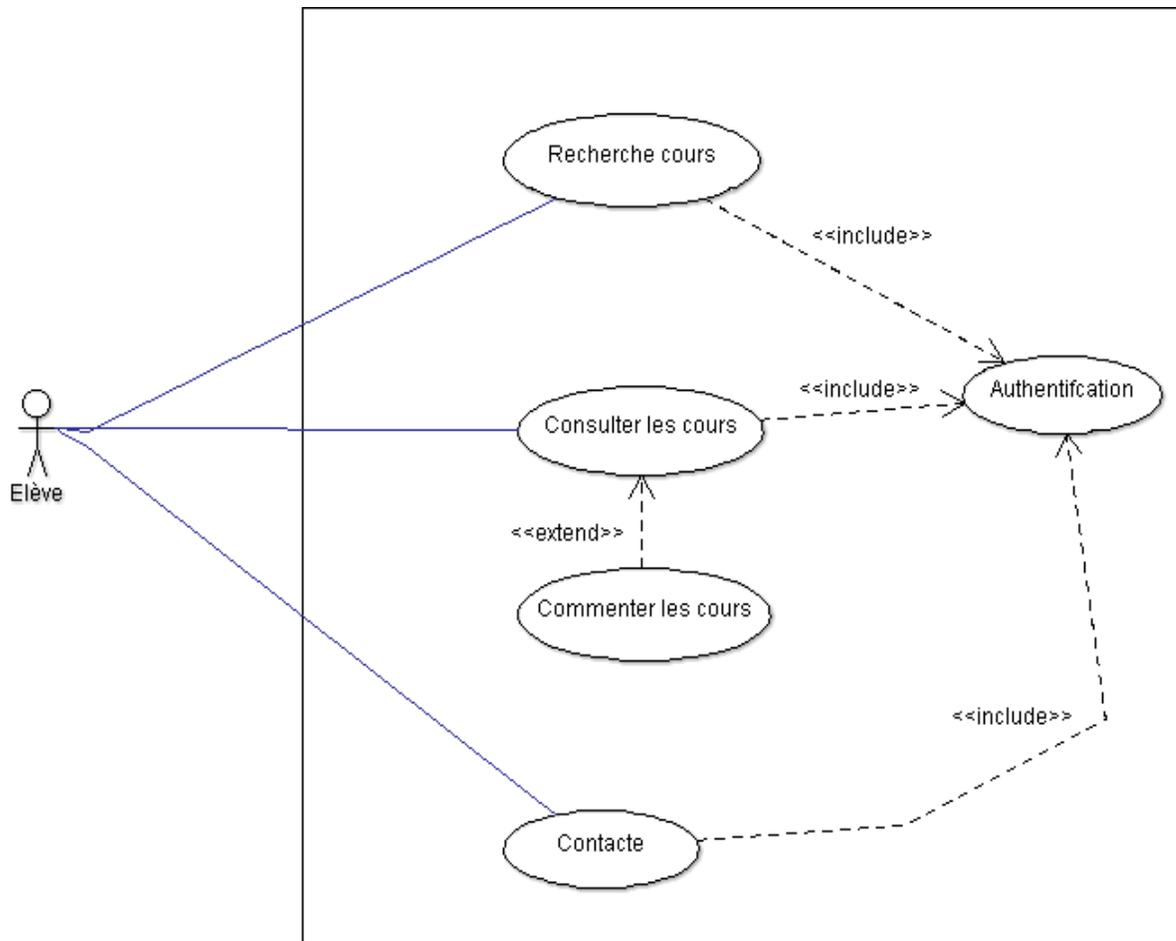


Figure 2.3 : diagramme de cas d'utilisation Elève

2.4. Description textuelle des cas d'utilisation :

2.4.1 Description textuelle pour cas d'utilisation « authentification » :

Objectif : vérifier l'identité de l'utilisateur et contrôler l'accès.

Le tableau 2.2 montre le sommaire d'identification et la description de l'enchaînement du sous cas « Authentification ».

Sommaire d'identification
Titre: Authentification. Résumé: chaque utilisateur doit introduire son email et son mot de passe pour accéder à l'interface qui lui correspond. Acteur: Administrateur, Formateur, élève.
Description de l'enchaînement
Pré conditions: l'utilisateur a bien accès à l'application. Post conditions : l'utilisateur est authentifié, il accède l'interface qui le concerne. Scénario nominal: <ol style="list-style-type: none">1. L'utilisateur tape son propre email et son mot de passe.2. Le système vérifie la validité des coordonnées de l'utilisateur.3. L'utilisateur va accéder la session qui le concerne.4. Scénario alternatif:<ul style="list-style-type: none">• le système vérifie la validité des coordonnées de l'utilisateur et indique que les champs obligatoires ne sont pas remplis, alors il ne peut pas accéder à son compte.

Table 2.2 : Description textuelle du cas d'utilisation Authentification

2.4.2. Description textuelle pour cas d'utilisation « Gestion des cours » :

Objectif : Gérer les cours Associés à une matière.

➤ **Sous cas « ajouter un cours »**

Le tableau 2.3 montre le sommaire d'identification et la description de l'enchaînement du sous cas « ajouter un cours ».

Sommaire d'identification
Titre : Ajouter un cours. Résumé : le Formateur se charge de la gestion des cours donc c'est lui le responsable de l'ajout d'un cours particulier. Acteur : Formateur.

Description de l'enchaînement
<p>Pré condition : Authentification.</p> <p>Post condition : Cours ajouté.</p> <p>Scénario nominale :</p> <ol style="list-style-type: none"> 1. L'utilisateur demande le formulaire d'ajout d'un cours. 2. Le système affiche le formulaire. 3. L'utilisateur remplit le formulaire puis le valide. 4. Le système enregistre la demande d'ajout. <p>Scénarios alternatifs :</p> <ul style="list-style-type: none"> • le système vérifie la validité des informations ajoutées par le formateur et indique que les champs obligatoires ne sont pas remplis, alors il lui renvoie la page de gestion de cours.

Table2.3 : Description textuelle du cas d'utilisation ajouter un cours

➤ **Sous cas « Modifier un cours »**

Le tableau w2.4 montre le sommaire d'identification et la description de l'enchaînement du sous cas « Modifier un cours ».

Sommaire d'identification
<p>Titre : Modifier un cours.</p> <p>Résumé : le Formateur se charge de la gestion des cours, vu que c'est lui qui a ajouter le cours il peut aussi le modifier.</p> <p>Acteur : Formateur.</p>
Description de l'enchaînement
<p>Pré condition : Authentification.</p> <p>Post condition : Cours modifié.</p> <p>Scénario nominale :</p> <ol style="list-style-type: none"> 1. Le formateur demande le droit d'accès à son compte. 2. Le système affiche le formulaire. 3. Le formateur remplit les modifications nécessaires puis le valide. 4. Le système enregistre les modifications établis.

Table2.4 : Description textuelle du cas d'utilisation modifier un cours

➤ **Sous cas « Supprimer un cours »**

Le tableau 2.5 montre le sommaire d'identification et la description de l'enchaînement du sous cas « supprimer un cours ».

Sommaire d'identification
Titre : Supprimer un cours. Résumé : le Formateur se charge de la gestion des cours, vu que c'est lui qui a ajouter le cours il peut aussi le supprimer. Acteur : Formateur.
Description de l'enchaînement
Pré condition : Authentification. Post condition : Cours supprimé. Scénario nominale : 1. Le formateur demande le droit d'accès à son compte. 2. Le système affiche le compte. 3. Le formateur choisit un cours permis les cours ajouté et le supprime. 4. Le système enregistre les modifications établis.

Table2.5 : Description textuelle du cas d'utilisation supprimer un cours

2.4.5. Description textuelle pour cas d'utilisation « Gestion des tests » :

Le tableau 2.6 montre le sommaire d'identification et la description de l'enchaînement du cas « Gestion des tests ».

Sommaire d'identification
Titre : gestion des tests. Résumé : le Formateur se charge de la gestion test, vu que c'est lui qui a ajouter test il peut aussi le modifier ou bien le supprimer.

Acteur : Formateur.
Description de l'enchaînement
<p>Pré condition : Authentification.</p> <p>Post condition : test créé.</p> <p>Scénario nominale :</p> <ol style="list-style-type: none"> 1. Le formateur demande le droit d'accès à son compte. 2. Le système affiche le formulaire. 3. Le formateur remplit les modifications nécessaires puis le valide. 4. Le système enregistre les modifications établis. <p>Scénarios alternatifs :</p> <ul style="list-style-type: none"> • le système vérifie la validité des informations ajoutés par le formateur et indique que les champs obligatoires ne sont pas remplis, alors il lui renvoie la page de gestion de tests.

Table2.6 : Description textuelle du cas d'utilisation gestion des tests

2.5. Diagrammes de séquence :

2.5.2. Diagramme de séquence de cas d'utilisation « Authentification » :

Cette section traite le diagramme de séquence qui modélise les interactions du cas d'utilisation « Authentification ». A partir de ce diagramme de séquence Figure 2.4, on déduit l'opération suivante : S'authentifier.

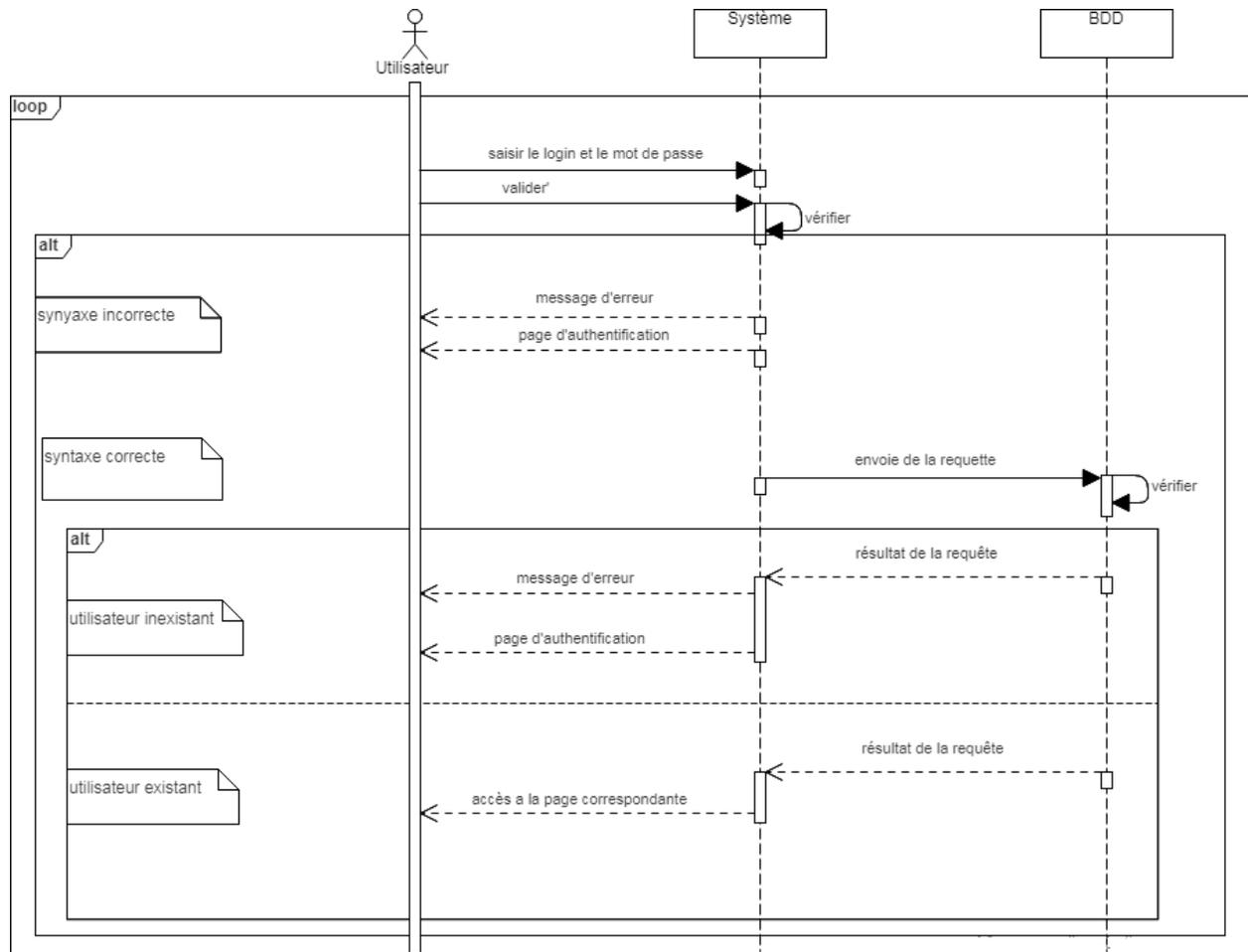


Figure 2.4 : Diagramme de séquence de cas d'utilisation « Authentification »

2.5.3. Diagramme de séquence de cas d'utilisation « Recherche cours »

Cette section traite le diagramme de séquence qui modélise les interactions du cas d'utilisation « Recherche cours ». A partir de ce diagramme de séquence Figure 2.5, on déduit l'opération suivante : rechercher un cours.

