

**DOCUMENTATION DEVELOPPEUR**

**Attribution des places de parking**

TRAN DIEP Mai Thi – Melvin REDUREAU

Table des matières

[I. Contexte : 3](#_Toc70819327)

[II. Versionning : 4](#_Toc70819328)

[III. Base de données : 5](#_Toc70819329)

[IV. Spécifications fonctionnelles : 6](#_Toc70819330)

[1. Partie de connexion : 6](#_Toc70819331)

[a. Page de connexion : 6](#_Toc70819332)

[b. Page « Mot de passe oublié » : 6](#_Toc70819333)

[c. Page « Réinitialiser mot de passe » 6](#_Toc70819334)

[d. Page d’inscription : 7](#_Toc70819335)

[2. Partie d’utilisateur : 8](#_Toc70819336)

[a. Page d’accueil : 8](#_Toc70819337)

[b. Page « Faire une réservation » : 8](#_Toc70819338)

[3. Partie d’administrateur : 10](#_Toc70819339)

[a. Page d’accueil : 10](#_Toc70819340)

[b. Page « Liste des statuts » : 11](#_Toc70819341)

[c. Page « Liste des utilisateurs » : 11](#_Toc70819342)

[d. Modal « Modifier un utilisateur » : 12](#_Toc70819343)

[e. Modal « Créer utilisateur » : 13](#_Toc70819344)

[f. Page « Liste des places » : 13](#_Toc70819345)

[g. Modal « Créer place » : 14](#_Toc70819346)

[h. Modal « Modifier une place» : 14](#_Toc70819347)

[i. Page « Liste des réservation » : 15](#_Toc70819348)

[j. Page « Liste d’attente » : 15](#_Toc70819349)

[k. Modal « Modifier rang d’attente d’un utilisateur » 16](#_Toc70819350)

[V. Composants logiciels : 17](#_Toc70819351)

[VI. Testes unitaires : 19](#_Toc70819352)

[1. Test de connexion avec un mauvais mail et ou mot de passe 19](#_Toc70819353)

[2. Test de vérification des champs 19](#_Toc70819354)

[3. Test si l’utilisateur a déjà réservé une place 20](#_Toc70819355)

# Contexte :

Le projet Parking est une application de gestion des places d’un parking. Elle permet aux utilisateurs de réserver une place du parking pour pouvoir s’y garer. Toutefois, avant de pouvoir utiliser l’application, les utilisateurs doivent s’inscrire via un formulaire. Leur inscription sera ensuite validée ou non par l’administrateur de l’application.

L’administrateur a la capacité de gérer manuellement les places de parking en changeant par exemple leur nom, leur attribution ainsi que leur réservation. Il peut aussi ajouter ou supprimer une place.

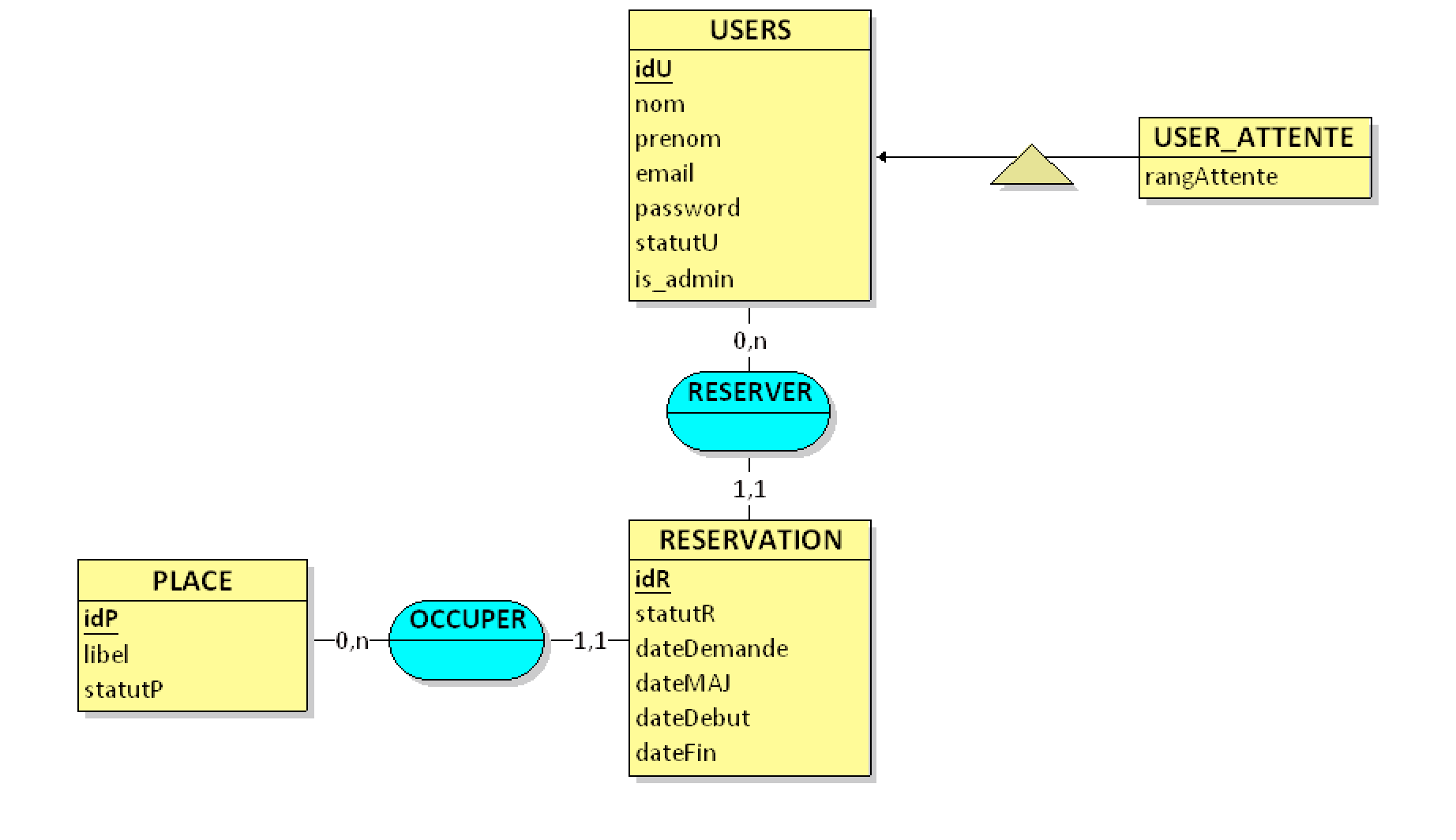
# Versionning :

Afin de sauvegarder toute modification effectuée sur le code, nous avons utilisé l’outil Git, qui permet d’envoyer le projet et de sauvegarder toute modification sur le repository GitHub. Grâce au repository GitHub et git, nous pouvons annuler toute modification et revenir en arrière lorsqu’un bug se produit, ou une fonctionnalité n’est pas voulue.

