

# Vitrine

---

Rapport du projet de développement mobile

Loris Ceschin & Joaquim Perez  
HE-Arc ingénierie - INF3dlm-b  
24.01.2017

# Abstract

Ce projet est réalisé dans le cadre du cours de développement mobile de 3ème année Bachelor à la Haute Ecole Arc - Ingénierie, section Développement logiciel et multimédia.

Le projet consiste en la réalisation d'une application mobile sur Android, le sujet de l'application est libre. Nous avons choisi de développer un réseau social de partage de photos basé sur la localisation.

Le présent document décrit le déroulement du développement, le contenu et le fonctionnement de l'application, ainsi que les difficultés rencontrées et les évolutions possibles.

# Table des matières

<b>Abstract</b>	<b>2</b>
<b>Table des matières</b>	<b>3</b>
<b>Introduction</b>	<b>5</b>
Idée de base	5
Les Live Stories Snapchat	5
Concept	5
Exemples d'utilisation	5
<b>Guide utilisateur</b>	<b>6</b>
Authentification	6
Exploration	6
Abonnements	6
Contribution	6
Affichage de la vitrine	6
Se déconnecter	7
<b>Guide développeur</b>	<b>7</b>
Architecture	7
Description des activités et fragments	8
Cas d'utilisation	9
Première authentification	9
Nième authentification	9
Affichage des abonnements	9
Contribution	10
Création de Vitrine	11
Affichage des Vitrines proches	11
Communication avec le serveur	12
Modélisation de la base de données	12
Auto-critique du code	13
<b>Difficultés rencontrées</b>	<b>13</b>
Autorisation localisation	13
Requêtes http	13
Upload d'image en post	13
<b>Avenir de l'application</b>	<b>14</b>
<b>Bugs connus</b>	<b>14</b>
<b>Conclusion</b>	<b>15</b>



# Introduction

## Idée de base

Sur les réseaux sociaux populaires, tels que Facebook, Twitter et Instagram, il y a des fonctionnalités permettant aux utilisateurs de partager avec leur communauté l'endroit où ils se trouvent. Certains utilisateurs sont particulièrement friands de ces fonctionnalités. Cela montre que les gens ne sont pas intéressés qu'à des personnes, mais aussi à des lieux particuliers.

## Les Live Stories Snapchat

L'application Snapchat offre une fonctionnalité intéressante dans cette idée, les "Live Stories". De temps en temps, l'équipe de snapchat sélectionne un événement à mettre en avant (par exemple, un match de foot, un concert, ...). Les utilisateurs qui sont à l'événement peuvent publier des images et des vidéos dans le Live story. Tous les autres utilisateurs peuvent alors avoir un aperçu de l'événement en question. Cependant, il n'y a pas de possibilité de créer ses propres Live Stories, et les publications sont filtrées par l'équipe snapchat pour ne garder que les plus "politiquement correctes".

## Concept

L'idée du projet Vitrine est de reprendre le concept des "Live Stories" Snapchat en les étendant à tout le monde. C'est-à-dire que tous les utilisateurs peuvent créer un Live – une Vitrine – à l'endroit où ils se trouvent. Les utilisateurs pourront ensuite lister les vitrines près d'eux et en lire le contenu. Seuls les utilisateurs qui se trouvent dans le périmètre d'une vitrine peuvent y contribuer.

## Exemples d'utilisation

- Vendredi soir, 22h, Marc\* hésite à sortir. Il ouvre donc Vitrine et va voir si il reconnaît des gens dans les vitrines de ses deux bars préférés, il compare l'ambiance et pourra faire un choix.
- Josiane\* organise un événement dans sa rue. Elle crée donc une vitrine sur laquelle elle publie une première photo avec un message : "une boisson offerte à quiconque publie une photo de lui dans cette vitrine". Cela va motiver les gens à venir à son événement, et amené d'autres personnes qui auront remarqué l'évènement sur Vitrine.
- David\* sait que quand il boit, il a des amnésies, heureusement, il peut documenter toute sa soirée sur les vitrines où il a été. Ainsi, le lendemain, David pourra retracer son parcours et se remémorer d'excellents souvenirs.

