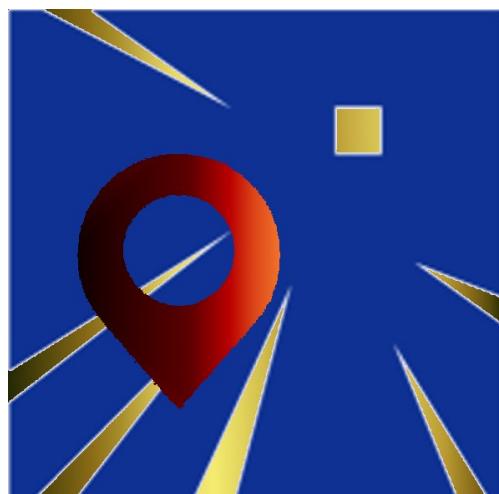


Module : I243  
Projet : Local Positioning  
System



Encadrant :  
Emmanuel BRUNO

Équipe :  
Léa GLANDUS  
Charles BIANCHERI  
Tiffany CARASCO  
Nylann LAHMAR



## Table des matières :

- Introduction
- Gestion de projet
- Conception UML
- Exigences
- Spécifications fonctionnelles
- Maquettage de l'application
- Structure du projet
- Code
- Base de données
- Rôles des membres de l'équipe
- Perspectives d'évolution
- Conclusion
- IHM et Manuel utilisateur
- Dictionnaire des termes et actions

# Introduction

Nous avons travaillé en équipe sur le projet Local Positioning System dont le but était de créer un système de localisation dans les bâtiments. Notre groupe se compose de GLANDUS Léa, BIANCHERI Charles, CARASCO Tiffany et LAHMAR Nylann.

Pour réaliser ce projet nous avons utilisé la méthode Scrum qui est la méthode agile la plus populaire afin de découper le projet et d'avancer progressivement.

Nous avons commencé par faire une phase de conception en réalisant des diagrammes de cas d'usage, de classes, de séquences et d'états. Le but étant de déterminer les fonctionnalités nécessaires, les états dans lesquels sont les objets, les séquences qui sont échangées ainsi que les classes et les liens entre elles.

Nous sommes ensuite passés à la phase d'implémentation où nous avons codé en Java8, Python et Android en n'oubliant pas de le rendre défensif, maintenable et réutilisable.

Certains outils et bibliothèques nous ont été très utiles pour réaliser ce projet :

- Pour la modélisation : plantuml
- Pour l'implantation : maven, intellij idea, android studio
- Pour la gestion de la base de données : REST et JPA
- Pour la gestion de projet : github
- Pour communiquer : Teams, Portable et Discord

# Gestion de projet

## a) Organisation

Pour réaliser ce projet nous avons créé un planning permettant de définir ce que nous devions faire chaque jour et semaine. Nous appelons ça des itérations ou sprints. Nous avons priorisé les différentes fonctionnalités à réaliser pour notre application afin de commencer par celles qui sont prioritaires et donc de finir par les moins importantes. Le niveau de priorité est déterminé tout d'abord par l'importance de la fonctionnalité pour l'utilisateur puis par le niveau de risque. Elles sont ensuite affectées d'un chiffrage représentant le temps jugé nécessaire à leur réalisation. Nous pouvons alors les répartir dans les sprints à des développeurs suivant le temps alloué au sprint pour chacun en fonction du niveau de priorité, du risque et du chiffrage.

Cas Utilisation	Fonctionnalité	Sous-fonctionnalités (si existante)	Règles de gestion (contraintes)	Priorité
Gérer un admin	En tant qu'admin je veux pouvoir me connecter dans le but d'avoir accès à un espace privilégié.		L'admin doit avoir un <u>login</u> et un mot de passe qui doivent être enregistrés dans la base de données	P2
	En tant qu'admin je veux pouvoir voir les demandes pour devenir superviseur dans le but d'accepter ou de décliner un nouveau superviseur.		L'admin doit être connecté à l'application	P4
	En tant qu'admin, je veux pouvoir ajouter une donnée pour une salle dans le but de collecter des informations pour entraîner le modèle.		L'admin doit être connecté à l'application	P1
	En tant qu'admin, je veux pouvoir créer une salle dans le but d'en ajouter une nouvelle.		L'admin doit être connecté à l'application	P3
	En tant qu'admin, je veux pouvoir créer un étage dans le but d'en ajouter un nouveau.		L'admin doit être connecté à l'application	P3
	En tant qu'admin, je veux pouvoir créer un campus dans le but d'en ajouter un nouveau.		L'admin doit être connecté à l'application	P3
	En tant qu'admin, je veux pouvoir visualiser les données d'une salle dans le but de vérifier que les informations sont correctes.		L'admin doit être connecté à l'application	P2
	En tant qu'admin, je veux pouvoir supprimer les données d'une salle dans le but de mettre à jour les informations.		L'admin doit être connecté à l'application	P2
	En tant qu'admin, je veux pouvoir supprimer les données d'un superviseur dans le but de garder des données correctes.		L'admin doit être connecté à l'application	P2
	En tant qu'admin je veux pouvoir supprimer le compte d'un superviseur dans le but d'avoir des superviseurs habiles.		L'admin doit être connecté à l'application	P3
	En tant qu'admin je veux pouvoir supprimer un campus dans le but d'avoir de mettre à jour les données.		L'admin doit être connecté à l'application	P3
	En tant qu'admin je veux pouvoir supprimer un bâtiment dans le but d'avoir de mettre à jour les données.		L'admin doit être connecté à l'application	P3
	En tant qu'admin je veux pouvoir supprimer un étage dans le but d'avoir de mettre à jour les données.		L'admin doit être connecté à l'application	P3
	En tant qu'admin je veux pouvoir supprimer une salle dans le but d'avoir de mettre à jour les données.		L'admin doit être connecté à l'application	P3
	En tant qu'admin je veux pouvoir réinitialiser la base de données dans le but d'enlever toutes les données si le modèle est mauvais.		L'admin doit être connecté à l'application	P4
Gérer un superviseur				
	En tant que superviseur je veux pouvoir me connecter dans le but d'avoir accès à un espace privilégié.		Le superviseur doit avoir un <u>login</u> et un mot de passe qui doivent être enregistrés dans la base de données	P2
	En tant que superviseur, je veux pouvoir ajouter une donnée pour une salle dans le but de collecter des informations pour entraîner le modèle.		Le superviseur doit être connecté à l'application	P1
	En tant que superviseur, je veux pouvoir visualiser les données d'une salle dans le but de vérifier que les informations sont correctes.		Le superviseur doit être connecté à l'application	P2
	En tant que superviseur, je veux pouvoir supprimer les données d'une salle dans le but de mettre à jour les informations.		Le superviseur doit être connecté à l'application	P2
Gérer un utilisateur				
	En tant qu'utilisateur, je veux pouvoir me localiser pour connaître ma position dans un bâtiment.		L'application possède un bouton qui permet de se localiser	P0
	En tant qu'utilisateur, je veux pouvoir choisir le mode d'affichage dans le but d'avoir la vue souhaitée.	- Reporter un mauvais plan - Reporter un dysfonctionnement de l'application - Reporter une mauvaise localisation	L'application possède un bouton pour changer de mode d'affichage	P4
	En tant qu'utilisateur, je veux pouvoir devenir superviseur dans le but de collecter des données pour le modèle.		L'application possède un bouton pour devenir superviseur	P4
	En tant qu'utilisateur, je veux pouvoir reporter un problème dans le but d'améliorer l'application.		L'application possède un bouton pour reporter un problème	P4

## b) Difficultés rencontrées

### 1. L'équipe

Nous avons eu des problèmes de communication dans le groupe. Ce qui nous a fait perdre beaucoup de temps.

### 2. Le projet

Nous n'avons pas eu de difficultés particulières concernant la conception. Cependant, nous nous sommes rendus compte qu'il nous manquait des fonctionnalités. Nous les avons donc rajoutées au fur et à mesure afin de compléter notre projet. Un point qui nous a rendu l'avancement du projet difficile est le fait que nous ayons pleins de projets en parallèle.

### 3. Les outils

Nous avons rencontré des problèmes avec Android Studio car il n'est pas maintenable. De plus, le nombre de scans que nous pouvons faire afin d'obtenir la liste des wifis est limité pour les android 8 et 9.

# Conception UML

Nous avons réalisé 4 types de diagrammes.

Diagramme Use case : client, serveur

Diagramme de classes : client, serveurModele, serveurSGBD

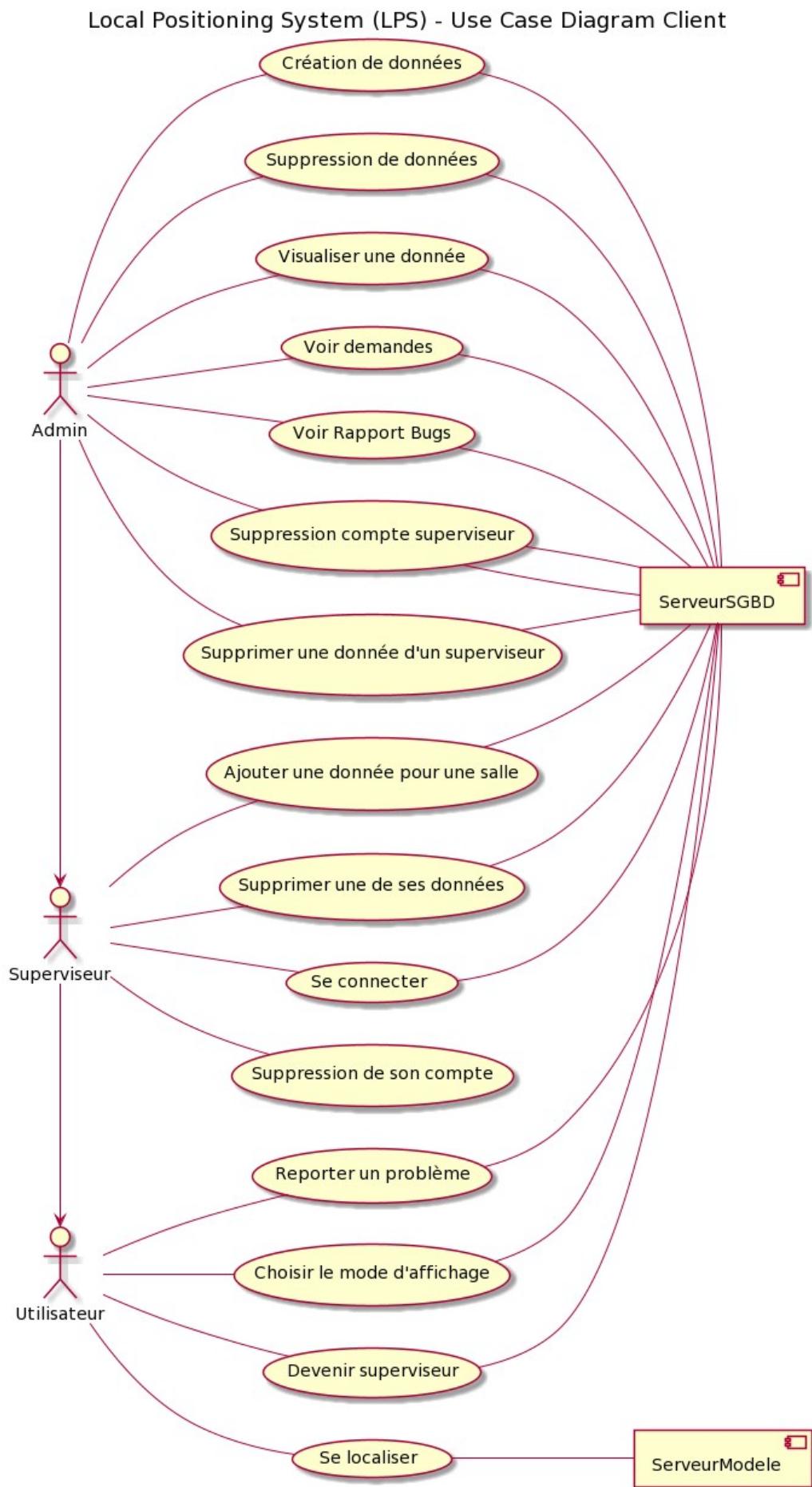
Diagramme de séquences

Diagramme d'état

a. Diagramme de cas d'usage

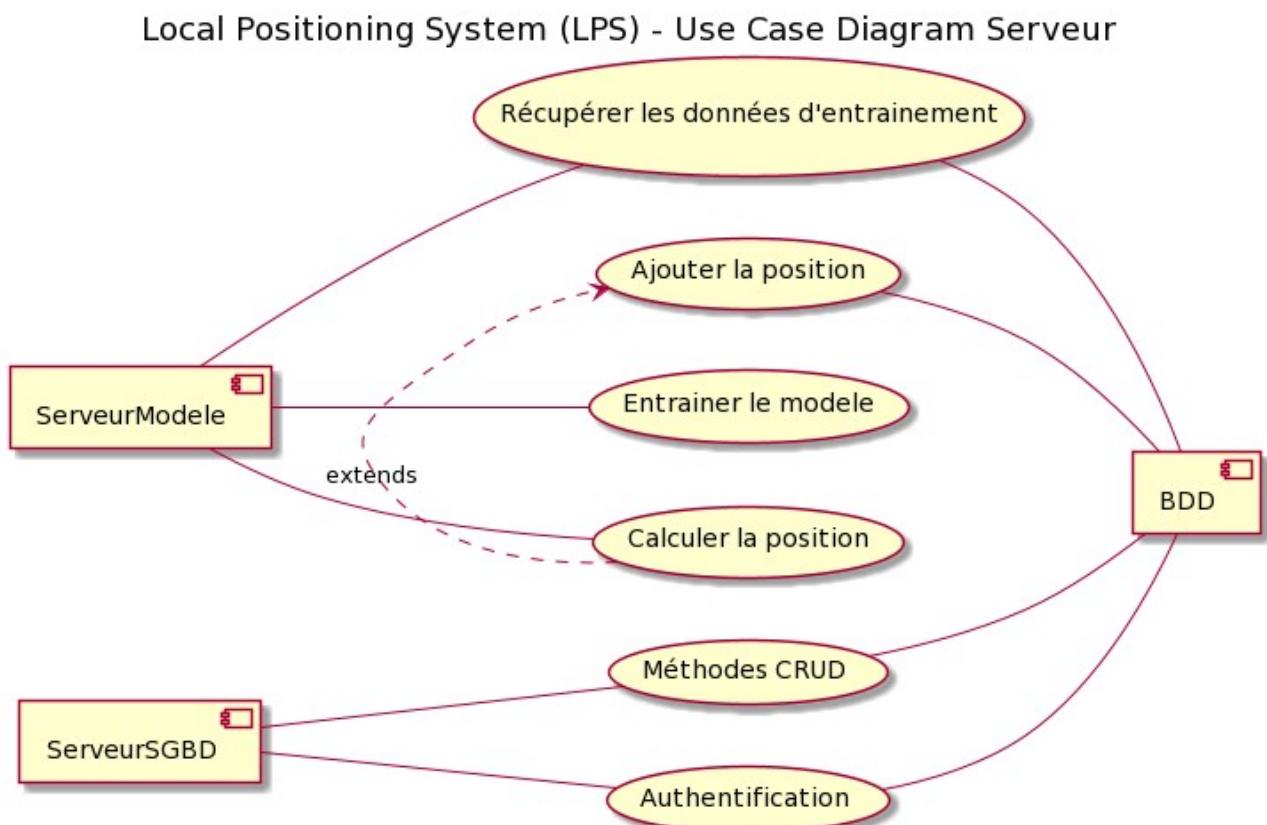
Un diagramme de cas d'usage sert à déterminer toutes les fonctionnalités que va avoir l'application avec les différents acteurs qui vont l'utiliser.

