

2022 -2023

PROGRAMMATION MOBILE I:

Application dédiée aux Speedrun :

Helb Blitz !

Réalisation d'une application mobile avec Android Studio

RIGGIO JONATHAN





Tables des matières

Tables des matières.....	1
Introduction	2
Schéma relation entre les éléments	3
Schéma relation entre les classes.....	4
Fonctionnalités de base	7
Fonctionnalités supplémentaires	13
Limitations	14
Feedback des utilisateurs	15
Conclusion.....	16
Bibliographie	17



Introduction

Dans le cadre du cours de développement Mobile I, une application mobile Android en Java dédiée aux speedruns a été créée. L'application, appelée "HelbBlitz", permet aux joueurs de créer un compte en renseignant leur pseudo, leur nationalité et d'autres informations supplémentaires.

Les utilisateurs ont la possibilité de rechercher les jeux pour lesquels ils souhaitent obtenir le meilleur score et de les ajouter à leur bibliothèque personnelle. Afin de fournir des informations détaillées sur les jeux, telles que des images, des descriptions et les plateformes compatibles, l'application utilise des API (Rawg.io) pour récupérer ces données. Les joueurs ont également la possibilité de partager leurs propres scores de speedruns pour les jeux disponibles sur l'application.

De plus, l'application HelbBlitz inclut une carte (Openstreetmap) permettant aux joueurs de visualiser les meilleurs scores enregistrés par pays. Un système de notification a été mis en place pour informer les joueurs sur de nouveaux jeux à découvrir chaque jour à une heure précise.

Ce rapport présentera en détail les différentes fonctionnalités de l'application HelbBlitz, en mettant l'accent sur son architecture, les technologies utilisées, l'analyse du projet avec des représentations en schéma, ainsi que les défis rencontrés lors du développement et les solutions apportées. Les feedbacks d'autres développeurs sur l'application seront également inclus.

En conclusion, l'application HelbBlitz offre aux utilisateurs une plateforme conviviale et pratique pour explorer les speedruns, partager leurs scores et découvrir de nouveaux jeux passionnants. Le développement de cette application a permis d'acquérir des compétences précieuses en matière de développement mobile, tout en offrant une expérience engageante pour les utilisateurs de l'application.



Schéma relation entre les éléments

Voici mon schéma de relation entre les activités et les fragments. J'ai utilisé des fragments dans mon application pour plusieurs raisons. Tout d'abord, cela m'a permis de découper l'interface utilisateur en modules réutilisables. Par exemple, j'ai créé un fragment pour la barre de navigation en bas de l'écran, et chaque fragment représente une vue différente qui peut être chargée dans la partie centrale de l'écran.

L'utilisation des fragments a rendu mon interface graphique plus modulaire et flexible. Plutôt que de changer toute l'activité lorsque je clique sur un bouton de la barre de navigation, seul le fragment correspondant à cette vue spécifique est remplacé. Cela permet une transition plus fluide entre les différentes vues de l'application.

Voici le diagramme avec le texte explicatif :

