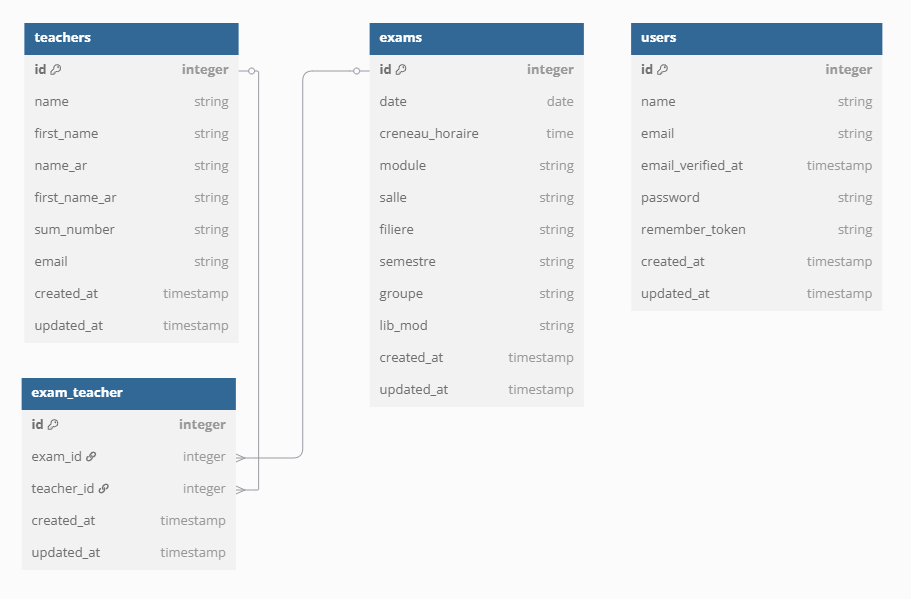
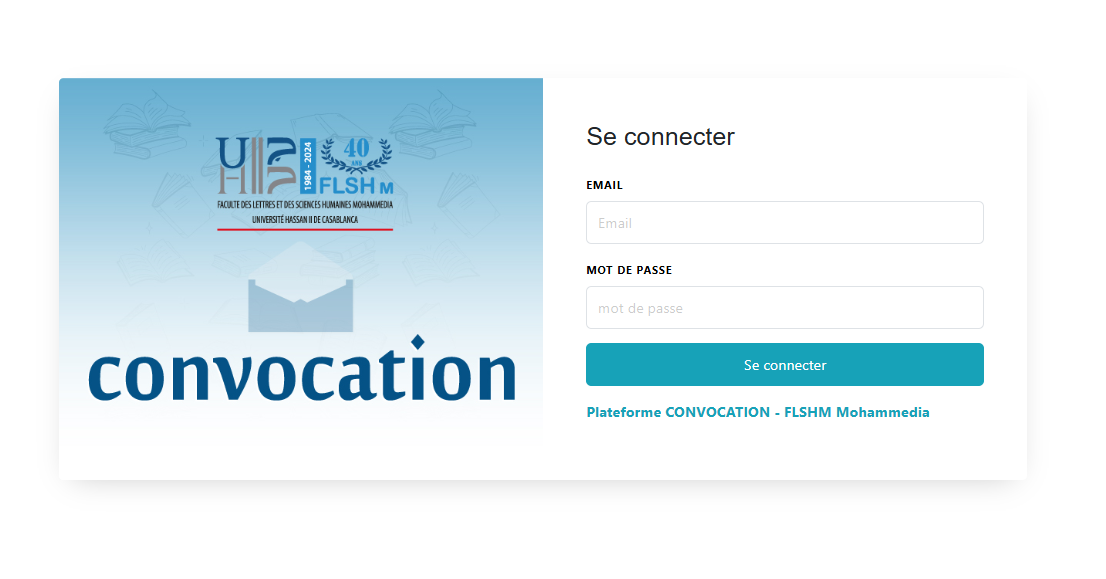