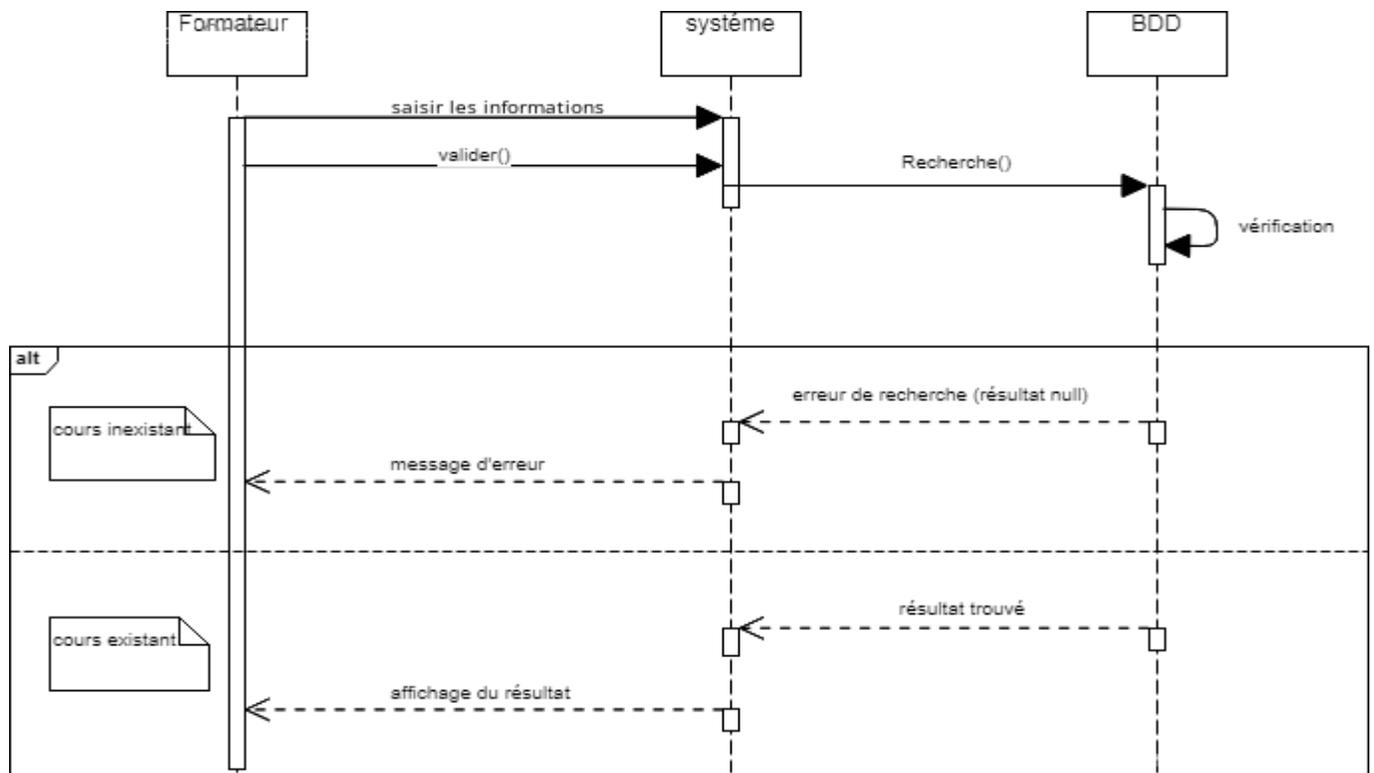


Figure 2.5 : Diagramme de séquence de cas d'utilisation « RechercheCours »

2.5.4. Diagramme de séquence de cas d'utilisation « Recherchetest »

Cette section traite le diagramme de séquence qui modélise les interactions du cas d'utilisation « RechercheTest ». A partir de ce diagramme de séquence Figure 2.6, on déduit l'opération suivante : rechercher un test.

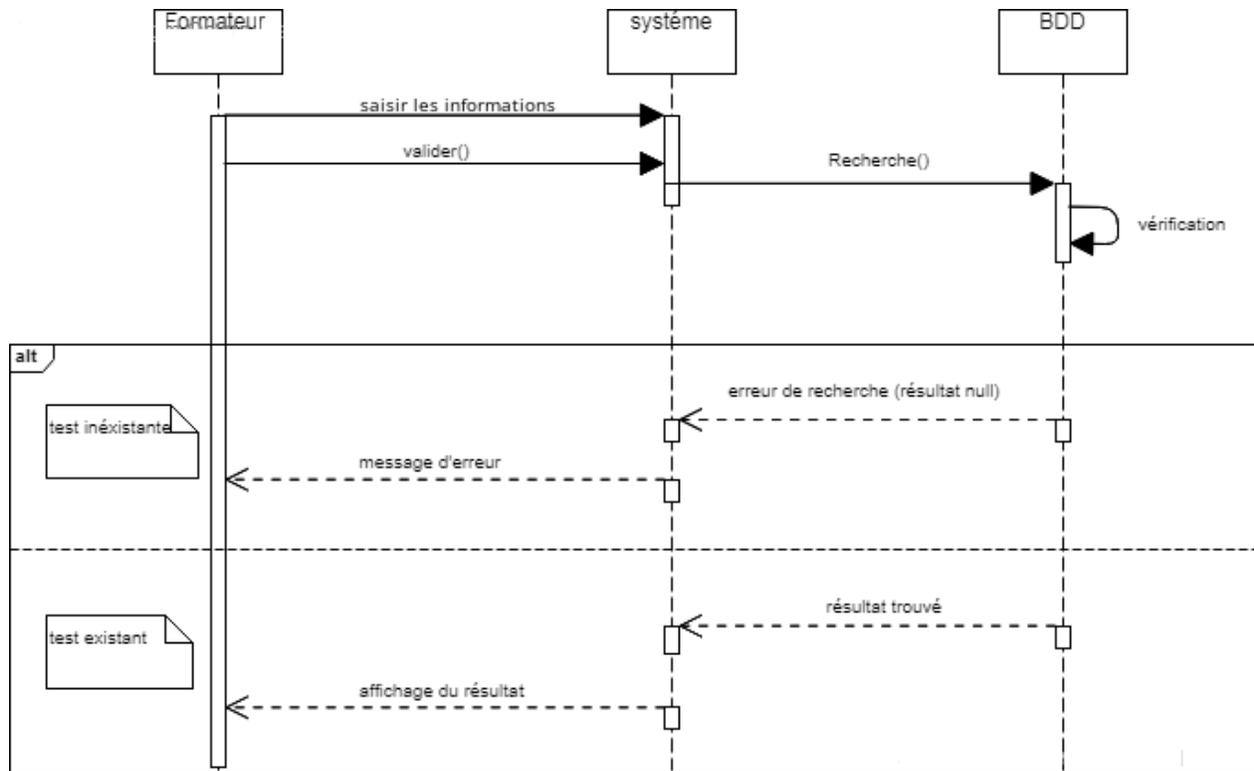
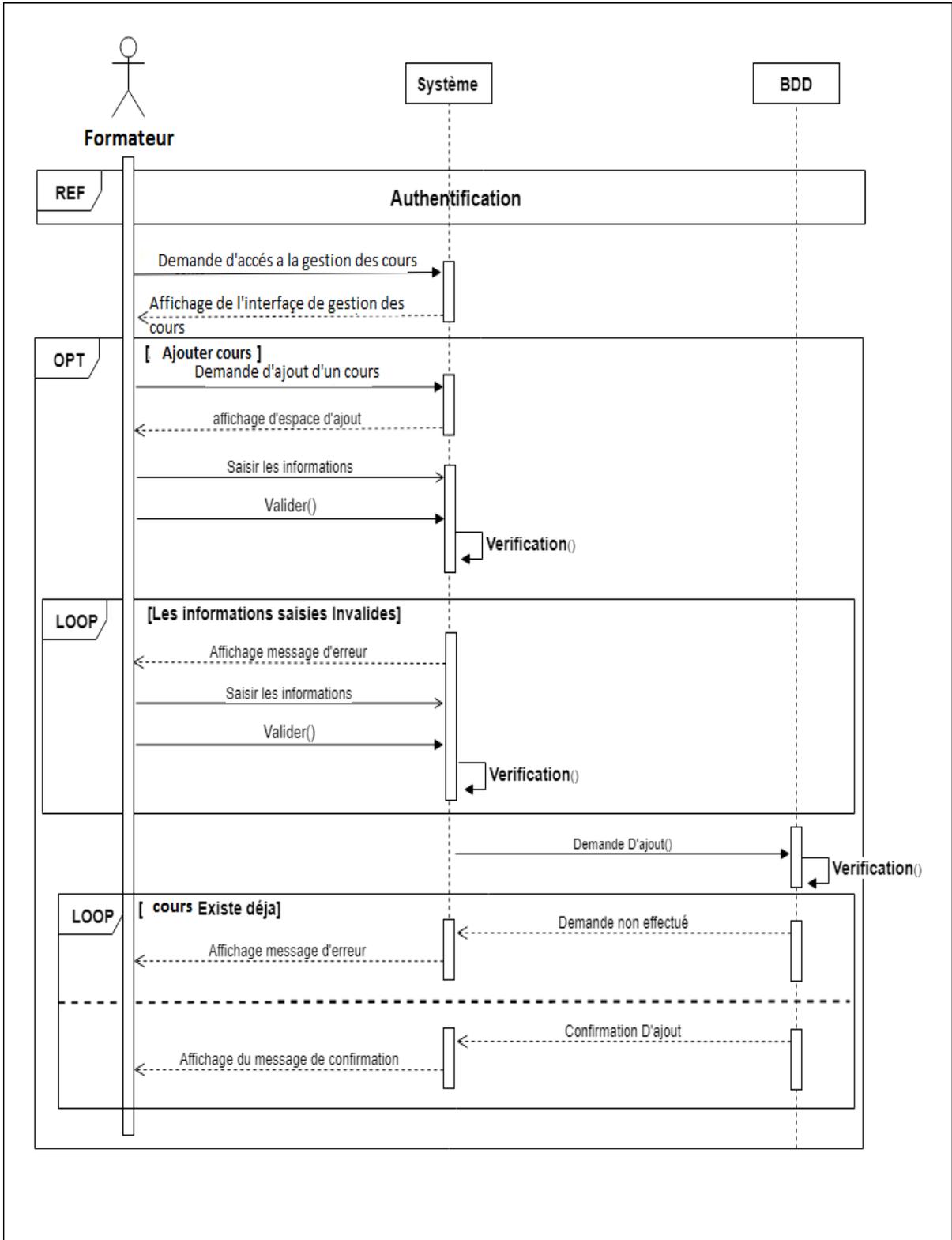


Figure 2.6 : Diagramme de séquence de cas d'utilisation « Recherchetest »

2.5.5. Diagramme de séquence de cas d'utilisation « Gestion des cours »

Cette section traite le diagramme de séquence qui modélise les interactions du cas d'utilisation « Gestion des cours ». A partir de ce diagramme de séquence Figure 2.7, on déduit les opérations suivantes : Ajouter un cours, Modifier un cours, supprimer un cours.