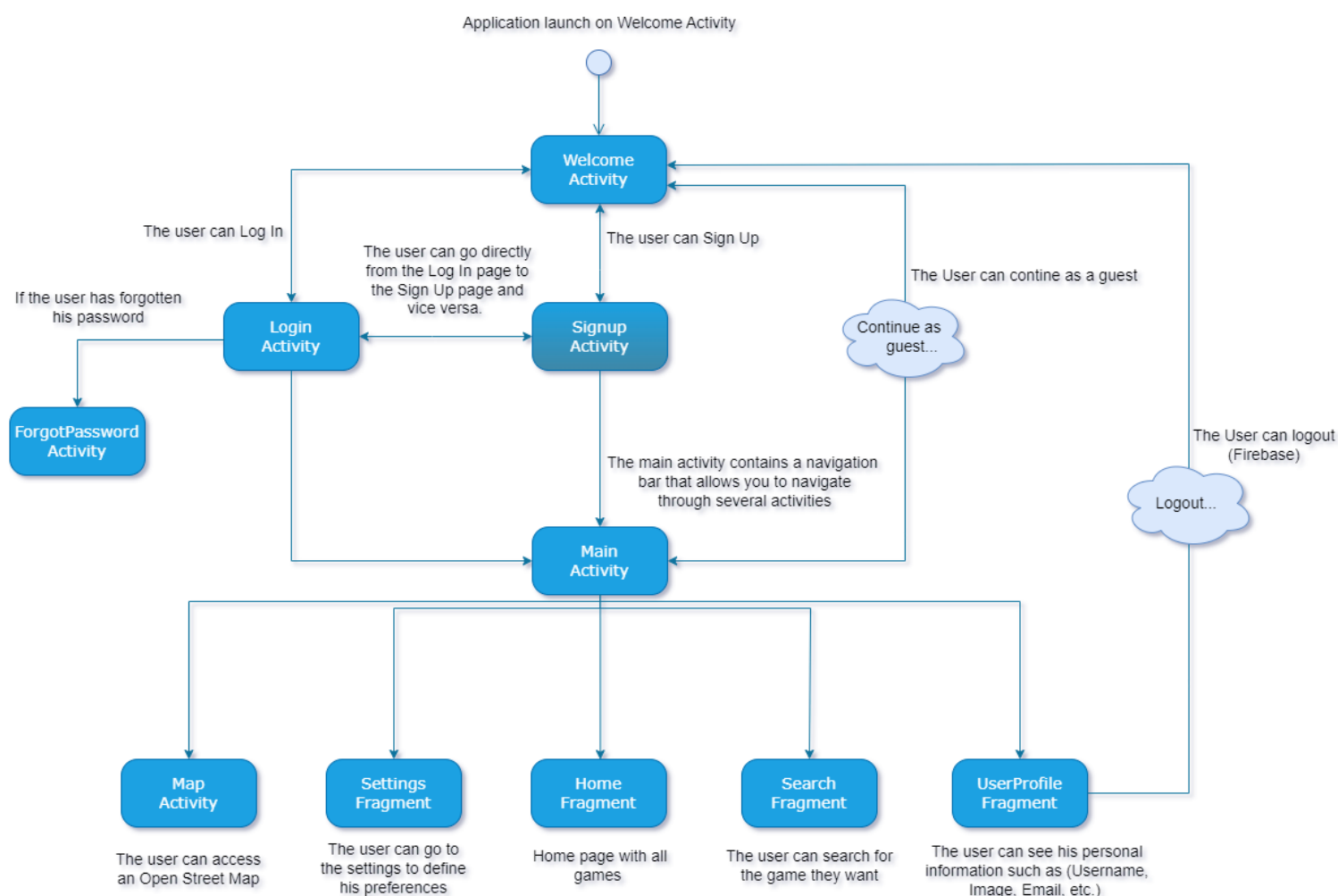
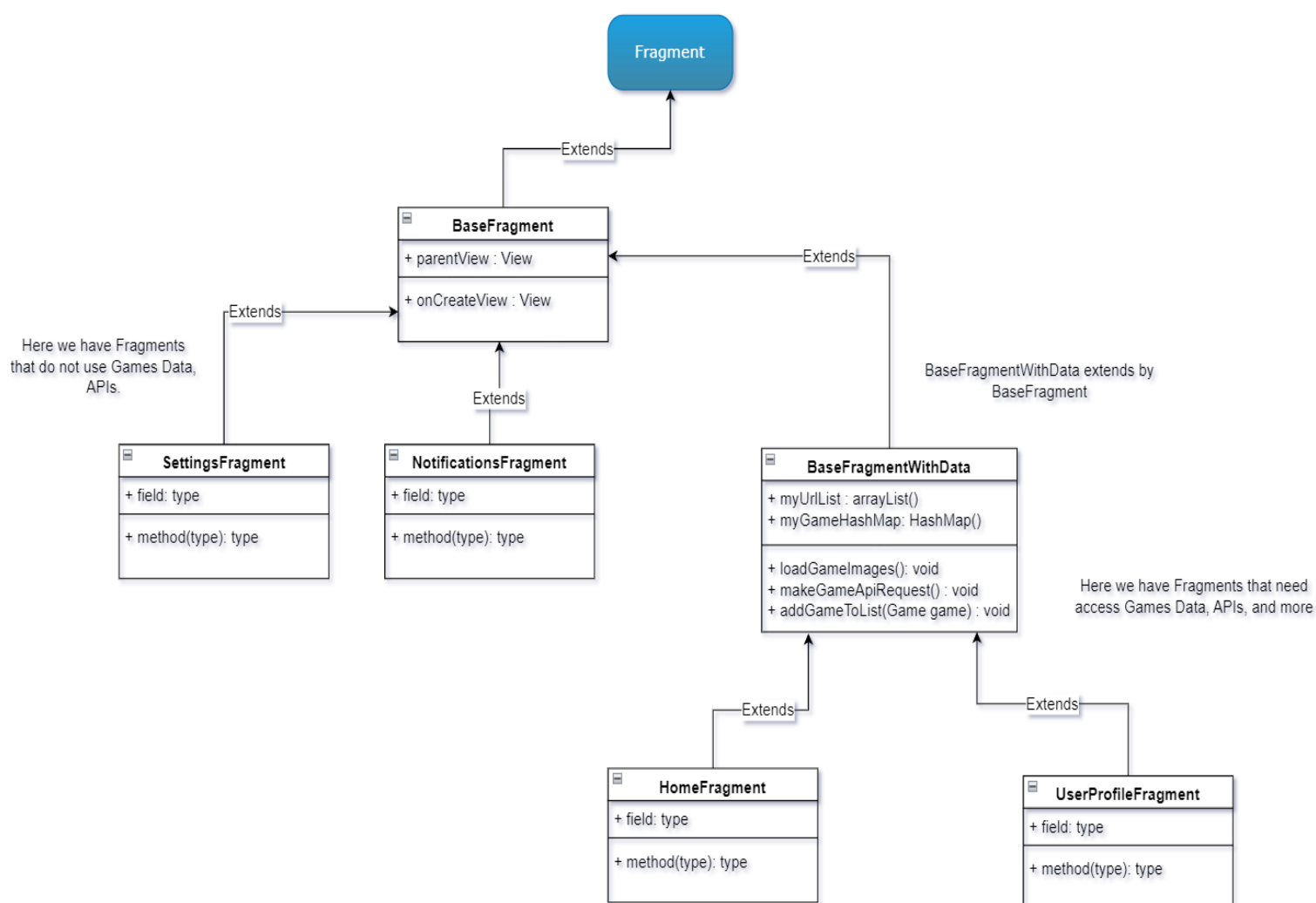