Voici le diagramme de cas d'usage client :



Notre diagramme client se compose de 3 acteurs principaux qui sont l'utilisateur, le superviseur et l'administrateur. Mais également de 2 acteurs secondaires qui sont les serveurs modèle et SGBD. La fonctionnalité « Se Localiser » utilise le serveur modèle (Python) tandis que les autres utilisent le serveur SGBD (Java).

Voici le diagramme de cas d'usage serveur :

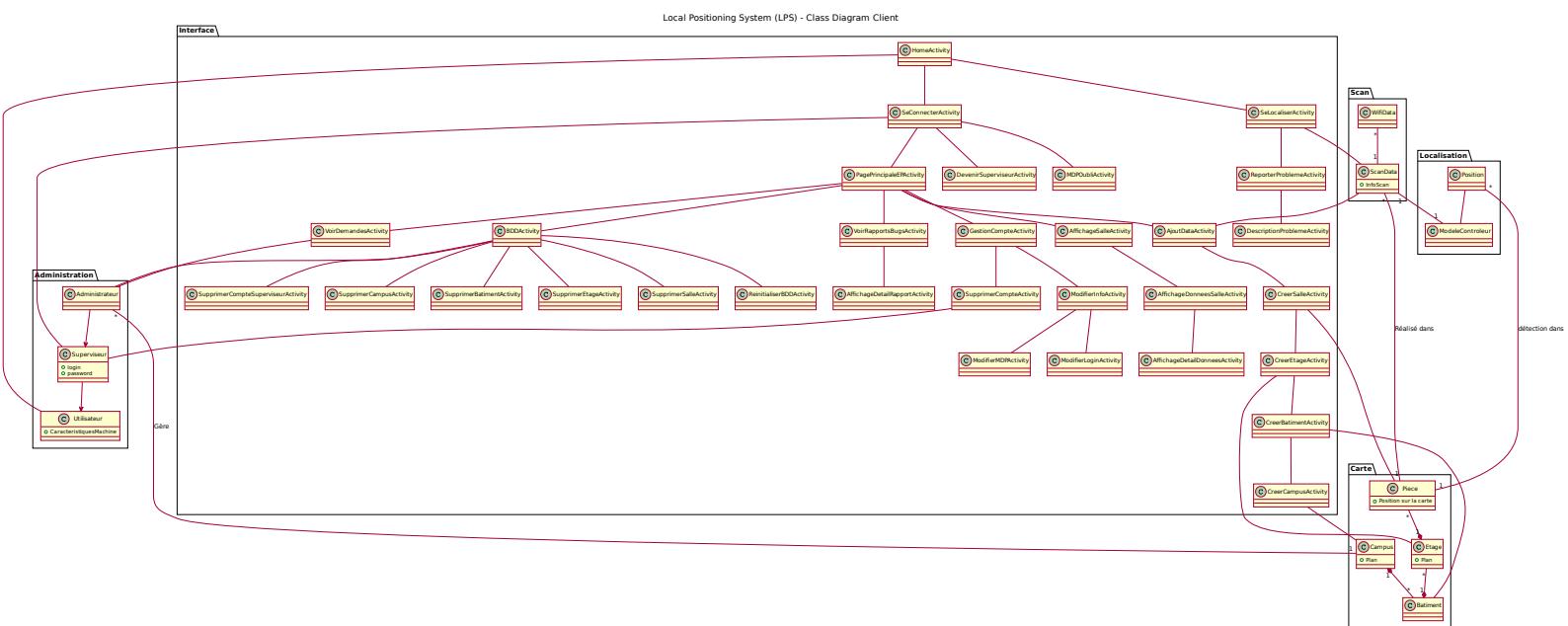


Notre diagramme serveur se compose de 2 acteurs principaux qui sont les serveurs modèle et SGBD et d'un acteur secondaire qui est la BDD. Le serveur modèle gère tout ce qui concerne les données d'entraînement et le calcul de la position alors que le serveur SGBD gère l'authentification ainsi que les méthodes CRUD.

## b. Diagramme de classes

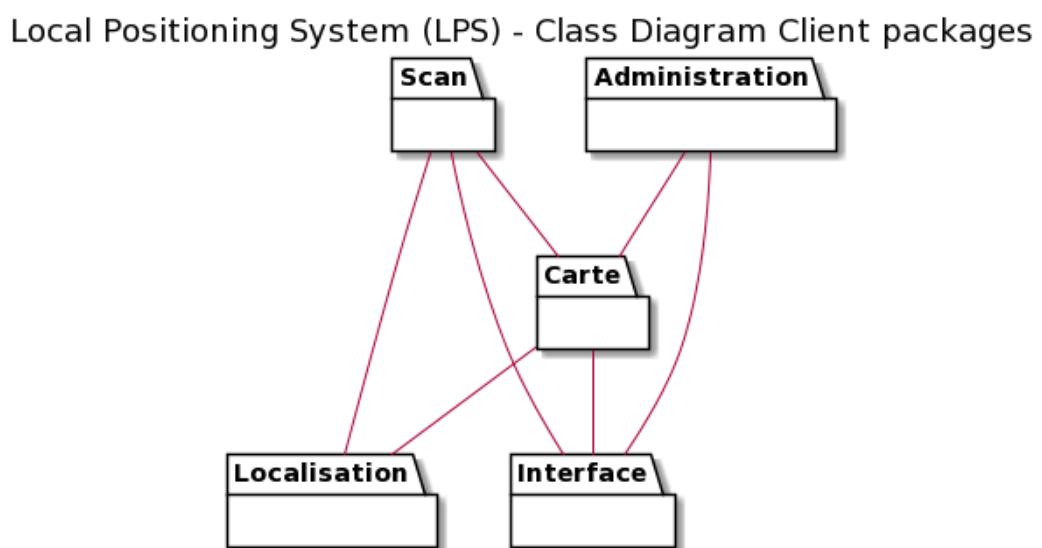
Un diagramme de classes sert à faire les liens entre les différentes classes donc les objets.

Voici le diagramme de classes client :



Notre diagramme de classes client se compose de 42 classes regroupées en 5 packages qui sont Administration, Localisation, Carte, Scan ainsi qu'Interface.

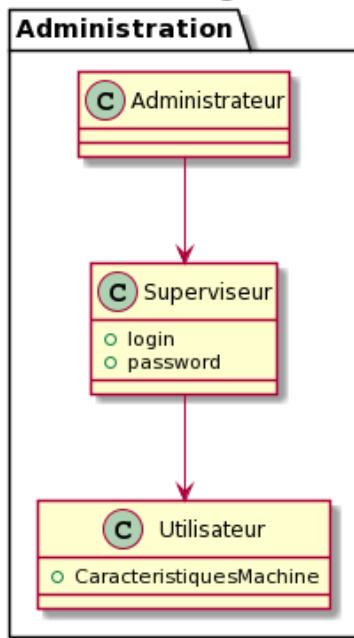
Voici comment ils se relient de manière simplifiée :



Nous allons effectuer des zooms sur chacun des paquets afin de voir plus en détail ce qu'il contienne.

Package Administration :

Local Positioning System (LPS) - Class Diagram Client package Administration

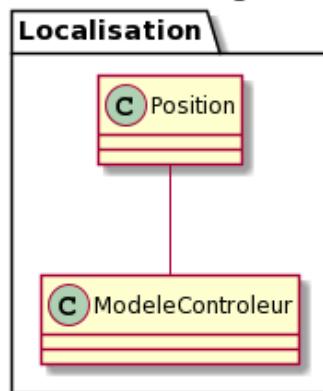


Il se compose de 3 classes représentant nos acteurs. Ce sont l'Administre, le Superviseur ainsi que l'Utilisateur.

L'Administre hérite du Superviseur qui hérite de l'Utilisateur. D'après la hiérarchie l'Administre est celui qui a le plus de droit et l'Utilisateur le moins.

Package Localisation :

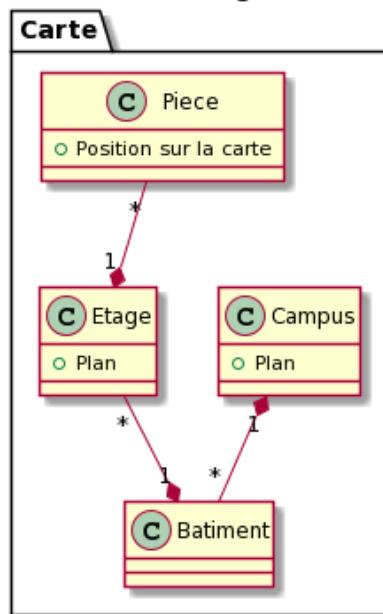
Local Positioning System (LPS) - Class Diagram Client package Localisation



Ce package se compose de 2 classes qui sont la Position et le Modèle contrôleur. La classe Position renvoie la position de la pièce et la classe Modèle contrôleur renvoie la liste des pièces possibles par le modèle.

### Package Carte :

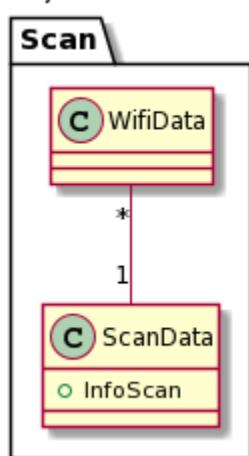
Local Positioning System (LPS) - Class Diagram Client package Administration



Il se compose de 4 classes qui sont le Campus, le Bâtiment, L'Étage ainsi que la Pièce. Un Campus peut se composer de plusieurs Bâtiments qui peut se composer de plusieurs Étages qui peut se composer de plusieurs Pièces.

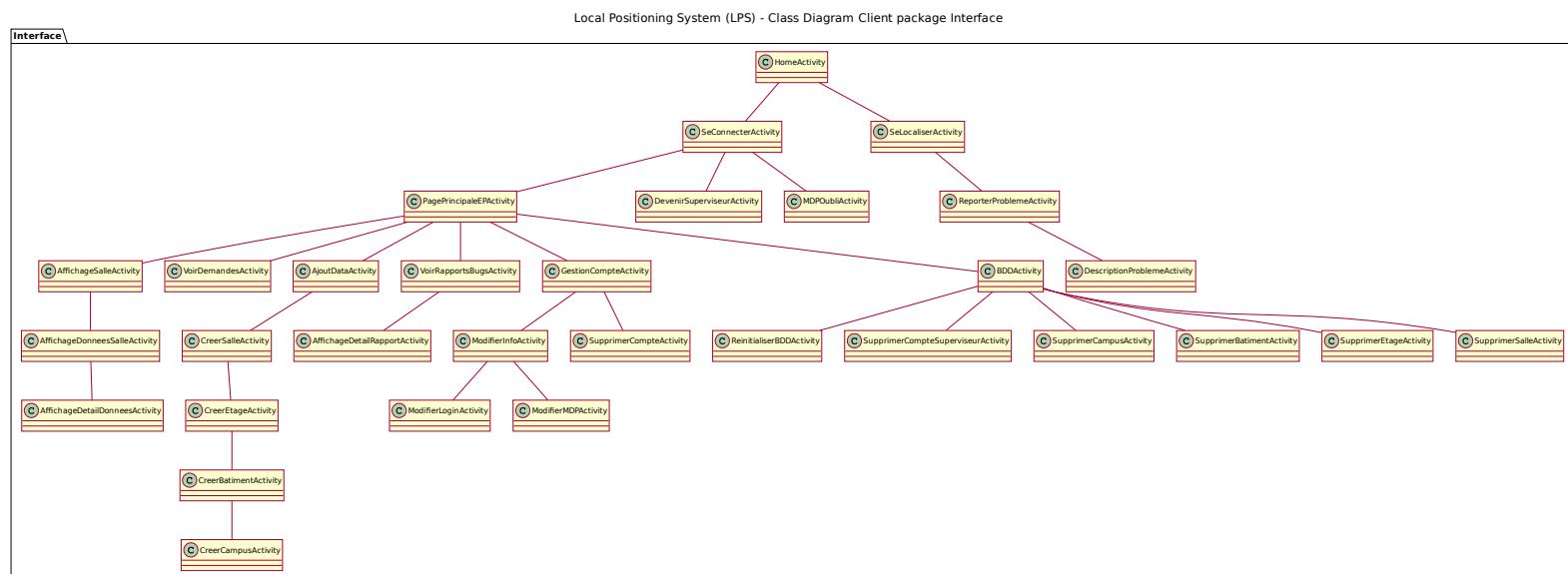
### Package Scan :

Local Positioning System (LPS) - Class Diagram Client package Scan



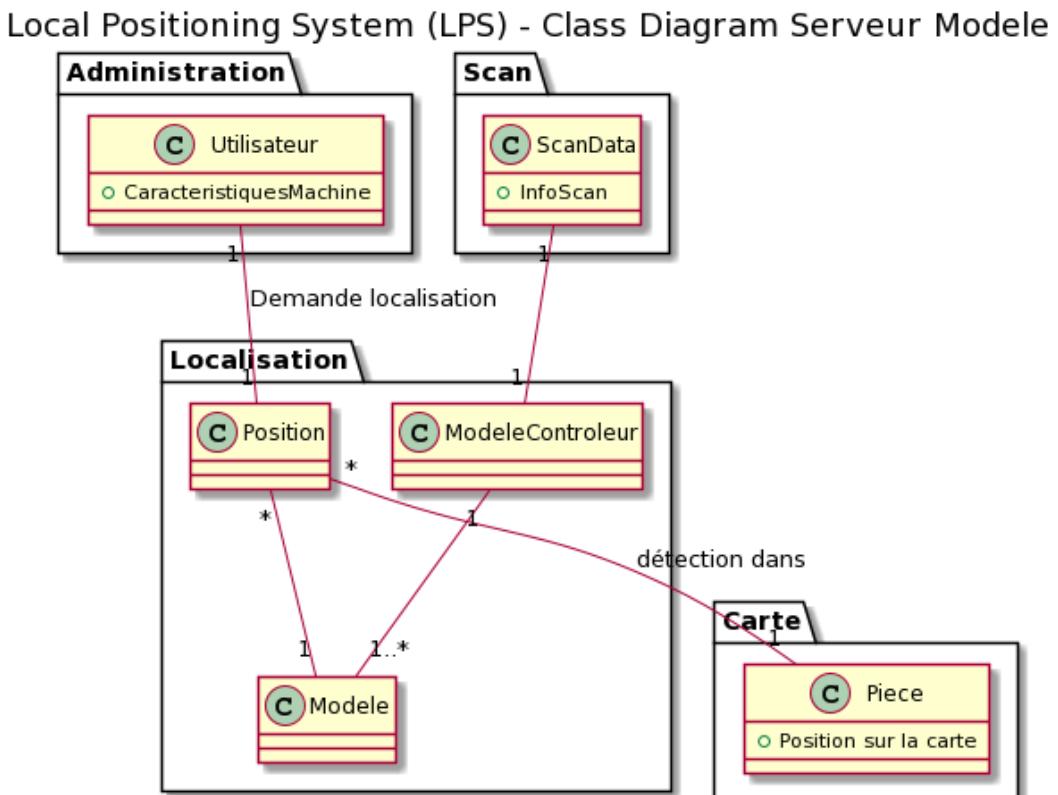
Ce package se compose de 2 classes qui sont WifiData et ScanData. Un ScanData peut se composer de plusieurs WifiScan.

