

Année Universitaire : 2021-2022

Année Universitaire : 2021-2022

Dédicaces

Nous dédions ce modeste travail à nos chers parents en premier lieu, à tout le reste de la famille et à tous nos chers amis, nos collègues de promotion ainsi à nos formateurs à l'institut ESTA, qui ont assisté à notre formation durant ces deux années.

Aussi à tous ceux qui ont contribué de près ou de loin à la réalisation de ce projet.

Remerciements

Avant tout développement sur cette expérience professionnelle, il apparaît opportun de commencer ce rapport par des remerciements, aux formateurs et cadres qui nous ont formés durant notre formation à L'école supérieur de technologie à Agadir.

Qu'il nous soit permis d'exprimer nos gratitude à Monsieur Omar Ait Zemzami ; enseignant à l'école Supérieur de Technologie qui a accepté de nous encadrer et qui nous a dirigé tout au long de cette expérience. Qu'il trouve ici, le témoignage de notre sincère reconnaissance et nos vifs remerciements.

Enfin, nous tenons à remercier toutes les personnes ; qui par leurs conseils, leurs suggestions ou par leur présence nous ont permis de rendre notre travail efficace et bien fini.

Sommaire

Dédicaces	4
Remerciements.....	5
Sommaire	6
Liste des figures.....	8
Introduction générale	10
CHAPITRE I : Cadre général du projet	11
Introduction	12
I. Cahier de charge.....	12
1. Le contexte général.....	12
2. Problématique.....	12
3. Solution proposée	12
4. Objectif.....	13
5. Acteurs	13
6. Etudes des besoins	13
II. Méthodologie suivie	15
1. Logiciel	15
2. Langages de programmation.....	15
Conclusion.....	16
CHAPITRE II : Conception et analyse	17
Introduction	18
I. Conception générale	18
1. Cycle de vie.....	18
2. Méthodologie de conception.....	19
II. Conception détaillée.....	19
1. Diagramme de cas d'utilisation.....	19
2. Diagramme de séquence	21

3. Diagramme d'activité	24
4. Modèle relationnel de la base de données	25
Conclusion	25
CHAPITRE III : Réalisation et mise en œuvre	26
Introduction	27
I. Interfaces et fonctionnalités	27
1. Page d'accueil.....	27
2. Page d'authentification	28
3. Partie Professeur	29
4. Partie Etudiant	35
5. Partie administrateur	41
Conclusion	42
Conclusion générale	43
Webographie	44

Liste des figures

Figure 1 : Modèle de cycle de vie en V.....	18
Figure 2 : Diagramme de cas d'utilisation d'un admin	19
Figure 3 : Diagramme de cas d'utilisation général d'un utilisateur	20
Figure 4 : Diagramme de séquence pour l'authentification	21
Figure 5 : Diagramme de séquence pour la programmation du devoir.....	22
Figure 6: Diagramme de séquence pour le visionnage du cours par un étudiant	23
Figure 7 : Diagramme d'activité pour l'authentification d'un utilisateur	24
Figure 8 : Schéma relationnel de base de données	25
Figure 9 : Page d'accueil.....	27
Figure 10 : Page d'authentification.....	28
Figure 11 : Tableau de bord du professeur en DarkMode.....	29
Figure 12 : Panneau pour ajouter une ressource	29
Figure 13 : Choisir type un cours.....	30
Figure 14 : Dialogue de suppression.....	30
Figure 15 : Alerte de succès lors de suppression du cours.....	31
Figure 16 : Programmer un devoir	31
Figure 17 : Devoir ajouté avec succès.....	32
Figure 18 : Mise à jour d'un devoir	32
Figure 19 : Devoir mis a jour.....	32
Figure 20 : Voir les comptes rendus des étudiants	33
Figure 21 : Choix de type de ressource -> examen.....	33
Figure 22 : Configuration de l'examen.....	34
Figure 23 : Ajout des questions de l'examen	34
Figure 24 : Publier l'examen.....	35
Figure 25 : Tableau de bord Etudiant.....	36
Figure 26 : Contenu d'un élément	36
Figure 27 : Accéder au devoir	37
Figure 28 : Deposer le devoir	37
Figure 29 : Delai expirer	38

Figure 30 : Examen Avant ouverture	38
Figure 31 : Faire une tentative	39
Figure 32 : Passer un examen	39
Figure 33 : Correction de l'examen par le prof	40
Figure 34 : Affichage de la note	40
Figure 35 : Lancement du serveur de développement	41
Figure 36 : Page d'accueil Django	41

Introduction générale

Dans le cadre de notre formation à l'Ecole Supérieure de Technologie d'Agadir, nous sommes amenés à réaliser un projet de fin d'études en travaillant en groupe. Ce dernier a plusieurs objectifs ; valider nos compétences théoriques acquises lors des deux années du DUT ; approfondir nos connaissances dans une situation réelle ; se confronter dans le domaine informatique au milieu du travail

De nos jours, les étudiants ont souvent besoins d'informations ou bien de ressources qui leurs aident à se préparer aux examens ou aux concours, les enseignants aussi trouvent des difficultés au cours de transmission des cours qui sont mal transmises au sein de quelques établissements.

C'est pour cela les plateformes E-Learning sont apparues, comme étant une solution pour l'apprentissage en ligne et pour faciliter la communication entre les enseignants et les étudiants ainsi que l'apprentissage va être plus facile avec moins d'effort et sans déplacement.

Notre projet de fin d'études a pour objectif de mettre en place une plateforme E-Learning mieux que la plateforme actuelle « Moodle ». Une plateforme avancée qui est capable d'assurer l'enseignement en ligne en diffusant les cours, les Travaux Dirigés et Pratiques encadrés par les professeurs, de même de passer un examen en ligne sans tricherie. Ce dernier nous aidé à créer la dynamique de groupe et l'esprit d'un travail collectif et bien sûr à mettre en pratique la formation reçue.

Notre rapport est basé sur trois chapitres principaux. Nous allons commencer par le cadre général de notre projet puis une vue générale a propos de la conception de notre application et finalement le dernier chapitre où nous allons mettre accent sur le travail que nous avons fourni tout au long de ce projet.

CHAPITRE I : CADRE GENERAL DU PROJET

• Introduction

Dans ce chapitre, nous allons mettre notre projet dans son contexte. Pour cela nous avons réalisé un cahier de charges qui détermine le contexte, la problématique du projet et sa solution, l'objectif ainsi les besoins fonctionnels et non fonctionnels de notre plateforme d'enseignement à distance.

I. Cahier de charge

1. Le contexte général

Notre projet de fin d'études se centre sur la création d'une plateforme E-Learning à zéro par codage.

Cette plateforme va permettre aux étudiants et aux enseignants de se communiquer facilement entre eux, tout en facilitant le dépôt des cours et des devoirs par les professeurs ainsi le suivi des cours par les étudiants d'une manière sophistiquée.

Notre plateforme permet la gestion et la diffusion des cours sous format PDF ou vidéos explicatifs pour les étudiants à distance ainsi la possibilité de programmer des devoirs ou des examens séquentiels de tous types de questions. Tout cela d'une manière simple et bien organisée.

2. Problématique

La propagation du CORONA VIRUS a perturbé notre vie et spécialement le mode d'enseignement. Ce dernier a imposé la formation à distance comme une alternative viable lors du confinement totale.

Lors de la pratique, les problèmes de l'enseignement à distance sont demeurés nombreux. Certains sont dus aux problèmes techniques et d'autres sur les méthodes d'enseignement suivi par les profs aussi le manque de respect de quelques étudiants et la mauvaise communication. Cela nous a poussés à chercher des solutions mieux convenables.

Comment approuver un enseignement à distance d'une manière faisable sans tricherie lors ?

3. Solution proposée

Après plusieurs tentatives, il est semblable que les plateformes d'enseignement en ligne sont la solution qui répond au besoin des écoles. Dans notre cas les étudiants des écoles publiques nous cherchons une plateforme gratuite et facile à utiliser ainsi responsive ; fonctionnelle sur tous les appareils.

4. Objectif

L'objectif de ce projet est la création d'une plateforme moderne E-learning qui réponds aux exigences des étudiants/professeurs :

- ❖ L'accès rapide à la plateforme
- ❖ Extension du Dark Mode
- ❖ La facilité d'utilisation
- ❖ Suivi des cours
- ❖ Passer des examens sans tricherie
- ❖ Sécurité
- ❖ Flexibilité

5. Acteurs

- **L'Administrateur** : C'est celui qui a le contrôle total de la gestion de la plateforme.
- **Le Chef de filière** : C'est un super professeur qui a plusieurs fonctionnalités de plus qu'un professeur normal.
- **Le professeur** : C'est un utilisateur qui est responsable de la gestion des cours de son module et le suivi du progrès des étudiants.
- **L'étudiant** : C'est un utilisateur qui a accès aux ressources déposé par son professeur afin de suivre ses cours.

6. Etudes des besoins

a) Besoins fonctionnels

La plateforme doit offrir plusieurs fonctionnalités qui répondent aux besoins des utilisateurs.

Les besoins fonctionnels se présentent en 10 grandes parties :

- **Authentification des utilisateurs** : Les utilisateurs peuvent s'authentifier via leurs 'email et 'mot de passe' après avoir être ajouter par l'admin.
- **Recherche** : Les utilisateurs peuvent chercher dans la plateforme et obtenir des résultats globaux.
- **DarkMode** : Les utilisateurs peuvent alterner entre les modes light & dark.
- **Ajout des utilisateurs** : L'admin possède la permission d'ajouter des utilisateurs comme étant un professeur ou étudiant.

- **L'ajout des ressources :** Le professeur a la possibilité d'ajouter des cours sous différents formats (PDF, jpg, mp4 ...) avec la fonctionnalité drag & drop, ainsi des annonces pour informer les étudiants. Tout cela sous forme de section selon chaque type de ressource.
- **La programmation d'un devoir à rendre :** Le professeur peut programmer des devoirs à rendre en précisant le délai du dépôt.
- **La création d'un examen (test à distance) :** Le professeur peut créer un examen séquentiel/normal de plusieurs types de questions (QCM, questions directes...) ainsi spécifier sa date de début et sa durée.
- **La modification/suppression des ressources :** Le professeur a le droit de modifier ou supprimer un cours/devoir ou bien une annonce en cas d'une erreur.
- **Le suivi des cours :** L'étudiant possède l'accès à toutes les ressources déposées par leur enseignant.
- **Le dépôt des devoirs à rendre :** L'étudiant peut déposer un travail à rendre programmé par son professeur.
- **Passer des examens à distance :** L'étudiant peut passer un examen en ligne avec un système pour éviter la tricherie.

b) Besoins non fonctionnels

Un besoin non fonctionnel est une restriction ou une contrainte qui pèse sur un service du système, telles que les contraintes liées à l'environnement et à l'implémentation et les exigences en matière de performances, les dépendances du projet, la facilité de maintenance, d'extensibilité et de fiabilité.

Notre projet doit répondre aux besoins suivants :

- Fiabilité
- Accès rapide
- Interface ergonomique
- Rentabilité
- Facilité
- Plateforme portable et responsive
- Confidentialité des données

II. Méthodologie suivie

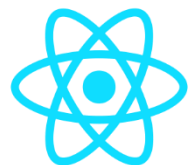
1. Logiciel

- **Sublime Text** : Editeur de texte générique code en C++ et python disponible sur les systèmes d'exploitation Windows, Linux et Mac.
- **LucidChart** : plateforme basée sur le web qui permet aux utilisateurs de collaborer sur le dessin, la révision et le partage de graphiques et de diagrammes.
- **Adobe Photoshop** : Logiciel de retouche, de traitement et de dessin assisté par ordinateur.
- **GitHub** : GitHub est une plate-forme d'hébergement de code pour le contrôle de version et la collaboration. Il vous permet, à vous et à d'autres, de travailler ensemble sur des projets où que vous soyez.



2. Langages de programmation

- **JavaScript** : Un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web
- **React** : Une bibliothèque JavaScript libre développée par Facebook depuis 2013. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendants d'un état et générant une page HTML à chaque changement d'état.
- **Python** : Un langage de programmation interprété, qui favorise la programmation impérative structurée ainsi fonctionnelle et orientée objet.



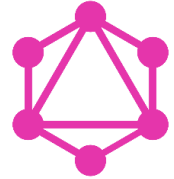
- **Django** : Un cadre de développement web open source en Python. Il a pour but de rendre le développement web 2.0 simple et rapide.