\* prénoms d'emprunt

# Guide utilisateur

Ce chapitre explique les fonctionnalités de l'application, du point de vue de l'utilisateur.

## Authentification

Au premier lancement de l'application, il vous sera demandé de vous authentifier.

Il y a deux comptes à disposition (avec des données de base) pour:

username: loris      password: test

username: joaquim    password: joaquim

Après la première connexion, l'utilisateur est stocké en mémoire, il n'est donc pas nécessaire de se connecter à chaque lancement de l'application.

Vous pouvez aussi créer un compte en cliquant sur le bouton "Register", cela va ouvrir le navigateur web sur une page dédiée à cette fonctionnalité. Une fois que le message de confirmation apparaît, vous pouvez revenir sur l'application Vitrine.

## Exploration

Dans l'onglet "Explore", on peut se balader sur une carte, les vitrines sont représentées par des cercles sur la carte. On peut cliquer sur un cercle pour afficher le nom de la vitrine et un bouton pour l'ouvrir. Si une vitrine est "cachée" sous une autre, deux clics successifs permettront de mettre la vitrine du dessus en arrière plan.

Le bouton "Open" permet de voir le contenu d'une Vitrine et de s'y abonner/désabonner.

## Abonnements

Dans l'onglet "Subscribed", il y a une liste de vitrines auxquelles l'utilisateur est abonné.

Pour voir les photos de la vitrine, il suffit de cliquer dessus.

Lors d'un clic maintenu sur une vitrine, cela propose de voir la vitrine mais aussi de se désabonner.

## Contribution

Dans cet onglet, seules les vitrines dans lesquelles on peut publier sont présentes (basé sur notre position). Lors d'un clic sur une vitrine, l'appareil photo se lance. Dès que la photo est validée, elle est envoyée au serveur.

## Affichage de la vitrine

Dès que l'on affiche une vitrine, la première photo est chargée. Pour afficher la prochaine photo il suffit de cliquer dessus. Après la dernière photo, la première est de nouveau affichée et la boucle recommence. Vous pouvez alors revenir en arrière pour quitter la galerie photo.

[Se déconnecter](#)

Le bouton menu en haut à droite de l'application permet de vous déconnecter. Cela va quitter l'application. Cela permet de changer d'utilisateur ou d'en créer un nouveau, au prochain lancement de l'application.