Package Interface :



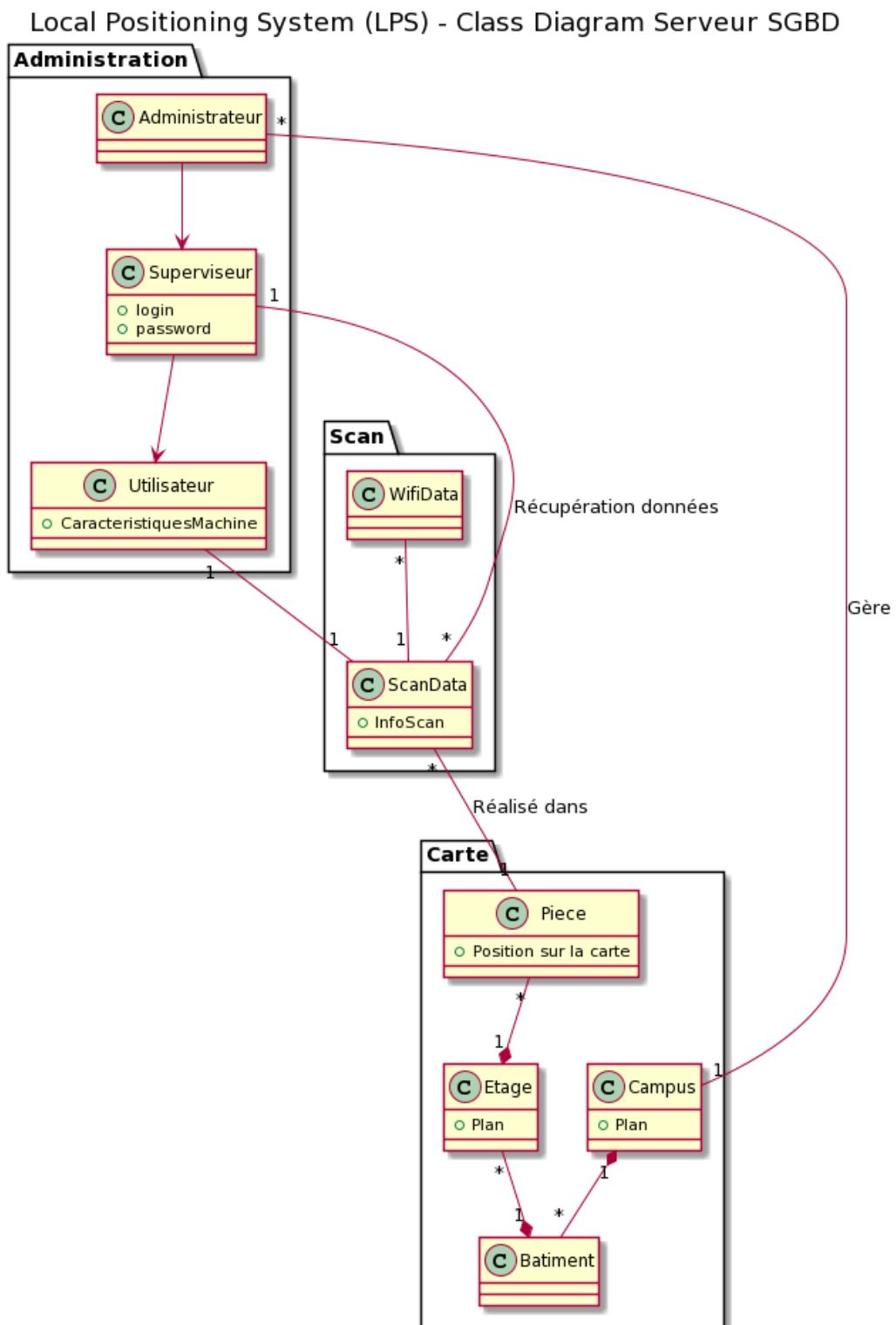
Ce package se compose de toutes les activités présentent dans notre application.

Voici le diagramme de classes serveurModele :



Notre diagramme de classes serveur Modele se compose de 6 classes regroupées en 4 packages qui sont Administration, Scan, Localisation ainsi que Carte. Ce diagramme explique le fonctionnement du calcul de la position à partir d'un scandata réalisé par un utilisateur et donc le fonctionnement du serveurModele c'est à dire du serveur Python.

Voici le diagramme de classes serveurSGBD :

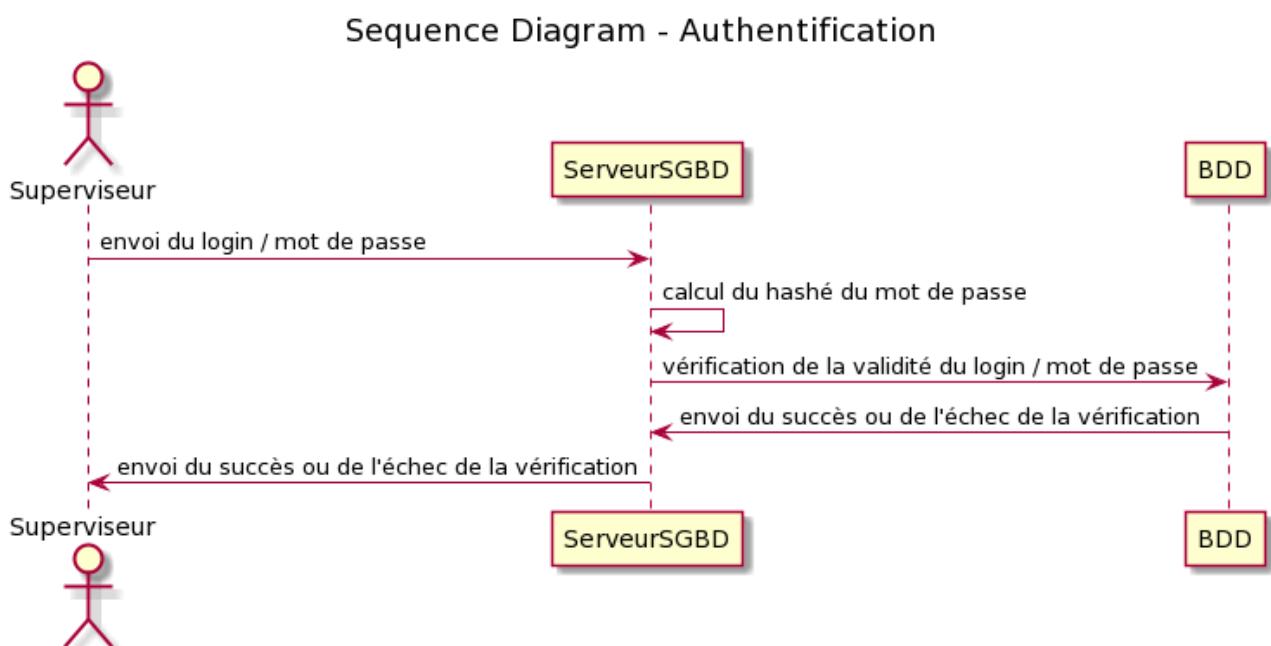


Notre diagramme de classes serveur SGBD se compose de 9 classes regroupées en 3 packages qui sont Administration, Scan et Carte. Ce diagramme montre le fonctionnement du serveurSGBD c'est à dire du serveur Java.

### c. Diagramme de séquences

Un diagramme de séquences sert à représenter graphiquement les interactions entre les acteurs et le système dans l'ordre chronologique dans lequel elles se déroulent.

Voici le diagramme de séquences :

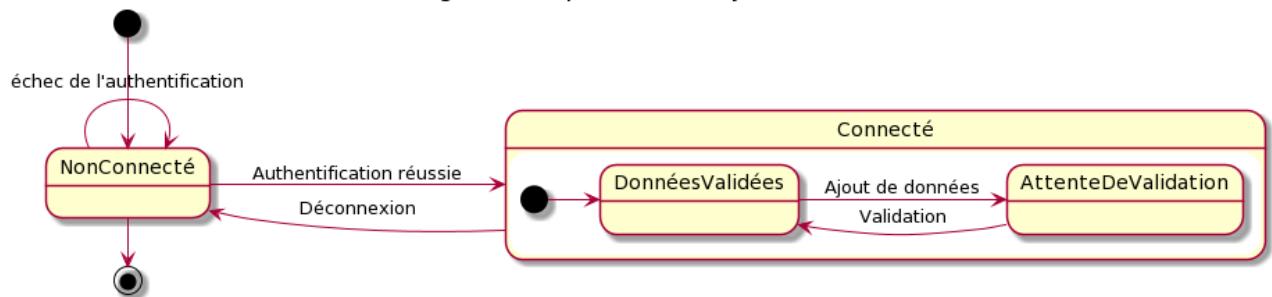


Ce diagramme de séquences illustre l'authentification et donc les échanges avec le superviseur, le serveur et la base de données nécessaires à l'authentification.

### d. Voici le diagramme d'états :

Un diagramme d'états sert à représenter graphiquement le comportement interne d'un objet à l'aide d'un automate à états finis. Il présente donc les séquences possibles d'états et d'actions qu'une instance de classe peut traiter au cours de son cycle de vie.

### State Diagram - Superviseur - Ajout de données



Ce diagramme d'états illustre les différents états dans lequel passe un superviseur lorsqu'il ajoute des données.

## Exigences

Les exigences permettent de représenter ce que doit pouvoir faire et voir un acteur particulier sur l'application. Voici la liste de nos exigences triées par acteur.

Pour l'administrateur :

En tant qu'admin je veux pouvoir me connecter dans le but d'avoir accès à un espace privilégié.

En tant qu'admin, je veux pouvoir voir les demandes pour devenir superviseur dans le but d'accepter ou de décliner un nouveau superviseur.

En tant qu'admin, je veux pouvoir ajouter une donnée pour une salle dans le but de collecter des informations pour entraîner le modèle.

En tant qu'admin, je veux pouvoir créer une salle dans le but d'en ajouter une nouvelle.

En tant qu'admin, je veux pouvoir créer un étage dans le but d'en ajouter un nouveau.

En tant qu'admin, je veux pouvoir créer un bâtiment dans le but d'en ajouter un nouveau.

En tant qu'admin, je veux pouvoir créer un campus dans le but d'en ajouter un nouveau.

En tant qu'admin, je veux pouvoir visualiser les données d'une salle dans le but de vérifier que les informations sont correctes.

En tant qu'admin, je veux pouvoir supprimer les données d'une salle dans le but de mettre à jour les informations.

En tant qu'admin, je veux pouvoir supprimer les données d'un superviseur dans le but de garder des données correctes.

En tant qu'admin je veux pouvoir supprimer le compte d'un superviseur dans le but d'avoir des superviseurs fiables.

En tant qu'admin je veux pouvoir supprimer un campus dans le but d'avoir de mettre à jour les données.

En tant qu'admin je veux pouvoir supprimer un bâtiment dans le but d'avoir de mettre à jour les données.

En tant qu'admin je veux pouvoir supprimer un étage dans le but d'avoir de mettre à jour les données.

En tant qu'admin je veux pouvoir supprimer une salle dans le but d'avoir de mettre à jour les données.

En tant qu'admin je veux pouvoir réinitialiser la base de données dans le but d'enlever toutes les données si le modèle est mauvais.

### **Pour le superviseur :**

En tant que superviseur je veux pouvoir me connecter dans le but d'avoir accès à un espace privilégié.

En tant que superviseur, je veux pouvoir ajouter une donnée pour une salle dans le but de collecter des informations pour entraîner le modèle.

En tant que superviseur, je veux pouvoir visualiser les données d'une salle dans le but de vérifier que les informations sont correctes.

En tant que superviseur, je veux pouvoir supprimer les données d'une salle dans le but de mettre à jour les informations.

### **Pour l'utilisateur :**

En tant qu'utilisateur, je veux pouvoir me localiser pour connaître ma position dans un bâtiment.

En tant qu'utilisateur, je veux pouvoir choisir le mode d'affichage dans le but d'avoir la vue souhaitée.

En tant qu'utilisateur, je veux pouvoir devenir superviseur dans le but de collecter des données pour le modèle.

En tant qu'utilisateur, je veux pouvoir reporter un problème dans le but d'améliorer l'application.

# Spécifications fonctionnelles

Les spécifications fonctionnelles permettent de détailler des fonctionnalités et de définir à l'avance le fonctionnement de l'application pour les réaliser. Elles se composent d'une description générale d'une fonctionnalité, de l'acteur principal ainsi que des préconditions nécessaires pour réaliser une fonctionnalité. Elles décrivent également les données ainsi que les règles de gestion. Elles sont souvent illustrées par des maquettes correspondant à la description générale et aux données.

