



Documentation Technique

UE TAC

Florentin Bugnon

Master 1 informatique, parcours E-Services

2022

2023

Spécifications techniques	2
Affichage	2
Récupération des données	2
Stockage des données	3
Diagramme de séquences	4
Récupération des données	4
Sauvegarde des données en local	5
Diagramme de classe	6
Front-end	6
Back-end	7
Contrat d'interface	8
Récupérer la liste des champions du jeu	8
Récupérer les informations détaillées d'un champion	8

Spécifications techniques

Affichage

Lors de l'ouverture de l'application, une 'MainActivity' est instanciée.
Celle-ci affiche l'écran principal de l'application, qui contient la liste des champions.

Pour pouvoir naviguer entre la liste des champions et la liste des favoris, un 'ViewPager2' a été choisi.

Ce composant permet de naviguer entre deux listes distinctes, et de faire défiler les écrans horizontalement.

Pour gérer l'affichage dans chaque onglet, un 'FragmentStateAdapter' a été utilisé.
Ainsi, dans l'onglet 'Champions', on affichera le fragment 'ChampionsListFragment',
et dans l'onglet 'Favoris', on affichera le fragment 'FavorisListFragment'.

Chacun de ces fragments contient un recyclerView, qui affichera la liste des items.
Le contenu de ces listes sera géré par un 'RecyclerView.Adapter'.

Dans l'activité principale, il y a un bouton 'Switch' en haut à droite de l'écran, qui permet de passer d'un affichage en liste à un affichage en grille. L'activation de celui-ci met à jour le LayoutManager du recyclerView.

Pour afficher les items d'une liste, un 'RecyclerView.ViewHolder' a été utilisé.
Pour chaque item, on affiche :

- un bouton 'Add to favorites'.
- une image
- un nom

Pour l'affichage des images, la librairie Glide a été choisie.

Lors d'un clic sur un item, on affiche les informations détaillées du champion, on crée alors une nouvelle activité.

Cette activité contient un 'ChampionDetailsFragment'.

Il est possible de revenir en arrière grâce à la flèche de retour en haut à gauche de l'écran.
Cette action termine alors l'activité précédemment créée.

Récupération des données

La récupération de données se fait en 2 temps :

- Dans un premier temps, on fait un appel API grâce à la classe 'RiotCalls' pour récupérer la liste des champions.
- Dans un second temps, on fait un appel à notre base de données grâce à la classe 'mainViewModel' pour récupérer la liste des favoris