Il est possible d’y accéder par ce lien : <https://github.com/maithi-trandiep/Parking>

# Base de données :

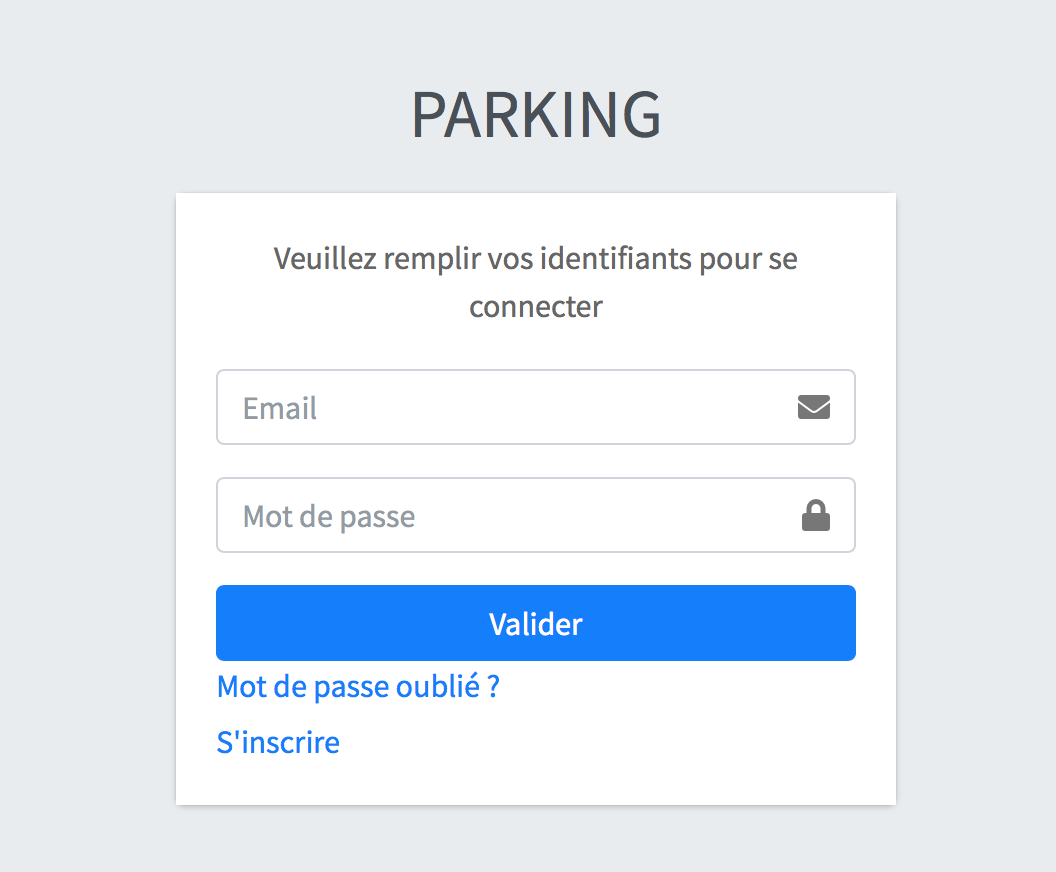
Le modèle conceptuel des données



# Spécifications fonctionnelles :

## Partie de connexion :

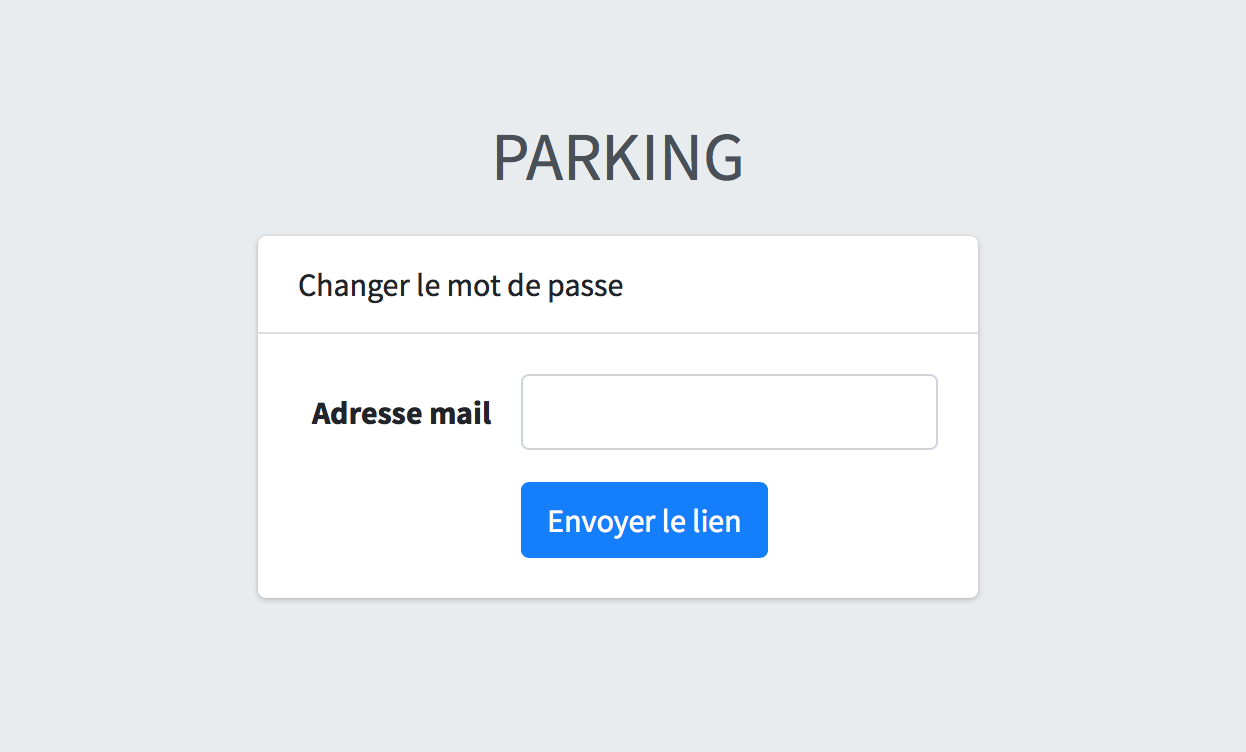
### Page de connexion :



La page de connexion comporte les champs identifiants (qui est l’email utilisé pour s’inscrire) et mot de passe (choisi par l’utilisateur).