Voici quelques unes des spécifications que nous avons réalisées :

UC1 : Récupérer les différentes données

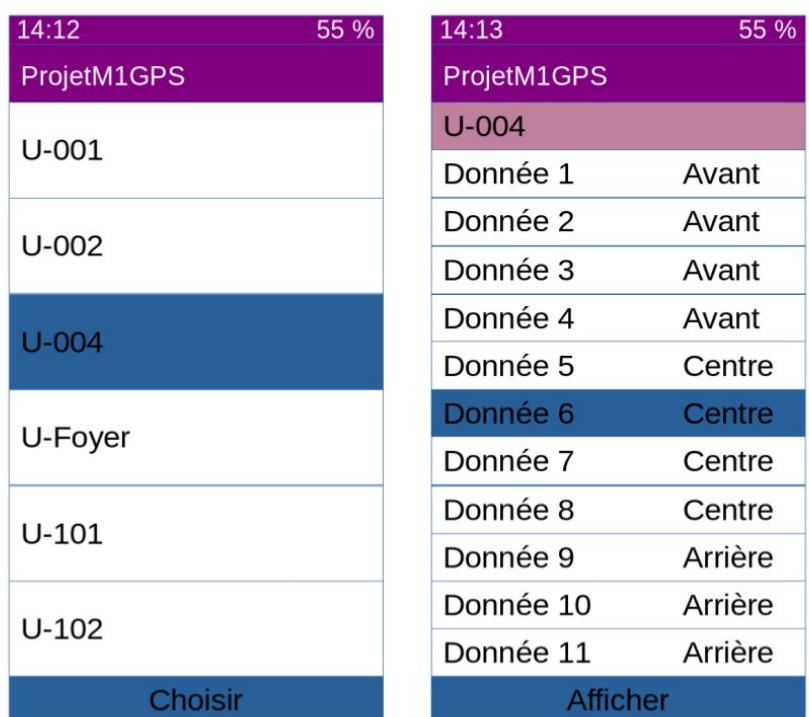
US11 : Page permettant la récupération des données

Description générale

Specifications fonctionnelles detaillees.odt

<b>Description générale</b>	<i>Page permettant l'affichage d'une salle et donc des données.</i>
<b>Acteur principal</b>	<i>Admin</i>
<b>Précondition</b>	<i>L'admin doit avoir un identifiant et mot de passe enregistrés dans la base de données. L'admin doit être autorisé à se connecter (identifiant et mot de passe corrects).</i>

Maquette



## Données

Type : N Numérique, C Caractères, D Date, H Heure, B Boutons, L Liens, LD Liste déroulante, I Image/picto

<b>Libellé</b>	<b>Type</b>	<b>Affichage (A) ou Saisie obligatoire (O/N)</b>	<b>Commentaires</b>
<i>Nom de la salle</i>	C	A	<i>Nom de la salle entré lors du scan</i>
<i>Nom de la donnée</i>	C	A	<i>Nom de la donnée récupérée</i>
<i>Info du scan</i>	C	A	<i>Information entrée lors du scan</i>
<i>Actions</i>			
<i>Choisir</i>	B	A	<i>Permet de choisir une salle</i>
<i>Afficher</i>	B	A	<i>Permet d'afficher une donnée</i>

## Règles de gestion

<b>Identifiant</b>	<b>Description de la règle</b>
US2_RG01	<i>La connexion à l'application est possible si l'admin ou le superviseur a un identifiant enregistré dans la base de données.</i>
US2_RG02	<i>La connexion à l'application est possible si l'identifiant et le mot de passe sont corrects.</i>
US2_RG03	<i>Si mauvais login ou mot de passe, affichage du message "Login ou mot de passe incorrects veuillez réessayer". Affichage du message en rouge.</i>
US2_RG04	<i>Le bouton Choisir ne fonctionne que si une salle est sélectionnée.</i>
US2_RG05	<i>Le bouton Afficher ne fonctionne que si une donnée est sélectionnée.</i>

UC2 : Supprimer une donnée

US21 : Page permettant la suppression d'une donnée

#### Description générale

<b>Description générale</b>	<i>Page permettant la suppression d'une donnée afin de mettre à jour les informations.</i>
<b>Acteur principal</b>	<i>Admin</i>
<b>Précondition</b>	<i>L'admin doit avoir un identifiant et mot de passe enregistrés dans la base de données. L'admin doit être autorisé à se connecter (identifiant et mot de passe corrects).</i>

#### Maquette

14:12	55 %
ProjetM1GPS	
U-001	
U-002	
U-004	
U-Foyer	
U-101	
U-102	
Choisir	

14:13	55 %
ProjetM1GPS	
U-004	
Donnée 1	Avant
Donnée 2	Avant
Donnée 3	Avant
Donnée 4	Avant
Donnée 5	Centre
Donnée 6	Centre
Donnée 7	Centre
Donnée 8	Centre
Donnée 9	Arrière
Donnée 10	Arrière
Donnée 11	Arrière
Supprimer	

## Données

Type : N Numérique, C Caractères, D Date, H Heure, B Boutons, L Liens, LD Liste déroulante, I Image/picto

<b>Libellé</b>	<b>Type</b>	<b>Affichage (A) ou Saisie obligatoire (O/N)</b>	<b>Commentaires</b>
<i>Nom de la salle</i>	C	A	<i>Nom de la salle entré lors du scan</i>
<i>Nom de la donnée</i>	C	A	<i>Nom de la donnée récupérée</i>
<i>Info du scan</i>	C	A	<i>Information entrée lors du scan</i>
<i>Actions</i>			
<i>Choisir</i>	B	A	<i>Permet de choisir une salle</i>
<i>Supprimer</i>	B	A	<i>Permet de supprimer une donnée</i>

## Règles de gestion

<b>Identifiant</b>	<b>Description de la règle</b>
US2_RG01	<i>La connexion à l'application est possible si l'admin ou le superviseur a un identifiant enregistré dans la base de données.</i>
US2_RG02	<i>La connexion à l'application est possible si l'identifiant et le mot de passe sont corrects.</i>
US2_RG03	<i>Si mauvais login ou mot de passe, affichage du message "Login ou mot de passe incorrects veuillez réessayer". Affichage du message en rouge.</i>
US2_RG04	Le bouton Choisir ne fonctionne que si une salle est sélectionnée.
US2_RG05	Le bouton Supprimer ne fonctionne que si une donnée est sélectionnée.

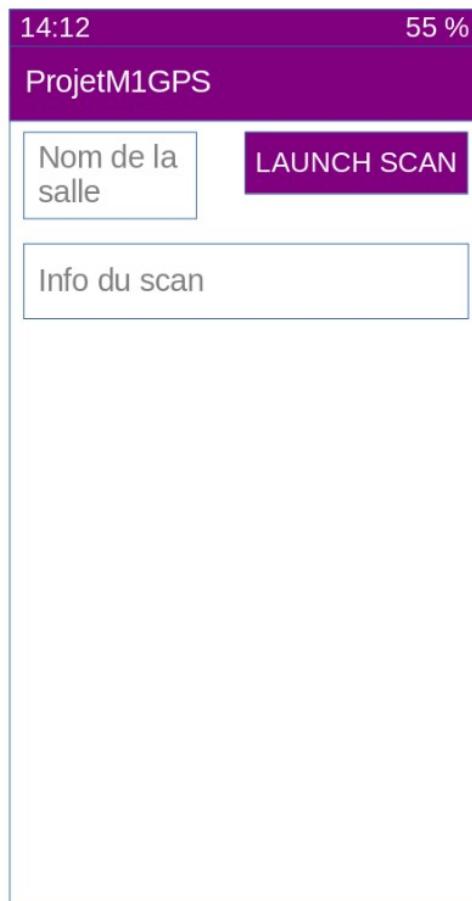
UC3 : Ajouter une donnée pour une salle

US31 : Page permettant d'ajouter une donnée pour une salle

#### Description générale

<b>Description générale</b>	<i>Page permettant la récupération de données afin de pouvoir collecter des informations pour l'algorithme d'apprentissage</i>
<b>Acteur principal</b>	Admin
<b>Précondition</b>	<i>L'admin doit avoir un identifiant et mot de passe enregistrés dans la base de données. L'admin doit être autorisé à se connecter (identifiant et mot de passe corrects).</i>

#### Maquette



## Données

Type : N Numérique, C Caractères, D Date, H Heure, B Boutons, L Liens, LD Liste déroulante, I Image/picto

<b>Libellé</b>	<b>Type</b>	<b>Affichage (A) ou Saisie obligatoire (O/N)</b>	<b>Commentaires</b>
<i>Nom de la salle</i>	C	A, O	<i>Nom de la salle associé au scan</i>
<i>Info du scan</i>	C	A, O	<i>Information précise du scan</i>
<i>Actions</i>			
LAUNCH SCAN	B	A	<i>Permet de lancer le scan</i>

## Règles de gestion

<b>Identifiant</b>	<b>Description de la règle</b>
US3_RG01	<i>La connexion à l'application est possible si l'admin a un identifiant enregistré dans la base de données.</i>
US3_RG02	<i>La connexion à l'application est possible si l'identifiant et le mot de passe sont corrects.</i>
US3_RG03	<i>Si mauvais login ou mot de passe, affichage du message "Login ou mot de passe incorrects veuillez réessayer". Affichage du message en rouge.</i>

## Maquette



### Données

Type : N Numérique, C Caractères, D Date, H Heure, B Boutons, L Liens, LD Liste déroulante, I Image/picto

Libellé	Type	Affichage (A) ou Saisie obligatoire (O/N)	Commentaires
Login	C	A, O	<i>Champ permettant de rentrer le login</i>
Mot de passe	C	A, O	<i>Champ permettant de rentrer le mot de passe</i>
<i>Actions</i>			
Se connecter	B	A	Permet de se connecter à l'application

US4\_RG02 corrects.

US4\_RG03 Si mauvais login ou mot de passe, affichage du message "Login ou mot de passe incorrects veuillez réessayer". Affichage du message en rouge.

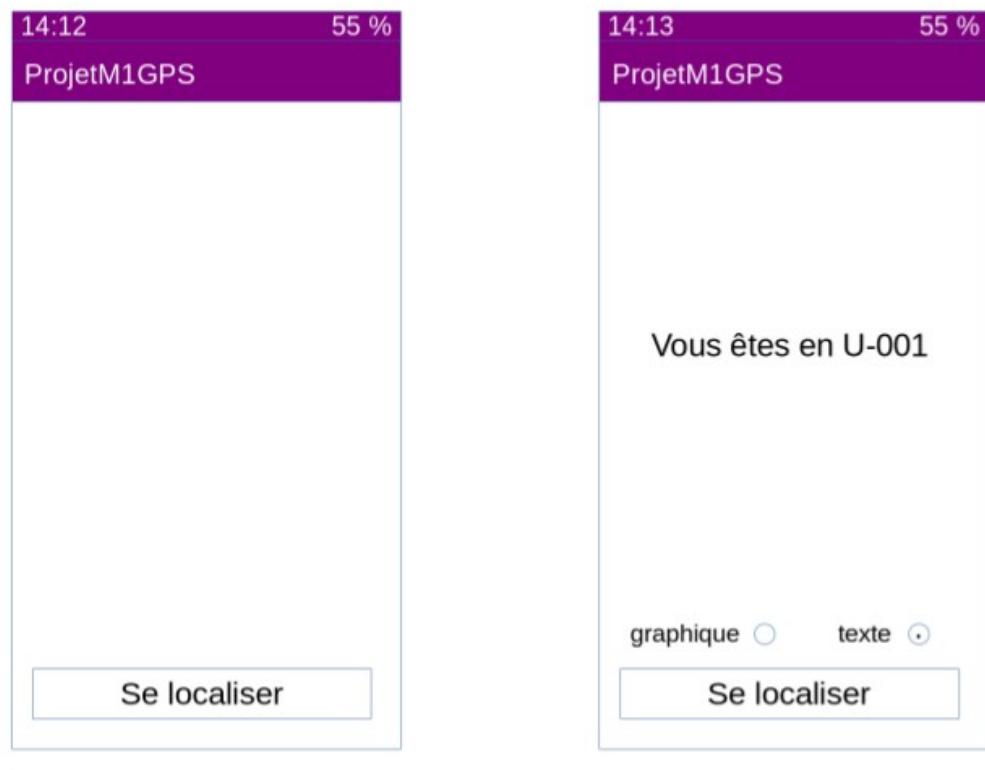
UC5 : Se localiser

US51 : Page permettant de savoir où l'on se trouve

#### Description générale

<b>Description générale</b>	<i>Page permettant de savoir où l'on se trouve.</i>
<b>Acteur principal</b>	Utilisateur
<b>Précondition</b>	<i>L'utilisateur doit avoir appuyé sur le bouton « Se localiser ».</i>

#### Maquette



Avant et après avoir cliqué sur le bouton.

UC6 : Choisir le mode d'affichage

US61 : Bouton permettant de choisir le mode d'affichage

Description générale

<b>Description générale</b>	<i>Bouton permettant de choisir le mode d'affichage</i>
<b>Acteur principal</b>	Utilisateur

Maquette



## Données

Type : N Numérique, C Caractères, D Date, H Heure, B Boutons, L Liens, LD Liste déroulante, I Image/picto, R Radiobutton

Libellé	Type	Affichage (A) ou Saisie obligatoire (O/N)	Commentaires
<i>Actions</i>			
graphique	R	A	Affichage graphique de l'endroit où l'on se trouve.
texte	R	A	Affichage textuelle de l'endroit où l'on se trouve.