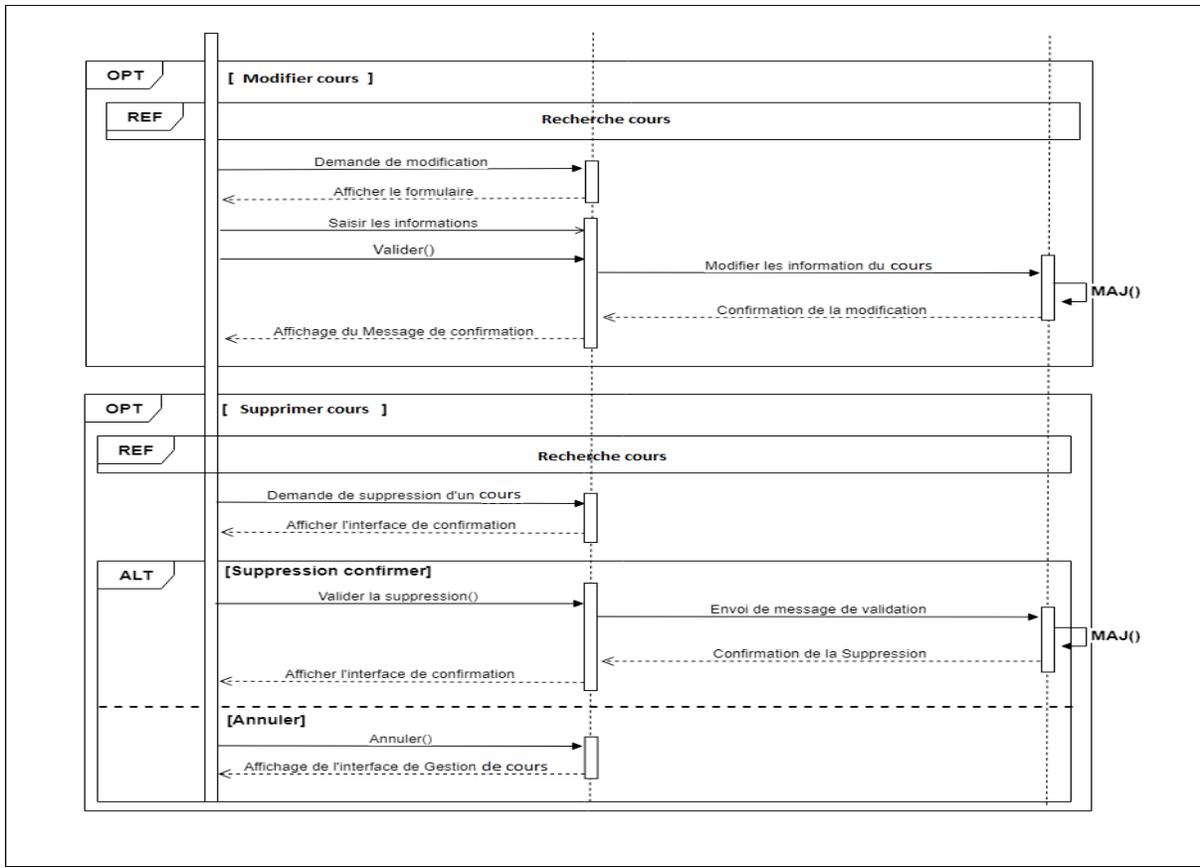
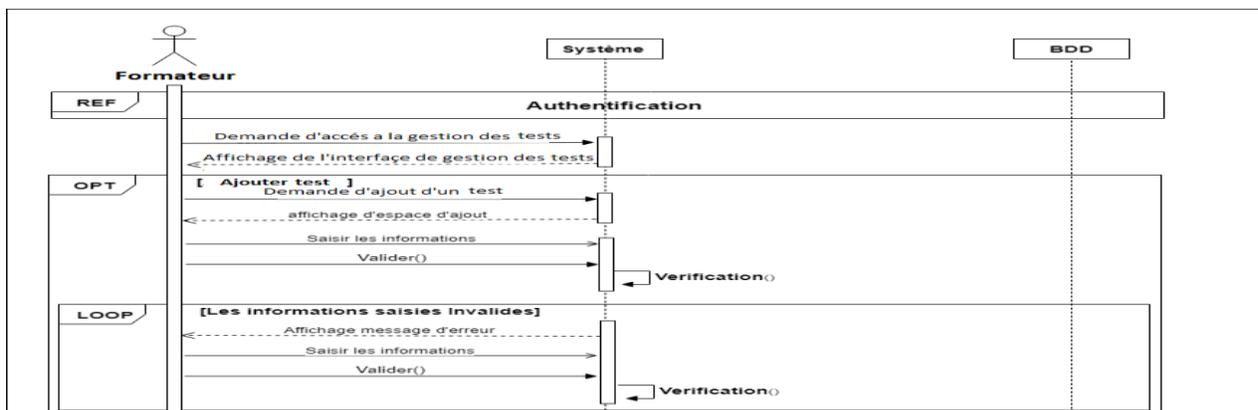


Figure 2.7 : Diagramme de séquence de cas d'utilisation « Gestion des cours »

2.5.6 Diagramme de séquence de cas d'utilisation « Gestion des tests »

Cette section traite le diagramme de séquence qui modélise les interactions du cas d'utilisation « Gestion des tests ». A partir de ce diagramme de séquence Figure 2.8, on déduit les opérations suivantes : Ajouter un test, Modifier un test, supprimer un test.



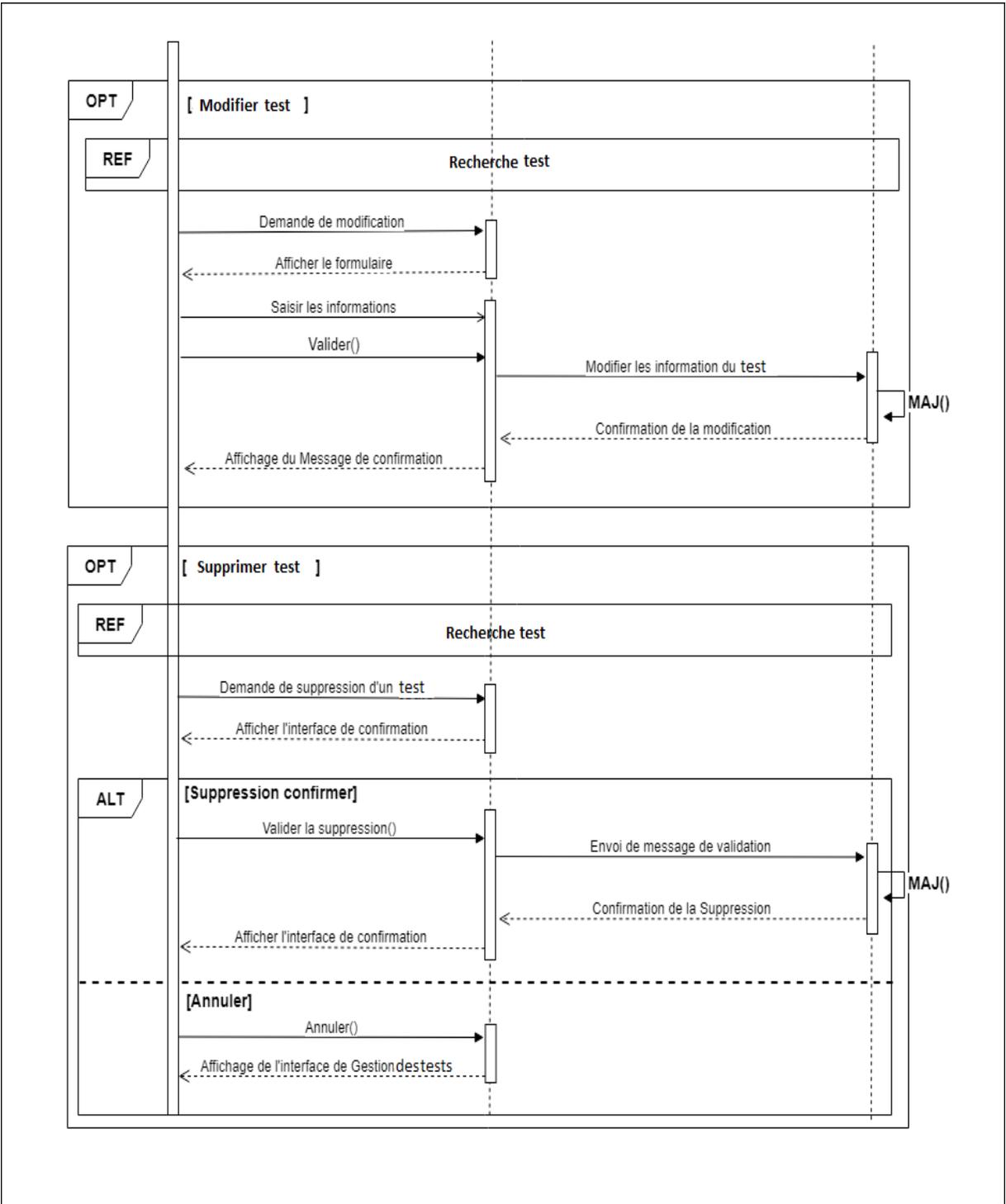


Figure 2.8: Diagramme de séquence de cas d'utilisation « Gestion des tests »

2.6. Diagramme de classe :

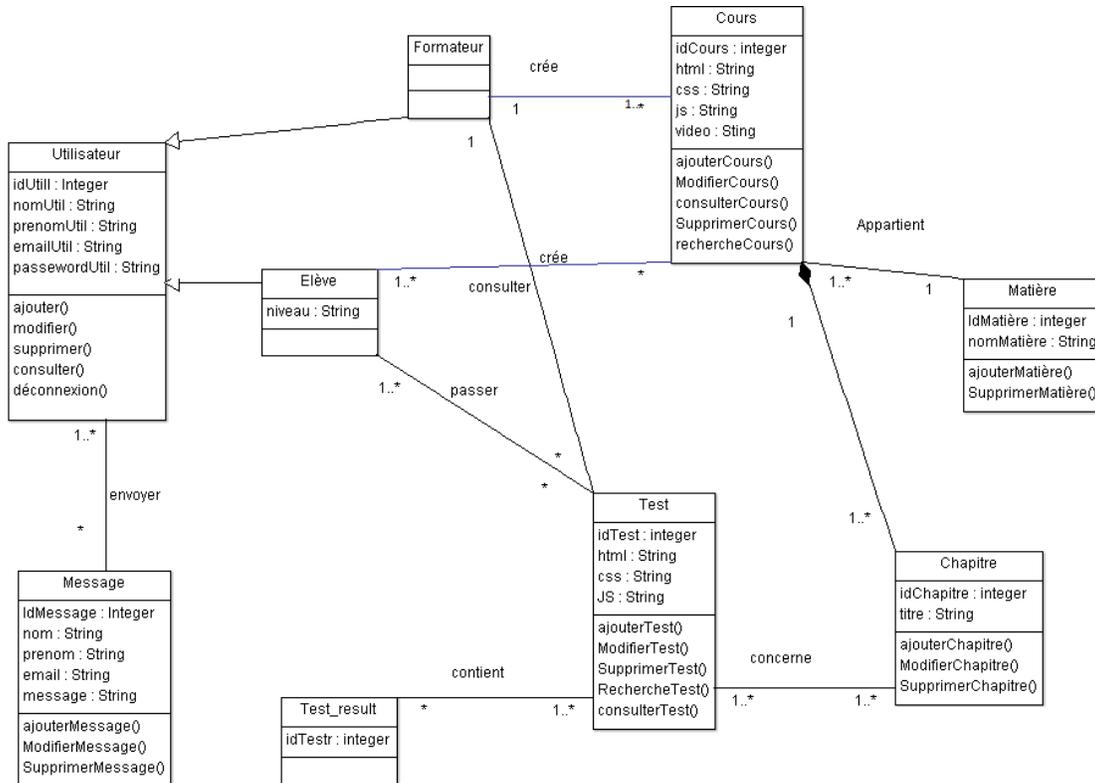


Figure 2.9 : Diagramme de Classe

2.7. Dictionnaire de données :

Classes	Attributs	Designation	Type	Méthodes
	IdUtilisateur	Identifiant de	INT	Ajouter()