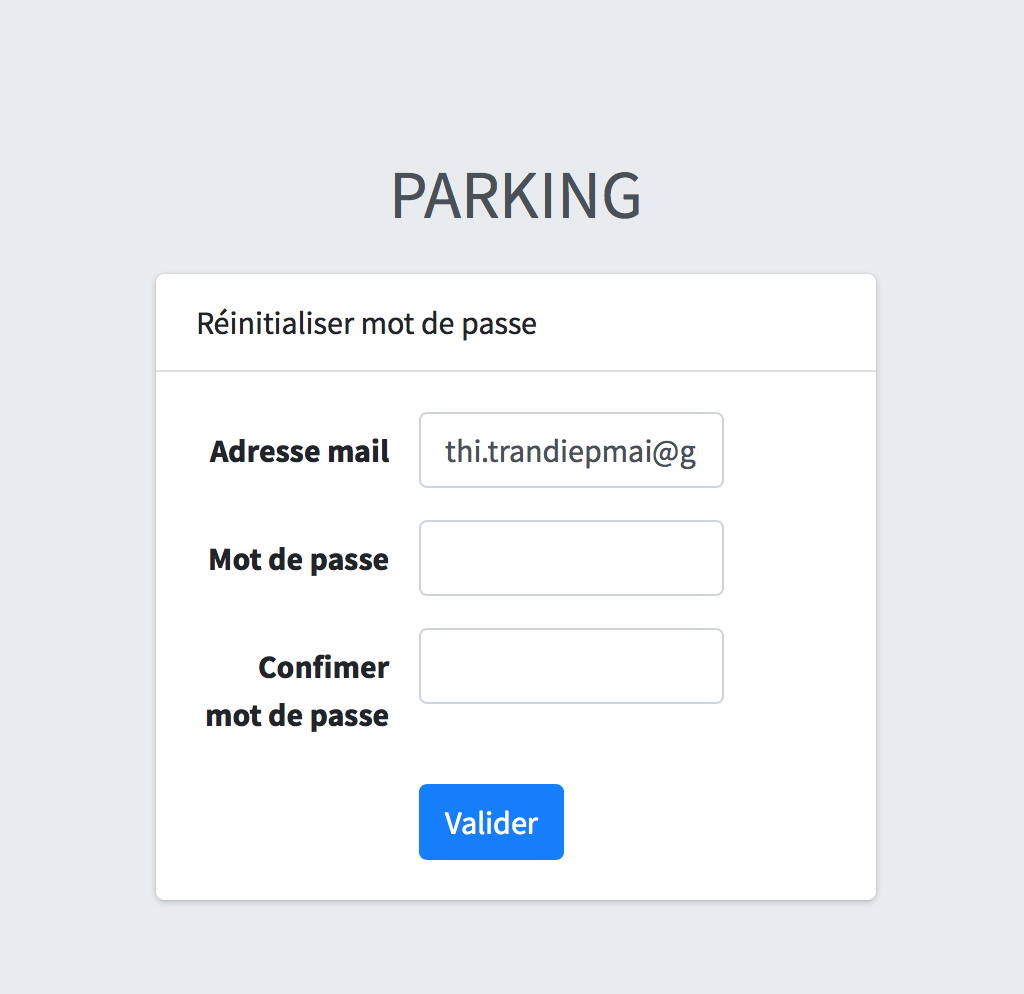
De plus, on y trouve aussi les options « Mot de passe oublié ? » envoyant sur une page de changement de mot de passe et « S’inscrire » permettant d’accéder au formulaire d’inscription.

### Page « Mot de passe oublié » :

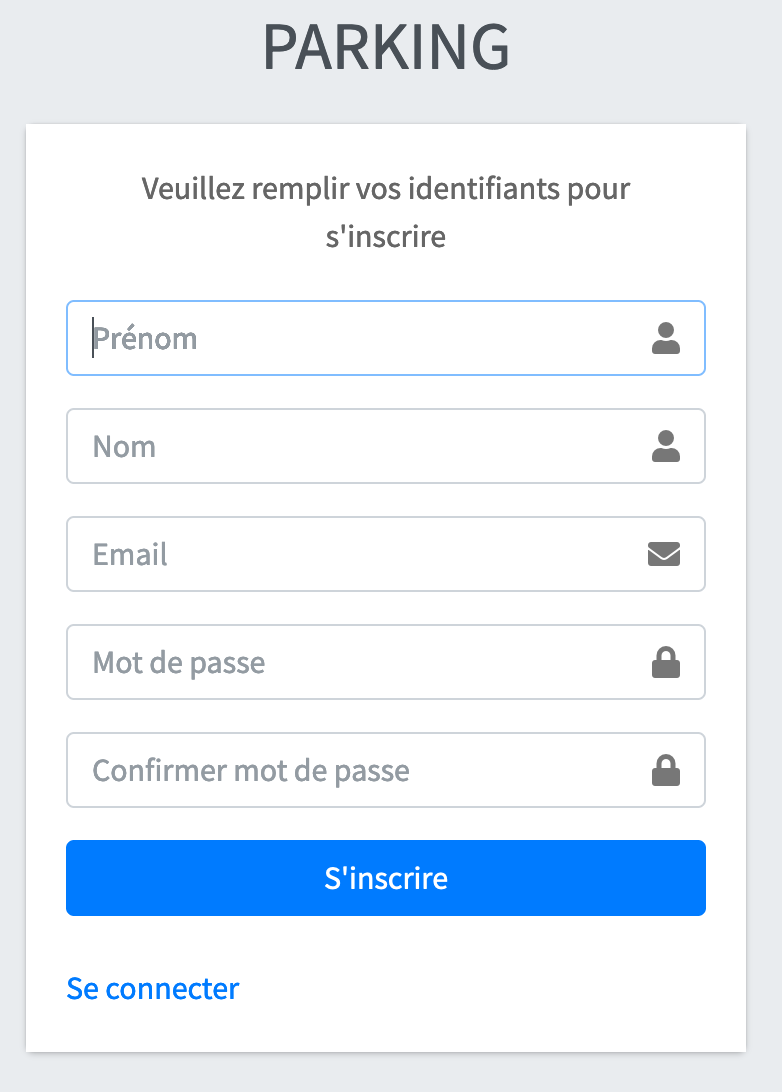


Cette page permet de récupérer son mot de passe en envoyant un lien par mail. L’email spécifié dans le champ est celle qui a servie à l’inscription à l’application.

### Page « Réinitialiser mot de passe »



### Page d’inscription :

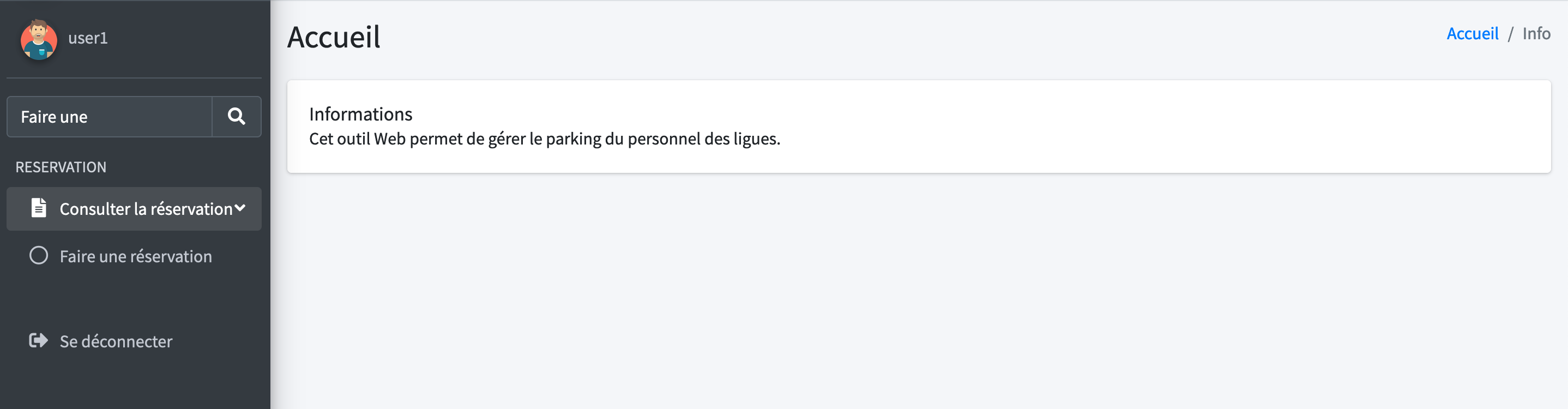


Cette page est un formulaire permettant l’inscription à l’application. Le formulaire comporte les champs Prénom, Nom, E-mail, Mot de passe et Confirmer mot de passe. Tous les champs sont obligatoires.

Au bas du formulaire se trouve aussi l’option se connecter qui renvoie à la page de connexion.

## Partie d’utilisateur :

### Page d’accueil :



La page d’accueil utilisateur comporte un menu constitué de deux entrées :

* Consulter la réservation un sous menu comprenant un lien « Faire une réservation » qui envoie sur la page permettant de réserver, annuler et consulter une réservation.
* Se déconnecter qui sert à quitter l’application tout en fermant la session.

Au-dessus du menu se trouve un champ de recherche permettant de chercher une entrée dans le menu.