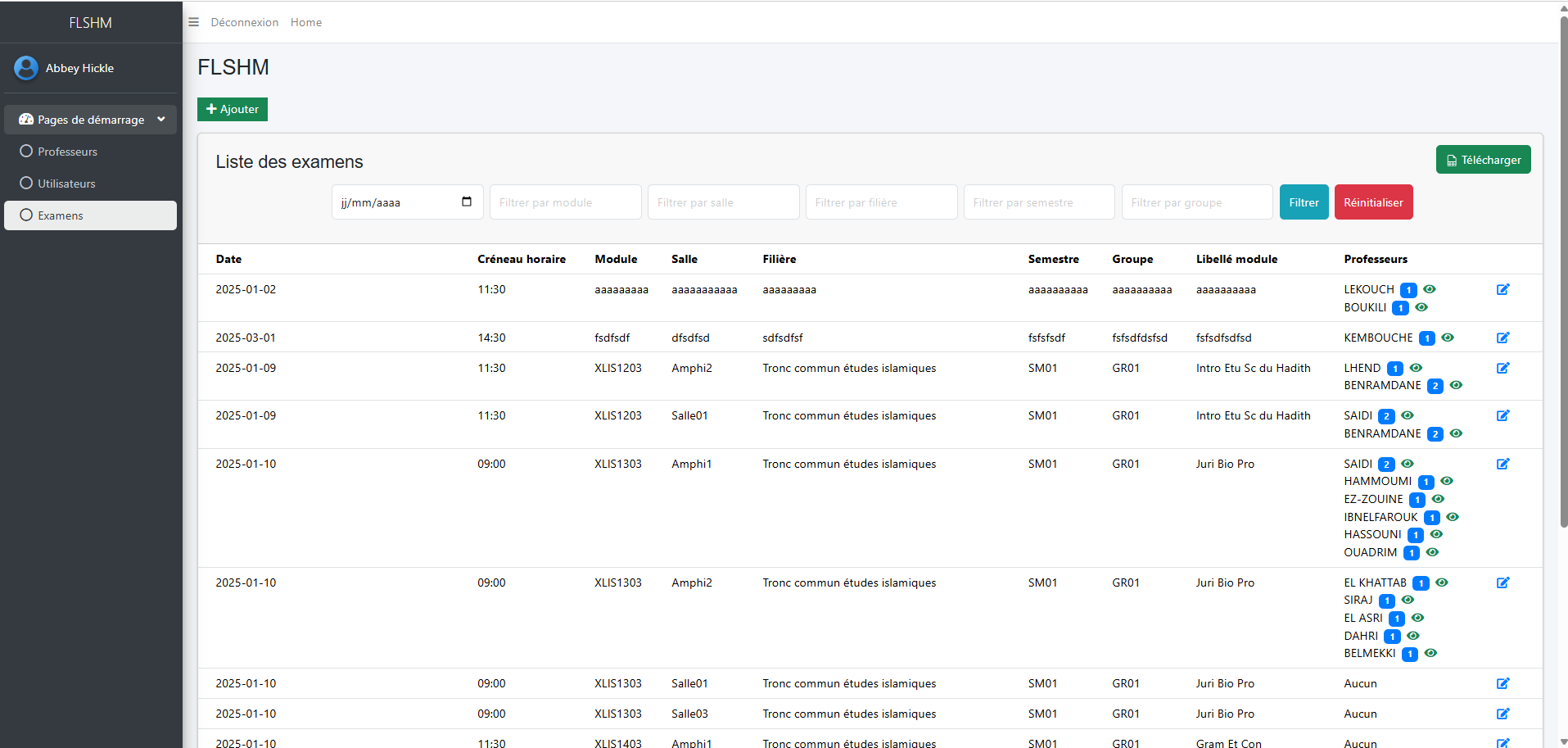
**Base de données**