Schéma relation entre les classes

Bien que la qualité du code n'ait pas été l'objectif principal du cours de Programmation Mobile I, j'ai tout de même accordé une attention particulière à l'application de certains concepts d'orientés objets. Cette démarche m'a permis de renforcer mes compétences en développement mobile et de me préparer pour des projets futurs plus complexes, tels que le cours de Programmation Java IV où la qualité du code est un point central du projet.

En appliquant des principes d'orientés objets tels que, l'encapsulation, l'héritage, et bien d'autres. J'ai pu organiser et structurer mon code de manière plus claire et modulaire. Cela facilite la maintenance, la lisibilité et l'évolutivité de l'application.





Dans le schéma ci-dessus, j'ai créé une classe "BaseFragment" qui hérite de "Fragment" et qui permet à ses classes filles de bénéficier des fonctionnalités proposées par les fragments. Ensuite, j'ai une classe plus spécifique appelée "BaseFragmentWithData" qui hérite de "BaseFragment" et qui gère également les données, etc. En résumé, tous les fragments qui ont besoin d'API ou d'informations sur les jeux héritent de "BaseFragmentWithData", tandis que ceux qui n'ont pas besoin de données héritent simplement de "BaseFragment". Pour faire le lien avec le cours de Programmation Java IV, on peut dire que cela suit le principe de forte cohésion, où chaque classe se concentre sur une tâche spécifique.

Maintenant que les sous classes HomeFragment et UserProfileFragment héritent de BaseFragmentWithData. Ils peuvent uniquement appeler la méthode loadGamesImages() pour afficher les jeux à l'écran. A condition bien sûr d'avoir adapté leurs layout.xml.

Voici un exemple avec HomeFragment.java :

```
public class HomeFragment extends BaseFragmentWithData {

    4 usages
    public HomeFragment() {
        // Required empty public constructor
    }

    public static HomeFragment newInstance() { return new HomeFragment(); }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        View view = inflater.inflate(R.layout.fragment_home, container, attachToRoot: false);

        return view;
    }

    @Override
    public void onViewCreated(@NonNull View view, @Nullable Bundle savedInstanceState) {
        super.onViewCreated(view, savedInstanceState);

        // ----- On charge nos jeux ----- //
        loadGameImages();
    }
}
```

Avec uniquement une méthode nous avons l'affichage des jeux, un setOnClickListener() pour chaque images, et les requêtes API qui sont également fait dans cette méthodes.



UI/UX

Pour ce qui concerne l'UI (User Interface) et l'UX (User Expérience), j'ai effectué des recherches approfondies sur Internet, notamment en consultant la documentation d'Android Studio. Cette documentation fournit des informations détaillées sur les bonnes pratiques de conception d'interfaces utilisateur pour les applications Android, y compris les principes de conception, les directives de conception matérielle, les composants d'interface utilisateur et bien plus encore. (Source en Bibliographie).

Pour la conception de l'interface utilisateur (UI), je me suis inspiré des applications les plus attrayantes telles que Twitter et Instagram. J'ai notamment adopté l'idée de la barre de navigation en bas de l'écran, qui facilite la navigation entre les différentes pages de l'application. Cela offre aux utilisateurs une expérience de navigation plus fluide et intuitive.

En ce qui concerne l'expérience utilisateur (UX), je me suis également inspiré des applications cités précédemment. Mais également des slides présentés aux cours. Par exemple, j'ai pris en compte l'importance d'améliorer la fonction de recherche de jeux. Pour mettre en place une barre de recherche efficace et conviviale.