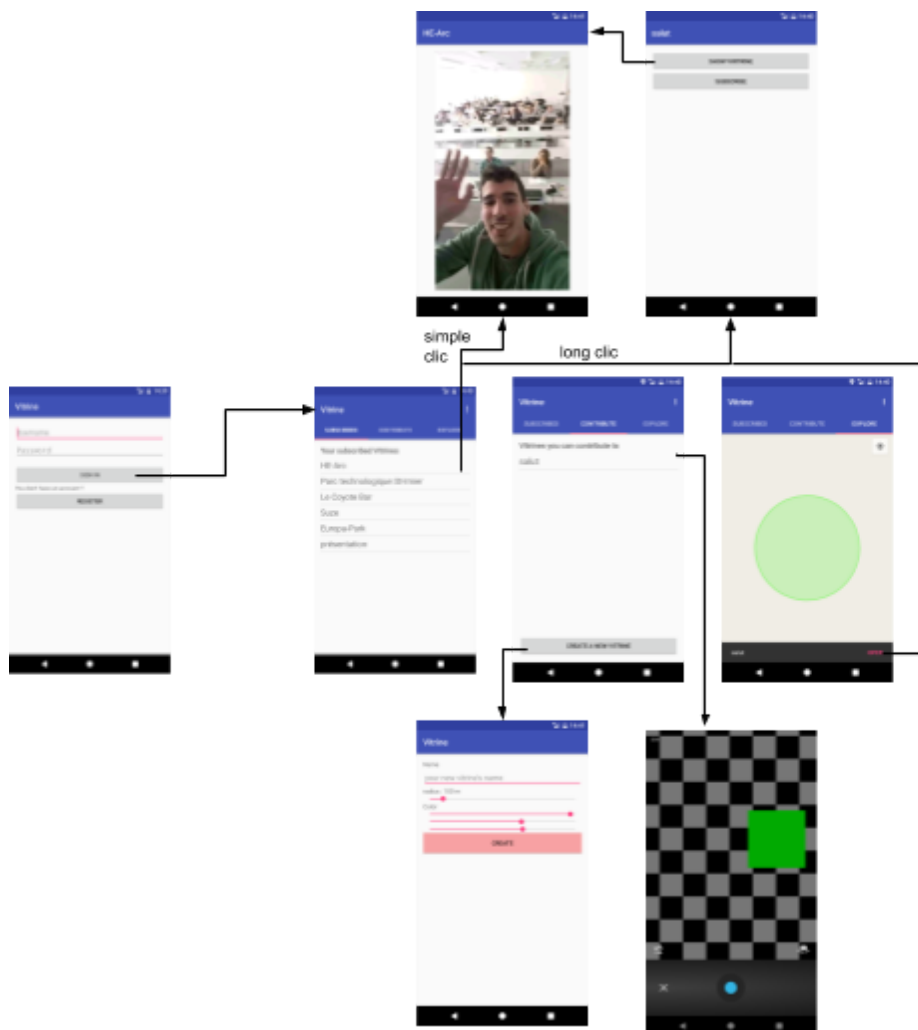
# Guide développeur

Ce chapitre décrit le fonctionnement de l'application et du serveur d'un point de vue développeur. Ceci peut faciliter le travail à quelqu'un qui voudrait reprendre le développement de l'application.

# Architecture

L'application contient six activités et trois fragments.

Le diagramme ci-dessous représente les interactions entre les activités/fragments.



## Description des activités et fragments

### LoginActivity

Activité pour se connecter à l'application. Elle ouvre TabActivity lorsque la connexion est établie (et passe l'utilisateur à travers l'intent)

### TabActivity

Activité qui contient un ViewPager pour afficher les trois fragments.  
Elle contient le googleApiClient pour récupérer la location

### CreateActivity

Simple activité pour créer une vitrine

### ContributeActivity

Activité de traitement, dès qu'elle est ouverte, appelle l'appareil photo. Ensuite compresse la photo et l'envoie au serveur

### VitrineInfoActivity

Permet de voir le contenu de la vitrine et de s'abonner / désabonner

### VitrineActivity

Affiche les photos de la vitrine

### SubscribedVitrineFragment

Contient une liste des vitrines auxquelles nous sommes abonné, elle sont affichées avec un adapter. Lors d'un clic sur une vitrine, on ouvre VitrineActivity avec l'objet vitrine dans l'intent.

Si on maintient le clic sur un élément, on ouvre VitrineInfoActivity.

### ContributeFragment

Contient la liste des vitrines auxquelles nous pouvons contribuer. Autrement dit, les vitrines dans lesquelles nous sommes actuellement (par la position). Il y a un bouton pour créer une nouvelle vitrine a cet emplacement (CreateActivity).

Lorsque l'on clique sur une vitrine, on ouvre ContributeActivity

### DiscoverFragment

Contient la carte sur laquelle nous avons ajouté des cercles cliquables représentant les vitrines. Lors d'un clic sur un cercle, la carte zoom pour afficher le cercle en "fullscreen" et affiche une snackbar permettant d'ouvrir VitrineInfoActivity.