- **Graphene** : Une bibliothèque qui fournit des outils pour implémenter une API GraphQL en python.



- **GraphQL** : Un langage de requêtes et un environnement d'exécution, créé par Facebook en 2012. Inscrit dans le modèle Client-Serveur, il propose une alternative aux API REST.



- **Tailwind CSS** : un Framework CSS utilitaire pour créer des interfaces utilisateur personnalisées.



- **Conclusion**

Dans ce chapitre nous avons présenté le projet en général, ainsi que les fonctionnalités, les tâches que nous allons réaliser, la présentation des outils et langage que nous avons utilisé dans la réalisation de notre plateforme.

CHAPITRE II : CONCEPTION ET ANALYSE

• Introduction

Dans le cycle de vie dans notre projet, la conception représente une partie déterminante pour produire une application de haute qualité.

Cette partie expose la solution conceptuelle qui a pour but de rendre flexible la tâche de la gestion et une projection de la réalité physique. En premier lieu nous allons clarifier la vue globale, en décrivant l'architecture générale que nous allons suivre dans la partie réalisation de notre projet. Puis dans un deuxième lieu nous allons détailler notre choix conceptuel à travers plusieurs types de diagrammes.

I. Conception générale

1. Cycle de vie

Le cycle de vie d'une application comprend toutes les étapes depuis sa conception et sa réalisation jusqu'à sa mise en œuvre.

Nous avons opté pour le modèle de cycle de vie en V, afin de concevoir et développer notre application. Ce choix revient au fait que ce cycle est le plus efficace avec son principe de travail qui nécessite la vérification de chaque étape et la possibilité de corriger les fautes avant de se lancer vers l'étape suivante.

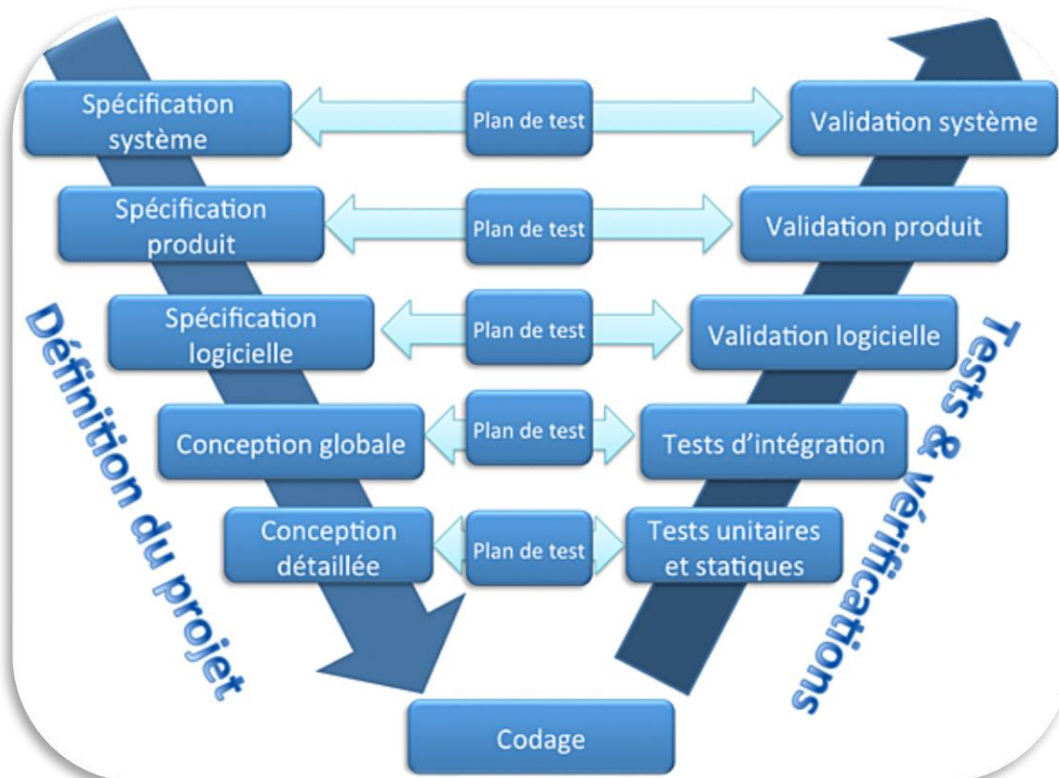


Figure 1 : Modèle de cycle de vie en V

2. Méthodologie de conception

Afin de faciliter notre tâche nous avons utilisé la notation UML (Unified Modeling Language) qui est un standard très largement utilisé dans l'analyse et la conception. Il est devenu une référence en termes de modélisation objet, à un tel point que sa connaissance devienne indispensable pour un développeur.

II. Conception détaillée

1. Diagramme de cas d'utilisation

Le rôle des diagrammes de cas d'utilisation est de recueillir, d'analyser et d'organiser les besoins, ainsi que de recenser les grandes fonctionnalités d'un système. Il s'agit donc de la première étape UML pour la conception d'un système.

a) Pour l'administrateur

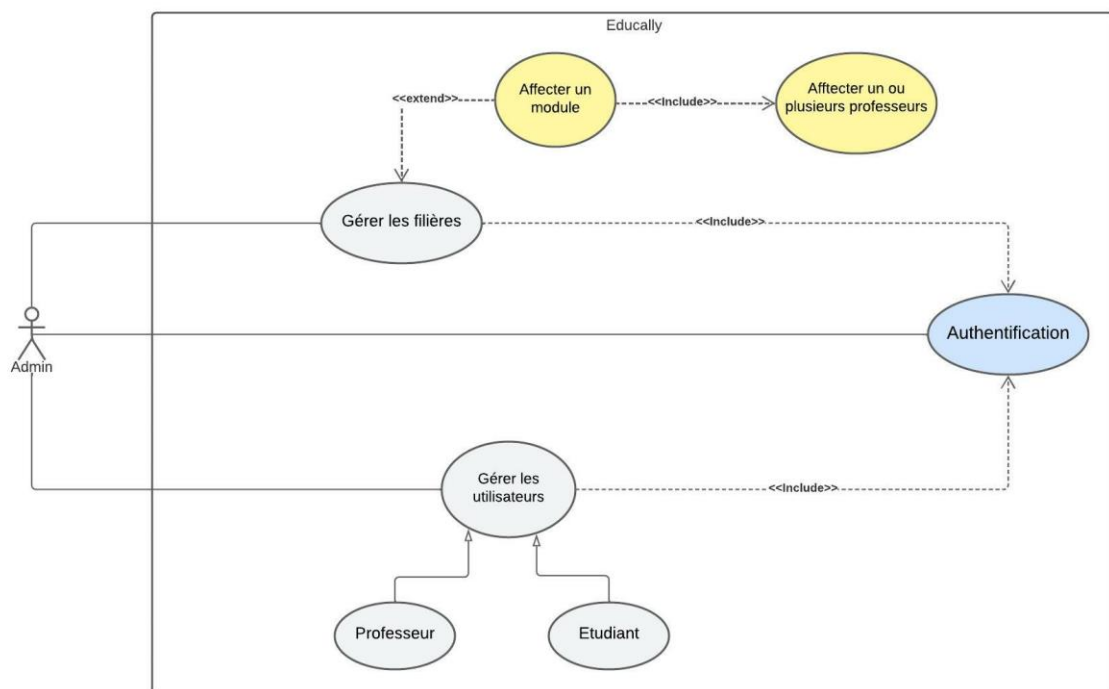


Figure 2 : Diagramme de cas d'utilisation d'un admin

Ce diagramme de cas d'utilisation, représente les permissions de l'administrateur, il s'occupe principalement de la partie technique de la plateforme. Dans notre cas cet acteur a la possibilité d'ajouter les filières, ainsi d'ajouter, de modifier ou de supprimer un étudiant/enseignant selon sa filière, également il peut gérer les cours mises à la plateforme.

b) Pour l'utilisateur

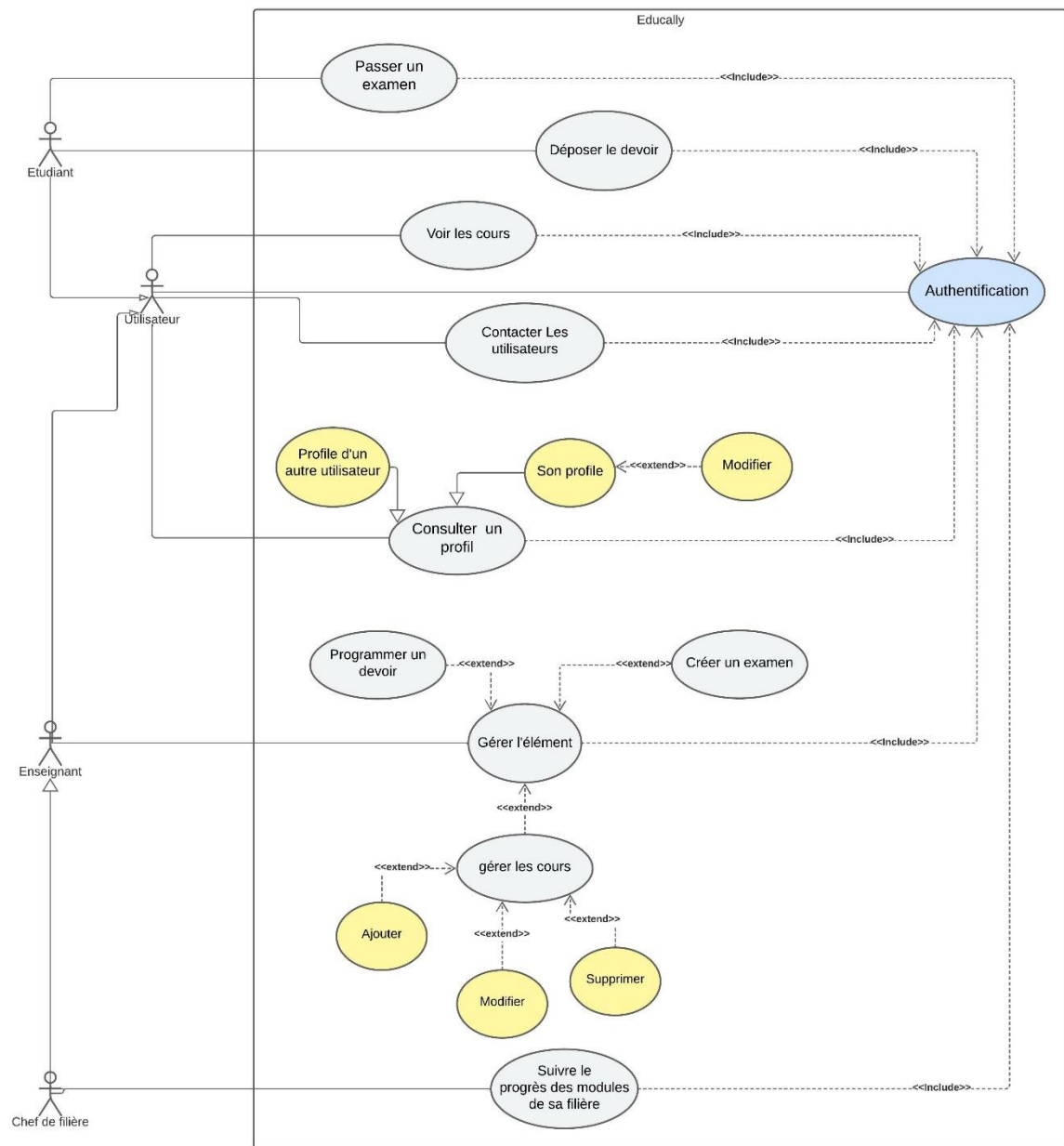


Figure 3 : Diagramme de cas d'utilisation général d'un utilisateur

Ce diagramme représente les différents cas d'utilisation des différents acteurs de notre plateforme :

- **Enseignant :** Lors de l'authentification, il a le droit de gérer son élément qui fait partie d'un module en ajoutant des cours ou bien en programmant des devoirs à rendre ou des examens. En cas d'erreur il peut modifier ou supprimer ses cours.

