

Fonctionnement des applications

Pour ces versions, dans un premier temps les utilisateurs renseignent la distance minimale qu'ils souhaitent être par rapport aux autres utilisateurs et leurs villes. Les applications récupèrent en temps réelle les positions et déterminent si la distance minimale est respectée. Si cette distance n'est pas respectée les applications lanceront une animation indiquant ce fait et fait vibrer le support (configuration du navigateur nécessaire pour les vibrations pour l'application web).

Fonctionnement technique

Les applications génèrent des identifiants unique à chaque utilisateur et chaque utilisation. Des autorisations pour les deux applications sont nécessaires afin d'obtenir la position. Les applications ne stockent que temporairement les positions et les supprimes après chaque utilisation. Les villes servent à filtrer les utilisateurs afin de faciliter les calculs des distances. Bien que ces dernières peuvent être récupérer à partir des positions, je les demandes dans ces versions pour éviter de nouveaux calculs.

L'application Android utilise l'application web.

Technologies

Les technologies utilisées sont les suivantes:

- Python: Flask comme framework (déjà utilisé pour d'autres projets), Google Maps pour le calcul des distances (les positions sont récupérées avec les navigateurs), connecteur MySQL pour les interactions avec la base de données. Gunicorn comme serveur web.
- JavaScript: jQuery, Ajax pour les protocoles de communication avec le serveur et les transitions.
- HTML5: pour l'interface.
- CSS: pour les styles de l'interface et animations.
- Bootstrap: pour la mise en forme de l'interface.
- SQL: pour la création et l'interaction avec la base de données.
- Android: pour l'application mobile.
- Google Cloud: App Engine pour le déploiement.
- db4free.net: stockage gratuit des données dans MySQL pour tester les applications.

Difficultés rencontrées

En local, l'application web marche bien mais une fois déployer elle ne supporte qu'une connexion. Et n'exécute pas les calculs de façon synchrone. Je pense que ceci est du au framework Flask car j'ai changé de serveur mais j'observe le même problème. Une solutions possible est l'utilisation d'un framework plus robuste que Flask comme Django.

La géolocalisation n'est pas très précise.

Installations de l'application mobile

Pour l'application web: vous pouvez la tester en local en installant les dépendances dans le fichier **requirements.txt** et exécuter la commande **flask run**.

Pour l'application mobile: configurer en amont votre téléphone pour pouvoir installer des applications de sources inconnues i.e ne provenant pas de Google Play Store. Pour ce fait, activer cette option dans vos paramètres.

Télécharger le fichier apk sur votre mobile, retrouver le fichier dans le dossier où vous l'avez téléchargé et cliqué dessus pour lancer l'installation. Une fois installer, données les autorisations nécessaires (ex. accès à la position).

Tester les applications

Vous pouvez tester les applications avec deux téléphones en mettant la même ville dans le formulaire.

Bonnes pratiques

Pour l'application web, à la première demande d'accès à votre position, enregistrer ce choix.