

# Développement Logiciel pour Mobile

## Rapport du 2ème Rendu

### Avancement et Architecture :

Nous travaillons actuellement sur le projet Rapace.

Pour exécuter notre code il vous faut Android Studio et lancer notre projet.

Nous avons la visualisation de notre Base de données à l'adresse suivante :

<http://vps814672.ovh.net/phpmyadmin/> avec l'identifiant : admin et le mot de passe : admin et pour se connecter à la base de données MySql sur phpmyadmin, identifiant : rapace et mdp : rapace.

Pour les détails de la base de données, reportez-vous au compte rendu 1 de notre projet.

Vous pouvez toujours utiliser les deux utilisateurs pré-rentés en base qui sont :

- identifiant : floran
- mdp : floran7

et :

- identifiant : jjj
- mdp : jjj

Vous pouvez tester notre application et ajouter des utilisateurs via l'enregistrement ou vous connecter avec ceux là ou ceux créés.

Nous avons ajouté plusieurs fonctionnalités à notre projet.

La toute première concerne le système d'affichage de nos différents locaux ainsi que de nos différentes caméras d'un local. Nous avons opté pour l'utilisation d'un "RecyclerView" capable de gérer des "CardView" (petites images cliquables sous forme de liste).

Voici un exemple de CardView représentant nos locaux :

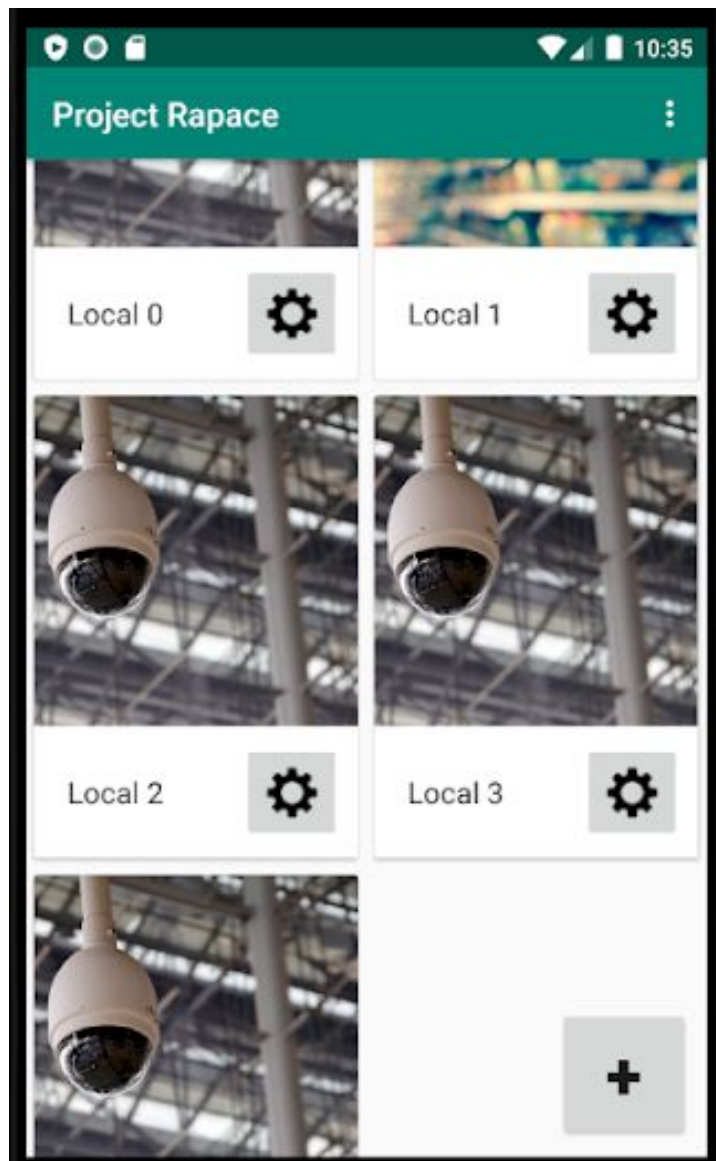
En appuyant sur le bouton "+" vous pouvez ajouter aisément un local supplémentaire. Comme vous pouvez le constater en cliquant sur la roue crantée d'un local vous serez redirigé vers la vue de modification de ce local et en cliquant

sur l'image du local vous serez redirigé vers la liste des CardView des caméras de ce local, fonctionnant de la même façon qu'ici.

