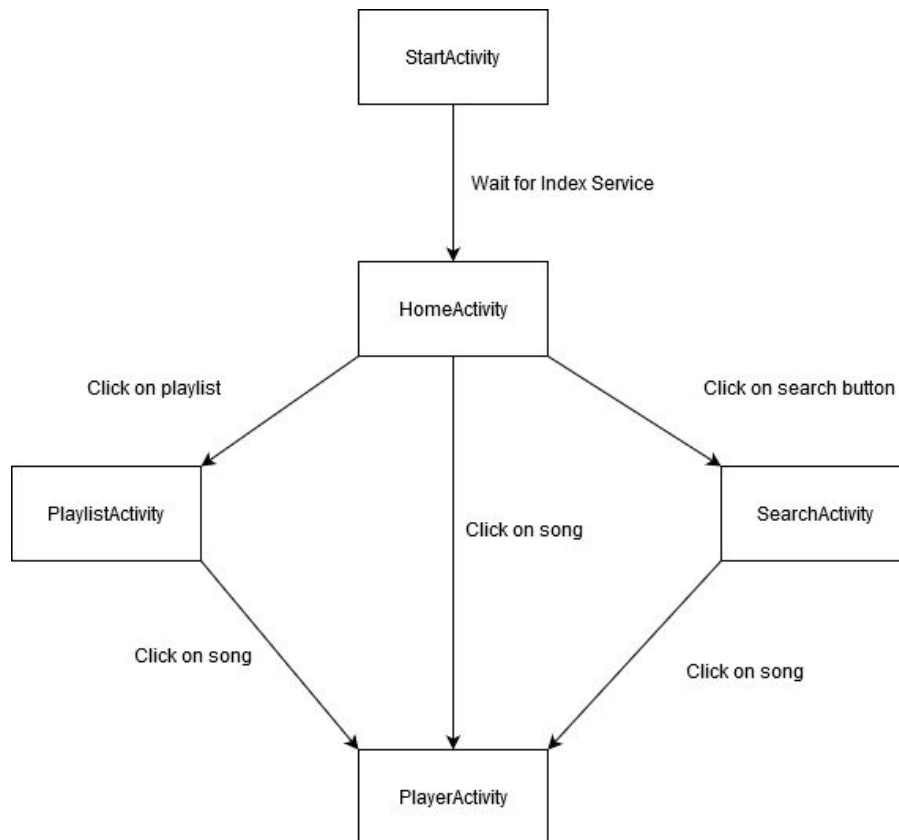


SOUNDROID

Rapport de développement

L'objectif de ce document est de présenter l'état du développement du projet.

I. Architecture



Le fonctionnement de l'application suit le schéma précédent. L'indexation des fichiers musicaux est effectuée à chaque démarrage de l'application, et ces fichiers sont enregistrés dans une base de donnée locale persistante.

Une fois cette étape terminée, l'utilisateur arrive sur la HomeActivity, permettant de naviguer entre les chansons, les playlists existantes, et la recherche.

Quand il clique sur une chanson, une activité de lecture s'ouvre, qui lit le morceau sélectionné. La lecture s'effectue dans un service Android.

L'utilisateur peut également rechercher dans la base de données des chansons selon certains critères.

II. Fonctionnalités implémentées

- **Base de pistes musicales / Indexation Niveau 3** : indexation des métadonnées des pistes musicales présentes dans les répertoires publics de l'appareil Android dans une base de données SQLite (Room) via le MediaMetadataRetriever.
- **Recherche de pistes Niveau 1** : interface de recherche permettant à l'utilisateur de réaliser des recherches sur un seul critère.
- **Etiquettes de piste Niveau 2** : mise en place d'un système de like permettant à l'utilisateur de gérer une playlist spéciale contenant les titres aimés
- **Listes de lecture Niveau 1** : utilisateur peut organiser ses morceaux en listes de lecture.
- **Lecteur musical Niveau 2** : activité permettant de contrôler les pistes musicales à lire. L'utilisateur peut choisir des listes à lire et a la possibilité de liker le morceau courant depuis cette activité.
- **Service de lecture Niveau 2** : lecture des pistes de musiques via un service afin d'assurer la lecture en arrière-plan
- **Pause automatique Niveau 3** : mise en pause de la musique lorsque le périphérique reçoit un appel

III. Fonctionnalités manquantes

Historique de lecture : bien qu'une playlist dédiée ait été créée, la sauvegarde de l'historique de lecture n'a pas été implémentée.

Playlists: des playlists sont créées automatiquement à l'indexation des chansons :

- une playlist *Musics* : contient toutes les musiques trouvées dans l'appareil.
- une playlist par artiste : contient tous les morceaux d'un artiste donné.
- une playlist par album : contient tous les morceaux d'un album donné.

Il n'est cependant pas possible d'ajouter manuellement des chansons dans une playlist, ni de créer une playlist manuellement.

Lecteur musical : Les boutons permettant d'activer la lecture aléatoire ou la lecture en boucle ne sont pas fonctionnels.

Base de donnée / Indexation : A chaque indexation, les morceaux sont réécrits dans la base, même s'ils y sont déjà. Pour éviter de remplir la base de doublons, la décision a été prise de supprimer tout son contenu avant chaque indexation. Cela a pour conséquence de ne pas enregistrer (par exemple) les likes donnés sur les morceaux de manière totalement persistante.

Recherche : Absence de message d'erreur / d'info dans le cas où la recherche ne renvoie pas de résultat.

Synthèse vocale/Historique de lecture/Sauvegarde et import de la base/Exposition de la base/Partage musical: Non implémentés.

IV. Répartition du développement

Après concertation autour des compétences de chacun, nous avons commencé par répartir les tâches de la manière suivante :

- Kevin souhaitait traiter l'ensemble des interfaces
- Tanguy souhaitait designer la base de données SQL
- Rémi souhaitait penser la structure projet et lister les différents services nécessaires.

Nous nous sommes réunis plusieurs fois afin de discuter les structures de données qui nous permettraient de faire communiquer chaque partie du projet. Après avoir pensé la base de données, Tanguy a rejoint Rémi sur l'implémentation des différentes fonctionnalités et services.

V. Difficultés rencontrées

Rémi : Lors de l'utilisation du MediaMetadataRetriever, je ne suis pas parvenu à extraire les images bitmap depuis les morceaux de musique.

Kevin : Difficultés à rendre les interfaces responsive par le principe que tous les téléphones ne possèdent pas tous la même résolution. De plus, selon le type de layout choisis, les contraintes de dispositions pouvait rendre l'interface complètement illisible.

Tanguy : L'implémentation du MediaPlayer en tant que service a pris beaucoup de temps, et en prenant du recul sur ce qui a été fait, je pense que l'architecture mise en place n'était pas adaptée. Le soucis était qu'il fallait faire le lien entre l'activité de lecture et la notification qui permettent toutes deux d'agir sur la musique. L'unification des deux n'est pas propre.

VI. Bilan

Ce projet nous a permis de mettre en avant une large palettes de compétences. Nous avons bien sûr progressé sur les notions apportées par le cours mais nous avons également fait appel à des connaissances d'algorithmique et de gestion de base de données.

De plus, nous avons eu l'occasion de manipuler plusieurs API d'Android que nous n'avions pas pu voir en cours. C'est donc un projet qui s'est révélé très intéressant pour nous trois et qui mérite d'être approfondi sur le temps libre.