Cahier des charges fonctionnel : LeCOINCOIN

# Backend

## Sécurité (restrictions des accès)

Comme demandé dans le cahier des charges, seuls les comptes utilisateurs ayant pour rôle ADMIN ou MODERATOR ont un accès à la gestion des utilisateurs et à la gestion des annonces dans la partie backend du site. Ainsi, les comptes utilisateurs ayant pour rôle USER n’ont accès à aucun contrôleur dans le backend à savoir : UserController, SaleAdController et IllustrationController.

## Interfaces

### L’index/menu principal

Concernant l’interface de l’index, nous avons décidé de mettre en place une barre de menu comprenant un lien vers la gestion des utilisateurs, un lien vers la gestion des annonces, un bouton de retour à la page principale ainsi qu’un bouton de connexion nommé « Profil ». Lorsqu’un utilisateur s’identifie, le bouton prend le nom de l’utilisateur authentifié.

### La gestion des utilisateurs

#### Index

* Le design du tableau présentant la liste des utilisateurs a été amélioré.
* Les mots de passe des utilisateurs n’apparaissent plus dans ce tableau

#### Add

* Nous avons limité le formulaire d’ajout d’un utilisateur à trois champs : Rôle, Identifiant et Mot de passe

#### Edit

* Le formulaire d’ajout de modification d’un utilisateur existant possède 8 champs :
  + Rôle, Identifiant, Mot de passe comme dans le formulaire d’ajout
  + Mot de passe expiré, compte verrouillé, compte expiré et compte actif pour permettre aux administrateurs et modérateurs de bloquer un compte utilisateur si besoin
  + Ajouter une annonce

#### Show

* Les différents champs ont été traduits en français et la visualisation du mot de passe de l’utilisateur a été supprimée.

### La gestion des annonces

#### Index

* Le design du tableau présentant la liste des annonces a été amélioré.

#### Add

* Le formulaire d’ajout d’une annonce comporte désormais un champ de type « text area » pour saisir la description détaillée de l’annonce.

#### Edit

* Comme dans le formulaire d’ajout d’une annonce, il y a désormais un champ de type « text area » pour modifier la description détaillée de l’annonce.

### La gestion des illustrations

Il est possible d’ajouter une illustration à une annonce déjà créée en l’éditant (cliquer sur « Modifier l’annonce » puis « ajouter une illustration »). Les illustrations ajoutées seront visibles dans la liste des annonces et les détails de l’annonce que l’on sélectionne.

Pour implémenter l’upload d’images, nous avons utilisé le service FileUploaderService que nous avons trouvé ici : <https://github.com/coderberry/FileUploader>

Malheureusement, par manque de temps, nous n’avons pas pu optimiser la gestion des illustrations en l’intégrant directement lors de la création de l’annonce.

# API REST

## Les ressources utilisées

### USER

Ressources :

* Individuelles : user
* Collection : users

### SALEAD

Ressources :

* Individuelles : saleAd
* Collection : saleAds

## Tableaux des actions gérées sur les ressources

### USER

|  |  |  |
| --- | --- | --- |
| **URI** | **Description** | **Réponse HTTP** |
| GET  http://server/users | Renvoie la liste de tous les utilisateurs | 200 OK  404 = ressource demandée non trouvable |
| POST  http://server/users | Crée un utilisateur | 201= la ressource a été créée  400 = requête mal formulée (username existant ou paramètre manquant)  404 = rôle non trouvé |
| GET  http://server/user/id | Renvoie les informations d’un utilisateur à partir de son id | 200 OK  400 = requête mal faite (il manque des paramètres)  404 = ressource demandée non trouvable  405 = méthode non acceptée (fonction non gérée)  406 = not acceptable (format non valide) |
| PATCH  http://server/user/id | Met à jour les informations d’un utilisateur | 204 No Content  400 = requête mal faite (il manque des paramètres)  404 = ressource demandée non trouvable  405 = méthode non accepté (fonction non gérée) |
| DELETE  http://server/user/id | Supprime un utilisateur | 204 No Content  400 = requête mal faite (il manque des paramètres)  404 = ressource demandée non trouvable  405 = méthode non acceptée (fonction non gérée) |

### SALEAD

|  |  |  |
| --- | --- | --- |
| **URI** | **Description** | **Réponse HTTP** |
| GET  http://server/saleAds | Renvoie la liste de toutes les annonces | 200 OK  404 = ressource demandée non trouvable |
| POST  http://server/saleAds | Crée une annonce | 201= la ressource a été créée  400 = requête mal formulée (le prix n’est pas un nombre ou paramètre manquant)  404 = auteur non trouvé |
| GET  http://server/saleAd/id | Renvoie les informations d’une annonce à partir de son id | 200 OK  400 = requête mal faite (il manque des paramètres)  404 = ressource demandée non trouvable  405 = méthode non acceptée (fonction non gérée)  406=not acceptable (format non valide) |
| PATCH  http://server/ saleAd/id | Met à jour les informations d’une annonce | 204 No Content  400 = requête mal faite (il manque des paramètres)  404 = ressource demandée non trouvable  405 = méthode non acceptée (fonction non gérée) |
| DELETE  http://server/ saleAd/id | Supprime une annonce | 200 OK  400 = requête mal faite (il manque des paramètres)  404 = ressource demandée non trouvable  405 = méthode non acceptée (fonction non gérée) |

RESULTATS :

Les résultats de ces requêtes sont disponibles sur POSTMAN dans le dossier Lecoincoin.