- **Un chef de filière** : C'est un super enseignant il hérite toutes les permissions de ce dernier ainsi il suit ou surveille le progrès des modules de sa filière.
- **Etudiant** : Après avoir être connecter, l'étudiant aura l'accès aux cours, aux annonces et tous ce qui est mis par leurs enseignants, aussi pour consulter son profil ou celui d'autres utilisateurs. Lors d'une activité programmer par le prof l'étudiant peut déposer son devoir ou passer un examen. Ainsi il peut contacter l'un des profs ou des collègues.

2. Diagramme de séquence

Un diagramme de séquence est un diagramme d'interaction qui expose en détail la façon dont les opérations sont effectuées : quels messages sont envoyés et quand ils le sont, entre les acteurs selon un ordre chronologique.

a) Pour l'authentification

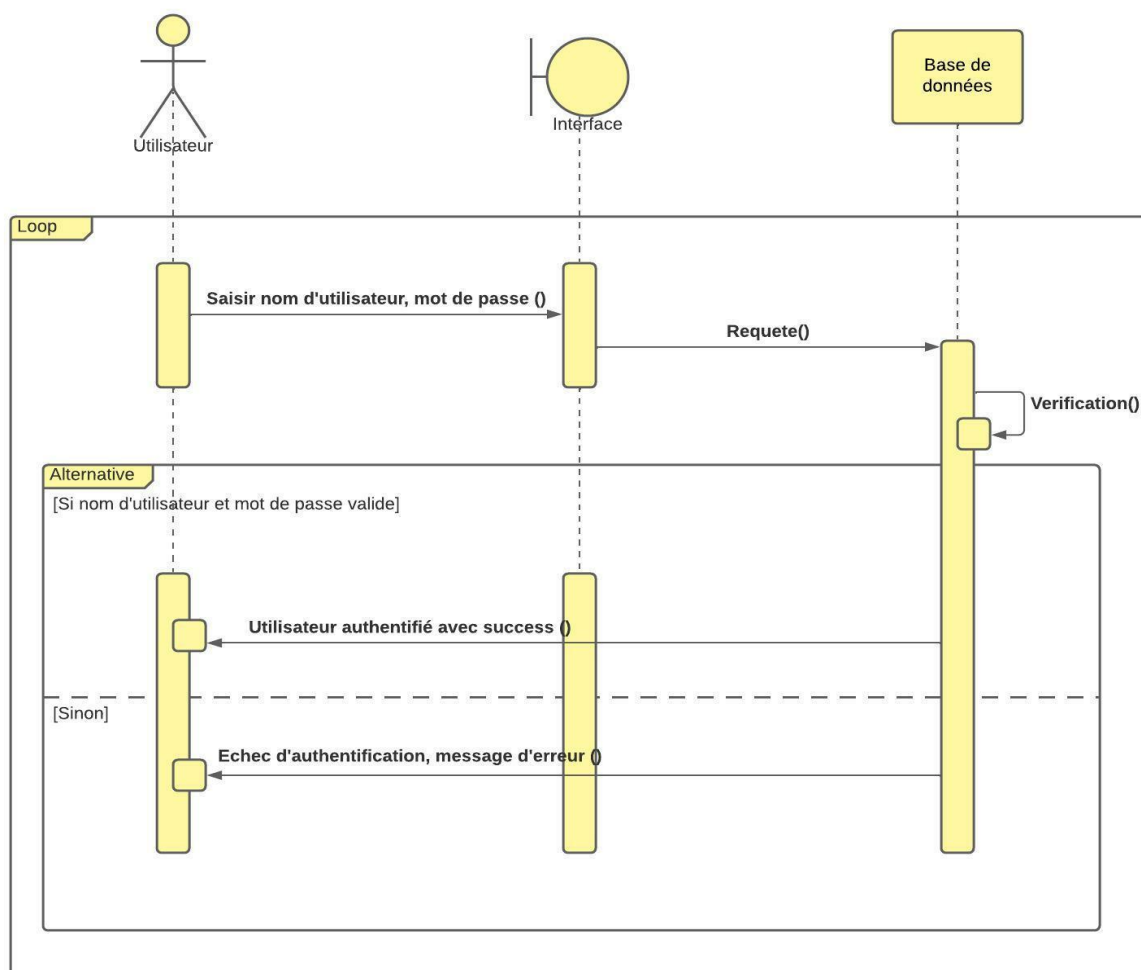


Figure 4 : Diagramme de séquence pour l'authentification

Ce diagramme de séquence représente le processus de l'authentification d'un utilisateur tout en commençant par la saisie du son email et son mot de passe dans l'interface « Login », en passant par la vérification de la validité des données dans la base de données. Finissant soit par l'accès à la plateforme en cas de succès ou bien l'affichage d'une alerte d'erreur lors d'échec de validation.

b) Pour la programmation d'un devoir

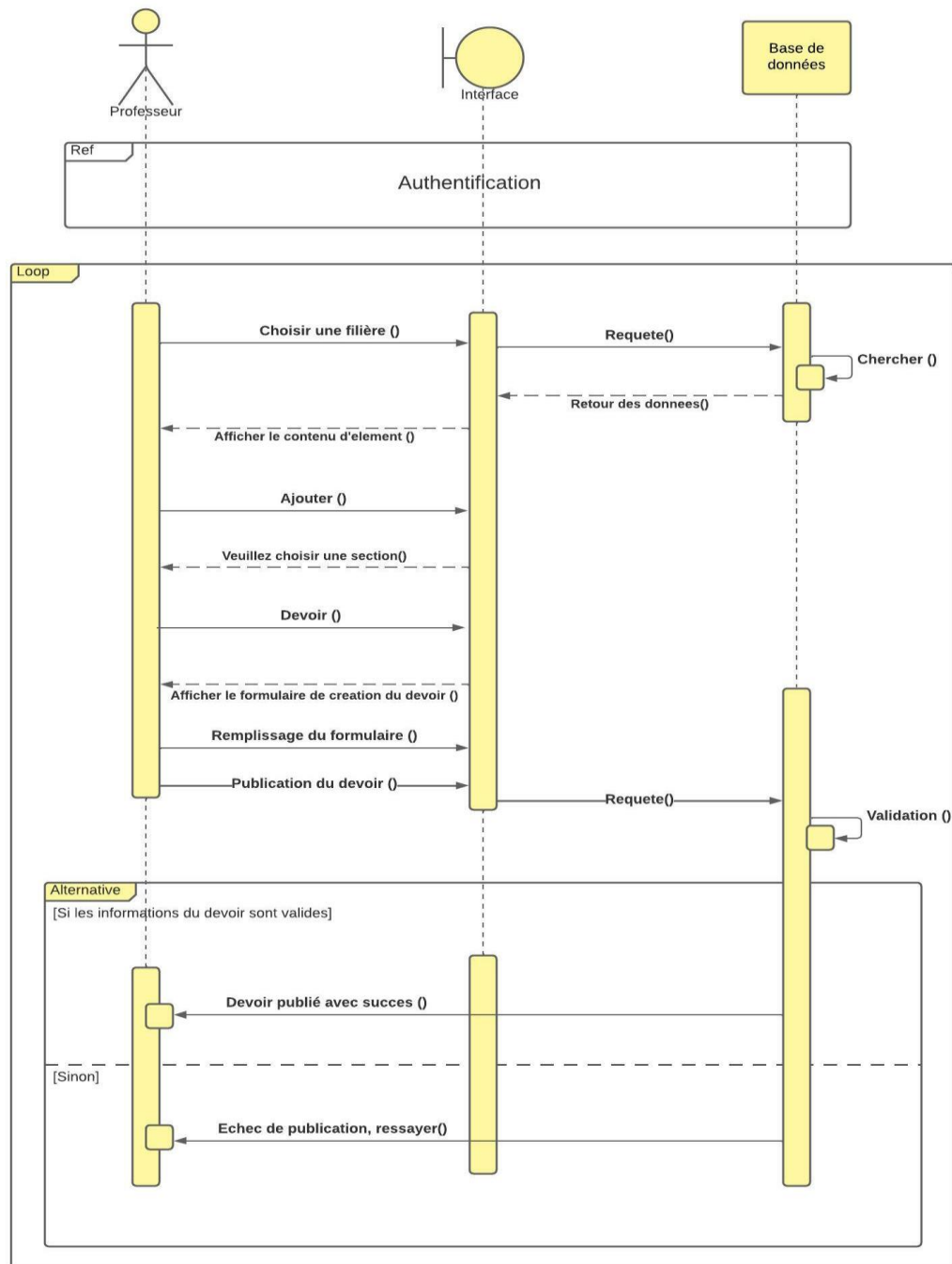


Figure 5 : Diagramme de séquence pour la programmation du devoir

Ce diagramme de séquence représente les étapes qu'un prof suit afin de programmer un devoir à rendre. Commenant par l'authentification du professeur, puis la définition du choix de filière concernée en communiquant avec la base de données à travers plusieurs étapes d'interaction avec l'interface. Finissant soit par la publication du devoir en cas de succès ou bien l'affichage d'une alerte d'erreur lors d'échec de validation des informations du devoir.

c) Pour le visionnage des cours

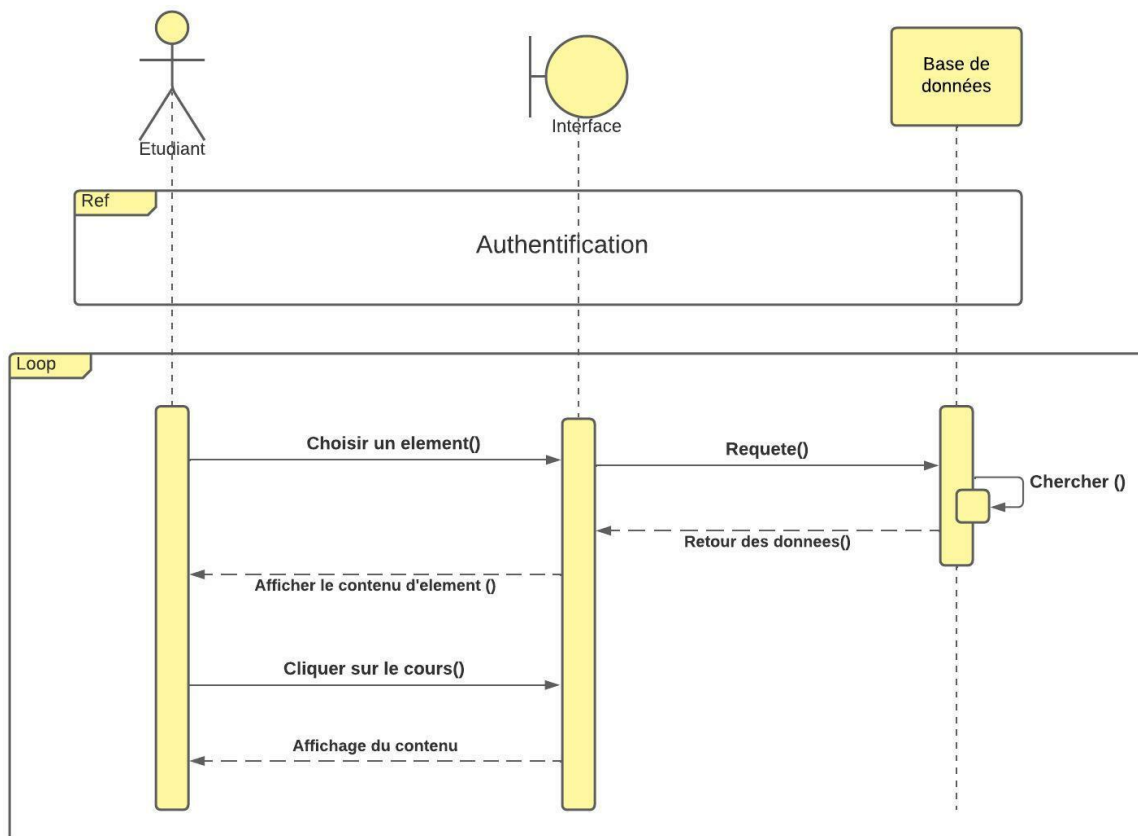


Figure 6: Diagramme de séquence pour le visionnage du cours par un étudiant

Ce diagramme de séquence représente les étapes qu'un étudiant procède afin de voir un cours. Commenant par son authentification, puis la définition du choix de l'élément qu'il veut voir.