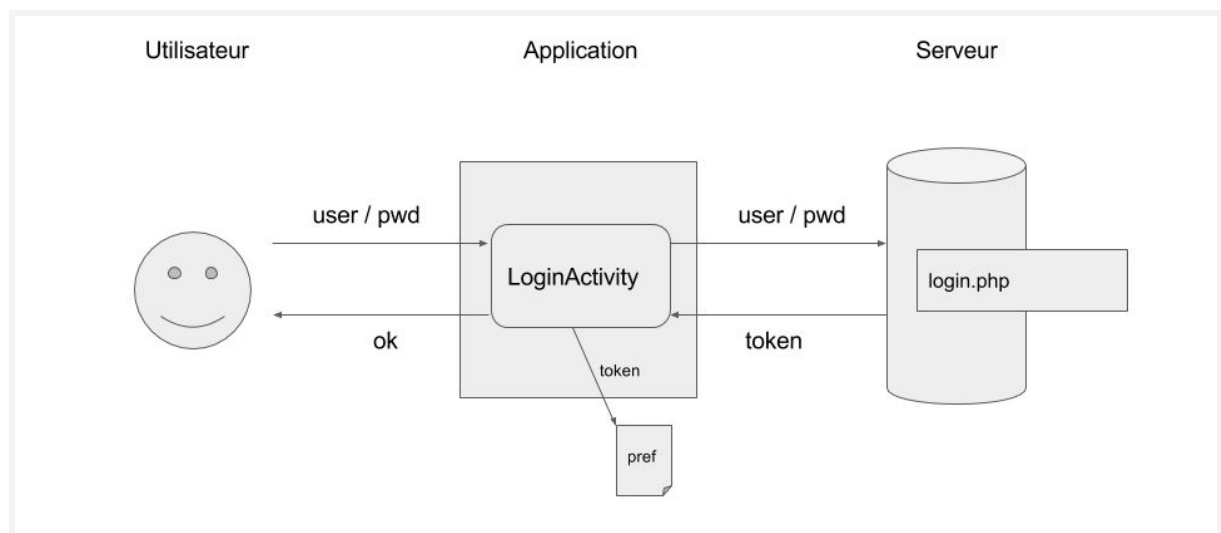


## Cas d'utilisation

### Première authentification

Lorsque l'utilisateur lance l'application pour la première fois, il lui sera demandé de se connecter avec son compte Vitrine. Pour éviter qu'il doive faire ce travail à chaque lancement de l'application, nous avons implémenté un système de **token**. Le script PHP *login.php* vérifie le couple *username/password* et retourne un token associé à l'utilisateur si les informations sont correctes.

L'application va alors sauvegarder ce token dans un fichier de préférences.

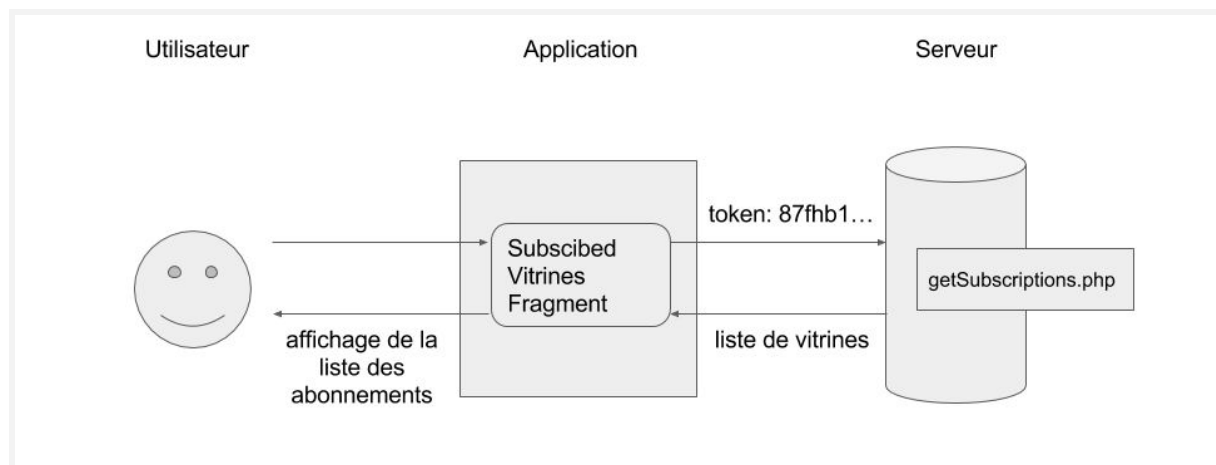


### Nième authentification

Au lancement, l'application va vérifier l'existence du fichier de préférence. Si un token existe, elle lance l'activité principale. L'utilisateur sera identifié uniquement grâce à ce token. Si le token n'existe pas, on est dans le premier cas.

