

***Rapport Application Android***

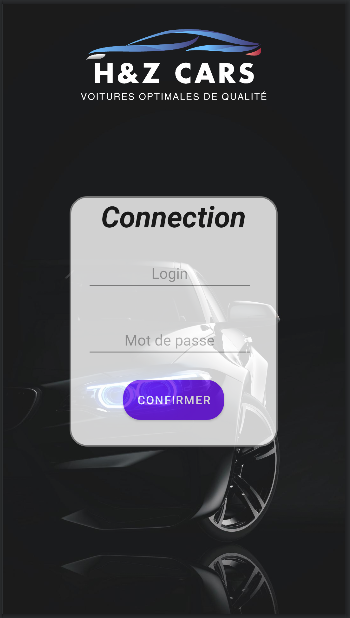
H&Z Cars est une application dédiée à la gestion d'une entreprise de location des voitures. Cette application permet à l'utilisateur d'effectuer des opérations CRUD sur les clients ainsi que sur les voitures.

***Sommaire :***

* Revue détaillée sur le fonctionnement générale de l’application pp 2-12.
* Mise en liste des classes java utilisées pp 13-14.
* Navigation pp 15.

La première interface de l'application permet à l'utilisateur de se connecter via un identifiant et un mot de passe (Login : “admin”, Mot de Passe : “admin”).

NB. : Pour des raisons de sécurité, le Sign Up (Inscription) n’était pas pris en compte, l’utilisateur de l’application est un Super User qui gère tout.



Après avoir connecté, un Dashboard est affiché : C’est le menu principal de l’application. Il contient deux rubriques : “Gestion Clients” et “Gestion Voitures”.

La première permet d'effectuer les opérations CRUD sur les clients. Quant à la deuxième, elle est spécifique aux voitures.



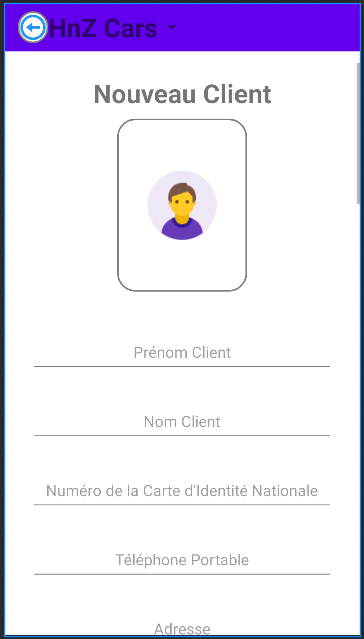
1. Dans le premier cas, il existe quatre propositions. Celles relatives aux opérations CRUD :

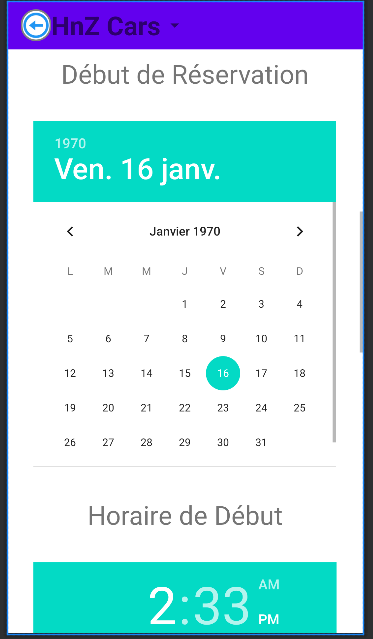


Les trois rubriques : “Ajouter”, “Supprimer” et “Liste” permettent respectivement l’ajout, la suppression et l’affichage d’un ou de plusieurs clients. L’opération de modification d’un client peut être effectuée directement sur la liste des clients en choisissant le client cible.

1. Rubrique “Ajouter” :

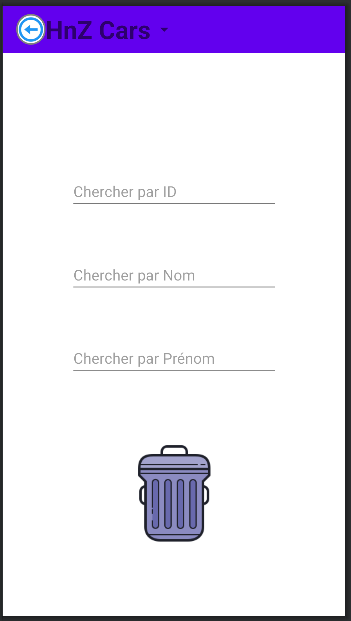
Elle permet d’ajouter et enregistrer les différentes informations relatives à un client :





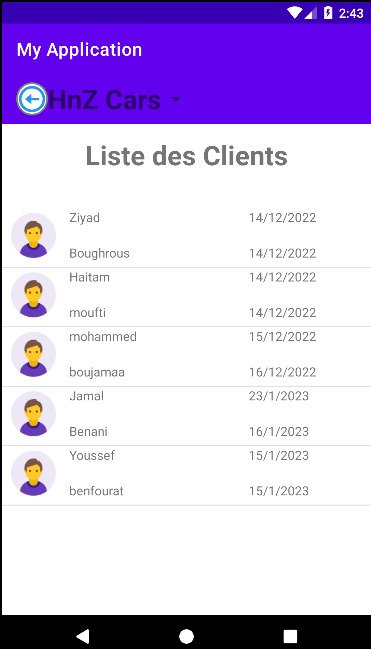
1. Rubrique “Supprimer” :

Elle est spécifique à la suppression d’un client. Pour ce faire, ce dernier peut être “mappé” par son ID, son Nom ou par son Prénom :



1. Rubrique “Liste” :

C’est là où on affiche la liste des clients qui sont présents dans la base de données. Chaque client de la liste apparaît avec son nom, son prénom, la date de début de l’emprunt de la voiture et la date de fin de cet emprunt. En cliquant sur un Item de la liste, la page de la modification du client en question est affichée avec ses informations actuelles permettant ainsi une mise à jour du client.