3. Diagramme d'activité

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallèles.

a) Pour l'authentification

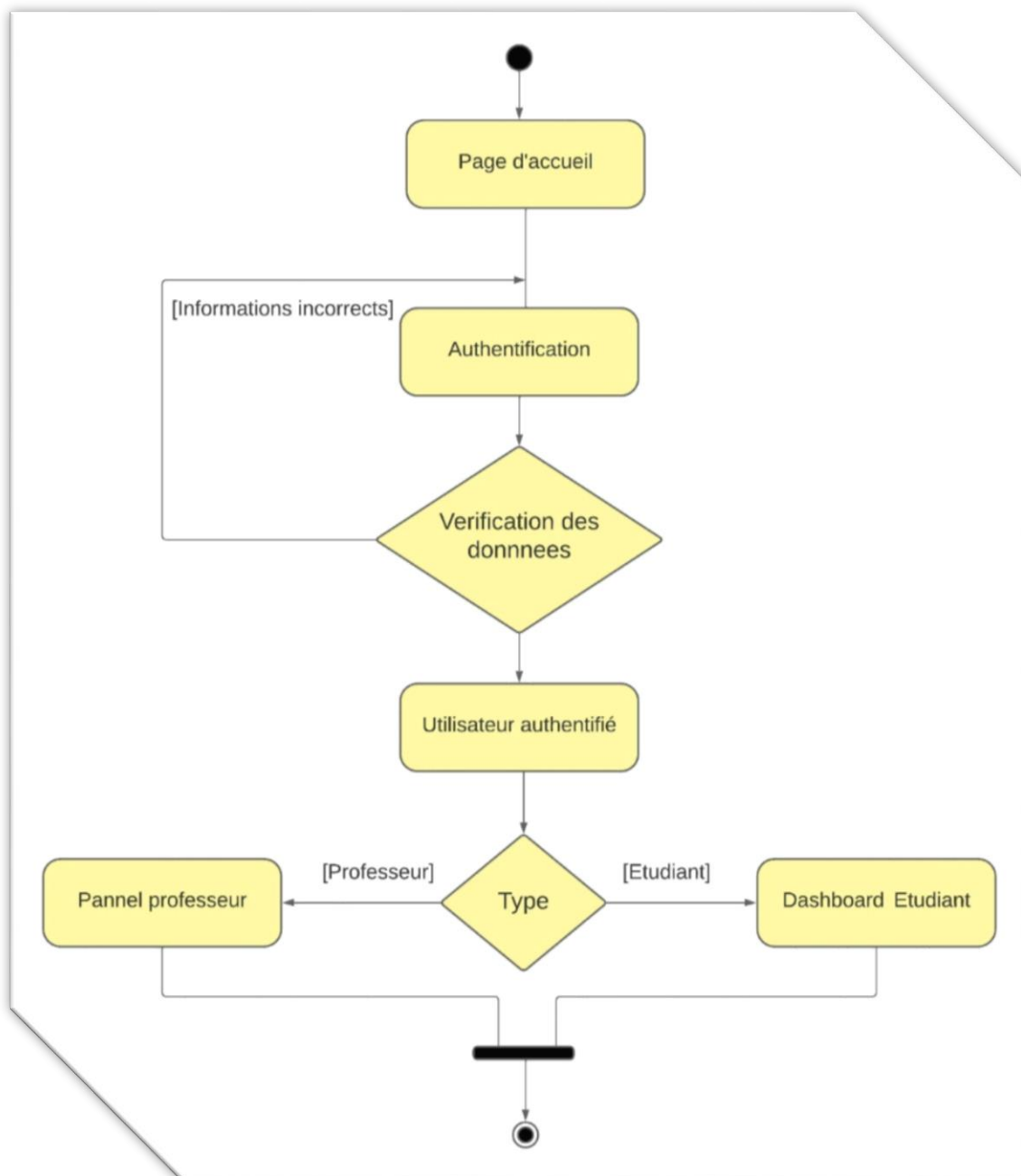


Figure 7 : Diagramme d'activité pour l'authentification d'un utilisateur

Après avoir être connecter l'utilisateur sera redirigé vers une interface selon son rôle.

4. Modèle relationnel de la base de données

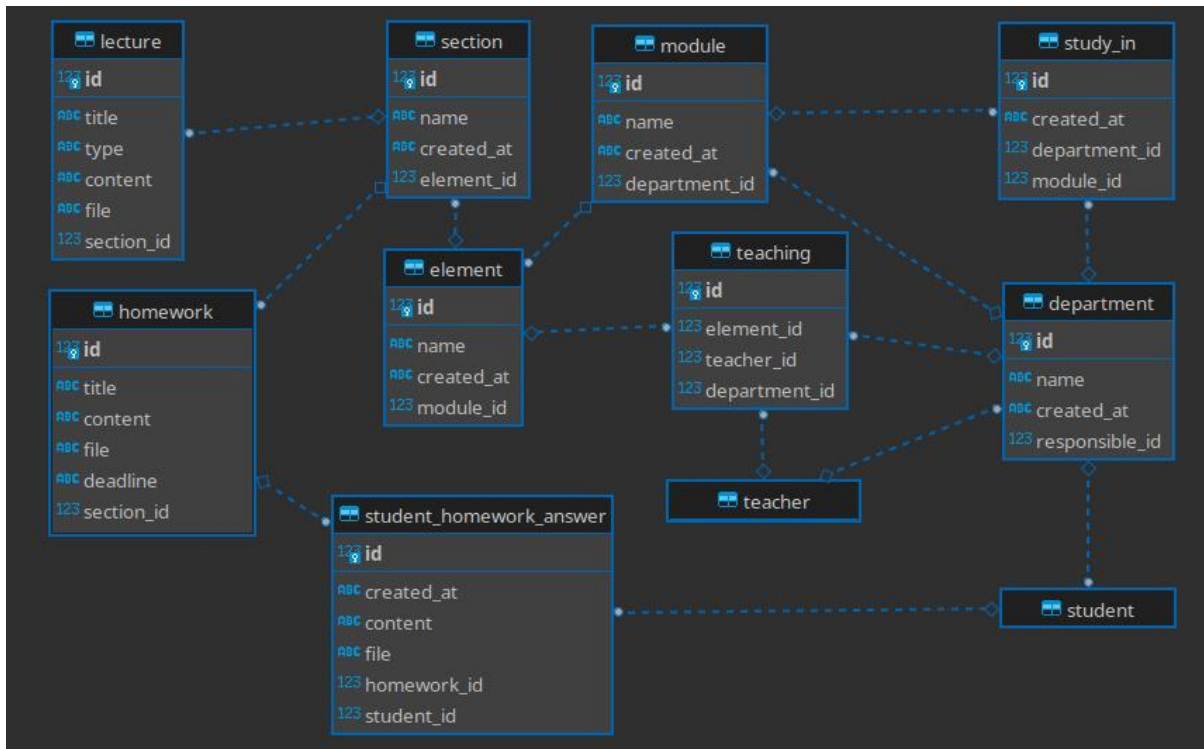


Figure 8 : Schéma relationnel de base de données

- **Conclusion**

La phase conceptuelle est une étape fondamentale pour la réalisation de n'importe quel projet, elle permet de faciliter le système d'information et réaliser l'implémentation de la base de données et le traitement. Cette partie a donné une vision sur notre travail et l'aspect conceptuel de la plateforme à travers les différents diagrammes décrits en UML.

CHAPITRE III : RÉALISATION ET MISE EN ŒUVRE

Introduction

Ce chapitre a pour objectif majeur de présenter le produit final. C'est la phase de réalisation de notre plateforme « Educally ». Dans cette partie, on va présenter quelques interfaces web dans le but de vous situer dans le projet.

I. Interfaces et fonctionnalités

1. Page d'accueil

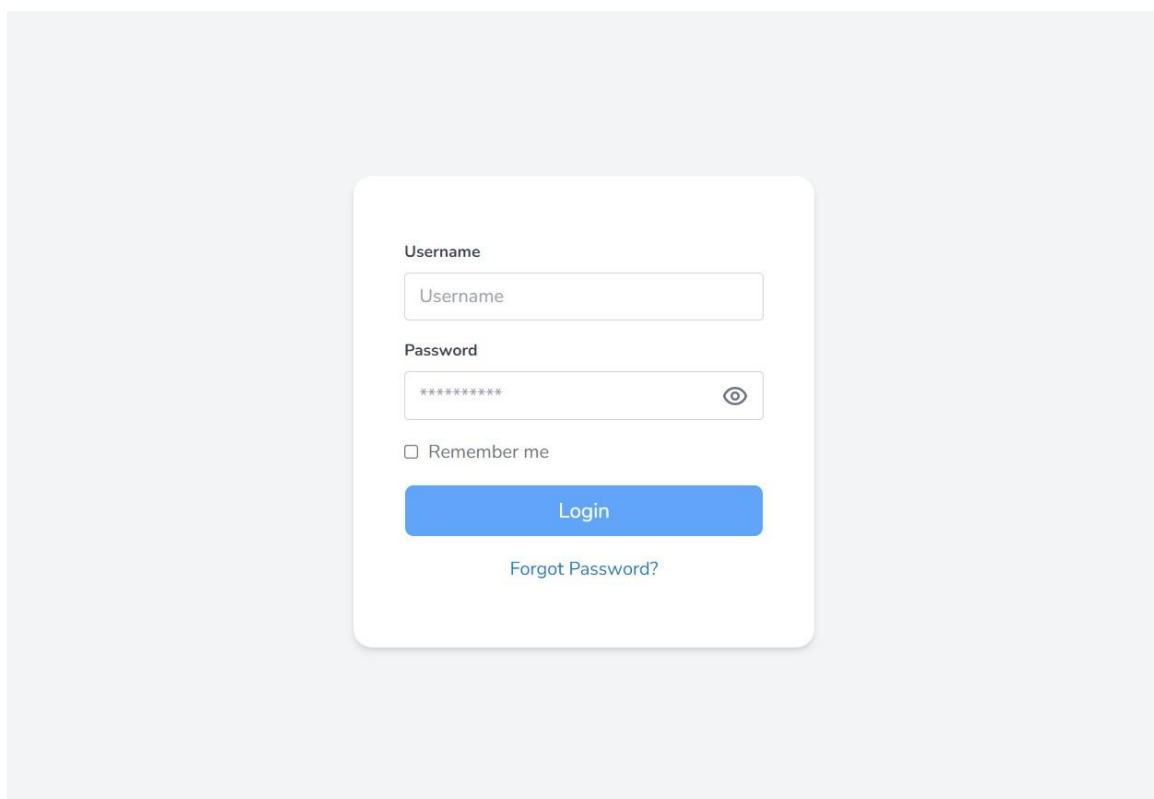


Figure 9 : Page d'accueil

2. Page d'authentification

C'est la page d'authentification qui s'affiche après la page d'accueil en cliquant sur l'icône qui réfère à l'authentification l'entrée à notre plateforme, qui représente la page de connexion, composée de :

- Un champ de texte pour l'Email.
- Un champ de texte pour le mot de passe.
- Vérifiez les informations ci-dessus.
- Sans oublier la case de **remember me**, aussi le cas d'oubli du mot de passe.

Lors de l'authentification chaque utilisateur va se rediriger vers son tableau de bord.

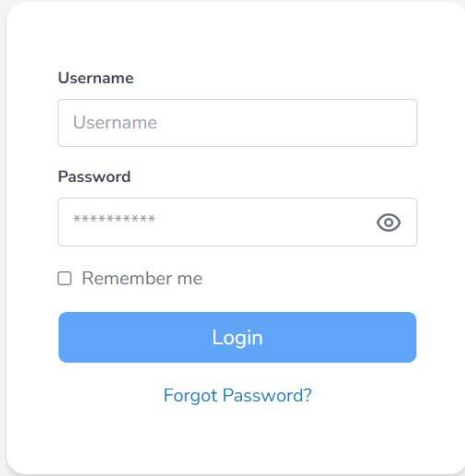
The image shows a login form centered on a light gray background. The form is a white rounded rectangle. It contains a 'Username' label above a text input field with 'Username' as a placeholder. Below that is a 'Password' label above a text input field with '*****' as a placeholder and an eye icon for toggling visibility. There is a checkbox labeled 'Remember me'. At the bottom of the form is a blue 'Login' button and a 'Forgot Password?' link in blue text.

