

En plus des champs par défaut du modèle `User` de Django, vous pouvez ajouter plusieurs champs personnalisés pour différencier les utilisateurs connectés à Django Admin de ceux qui se connectent sur votre site web. Voici quelques options à considérer:

1. Champ `is_staff`:

Ce champ est déjà inclus dans le modèle `User` par défaut et est utilisé pour distinguer les utilisateurs du personnel (ceux qui ont accès à Django Admin) des utilisateurs standard. Vous pouvez définir ce champ sur `True` pour les utilisateurs du personnel et sur `False` pour les autres utilisateurs.

2. Champ `last_login_admin`:

Ce champ stocke la dernière fois qu'un utilisateur s'est connecté à Django Admin. Vous pouvez l'utiliser pour identifier les utilisateurs qui sont actifs dans l'administration et ceux qui ne l'ont pas utilisée depuis un certain temps.

3. Champ `last_login_website`:

Ce champ stocke la dernière fois qu'un utilisateur s'est connecté à votre site web. Vous pouvez l'utiliser pour identifier les utilisateurs actifs sur votre site et ceux qui ne s'y sont pas connectés depuis un certain temps.

4. Champ `user_type`:

Vous pouvez ajouter un champ personnalisé `user_type` pour catégoriser explicitement les utilisateurs en fonction de leur mode de connexion. Par exemple, vous pouvez définir `user_type` comme `'admin'` pour les utilisateurs du personnel et `'website'` pour les utilisateurs du site web.

5. Champ `device_type`:

Ce champ peut stocker le type d'appareil utilisé par l'utilisateur pour se connecter (ordinateur, mobile, tablette). Cela peut être utile pour analyser les modèles d'utilisation et cibler les utilisateurs en conséquence.

6. Champ `ip_address`:

Stocker l'adresse IP de l'utilisateur peut vous donner une indication générale de leur emplacement et de leur activité. Cependant, il est important de tenir compte des problèmes de confidentialité liés à la collecte et au stockage des adresses IP.