



Université Mohammed Premier
École Nationale des Sciences Appliquées
- Oujda-

Rapport de projet de fin d'Année

Filière: Génie Informatique

Conception et développement d'une application web
PFEProgress

Réalisé Par :

BOULOUANE IMANE
BOUMESHOUL HAFIDA
CHARKAOUI WIJDANE

Encadré Par :

Mme BELLOUALI SAIDA
Monsieur BOUCHENTOUF TOUMI

Année universitaire : 2021/2022

Remerciements

En préambule à ce rapport, nous tenons à témoigner nos sincères remerciements à toutes les personnes qui ont contribué de près ou de loin à l'élaboration de ce modeste travail.

Nous tenons, dans un premier temps, à exprimer notre profonde reconnaissance à nos encadrants Mr. Bouchentouf Toumi et Mme Saida Belouali, pour leur patience, leur disponibilité et surtout leurs judicieux conseils durant la réalisation de ce projet de fin d'année.

Nous tenons également à remercier toute l'équipe pédagogique de l'ENSAO et les intervenants professionnels responsables de la filière génie informatique.

Nous exprimons notre sincère gratitude aux membres du jury, qui ont accepté d'évaluer ce projet de fin d'année.

Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis qui nous ont toujours accordé leur soutien, tant par leur gentillesse que par leur dévouement.

Table des abréviations

Abréviation	Désignation
JWT	JSON Web Token
HTTP	Hyper Text Transfer Protocol
API	Application Programming Interface
SGBD	Système de Gestion de Base de Données
SQL	Structured Query Language
UML	Unified Modeling Language

Table de figures

Figure1 : Diagramme des cas d'utilisation	13
Figure2 : Diagramme de classe	15
Figure3 : Diagramme de séquence « Authentification »	16
Figure4 : Diagramme de séquence « Affectation d'encadrant »	16
Figure5 : Diagramme de séquence « Gestion de soutenance »	17
Figure6 : Architecture 3-tiers	19
Figure7 : Interface d'authentification	25
Figure8 : Message d'erreur d'authentification.....	25
Figure9 : Interface d'accueil.....	26
Figure10 : Alerte « Encadrant supprimée »	26
Figure11 : Interface des encadrants.....	27
Figure12 : Interface Encadrant Details	27
Figure13 : Interface Ajouter Encadrant.....	28
Figure14 : Interface des étudiants	29
Figure15 : Alerte « étudiant supprimée »	29
Figure16 : Affecter Encadrant	30
Figure17 : Alerte « Valider l'affectation de l'encadrant »	30
Figure18 : Interface Ajouter étudiant	31
Figure19 : Fichier Excel des étudiants et PFE	31
Figure20 : Interface Etudiants Details	32
Figure21 : Alerte soutenance affectée avec succès.....	32
Figure22 : Alerte « Etudiant a déjà une soutenance »	33
Figure23 : Alerte Jury doivent être différents de l'encadrant	33
Figure24 : Alerte « Jurys doivent être différents ».....	33
Figure25 : Alerte « Jury n'est pas disponible ».....	34
Figure26 : Alerte « Interface des soutenances »	34
Figure27 : Alerte « Interface Soutenance Details ».....	35
Figure28 : Menu de l'application	36
Figure29 : Espace Etudiant.....	37

Table de matières

Introduction générale	6
Chapitre I : Présentation du projet.....	7
Introduction	6
Présentation générale du projet	6
Problématique	6
Solution proposée	6
Conclusion	6
Chapitre II : Spécification des besoins	9
Introduction	10
Besoins fonctionnels	10
Besoins non fonctionnels	11
Traduction des besoins en diagrammes de cas d'utilisation	11
Description des acteurs	11
Diagramme de cas d'utilisation global	13
Conclusion.....	13
Chapitre III : Conception et modélisation du système	14
Introduction	15
Conception générale	15
Diagramme de classes	15
Analyse.....	16
Modèle dynamique (diagrammes de séquences)	16
Diagramme d'authentification à l'application.....	16
Diagramme d'affectation d'encadrant	16
Diagramme de gestion de soutenance	17
Conclusion	17
Chapitre IV : Architecture de l'application.....	18

Introduction	19
Architecture client - serveur	19
Architecture 3-tiers	19
Environnement technique	20
Backend	20
Choix SGBD	21
Frontend	21
Environnement logiciel	22
Chapitre V : Implémentation	24
Introduction	25
Les interfaces de l'application web	25
Interface d'authentification	25
Interface d'accueil	26
Interface des encadrants	26
Interface Encadrant Détails	27
Interface Ajouter Encadrant	28
Interface des étudiants	29
Interface Ajouter Etudiant	30
Interface Etudiants Détails	32
Interface des soutenances	34
Interface Soutenance Détails	35
Exigences de suppression des encadrants et des étudiants	35
Menu de l'application	36
Interface Espace Etudiant	37
Conclusion et perspectives	38
Bibliographie	39

Introduction générale

La gestion est le pilotage de l'action collective au sein d'une organisation. Elle s'appuie notamment sur l'étude des organisations, objet des sciences de gestion.

Dans ce contexte, notre département informatique a proposé de lancer un projet qui consiste à développer une application pour automatiser le processus des projets PFE et des soutenances, en accompagnant tous les acteurs impliqués dans ce processus, en allant de la phase d'affectation d'encadrant jusqu'à le jour de la soutenance.

L'objectif de ce projet est d'analyser, concevoir et développer une application de gestion des PFE et des soutenances.

Ce projet doit permettre, en premier lieu, l'échange de l'information et de la communication entre les étudiants et les encadrants, le contrôle, le suivi et l'évaluation des mémoires. En deuxième lieu, l'application doit assurer une partie de paramétrage qui intégrera la gestion des soutenances, la gestion des encadrants, la gestion des étudiants, et la gestion des profils de chaque utilisateur.

Pour le faire, le présent rapport doit être bien structuré pour être exploité après la mise en place de l'application, et nous l'avons organisé de la manière suivante :

- Le premier chapitre donne une présentation générale du projet.
- Le deuxième chapitre intitulé « Spécification des besoins », présente les différents besoins fonctionnels et non fonctionnels auxquels doit satisfaire l'application.
- La conception de notre solution sera présentée dans le troisième chapitre.
- Le quatrième chapitre décrit l'architecture de l'application.
- Le cinquième chapitre présente l'implémentation de l'application (les tests)
- Et nous allons finir par une conclusion et des perspectives.

Chapitre 1

Présentation du projet

Introduction

Ce chapitre a pour objectif de situer le projet dans son contexte général, à savoir la problématique qui a inspiré la création de notre application, la description du projet et la solution proposée.

Présentation générale du projet

Problématique

La gestion des projets PFE est un problème majeur pour les institutions éducatives et universitaires. Le défi est de garantir les bonnes conditions de l'affectation de l'encadrant jusqu'à la soutenance.

Les méthodes manuelles traditionnelles ont montré leurs limites dans la gestion de grands flux d'informations notamment lorsque le nombre d'étudiants est important. De ce fait, l'intérêt est d'automatiser le processus de gestion des PFE et des soutenances.

Cette problématique a été le point de départ de notre projet. On a fixé comme objectif d'atténuer les problèmes qui se posent lors de l'affectation d'un encadrant, notamment lors du choix de la date de soutenance.

Solution proposée

Contrairement à la pratique traditionnelle d'affectation des encadrants et de gestion des soutenances qui engendre beaucoup de problèmes pour le chef de filière et les encadrants, notre projet donne au chef de filière les moyens nécessaires pour gérer les PFE et les soutenances dans les meilleures conditions.

Conclusion

Tout au long de ce chapitre, nous avons mis notre projet dans son cadre général, à savoir la présentation du projet et ses objectifs, et nous allons spécifier dans le prochain chapitre les différents besoins auxquels doit répondre notre application.

Chapitre 2

Spécification des besoins

Introduction

La réussite de tout projet dépend de la qualité de son départ. De ce fait, l'étape de spécification des besoins constitue la base de départ de notre travail, elle doit décrire sans ambiguïté le logiciel à développer.

Pour assurer les objectifs attendus, il est essentiel que nous parvenions à une vue claire des différents besoins escomptés de notre projet. Au cours de ce chapitre, nous allons dégager les fonctionnalités attendues du module **gestion des PFE et soutenances** en définissant les différents cas d'utilisation et quelques scénarios qui expliquent ces cas.