Figure 10 : Page d'authentification

3. Partie Professeur

Une fois l'enseignant est connecté, son tableau de bord s'affiche

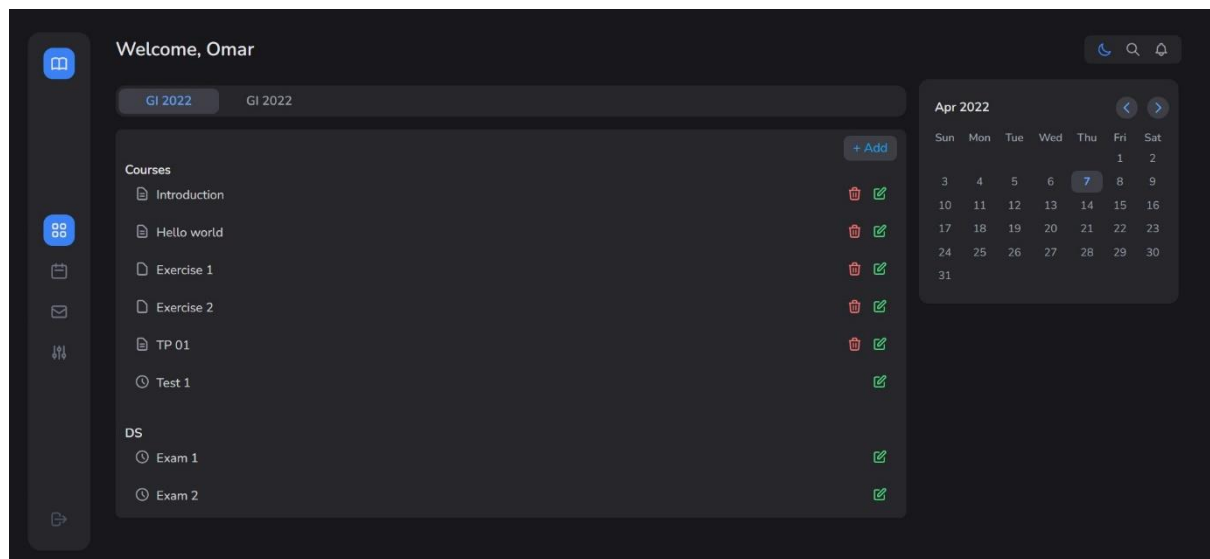


Figure 11 : Tableau de bord du professeur en DarkMode

Le professeur reçoit une vue générale à propos du contenu de son élément dans chaque filière. Aussi les différentes fonctionnalités sont visibles dans la barre de navigation tel que le centre de messagerie, les paramètres et sans oublier le DarkMode et les boutons de gestion (Modification et suppression) dans chaque section.

Lorsque vous appuyez sur le bouton d'ajout, un panneau apparaît composé de :

- ✓ Courses
- ✓ DS

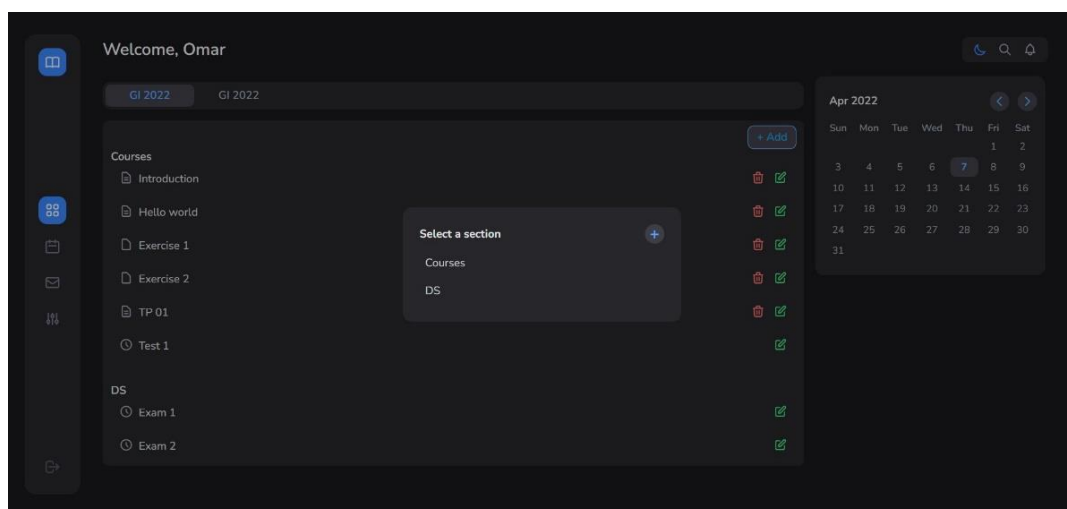


Figure 12 : Panneau pour ajouter une ressource

Les fonctionnalités du professeur :

Le professeur a un certain nombre de permissions, tels que :

✓ Ajouter/modifier/supprimer des cours.

En cliquant sur le bouton « Add » et en choisissant la section des cours le professeur peut ajouter un cours en spécifiant son type (Text, fichier), son titre, une description optionnelle et le contenu soit en cliquant pour uploader un fichier ou bien avec la méthode drag un drop qu'on a implémenté dans notre projet. Ainsi il peut modifier ce cours ou le supprimer en cliquant sur l'une des icônes.

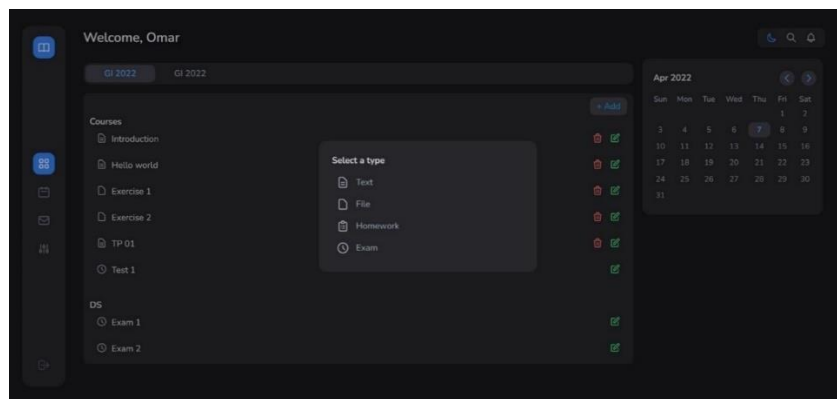


Figure 13 : Choisir type un cours

En cliquant sur l'icône qui réfère à la suppression, Le dialogue de suppression s'affiche comme suivant.

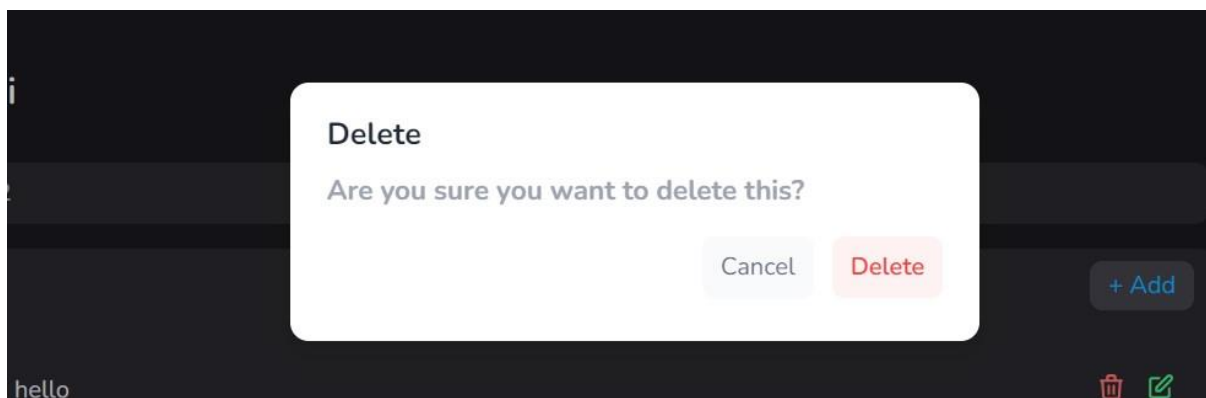


Figure 14 : Dialogue de suppression

Lors de la suppression le système affiche un message de succès à droite en bas

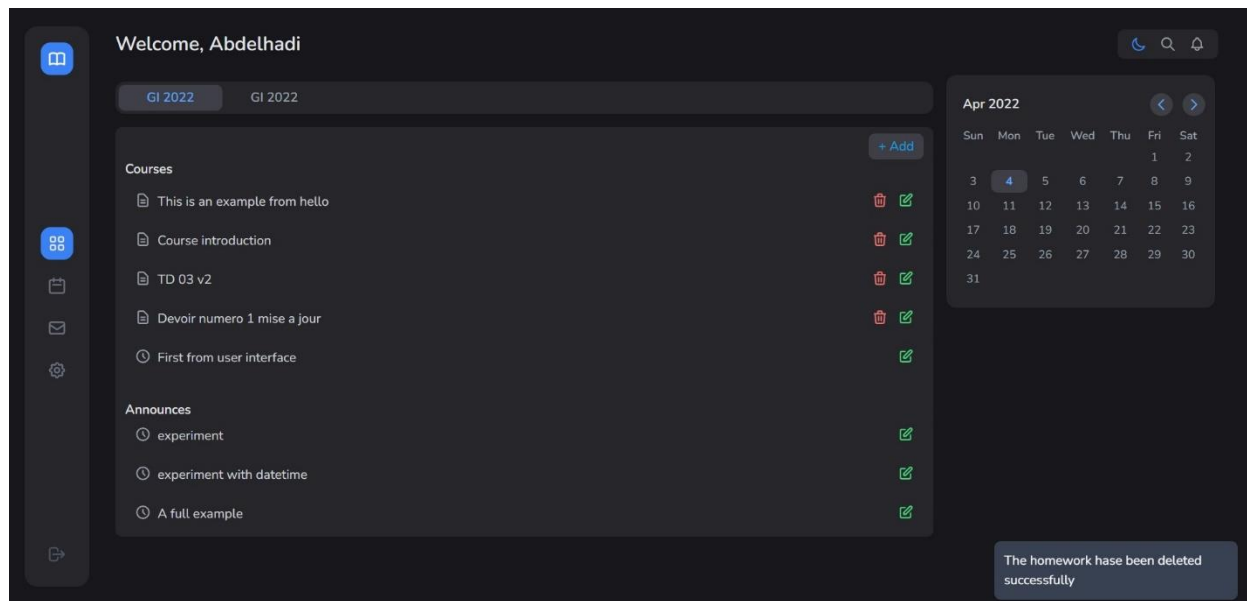


Figure 15 : Alerte de succès lors de suppression du cours

✓ **Ajouter/modifier des devoir à rendre.**

Le professeur peut programmer un devoir à rendre avant un délai précis en suivant la même méthode pour ajouter un cours, avec la précision du dernier délai. Il peut aussi le mettre à jour et voir les réponses des étudiants.