Utilisateur		l'utilisateur		
	Nom	Nom de l'utilisateur	VARCHAR(50)	Modifier()
	Prénom	Prénom de l'utilisateur		Supprimer()
	Email	Email de l'utilisateur		Consulter()
	Mot_de_passe	Mot de passé de l'utilisateur		Déconnexion()

Formateur				
Elève	Niveau	Le niveau de l'étudiant	VARCHAR(50)	
Test	IdTest	Identifiant du Test	INT	AjouterTest ()
	Html	Html du test	VARCHAR (50)	ModifierTest ()
	css	css du test		SupprimerTest ()
	Js	javascript du test		ConsulterTest ()
				RechercheTest()
Cours	IdCours	Identifiant du cours	INT	AjouterCours()
	html	Le code html du cours	VARCHAR (50)	ModifierCours()
	css	Le code css du cours		SupprimerCours()
	js	Le code javascript du cours		ConsulterCours()
	video	La video du cours		RechercherCours()
Chapitre	Idchapitre	Identifiant du chapitre	INT	AjouterChapitre ()
	Titre	Titre du chapitre	VARCHAR (30)	ModifierChapitre ()
				SupprimerChapitre()
Matière	IdMatière	Identifiant de la matière	INT	AjouterMatière()
	nom	Nom de la matière	VARCHAR (30)	SupprimerMatière()

Test_result	IdResult	Identifiant du resultat	INT	AjouterMessage()
Message	IdMessage	Identifiant du Message	VARCHAR (50)	ModifierMessage()
	nom	nom de l'expéditeur		SupprimerMessage()
	Prénom	Prénom de l'expéditeur		
	Email	Email de l'expéditeur		
	msg	Contenu du Message	VARCHAR (300)	

Table 2.7 : Dictionnaire de données.

2.8. Model Relationnel :

Les règles de passage au model relationnel :

Pour réaliser ce passage, nous avons suivi des règles strictes et précises permettant de traduire le contenu conceptuel du diagramme de classe en modèle relationnel. Ces règles sont :

- Transformation des classes : chaque classe du diagramme UML devient une relation.
- Transformation des associations : Nous distinguons trois familles d'associations :

Association 1..* : L'attribut de type de clé étrangère doit être ajouté à la sous-relation associée. Cet attribut porte le nom de la clé primaire de la relation parent associée.

Association *..* et n-aire et association de classe : L'association de classe devient une sorte de relation. La clé primaire de cette relation est la concaténation des identifiants des classes connectées.

Association 1..1 : L'attribut de type clé étrangère doit être ajouté à la relation dérivée de la classe avec la plus petite multiplicité égale à 1. Cet attribut porte le nom de la clé primaire de la relation dérivée de l'association de classe connectée [3].

L'existence de la généralisation (Méthode 1 : Push-ups) crée des relations avec tous les attributs de la classe. Ajoutez des attributs pour distinguer les types d'objets.

Me Méthode2 : push-down : Chaque sous-classe est convertie en une relation. La clé primaire de la superclasse migre vers la relation générée par la sous-classe et devient la clé primaire et la clé étrangère.

S'il existe une contrainte de totalité ou de partition sur l'association, il est possible de ne pas traduire la relation issue de la surclasse. Il faut alors faire migrer tous ses attributs dans la (les) relation(s) issue(s) de la (des) sous-classe(s).

- **Le Model relationnelle de notre application :**

Après l'application des règles de passage vers le modèle relationnel citées ci-dessus, nous avons abouti au modèle relationnel suivant :

Utilisateur (IdUtilisateur,nom,prénom,Email,mot_de_passe,type)

Cours (IdCours,#Idchapitre,#IdUtilisateur,html,css,js,video)

Test (IdTest,#Idchapitre, #IdUtilisateur,html,ss,js)

Matière (IdMatière,Nom)

Chapitre (Idchapitre,#IdMatière,titre)

Test_result (IdResult,#IdUtilisateur,#IDTest)

Messages (IdMessage,nom,prénom,Email,msg)

Conclusion

Dans ce deuxième chapitre, nous sommes en mesure d'exécuter et de concevoir nos Postuler pour un site d'apprentissage en s'adaptant d'abord au langage de modélisation Et sur des diagrammes de formalisme UML. De plus, nous avons également introduit les règles pour passer du modèle conceptuel au modèle relationnel.

Le chapitre suivant présentera spécifiquement la mise en œuvre et l'implémentation de cette application.

Chapitre 3

Réalisation

3. Chapitre 3 : Réalisation

Introduction :

La réalisation vient couronner le travail de l'étude préalable et de l'étude conceptuelle. Elle représente la dernière étape et elle est très importante puisque grâce à elle le projet informatique va exister réellement. Ce chapitre est consacré à la présentation et à la définition des outils liés à la réalisation de notre application ainsi qu'à un aperçu sur le fonctionnement des différentes interfaces réalisées.

3.1 Environnement de développement :

3.1.1. Visual studio code:



Visual Studio Code est un éditeur de code simplifié prenant en charge les opérations de développement telles que le débogage, l'exécution de tâches et le contrôle de version. , il est développé par Microsoft pour Windows, Linux et MacOS. Il vise à fournir uniquement les outils dont un développeur a besoin pour un cycle de création de code-débogage rapide et laisse les flux de travail plus complexes à des IDE plus complets.

3.1.2. WAMPserver :



WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données.

WAMPServer est en mesure de lancer des sites web dynamiques qu'il soit connecté à Internet ou en local. Il peut être utilisé avec de simples navigateurs et contient un serveur Apache permettant de transférer des fichiers en ligne. Ce logiciel prend en charge la gestion des banques d'informations.

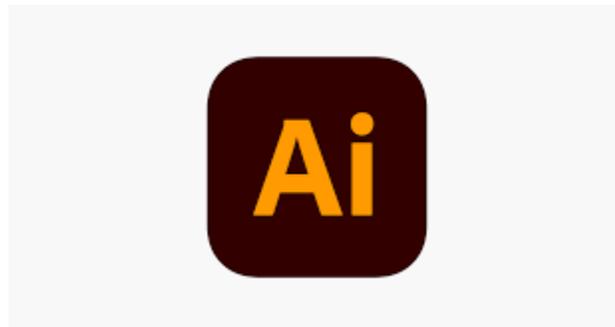
3.1.3. ArgoUML :



ArgoUML est un logiciel libre de création de diagrammes UML. Il supporte sept types de

diagramme : cas d'utilisation, classes, séquence, état, collaboration, activité et déploiement. La génération de code à partir de diagrammes de classes est supportée dans les langages suivants : Java, C++, PHP, C# et SQL.

3.1.4. Adobe Illustrator :



Adobe Illustrator est un logiciel de création graphique vectorielle ce qui signifie que toutes les illustrations se définissent comme une série de coups (frontières) et remplis (la coloration intérieure). Il fait partie de la gamme Adobe, peut être utilisé indépendamment ou en complément de Photoshop, et offre des outils de dessin vectoriel puissants. Les images vectorielles sont constituées de courbes générées par des formules mathématiques. Les formats vectoriels d'Illustrator peuvent être adaptés à n'importe quelle taille ou une résolution sans produire les effets de pixilation que, disons, agrandissant une image JPEG à créer un graphique affiche. Cela permet aux œuvres d'art créées dans Illustrator d'être revues et réutilisées à maintes reprises - des cartes de visite aux panneaux d'affichage - sans être refaites fois interminables.

3.2. Technologieslogicielles :

3.2.1 PrésentationHtml :



HTML est un langage informatique descriptif. Plus précisément, il s'agit d'un format de données utilisé dans le monde Internet pour formater des pages Web. Il permet non seulement l'écriture d'hypertexte, mais permet également l'introduction de ressources multimédias dans le contenu. Le format ou langage HTML développé par le W3C (World Wide Web Consortium) et le WHATWG (Web HyperText Application Technology Working Group) est apparu dans les années 1990 et a été progressivement révisé, et une version HTML5 plus complète est fournie depuis 2014. HTML est basé sur un système de balises, qui permet d'ajouter des titres, des sous-titres, des caractères gras, etc., du texte et l'introduction d'éléments interactifs, tels que des images, des liens, des vidéos... HTML en tant que langage est plus facile pour les robots des moteurs de recherche pour comprendre JavaScript, et il est également utilisé pour rendre la page plus interactive.

3.2.2 Présentation CSS :



Les feuilles de style ("Cascading Style Sheets" en anglais, en abrégé CSS) est un langage qui

permet de gérer la présentation des pages web. CSS est une norme recommandée du World Wide Web Consortium (W3C), similaire à HTML ou XML.

Le but de CSS est de séparer la structure d'un document HTML de sa présentation. En effet, en utilisant HTML, on peut définir la structure (la structure hiérarchique entre le contenu et les différentes parties du document) et la présentation. Mais cela posera quelques problèmes. En utilisant des paires HTML/CSS, nous pouvons créer des pages Web où la structure du document est dans un fichier HTML et la présentation est dans un fichier CSS.

3.2.3 Présentation javascript :



JavaScript est un langage de programmation qui vous permet d'implémenter des mécanismes complexes sur des pages Web. Chaque fois qu'une page Web affiche plus que du contenu statique mis à jour, des cartes interactives, des animations 2D/3D, des menus vidéo défilants, etc. à un moment donné, JavaScript est susceptible d'être impliqué. Il s'agit de la troisième couche de la technologie Web standard, les deux premières couches (HTML et CSS)

3.2.4 Présentation MYSQL :



MySQL est un système de gestion de base de données relationnelle (SGBDR) propriétaire, gratuit, puissant, très populaire, multi-thread et multi-utilisateurs. MySQL est principalement un serveur de base de données. Afin de s'y connecter localement ou à distance, nous utilisons un client. Cela peut être une commande MySQL ou un script PHP normal. Dans ce cas, vous devez installer le module PHP MySQL qui permet à PHP de communiquer avec le serveur MySQL

3.2.5 Présentation des Framework utilisés:

3.2.5.1. Définition du Framework :

Un Framework est un ensemble d'outils et de composants logiciels organisés selon des plans architecturaux et des modèles de conception. L'ensemble forme un squelette de programme. Il est généralement fourni sous la forme d'une bibliothèque logicielle avec un schéma directeur pour l'architecture cible du Framework. En programmation orientée objet, un Framework est généralement composé de classes parentes, et ces classes parentes seront dérivées et étendues par héritage en fonction des besoins spécifiques de chaque logiciel utilisant le Framework.

3.2.5.2. Présentation de Laravel :



Laravel est un Framework d'application Web avec une syntaxe expressive et élégante. Laravel simplifie le développement en facilitant les tâches courantes utilisées dans de nombreux projets Web, tels que :

1. Moteur de routage simple et rapide.
2. Puissant conteneur d'injection de dépendances.
3. Plusieurs backends pour le stockage de session et de cache.
4. Base de données ORM expressive et intuitive.
5. Migration en mode indépendant de la base de données.
6. Traitement robuste des tâches en arrière-plan.
7. Diffusion d'événements en temps réel.

3.2.5.3. Présentation de Bootstrap :



Bootstrap est un framework de développement frontal gratuit et open source pour la création de sites Web et d'applications Web. Le framework Bootstrap est construit sur HTML, CSS et

JavaScript (JS) pour faciliter le développement de sites et d'applications réactifs et mobiles.

La conception réactive permet aux pages Web ou aux applications de percevoir la taille et l'orientation de l'écran du visiteur, et d'ajuster automatiquement l'affichage en conséquence ; l'approche axée sur le mobile suppose que les Smartphones, les tablettes et les applications mobiles pour des tâches spécifiques sont des employés qui terminent leur travail et se rencontrent ces technologies dans la conception Le principal outil requis.

