



Travaux Pratiques : Enoncés

TP_ReviewPhoto_Partie_05

Objectif : Consolidation sur le framework Symfony 6
Affichage de la vue d'une photo

Lorsque l'on clique sur le titre d'une photo, on veut afficher cette photo en plus grand avec à sa droite ses caractéristiques et en dessous la liste des commentaires qui ont déjà été faits.

De plus, si l'utilisateur est connecté, un formulaire de saisie d'un nouveau commentaire apparaîtra en dessous des caractéristiques de la photo.

Réalisations :

1 : Il faut ajouter au contrôleur des photos la méthode « show() ».

La route pour atteindre l'action du contrôleur sera de la forme : photo/15 où 15 est un exemple qui représente l'id de la photo.

- création de la méthode show() avec deux paramètres :
 - le repository de l'entité Photo afin de la récupérer dans la base
 - la requête http reçue qui contient l'id de la photo cherchée

```
public function show(PhotoRepository $repository, Request $request){...}
```

- Annoté cette méthode avec la route :

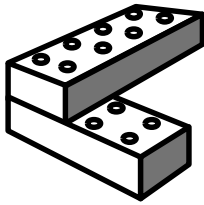
```
#[Route('/photo/{id}', name : 'photo.show')]
```

La notation {id} présente dans l'url, notifie à symfony qu'il y a un paramètre de query string qui porte ce nom

Dans la méthode show() :

- Récupération de la photo dont l'id est donné dans la requête :
 - Première solution : on a mis un paramètre de type « PhotoRepository » dans cette méthode, on peut donc appeler le méthode « find() » qui existe par défaut dans ce repository

```
$photo = $PhotoRepository->find($request->get('id')) ;
```
 - Deuxième solution encore plus rapide : au lieu de mettre un paramètre de type « PhotoRepository », mettons un paramètre de type « Photo » que l'on appelle « \$photo ». Symfony va directement (via doctrine) nous donner dans ce paramètre l'objet « Photo » correspondant à l'id. Nous n'avons rien à faire d'autre.
- Retourner la vue « show.html.twig » en lui passant cette photo



Travaux Pratiques : Enoncés

TP_ReviewPhoto_Partie_05

```
return $this->render('photo/show.html.twig', ['photo' => $photo]) ;
```

- Création de la vue « show.html.twig » dans le dossier « photo » du dossier « templates »
 - Afficher les caractéristiques de la photo dans un « jumbotron »
 - Afficher tous les commentaires dans une table html

Il va nous falloir tester si l'utilisateur est connecté pour afficher le formulaire de saisie d'un nouveau commentaire.

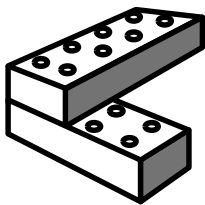
Pour cela, une fois connecté, l'objet « user » est mis en session par Symfony et restauré à chaque nouvelle requête.

Il est possible de le récupérer en faisant appel à la méthode « getUser() » de l'objet « app » ce qui se traduit en « twig » par « app.user ».

Il suffit de faire un « {% if app.user %} ».

- Code :

```
{% block body %}
    <div class="jumbotron">
        <div class="container">
            <div class="row">
                <div class="col-md-8">
                    
                </div>
                <div class="col-md-4">
                    <h1>{{ photo.title }}</h1>
                    <h2>Postée le : {{ photo.postAt|date("d/m/Y") }}</h2>
                    <div class="text-primary" style="font-weight: bold; font-size: 1.3rem">
                        <p>Par : {{ photo.user.pseudo }}</p>
                    </div>
                    {% if app.user %}
                        <a href="#" class="btn btn-primary">Mettre un commentaire</a>
                    </div id="commentForm" class="mt-4">
                        <p>formulaire de saisie du commentaire</p>
                    </div>
                {% endif %}
            </div>
        </div>
    </div>
    <div class="container mt-4">
        <div class="row">
            <div class="col-md-8">
                <h2>Commentaires</h2>
                <table class="table table-striped">
                    <tr>
                        <th>Commentée par</th>
                        <th>Date</th>
                        <th>Commentaire</th>
                    </tr>
                    <tr>
                        <td>le pseudo</td>
                        <td>la date de création</td>
```



Travaux Pratiques : Enoncés

TP_ReviewPhoto_Partie_05

```
<td>le commentaire</td>
</tr>
</table>
</div>
</div>
</div>
{% endblock %}
```

- **Modifier la vue « list.html.twig » afin de mettre à jour le lien du titre de la photo**

```
<a href="{{ path('photo.show', {id: photo.id}) }}">{{ photo.title }}</a>
```

4 : tester