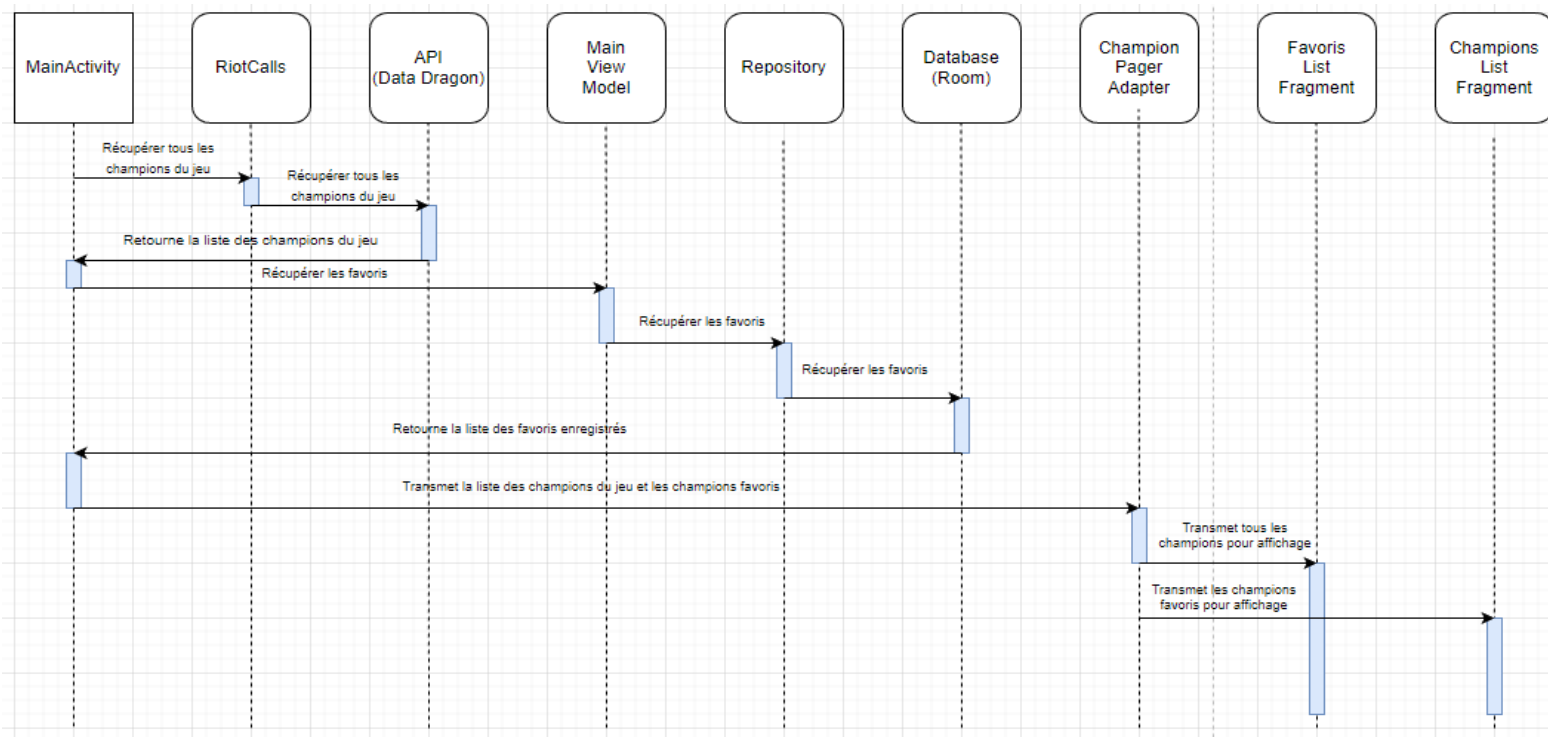
Stockage des données

Pour stocker les données de notre application, une base de données 'Room' a été mise en place.

Cela nous permet d'établir une liste de favoris pour l'utilisateur et de stocker des informations sur les champions.