Bootstrap comprend des composants d'interface utilisateur, une mise en page et des outils JS et un cadre de mise en œuvre. Le logiciel peut être précompilé ou utilisé comme code source.

3.3 Présentation de l'application :

Notre application d'apprentissage permet de :

- Gérer les cours par le formateur.
- Gérer les tests par le formateur.
- Gérer les utilisateurs par l'administrateur.

Les différentes tâches que notre application est capable de faire :

3.3.3 Interface « Authentification » :

L'interface d'authentification permet à un utilisateur déjà inscrit, de s'authentifier pour accéder à un espace bien déterminé.

L'authentification se fait, en saisissant dans la zone d'identification le nom d'utilisateur et le mot de passe.

L'authentification se fait, en saisissant dans la zone d'identification le nom d'utilisateur et le mot de passe.

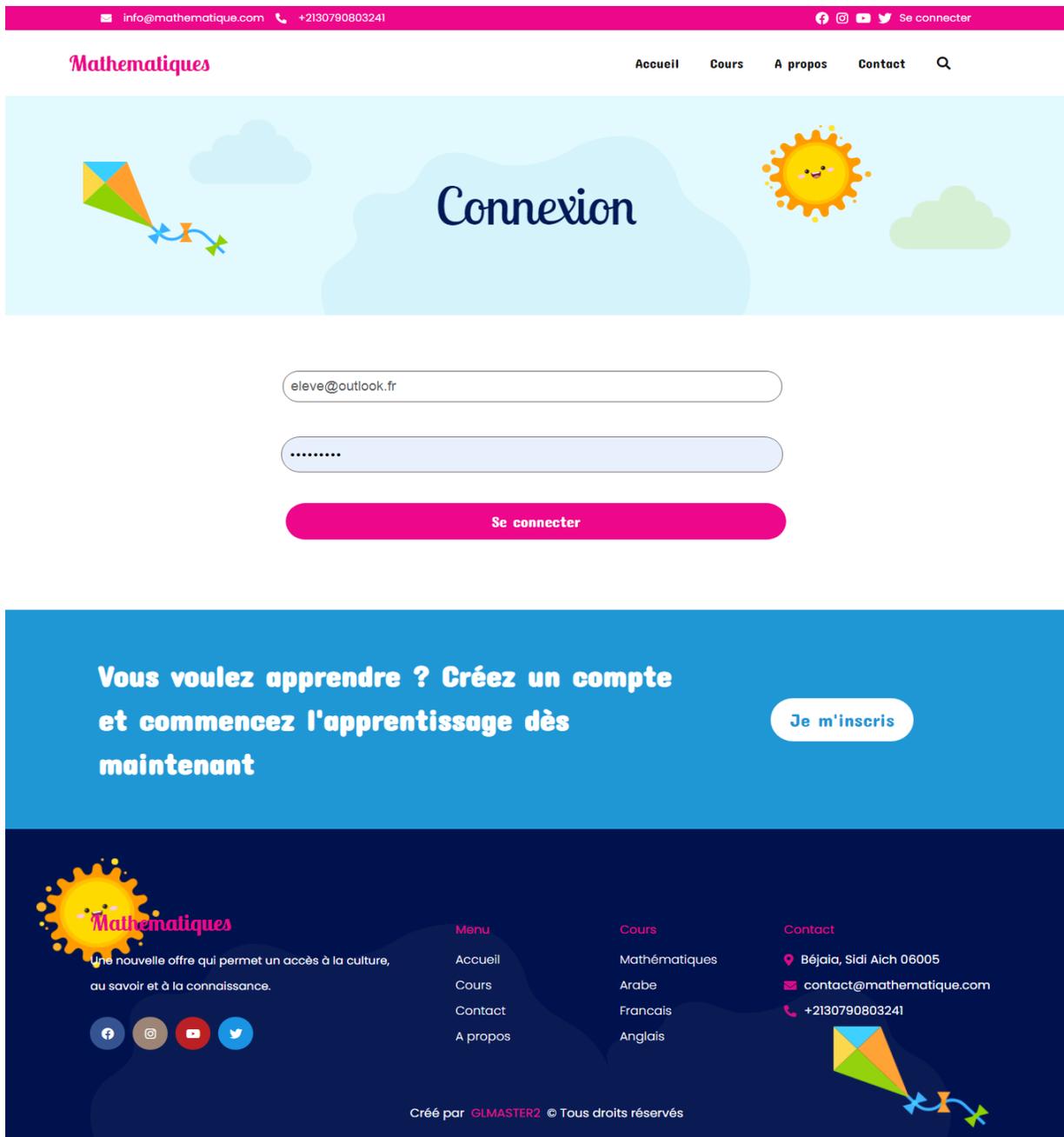


Figure 3.1 : Interface Authentification

3.3.4 Interface « gestion des cours » :

La gestion des cours est une tâche basique du formateur, elle permet d'éliminer Modifier ou ajouter de nouveaux cours en saisissant les informations nécessaires.

Mathématiques

Profile

Ajouter un cours

Mes Cours

Messages

OVERVIEW

Listes des cours

Mes cours

#	Titre	Créé le	Modifier	Supprimer
18	Les entiers de 0 à 10	2021-09-03 20:01:49	Modifier	Supprimer
21	Le nombre 10	2021-09-03 21:02:11	Modifier	Supprimer
22	Comparer deux nombres de 0 à 9	2021-09-04 01:20:56	Modifier	Supprimer
23	Comparer deux nombres à deux chiffres	2021-09-04 01:25:58	Modifier	Supprimer
24	Ranger plusieurs nombres de 0 à 9	2021-09-04 01:31:56	Modifier	Supprimer
25	Nombres de 10 à 69	2021-09-04 01:41:49	Modifier	Supprimer
26	Nombres de 70 à 79	2021-09-04 01:45:19	Modifier	Supprimer
27	Nombres de 80 à 99	2021-09-04 01:52:38	Modifier	Supprimer

Accueil [Cours](#) [A propos](#) [Contact](#)

Copyright © 2021 GLMASTER2

Figure 3.2 : Interface gestion de cours

3.3.5 Interface « Consulter un cours » :

Cette interface est réservée pour l'élève afin de consulter les différents cours disponibles qui sont ajoutés par ces professeurs.



Introduction Vidéo Je m'entraîne

Nombres de 70 à 79

Dans cette vidéo, on apprend à compter les nombres de 70 à 79. Quand on compte ces nombres, on entend le « soixante », et pourtant le nombre commence par le chiffre 7. Pour compter de 70 à 79, c'est-à-dire des nombres qui ont sept dizaines, on utilise le mot « soixante » et on complète par les nombres de dix à dix-neuf.



Je m'entraîne

trouve le numéro manquant et place le dans la case blanche

Car parking exercise with numbers 73, 74, 76 and empty boxes for 75, 77, 78, 79.

fait glisser les numéros ci-dessous

Number selection buttons: 77, 75, 78, 72 and a 'Valider' button.

Figure 3.3 : Interfaceconsultercours

3.3.6 Interface « liste des cours »:

Cette interface est responsable d'afficher tous les titres des cours disponible sous forme de lien et dès qu'on clique sur un titre ça nous amène un cours particulier.



Apéru du cours

dans ce cours dédié au mathématiques. l'élève pourra apprendre à compter, calculer (addition, soustraction....), encadrer des nombres. et pour cela nous avons mis à sa disposition de nombres cours répartis en chapitres où chaque cours contient une introduction, une vidéo explicative et une section pour entrainement avec des exemples. à la fin de chaque chapitre se trouve un test évalué

- nombre et compter** ×
 - Les entiers de 0 à 10
 - Le nombre 10
 - Comparer deux nombres de 0 à 9
 - Comparer deux nombres à deux chiffres
 - Ranger plusieurs nombres de 0 à 9
 - Nombres de 10 à 69
 - Nombres de 70 à 79
 - Nombres de 80 à 99
- décomposer des nombres** +

Vous voulez apprendre ? Créez un compte et commencez l'apprentissage dès maintenant Je m'inscris

Figure3.4 : Interface liste des cours

3.3.7 Interface « gestion des tests » :

La gestion des tests est une tâche basique du formateur, elle permet d'ajouter Modifier ou supprimer un test en saisissant les informations nécessaires.

The screenshot shows a web application interface for managing tests. On the left is a sidebar with a menu: 'Mathématiques' (selected), 'Profile', 'Ajouter un cours', 'Ajouter un test', 'Mes Cours', 'Mes Tests', and 'Messages'. The main content area is titled 'TESTS > Listes des Tests'. Below the title is a section 'Mes cours' containing a table with the following data:

#	Titre	Crée le	Modifier	Supprimer
2	Test (nombre et compter)	2021-09-01 01:36:54	Modifier	Supprimer

At the bottom of the page, there are navigation links: 'Accueil', 'Cours', 'Apropos', 'Contact' and a copyright notice: 'Copyright © 2021 GLMASTER2'.

Figure 3.5 : Interface gestion des tests

3.4. Architecture de l'application

L'image ci-dessous représente l'architecture de notre application en Laravel :