### Affichage des abonnements

Le fragment *SubscribedVitrinesFragment* opère une requête http GET sur le script *getSubscriptions.php*. Le serveur récupère les vitrines de l'utilisateur identifié par le token passé en argument de la requête. Le résultat est renvoyé sous forme d'une liste encodée en JSON.

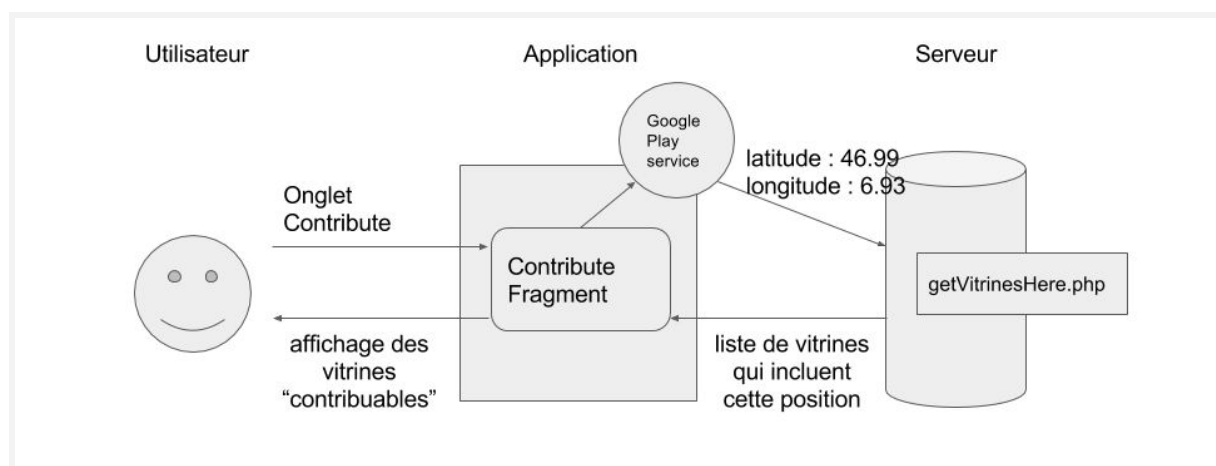


## Contribution

Le fragment *ContributeFragment* doit afficher les Vitrines dans lesquelles l'utilisateur se trouve. Pour cela, on récupère la dernière position connue à partir du service Google Play. On envoie ensuite une requête au serveur avec la latitude et la longitude correspondant à la position de l'utilisateur.

Sur le serveur, dans le script *getVitrineHere.php*, une requête SQL complexe permet de sélectionner les vitrines qui incluent ce point GPS.

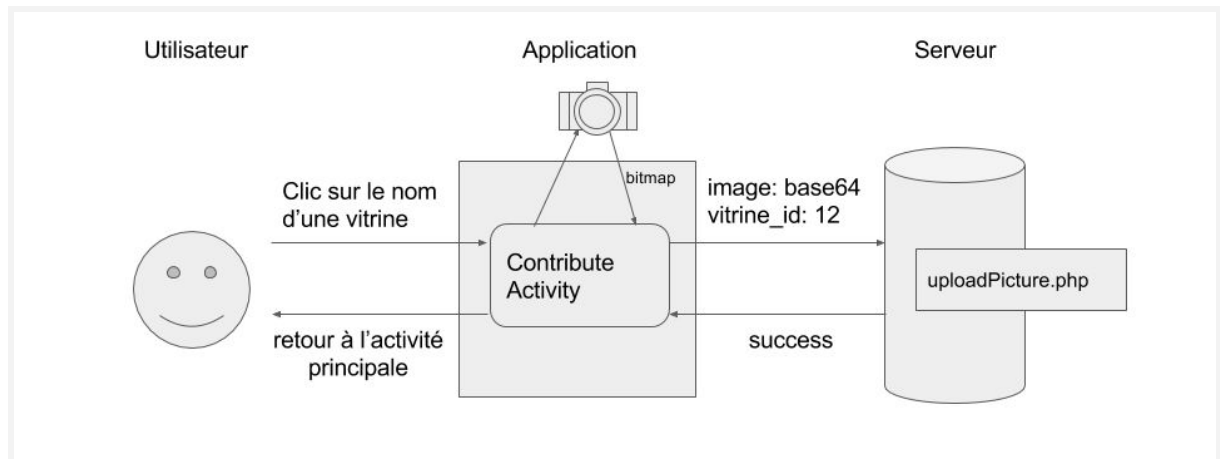
Les vitrines dans lesquelles l'utilisateur peut contribuer lui sont alors affichées.