Besoins fonctionnels

Le module **gestion des PFE et soutenances** doit en premier temps mettre à la disposition de l'administrateur les fonctionnalités suivantes :

- L'administrateur peut ajouter, supprimer un étudiant ou bien un encadrant.
- L'administrateur peut importer un fichier Excel contenant la liste des étudiants.
- L'administrateur peut affecter un encadrant à un étudiant.
- L'administrateur peut gérer les soutenances des étudiants.
- L'administrateur peut visualiser la date des soutenances.

Le module doit également permettre à l'utilisateur (Etudiant) de gérer la liste des tâches (ToDo List) à effectuer.

- Ajouter, modifier ou bien supprimer une tâche.
Ce module doit mettre à la disposition des encadrants de :
- Consulter la liste des tâches des étudiants.

Besoins non-fonctionnels

- L'application doit être facile à utiliser.
- Le temps de réponse de l'application doit être le plus rapide possible.
- Possibilité d'optimiser l'application plus tard.
- L'application doit sécuriser les informations de ses utilisateurs.

Traduction des besoins en diagrammes de cas d'utilisation :

Le diagramme des cas d'utilisation est un modèle simplifié du fonctionnement de l'application qui spécifie les acteurs qui l'utilisent et les services qu'il leur offre.

Description des acteurs

Un acteur représente une abstraction d'un rôle joué par des entités externes au système qui interagissent directement avec lui.

Dans cette application web nous identifions l'administrateur et l'utilisateur.

L'utilisateur : il s'agit de l'étudiant et l'encadrant

L'administrateur : c'est le chef de filière, c'est lui qui gère les fonctionnalités principales de l'application.

Diagramme de cas d'utilisation global

Le diagramme de la **figure 1** illustre le rôle de chaque acteur ainsi que l'étendue de ses responsabilités.

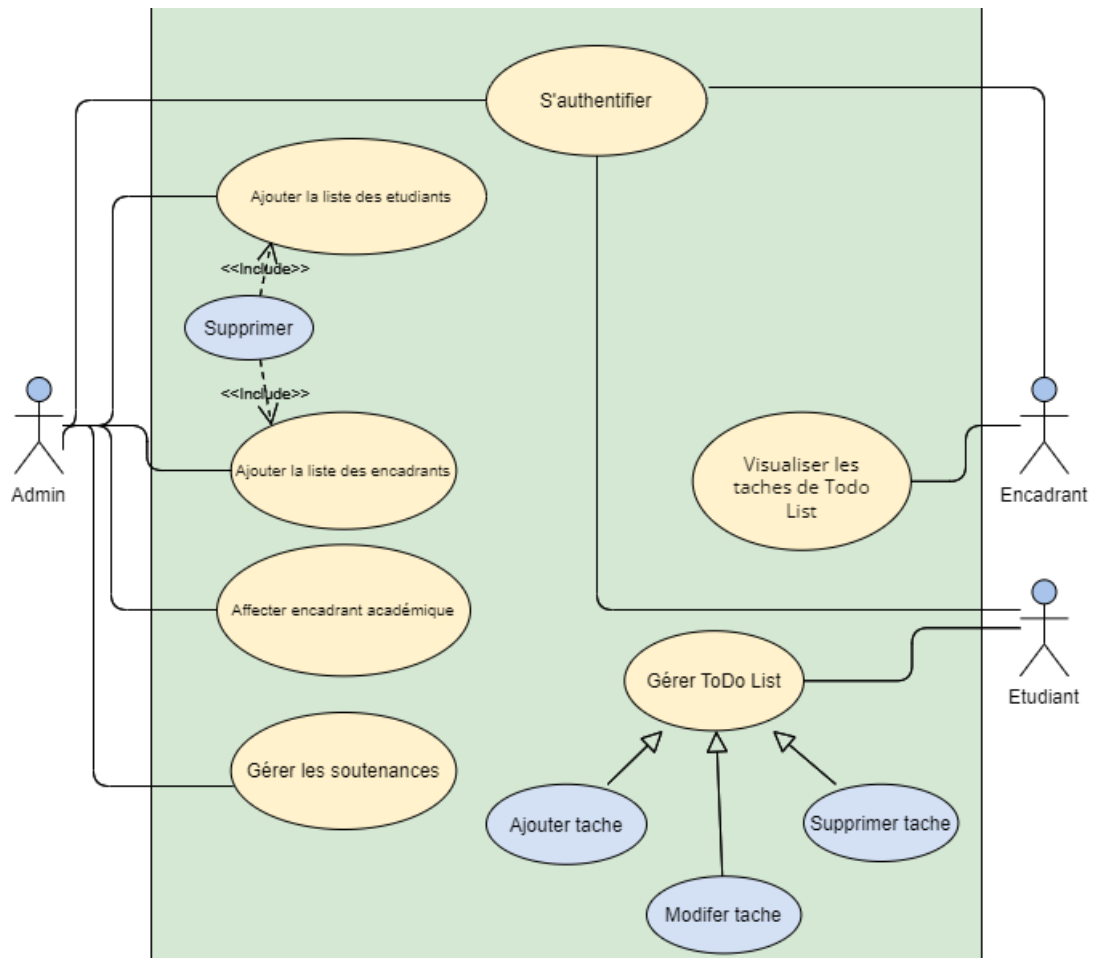


Figure 1: diagramme des cas d'utilisation

Conclusion

Ce chapitre nous a permis de couvrir les différents besoins fonctionnels et non fonctionnels auxquels doit satisfaire notre application. Nous avons aussi détaillé ces besoins à travers un diagramme des cas d'utilisation pour passer par la suite à la conception de notre application qui sera présentée dans le chapitre suivant.

Chapitre 3

Conception et modélisation du système

Introduction

Dans le présent chapitre, nous allons entamer une partie importante du développement de l'application qui constitue un pont entre la spéciation et la réalisation. Nous allons présenter dans un premier temps la conception générale de notre application, puis la conception détaillée comprenant les vues statiques via les diagrammes de classes ainsi que les diagrammes de séquence.

1. Conception générale

Diagramme de classes

Ce diagramme est le diagramme de classes abordé dans le développement de notre application en présentant les différentes classes.