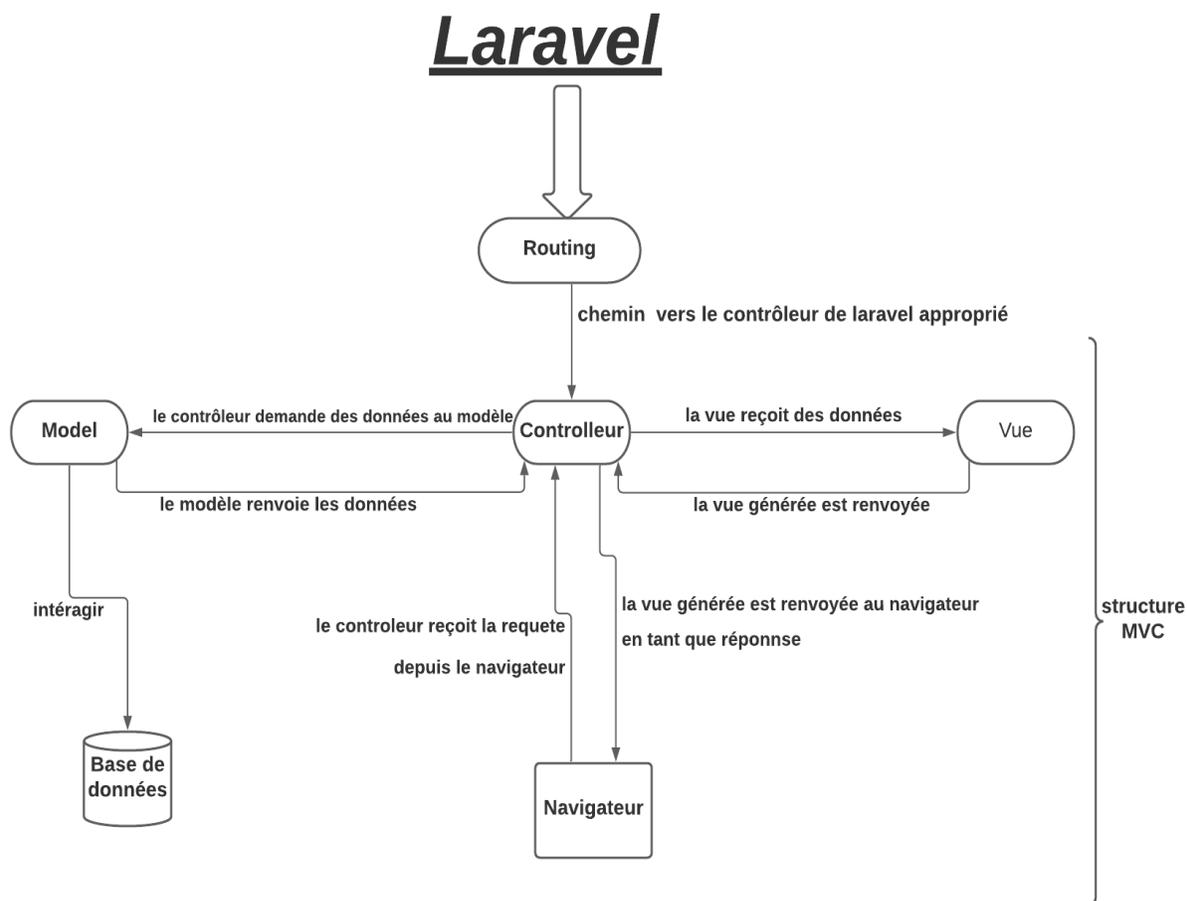
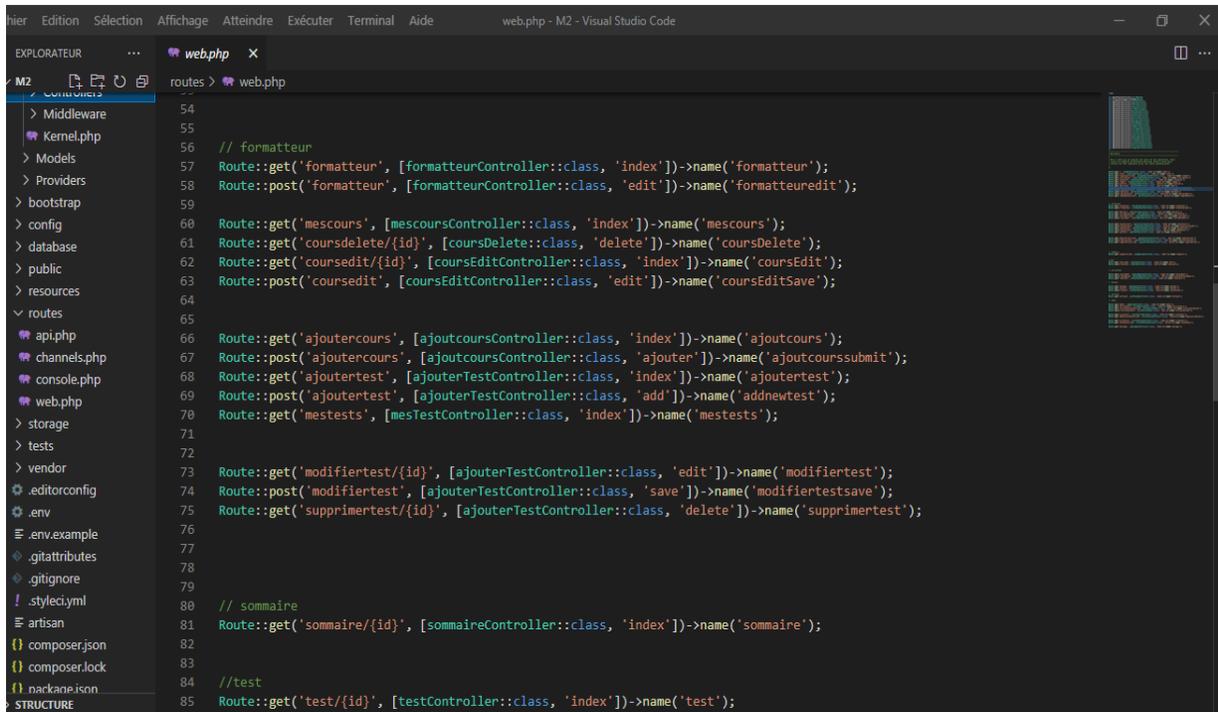


Figure 3.6 : image représentant l'architecture de notre application

3.4.1. Les routes :

Tous les routes de notre application se trouvent dans le dossier Routes/web.php c'est là où se trouve le chemin vers le contrôleur de laravel approprié.

C'est par ce fichier qu'il faut analyser et orienter la requête, Jetons un coup d'œil à ce qui est contenu dans le fichier de routage sur la figure suivante :



```
54
55
56 // formatteur
57 Route::get('formatteur', [formatteurController::class, 'index'])->name('formatteur');
58 Route::post('formatteur', [formatteurController::class, 'edit'])->name('formatteuredit');
59
60 Route::get('mescours', [mescoursController::class, 'index'])->name('mescours');
61 Route::get('coursdelete/{id}', [coursDelete::class, 'delete'])->name('coursDelete');
62 Route::get('coursedit/{id}', [coursEditController::class, 'index'])->name('coursEdit');
63 Route::post('coursedit', [coursEditController::class, 'edit'])->name('coursEditSave');
64
65
66 Route::get('ajoutercours', [ajoutcoursController::class, 'index'])->name('ajoutercours');
67 Route::post('ajoutercours', [ajoutcoursController::class, 'ajouter'])->name('ajoutercourssubmit');
68 Route::get('ajoutertest', [ajoutertestController::class, 'index'])->name('ajoutertest');
69 Route::post('ajoutertest', [ajoutertestController::class, 'add'])->name('addnewtest');
70 Route::get('mestests', [mesTestController::class, 'index'])->name('mestests');
71
72
73 Route::get('modifiertest/{id}', [ajoutertestController::class, 'edit'])->name('modifiertest');
74 Route::post('modifiertest', [ajoutertestController::class, 'save'])->name('modifiertestsave');
75 Route::get('supprimertest/{id}', [ajoutertestController::class, 'delete'])->name('supprimertest');
76
77
78
79 // sommaire
80
81 Route::get('sommaire/{id}', [sommaireController::class, 'index'])->name('sommaire');
82
83
84 //test
85 Route::get('test/{id}', [testController::class, 'index'])->name('test');
```

Figure 3.7 : image représentant le fichier web.php

3.4.2. Les migrations :

Les migrations sont utilisées pour créer, modifier et partager le schéma de base de données de notre application, au début nous devons d'abord configurer le fichier « .env » on remplissant les champs qui contient les informations de notre base de données (figure9)ensuite Pour créer une migration nous utilisons la requête « php artisan make : migration create_utilisateurs_table », nous allons présenter si dessous l'ensemble des champs utilisé dans notre table utilisateurs figure 10.

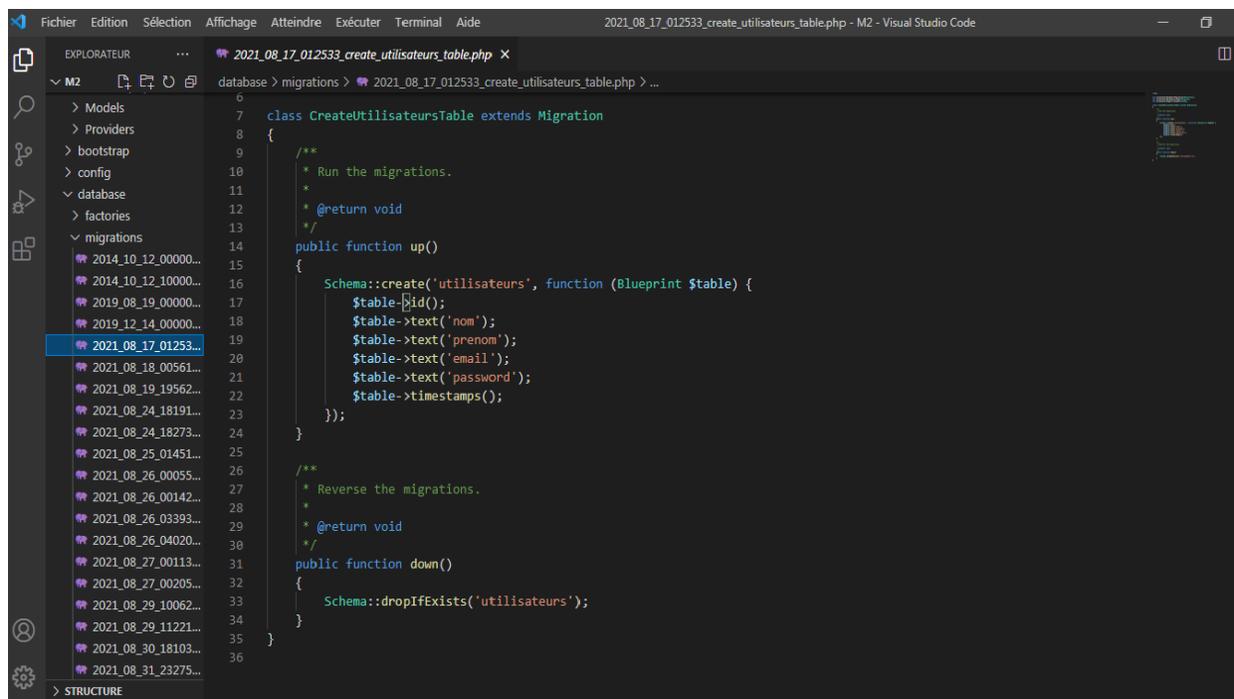


Figure 3.8: image représentant la migration de la table utilisateurs

3.4.3. Les contrôleurs :

Un contrôleur est une classe qui va contenir différentes méthodes. La tâche d'un contrôleur est de réceptionner une requête qui a déjà été sélectionnée par une route et de définir la réponse appropriée. On utilisant commande : « php artisan make :controller listCoursController » nous créons le contrôleur listCours , ce contrôleur contient la méthode index qui est chargée