L'utilisateur peut alors cliquer sur l'une des vitrines, cela va ouvrir l'appareil photo. L'image est récupérée depuis le système de fichier, traitée (redimensionnée, compressée, pivotée), puis encodée en format *base64*<sup>1</sup>. Le script *uploadPictures.php* décode l'image, la sauvegarde dans l'arborescence du serveur et insert une ligne dans la table *pictures* de la

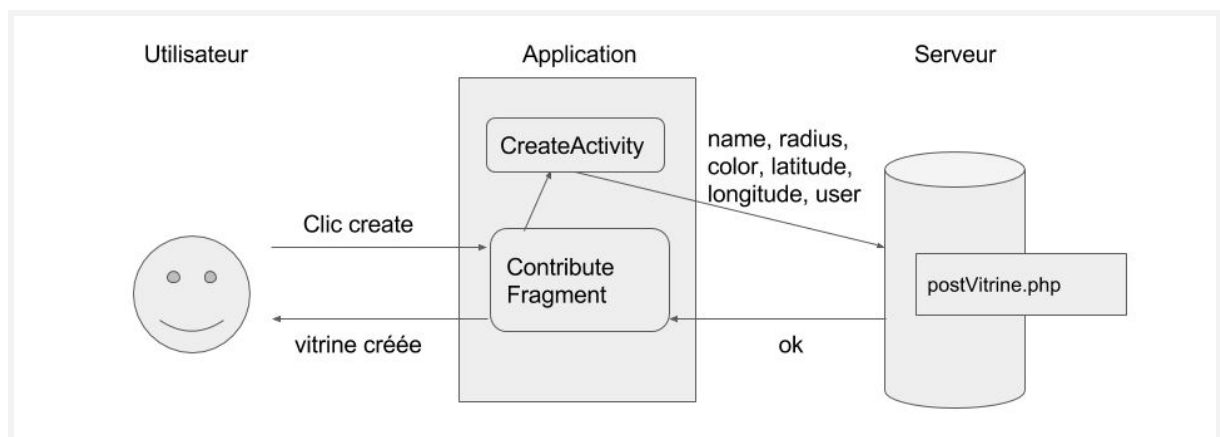
<sup>1</sup> base64 est un encodage qui permet de convertir n'importe quelle donnée binaire en texte. Nous utilisons cela car il est plus pratique d'envoyer du texte que des données binaires (HTTP est un protocole texte)

base de données. Le résultat est affiché à l'utilisateur dans un *Toast*, Il indique si l'insertion de l'image a réussi ou échoué.