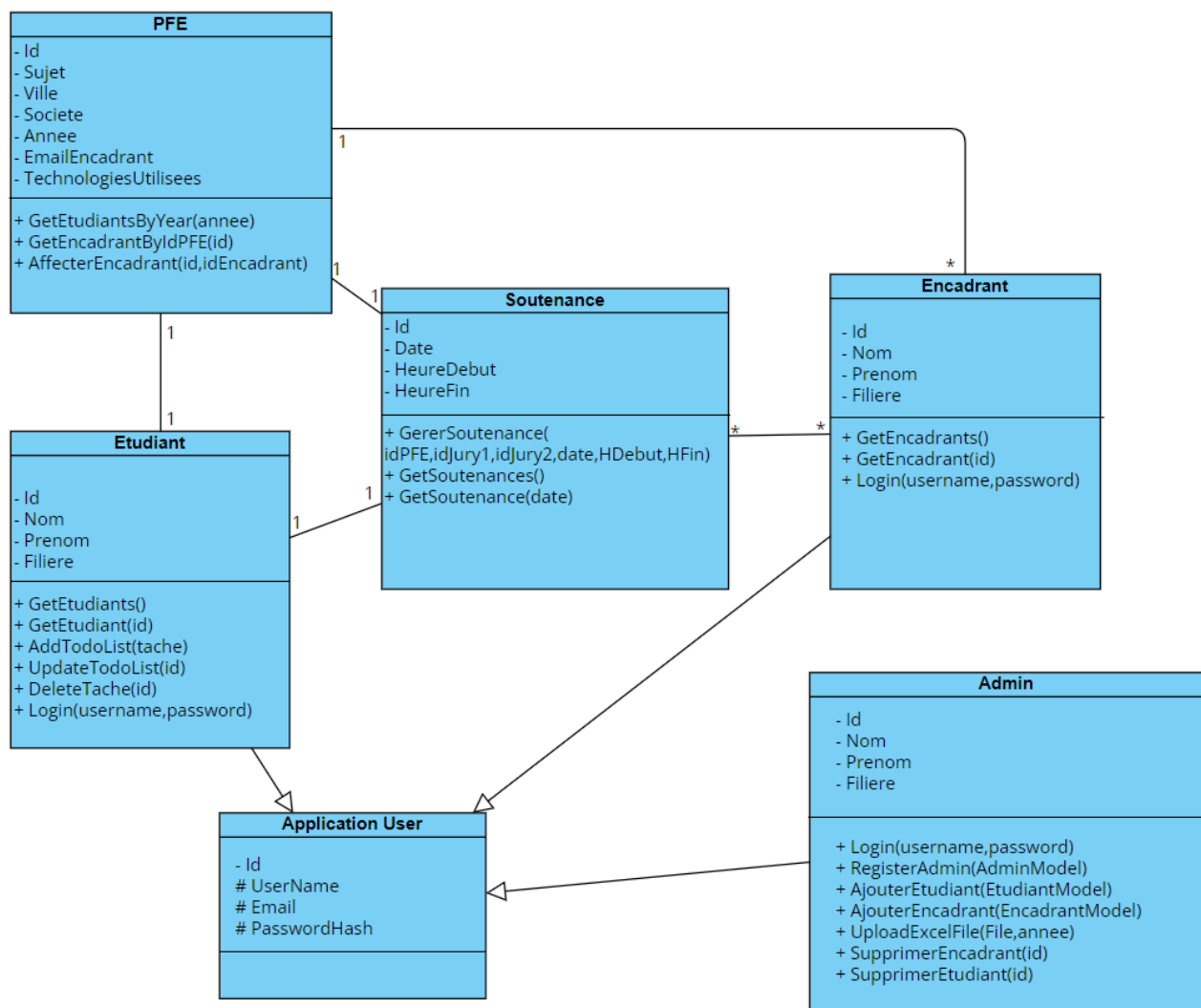


Figure 2: diagramme de class

2. Analyse

Modèle dynamique (diagrammes de séquence)

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML.

Diagramme de séquence

- *Diagramme d'authentification à l'application*

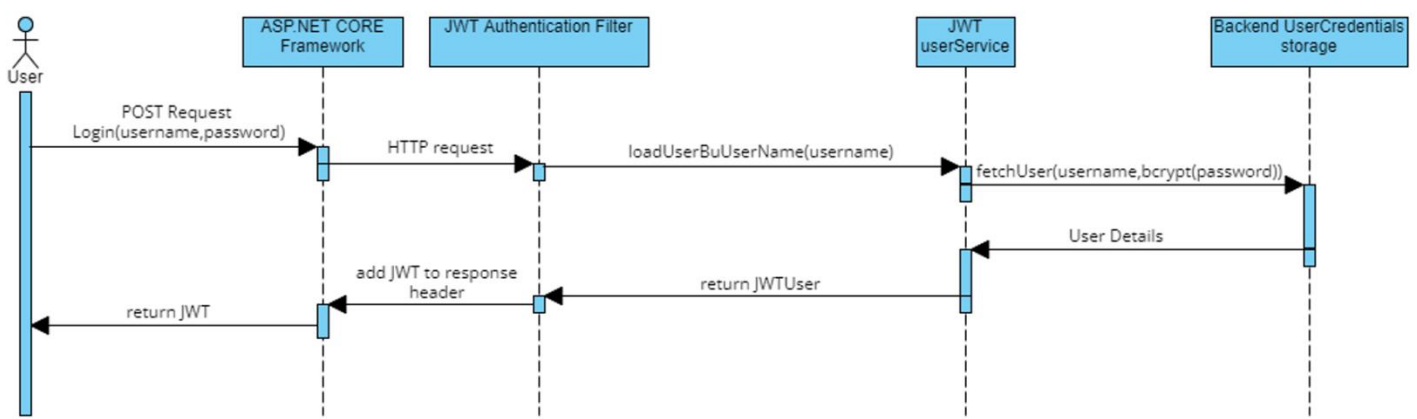


Figure 3: diagramme de séquence « Authentification »

- *Diagramme d'affectation d'encadrant:*

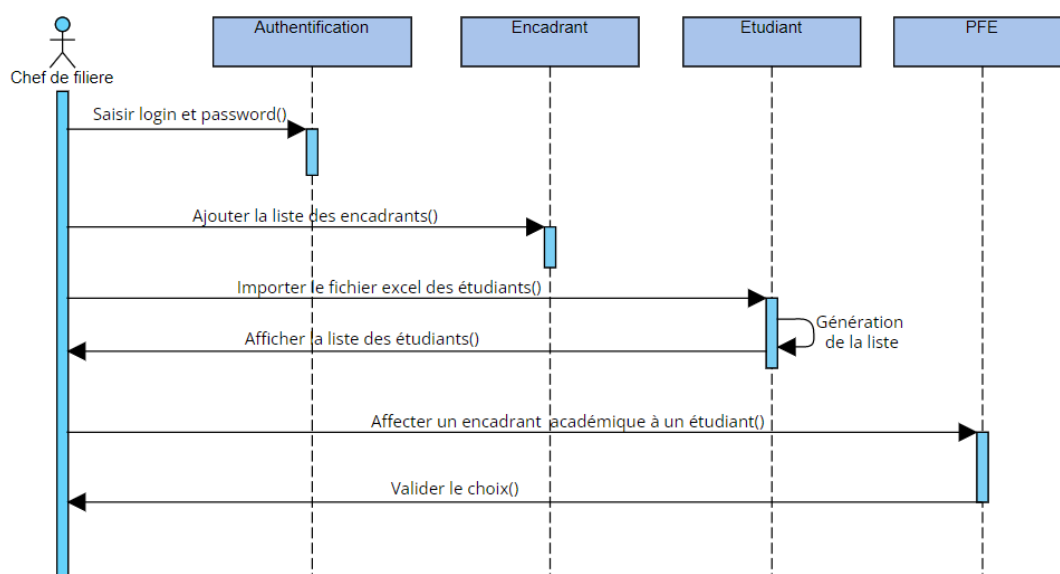


Figure 4: diagramme de séquence « Affectation d'encadrant »

- *Diagramme de gestion de soutenance :*

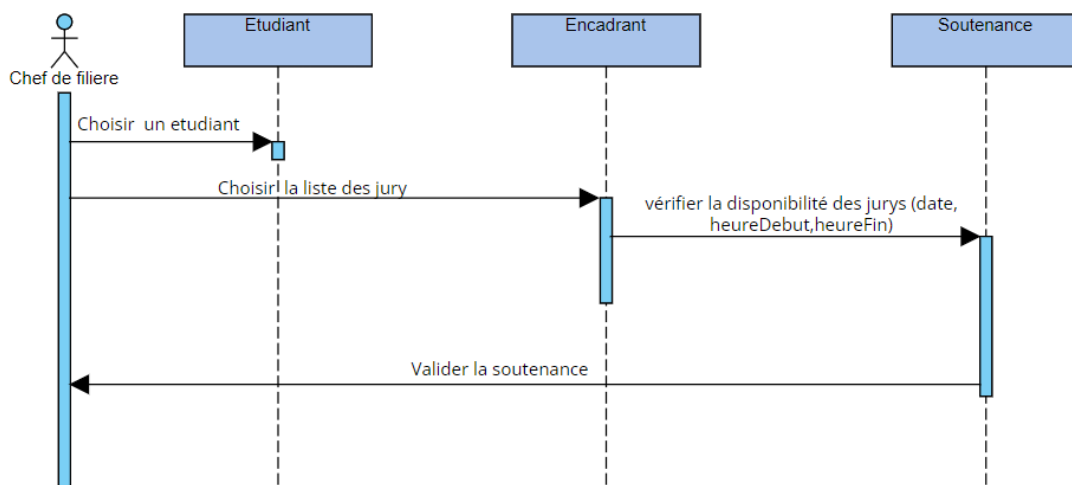


Figure 5: diagramme de séquence « Gestion de soutenance »

Conclusion

Dans ce chapitre, nous avons présenté la conception et l'analyse de notre système. Il a été consacré à la modélisation de l'aspect dynamique de notre application, en se basant sur les spécifications des chapitres précédents.

Dans le prochain chapitre nous abordons la partie réalisation ainsi que la description de quelques choix techniques effectués pendant le développement.

Chapitre 4

Architecture de l'application

Introduction

Après avoir entamé la partie de conception, ce chapitre présente les étapes nécessaires pour réaliser notre projet et décrit l'architecture de l'application ainsi que les outils et les langages de développement.

Architecture client-serveur

Une architecture client-serveur représente l'environnement dans lequel des applications de machines clientes communiquent avec des applications de machines de type serveurs via des requêtes (protocole HTTP).

Architecture 3-tiers

Notre application est d'architecture à 3 niveaux (architecture 3-tiers).