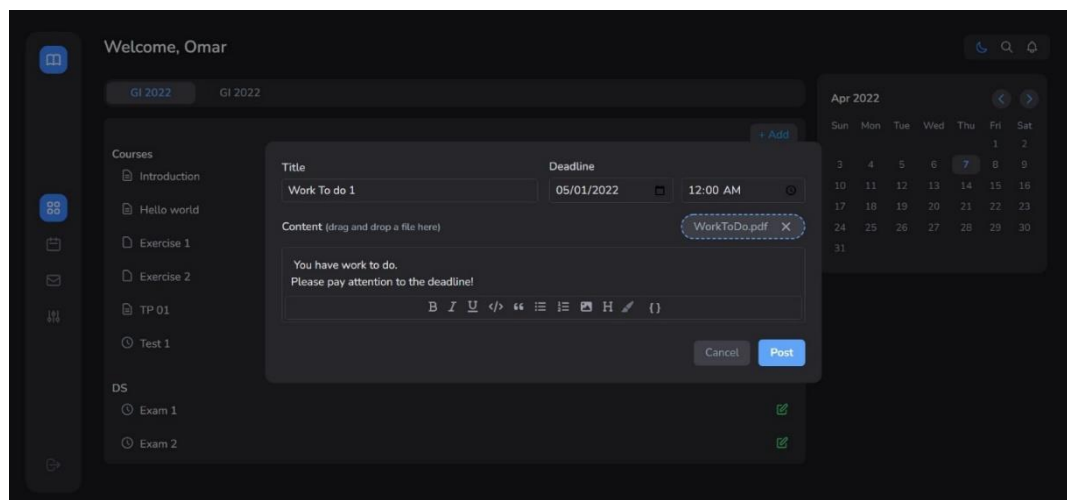


Figure 16 : Programmer un devoir

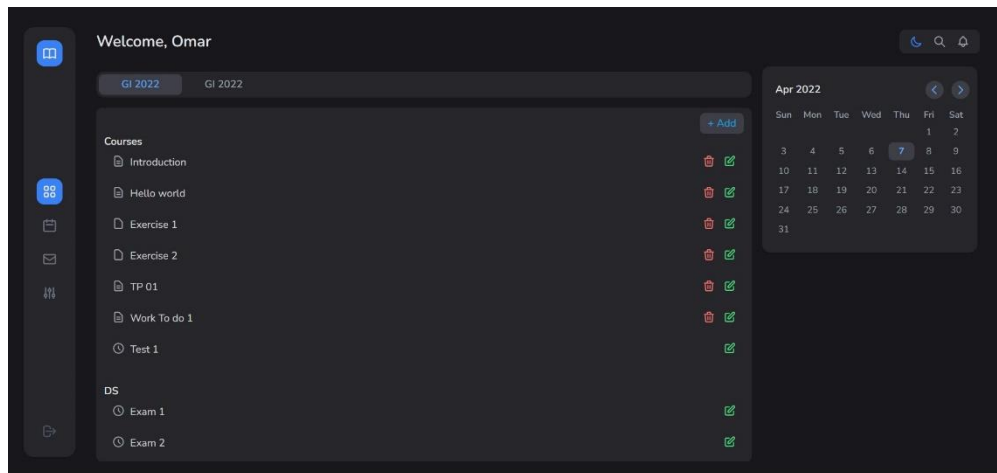


Figure 17 : Devoir ajouté avec succès

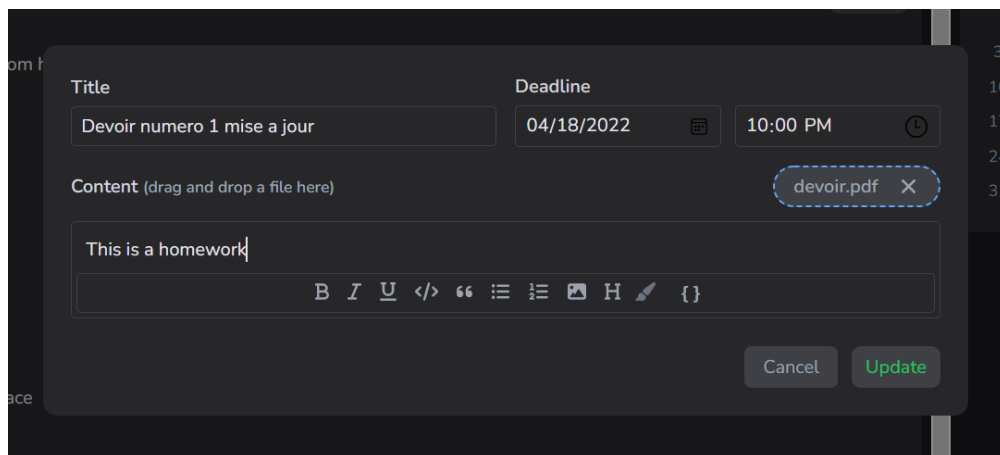


Figure 18 : Mise à jour d'un devoir

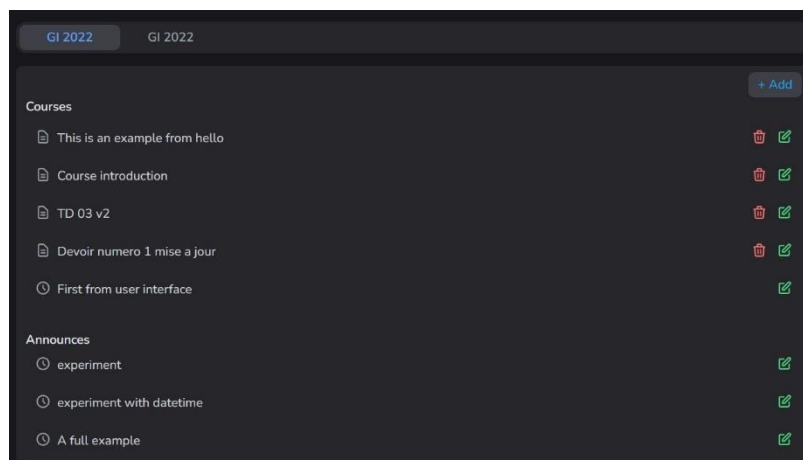


Figure 19 : Devoir mis a jour

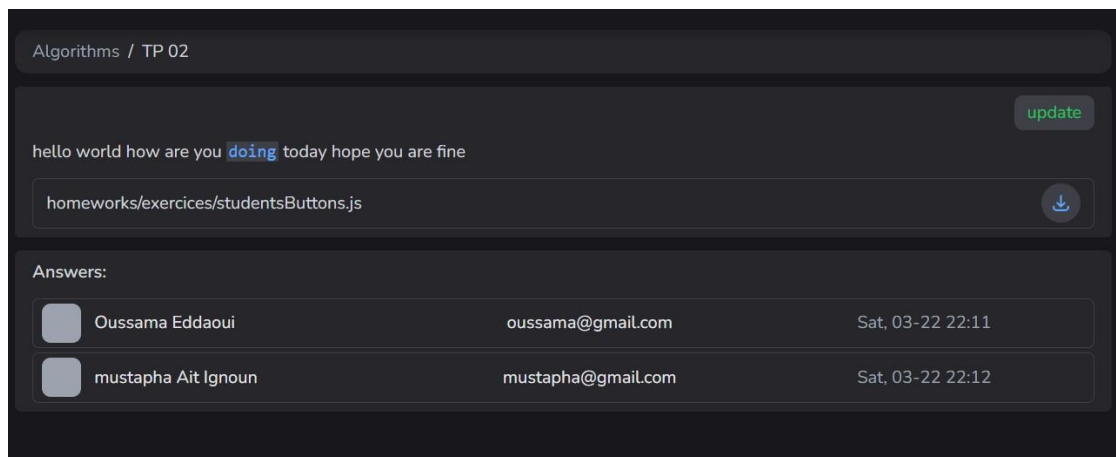


Figure 20 : Voir les comptes rendus des étudiants

✓ Ajouter/modifier des examens.

C'est la fonctionnalité la plus importante dans notre projet, d'où le professeur peut modifier/créer un examen séquentiel/normal d'une durée limitée en précisant le nombre de tentatives. Cet examen peut contenir plusieurs questions de différents types (QCM, QCU, Question directe).

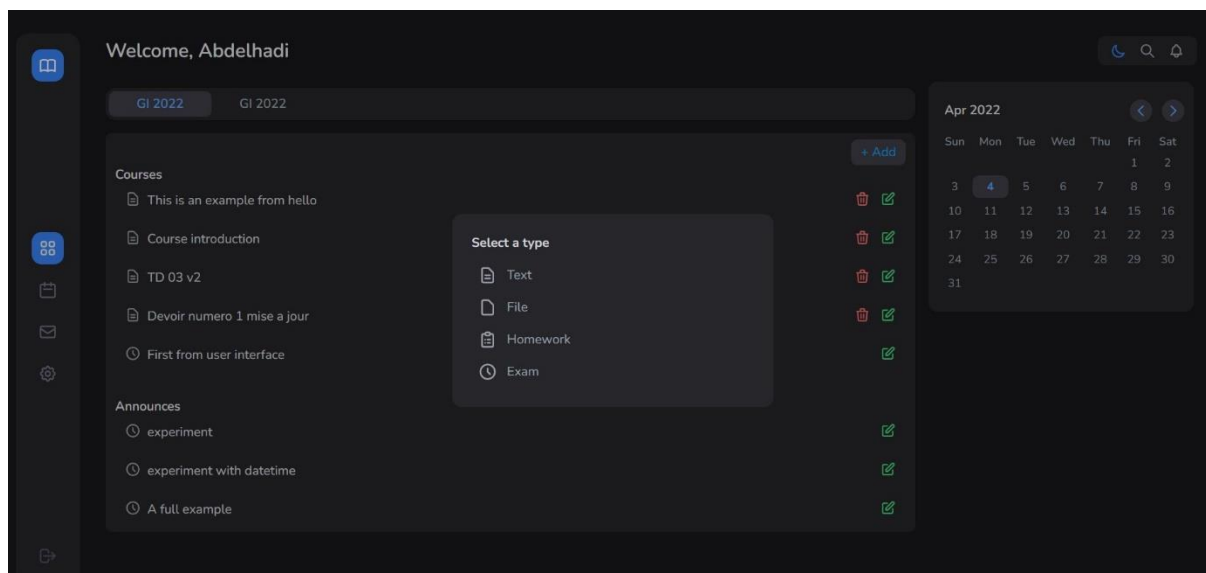


Figure 21 : Choix de type de ressource -> examen

Welcome, Oussama

Starts at: 10/05/2022 02:30 PM

Duration: 01:00

Attempts: 1

Sequentiel: Sequentiel

Title: Exam Simulation

Description: This is report exam

Post

Apr 2022

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Figure 22 : Configuration de l'examen

Welcome, Oussama

Type: Single choice

Mark: 5

Question: What is Ram?

Choices:

- Random access memory
- Right amie House
- Others

Enter a choice

Publish

Q-1 +

Figure 23 : Ajout des questions de l'examen

Après l'insertion de toutes les questions avec leur barème, le professeur doit publier l'examen tout en cliquant sur le bouton « publish ».

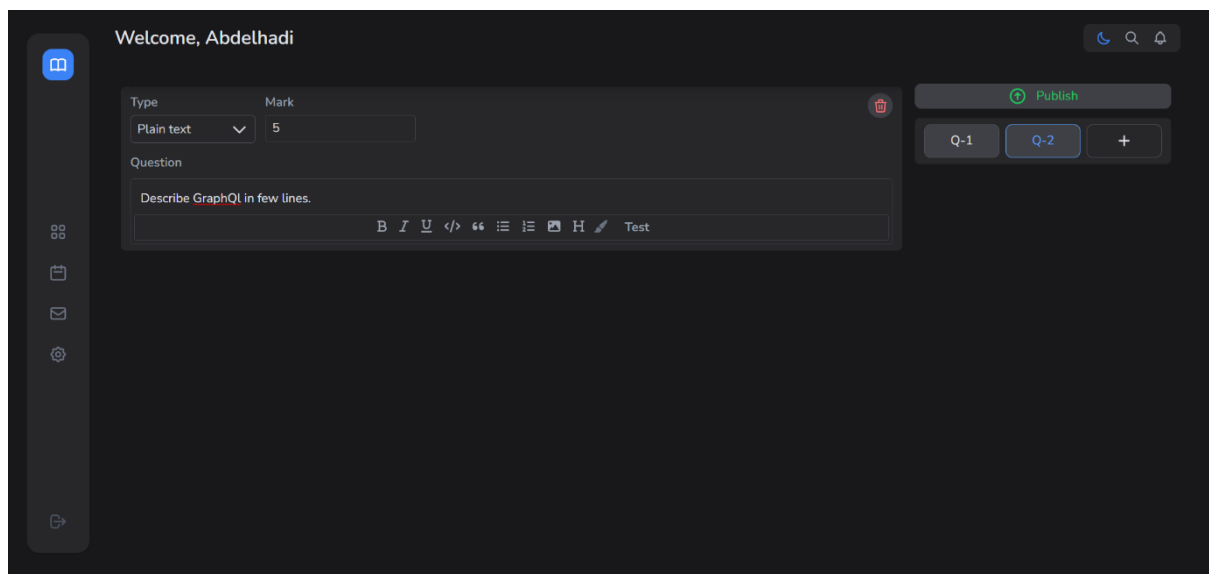


Figure 24 : Publier l'examen

Remarque :

Le professeur et le chef de département ont les mêmes fonctions. Le chef de département peut être considéré comme un professeur, mais l'inverse est invalide.

Il y a plusieurs fonctionnalités qu'on va développer dans les versions futures. Maintenant nous allons voir le côté Etudiant.

4. Partie Etudiant

Une fois que l'étudiant s'est inscrit, sa page de tableau de bord apparaît, on voit qu'il est divisé en deux parties :

Tout d'abord, nous voyons tous les éléments pour l'étudiant sous forme de boutons, et chaque bouton vous amènera à l'élément souhaité.