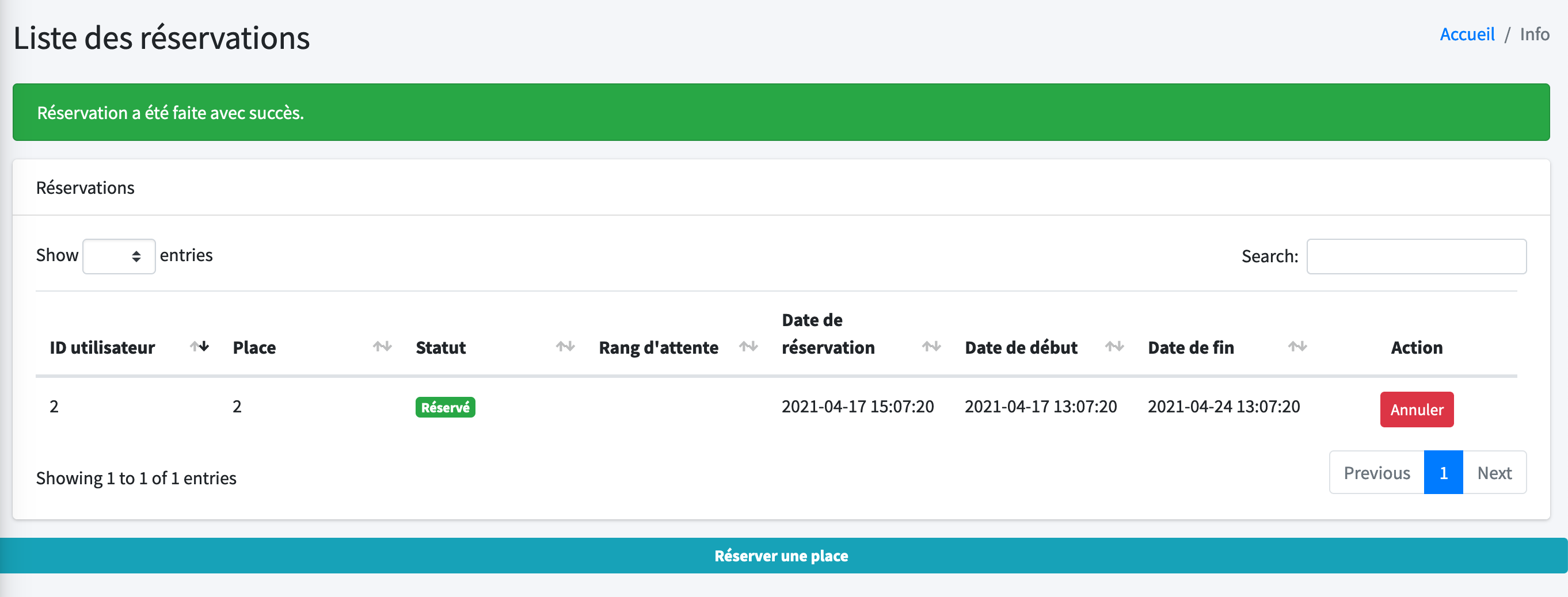
### Page « Faire une réservation » :

Une image contenant texte

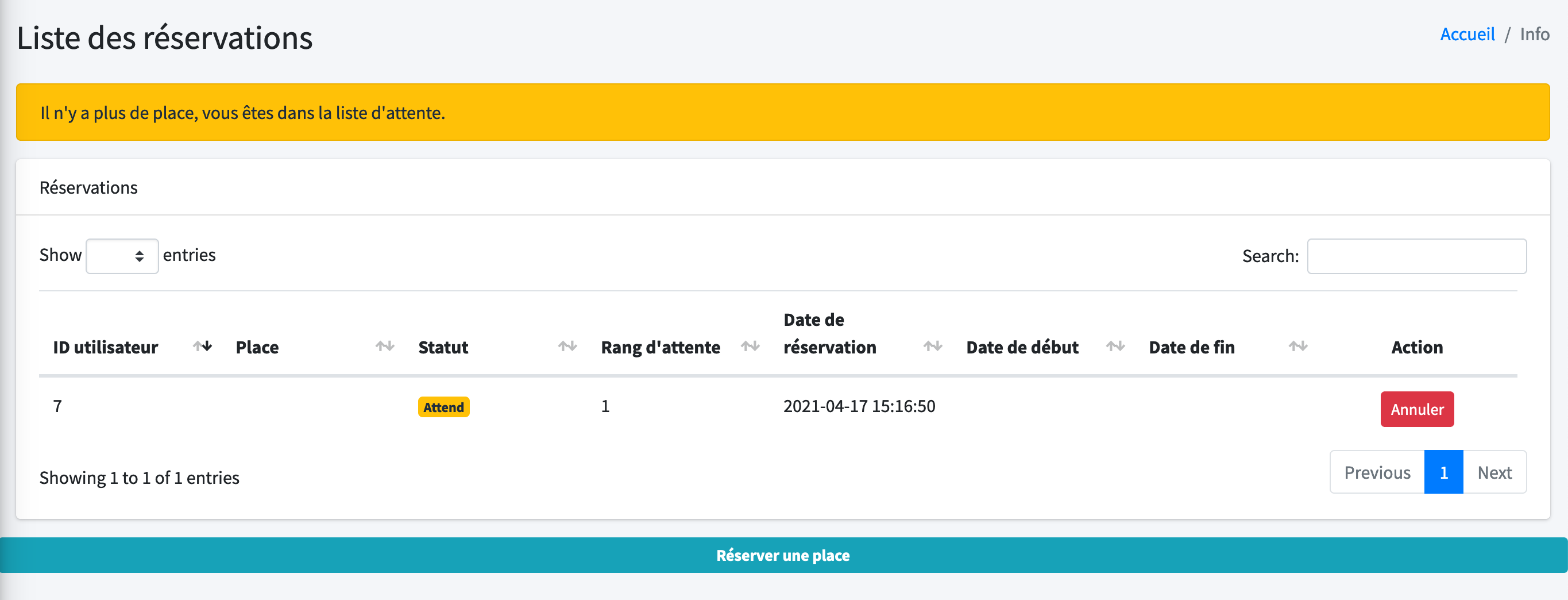
Description générée automatiquement

Cette page permet d’effectuer une réservation. Pour cela, on doit cliquer sur le bouton « Réserver une place ».

Il sera alors attribué à l’utilisateur une place aléatoire parmi les places libres.