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\cours;
6 use Illuminate\Http\Request;
7
8 class listCoursController extends Controller
9 {
10
11     public function index(Request $request){
12         if($request->session()->has('loggedAdmin')){
13             $data = cours::all();
14             return view('admin/listecours', ['data'=> $data]);
15         }else{
16             return redirect('/connexion');
17         }
18     }
19
20
21     public function delete($id){
22         $cours = cours::where('id', '=', $id)->delete();
23         if($cours){
24             return back()->with('success', 'ok');
25         }else{
26             return back()->with('fail', 'no');
27         }
28     }
29 }
```

Figure 3.9: image représentant le contrôleur listCours.

3.4.4. Les modèles :

Le modèle est une classe qui représente la structure logique et la relation de la table de données sous adjacente, chaque table dans la base de données a un modèle qui nous permet d'interagir avec cette table en utilisant eloquent qui est un ORM. Pour créer un modèle en utilisant la commande « php artisan make:model test »

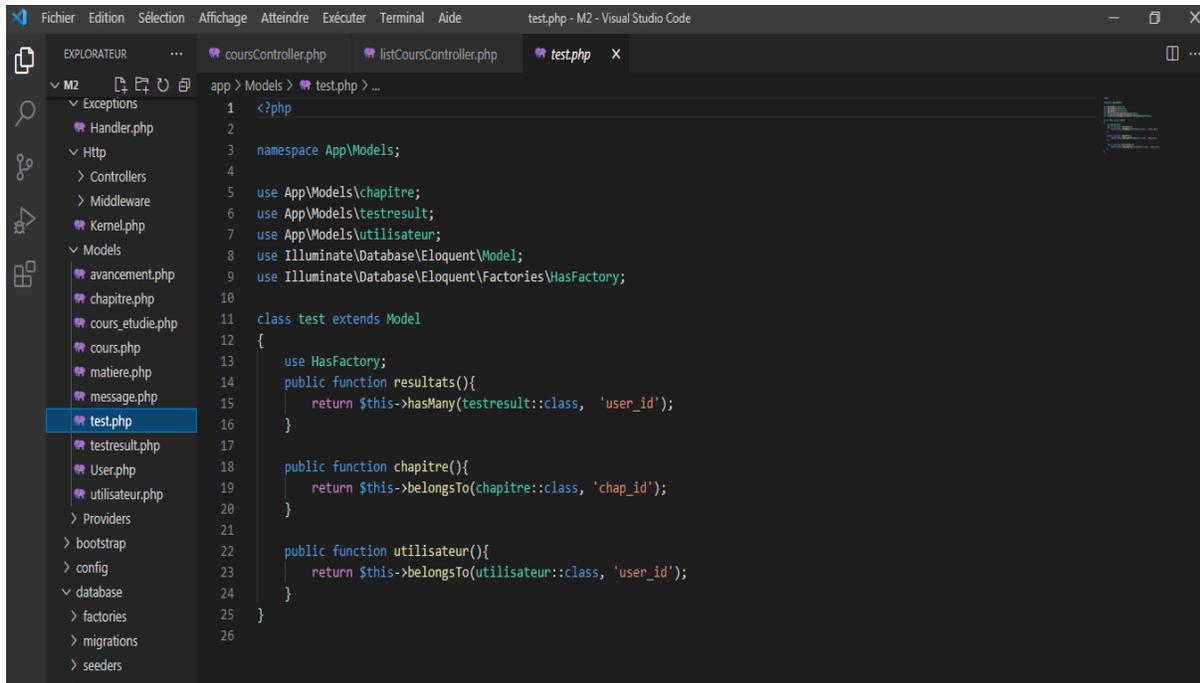


Figure 3.10: image représentant le modèle test.

3.4.5. Les vues :

La vue est en charge d'afficher le code HTML qui sera retourné par une route.

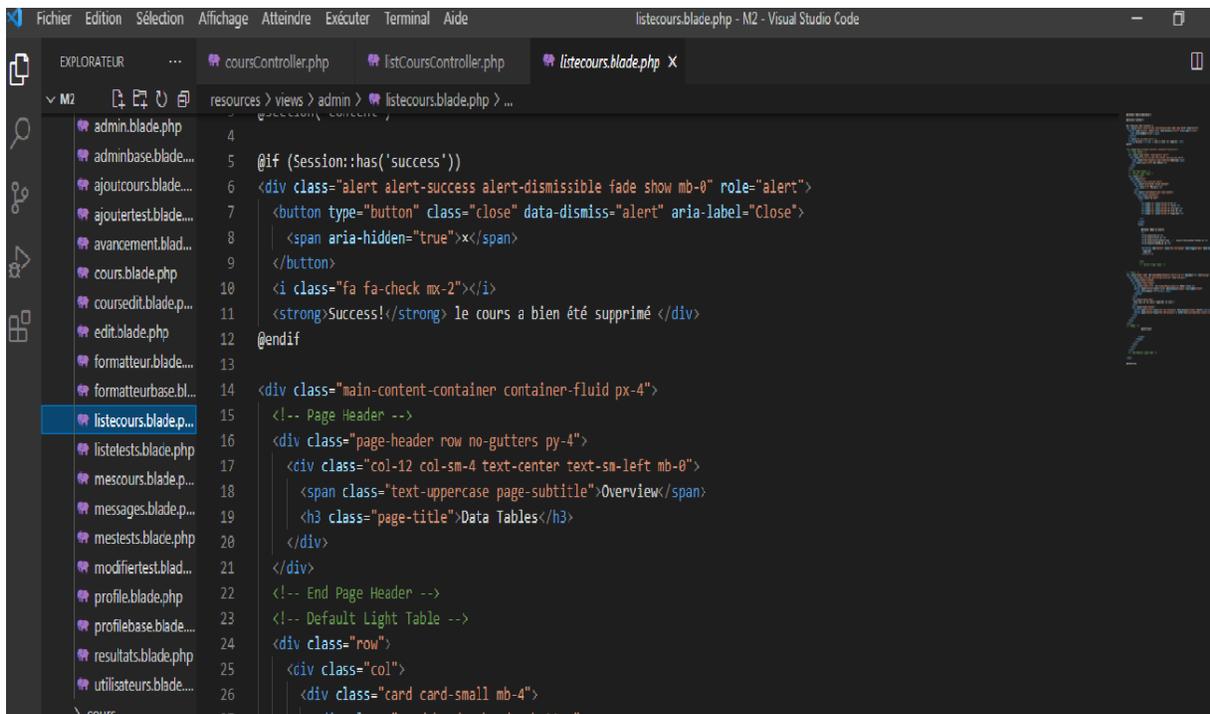


Figure 3.11: image représentant la vue listecours.

Conclusion :

Tout au long de ce chapitre nous avons expliqué les différentes technologies utilisées pour implémenter notre application, ensuite nous avons présenté les différentes interfaces de notre application et enfin l'architecture d'implémentation d'où nous atteignons la fin de notre projet.

Conclusion général et perspective

Ce projet de fin d'études avait pour ambition d'établir un site d'apprentissage en ligne pour des élèves de 1ère année primaire, afin d'aider les enfants à apprendre leurs leçons en ligne tout en s'amusant.

Ce mémoire mentionne toutes les étapes traversées afin d'arriver au résultat final et souhaité. Dans un premier temps nous avons énumérer tous les besoins existants, et nous avons pu identifier les exigences de notre future application web.

Nous avons par la suite fait une recherche sur des sites web existants et qui proposent le même service afin de connaître les fonctionnalités essentielles, et c'est ce qui nous a permis d'entamer la phase d'analyse et de conception.

Pour conclure le dernier chapitre, notre projet d'étude touche à sa fin. Nous expliquons dans ce chapitre les différentes technologies, les Frameworks et outils utilisés afin d'implémenter notre application web.

Ce travail nous a été très formateur, puisqu'il nous a permis d'apprendre et travailler avec de nouvelles technologies, et nous a également permis d'apprendre à gérer le temps dans un projet informatique, de plus, ce projet nous a permis d'approfondir nos connaissances dans le domaine de l'ingénierie logicielle.

Perspectives :

Ceci n'est que la première version de ce site web, en effet plusieurs perspectives s'offrent à ce projet :

Afin de faciliter la création de cours et tests, nous allons ajouter un système de création de quiz avec plusieurs Templates prédéfinies.

Pour créer des liens entre les élèves et les enseignants, nous allons ajouter une messagerie instantanée et un espace « commentaires » pour évaluer les cours.

Finalement, vu l'accomplissement de projet, nous souhaitons très fortement qu'il soit le fruit du progrès, de l'évolution et qu'il reste à la hauteur des exigences de la société.

Références bibliographiques et webographies:

Bibliographique

- [1] Jean-paul blan, pour comprendre les mathématiques .hachette éducation ,2014.Consulté le 11/04/2021 à 18h14
- [2] Lucien castioni,Christelle Allilaire, Patricia Le Trong, aurélie Romet .Maths explicites.Hachette Education Maths Explicites, 2019. Consulté le 20/05/2021 à 11h08

Webographie

- [3] Motivation en apprentissage : <https://journals.openedition.org/apliut/2889> consulté le 10/05/2021 à 10 h 30
- [4] Recours à la technologie dans l'éducation pour apprendre et enseigner : <https://www.intel.fr/content/www/fr/fr/education/technology-in-education-for-learning-and-teaching.html> consulté le 15/05/2021 à 14h20
- [5] Science et technologie a l'école primaire : https://www.academie-sciences.fr/pdf/rapport/202011_science techno ecole primaire.pdf consulté le 18/05/2021 à 12 h 10
- [6] Visual studio code sur : <https://code.visualstudio.com/> consulté le 03/06/2021 à 18 h 01
- [7] Wamp server sur : <https://www.wampserver.com/> consulté le 15/07/2021 à 10 h 10
- [8] ArgoUML sur : http://www-igm.univ-mlv.fr/~dr/XPOSE2000/Site/Version_redigee/ArgoUML.htm consulté le 1/07/2021 à 17h30
- [9] Adobe Illustrator : <http://www.ordinateur.cc/Logiciel/Adobe->

[Illustrator/96877.html](#) consulté le 20/07/2021 à 15 h 05

[10] Présentation de Framework sur :

<https://www.techno-science.net/glossaire-definition/Framework.html> consulté le 05/07/2021 à 11h15.

[11] Documentation sur Laravel : <https://laravel.com/docs/8.x> consulté le 06/07/2021 à 19h10.

[12] Présentation de Bootstrap : <https://whatis.techtarget.com/definition/bootstrap> consulté le 05/07/2021 à 14h.

[13] HTML sur :

<https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203255-html-hypertext-markup-langage-definition-traduction/> /consulté le 10/05/2021 à 13 h 01.

[14] CSS sur :

<https://www.futura-sciences.com/tech/definitions/internet-css/> consulté le 10/08/2021 à 13 h 01

[15] JavaScript sur :

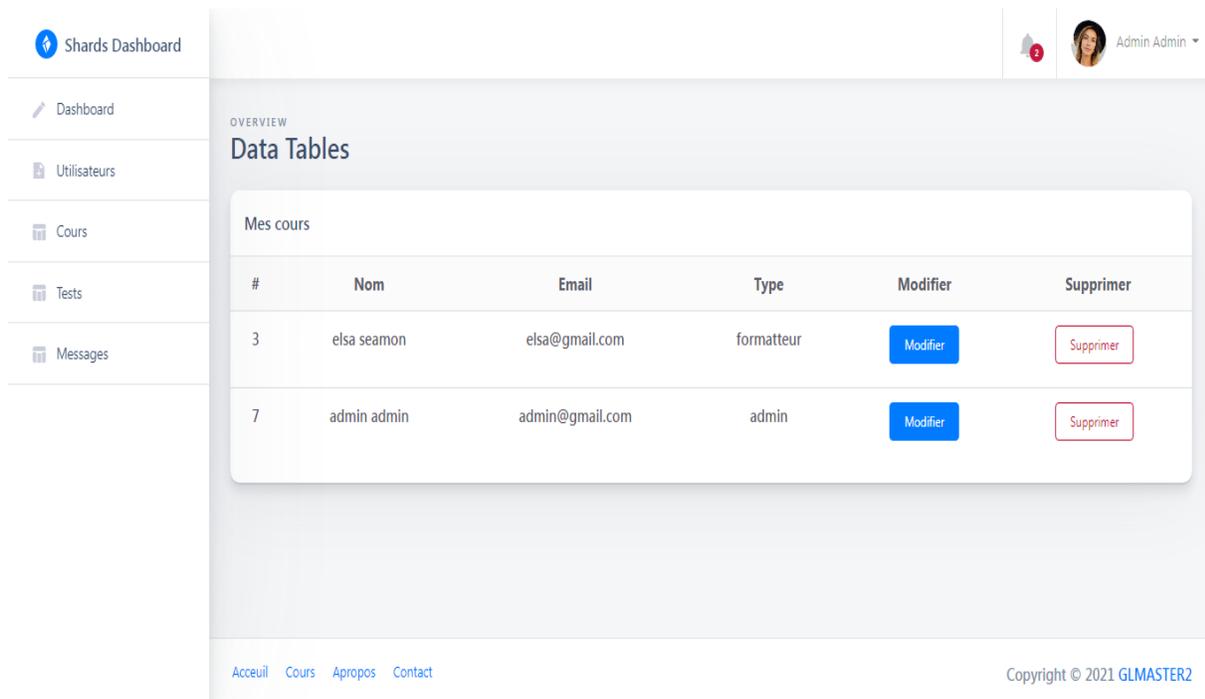
https://developer.mozilla.org/fr/docs/Learn/JavaScript/First_steps/What_is_JavaScript consulté le 10/08/2021 à 21 h 30.

[16] Découvrez le Framework phpLaravel sur :

<https://openclassrooms.com/courses/decouvrez-le-framework-php-laravel-1> consulté le 13/08/2021 à 19 h 26

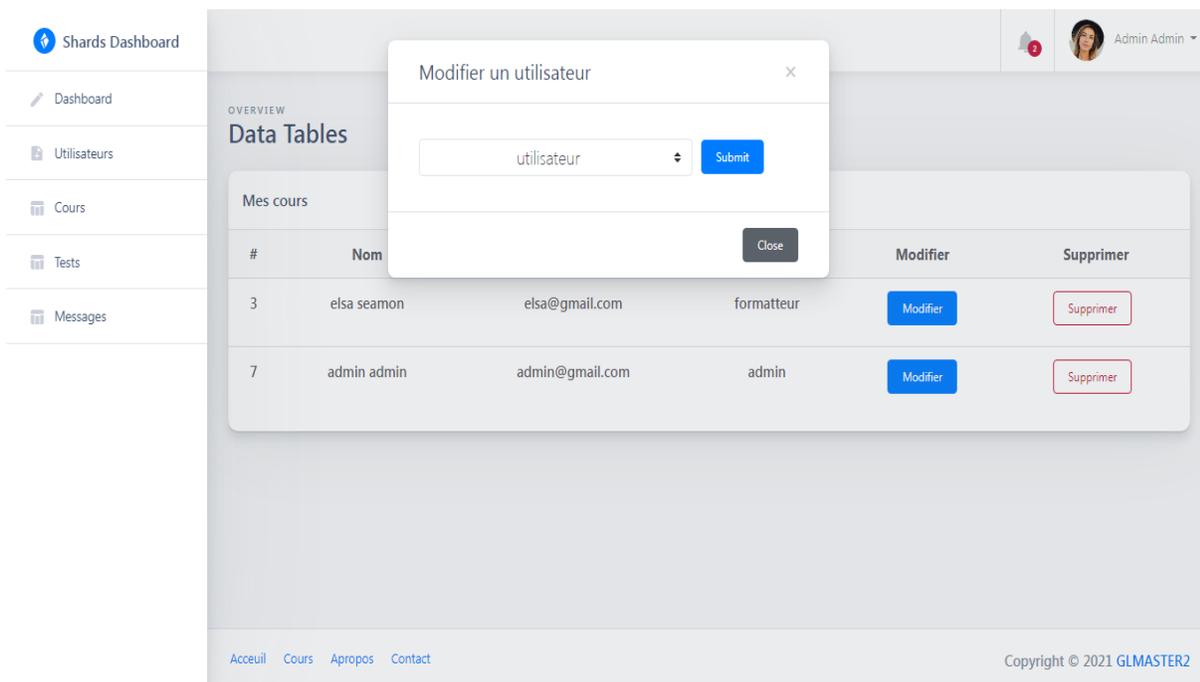
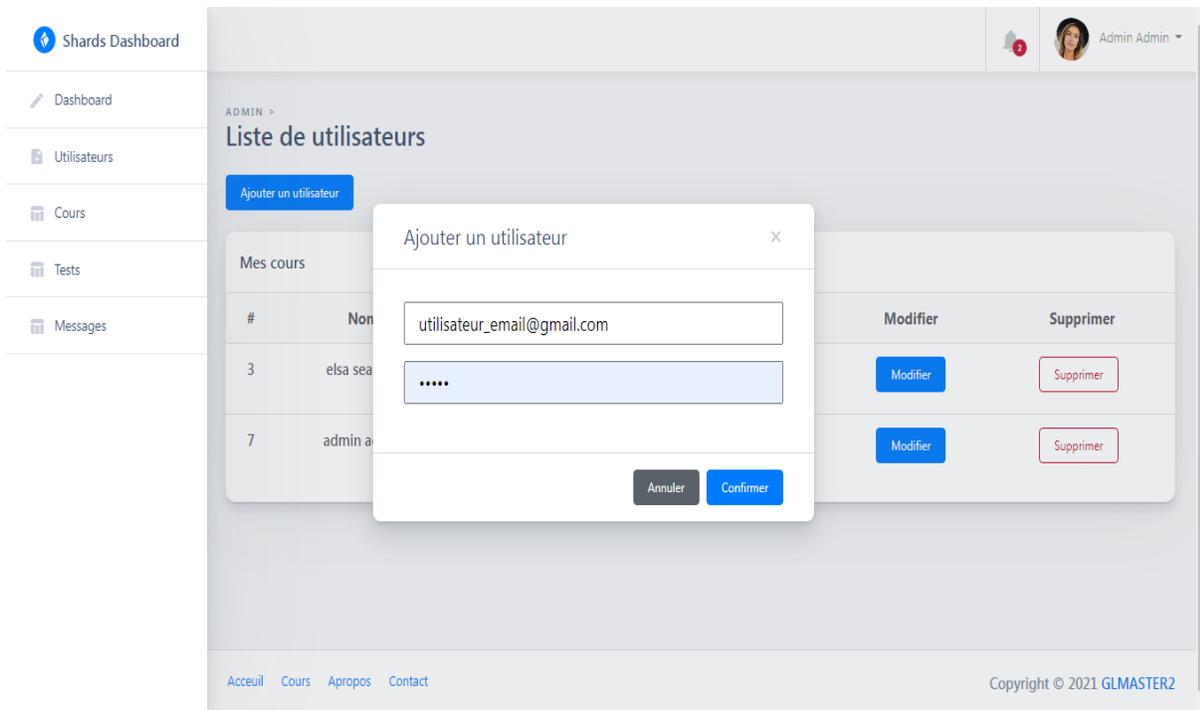
Annexe

Images qui représentent les étapes de gestion d'utilisateurs



The image shows a screenshot of a web dashboard. On the left is a vertical sidebar with navigation items: 'Shards Dashboard', 'Dashboard', 'Utilisateurs', 'Cours', 'Tests', and 'Messages'. The main content area is titled 'OVERVIEW Data Tables' and contains a table with the heading 'Mes cours'. The table has six columns: '#', 'Nom', 'Email', 'Type', 'Modifier', and 'Supprimer'. It lists two users: 'elsa seamon' (formatteur) and 'admin admin' (admin). Each row has a blue 'Modifier' button and a red 'Supprimer' button. The footer includes links for 'Accueil', 'Cours', 'Apropos', and 'Contact', and a copyright notice for 2021 GLMASTER2.

#	Nom	Email	Type	Modifier	Supprimer
3	elsa seamon	elsa@gmail.com	formatteur	Modifier	Supprimer
7	admin admin	admin@gmail.com	admin	Modifier	Supprimer



3.1. Description textuelle pour cas d'utilisation « Gestion des utilisateurs »

Objectif : Gérer les utilisateurs soit par l'ajout, la modification des droits d'accès ou bien la suppression.