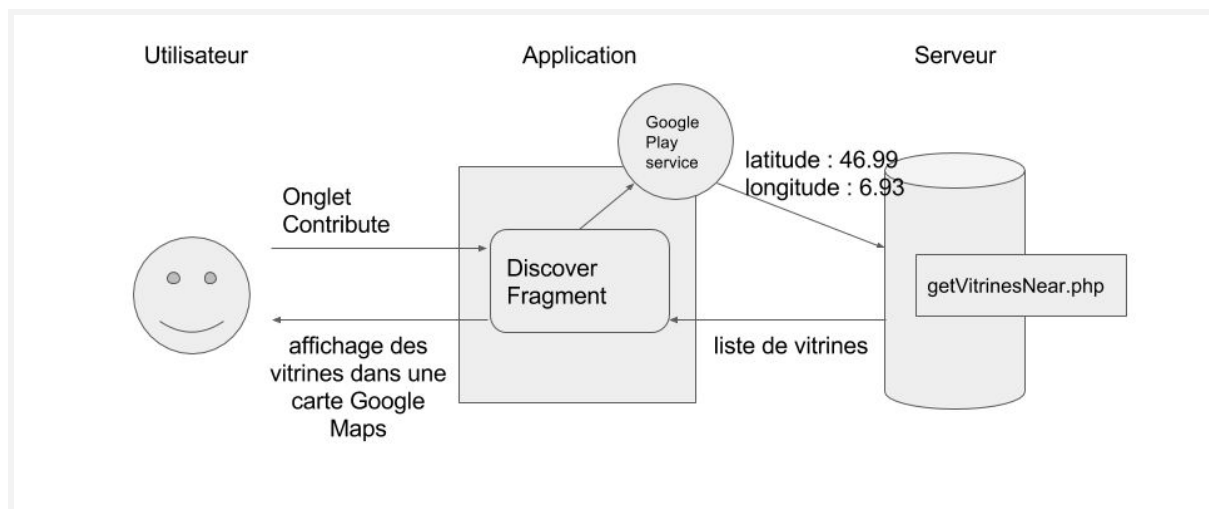
## Création de Vitrine

Le fragment *ContributeFragment* permet aussi de créer une Vitrine. L'utilisateur doit choisir un nom, une couleur et un rayon. Ces informations sont envoyées au serveur avec la position actuelle de l'utilisateur.



## Affichage des Vitrines proches

Le dernier fragment *DiscoverFragment* affiche les N vitrines les plus proches (pour l'instant, il affiche toutes les vitrines) dans une carte Google Maps. Le script *getVitrinesNear.php* effectue une requête SQL qui récupère les N vitrines en les triant par ordre de distance avec le point donné.



## Communication avec le serveur

La communication avec le serveur se passe comme cela:

Passer des paramètres pour avoir des données en retour : GET  
par exemple: `getVitrineHere.php?lat=XXXX&lng=XXXX`

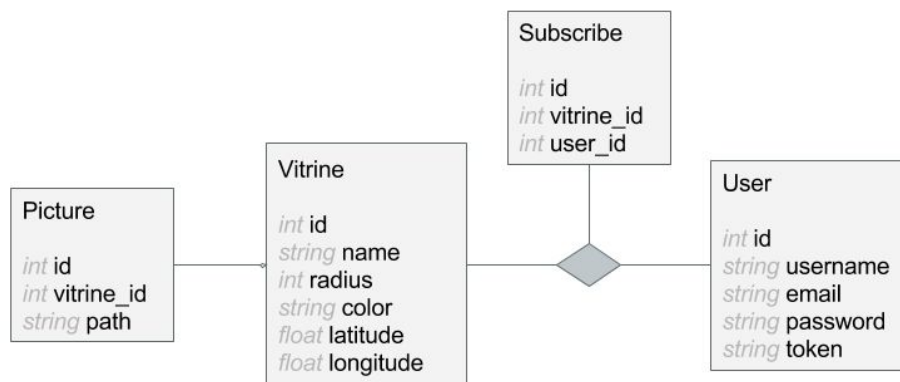
Passer des paramètres pour créer quelque chose : POST  
par exemple: `postVitrine.php`

Le serveur est hébergé par un fournisseur d'hébergement gratuit, nexgate.ch. Les accès à la base de données se font en PHP avec l'interface PDO. La base de données est MySQL. Des instructions concernant le déploiement de l'application côté-serveur se trouvent dans le fichier README.md.

## Modélisation de la base de données

La structure de la base de données est plutôt simple. Il y a deux entités principale : Les Vitrines et les Utilisateurs. Une vitrine a un nom, une position, un rayon et une liste de photos. Un utilisateur a des informations de login, un token et une liste d'abonnements. La liste d'abonnements est gérée par une table annexe plusieurs-à-plusieurs appelée *Subscribe*, elle met en relation des vitrines et des utilisateurs.

Une image a juste un chemin de type `"uploads/{vitrine_id}/{filename}.jpg"` qui permet de la télécharger.