Diagramme de séquences

Récupération des données



Sauvegarde des données en local

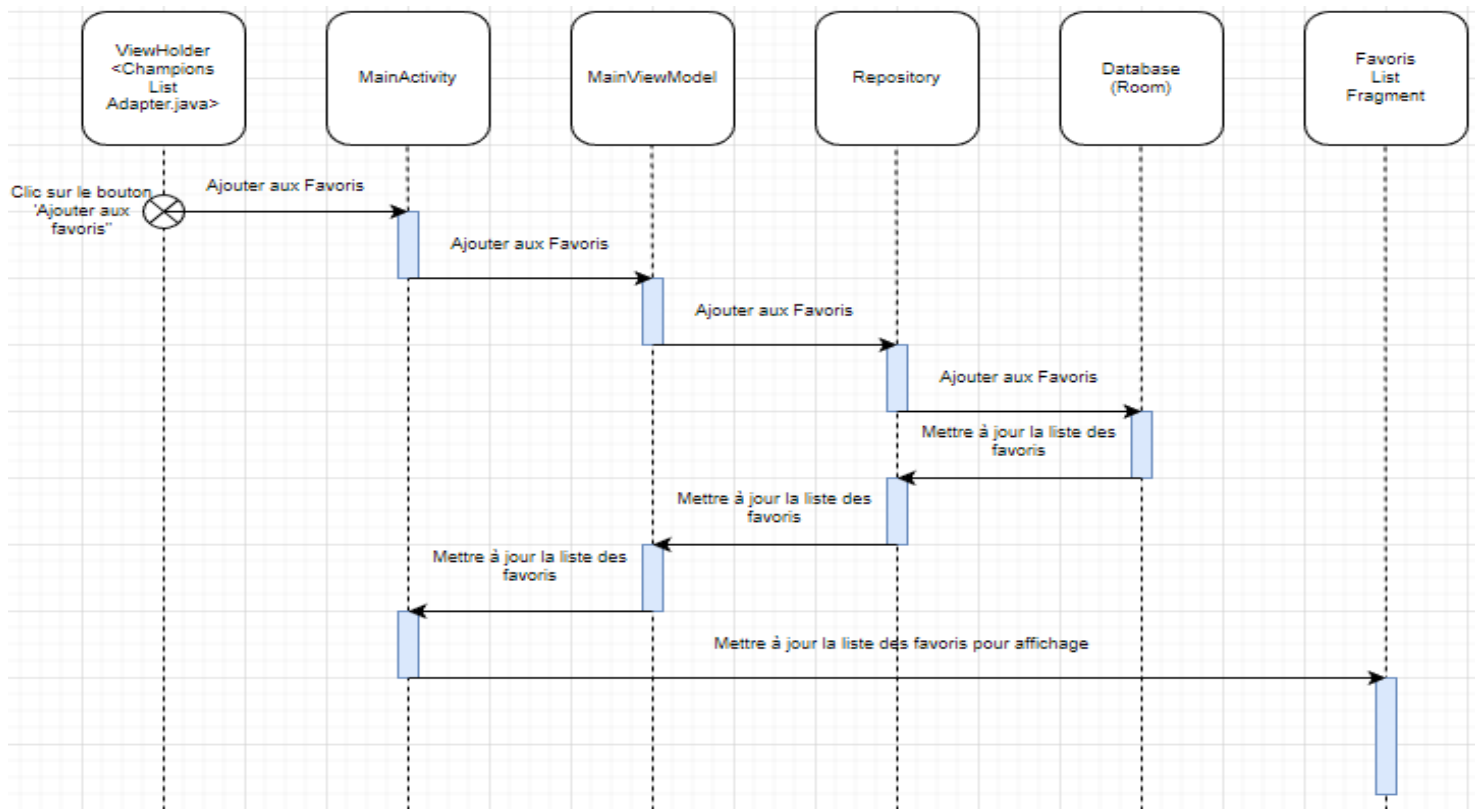
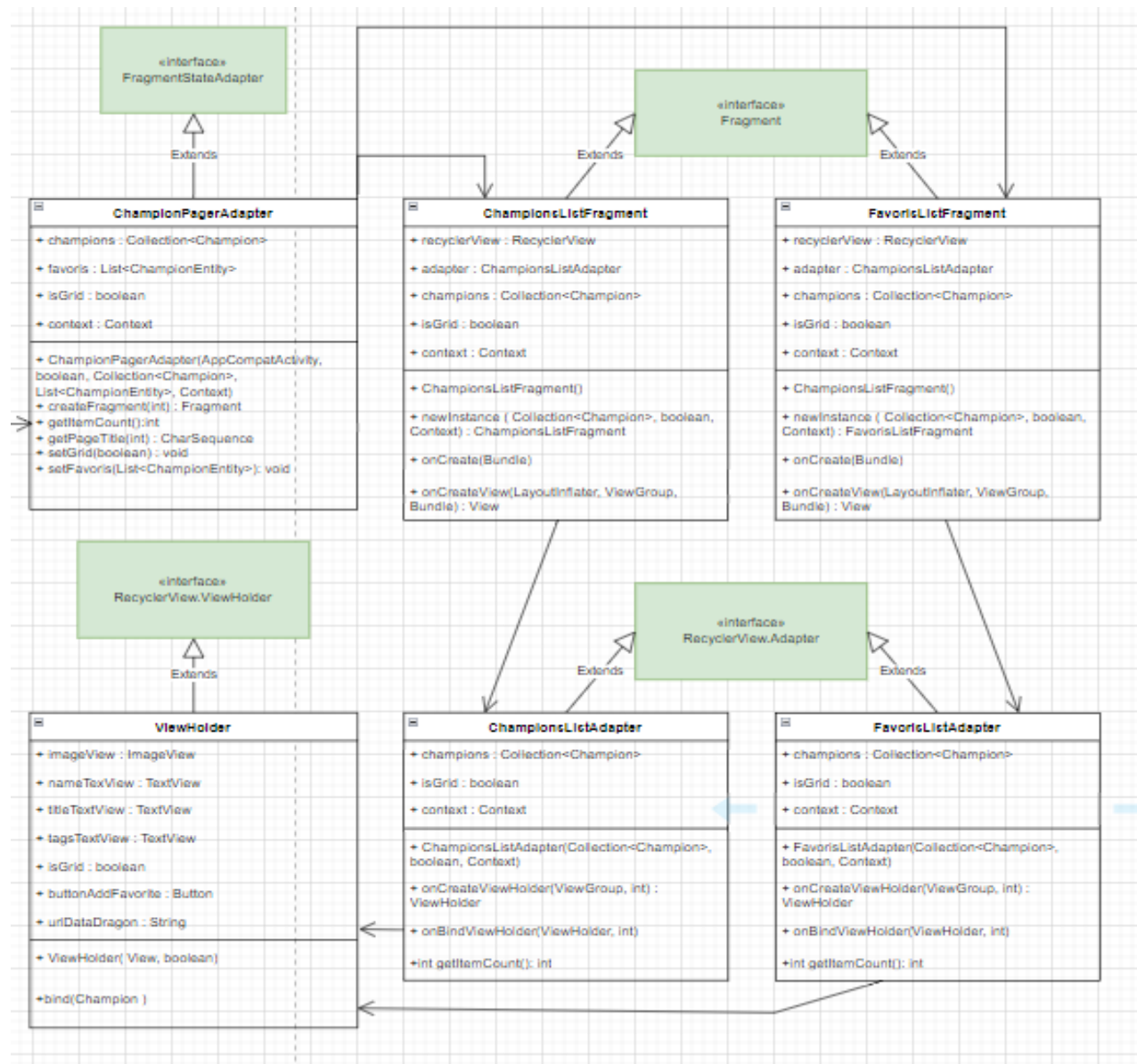