Nous avons donc ajouté à notre projet les classes "MainCardViewLocal" et "MainCardViewCamera" s'occupant de créer l'activité sur laquelle le "RecyclerView" gèrera l'affiche des "CardView".

Nous nous sommes également servi d'une classe "Adapter" nous permettant de faire le lien entre le "RecyclerView" et les "CardView", il reçoit une liste d'objet et se débrouille pour gérer les différentes classes que ce soit des locaux ou des caméras. L'adaptateur permet donc d'adapter la liste d'objet lui étant donné en une liste de locaux ou une liste caméras.

La classe "MyViewHolder" a été ajouté pour gérer le remplissage d'une cellule de la "CardView".



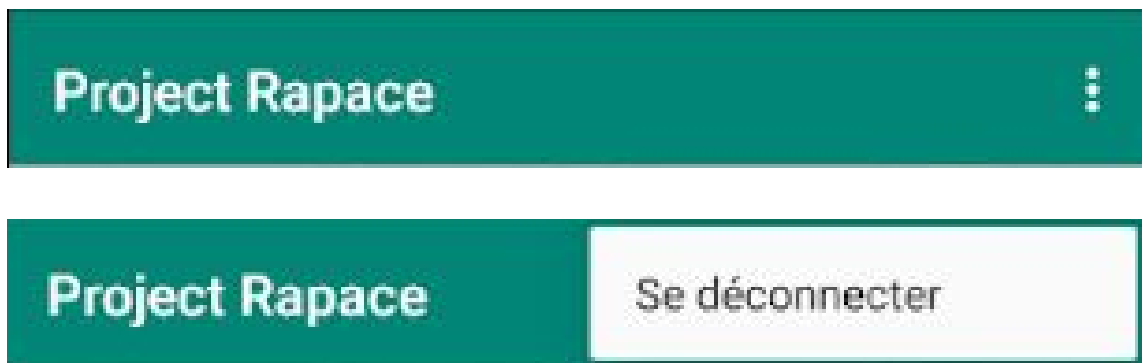
La deuxième grosse mise à jour dans notre projet, c'est la mise en oeuvre des Sessions d'utilisateurs. Nous avons utilisé des "Shared Preferences" pour simuler le système de Session, ce sont des fichiers enregistrés en local gardant en mémoire des données sous forme de dictionnaire : clé/valeur.

Nous avons donc créé la classe "SessionManager" qui gère notre système de session de sa création jusqu'à sa destruction.

Nous avons aussi ajouté la classe "RapaceService" représentant notre service qui est lancé au tout début de l'application et qui tournera tout le long de celle-ci ayant pour but de contrôler l'activité de l'utilisateur. Nous avons ajouté en BDD sur la table "users" un champs "dataAccess" et un autre "isLogged" nous permettant grâce à notre service de mettre à jour cette date d'accès à chaque activité réalisée sur l'application par l'utilisateur et de vérifier si il est inactif pendant un certain temps. Ce temps n'est fixé pour le moment qu'à une minute, donc au bout d'une minute l'utilisateur est déconnecté et s'il essaie de changer de changer d'activité il sera redirigé vers l'écran de connexion et le boolean "isLogged" sera remis à 0. Ce boolean nous permet de savoir quel utilisateur est connecté sur l'application et s'il est déconnecté qu'elle a été sa dernière connexion grâce à la date enregistrée en base de données.

Nous avons donc ajouté un bouton de déconnexion en cliquant sur n'importe activité en haut à droite sur les trois points à la vertical, ensuite vous pouvez cliquer sur le bouton vous déconnectant. Le boolean est donc mis à false, et notre Service s'occupe de déconnecter l'utilisateur et de remettre son boolean à false en BDD, puis de nettoyer la session pour la remettre à zéro en attendant un nouvelle connexion qui stockera le pseudo de l'utilisateur ainsi qu'un boolean à true et la date d'accès.