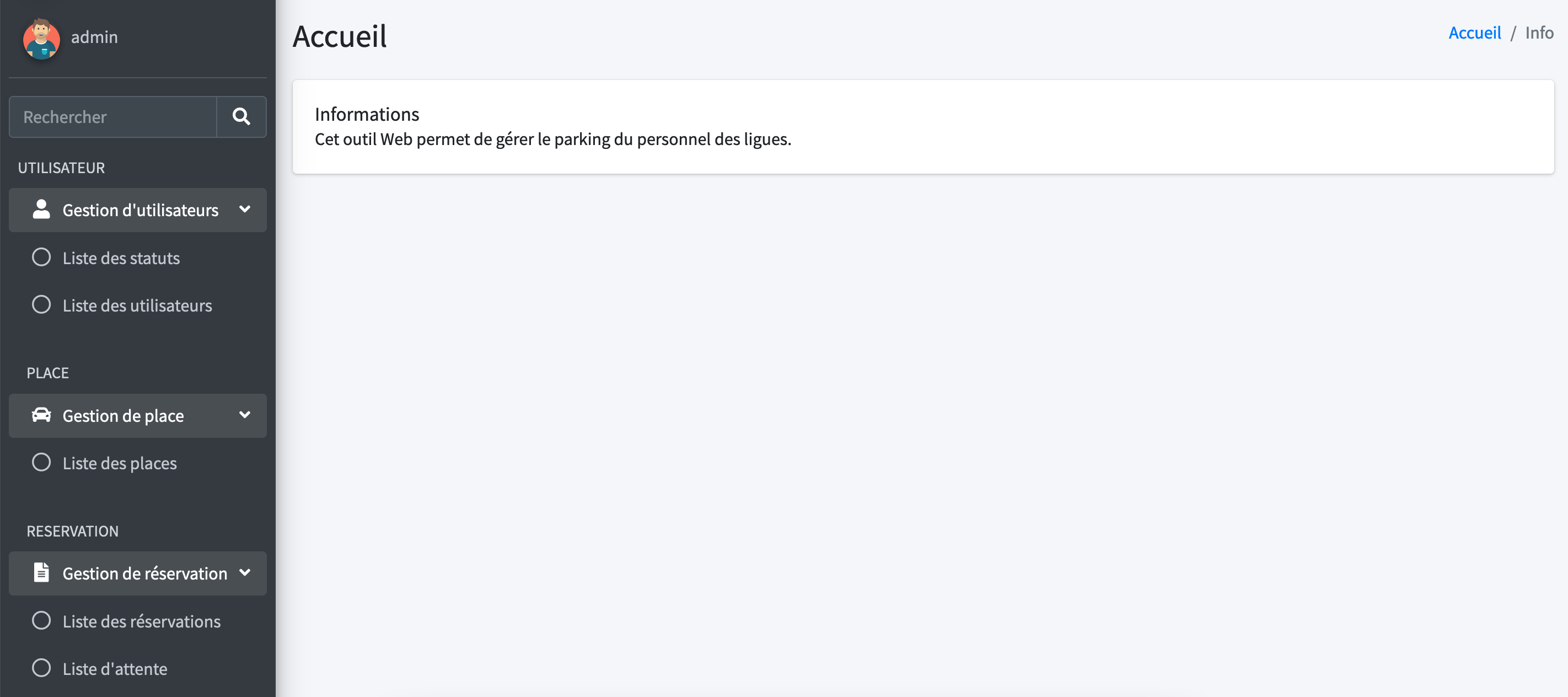
La durée de la réservation est de 7 jours, cette durée est fixe et ne peut être changée mais l’utilisateur peut à tout moment annuler la réservation si la place n’est au final pas utilisée pendant les 7 jours. Le numéro de la place est affiché dans la colonne place, il correspond à la numérotation du parking.



Si toutes les places sont occupées, la réservation est faite, l’utilisateur sera mis en attente le temps qu’une place se libère. Son rang dans la file d’attente est affiché dans la colonne « Rang d’attente ». L’utilisateur peut aussi annuler sa réservation.

## Partie d’administrateur :

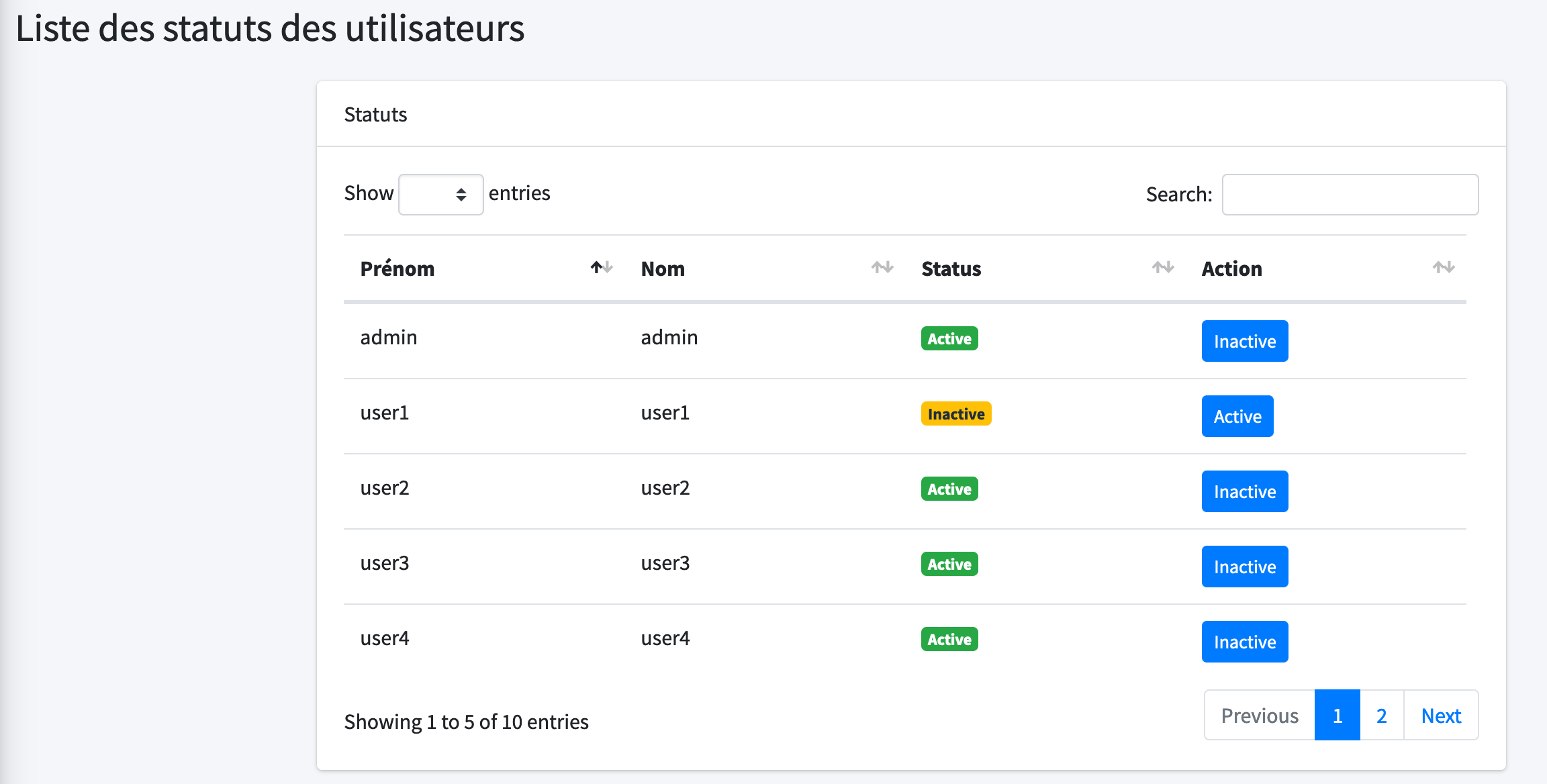
### Page d’accueil :



La page d’accueil de l’administrateur se présente comme la page d’accueil de l’utilisateur, la seule différence étant le nombre de sous menu. La page d’accueil administrateur comporte donc les sous menu et items.