➤ **Sous cas « ajouterutilisateur »**

Sommaire d'identification
Titre : Ajouter un utilisateur. Résumé : Cette fonctionnalité permet à l'administrateur d'ajouter des utilisateurs. Acteur : administrateur.
Description de l'enchaînement
Pré conditions : Authentification, non existence de l'utilisateur. Post conditions : l'utilisateur ajouté. Scénario nominal : <ol style="list-style-type: none">1. L'administrateur tape l'identifiant de l'utilisateur.2. Le système vérifier l'existence de l'identifiant.3. L'administrateur lui attribue un nom d'utilisateur et un mot de passe.4. le system affiche le message de confirmation. Scénario d'exception : <ul style="list-style-type: none">• L'utilisateur est déjà existant, le système va afficher un message d'erreur car l'opération est impossible.

Table3.1: Description textuelle du cas d'utilisation ajouter utilisateur

➤ **Sous cas«supprimerutilisateur »**

Sommaire d'identification
Titre : Supprimer un utilisateur. Résumé : Cette fonctionnalité permet à l'administrateur de supprimer des utilisateurs.

Acteur : administrateur.
Description de l'enchaînement
<p>Pré conditions : Authentification.</p> <p>Post conditions : l'utilisateur supprimé.</p> <p>Scénario nominal :</p> <ol style="list-style-type: none"> 1. L'administrateur tape l'identifiant de l'utilisateur. 2. Le système vérifier l'existence de l'identifiant. 3. L'administrateur supprime l'utilisateur ainsi que les coordonnées qui le concerne. 4. le system affiche le message de confirmation. <p>Scénario d'exception :</p> <ul style="list-style-type: none"> • Si l'identifiant de l'utilisateur n'existe pas dans la base de données le système vas afficher un message d'erreur car l'opération est impossible.

Table3.2: Description textuelle du cas d'utilisation supprimer utilisateur

➤ **Sous cas « modifier les droits d'accès d'un utilisateur »**

Sommaire d'identification
<p>Titre : modifier les droits d'accès d'un utilisateur.</p> <p>Résumé : Cette fonctionnalité permet à l'administrateur de modifier les droits d'accès des utilisateurs.</p> <p>Acteur : administrateur.</p>
Description de l'enchaînement
<p>Pré conditions : Authentification, utilisateur avec droits d'accès initiaux.</p> <p>Post conditions : utilisateur avec droits d'accès modifiés.</p> <p>Scénario nominal :</p> <ol style="list-style-type: none"> 1. L'administrateur tape l'identifiant de l'utilisateur. 2. Le système vérifier l'existence de l'identifiant. 3. L'administrateur modifie les droits d'accès de l'utilisateur de l'application. 4. le system affiche le message de confirmation.

<p>Scénario d'exception :</p> <ul style="list-style-type: none"> • Si l'identifiant de l'utilisateur n'existe pas dans la base de données le système vas afficher un message d'erreur car l'opération est impossible.

Table3.3: Description textuelle du cas d'utilisation Modifier l'accès au utilisateur

3.2. Description textuelle pour cas d'utilisation « Recherche cours» :

Sommaire d'identification
<p>Titre : Recherche cours.</p> <p>Résumé ce cas d'utilisation permet de trouver rapidement les informations Nécessaires sur un cours particulier.</p> <p>Acteur : élève.</p>
Description de l'enchaînement
<p>Pré condition : Authentification.</p> <p>Post condition : Cours trouvé.</p> <p>Scénario nominale :</p> <ol style="list-style-type: none"> 1. L'élève lance une recherche en tapant des mots clés. 2. Le système affiche le cours. <p>Scénarios alternatifs :</p> <ul style="list-style-type: none"> • Cours non existant affichage d'un message d'erreur.

Table3.4: Description textuelle du cas d'utilisation recherche cours

3.4. Description textuelle pour cas d'utilisation « Passer les tests» :

<p>Titre : Passer les tests.</p> <p>Résumé ce cas d'utilisation concerne l'élève après derrière chaque cours un test d'examen et réserver pour lui afin d'évaluer son niveau.</p> <p>Acteur : élève.</p>
Description de l'enchaînement

<p>Pré condition : Authentification.</p> <p>Post condition : test passé.</p> <p>Scénario nominale :</p> <ol style="list-style-type: none"> 1. L'élève demande de passer le test. 2. Le système affiche le test. 3. L'élève passe le test et le valide. 4. Le système rend la note.

Table3.5 : Description textuelle du cas d'utilisation passer les tests

3.5. Description textuelle pour cas d'utilisation « Contacte » :

<p>Titre : Contacte.</p> <p>Résumé ce cas d'utilisation concerne l'élève et le formateur pour l'élève s'il veut contacter son professeur et pour formateur s'il veut contacter l'administrateur.</p> <p>Acteur : Formateur, Elève.</p>
<p>Description de l'enchaînement</p>
<p>Pré condition : Authentification.</p> <p>Post condition : Message envoyé.</p> <p>Scénario nominale :</p> <ol style="list-style-type: none"> 1. L'élève ou le formateur demande le formulaire de contacte. 2. Le système affiche le formulaire. 3. L'élève ou le formateur remplissent les informations nécessaires puis valides. 4- Le système affiche un message de confirmation. <p>Scénario alternatif :</p> <ul style="list-style-type: none"> • Nature des champs saisie incorrecte, Le système signale une erreur des champs saisis. • Champs obligatoires vides, Le système réaffiche le formulaire

Table3.6: Description textuelle du cas d'utilisation contacte

Résumé

Le développement d'Internet et des outils de communication a permis de développer l'éducation par la naissance du mot: formation à distance.

L'objectif de l'apprentissage en ligne est de remplacer l'ancien temps, le lieu et la manière prédéterminée d'apprendre le contenu par un processus d'apprentissage rapide, ouvert et personnalisé.

Notre travail comprend un site web d'apprentissage à distance pour se faire nous avons utilisé le formalisme UML et le processus UP pour la conception de notre application ,ensuite pour l'implémentation nous avons utilisé le SGBD MySQL pour gérer notre base données. Enfin pour la création de notre application nous avons utilisé le fameux Framework d'application web Laravel qui est basé sur le langage PHP, il est exécuté par le logiciel Visual studio code.

Mots clés :

Education, site web, apprentissage en ligne, apprentissage à distance,UML, UP, MySQL, PHP, Visual studio code.

Abstract

The development of the Internet and communication tools has made it possible to develop education through the birth of the word: distance education.

The goal of online learning is to replace the old time, place and predetermined way of learning content with a fast, open and personalized learning process.

Our work includes a distance learning website to do this we used the UML formalism and the UP process for the design of our application, then for the implementation we used the MySQL DBMS to manage our database. Finally for the creation of our application we used the famous Laravel web application framework which is based on the PHP language; it is executed by the Visual studio code software.

Key words:

Education, website, distance learning, UML, UP, MySQL, PHP, Visual studio code.