## Règles de gestion

Identifiant	Description de la règle
US6_RG01	<i>L'appui sur le bouton « Se localiser » est obligatoire pour que les radiobuttons apparaissent.</i>

## UC7 : Reporter un problème

### Maquette

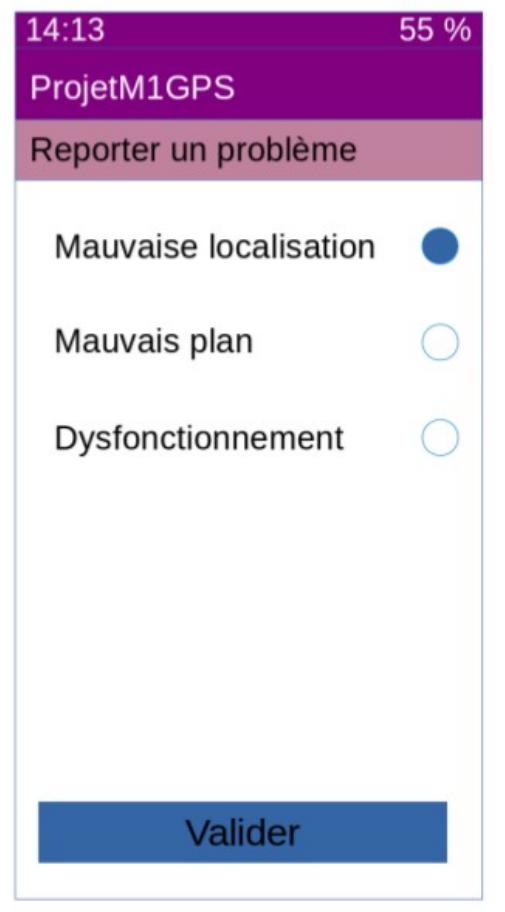


### US71 : Mauvaise localisation

#### Description générale

<b>Description générale</b>	<i>Bouton permettant de reporter un problème</i>
<b>Acteur principal</b>	Utilisateur

## Maquette



## Règles de gestion

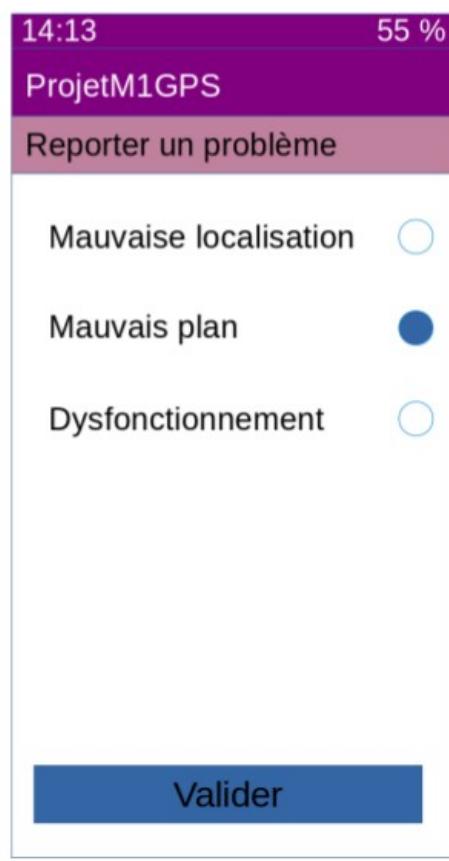
Identifiant	Description de la règle
US7_RG01	<i>L'utilisateur doit être connecté à internet.</i>

## US72 : Mauvais plan

### Description générale

<b>Description générale</b>	<i>Bouton permettant de reporter un problème</i>
<b>Acteur principal</b>	Utilisateur

### Maquette



## Données

Type : N Numérique, C Caractères, D Date, H Heure, B Boutons, L Liens, LD Liste déroulante, I Image/picto, R Radiobutton

Libellé	Type	Affichage (A) ou Saisie obligatoire (O/N)	Commentaires
<i>Actions</i>			
Reporter un problème	B	A	Bouton permettant de reporter un problème.
Mauvais plan	R	A	Radiobutton permettant de reporter le problème mauvais plan.
Valider	B	A	Bouton permettant de valider le report du problème.

## Règles de gestion

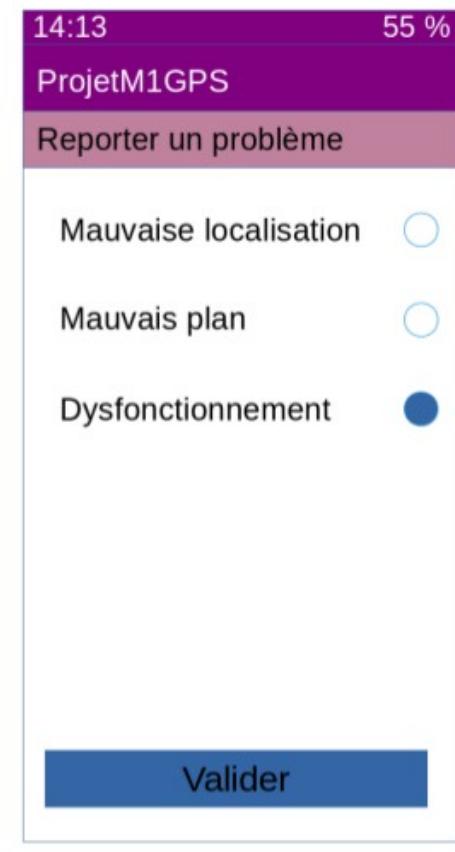
Identifiant	Description de la règle
US7_RG01	<i>L'utilisateur doit être connecté à internet.</i>

## US73 : Dysfonctionnement de l'application

### Description générale

Description générale	<i>Bouton permettant de reporter un problème</i>
Acteur principal	Utilisateur

## Maquette



### Données

#### Règles de gestion

Identifiant	Description de la règle	
US7_RG01	<i>L'utilisateur doit être connecté à internet.</i>	

Reporter un problème	B	A	Bouton permettant de reporter un problème.
Dysfonctionnement de l'application	R	A	Radiobutton permettant de reporter un problème de dysfonctionnement de l'application.
Valider	B	A	Bouton permettant de valider le report du problème.



# Maquettage de l'application

Les maquettes permettent de préparer l'implantation en se mettant d'accord sur l'interface graphique de l'application. Cela nous permet de définir à l'avance les éléments de chaque page liée aux fonctionnalités et de fixer leur placement ainsi que le fonctionnement de l'interface homme machine pour éviter tout imprévu après le début de l'implantation.

Dans notre cas, nous avons représentés les différentes fenêtres et onglets possibles de l'application. Ci-dessous voici quelques exemples de maquettes de notre application.

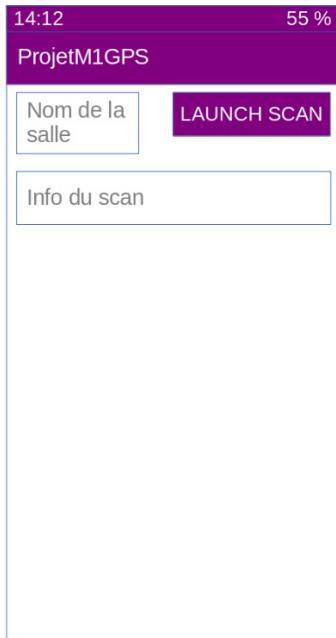
Maquette de visualisation des données

14:12	55 %
ProjetM1GPS	
U-001	
U-002	
U-004	
U-Foyer	
U-101	
U-102	
<b>Choisir</b>	
<b>Afficher</b>	

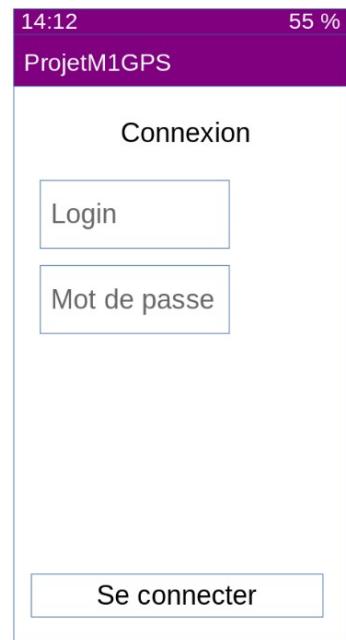
Maquette de suppression d'une donnée

14:12	55 %
ProjetM1GPS	
U-001	
U-002	
U-004	
U-Foyer	
U-101	
U-102	
<b>Choisir</b>	
<b>Supprimer</b>	

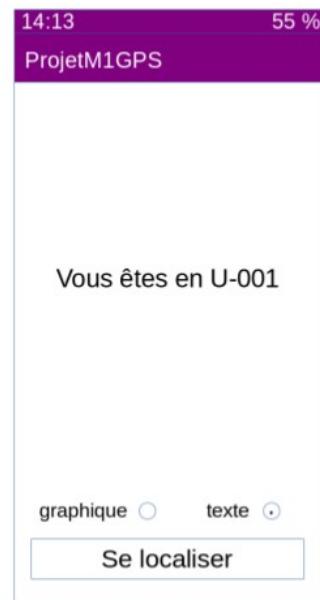
## Maquette de l'ajout d'une données



## Maquette de la page de connexion

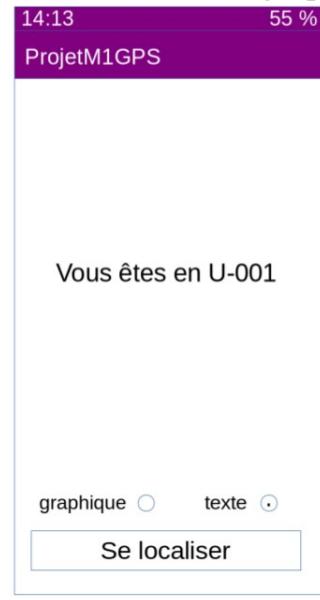


## Maquette pour se localiser version texte



Avant et après avoir cliqué sur le bouton

## Maquette pour se localiser version graphique



Avant et après avoir cliqué sur le bouton

[Reporter un problème](#)

## Maquette pour reporter un problème (page se localiser)

Vous êtes en U-001

graphique  texte [Se localiser](#)

## Maquette pour reporter un problème (mauvaise localisation, mauvais plan, dysfonctionnement)

14:13 55 %  
ProjetM1GPS  
[Reporter un problème](#)

Mauvaise localisation	<input checked="" type="radio"/>
Mauvais plan	<input type="radio"/>
Dysfonctionnement	<input type="radio"/>

[Valider](#)

14:13 55 %  
ProjetM1GPS  
[Reporter un problème](#)

Mauvaise localisation	<input type="radio"/>
Mauvais plan	<input checked="" type="radio"/>
Dysfonctionnement	<input type="radio"/>

[Valider](#)

14:13 55 %  
ProjetM1GPS  
[Reporter un problème](#)

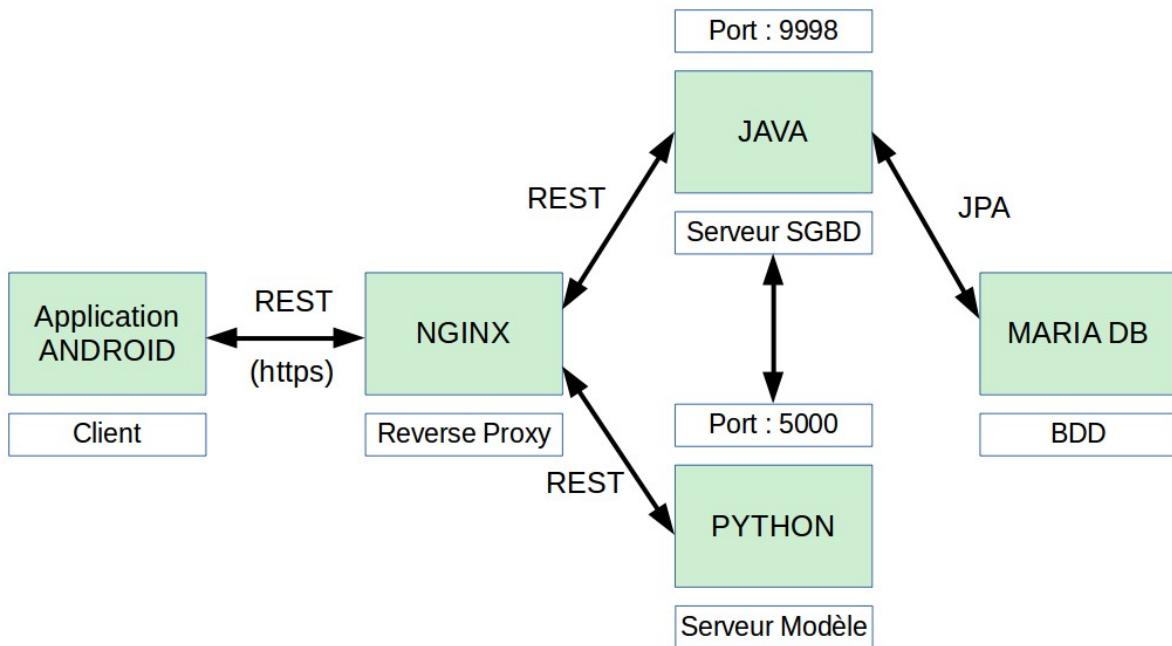
Mauvaise localisation	<input type="radio"/>
Mauvais plan	<input type="radio"/>
Dysfonctionnement	<input checked="" type="radio"/>

[Valider](#)

# Structure de notre projet

Notre projet se compose de 2 serveurs, d'un reverse proxy, d'une base de données ainsi qu'une application Android.

Voici en détails la structure de notre projet :



L'application Android communique avec le reverse proxy NGINX. Ils interagissent avec un protocole https et une API REST en cohérence avec la norme RESTful. Ce reverse proxy peut également communiquer avec les serveurs java et python encore grâce à une API REST. Selon l'url utilisée cela redirige soit vers le serveur java pour le port 9998, soit vers le serveur python pour le port 5000. Les 2 serveurs communiquent entre eux mais seulement le serveur java interagit avec la base de données maria-db grâce aux annotations JPA (Java Persistence API).

# Code

Les langages de programmation utilisés pour ce projet sont le Java8 et le Python3. Nous avons réparti le code en 3 catégories ; une partie Python, une partie Java ainsi qu'une partie Android.

La partie python gère tout ce qui est en rapport avec le modèle. Concernant notre modèle, il restera peu performant si nous utilisons seulement les wifis.