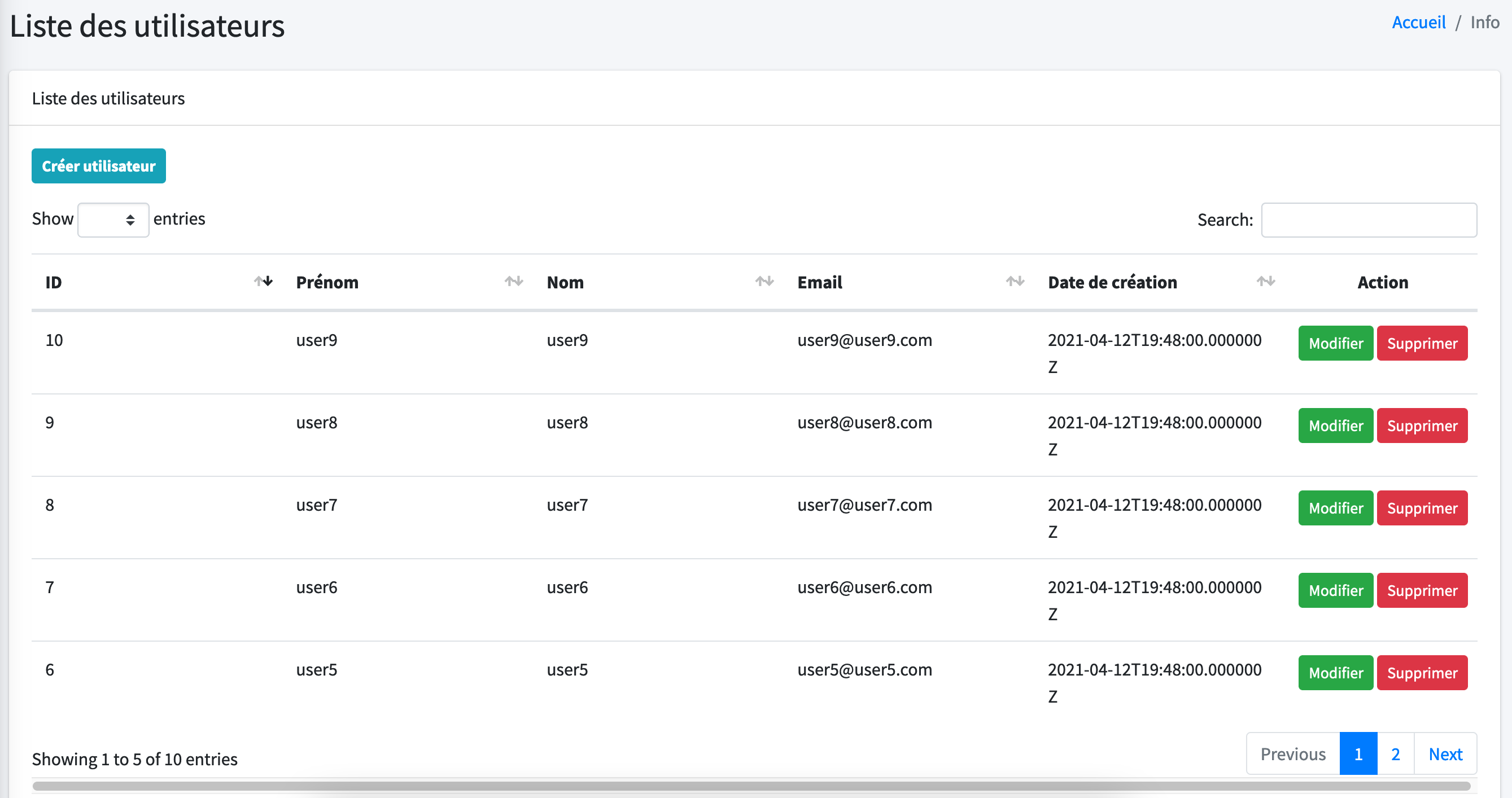
* Gestion d’utilisateurs :
  + Liste des statuts : renvoie vers la page d’activation ou de désactivation d’un compte utilisateur.
  + Liste des utilisateurs : renvoie vers la liste des utilisateurs, cette page sert aussi à créer, modifier ou supprimer un utilisateur ou son compte.
* Gestion de place
  + Liste des places : renvoie vers la liste des places du parking, cette page sert aussi à créer, modifier ou supprimer une place.
* Gestion de réservation
  + Liste des réservations : renvoie vers la liste des réservations en cours, cette page sert aussi à annuler une réservation.
  + Liste d’attente : renvoie vers la liste des réservations en attente, cette page sert aussi à annuler une réservation dans la liste d’attente.

### Page « Liste des statuts » :



Cette page permet de consulter la liste de compte actifs ou inactifs. Les boutons « Active » permettent d’activer un compte alors que les boutons « Inactive » permettent eux de désactiver un compte. Le champs « Search » permet de chercher un compte dans la liste.

### Page « Liste des utilisateurs » :



Cette page permet de consulter les différents utilisateurs de l’application. Sur cette page, on peut aussi modifier, supprimer ou bien créer un utilisateur grâce aux boutons correspondants.

### Modal « Modifier un utilisateur » :

Une image contenant texte

Description générée automatiquement

La modification d’un utilisateur ouvre une un formulaire pré remplit et modifiable, l’admin ne peut cependant pas modifier le mot de passe de l’utilisateur. Pour valider les changements, cliquez sur le bouton « Modifier », pour les annuler, cliquez sur le bouton « Fermer ».

### Modal « Créer utilisateur » :

Une image contenant texte

Description générée automatiquement

La création d’un utilisateur depuis la liste des utilisateur ouvre un formulaire à remplir permettant de créer un utilisateur. Il contient les champs « Prénom », « Nom », « E-mail » et « Password » pour le mot de passe.

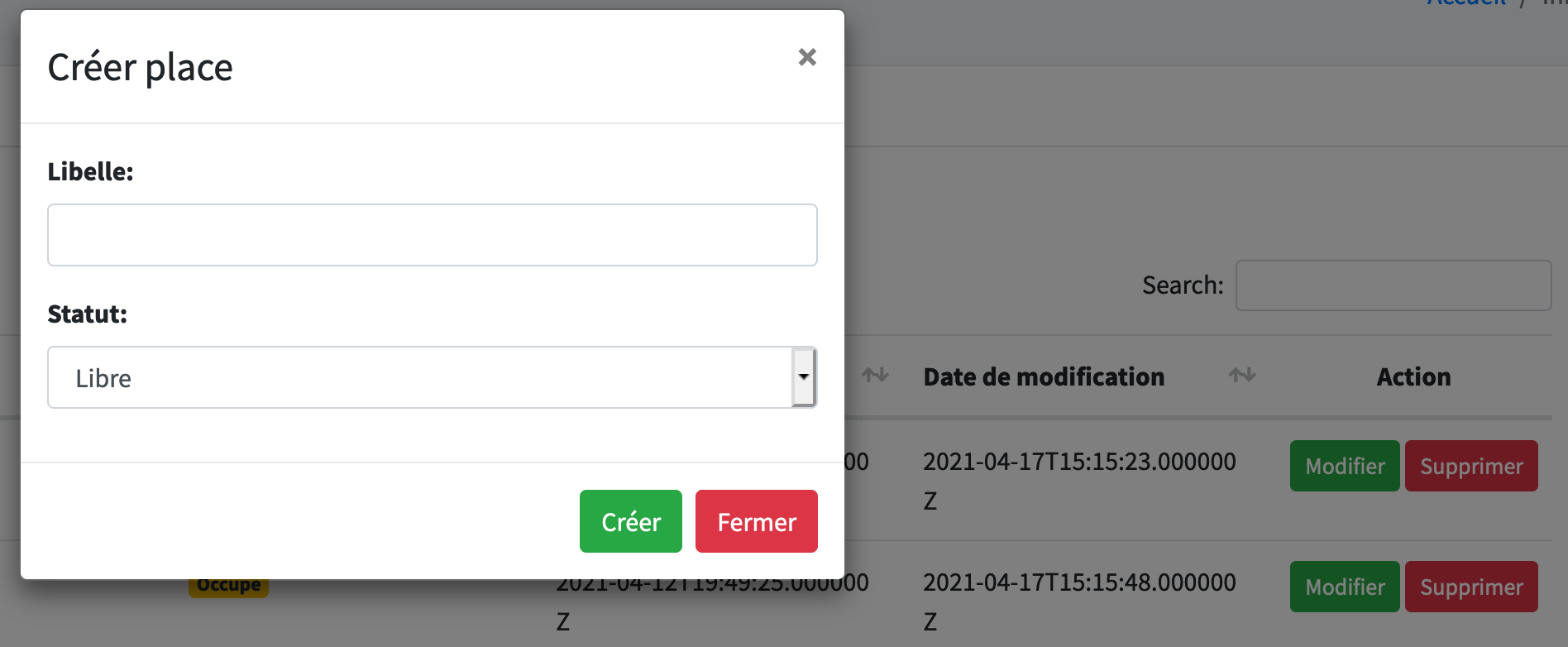
### Page « Liste des places » :