Il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre:

1. **Le client** : qui envoie la requête des ressources.
2. **Le serveur d'application** (appelé aussi *middleware*) le serveur chargé de fournir la ressource en faisant appel à un autre serveur.
3. **Le serveur Data** (un serveur de base de données), fournissant un service au premier serveur.

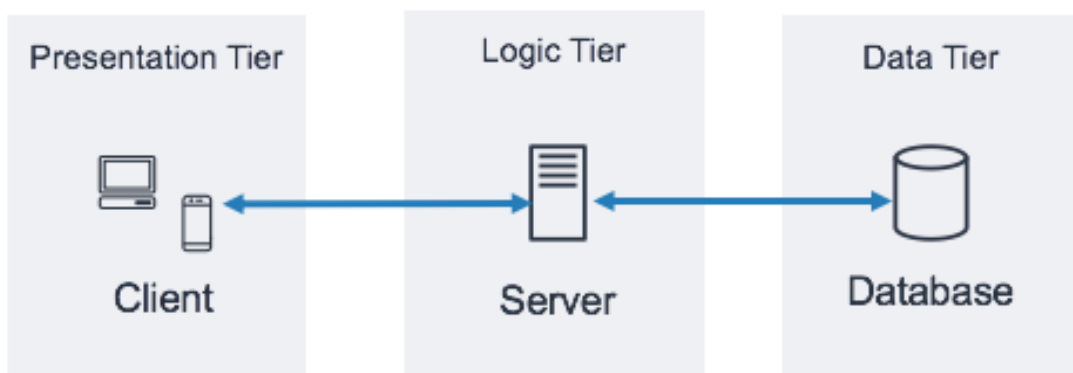


Figure 6: Architecture 3-tiers

Environnement technique

Backend :

Une API, ou interface de programmation d'application, est un ensemble de définitions et de protocoles qui facilite la création et l'intégration de logiciels d'applications.

FRAMEWORK ASP.NET CORE



ASP.NET Core est une infrastructure multiplateforme, haute performance et open source permettant de créer des applications modernes, compatibles avec le cloud et connectées à Internet.

ASP.NET Core est une nouvelle conception de ASP.NET 4.x, y compris les modifications architecturales qui entraînent un framework plus léger et plus modulaire. (Version 6.0)

Langage C#



C# Est un langage de programmation orienté objet à typage fort, créé par la société Microsoft. Il a été créé afin que la plate-forme Microsoft .NET soit dotée d'un langage permettant d'utiliser toutes ses capacités. Il est très proche du Java dont il reprend la syntaxe générale ainsi que les concepts (la syntaxe reste cependant relativement semblable à celles de langages tels que le C++ et le C).

ENTITY FRAMEWORK CORE

Est une version légère, extensible, open source et multiplateforme.

C'est un outil permettant de créer une couche d'accès aux données (DAL pour Data Access Layer) liées à une base de données relationnelle. Il propose la création d'un schéma conceptuel composé d'entités qui permettent la manipulation d'une source de données, sans écrire une seule ligne de SQL, grâce à LinQ To Entities. Comparé à d'autres solutions de mapping objet-relationnel (ORM), Entity Framework assure l'indépendance du schéma conceptuel (entités ou objets) du schéma logique de la base de données, c'est-à-dire des tables. Ainsi, le code produit et le modèle conceptuel ne sont pas couplés à une base de données spécifique.

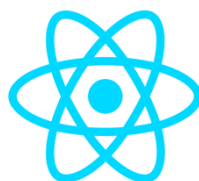
Choix SGBD : Microsoft SQLServer



Microsoft SQL Server est un **système de gestion de bases de données relationnelles** édité et commercialisé par **Microsoft** depuis 1994. C'est une plateforme de données d'entreprise permettant de gérer et stocker dans des bases de données tout type d'information.

Frontend :

ReactJS

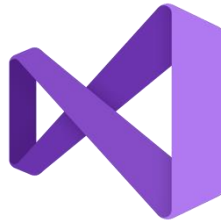


React est une bibliothèque JavaScript open source qui est utilisée pour construire des interfaces utilisateur spécifiquement pour des applications d'une seule page. Elle est utilisée pour gérer la couche d'affichage des applications web et mobiles. React nous permet également de créer des composants d'interface utilisateur réutilisables. (Version 17.0.2)

Environnement logiciel

Dans ce paragraphe nous décrivons les différents outils et logiciels utilisés pour la mise en œuvre de l'application.

Microsoft Visual studio



Microsoft Visual Studio est un environnement de développement intégré (IDE) de Microsoft. Il est utilisé pour développer des programmes informatiques, ainsi que des sites Web, des applications Web, des services Web et des applications mobiles. Visual Studio utilise des plates-formes de développement logiciel Microsoft telles que l'API Windows, Windows Forms, Windows Presentation Foundation, Windows Store et Microsoft Silverlight. Il peut produire à la fois du code natif et du code managé.

Visual studio code



Visual Studio Code est un éditeur de code multiplateforme édité par Microsoft. Cet outil destiné aux développeurs supporte plusieurs dizaines de langages de programmation comme le HTML, C++, PHP, Javascript, Markdown, CSS, etc. Visual Studio Code intègre plusieurs outils facilitant la saisie de code par les développeurs comme la coloration syntaxique ou encore le système d'autocomplétion IntelliSense.

Postman



Est un outil permettant de manipuler une API depuis une interface graphique agréable. Postman permet de construire et d'exécuter des requêtes HTTP, de les stocker dans un historique afin de pouvoir les rejouer, mais surtout de les organiser en Collections. Cette classification permet notamment de regrouper des requêtes de façon « fonctionnelle ».

GitHub



GitHub est une plate-forme de contrôle de version distribuée où les utilisateurs peuvent collaborer ou adopter des projets de code open source, fork du code, partager des idées et plus encore.

Git



Git est un système de contrôle de version distribué gratuit et open source conçue pour tout gérer, des petits aux très grands projets, avec rapidité et efficacité.

Chapitre 5

Implémentation

Introduction

Cette partie est consacrée à présenter les interfaces de l'application réalisée.

Interfaces de l'application web

L'interface d'authentification

Cette interface permet à l'utilisateur de l'application de s'authentifier en saisissant le username et le mot de passe. L'utilisateur ne peut pas accéder à la page d'accueil qu'après la phase d'authentification.

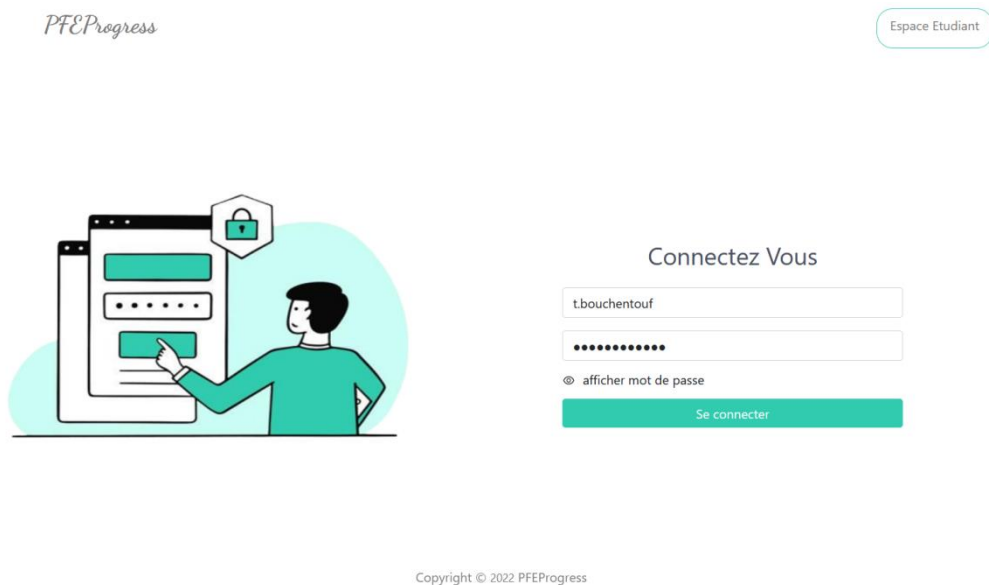


Figure 7: Interface d'authentification

Si les coordonnées saisies sont incorrectes, un message d'erreur est affiché :