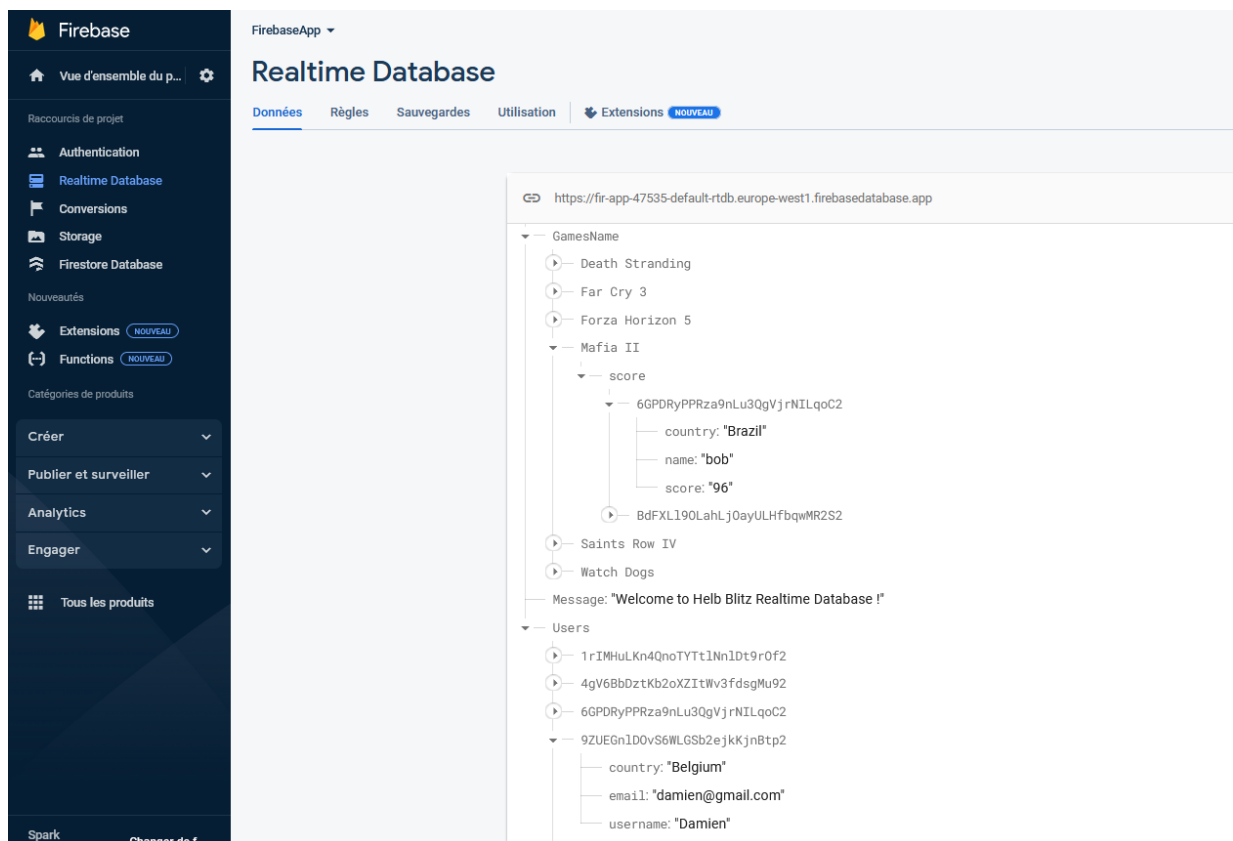


Fonctionnalités de base

1) Base de données Firebase

L'une des premières fonctionnalités que nous devons avoir dans notre application c'était d'avoir une base de données Firebase pour y stocker nos objets. La mise en place n'a pas été très compliqué nous avons suivis les vidéos de notre professeur disponible sur e-Campus.

Voici un exemple d'utilisation :



Sur cette représentation nous pouvons apercevoir qu'il y a un stockage des scores par jeux avec les informations personnelles de l'utilisateur ayant posté ce score. Mais aussi tous les utilisateurs ayant créer un compte sur l'application.