Une image contenant table

Description générée automatiquement

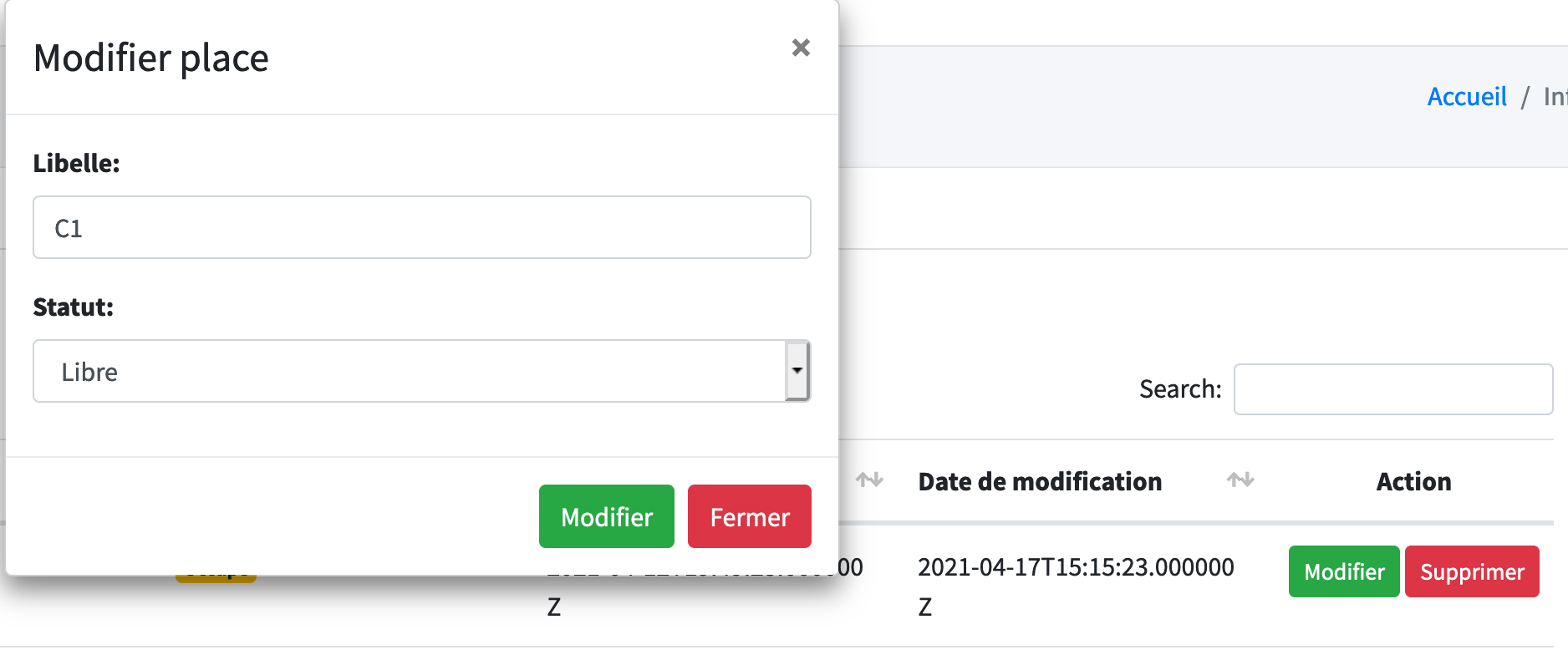
Cette page permet la consultation des places ainsi que leur création, modification ou suppression. Elle présente aussi le statut de la place, son libellé ainsi que la date de création et de modification de la place.

### Modal « Créer place » :



Cette page affiche un formulaire permettant créer une place avec son libellé et un statut de base.

### Modal « Modifier une place» :



Cette page permet la modification du libelle et du statut d’une place. On valide les changements grâce au bouton valider du formulaire.

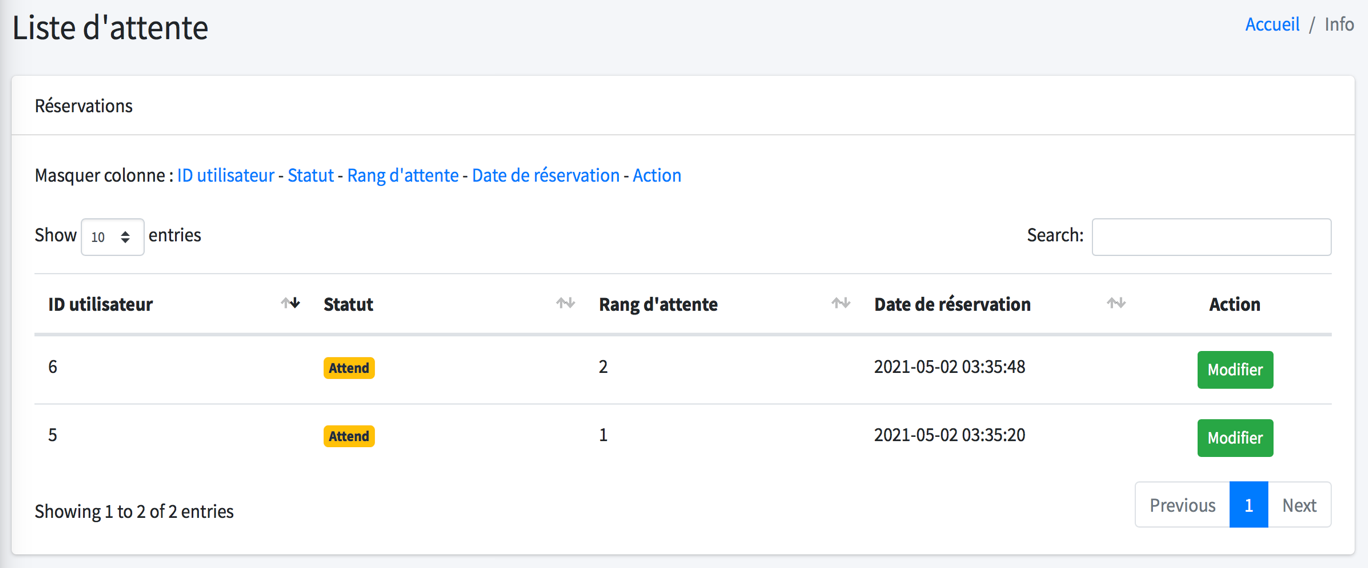
### Page « Liste des réservation » :

Une image contenant table

Description générée automatiquement

Cette page permet la consultation des différentes réservations ainsi que de leur statut. On peut aussi annuler une réservation en cours ou en attente. Les id des places et des utilisateurs les ayants réservés sont affiché de même pour la date de réservation la date de début et la date de fin de la réservation.