Voici la barre permettant la déconnexion :



## Maquettes et visuel de notre application :

Vue de l'interface de connexion :



The image shows a mobile application interface for 'Project Rapace'. At the top, there is a green header bar with the text 'Project Rapace'. Below the header, the title 'S'identifier' is centered. The form consists of two input fields: 'Pseudo' and 'Mot de passe'. The 'Pseudo' field contains the text 'Pseudo' and has a red underline. The 'Mot de passe' field contains the text 'Mot de passe' and has a red underline. Below the input fields, there is a button labeled 'SE CONNECTER'. Further down, there is a link that says 'Pas de compte ? Venez en créer un !'. Below this link, there is another button labeled 'S'ENREGISTRER'. The interface is displayed on a mobile device, with the status bar at the top showing the time as 11:11 and various icons.

Project Rapace

### S'identifier

Pseudo Pseudo

Mot de passe Mot de passe

SE CONNECTER

Pas de compte ? Venez en créer un !

S'ENREGISTRER

Vue de l'interface d'enregistrement :

The screenshot displays a mobile application interface for account creation. At the top, a green header bar contains the text "Project Rapace". Below this, the title "Création d'un compte" is centered. The form consists of four input fields, each with a label above it: "Pseudo", "Mot de passe", "Confirmation du mot de passe", and "Confirmation du mot de p". The "Pseudo" field is currently filled with the text "Pseudo". Below the input fields are two buttons: "VALIDER" and "RETOUR". The status bar at the top of the phone shows the time as 11:11 and various system icons.