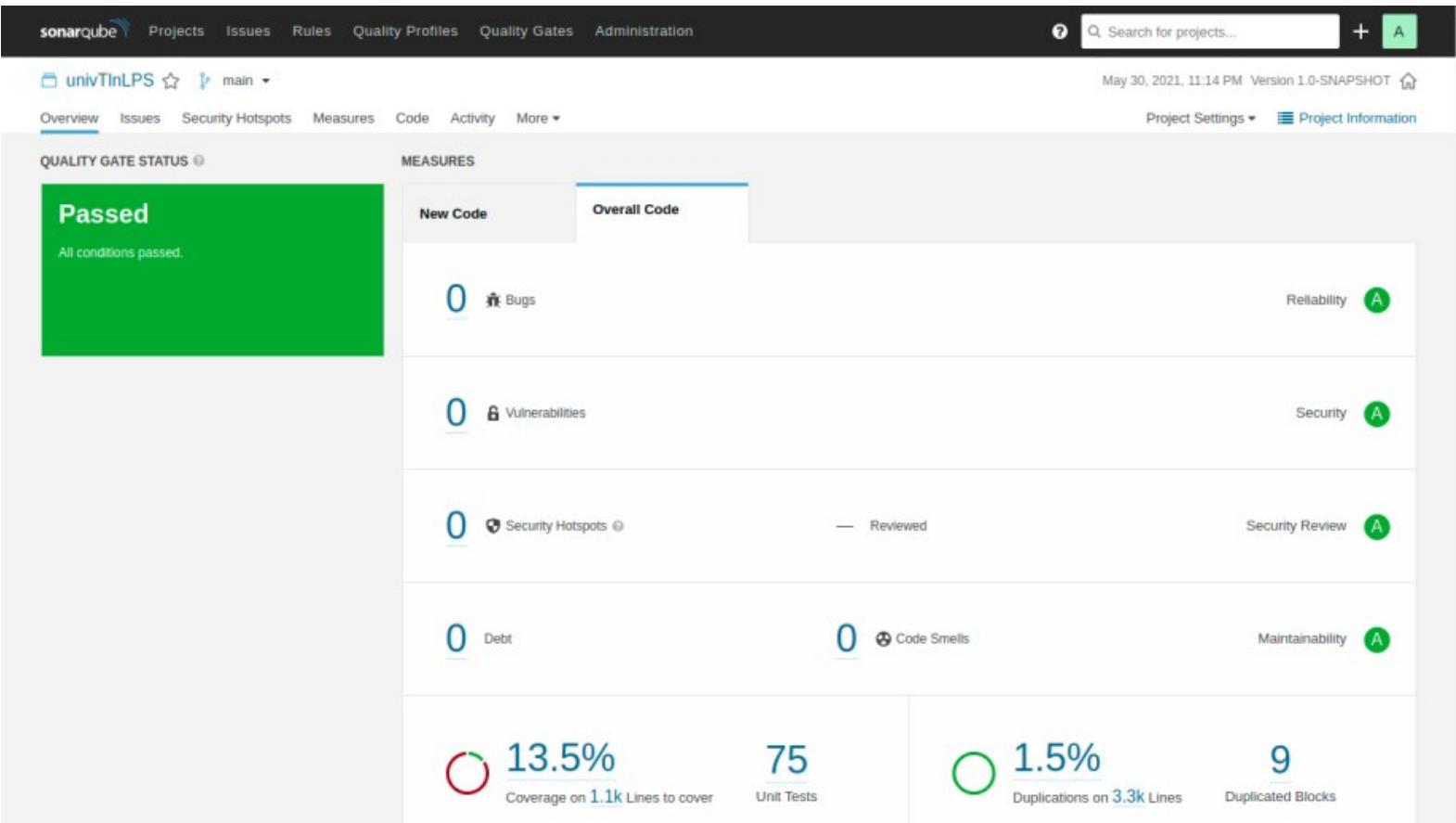
La partie Java s'occupe de toutes les méthodes CRUD car elle fait le lien entre la BDD et l'application Android.

La partie Android s'occupe elle de l'interface et de certaines méthodes permettant d'effectuer des actions qui seront transmises à la BDD via le Java.

Pour générer les tests unitaires et la javadoc nous nous sommes servis des plugins Diffblue et JavaDoc.

Afin de mesurer la qualité de notre code et de l'améliorer nous avons réalisé à l'aide de SonarQube des analyses de code.

Voici le détail d'une de ces analyses pour notre projet :



## Base de données

Pour notre projet, nous avons utilisé maria-db qui est un système de gestion de bases de données relationnelles. Elle est reliée au serveur SGBD qui est le serveur Java.

## Rôle des membres de l'équipe

Nous nous sommes répartis le travail de la manière suivante :

Léa : Récupération de données, diagrammes, Modèle machine learning, JPA, authentification, DAOS, SonarQube.

Charles : Récupération de données, diagrammes, Modèle machine learning, JPA, DAOS, REST et ihm android, REST Java, NGINX, docker.

Tiffany : Récupération de données, diagrammes, spécifications, plans, client REST android, méthodes REST et ihm pour android, REST Java et JPA, Rapport.

Nylann : Récupération de données, serveur REST Python, NGINX, mise en place des images docker avec NGINX pour le reverse proxy, le serveur web et images docker BDD, site web.

## Perspectives d'évolution

Voici quelques perspectives d'évolution intéressantes que nous pourrions ajouter à notre application.

Les voici :

- Ajouter les plans des bâtiments
- Gestion des données depuis le site
- Path pour entrer le modèle
- Gestion de mails
- Utilisation de l'application pour plusieurs campus
- Ajouter des champs au formulaire comme filière pour pouvoir sélectionner les superviseurs sur des critères plus précis
- Prendre en compte les caractéristiques machine

## Conclusion

Pour conclure, ce projet a été riche en langages et nous a amené à découvrir de nouvelles technologies qui nous ont permis de progresser et d'asseoir nos connaissances ainsi qu'à faire face aux quelques difficultés rencontrées en trouvant des solutions efficaces de façon plus autonome. Nous avons également un fort sentiment d'avoir progressé chacun sur le plan personnel au niveau des compétences sur de nouveaux domaines mais aussi de la gestion de groupe.

## Annexe :

### IHM et Manuel utilisateur

Nous allons maintenant présenter les différentes activités de notre application.

HomeActivity :



rentrer l'ip afin de pouvoir commencer à utiliser l'application (4)

bouton pour accéder à son compte superviseur ou faire une demande pour le devenir

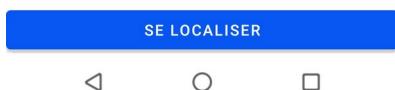
bouton pour continuer sans se connecter afin de se localiser (1)

SeLocaliser (1) :

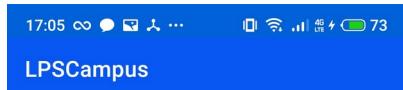


bouton permettant de reporter un problème (2)

bouton permettant de se localiser



## ReporterProbleme (2) :



- Mauvaise Localisation
- Mauvais Plan
- Dysfonctionnement

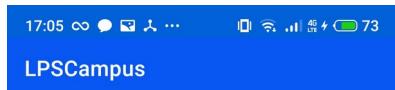
VALIDER

Choisir le motif du problème rencontré

valider le motif sélectionner (3)

◀ ○ □

## DescriptionProbleme (3) :



Décrivez le problème rencontré :

|

décrire précisément le problème rencontré

VALIDER

valider la description du problème

◀ ○ □

#### SeConnecter (4) :

Connexion

Login

Mot de passe

**SE CONNECTER**

Mot de passe oublié

DEMANDER À DEVENIR SUPERVISEUR

◀ ○ □

- rentrer son login
- rentrer son mot de passe
- bouton permettant de se connecter à son compte (5)
- si l'on clique dessus renvoi vers le formulaire de mot de passe oublié (27)
- bouton permettant d'accéder au formulaire pour devenir superviseur (28)

#### PagePrincipaleEP (5) : Superviseur

LPSCampus

▶ ○ □

AJOUTER DES DONNÉES

CONSULTER LES DONNEES

RAPPORTS DE BUGS

DEMANDES SUPERVISEUR

#### Administrateur

ENTRAINER LE MODÈLE
GÉRER BDD
AJOUTER DES DONNÉES
CONSULTER LES DONNEES
RAPPORTS DE BUGS
DEMANDES SUPERVISEUR

11:48

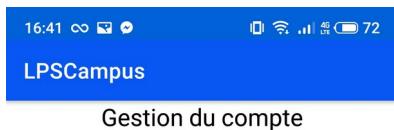
LPSCampus

▶ ○ □

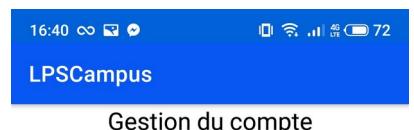
- bouton permettant d'accéder aux paramètres du compte (6)
- bouton de déconnexion
- bouton permettant d'entraîner le modèle
- bouton permettant de gérer la bdd (21)
- bouton permettant d'ajouter de nouvelles données (11)
- bouton permettant de consulter les données (16)
- bouton permettant de voir les rapports de bugs (18)
- bouton permettant de voir les demandes pour devenir superviseur (20)

## Gestion de compte (6) :

Superviseur



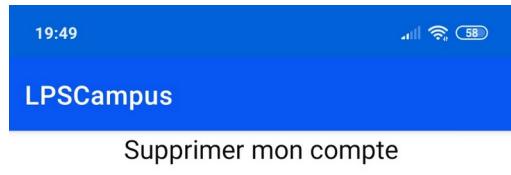
Administrateur



bouton permettant de supprimer son compte (uniquement pour le superviseur) (7)

bouton permettant de modifier le compte comme le login et le mot de passe (8)

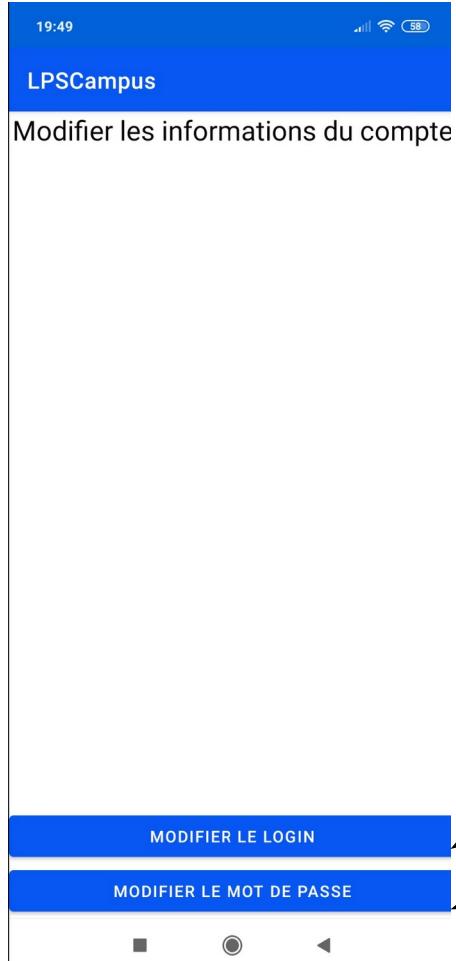
## SupprimerCompte (7) :



case cochée nécessaire pour supprimer son compte

bouton permettant de valider la suppression de son compte

## ModifierInfo (8) :



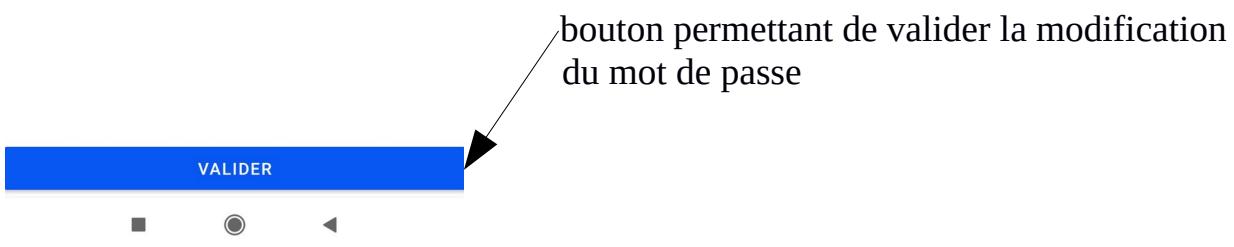
bouton permettant de modifier son login (9)

bouton permettant de modifier son mot de passe (10)

## ModifierLogin (9) :

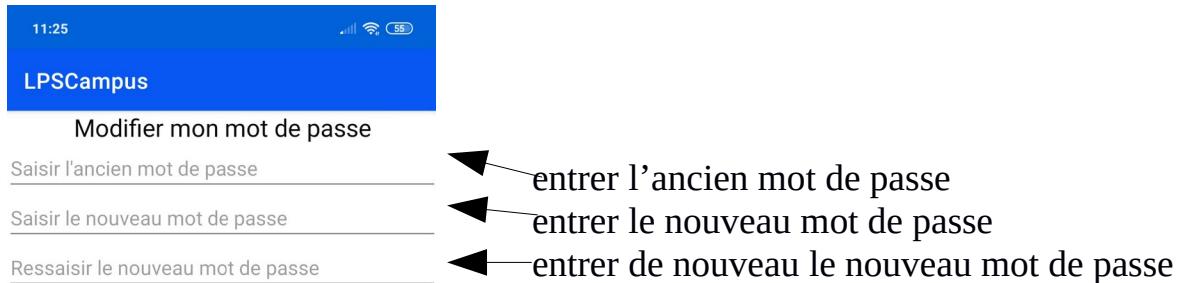


entrer une adresse mail qui sera le nouveau login



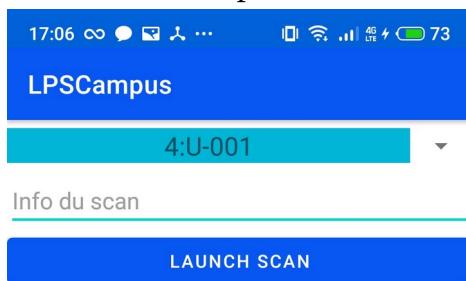
bouton permettant de valider la modification du mot de passe