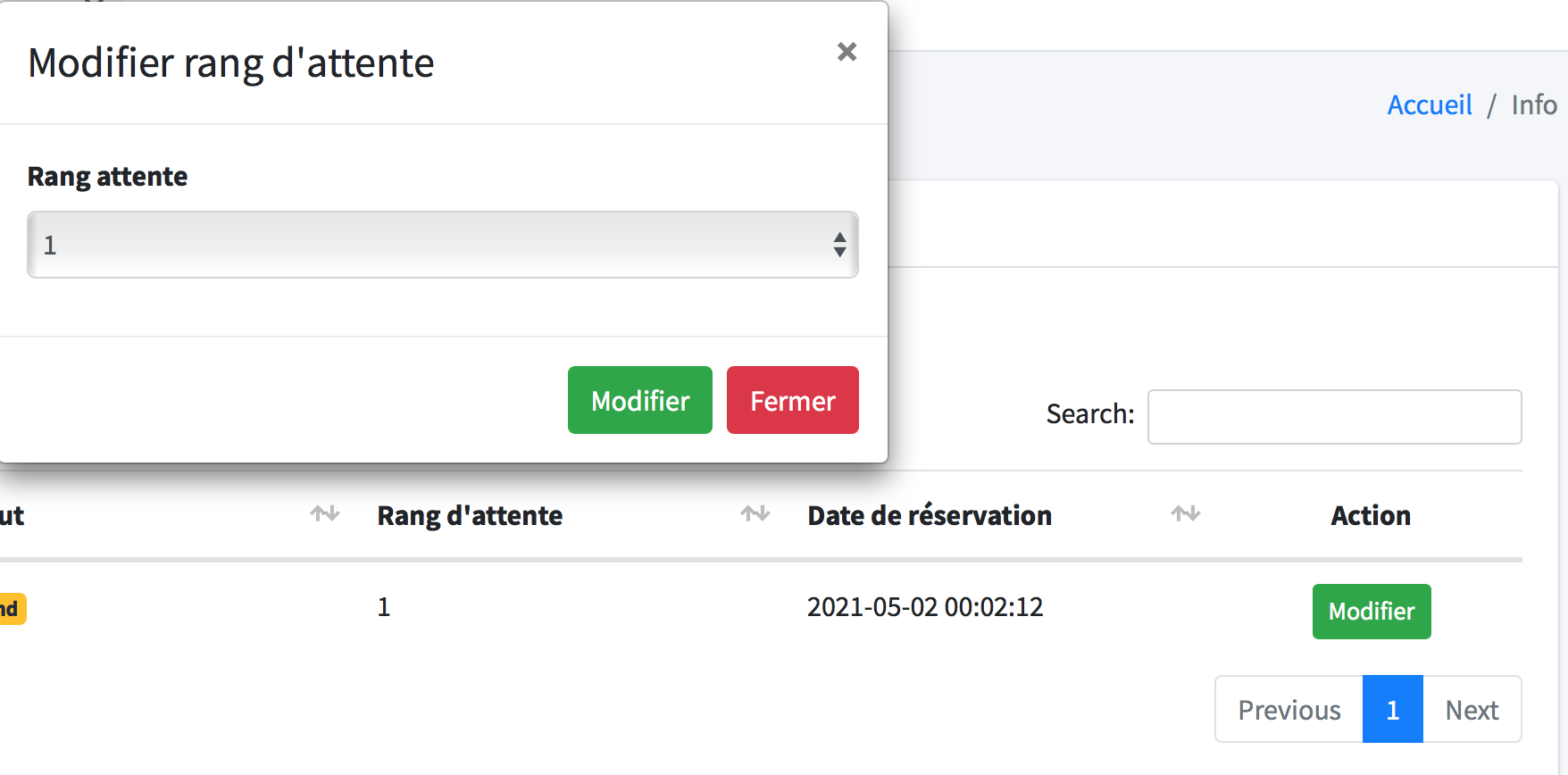
### Page « Liste d’attente » :



Cette page permet la consultation de la liste d’attente des réservations ainsi que l’id des utilisateurs qui ont fait les réservations. On peut modifier le rang d’attente d’une réservation en attente.

*ATTENTION : Les places libérées seront automatiquement attribuées au premier utilisateur de la liste d’attente.*

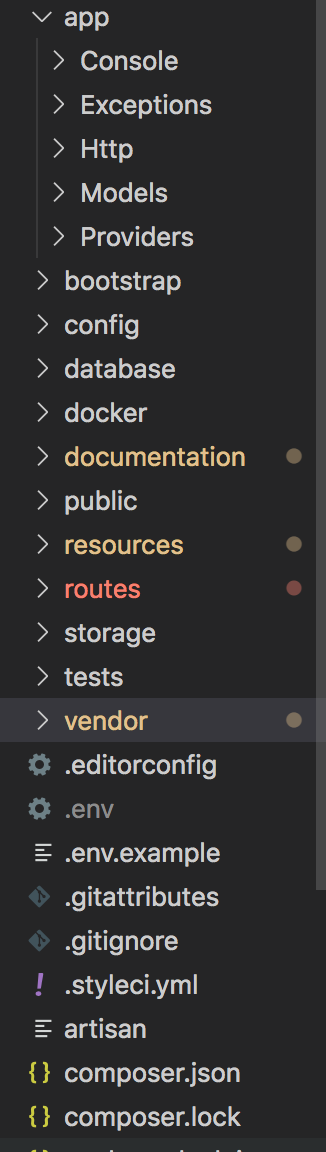
### Modal « Modifier rang d’attente d’un utilisateur »



Cela permet à choisir un rang dans la liste de rang et désigner à un utilisateur.

# Composants logiciels :

L’application est codée sous Laravel 8 avec Bootstrap et le plugin jQuery Datatables, HTML/CSS



* Le dossier templates dans le dossier public contient tous les fichiers d’une template admin bootstrap permettant de mettre en forme visuellement le site.
* Le dossiers database permet la gestion de la base données, il contient les fichiers migrations et seeders pour créer la base de données.
* Le dossier resources contient toutes les vues (incluant les sub-vues) c’est à dire toute la partie visuelle du site, comme les pages de connexion, la liste des utilisateurs et leurs statuts, la liste des places, la liste des réservations, la liste d’attentes, etc.
* Les vues communiquent avec les fichiers controllers et models pour effectuer du traitement et envoyer ou recevoir des données. C’est le cœur du fonctionnement du site.
* Le dossier routes fait le lien entre l’URL entré par l’utilisateur et le code qui sera exécuté.
* Le dossier vendor contient toutes les librairies utilisées pour le site, en particulier DataTables (un plug-in jQuery open-source permettant de dynamiser un tableau HTML).
* Le fichier .env est le fichier à éditer lors de la mise en place du projet sur une autre base de données, il faut alors modifier : DB\_CONNECTION, DB\_HOST, DB\_PORT, DB\_DATABASE, DB\_USERNAME, DB\_PASSWORD avec vos informations de connexions.

\*Extrait d’une partie de code :

Voici la fonction qui permet à modifier le rang d’attente d’un utilisateur.

# Testes unitaires :

## Test de connexion avec un mauvais mail et ou mot de passe

## Test de vérification des champs

## Test si l’utilisateur a déjà réservé une place