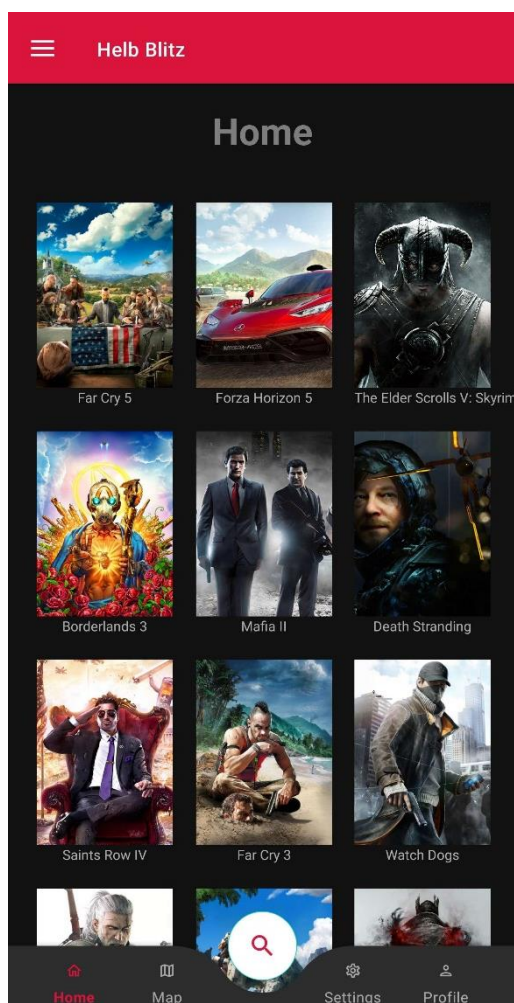


2) Utilisation d'API

La fonctionnalité la plus importante de ce projet était l'intégration des API. Avant ce projet, je n'étais pas très familier avec ce terme. Pour implémenter cette fonctionnalité, j'ai utilisé deux API différentes. J'ai utilisé une API gratuite fournie par Rawg.io pour obtenir des informations sur les jeux, et j'ai également utilisé une autre API pour intégrer la fonctionnalité de la carte, dont je discuterai plus en détail dans ce rapport.

L'utilisation de l'API pour récupérer les informations sur les jeux n'a pas été très complexe, principalement grâce à la documentation claire et complète fournie par Rawg.io. En suivant quelques étapes simples, j'ai pu effectuer des requêtes et obtenir les données nécessaires pour afficher les informations sur les jeux.

Voici un exemple d'utilisation :



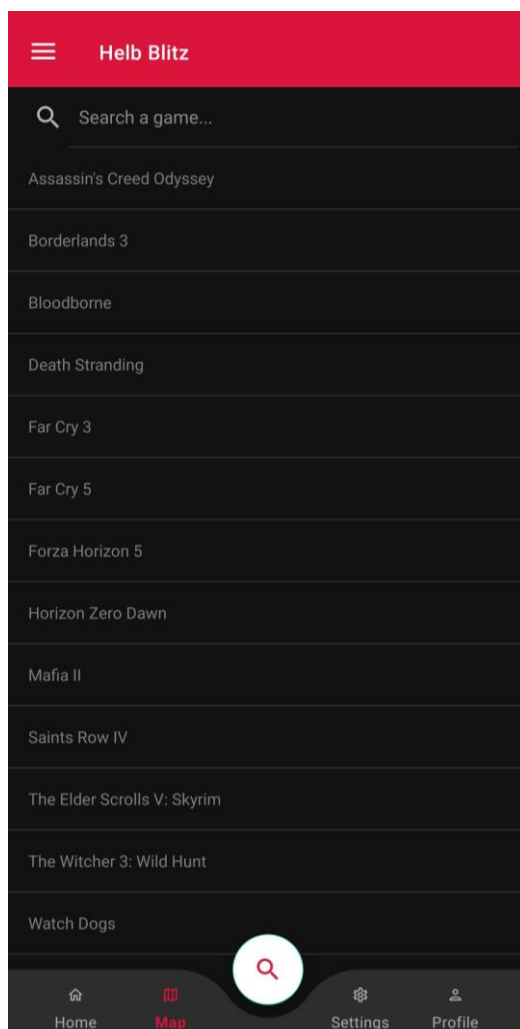


3) Barre de Recherche

Pour la barre de recherche j'ai fait attention aux conseils données par le professeurs au cours qui était dédiée à l'UI/UX. Qui parlait de l'importance de mettre en place un système de recherche efficace et qui offre une expérience utilisateur fluide et intuitive.

Cependant, il faut savoir qu'il y a une petite limitation avec la barre de recherche dont j'en parlerai plus tard dans le rapport.

Voici un exemple d'utilisation :