Ensuite, nous trouvons les derniers éléments pour lesquels l'étudiant a assisté.

De plus, nous retrouvons tous les articles qui se trouvent dans la Section des étudiants.

Les éléments possèdent une barre de progression pour suivre le progrès de chaque élément.

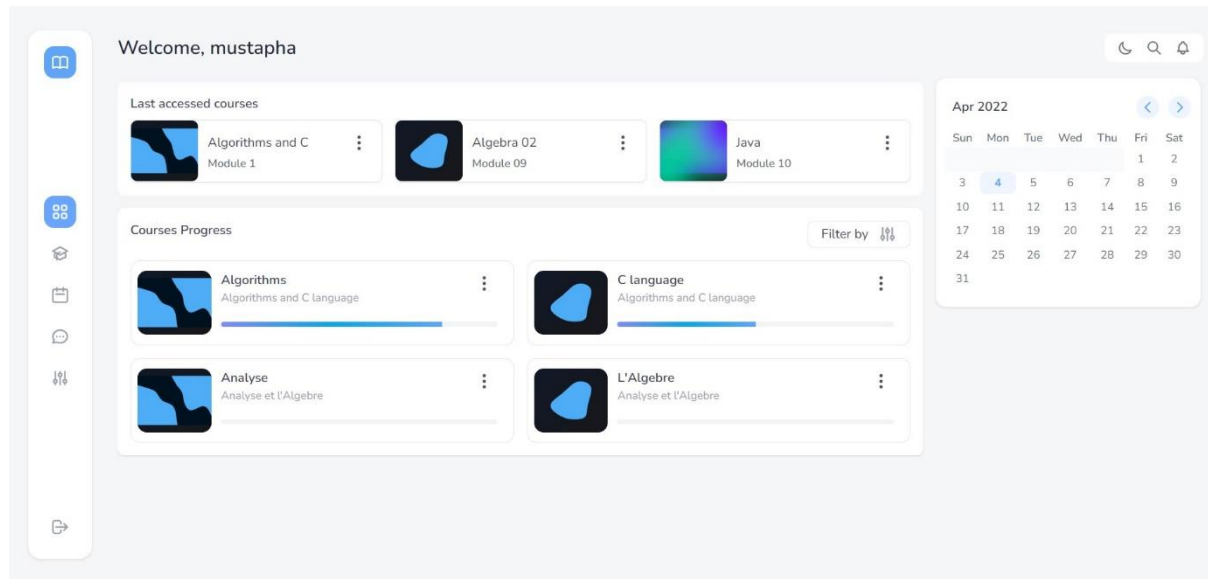


Figure 25 : Tableau de bord Etudiant

Les fonctionnalités de l'étudiant :

Ce dernier a un quelques permissions, tels que :

- ✓ Visionner un cours

Après avoir sélectionné l'élément souhaité, une nouvelle page apparaîtra composée de :

- Courses (contenant les lectures et les devoirs).
- DS (examen)

D'où il peut cocher les cases quand il finit chaque partie.

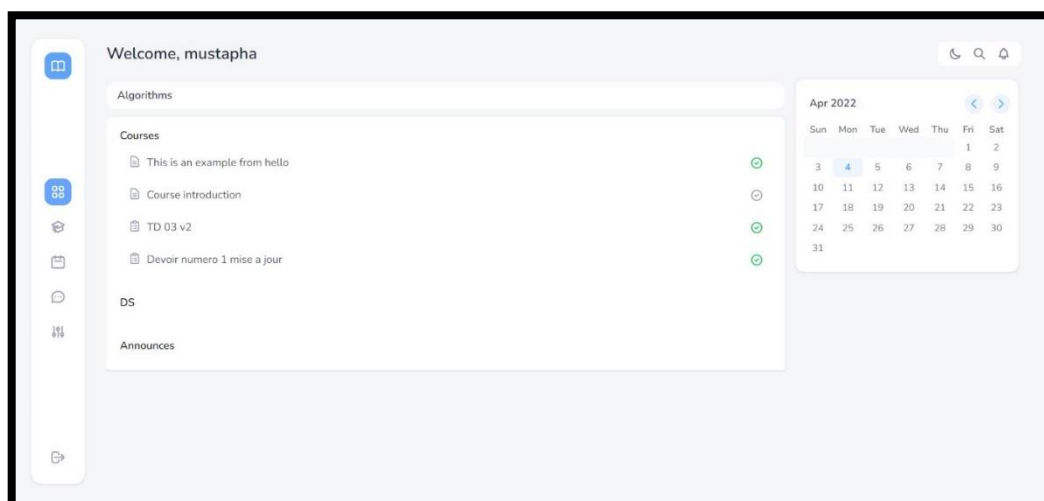


Figure 26 : Contenu d'un élément

✓ Envoyer un devoir

L'étudiant doit rendre son travail lors d'un devoir programmer par son prof avant le délai précisé.

Welcome, mustapha

Algorithms / Devoir numero 1 mise a jour

Open untill Mon, 04-22 22:00

This is a homework

Add your work here. Drag and drop file here.

Type some text..

B I U </> ¶ ☰ ☷ H {}

Submit

Apr 2022

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Figure 27 : Accéder au devoir

Tant que le devoir est ouvert l'étudiant peut modifier sa réponse.

Welcome, mustapha

Algorithms / Devoir numero 1 mise a jour

Open untill Mon, 04-22 22:00

This is a homework

My answer:

done

Update

Figure 28 : Deposer le devoir

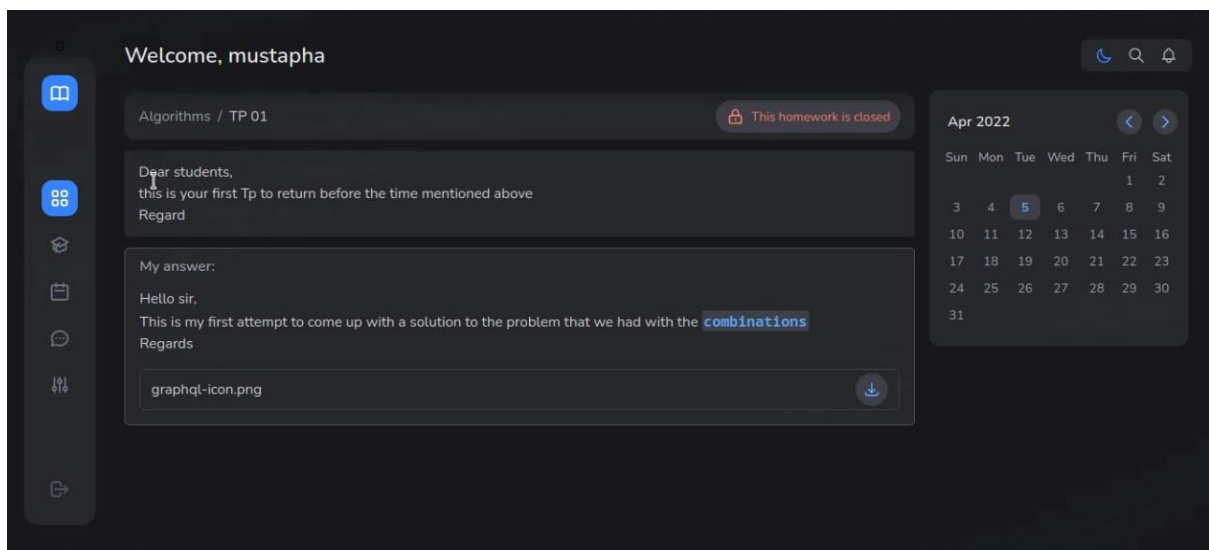


Figure 29 : Delai expirer

✓ Passer un examen en ligne

L'étudiant peut passer un examen programme par son prof, tout cela avec un processus bien développé pour éviter la tricherie (à mieux comprendre lors de la simulation dans la soutenance)

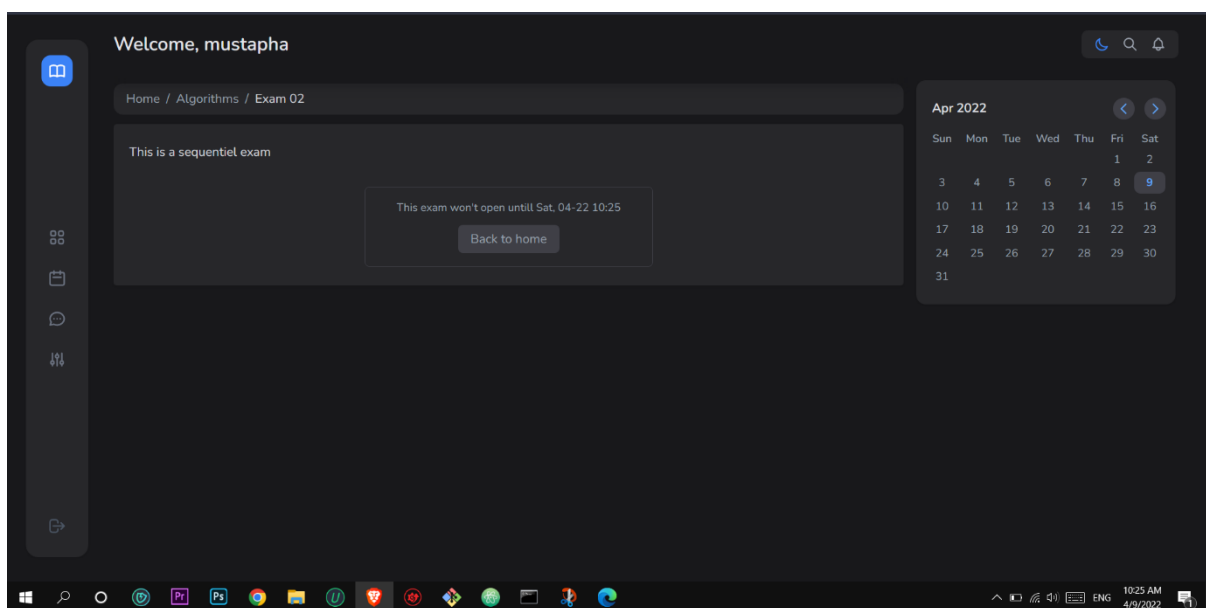


Figure 30 : Examen Avant ouverture

Une fois l'examen débute, l'étudiant aura la possibilité de faire une tentative pour passer l'examen