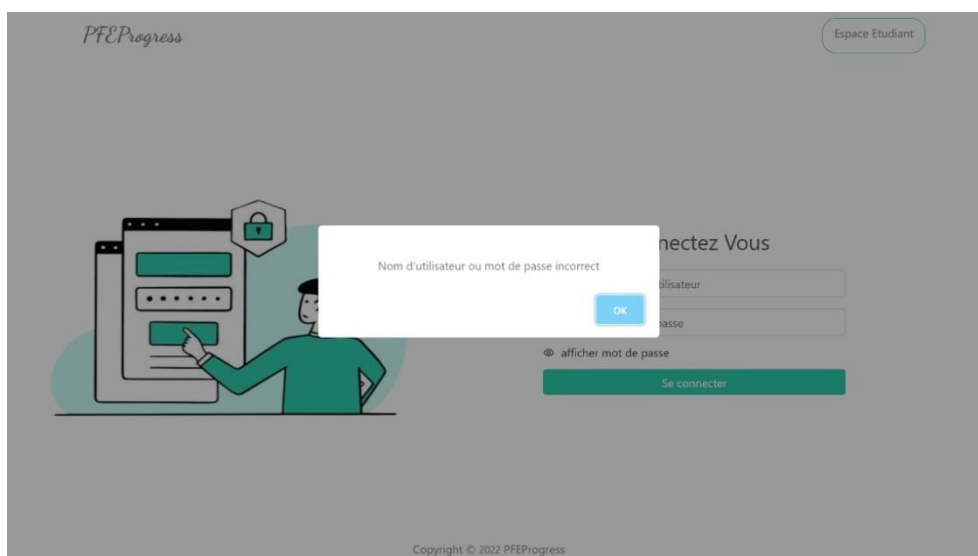


Figure 8: Message d'erreur d'authentification

L'interface d'accueil (Espace d'administrateur)

Cette interface liste les différentes interfaces de notre application, avec le lien pour y accéder.



Figure 9: Interface d'accueil

L'interface des encadrants

Cette interface affiche la liste des encadrants académiques du département informatique. L'administrateur peut à tout moment consulter le profil d'un encadrant ou bien de le supprimer.

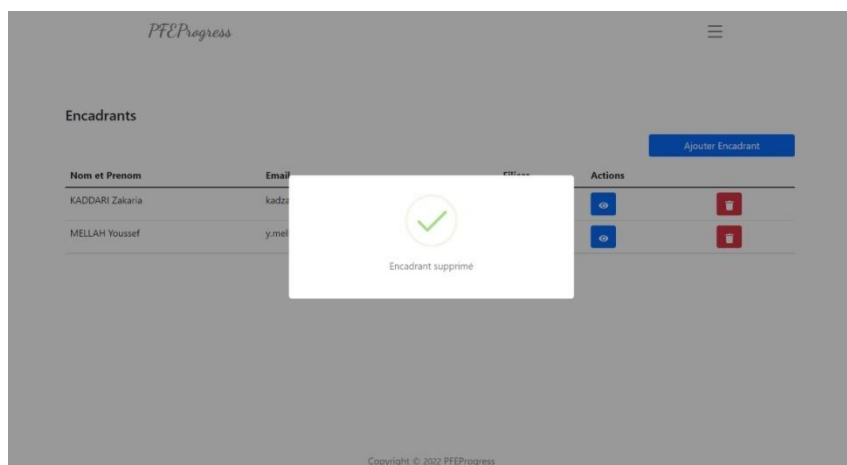


Figure 10: Alerte « Encadrant supprimé »

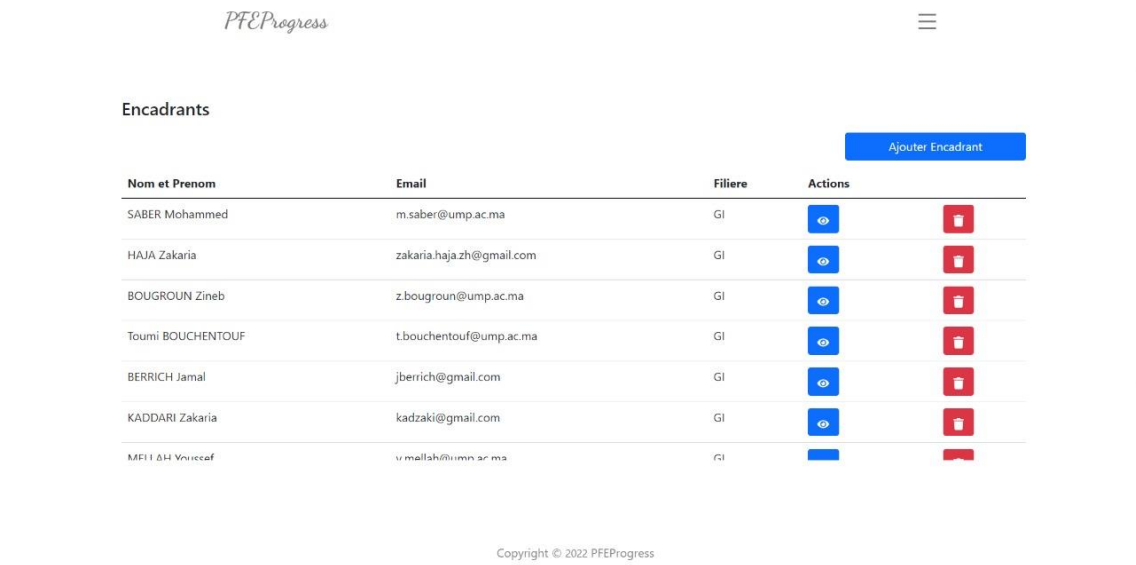


Figure 11: Interface des encadrants

L'interface Encadrant Détails

Cette interface affiche les informations d'un encadrant.

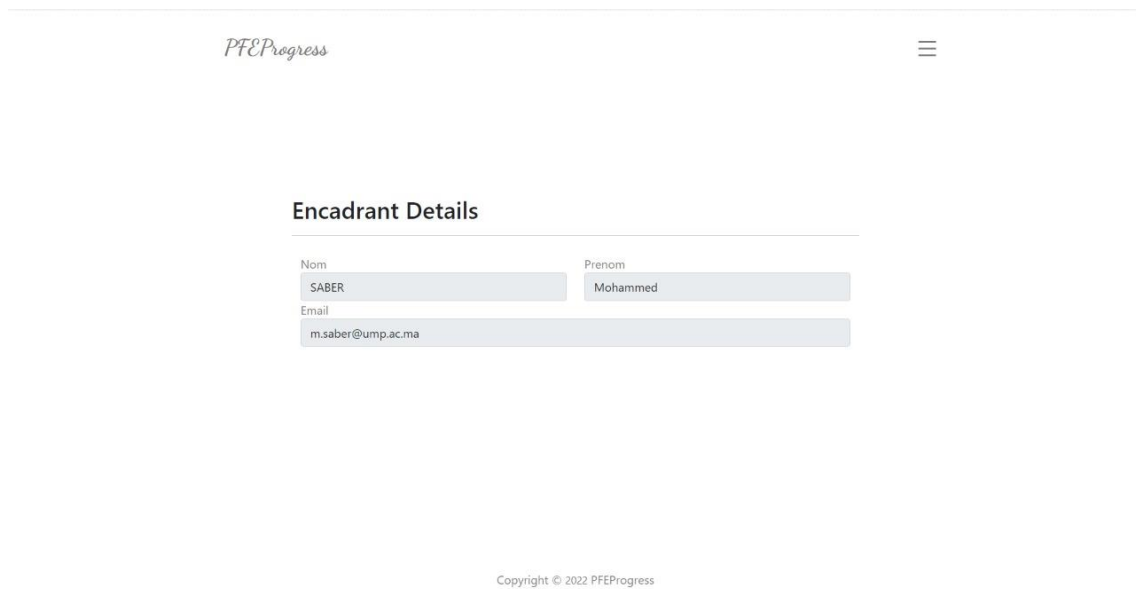
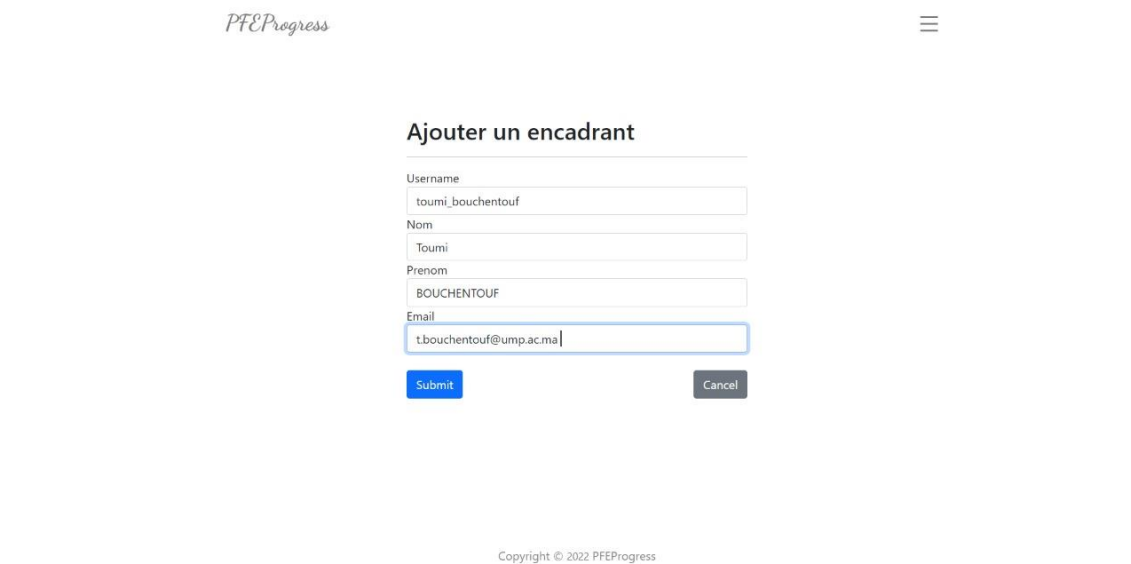