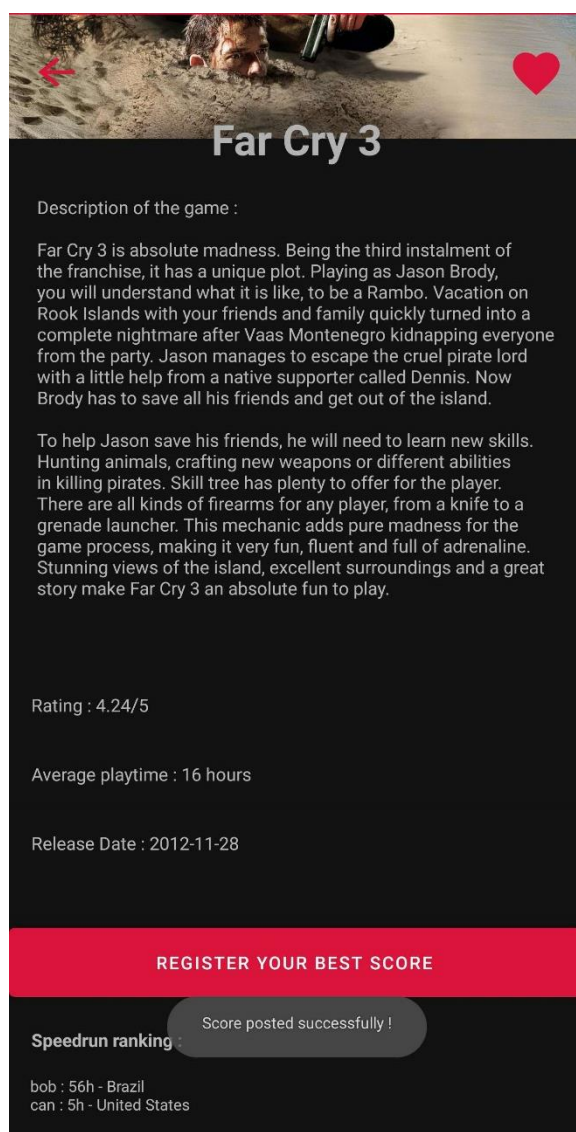


4) Post de score

Le post de score est également une des fonctionnalités les plus importantes de mon projet. L'utilisateur cherche et choisit le jeu qu'il souhaite. Lorsqu'il appuie sur un jeu, une nouvelle activité s'ouvre dans laquelle se trouvent des informations plus détaillées du jeu qui proviennent également d'API.

Les joueurs ont la possibilité de publier leur propre score pour le jeu sélectionné.

Voici un exemple d'utilisation :



Far Cry 3

Description of the game :

Far Cry 3 is absolute madness. Being the third instalment of the franchise, it has a unique plot. Playing as Jason Brody, you will understand what it is like, to be a Rambo. Vacation on Rook Islands with your friends and family quickly turned into a complete nightmare after Vaas Montenegro kidnapping everyone from the party. Jason manages to escape the cruel pirate lord with a little help from a native supporter called Dennis. Now Brody has to save all his friends and get out of the island.

To help Jason save his friends, he will need to learn new skills. Hunting animals, crafting new weapons or different abilities in killing pirates. Skill tree has plenty to offer for the player. There are all kinds of firearms for any player, from a knife to a grenade launcher. This mechanic adds pure madness for the game process, making it very fun, fluent and full of adrenaline. Stunning views of the island, excellent surroundings and a great story make Far Cry 3 an absolute fun to play.

Rating : 4.24/5

Average playtime : 16 hours

Release Date : 2012-11-28

REGISTER YOUR BEST SCORE

Score posted successfully !

Speedrun ranking :

bob : 56h - Brazil
can : 5h - United States



5) Utilisation de la Map

En ce qui concerne l'utilisation d'une API pour la Map j'ai décidé d'utiliser OpenStreetMap plutôt que Google Maps du fait qu'il soit plus libre et accessible pour les étudiants.

Pour la réalisation de cette fonctionnalité je me suis inspiré de la vidéo YouTube de Frédéric RALLO. (lien en bibliographie). Par après je l'ai adapté à ma manière pour ajouter mes propres Overlays, et les données dont je souhaitais afficher à l'écran.

Cette fonctionnalité a également une petite limitation dont j'en parlerai dans ce rapport.

Voici un exemple d'utilisation :





6) Notification

Pour la fonctionnalité de notification dans mon application, l'idée initiale était de permettre aux utilisateurs de recevoir des notifications lorsqu'ils se faisaient battre leur score dans un jeu spécifique. Cependant, en raison de contraintes de temps et après discussion avec le professeur, nous avons convenu de modifier cette fonctionnalité pour qu'elle devienne une notification quotidienne qui incite les utilisateurs à découvrir un jeu spécifique.

Voici un exemple d'utilisation :

```
1 usage
private void methodeAlarme() {
    Calendar calendar = Calendar.getInstance();
    calendar.set(Calendar.HOUR_OF_DAY, 18); // heures
    calendar.set(Calendar.MINUTE, 10); // minutes

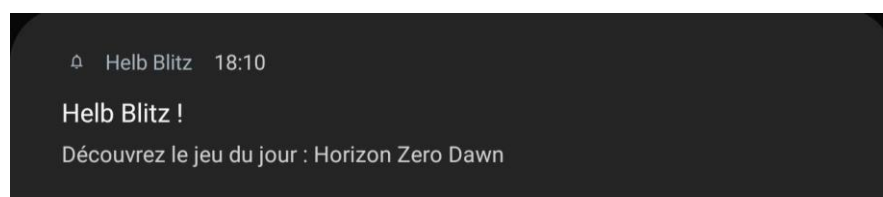
    // Intent qui pointe vers le BroadcastReceiver
    Intent intent = new Intent( packageContext: this, AlarmReceiver.class);

    // PendingIntent sera déclenché lorsque l'alarme se déclenche
    PendingIntent pendingIntent = PendingIntent.getBroadcast( context: this, requestCode: 0, intent, PendingIntent.FLAG_UPDATE_CURRENT);

    AlarmManager alarmManager = (AlarmManager) getSystemService(ALARM_SERVICE);

    // Alarme qui se répète tous les jours à la même heure
    alarmManager.setRepeating(AlarmManager.RTC_WAKEUP, calendar.getTimeInMillis(), AlarmManager.INTERVAL_DAY, pendingIntent);
}
```

Voici le rendu :



Evidemment les jeux du jour changent à chaque notification de manière aléatoire.