## ModifierMDP (10) :

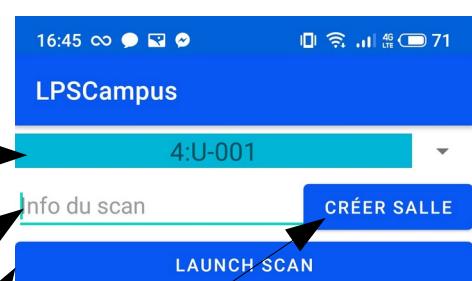


## AjoutData (11) :

Superviseur



## Administrateur



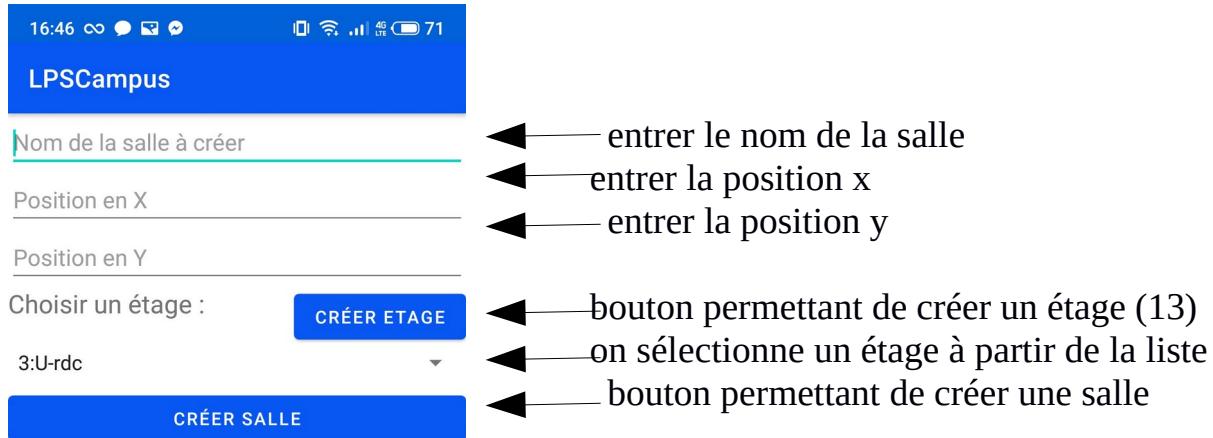
on sélectionne la salle à partir  
de la liste

on rentre une information

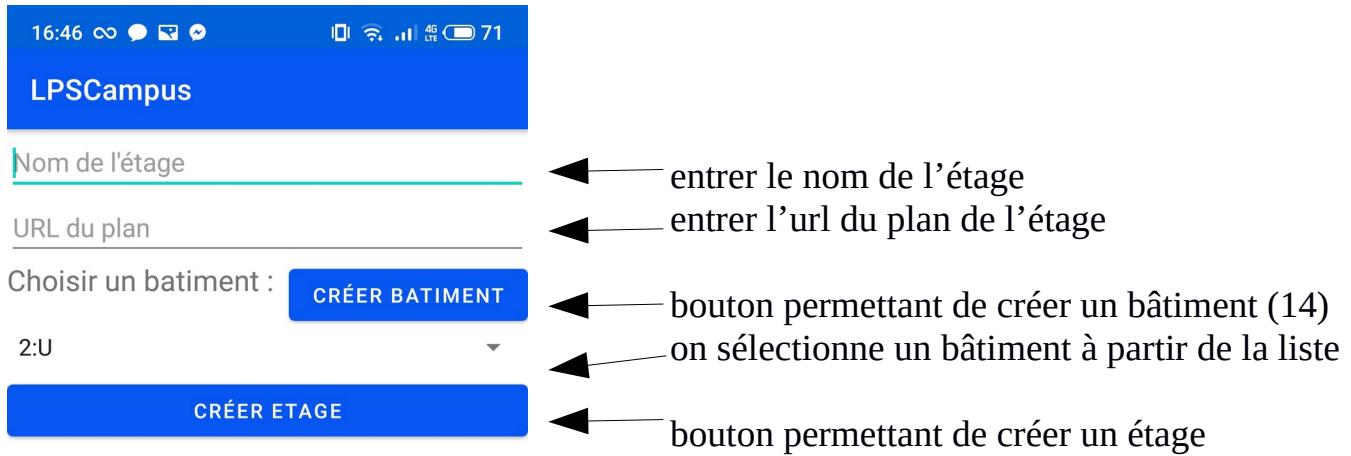
bouton permettant de lancer  
un scan

bouton permettant de créer une  
salle si elle n'est pas trouvée  
dans la liste (12)

## CreerSalle (12) :



## CreerEtage (13) :



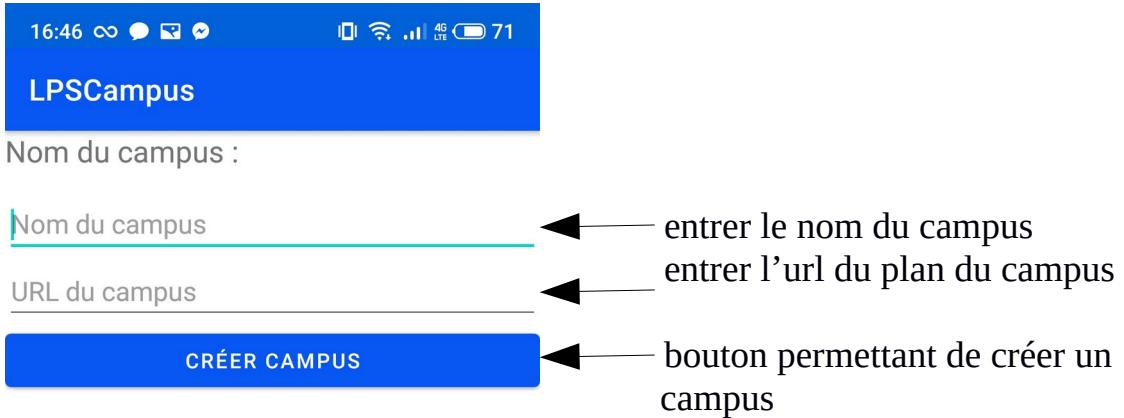
## CreerBatiment (14) :



- Nom du batiment entrer le nom du bâtiment
- Position en x entrer la position x
- Position en y entrer la position y
- Choisir un campus : bouton permettant de créer un campus (15)
- 1:tln on sélectionne un campus à partir de la liste
- CRÉER BATIMENT** bouton permettant de créer un bâtiment



## CreerCampus (15) :

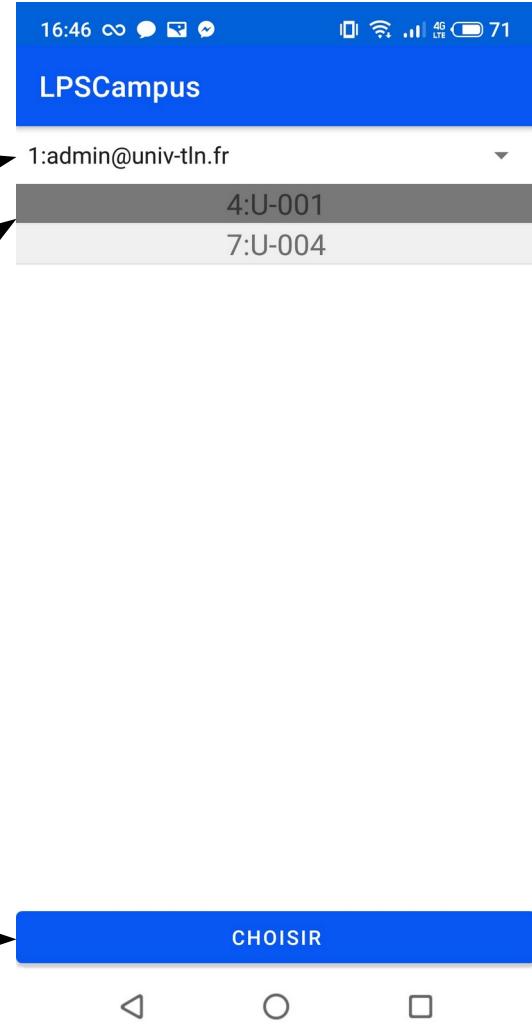


◀ ○ □

## AffichageSalles (16) : Superviseur



## Administrateur



choisir un superviseur afin de  
consulter ses salles et données

cliquer sur une salle pour la  
sélectionner

4:U-001  
7:U-004

1:admin@univ-tln.fr

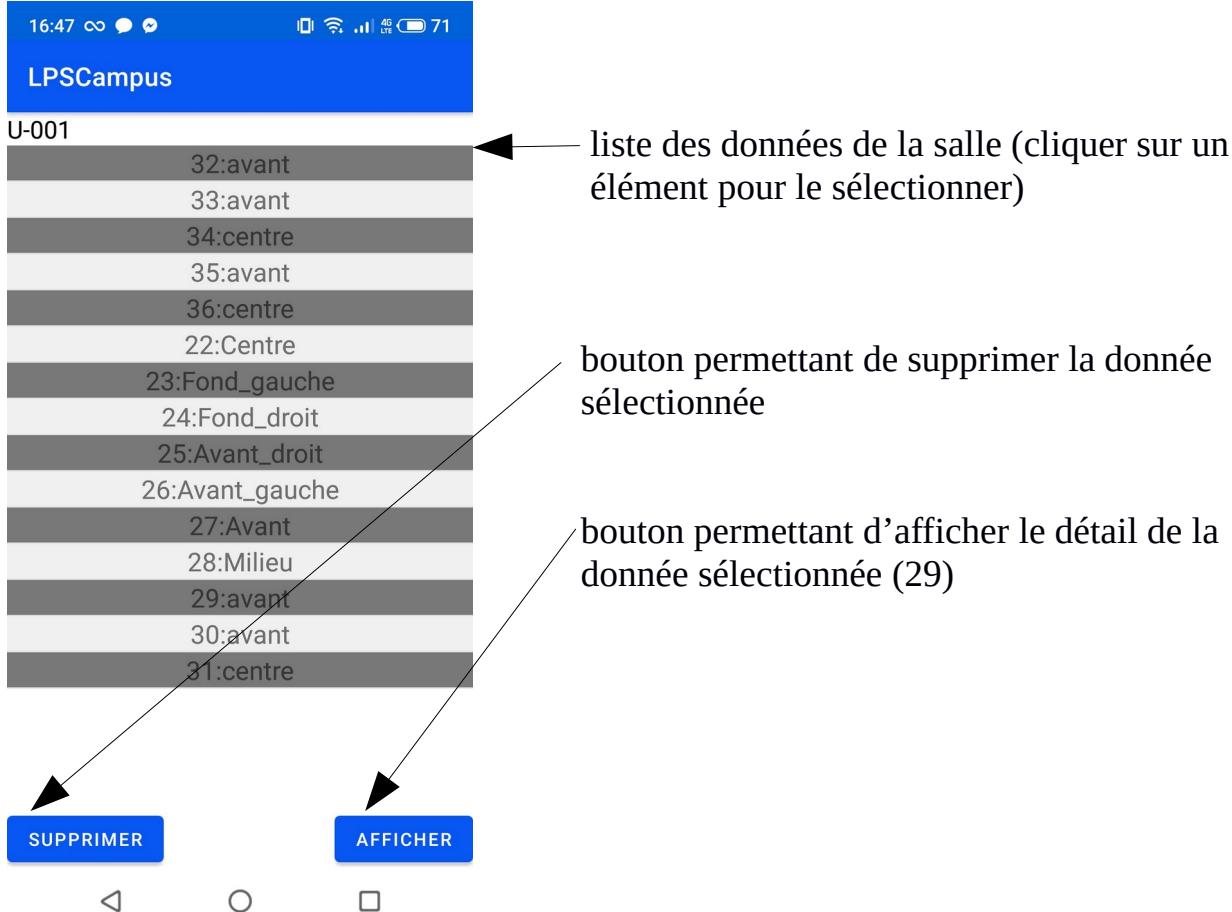
16:46 ∞ 4G 71

4:U-001

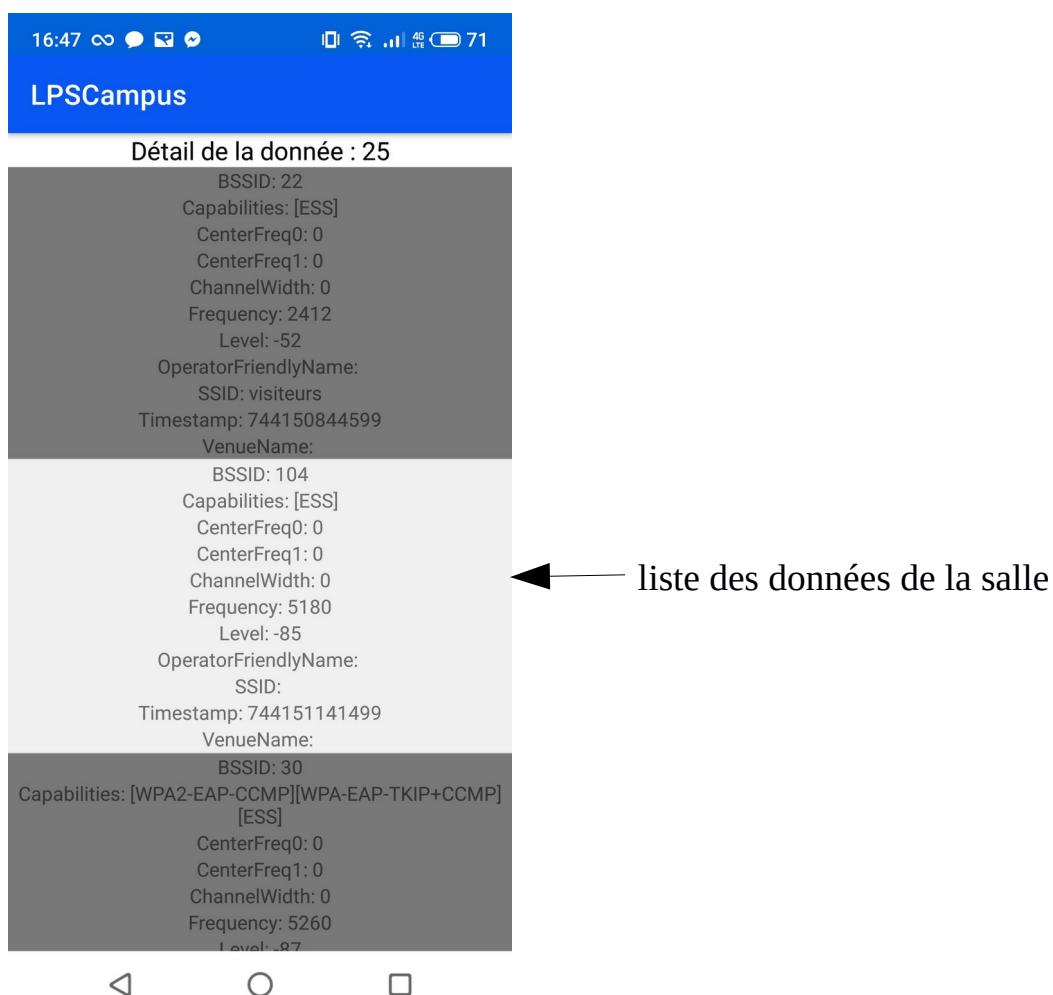
CHOISIR

CHOISIR

## AffichageDonneesSalle (17) :

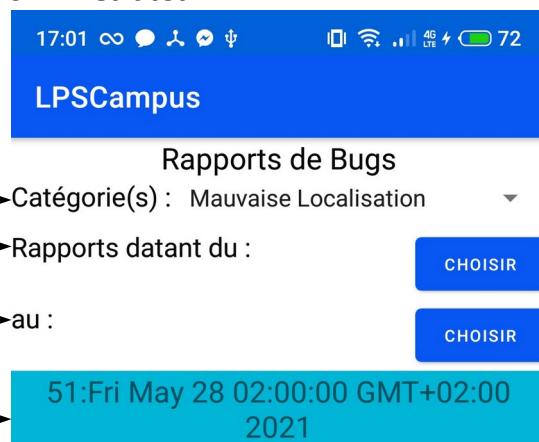


## AffichageDetailDonnee (29) :

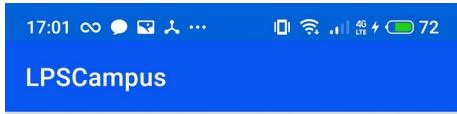


## VoirRapportBug (18) : Superviseur

Administrateur

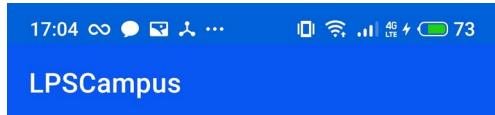


## AffichageDetailRapport (19) :



descriptif du rapport de bug

## VoirDemandes (20) :



liste des demandes pour devenir superviseur  
(cliquer sur un élément pour le sélectionner)

supprimer l'élément sélectionné

ACCEPTER

accepter la demande de l'élément sélectionné

REFUSER



## BDD (21):



REINITIALISER LA BDD

SUPPRIMER COMPTE SUPERVISEUR

SUPPRIMER CAMPUS

SUPPRIMER BATIMENT

SUPPRIMER ETAGE

SUPPRIMER SALLE

- bouton permettant de réinitialiser la bdd
- bouton permettant de supprimer un compte superviseur (22)
- bouton permettant de supprimer un campus (23)
- bouton permettant de supprimer un bâtiment (24)
- bouton permettant de supprimer un étage (25)
- bouton permettant de supprimer une salle (26)