Project Rapace

## Création d'un compte

Pseudo

Pseudo

Mot de passe

Mot de passe

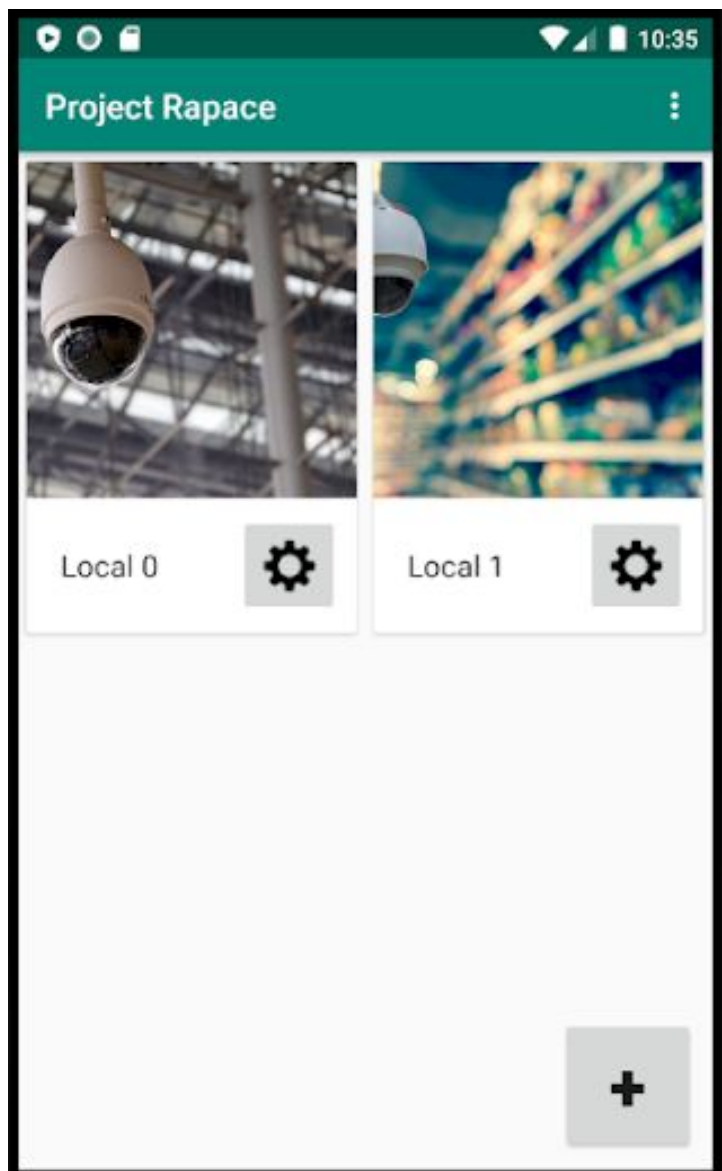
Confirmation du mot de passe

Confirmation du mot de p

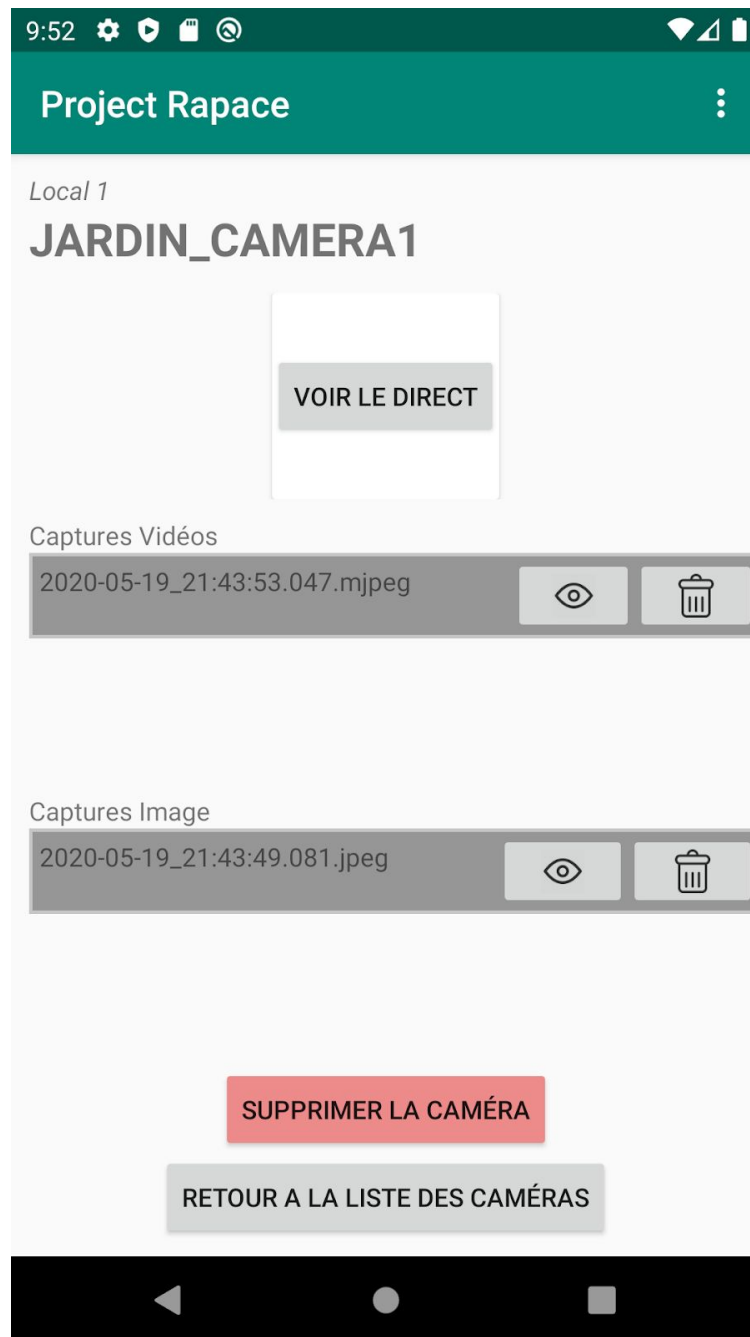
VALIDER

RETOUR

Vue de l'interface de liste "CardView" des différents locaux d'un Utilisateur, idem pour les caméras :



Vue de l'interface de gestion d'une caméra



Vue de l'affichage d'un enregistrement