Figure 12: Interface Encadrant Détails

L'interface Ajouter Encadrant

L'administrateur doit ajouter les encadrants via le bouton **Ajouter Encadrant**.

Il suffit de remplir le formulaire avec les coordonnées de l'encadrant.

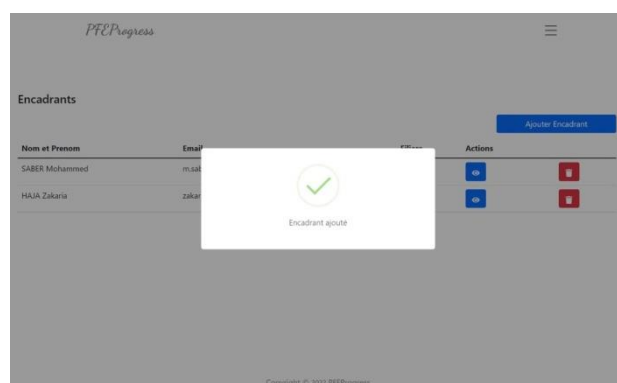
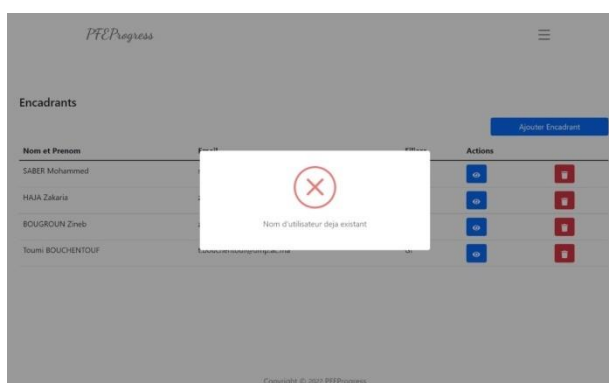


The screenshot shows the 'Ajouter un encadrant' form in the PFEProgress application. The form has a title 'Ajouter un encadrant' and a hamburger menu icon in the top right corner. The form fields are: Username (toumi_bouchentouf), Nom (Toumi), Prenom (BOUCHENTOUF), and Email (t.bouchentouf@ump.ac.ma). There are 'Submit' and 'Cancel' buttons at the bottom. The footer shows 'Copyright © 2022 PFEProgress'.

Figure 13: Interface « Ajouter Encadrant »

Si l'administrateur saisit des coordonnées existantes, une alerte est affichée.

Si non, l'encadrant est ajouté avec succès dans notre base de données.



L'interface des étudiants

Cette interface affiche la liste des étudiants du département informatique. L'administrateur peut à tout moment consulter le profil d'un étudiant ou bien de le supprimer ainsi que de lui affecter un encadrant académique.

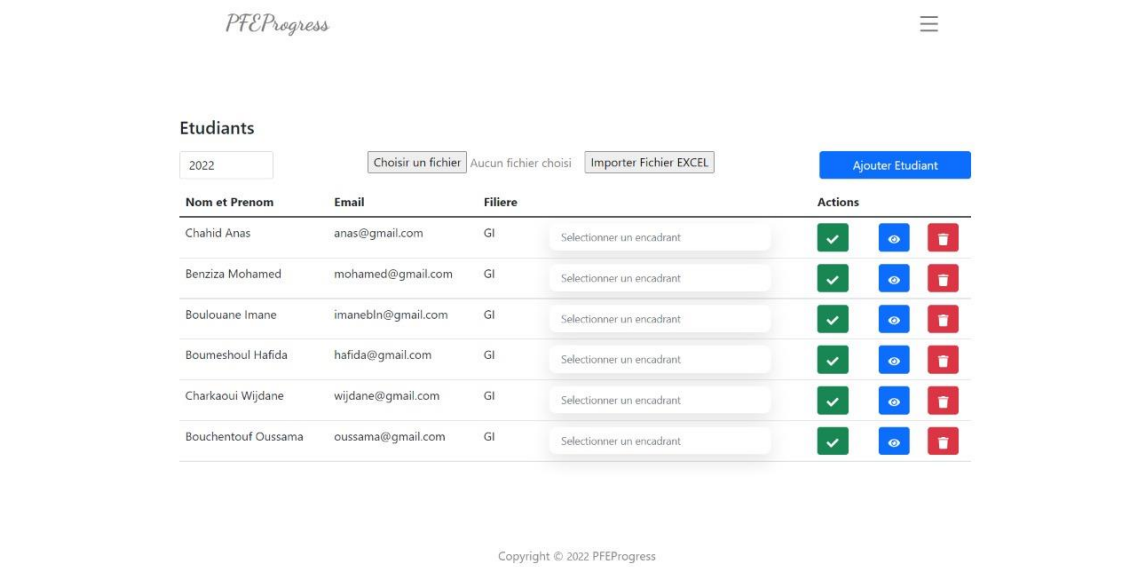


Figure 14: Interface des étudiants

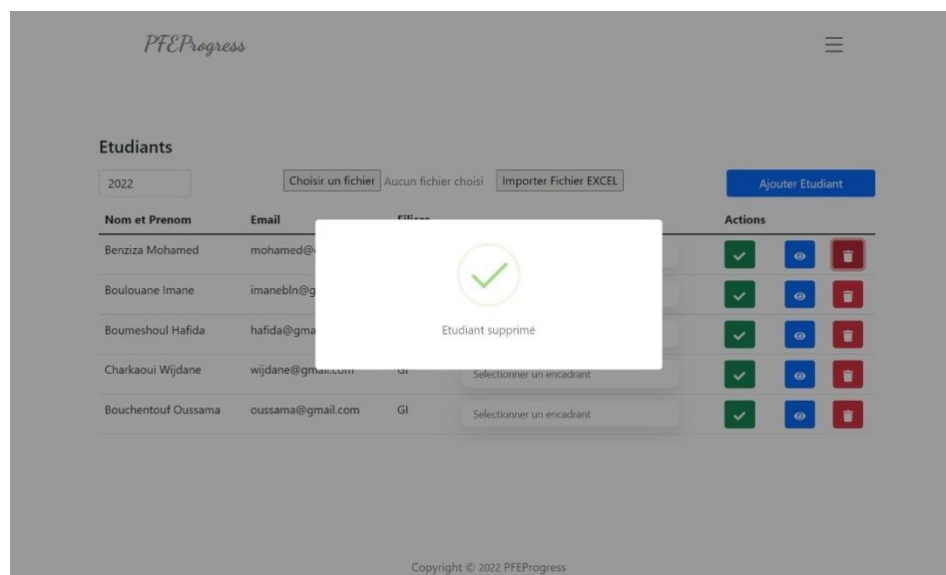


Figure 15: Alerte « étudiant supprimée »

Affecter un encadrant

Pour affecter un encadrant académique à un étudiant, il suffit de choisir un encadrant depuis le **Dropdown** ensuite cliquer sur le bouton vert pour valider l'affectation.