**Page Login**



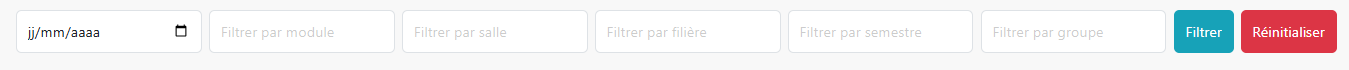
**Page Examens**



**1 -** Système de gestion CRUD

2 - Télécharger en ExcelC:\Users\user\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Capture d’écran 2025-03-25 130346.png

3 - Filtrage sur tous les champs



4 - Explication avec les détails de cette page ( affectation )

* **Si le créneau horaire de l'examen est** **09:00:00**
  + On récupère les enseignants qui n'ont **pas** d'examen à 09:00:00 ce jour-là.
  + On récupère aussi ceux qui ont un examen à d'autres horaires (11:30:00, 14:00:00, 14:30:00, 16:30:00, 17:00:00).
* **Si le créneau horaire de l'examen est 11:30:00**
  + On récupère les enseignants sans examen à 11:30:00 ce jour-là.
  + On récupère ceux qui ont un examen à 09:00:00, 14:00:00, 14:30:00, 16:30:00, ou 17:00:00.
* **Si le créneau horaire de l'examen est 14:00:00 ou 14:30:00**
  + On récupère les enseignants sans examen à 14:00:00 ou 14:30:00 ce jour-là.
  + On récupère ceux qui ont un examen à d'autres horaires (09:00:00, 11:30:00, 16:30:00, 17:00:00).
* **Si le créneau horaire de l'examen est 16:30:00 ou 17:00:00**
  + On récupère les enseignants sans examen à 16:30:00 ou 17:00:00 ce jour-là.
  + On récupère ceux qui ont un examen à d'autres horaires (09:00:00, 11:30:00, 14:00:00, 14:30:00).
* **Si le créneau horaire ne correspond à aucun des cas ci-dessus**
  + On considère que tous les enseignants sont disponibles.

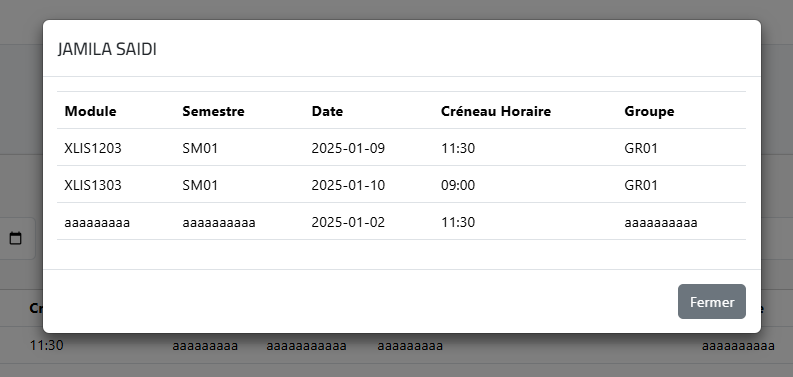
**Tri des enseignants**

* Les enseignants qui sont déjà occupés à une autre date sont placés en bas de la liste.

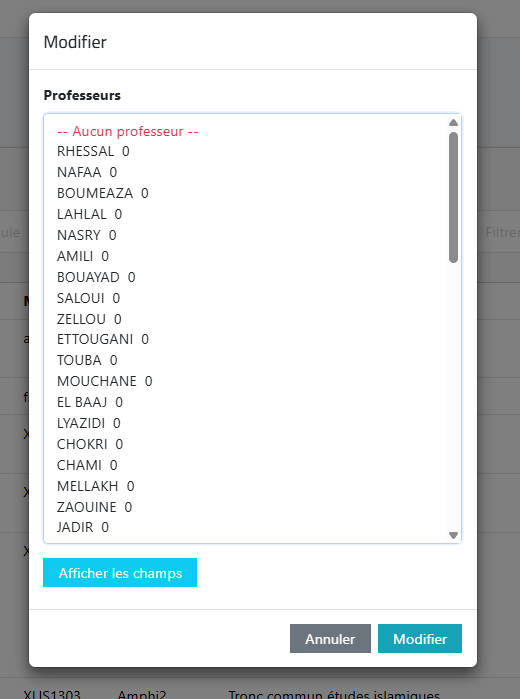
5 - Affichage du numéro de créneau à côté de chaque professeur



