

Documentation DictionaryAdminApp

Minh Kha NGUYEN

Janvier 2022

1 Introduction et les idées générales du projet

Ce projet est une application dictionnaire mobile. L'utilisateur va pouvoir Utiliser une application Android qui communique avec un service web qui comporte une base de donnée dictionnaire. Le projet utilisera principalement le langage JAVA.

2 Définition des fonctionnalités

2.1 La base de donnée

La base de donnée dictionnaire est un dictionnaire Anglais, récupérée depuis [ce site](#) pour des raisons de facilité. La base de donnée comporte :

- Table “entries”
 - “word” de type varchar(25)
 - “wordtype” de type varchar(20)
 - “definition” de type text

2.2 Le service web

Le service web récupère une requête de l'utilisateur puis se connecte avec JDBC vers la base de donnée.

Le service web possède les classes suivantes:

- Model, *le modèle d'une donnée de la base*
 - String word, *le mot du dictionnaire*
 - String type, *le type du mot à rechercher*
 - String definition, *la définition de ce mot*
- Resource, *traduction de la requête de l'utilisateur selon un verbe HTTP*
 - créer un *Service* et l'exécuter avec la requête de l'utilisateur

- Service, *définir et créer la connexion vers la base de donnée*
 - Objet Connection con, *l'objet qui va connecter le web service vers la base de donnée*
 - getMySQLConnection, *définir l'objet connexion JDBC vers la base de donnée*
 - Constructeur CrudService(), *affectation de l'objet selon les paramètres définis dans getMySQLConnection*
 - CrudModel getEntries, *une fonction qui prépare la requête SQL avec le mot à rechercher entré en paramètre*
 - CrudModel addEntries, *une fonction qui prépare la requête SQL pour ajouter dans la base de donnée avec le mot entré en paramètre*
 - CrudModel deleteEntries, *une fonction qui prépare la requête SQL pour supprimer dans la base de donnée avec le mot entré en paramètre*
 - CrudModel updateEntries, *une fonction qui prépare la requête SQL pour modifier un mot dans la base de donnée avec le mot entré en paramètre*

2.3 L'interface utilisateur (Android)

Une application simple à deux activités:

- LauncherActivity
 - L'activité que l'application va lancer en premier pour faire des vérifications de mise à jour.
- MainActivity
 - Dans cette activité, l'utilisateur va pouvoir choisir l'opération qu'il veut faire. Créer un mot, Modifier un mot ou Supprimer un mot. L'activité comporte 3 boutons de navigation
- Activity Create
 - Dans cette activité, l'utilisateur va pouvoir définir et créer son mot pour insérer dans la base de donnée.
 - l'activité comporte :
 - * 3 champs de texte pour que l'utilisateur puisse entrer le mot, son type et sa définition.
 - * 1 bouton pour soumettre le mot à la base de donnée.
 - * 1 toast qui affiche la réponse du serveur.
- Activity Update
 - Dans cette activité, l'utilisateur va pouvoir afficher le mot à changer avant de le modifier.

- l’activité comporte :
 - * 1 champ de texte pour entrer le mot à chercher.
 - * 1 bouton pour trouver le mot et afficher le mot cherché.
 - * 1 champ de texte pour entrer la définition à changer pour le mot cherché.
 - * 1 bouton pour sousmettre le changement.
- Activity Delete
 - Dans cette activité, l’utilisateur va pouvoir entrer le mot, son type et sa définition pour supprimer de la base de donnée.
 - l’activité comporte:
 - * 3 champs de texte pour entrer le mot, son type et sa définition.
 - * 1 bouton pour sousmettre la suppression à la base de donnée.

3 Définition des technologies utilisées

3.1 La base de donnée

mysql Ver 8.0.27-0ubuntu0.20.04.1 for Linux on x86_64 ((Ubuntu))

3.2 Le service web

- jdk1.8.0_202
- Java compiler 1.8
- Server Tomcat v10.0
- Archetypes Jersey-quickstart-webapp 3.0.3
- mysql-connector-java-8.0.27

3.3 L’interface utilisateur (Android)

- Android studio Arctic Fox 2020.3.1 patch 3
- VM: OpenJDK 64-Bit Server VM

4 Procédure d’instalation

4.1 Environnement développement

- Eclipse IDE for Enterprise Java and Web Developers
 - * Version: 2021-09(4.21.0).

- mysql Ver 8.0.27-0ubuntu0.20.04.1 for Linux on x86_64 ((Ubuntu))
- Java 8
 - * jdk1.8.0 202
 - * Java compiler 1.8
- Server Tomcat v10.0
 - * Archetypes Jersey-quickstart-webapp 3.0.3
 - * mysql-connector-java-8.0.27
- Android studio Arctic Fox 2020.3.1 patch 3

4.2 Environnement de production

- Émulateur android Pixel _ 5 _ API _ 30
 - * Android 11

5 Test

- Tests base de donnée
 - En utilisant l'invite de commande Ubuntu et mysql commands.
- Tests serveur
 - Pour tester les requêtes, j'ai utilisé Advanced Rest Client pour vérifier les requêtes GETs.
 - Les tests ont été effectués depuis la machine localhost et depuis une autre machine qui connecte au même réseau en utilisant l'adresse IP de la machine serveur.
 - Les tests depuis un autre réseau n'ont pas été aboutis car le temps d'attente a été trop long.
- Tests Application
 - Testé sur l'émulateur android depuis AndroidStudio.
 - Testé sur un téléphone personnel.

6 Conclusion

Le projet a été abouti le 14 Janvier 2022. Le rendu est une application android qui connecte bien au serveur localhost et utilise la base de donnée dictionnaire. Vous retrouverez le projet sur mon [GitHub](#).