Figure 16: Affecter encadrant

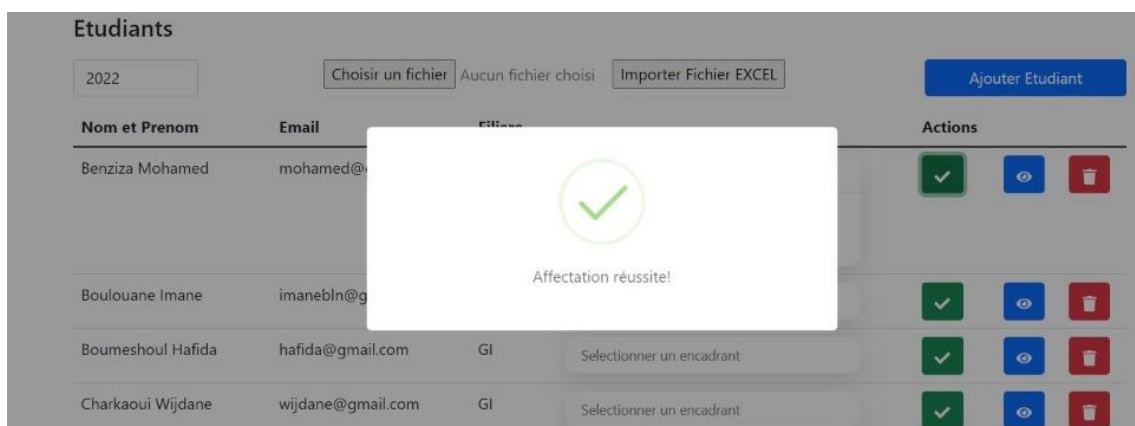


Figure 17: Alerte « valider l'affectation d'un encadrant »

L'interface Ajouter Etudiant

L'administrateur peut ajouter les étudiants manuellement via le bouton **Ajouter Etudiant**, Il suffit de remplir le formulaire avec les coordonnées de l'étudiant.

PFEProgress ☰

Ajouter un Etudiant

Username

Nom

Prenom

Email

Email Encadrant

Sujet

Nom de societe

Ville

technologie utilise

Annee

Copyright © 2022 PFEProgress

Figure 18: Interface Ajouter étudiant

Pour simplifier l'ajout des étudiants, notamment lorsque le nombre est important, L'interface des étudiants contient un bouton pour importer un fichier EXCEL contenant les informations des étudiants et de leurs PFE.

NB : Il est obligatoire d'utiliser les mêmes noms de colonnes que l'exemple de fichier ci-dessous.

B	C	D	E	F	G	H	I	J
Nom	Prenom	Email	Sujet	NomSociete	Ville	TechnologiesUtilisees	EmailEncadrant	Filiere
Chahid	Anas	anas@gmail.com	medecine	sqli	Oujda	asp + react	encadrant@sqli.com	GI
Bouchentouf	Oussama	oussama@gmail.com	medecine	sqli	Oujda	react native	encadrant@novelis.com	GI
Benziza	Mohamed	mohamed@gmail.com	MBA	sqli	Oujda	spring + angular	encadrant@capgemini.com	GI
Boulouane	Imane	imanebln@gmail.com	pfe progress	sqli	Oujda	asp + react	encadrant@cgi.com	GI
Boumeshoul	Hafida	hafida@gmail.com	pfe progress	sqli	Oujda	asp + react	encadrant@aevaweb.com	GI
Charkaoui	Wijdane	wijdane@gmail.com	pfe progress	sqli	Oujda	asp + react	encadarn@opensnz.com	GI

Figure 19: Fichier Excel des étudiants et PFE

L'interface Etudiant Détails

Cette interface affiche les informations d'un étudiant.

The screenshot shows the 'Etudiant Details' form in the PFEProgress application. The form is divided into two main sections. The left section contains fields for student information: Nom (CHARKAOUI), Prenom (Wijdane), Email (wijdanecharkaoui@gmail.com), Sujet (Project Name), Les technologies utilisees (ASP.NET), Societe (SQLI), Ville (Oujda), Année (2022), Email Encadrant (ysf.naimi@sqli.com), and Encadrant Academique (KADDARI Zakaria). The right section contains fields for defense details: Selectionner le jury 1, Selectionner le jury 2, Date de Soutenance (18-06-2022), Heure du debut (14:00), and Heure de fin (16:00). A green 'Confirmer' button is located at the bottom right of the form. The footer of the application shows 'Copyright © 2022 PFEProgress'.

Figure 20: Alerte « Etudiant Details »

L'interface **Etudiant Details** permet aussi de gérer la soutenance de chaque étudiant.

L'administrateur doit sélectionner les deux jurys qui vont assister à la soutenance, ainsi que la date et l'heure de début et l'heure de fin.

Gestion des alertes :

- Si l'affectation de soutenance a réussi, l'alerte suivante sera affichée :

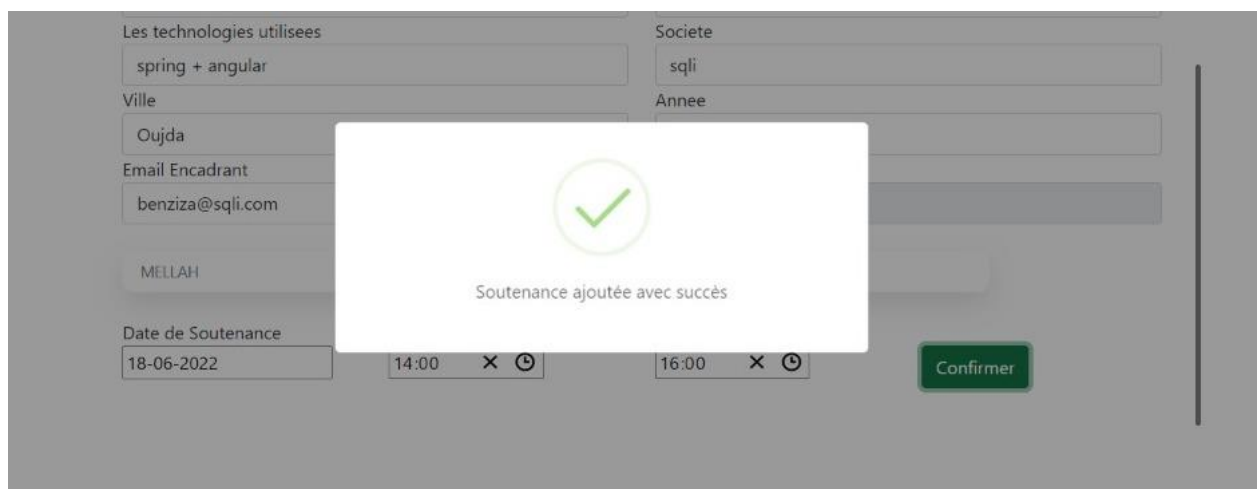


Figure 21: Alerte « soutenance affectée avec succès »

- Si l'étudiant a déjà une soutenance, l'alerte suivante sera affichée :

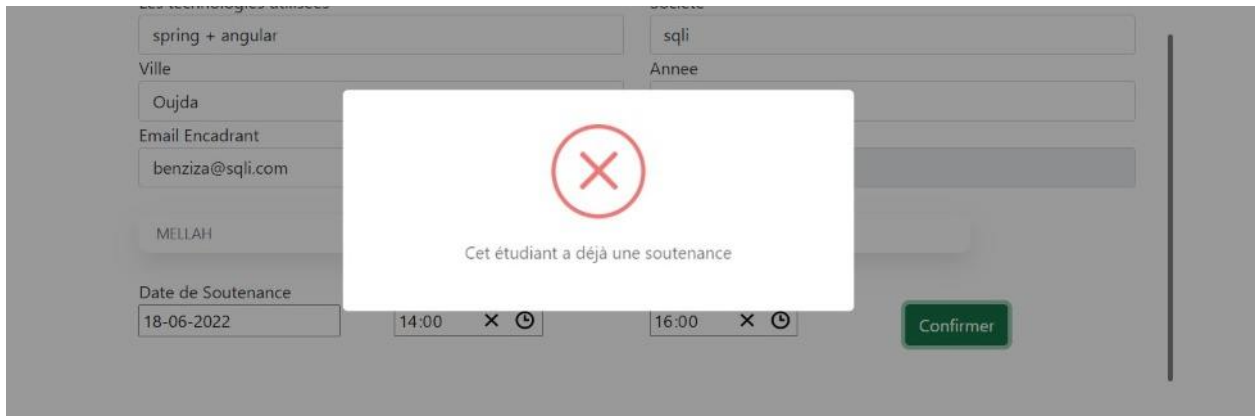


Figure 22: Alerte « étudiant a déjà une soutenance »