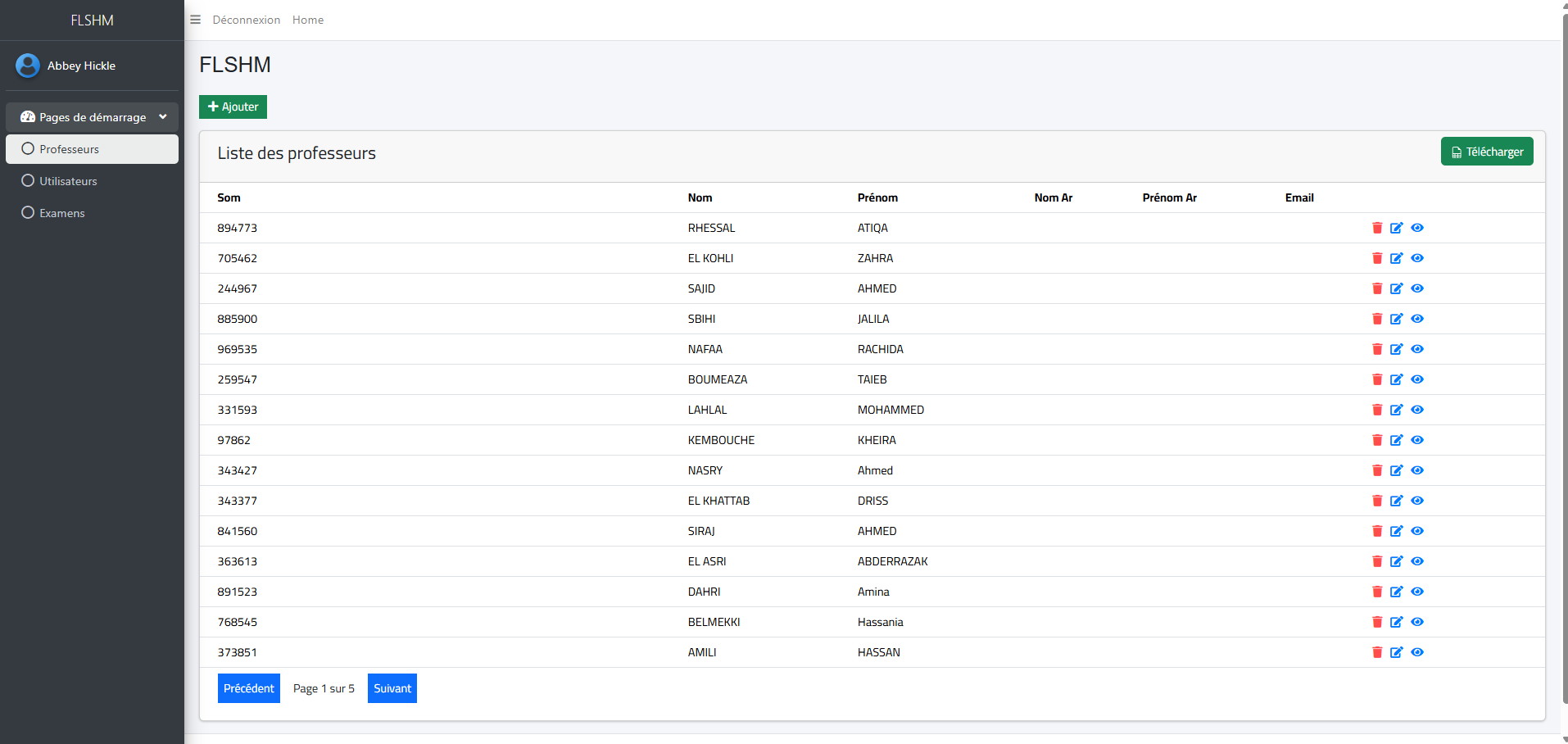
6 – Affichage de la liste des créneaux de chaque professeur.