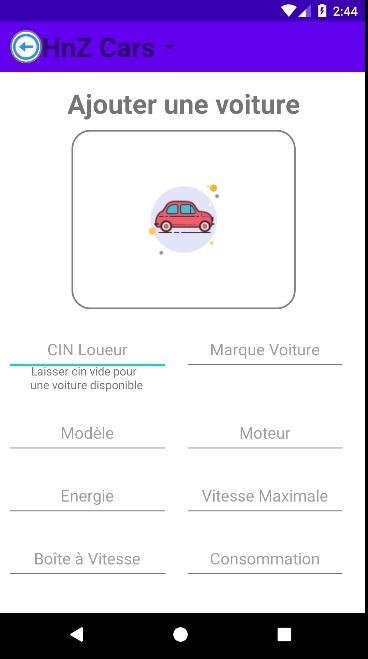
1. De même, dans le second cas, il existe quatre propositions. Celles relatives aux opérations CRUD :



1. Ajouter une Voiture :

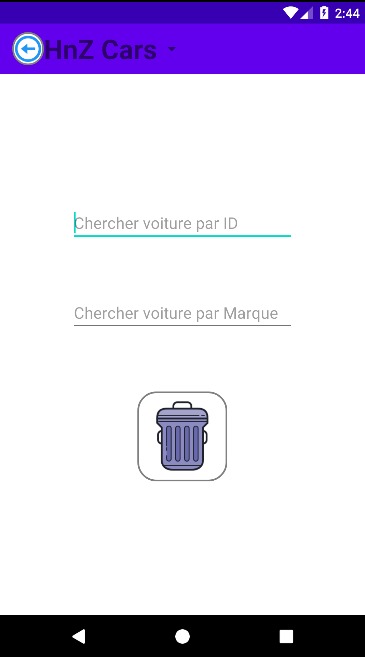
La particularité avec l’ajout d’une voiture par rapport à l’ajout d’un client, c’est que celle-ci comprend trois cas possibles :

* La voiture peut être ajoutée sans “CIN loueur”, i.e. sans mentionner la personne qui a louée cette voiture. Dans ce cas, on dit que c’est une voiture disponible.
* La voiture peut être ajoutée avec une “CIN loueur”, i.e. en mentionnant le loueur de la voiture. Avant d’ajouter une telle voiture, une vérification de la validité/existence de la CIN saisie sera effectuée. Dans ce cas, on dit que c’est une voiture louée.
* Dans le cas où la “CIN loueur” saisie ne correspond à aucune CIN d’un des clients présents dans la base de données, on dit que c’est une CIN invalide et la voiture ne sera pas ajoutée.



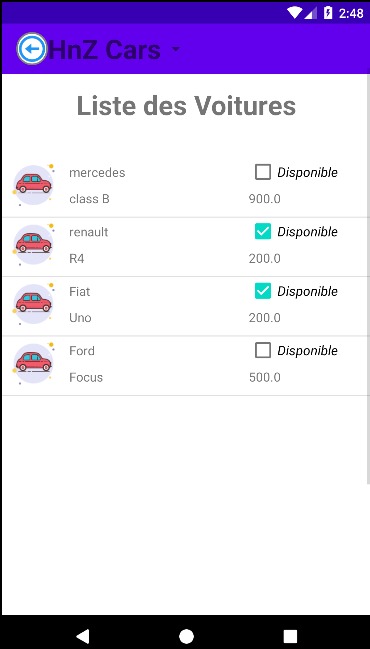
1. Supprimer une voiture :

C’est exactement la même chose qu’on a fait avec les clients, sauf que, dans ce cas, une voiture est identifiée par son ID ou par sa marque.



1. Listes des voitures :

C’est la même approche qu’en Liste des clients, sauf que, pour tout Item, ce sont la marque, le modèle, la disponibilité et le prix par jour de la voiture qui sont affichés.



***Classes Java utilisées :***

* Concernant les clients :

Pour l’ajout d’un client : “Add\_Clt.java”.

Pour la suppression d’un client : “Delete\_Clt.java”.

Pour la modification d’un client : “Modif\_Clt.java”.

Pour l’affichage des clients : “Liste.java”.

L’adaptateur de la liste des clients est contenu dans : “Customize\_Adapter.java”.

* Concernant les voitures :

Pour l’ajout d’une voiture : “Add\_Voiture.java”.

Pour la suppression d’une voiture : “Delete\_Voiture.java”.

Pour la modification d’une voiture : “Modif\_Voiture.java”.

Pour l’affichage des voitures : “Liste\_V.java”.

L’adaptateur de la liste des voitures est contenu dans : “Customize\_Adapter\_V.java”.

* Classes Java Bean utilisées :

1. Client.java
2. Voiture.java

* La Classe “DatabaseHandler.java” contient les déclarations et les fonctions permettant la création, la mise à jour et la destruction de la base de données, des tables Clients, Voitures et Users..ainsi que les fonctions relatives aux opérations CRUD relatives à chaque entité avec d’autres fonctions utiles.
* Les Classes “My\_Dashboard.java” et “MainActivity.java” ont pour fonctions respectivement de contrôler les layouts “my\_dashboard.xml” et “activity\_main.xml”.

**Navigation :**

Pour permettre à l’utilisateur de l’application une bonne expérience de navigation, un Spinner a été ajouté avec deux options selon le besoin : “Menu Principal” pour revenir à “My Dashboard” et “Se déconnecter” pour pouvoir quitter la session en cours.

Pour atteindre le même objectif, un bouton de retour en arrière a été ajout.