Fonctionnalités supplémentaires

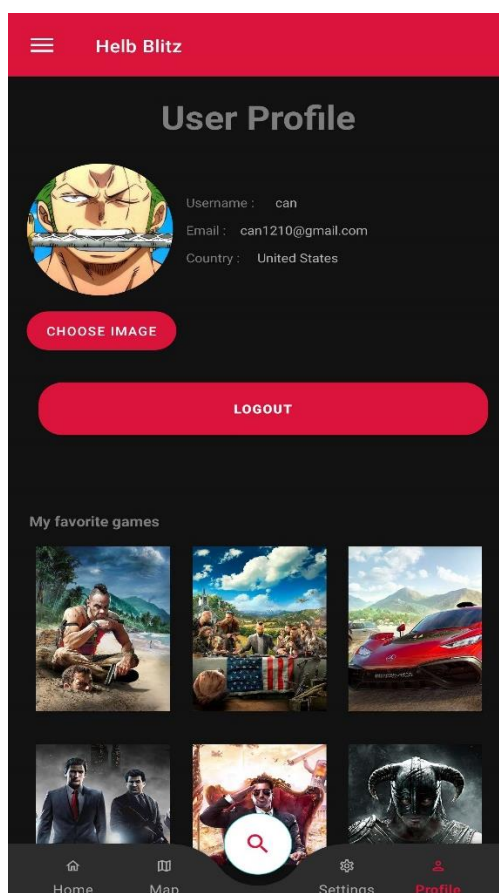
1) User profile

En tant que fonctionnalité supplémentaire de mon application mobile, j'ai ajouté un fragment UserProfile qui permet aux utilisateurs d'accéder à leurs informations personnelles, y compris leur photo de profil, ainsi qu'à leur bibliothèque de jeux favoris.

Pour stocker et afficher la photo de profil dans mon application, j'ai utilisé la bibliothèque Picasso qui est une bibliothèque populaire et facile à utiliser. Elle facilite le chargement et l'affichage d'images à partir de différentes sources, y compris Firebase.

Pour récupérer une image depuis ma base de données Firebase et l'afficher à l'écran, j'ai utilisé la méthode load() de Picasso en spécifiant l'URL de l'image.

Voici un exemple d'utilisation :

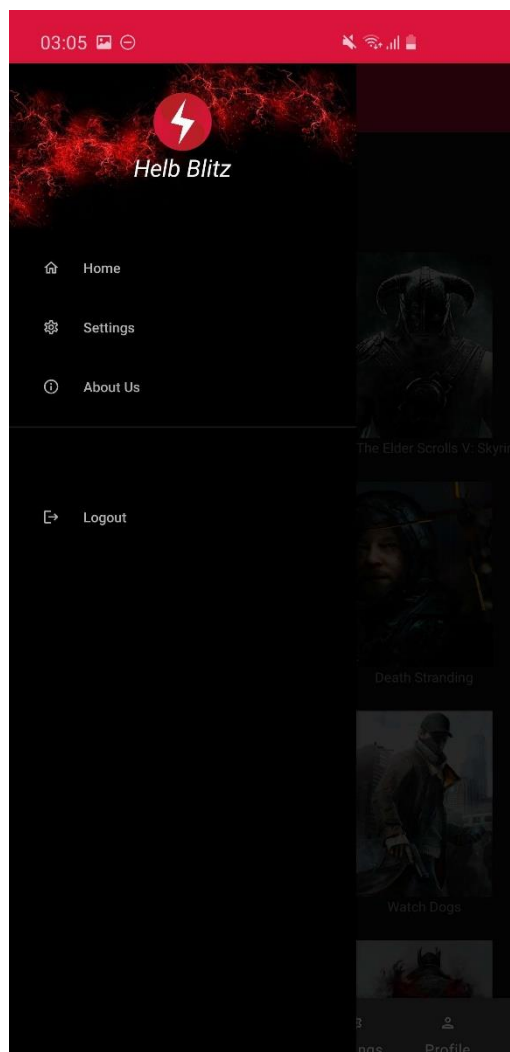




Limitations

- 1) Pour la fonctionnalité de la Map, je n'ai pas pu récupérer les informations pour le meilleur score pour les afficher directement sur la carte. C'est pourquoi j'ai choisi de les ajouter en utilisant des données prédéfinies.
- 2) Un joueur peut poster uniquement un score, le prochain score qu'il postera remplacera son post de score précédent.
- 3) La barre de navigation défilant n'était pas une des fonctionnalités de mon application mais je l'avais quand même implémenté. Il n'a aucun comportement lorsque je clique sur un des Items qu'il contient.