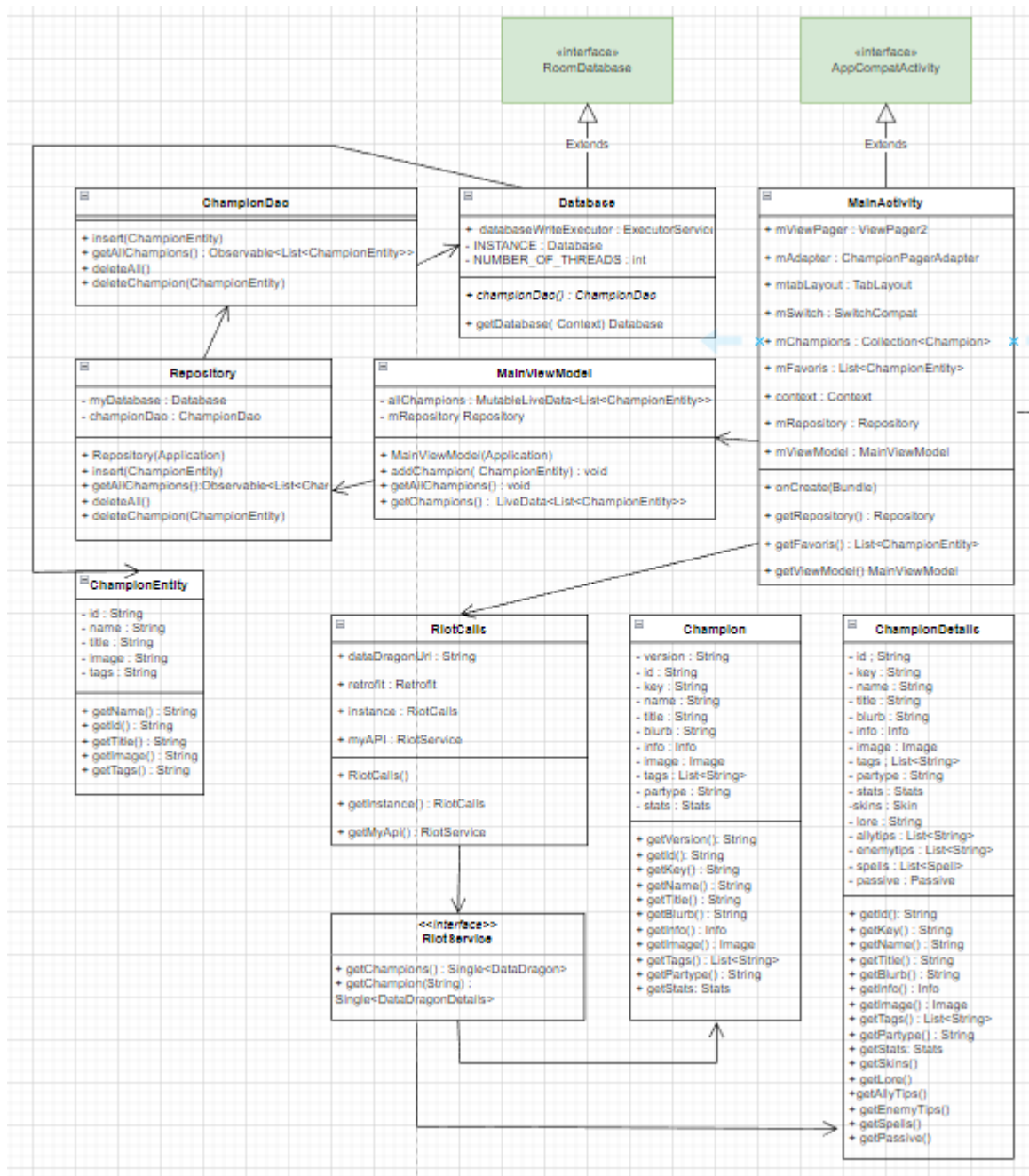
Diagramme de classe

Front-end

🚩 Les rectangles verts représentent les classes présentes dans Android de base.



Back-end



Contrat d'interface

url de l'api : `"https://ddragon.leagueoflegends.com/cdn/12.23.1/"`

Récupérer la liste des champions du jeu

```
@GET("data/fr_FR/champion.json")
```

Cette requête ne prend pas de paramètre en compte.

La réponse attendue est un objet json contenant 4 champs:

- type
- format
- version
- data

Le champ qui nous intéresse est 'data', qui contient un dictionnaire des champions du jeu

Récupérer les informations détaillées d'un champion

```
@GET("data/fr_FR/champion/{champion}.json")
```

Cette requête prend en compte un paramètre :

- l'id d'un champion

La réponse attendue est un objet json contenant 4 champs:

- type
- format
- version
- data

Le champ qui nous intéresse est 'data', qui tous les détails concernant le champion.