- Si l'administrateur a saisi l'encadrant académique en tant que jury, l'alerte suivante sera affichée :

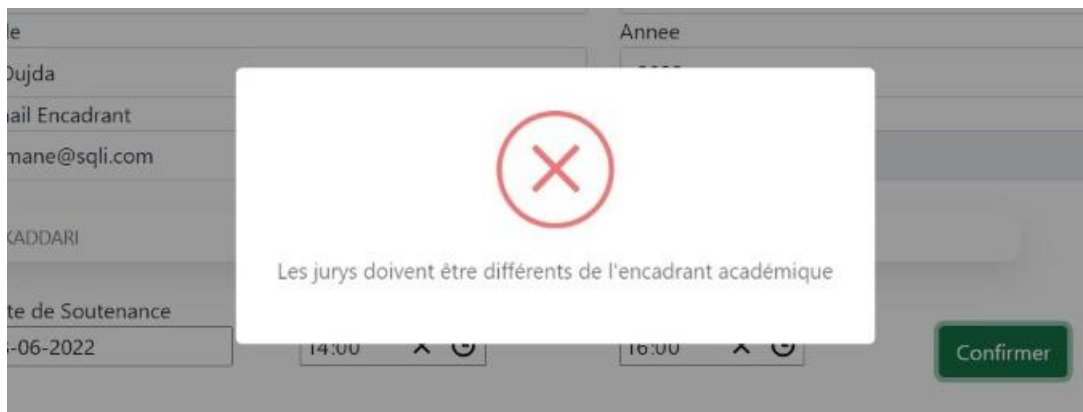


Figure 23: Alerte « Jury doivent être différents de l'encadrant »

- Si l'administrateur a saisi le même encadrant en tant que jury1 et jury2 l'alerte suivante sera affichée :



Figure 24: Alerte « jurys doivent être différents »

- Si l'administrateur a choisit un jury qui n'est pas disponible l'heure ou bien le jour saisit.

Figure 25: Alerte « jury n'est pas disponible »

L'interface des soutenances

Cette interface liste les soutenances qui ont lieu le jour saisit en haut de la page.

L'administrateur peut afficher les détails d'une soutenance ou bien de la supprimer.





Etudiant	Sujet	Date de soutenance	Actions
Bouchentouf Oussama	medecine	18/06/2022	 
CHARKAOUI Wijdane	Project Name	18/06/2022	 

Figure 26: Interface des soutenances

L'interface Soutenance Détails

The screenshot shows a web interface titled 'Soutenance Détails'. At the top left is the 'PFEProgress' logo, and at the top right is a hamburger menu icon. The main content area is titled 'Soutenance Détails' and contains a form with the following fields:

Nom de l'étudiant	Sujet
Bouchentouf Oussama	medecine
Encadrant	
MELLAH Youssef	
Jury	Jury
HAJA Zakaria	BOUGROUN Zineb
Date	Durée
18/06/2022	14:00 -> 16:00

At the bottom of the page, there is a copyright notice: 'Copyright © 2022 PFEProgress'.

Figure 27: Interface Soutenance Détails

Exigences de la suppression des encadrants et des étudiants :

Lorsque l'administrateur supprime un encadrant, il doit affecter de nouveau un encadrant aux étudiants ayant l'encadrant supprimé comme encadrant académique et la soutenance programmée à celui-ci (soit encadrant soit jury) sera supprimée.

Lorsque l'administrateur supprime un étudiant, le PFE et soutenance de cet étudiant seront supprimés.

Le menu de l'application

Ce menu contient les liens vers les interfaces de l'application ainsi que le bouton **logout** pour se déconnecter.



Figure 28: Menu de l'application

Espace Etudiant

Cet espace est dédié pour l'étudiant, il contient un tableau de gestion de projet permettant de répartir les tâches, sous forme de cartes, au sein des colonnes :

- ToDo
- In Progress
- Testing
- Done

Cet Espace permettra aussi, à l'encadrant de poursuivre le projet PFE et d'affecter des taches à l'étudiant.

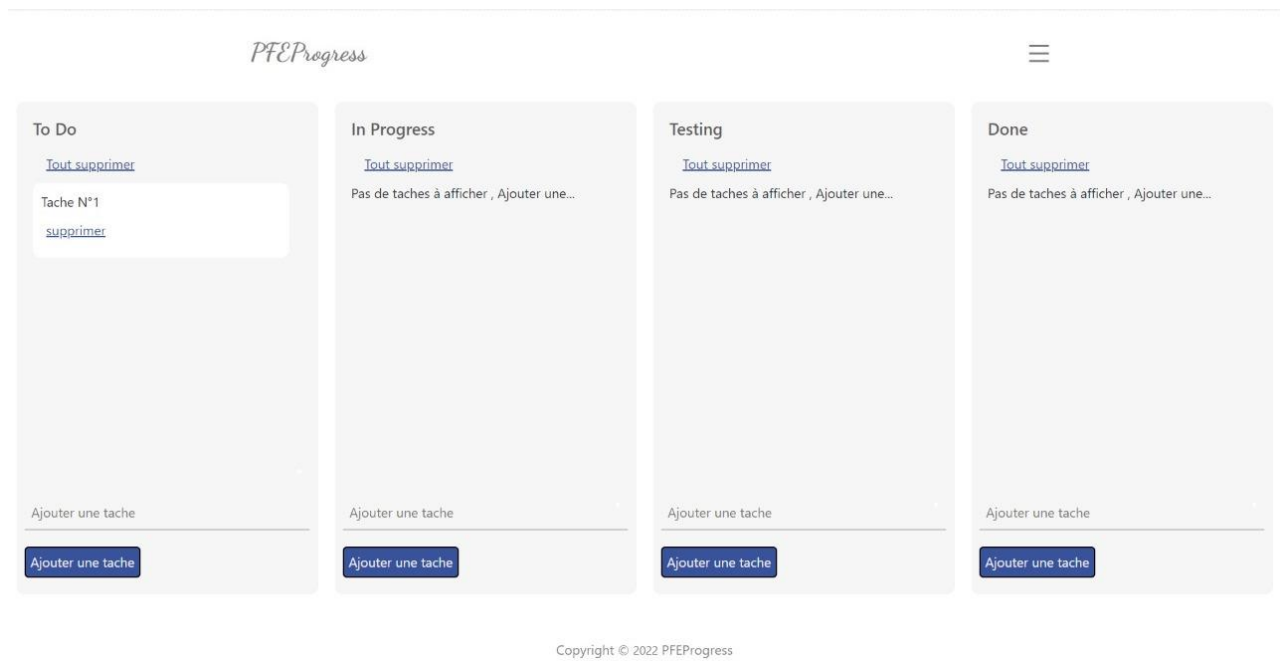


Figure 29: Interface Espace Etudiant

Bibliographie

- <https://stackoverflow.com>
- <https://github.com>
- <https://dotnet.microsoft.com>
- <https://reactjs.org/docs>
- <https://www.codeproject.com/>
- <https://codepen.io/>
- <https://c-sharpcorner.com>

Conclusion et perspectives

Dans ce rapport, nous avons exposé les étapes de conception et de développement de notre application qui consiste à créer une application web pour la gestion des PFE et des soutenances.

Au-delà de l'intérêt et du bénéfice que pourraient avoir ce projet pour le département informatique, sa réalisation a été, pour nous, une véritable expérience pédagogique et de recherche. En effet, il nous a permis de vivre pratiquement les étapes principales de développement d'applications logicielles. Dès le début de ce travail, nous nous sommes confronté au choix d'une méthode de conception qui soit à la fois cohérente et simple d'utilisation.

L'un des points essentiels de cette démarche est d'assurer une cohérence entre les différentes étapes du processus de développement. Arrivé à la fin de la conception, s'est posé, pour nous, le problème de l'implémentation. Là aussi, il a fallu faire des choix et surtout acquérir de nouvelles compétences comme la maîtrise des technologies Dotnet et ReactJs, la programmation orientée objet et l'application des API.

En guise de perspective, certaines améliorations peuvent être apportées au travail afin de prendre en charge de nouvelles fonctionnalités. Nous pouvons citer par exemple, l'amélioration de l'espace étudiant et l'ajout d'un espace personnalisé à l'encadrant.