## SupprimerCompteSuperviseur (22) :

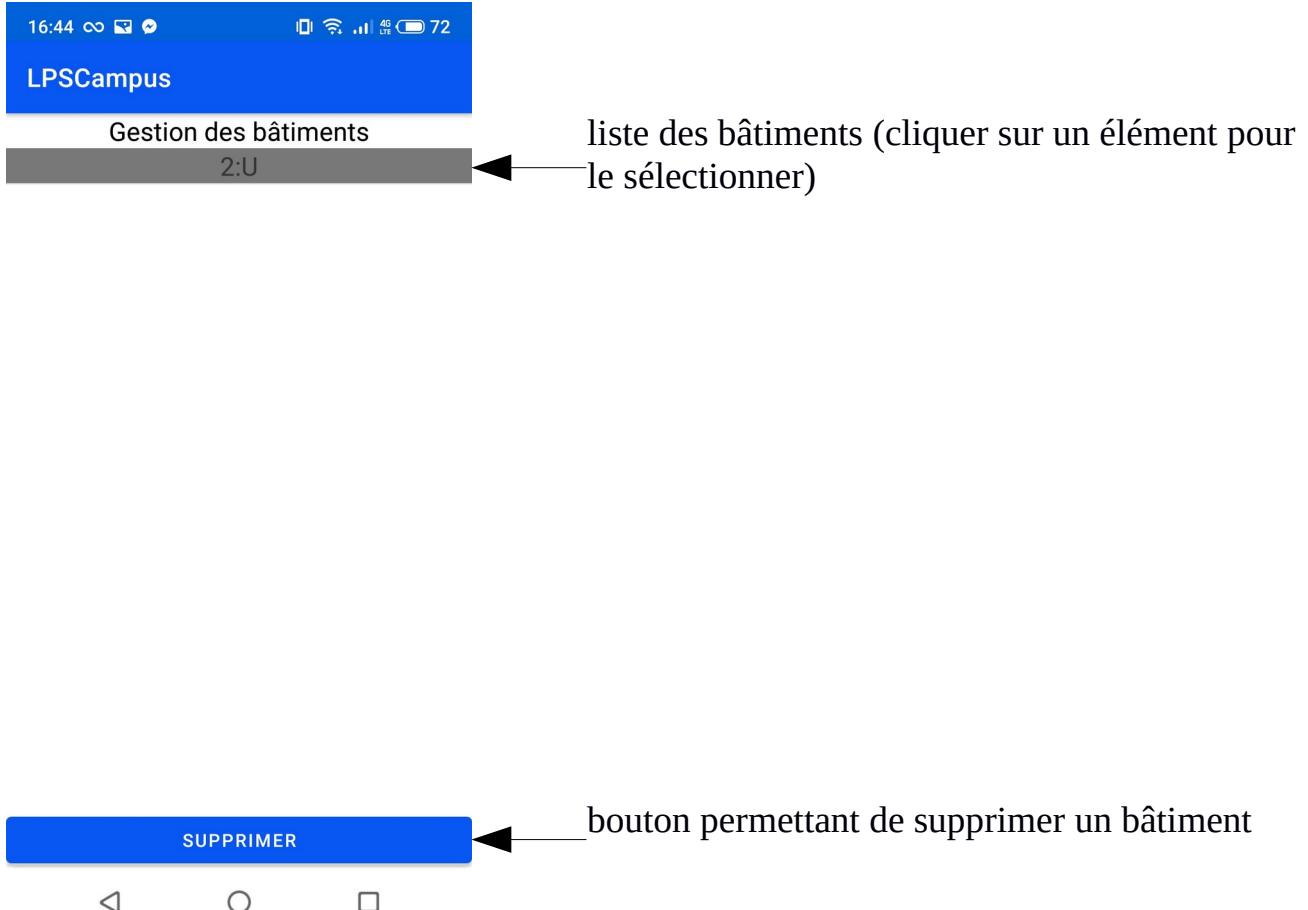


liste des superviseurs (cliquer sur un élément pour le sélectionner)

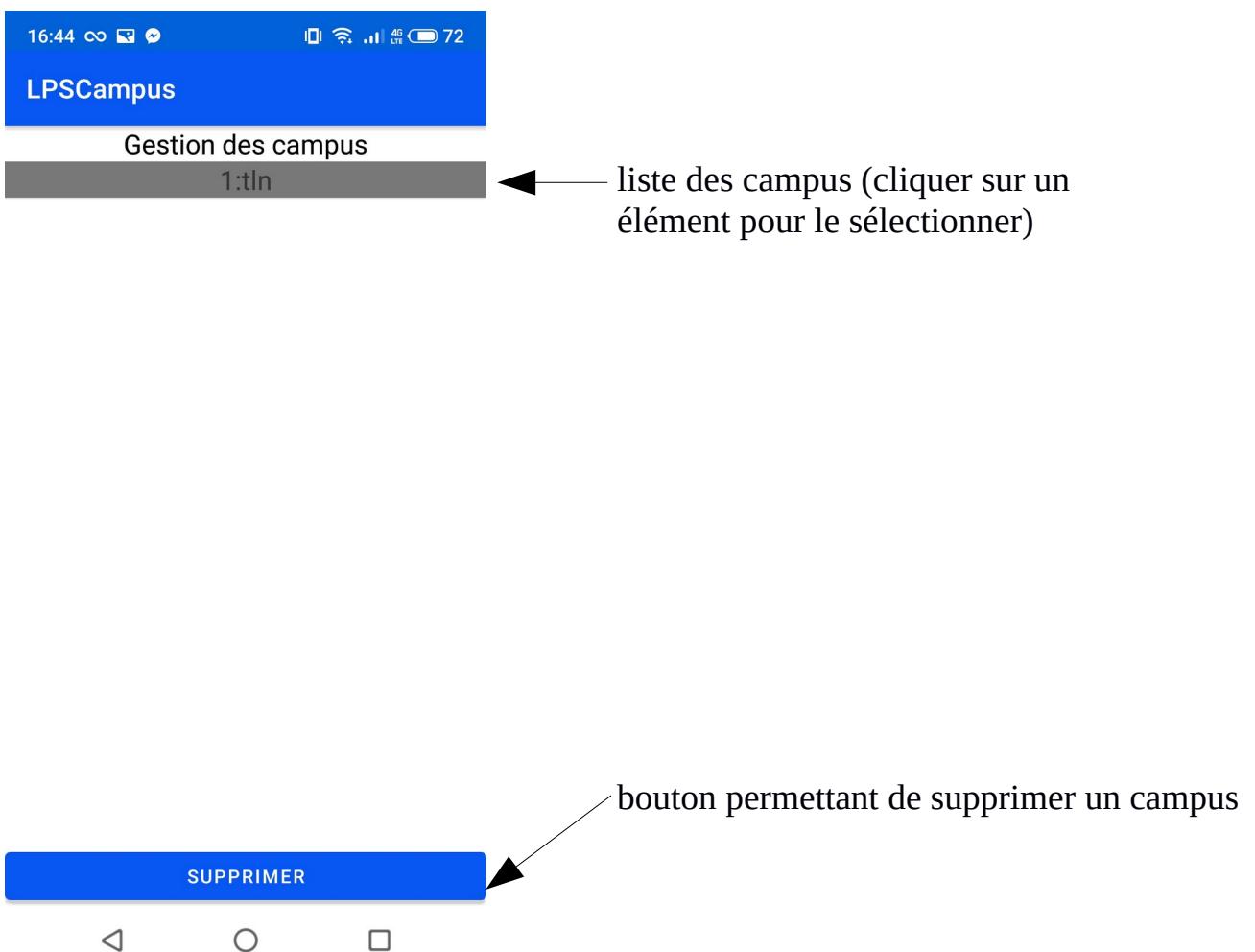
bouton permettant de supprimer un compte d'un superviseur

SUPPRIMER

### SupprimerBatiment (23) :



### SupprimerCampus (24) :



### SupprimerEtage (25) :



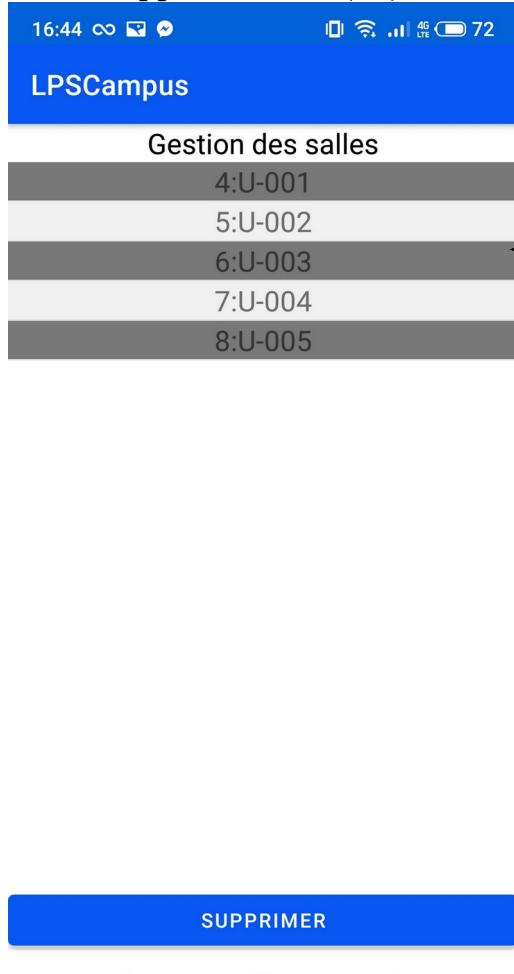
liste des étages (cliquer sur un élément pour le sélectionner)

SUPPRIMER

bouton permettant de supprimer un étage



### SupprimerSalle (26) :



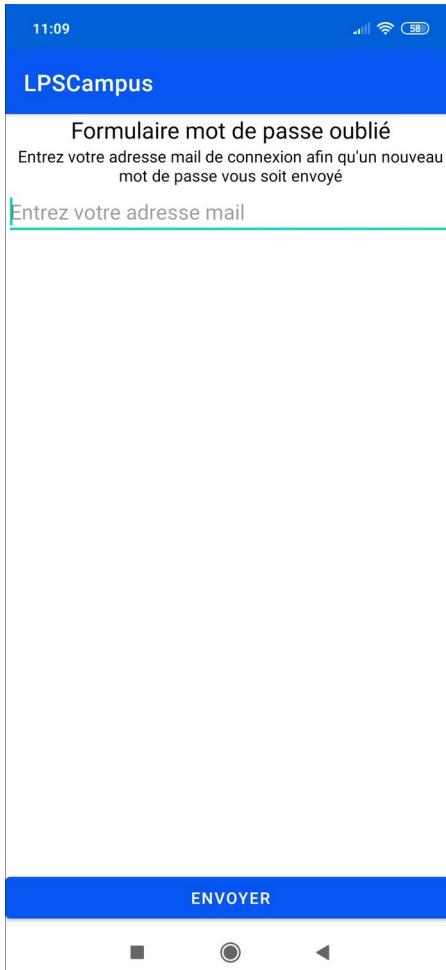
liste des salles (cliquer sur un élément pour le sélectionner)

bouton permettant de supprimer une salle

SUPPRIMER



## MPDOubli (27) :



saisir une adresse mail afin qu'un nouveau mot de passe vous soit envoyé à cette adresse

bouton permettant d'envoyer la demande de nouveau mot de passe

## DevenirSuperviseur (28) :



entrer son adresse mail universitaire

entrer un mot de passe

entrer de nouveau son mot de passe

bouton permettant d'envoyer sa demande pour devenir superviseur à l'administrateur

# Dictionnaire des termes et actions

Nous allons lister les points importants qui permettent de mieux comprendre le fonctionnement de l'application.

Voici une liste de termes importants :

- administrateur : personne qui s'occupe de gérer l'ajout, la suppression, ainsi que la consultation des données présentes dans la bdd. Elle s'occupe également de gérer les bugs qui sont remontés.
- superviseur : personne qui s'occupe de l'ajout de données à partir de pièces existantes, de consulter ses données ainsi que de la suppression de ses données.
- utilisateur : personne qui peut se localiser et remonter des bugs.
- wifi : informations caractérisant le wifi (BSSID et level).
- scan : liste de wifis pour une pièce.

Voici une liste des actions possibles :

- Pouvoir se connecter
- Pouvoir voir les demandes pour devenir superviseur
- Pouvoir ajouter une donnée pour une salle
- Pouvoir créer une salle
- Pouvoir créer un étage
- Pouvoir créer un bâtiment
- Pouvoir créer un campus
- Pouvoir visualiser les données d'une salle
- Pouvoir supprimer les données d'une salle
- Pouvoir supprimer les données d'un superviseur
- Pouvoir supprimer le compte d'un superviseur
- Pouvoir supprimer un campus
- Pouvoir supprimer un bâtiment
- Pouvoir supprimer un étage
- Pouvoir supprimer une salle
- Pouvoir réinitialiser la base de données
- Pouvoir se localiser
- Pouvoir choisir le mode d'affichage
- Pouvoir devenir superviseur
- Pouvoir reporter un problème