## Auto-critique du code

Le code n'est pas très bien structuré, il ne respecte pas de convention de nommage et rigueur, c'est souvent du code "spaghetti".

Il y a des répétitions de code. Par exemple, plusieurs activités effectuent des requêtes HTTP, et le code est très semblable. On pourrait par exemple créer une classe NetworkTools qui réunit les méthodes de communication avec le serveur.

## Difficultés rencontrées

### Autorisation localisation

Les outils que proposent Android pour gérer les autorisations d'accès aux capteurs de l'appareil sont assez compliqués à prendre en main. Ce problème nous a pris pas mal de temps à résoudre (d'ailleurs, il n'est pas tout à fait résolu, voir Bugs connus).

### Requêtes http

Au départ, nous avons utilisé HttpURLConnection pour récupérer du JSON. Cela fonctionnait bien pour faire des requêtes GET mais lorsque nous avons dû passer la photo à upload en POST c'est devenu très vite compliqué. Suite à ça, nous sommes passé sur Volley ce qui a simplifié grandement la chose.

### Upload d'image en post

Après l'upload, les images étaient corrompues. Le problème venait du fait que les requêtes post étaient limitées en taille par le serveur. N'ayant pas la possibilité de changer ce paramètre sur le serveur, nous avons scale les images pour que la taille soit moins grande.

## Avenir de l'application

Nous pensons sincèrement que l'application a des perspectives d'avenir sérieuse. C'est un concept nouveau qui peut potentiellement intéresser beaucoup de monde, notamment chez les jeunes.

Voici une liste non-exhaustive de fonctionnalités à ajouter pour que l'application soit plus complète :

- **Suppression des vitrines.** Pour l'instant les vitrines sont créées mais jamais effacées. C'est un problème car ça risque de "polluer" la carte avec des vitrines inactives.
- **Recherche.** On ne peut trouver des vitrines que en se baladant sur la Google Maps. Il serait pratique de pouvoir rechercher les vitrines par nom, par exemple.
- **Vidéo.** Les images, c'est bien mais parfois, les vidéos, c'est mieux.
- **Textes.** Il serait intéressant de pouvoir écrire des textes par dessus une photo, comme on peut le faire sur Snapchat, par exemple.
- **Signatures.** Pour l'instant, les photos sont anonymes. On pourrait afficher le nom de l'utilisateur sur les images dans une vitrine.
- **Modération.** N'importe qui peut ajouter n'importe quoi à une vitrine. Un système de signalement par les autres utilisateurs permettrait de supprimer les images inappropriées et bannir les mauvais utilisateurs s'ils récidivent.
- **Design et ergonomie.** Nous n'avons pas beaucoup travaillé sur cet aspect, l'application n'a pas de "caractère" particulier qui la démarquerait des autres applications (thème Google par défaut, etc...).
- **Éphémérité** (durée de vie des images et des vitrines). Les images restent dans les vitrines pour toujours, il serait intéressant de supprimer les images automatiquement après un certain laps de temps, comme sur Snapchat. L'idée de la photo qui disparaît après 24 heures est bonne car elle incite les utilisateurs à se connecter à l'application tous les jours, pour ne pas risquer de manquer quelque chose.

## Bugs connus

- Rotation de l'écran fait crasher l'application (plus ou moins patché en forçant le mode portrait dans le manifest)
- Au premier lancement, après avoir autorisé le service de localisation, l'application risque de crasher aussi. Cela est dû au fait qu'on essaie d'accéder au GPS avant d'avoir les droits nécessaires. Nous n'avons pas eu le temps de résoudre ce problème qui est apparemment plus complexe qu'il n'y paraît.
- Il peut arriver que le lancement de l'appareil photo pose problème aussi. Si c'est le cas, il faut donner les autorisations nécessaires à l'application depuis les préférences.

# Conclusion

Ce projet nous a permis d'approfondir de nombreux aspects du développement mobile Android. Nous avons utilisé beaucoup de périphériques et capteurs de l'appareils (GPS, système de fichier, appareil photo, préférences) et touché à diverses technologies, ce qui n'était pas de tout repos.