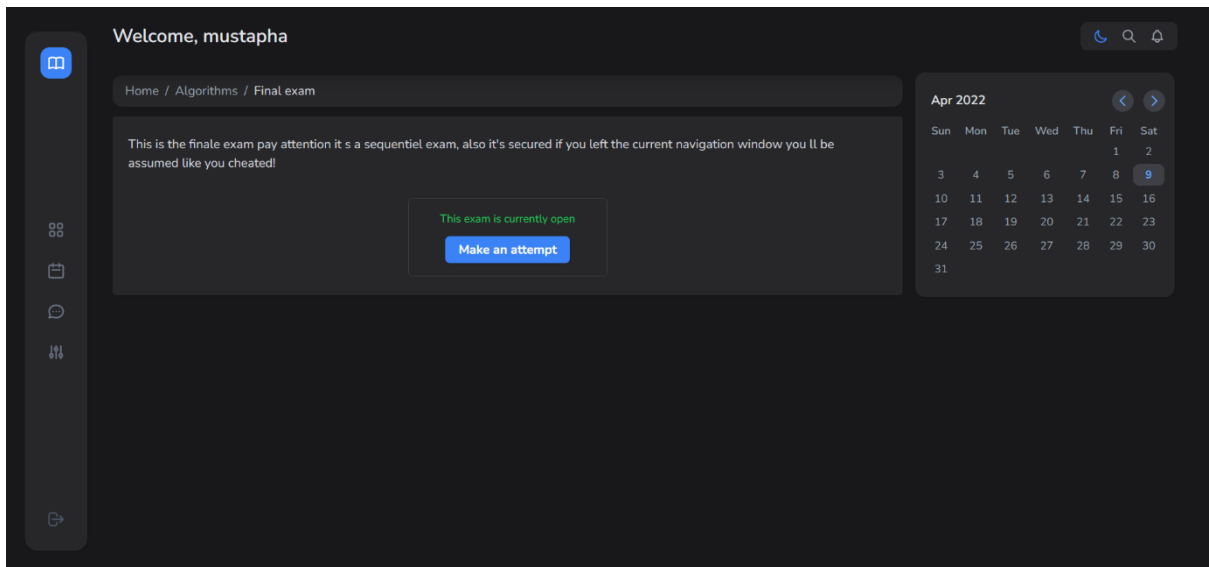


Figure 31 : Faire une tentative

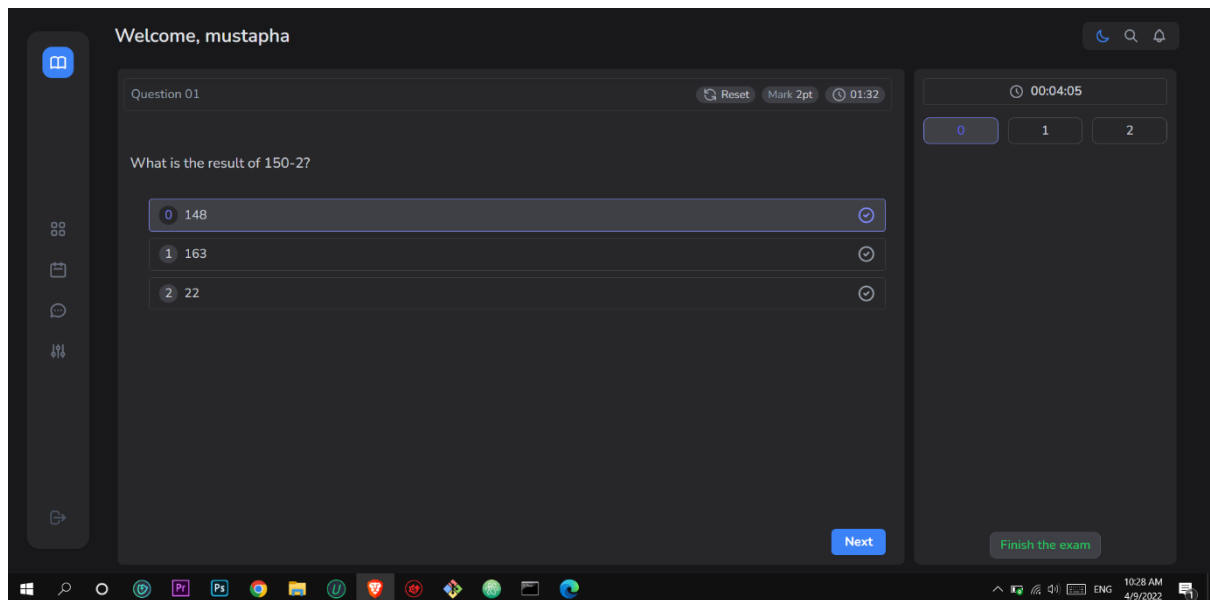


Figure 32 : Passer un examen

Lors de la fin l'étudiant clique sur le bouton « Finish the exam » afin d'envoyer les réponses, ensuite le professeur corrige l'examen question par question en donnant une note pour chacune.

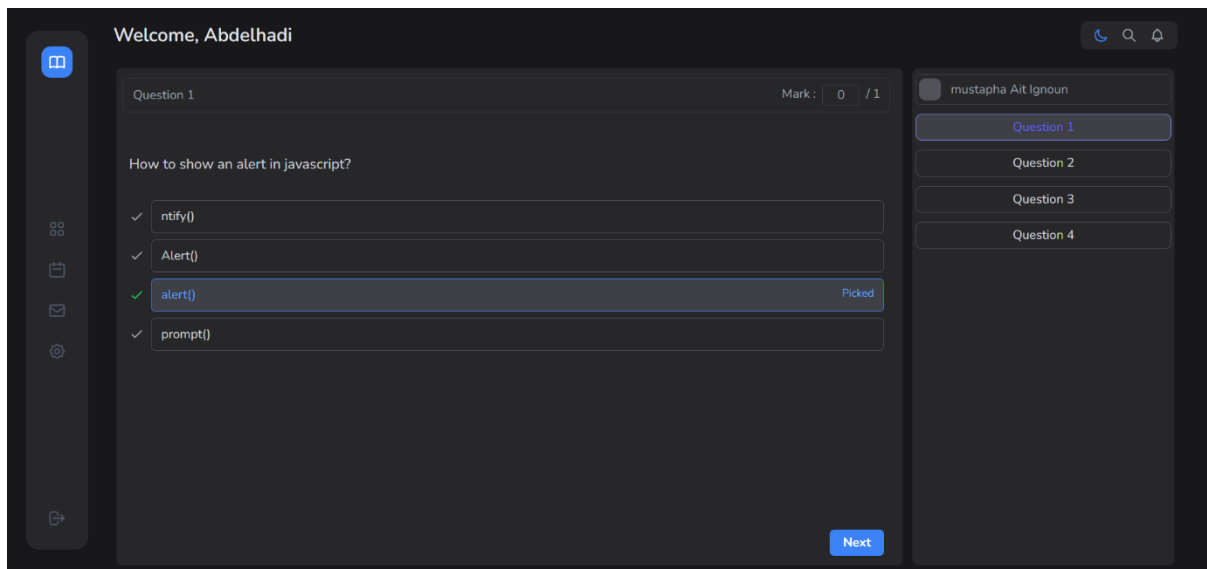


Figure 33 : Correction de l'examen par le prof

Enfin l'étudiant aura sa note immédiatement après la correction de l'examen par son prof

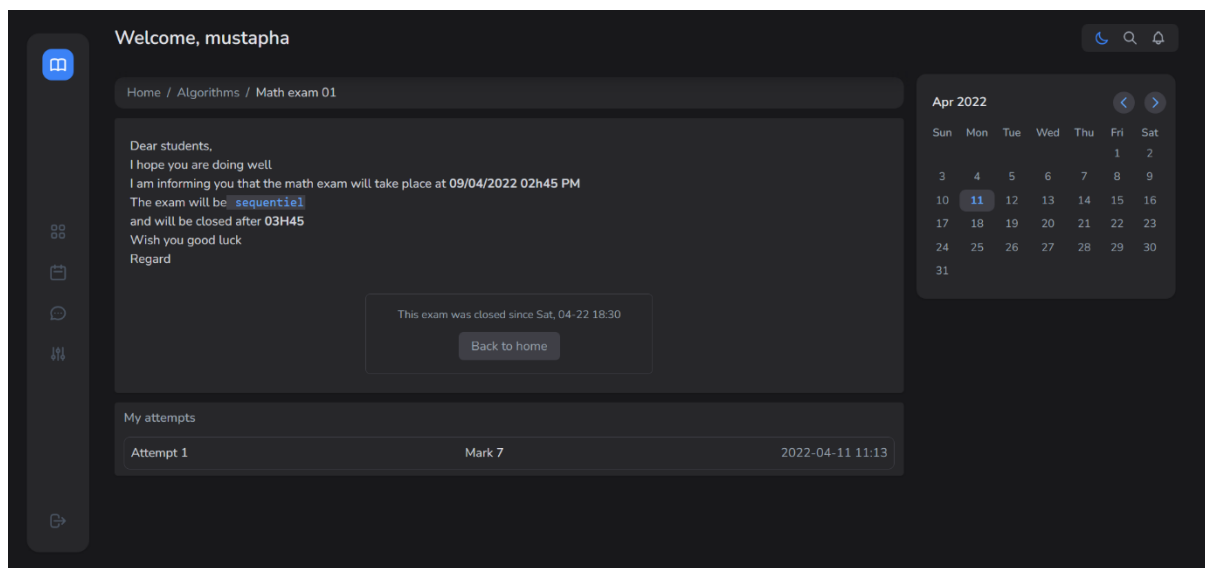


Figure 34 : Affichage de la note

5. Partie administrateur

Nous avons travaillé avec le cadre de développement Django afin de faciliter la création et la gestion du système de notre plateforme

Tout d'abord on lance le serveur local après l'activation de l'environnement virtuel en suivant les commandes suivantes

```
Command Prompt - python manage.py runserver
Microsoft Windows [Version 10.0.19044.1586]
(c) Microsoft Corporation. All rights reserved.

C:\Users\fardo>cd desktop\ecour-platform
The system cannot find the path specified.

C:\Users\fardo>cd desktop

C:\Users\fardo\Desktop>cd ecour-platform

C:\Users\fardo\Desktop\ecour-platform>cd ecour-platform

C:\Users\fardo\Desktop\ecour-platform\ecour-platform>venv\Scripts\activate

(venv) C:\Users\fardo\Desktop\ecour-platform\ecour-platform>cd backend

(venv) C:\Users\fardo\Desktop\ecour-platform\ecour-platform\backend>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
April 09, 2022 - 10:09:36
Django version 4.0.3, using settings 'backend.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Figure 35 : Lancement du serveur de développement

L'admin a le contrôle total afin de gérer la plateforme ou il peut ajout des utilisateurs, des filières...

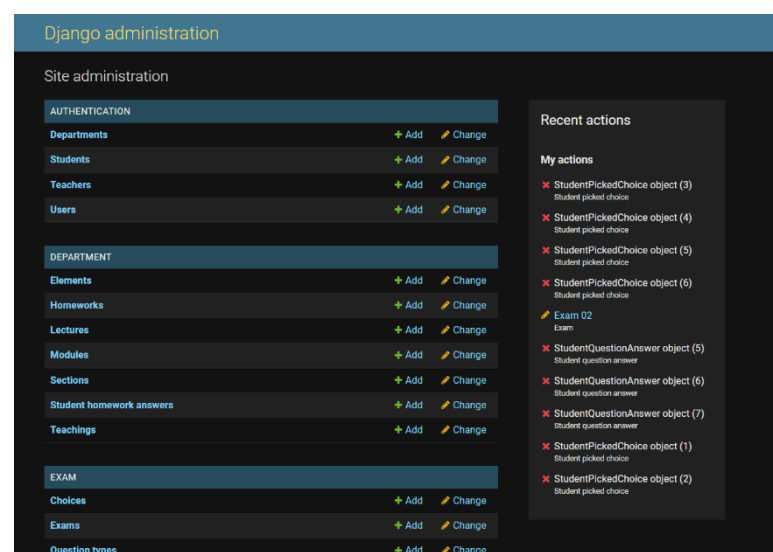


Figure 36 : Page d'accueil Django

Dans les prochaines versions nous allons développer plusieurs fonctionnalités tels que :

- ✓ Consultation des profils (Version prochaine)
- ✓ Fonctionnalité de messagerie (Version prochaine)

Conclusion

Dans ce chapitre nous avons mentionné le travail que nous avons réalisé en présentant les interfaces et les fonctionnalités afin de clarifier l'utilisation de notre plateforme.

Conclusion générale

Ce projet de fin d'étude consiste à réaliser une plateforme e-learning plus développée que Moodle afin de faciliter l'enseignement à distance.

Au cours de ce rapport, nous avons présenté les différentes étapes de la gestion de notre projet commençant par la définition du contexte général, suivi par la conception la méthode UML et enfin finissant par la phase de réalisation de notre projet afin d'exposer la plateforme.

Ce projet a fait l'objet d'une expérience intéressante, qui nous a permis d'améliorer nos connaissances et nos compétences dans le domaine informatique. Nous avons appris à mieux manipuler les langages JavaScript, React, Python, ainsi la découverte des nouvelles technologies comme GraphQL : Language de requêtes API et Tailwind CSS pour créer des interfaces magnifiques.

Malgré les difficultés rencontrées nous avons réussi à livrer un produit performant qui répond aux exigences de l'utilisateur.

En effet, ce travail étant une œuvre humaine, n'est pas un modèle unique et parfait, c'est pourquoi nous restons ouverts à toutes les critiques et nous sommes prêts à recevoir toutes les suggestions et remarques tendant à améliorer d'avantage ce travail. Etant donné que tout travail informatique a été toujours l'œuvre d'une équipe.

Webographie

<https://fr.wikipedia.org/>

<https://ecours.esta.ac.ma/>

<https://tailwindcss.com/>

<https://docs.djangoproject.com/en/4.0/>

<https://www.youtube.com/watch?v=ZQL7tL2S0oQ>

<https://scrimba.com/learn/react>

<https://www.udemy.com/>

<https://github.com/>

<https://stackoverflow.com/>

<https://www.python.org/>

<https://heroicons.dev/>

<https://boxicons.com/>

<https://headlessui.dev/react/menu>