Voici une représentation :





Feedback des utilisateurs

En ce qui concerne les feedbacks sur mon application. Ces feedbacks se sont révélés extrêmement utiles pour améliorer l'ergonomie et l'expérience utilisateur de mon application.

Une remarque pertinente que j'ai reçue concernait l'accès au profil utilisateur pour les utilisateurs invités (Guest). Il a été souligné que cela n'avait pas beaucoup de sens et que des mesures devraient être prises pour résoudre ce problème. Finalement, j'ai mis en place une alerte d'information qui explique à l'utilisateur connecté en tant qu'invité que pour accéder à l'utilisateur profile il faudra créer un compte.

Comment évalueriez-vous l'ergonomie de l'application ?

6 réponses



Un autre feedback intéressant ici c'est la remarque en mauve qui dit : « On ne peut pas savoir si les gens disent la vérité ». Effectivement, l'application n'offre pas une vérification poussée pour savoir si les scores des joueurs sont véridiques. On fait confiance à l'utilisateur.

Si vous vous intéressiez aux speedruns, est-ce que vous utiliseriez Helb Blitz régulièrement ? Sinon, pouvez-vous expliquer les raisons pour lesquelles vous ne l'utiliseriez pas

6 réponses





Conclusion

En conclusion, le développement de l'application HelbBlitz a été une expérience très plaisante et enrichissante. J'ai particulièrement apprécié la possibilité de créer une base de données realtime avec Firebase, ce qui a ajouté une nouvelle dimension à notre projet par rapport aux bases de données SQL auxquelles nous sommes habitués.

Bien que j'ai rencontré quelques difficultés en cours de route, telles que des erreurs de syntaxe, des problèmes de configuration et des ajustements nécessaires dans la gestion des scores, ces obstacles nous ont permis de renforcer nos compétences en résolution de problèmes et en collaboration d'équipe. L'utilisation de ChatGPT a également permis de mieux comprendre certains concepts spécifique à Android Studio évidemment, Toutes les requêtes et les prompts que j'ai utilisés sont soigneusement documentés dans le fichier CodeLogs.pdf, disponible dans notre projet.

Si j'avais eu plus de temps, j'aurais aimé perfectionner certains aspects de l'application. Par exemple, Pour la map certaines informations sont hard codé. De plus, j'aurai pu intégrer davantage de fonctionnalités, telles que la possibilité pour les utilisateurs de discuter entre eux en commentant sous leur jeux favoris, etc.

En résumé, malgré les défis rencontrés, le projet HelbBlitz a été pour moi une expérience incroyablement enrichissante. J'ai pu mettre en pratique mes nouvelles connaissances en développement mobile, découvrir de nouvelles technologies et développer des compétences précieuses en matière de résolution de problèmes.



Bibliographie

- 1) IT-Point. (2021a, février 22). *Cours 2 - Activites et Intent* [Vidéo]. YouTube.
<https://www.youtube.com/watch?v=qYPLAvGnFYU>
- 2) IT-Point. (2021, 8 mars). *Cours 3 Styles, Themes, Toast et images* [Vidéo]. YouTube.
<https://www.youtube.com/watch?v=1qQ5c3okMkg>
- 3) IT-Point. (2021b, mai 3). *HTTP Request - JSON* [Vidéo]. YouTube.
<https://www.youtube.com/watch?v=6BqoHsp9B4Q>
- 4) IT-Point. (2021b, avril 19). *Développement Mobile - Cours 5 - Firebase* [Vidéo].
YouTube. <https://www.youtube.com/watch?v=WykjSz9K05A>
- 5) Android Pro-grammation. (2021, août 2). *Créer l'application Instagram en JAVA ||
Android Studio PARTIE 1(la barre d'items)* [Vidéo]. YouTube.
<https://www.youtube.com/watch?v=8cwM1xL-u8E>.
- 6) *Develop UI for Android*. (s. d.). *Android Developers*.
<https://developer.android.com/develop/ui>
- 7) Frédéric RALLO. (2020, 19 avril). *Tutoriel créer une application avec Open Street
Map pour Android* [Vidéo]. YouTube.
<https://www.youtube.com/watch?v=4JnYKkSSZgc>.
- 8) Osmroid. (s. d.). *Home*. GitHub. <https://github.com/osmdroid/osmdroid/wiki/>.
- 9) Knowledge, A. (2023, 9 mars). *Navigation Drawer in Android Studio using Java -
Android Knowledge*. Android Knowledge. <https://androidknowledge.com/navigation-drawer-android-studio/>
- 10) Développeur Libre. (2022, 4 février). *7 ASTUCES pour créer un design magnifique
d'application mobile* [Vidéo]. YouTube.
<https://www.youtube.com/watch?v=t2ikO9jvB3M>