

Tp 05 Les formulaires

AMONA Birewa Audrey

21 novembre 2023

Table des matières

1 Mise en route	1
1.1 Compréhension	1
1.2 Création	2

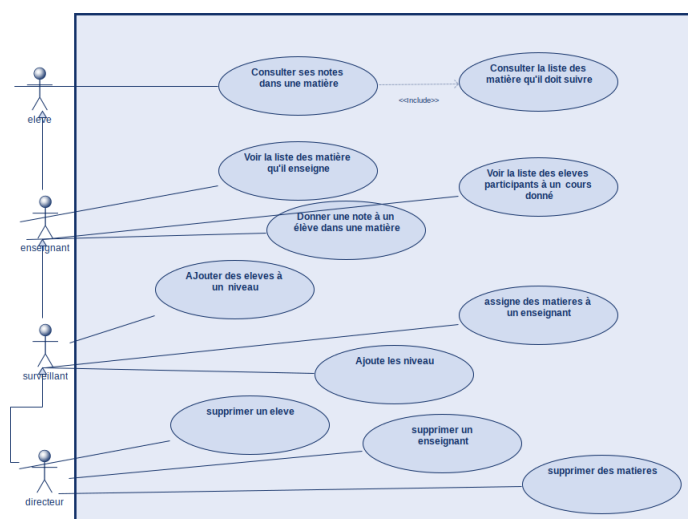
1 Mise en route

Les formulaires sont les parties les plus importantes, donc dangereuses, de tout site web. C'est depuis là que l'utilisateur peut modifier la base de données au travers de la méthode POST de HTTP. Nous allons créer des formulaires pour les modèles de notre application dont nous ne nous sommes pas occupés jusqu'à présent.

1.1 Compréhension

1. Pour éviter tout problème avec la base de données, nous allons gérer les éléments des pages de notre site pour éviter qu'un utilisateur qui n'a pas un droit effectue une action.
2. Pour notre site de **notes** on peut dégager les utilisateurs suivants :
 - Enseignant
 - Élève
 - Surveillant
 - Directeur d'étude

Le diagramme de cas d'utilisation suivant illustre les accès de chaque utilisateur :



Les modèles déjà créés n'ont pas besoin de vues de formulaire à cause du site administrateur

3. Dans les Tp précédents, nous n'avons pas créé de vue pour le modèle **Note**. Seul les enseignants, les surveillants et le directeur auront accès aux vues de formulaires de ce modèle. Ces vues serviront à afficher les templates de formulaires et à ajouter les notes à des matières pour un étudiant donné.

1.2 Création

1. Dans le fichier **views.py**, nous allons créer une vue pour le modèle **Note** nommée **add_note**. Cette vues aura trois entrées : le request, l'id de l'eleve, et l'id d'une matière qu'il suit.

```
1
2 def add_note(request, eleve_id, matiere_id):
3     return HttpResponse("Note d'un élève dans une matière.")
4
5
```

2. Nous allons ajouter cette vue dans **urls.py** Et tester l'affichage. Dans la liste **urlpatterns** :

```
1 path("add_note/<int:eleve_id>/<int:matiere_id>/", views.add_note, name="add-note"),
2
```

Note d'un élève dans une matière.

3. Nous allons créer le template **add_note.html** dans le répertoire **notes/templates/notes**.

```
1 {% extends 'notes/index.html' %}
2
3
4 {% block content %}
5
6     <div class="newsletter-subscribe">
7         <div class="container">
8             <div class="intro">
9                 <h2 class="text-center">Ajouter une note</h2>
10                <p class="text-left">Étudiant : {{ eleve.nom }} {{ eleve.prenom }}</p>
11                <p class="text-left">Matière : {{ matiere.nom }}</p>
12            </div>
13            <form class="form-inline" method="post">
14                <div class="form-group"><input class="form-control" type="number" name="valeur" placeholder="La note"></div>
15                <div class="form-group"><button class="btn btn-primary" type="submit">Ajouter </button></div>
16            </form>
17        </div>
18    </div>
19
20
21 {% endblock %}
```