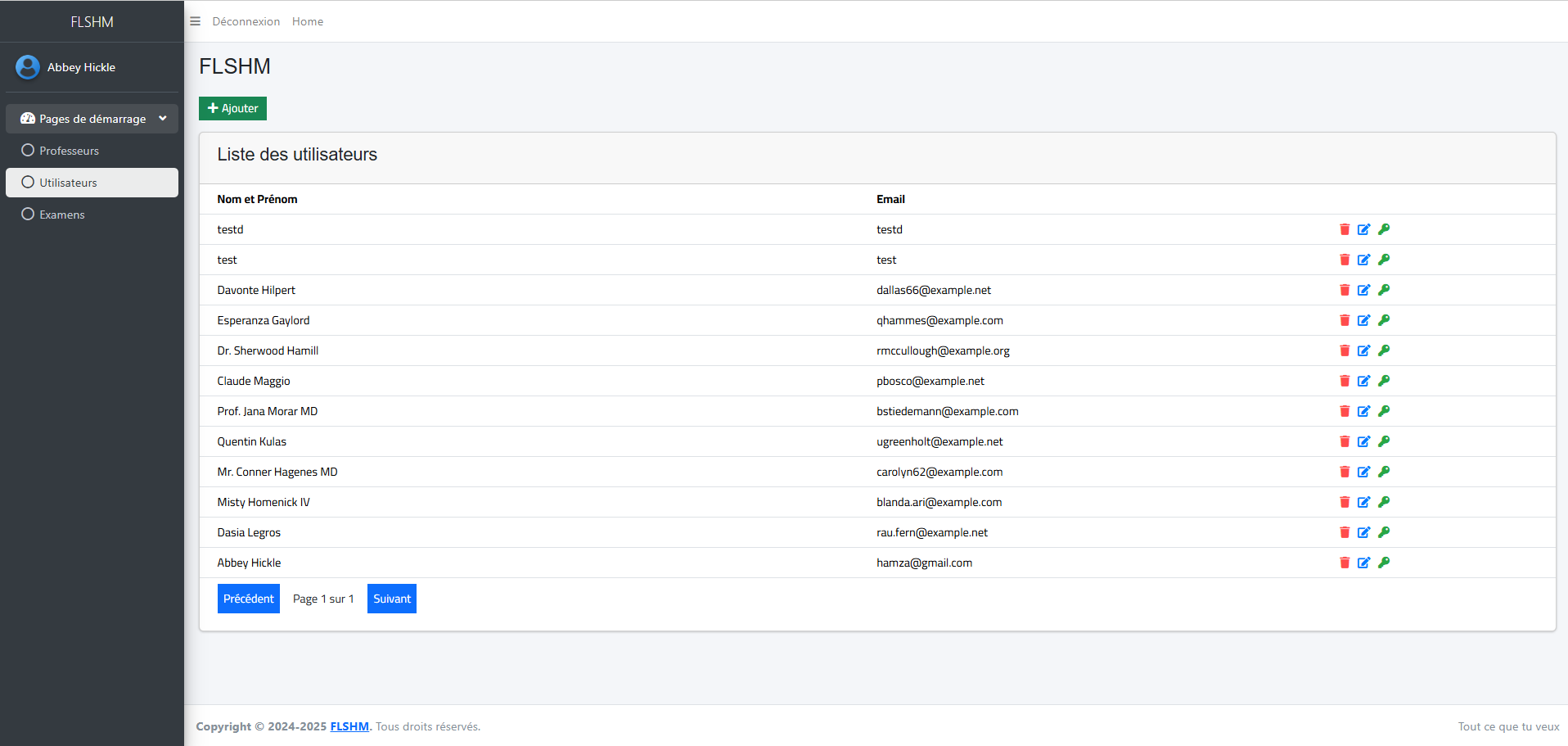
7 – Modal des listes des professeurs pour affecter un professeur ou plusieurs professeurs à l'examen.



### 8 – Page Professeurs



9 – Page Utilisateurs



**En Back-end**

**Controller ExamController :**

 **all()** : Récupère tous les examens avec leurs enseignants associés.

 **index()** : Liste les examens avec des filtres comme la date, le module, etc.

 **store()** : Crée un nouvel examen et associe les enseignants si fournis.

 **show()** : Affiche un examen spécifique.

 **update()** : Met à jour un examen existant et ses enseignants associés.

 **destroy()** : Supprime un examen spécifique.

**Controller TeacherController :**

 **getExamDunProf($id)** : Récupère les examens d'un professeur spécifique en fonction de son ID, avec les détails du professeur (nom, prénom).

 **getTeachersDisponibles(Request $request)** : Récupère les enseignants disponibles à une heure et une date données, en fonction des créneaux horaires déjà réservés.

 **all()** : Récupère tous les enseignants avec leurs examens associés.

 **index()** : Liste les enseignants avec pagination, triés par date de création.

 **store(StoreTeacherRequest $request)** : Crée un nouvel enseignant avec les informations fournies par la requête.

 **show(Teacher $teacher)** : Affiche les détails d'un enseignant spécifique.

 **update(UpdateTeacherRequest $request, Teacher $teacher)** : Met à jour les informations d'un enseignant spécifique.

 **destroy(Teacher $teacher)** : Supprime un enseignant spécifique de la base de données.

**Controller UserController :**

 **index()** : Récupère tous les utilisateurs, triés par date de création, et les affiche avec une pagination de 15 utilisateurs par page.

 **store(StoreUserRequest $request)** : Crée un nouvel utilisateur avec les informations de la requête (nom, email, mot de passe). Le mot de passe est crypté avant d'être enregistré.

 **show(User $user)** : Affiche les détails d'un utilisateur spécifique.

 **update(UpdateUserRequest $request, User $user)** : Met à jour les informations d'un utilisateur (nom, email, mot de passe) en fonction de la requête, puis sauvegarde les modifications.

 **updatePassword(Request $request, User $user)** : Permet de mettre à jour le mot de passe d'un utilisateur en utilisant un nouveau mot de passe envoyé dans la requête.

 **destroy(User $user)** : Supprime un utilisateur de la base de données.