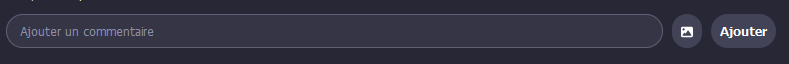
# Fonctionnalités Yin

Ce document donne des indices, mais bien entendu, il n’est ni exhaustif, ni parfait.

Étape C (Notes de la semaine 11)

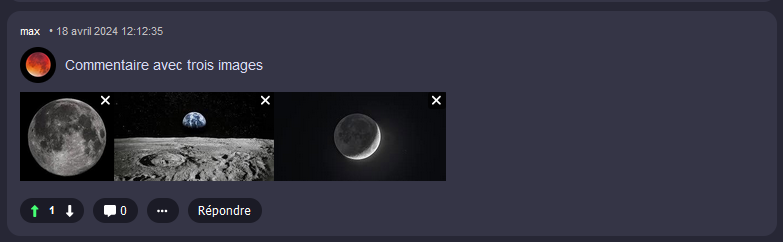
« Lorsqu’on crée un commentaire, on doit être capable d’y joindre une ou plusieurs images. »

* Attention, on peut créer un commentaire à deux endroits : sous le commentaire principal d’un post (composant **PostComponent**) et sous n’importe quel commentaire (composant **CommentComponent**)





* Cliquer sur le bouton avec l’icône d’image permet d’afficher un input.
* Dans les notes de cours, on a seulement vu comment joindre une image à un input pour envoyer une seule image au serveur. Dans ce cas, ce sera un peu différent. Ce problème se règle entre autres avec une boucle while. (Vous pourriez donner à chaque image l’étiquette "image0", "image1", "image2", etc. dans le formData)
* Le serveur reçoit un **CommentDTO** dans le corps de la requête actuellement pour créer un nouveau commentaire… ce ne sera plus possible car on va maintenant envoyer un **formData**. Il faudra remplacer le DTO par un formData.
* N’oubliez pas d’afficher les images dans le composant **CommentComponent**. Pour y arriver, il faudra que le client Angular reçoivent la liste des ids des Pictures du commentaire.
  + Gardez à l’esprit que le serveur n’envoie pas de **Comment.cs** à Angular, mais bien des **CommentDisplayDTO.cs** !



Étape D

« Les utilisateurs doivent pouvoir choisir un avatar personnalisé. Il est affiché à côté de leurs commentaires. (Mais pas dans le message principal d’un post) L’avatar peut être changé à tout moment. »





* Ceci se déroule surtout dans le composant **ProfileComponent** et dans le **UsersController**. (N’oubliez pas l’affichage dans le composant **CommentComponent** !)
* N’utilisez pas une classe séparée pour les avatars, ajoutez seulement un FileName et un MimeType dans la classe User, c’est plus simple. Exceptionnellement, vous pouvez vous en tirer sans utiliser de service côté serveur pour cette fois. (À l’aide de UserManager)
* La requête pour afficher l’avatar sera plus simple si le paramètre dans l’URL est le pseudo de l’utilisateur.

Étape E

vs

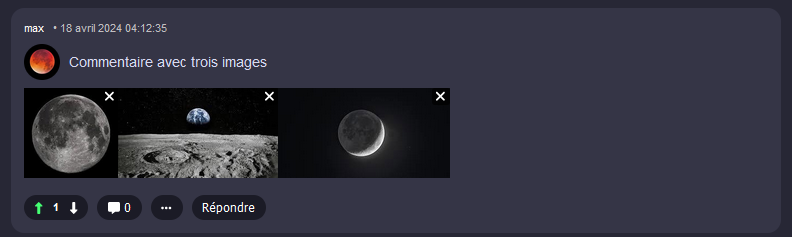
« Cliquer sur une image dans un post (commentaire principal d’un post) doit permettre de l’afficher en pleine taille, dans un autre onglet. »

* Ne vous compliquez pas la vie : Ajoutez une balise <a> avec un href qui contient la requête vers l’image en pleine taille. Cela redirigera vers une autre page qui contient seulement l’image. C’est suffisant. Vous devez ouvrir l’image dans un nouvel onglet.
* Ceci concerne TOUTES les images qui ne sont pas des avatars. (Dont le carrousel fait par l’équipier **Yang** à l’étape D) N’en oubliez pas.

Étape F

« On doit pouvoir supprimer les images d’un commentaire / post, individuellement. »

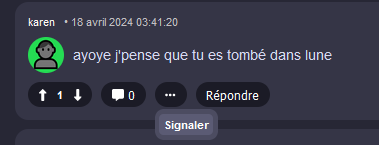
* L’image doit disparaître immédiatement de la page lorsqu’on le fait.
* Cette suppression concerne une seule image à la fois. Les autres ne sont pas touchées.
* Bien entendu, on peut seulement supprimer les images de nos propres commentaires / posts. On ne veut pas voir le petit X si on n’est pas l’auteur du message.



Étape G

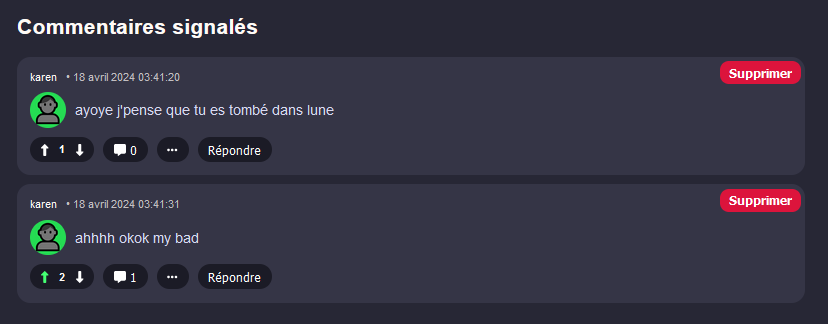
« Les utilisateurs peuvent signaler (Report) les commentaires / posts des autres utilisateurs. »

* Il n’est pas obligatoire d’offrir un feedback dans la page lorsqu’on clique sur ce bouton. Tant que le commentaire est flaggé comme signalé côté serveur.



Étape H

« Un rôle modérateur existe. Les modérateurs peuvent voir la liste des commentaires signalés. Ils peuvent supprimer les commentaires de leur choix via cette liste. Un utilisateur avec le rôle modérateur est ajouté dans le seed. »



* Trouvez un moyen de rendre accessible le composant ModerateCommentsComponant aux modérateurs. (Par exemple, avec un routerLink menant vers /reports dans **ProfileComponent**).
* Le bouton pour supprimer les commentaires appellera exactement la même action du serveur si l’utilisateur supprimait son propre commentaire… Donc le critère pour pouvoir supprimer un commentaire devient « être l’auteur OU être modérateur ».