4. NOus allons maintenant ajouter des lien dans le fichier de details élève pour accéder au template précédemment créé.

```

1  {% extends 'notes/index.html' %}
2
3  {% block content %}
4      <!-- component -->
5      <div class="bg-gray-200 font-sans h-screen w-full flex flex-row justify-center items-center">
6          <div class="card w-96 mx-auto bg-white shadow-xl hover:shadow">
7              
8
9              <div class="text-center mt-2 text-3xl font-medium">{{ eleve.nom }} {{ eleve.prenom }}</div>
10             <div class="text-center mt-2 font-light text-sm">{{ eleve.niveau.nom }}</div>
11             <div class="text-center font-normal text-lg">{{ eleve.date_naissance }}</div>
12             <div class="px-6 text-center mt-2 font-light text-sm">
13                 <p>
14
15                 {% for matiere in eleve.niveau.matiere_set.all %}
16                 <a class="btn btn-primary" data-toggle="modal" data-target="#exampleModal{{ matiere.nom.split.0 }}" href="#" >{{ matiere }}</a>
17                 <!-- Modal -->
18                 <div class="modal fade" id="exampleModal{{ matiere.nom.split.0 }}" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
19                     <div class="modal-dialog">
20                         <div class="modal-content">
21                             <div class="modal-header">
22                                 <h5 class="modal-title" id="exampleModalLabel">{{matiere}}</h5>
23                                 <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
24                             </div>
25                             <div class="modal-body">
26                                 {% for note in matiere.note_set.all %}
27                                 <p>{{note}}</p>
28                                 {% endfor %}
29                                 <a class="btn btn-primary" href="{% url 'notes:add-note' eleve.id matiere.id %}" >Ajouter une note</a>
30                             </div>
31                         </div>
32                     </div>
33                 </div>
34                 {% endfor %}
35             </p>
36             </div>
37             <hr class="mt-8" />
38         </div>
39     </div>
40 {% endblock %}

```

5. Nous allons maintenant rectifier la vue de note pour qu'elle puisse avoir un comportement logique.

```

1
2  def add_note(request, eleve_id, matiere_id):
3      try:
4          eleve = Eleve.objects.get( pk=eleve_id)
5          matiere = Matiere.objects.get( pk=matiere_id)
6      except (Matiere.DoesNotExist) as e1:
7          error_message = "Une erreur est survenue: La matière spécifiée n'existe pas."
8          return HttpResponse(error_message)
9      except (Eleve.DoesNotExist) as e2:
10         error_message = "Une erreur est survenue: L'élève spécifié n'existe pas."
11         return HttpResponse(error_message)
12
13     if request.method == "POST":
14         valeur = request.POST.get("valeur")
15         Note.objects.create(valeur=valeur, eleve=eleve, matiere=matiere)
16         return render(request, "eleves/show.html", {"eleve": eleve})
17
18     if matiere not in eleve.niveau.matiere_set.all():
19         error_message = f"Une erreur est survenue: L'élève {eleve.nom} {eleve.prenom} ne suit pas le cours de matricule {matiere_id}"
20         return HttpResponse(error_message)
21
22     return render(request, "notes/add_note.html", {"matiere": matiere, "eleve": eleve})

```

Voici le contenu du template associé :

```

1  {% extends 'notes/index.html' %}
2
3
4  {% block content %}
5
6      <div class="newsletter-subscribe">
7          <div class="container">
8              <div class="intro">
9                  <h2 class="text-center">Ajouter une note</h2>
10                 <p class="text-left">Étudiant : {{ eleve.nom }} {{ eleve.prenom }}</p>
11                 <p class="text-left">Matière : {{ matiere.nom }}</p>
12             </div>
13             <form class="form-inline" method="post">
14                 <div class="form-group"><input class="form-control" type="number" name="valeur" placeholder="La note"></div>
15                 <div class="form-group"><button class="btn btn-primary" type="submit">Ajouter </button></div>
16             </form>
17         </div>
18     </div>
19
20
21 {% endblock %}

```

6. Voici le resultat dans le site administration d'une note créer :

15.0

Valeur:

Eleve:

Matiere: