

24/05/2023

Mobile I : rapport

BOOK FUTSAL



Un enseignement de proximité, ouvert sur le monde



ASSOU Hicham

Table des matières

1	Introduction.....	2
2	Description des technologies utilisé.....	2
3	Fonctionnalités	3
	3.1. Visualisation des centres sportifs sur une carte	3
	3.2. Page détail	5
	3.3. Réservations	7
	3.4. Météo	9
	3.5. Commentaires	9
	3.6. Gérer ses réservations.....	12
	3.7. Connexion / inscription	13
	3.8. Notifications	16
4	Analyse	17
	4.1. Diagramme de classe.....	18
	4.2. Diagramme de cas d'utilisation	19
5	UI/UX	20
	5.1. Design visuel attrayant	20
	5.2. Navigation facile	23
	5.3. Feedback utilisateur	23
	5.4. Adaptabilité aux différents appareils.....	23
	5.5. Possibilité d'ajouter des commentaires.....	23
	5.6. Conclusion	23
6	Limitations et développement futur	24
	6.1. La géolocalisation	24
	6.2. Ajout d'un nouveau centre sportif	24
	6.3. Gestion des horaires spéciaux	24
	6.4. Amélioration de la sécurité	24
7	Conclusion	25
8	Sources	26
9	ChatGPT	27

1 Introduction

Dans le cadre de notre cours de mobile I, nous avons été chargés de réaliser individuellement une application mobile au choix. Pour ce projet, j'ai choisi de développer une application de réservations de terrains de football dont le nom est « Book Futsal ». L'objectif de cette application est de permettre aux utilisateurs de réserver facilement un terrain de football en ligne, en choisissant un centre sportif disponible sur une carte et en sélectionnant la date et l'heure qui leur conviennent.

Dans ce rapport, je vais présenter en détail les fonctionnalités de l'application, ainsi que les choix techniques qui ont été faits pour les mettre en place. Je vais également évoquer les limitations de l'application et présenter les pistes de développement qui pourraient être améliorées à l'avenir. Enfin, je conclurai ce rapport avec une synthèse de l'ensemble du projet et de l'application que j'ai réalisée.

2 Description des technologies utilisé

Dans le cadre de ce projet de Mobile I, j'ai utilisé différentes technologies pour développer mon application de réservations de terrains de football. Les technologies choisies ont été déterminées en fonction de leur pertinence et de leur capacité à répondre aux exigences du projet.

Tout d'abord, j'ai utilisé Android Studio comme environnement de développement intégré (IDE). Android Studio est l'IDE officiel recommandé par Google pour le développement d'applications Android. Il offre une interface conviviale, des outils puissants de développement et de débogage.

Pour la gestion de la base de données, j'ai opté pour Firebase, une plateforme de développement backend proposée par Google. Firebase offre une multitude de services cloud, notamment la gestion des utilisateurs, le stockage des données en temps réel, etc. J'ai utilisé Firebase pour stocker les informations relatives aux réservations de terrains de football de manière sécurisée et pour permettre une mise à jour en temps réel des données.

Pour me partager le code, étant donné que je travaille sur 2 ordinateurs différents, j'ai utilisé Github, une plateforme de gestion de versions de code en ligne. Grâce à Github, j'ai pu travailler sur la dernière version du projet tout en conservant une trace des modifications apportées au code.

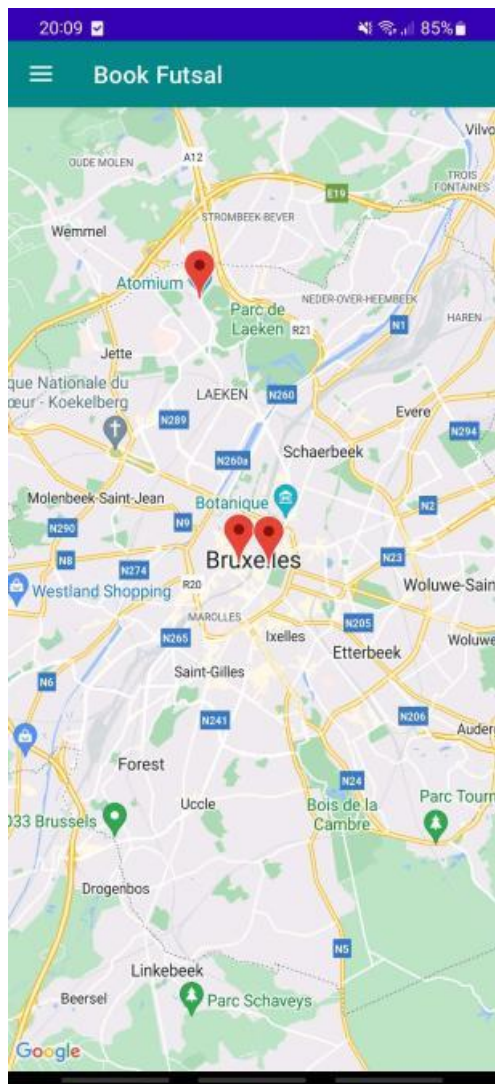
En résumé, pour le développement de mon application de réservations de terrains de football, j'ai utilisé Android Studio comme IDE principal, Firebase pour la gestion de la base de données et GitHub pour le partage du code et la gestion des versions. Ces technologies m'ont offert des outils puissants et des fonctionnalités avancées pour créer une application fonctionnelle, performante et fiable.

3 Fonctionnalités

Dans cette section, je vais présenter en détail les fonctionnalités implémentées dans mon application de réservations de terrains de football. Pour chaque fonctionnalité, je vais décrire son objectif spécifique ainsi que la méthode utilisée pour sa mise en place au sein de l'application. Je vais également expliquer les choix techniques effectués pour développer ces fonctionnalités et les justifier.

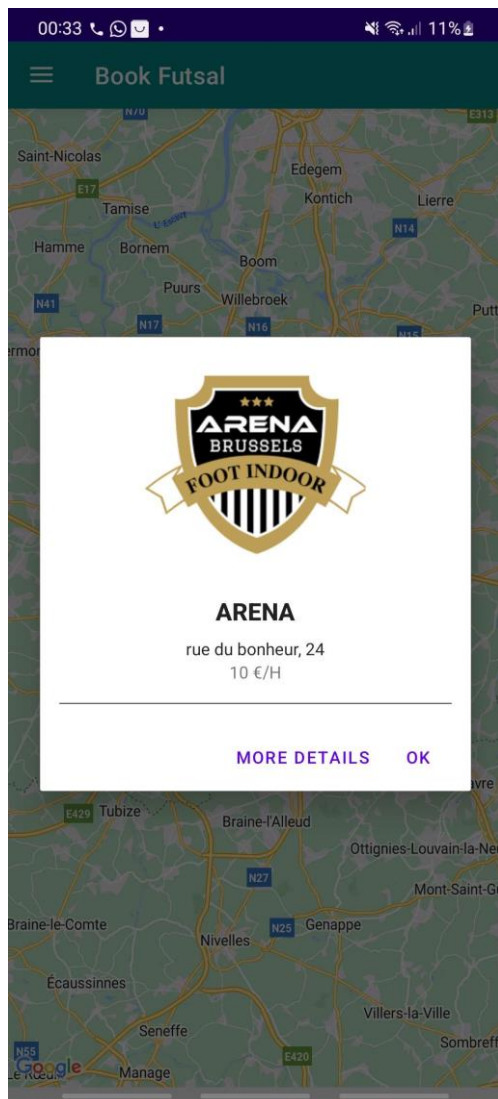
3.1. Visualisation des centres sportifs sur une carte

L'objectif de cette fonctionnalité est de permettre aux utilisateurs de visualiser les centres sportifs disponibles sur une carte interactive. Cela leur permet de localiser facilement les terrains de football et d'avoir une vue d'ensemble des options disponibles.



Pour mettre en place cette fonctionnalité, j'ai utilisé l'API Google Maps. Cette API offre des fonctionnalités de cartographie avancées, notamment l'affichage des marqueurs pour représenter les centres sportifs. J'ai intégré la carte dans l'interface de l'application et j'ai programmé la récupération des données sur les centres sportifs depuis la base de données.

J'ai choisi l'API Google Maps pour sa popularité, sa documentation complète et ses fonctionnalités riches. Elle permet aux utilisateurs de zoomer, de faire glisser la carte et d'obtenir des informations détaillées sur les centres sportifs en cliquant sur les marqueurs.



Cette fonctionnalité offre aux utilisateurs une expérience visuelle attrayante et intuitive, leur permettant de sélectionner facilement un centre sportif en fonction de sa localisation géographique.

3.2. Page détail

La fonctionnalité de la page détail permet aux utilisateurs d'accéder à des informations spécifiques sur un centre sportif sélectionné. Cette page fournit des détails essentiels tels que l'adresse du centre sportif, les prix, les horaires d'ouverture et la possibilité de réserver un terrain.

En accédant à la page détail d'un centre sportif, les utilisateurs peuvent consulter l'adresse complète, ce qui leur permet de planifier leur trajet et de se rendre facilement au terrain de football choisi.



ARENA

centre information

rue du bonheur, 24

04 65 32 15 48

Thursday : 08:00 - 02:00

Monday : 08:00 - 02:00

Friday : 08:00 - 02:00

Sunday : 08:00 - 02:00

Wednesday : 08:00 - 02:00

Tuesday : 08:00 - 02:00

Saturday : 08:00 - 02:00

Reservation calendar



La fonctionnalité de réservation sera présentée en détail dans une autre section, car elle nécessite une explication approfondie de son fonctionnement et de son intégration dans l'application.

La page détail permet également aux utilisateurs de laisser un commentaire sur le centre sportif et de consulter les commentaires laissés par d'autres utilisateurs. Cela favorise l'échange d'expériences et d'avis, offrant aux utilisateurs la possibilité de partager leur feedback et de prendre des décisions éclairées lors de la sélection d'un centre sportif.

Enfin, la météo de la date sélectionnée est également affichée sur la page détail. Cette fonctionnalité permet aux utilisateurs de vérifier les conditions

météorologiques prévues pour la date de leur réservation, ce qui peut influencer leur décision de jouer ou de choisir une autre date plus favorable.

En résumé, la page détail offre donc aux utilisateurs une vue complète et détaillée des informations essentielles liées à un centre sportif spécifique, leur permettant de prendre des décisions éclairées, de personnaliser leurs réservations et de laisser leurs avis.

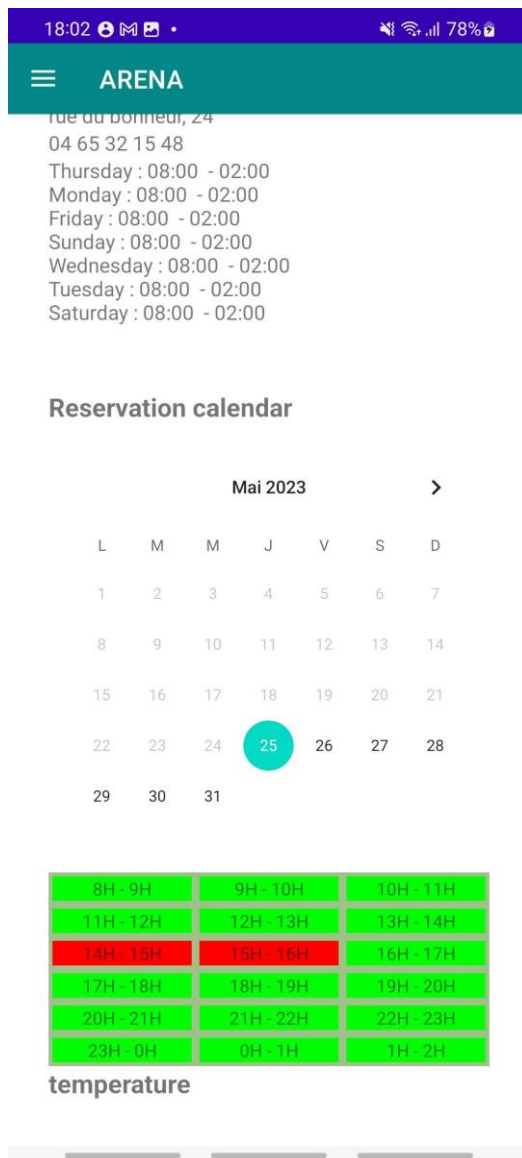
3.3. Réservations

La fonctionnalité de réservations constitue l'un des aspects essentiels de l'application Book Futsal. Elle permet aux utilisateurs de sélectionner une date et un créneau horaire disponibles pour réserver un terrain.

La mise en place de ce système de réservation a présenté certains défis techniques. L'un de ces défis était la gestion du calendrier et la visualisation des disponibilités des terrains. Pour surmonter ce défi, j'ai utilisé le composant "Calendar" fourni par Android Studio. Ce composant m'a permis d'afficher un calendrier interactif à l'écran, où les utilisateurs peuvent choisir la date de leur réservation.

Ensuite, pour afficher les créneaux horaires disponibles et permettre aux utilisateurs de les réserver, j'ai utilisé une grille (grid) contenant des TextView cliquables. Chaque TextView représente un créneau horaire (par exemple 08h-09h), et lorsqu'un utilisateur clique dessus, il peut effectuer une réservation pour ce créneau spécifique.

Un autre défi technique était de gérer les conflits de réservation et de garantir que deux utilisateurs ne puissent pas réserver au même temps. Pour résoudre ce problème, j'ai utilisé Firestore comme base de données pour stocker les informations sur les réservations. Lorsqu'un utilisateur souhaite réserver un terrain, une vérification est effectuée pour s'assurer que le créneau horaire sélectionné est toujours disponible. Si le créneau est déjà réservé, il sera affiché en rouge et l'utilisateur est informé de cette indisponibilité quand il clique dessus.



Enfin, la fonctionnalité de réservations a également été associée à un autre aspect essentiel de l'application, à savoir la gestion des utilisateurs. En effet, pour pouvoir effectuer une réservation, les utilisateurs doivent être connectés à l'application. Ainsi, lorsqu'un utilisateur tente de réserver un terrain disponible, une vérification est effectuée pour déterminer s'il est connecté ou non.

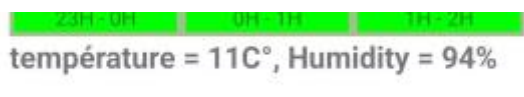
Si l'utilisateur n'est pas connecté, un message s'affiche lui indiquant qu'il doit d'abord se connecter pour pouvoir effectuer une réservation. Cette interaction entre la fonctionnalité de réservation et le système de connexion des utilisateurs garantit la sécurité et l'authenticité des réservations.

La fonctionnalité d'inscription et de connexion des utilisateurs sera abordée plus en détail dans une section dédiée de ce rapport.

En conclusion, la fonctionnalité de réservations a nécessité la mise en place d'un calendrier interactif et d'une grille horaire, ainsi que la gestion des conflits de réservation. Grâce à l'utilisation de composants fournis par Android Studio et à l'intégration de Firebase, j'ai pu surmonter ces défis techniques et offrir aux utilisateurs une expérience de réservation fluide et fiable.

3.4. Météo

La fonctionnalité de météo dans l'application Book Futsal a été mise en place en utilisant l'API météo d'OpenWeather (<https://openweathermap.org/api>). Cette API fournit des informations météorologiques précises et à jour.



Lorsque l'utilisateur sélectionne une date dans le calendrier, l'application utilise l'API météo pour récupérer les prévisions météorologiques du jour sélectionné. Ces informations sont ensuite affichées à l'utilisateur, lui permettant de prendre en compte les conditions météorologiques avant de réserver un terrain.

Cependant, il est important de noter qu'il existe une limitation dans la fonctionnalité de météo. Les prévisions météorologiques sont disponibles uniquement pour les 5 jours à venir maximum. Au-delà de cette période, les informations météorologiques ne sont pas fournies par l'API et ne peuvent donc pas être affichées dans l'application.

3.5. Commentaires

La fonctionnalité des commentaires dans l'application Book Futsal permet aux utilisateurs de laisser des commentaires sur les centres sportifs. Cette fonctionnalité est intégrée à la page de détails d'un centre sportif.

Dans la page de détails, seuls les trois commentaires les plus récents sont affichés initialement. Si d'autres commentaires sont disponibles, un bouton "More comments" s'affiche, permettant à l'utilisateur d'accéder à une autre page où tous les commentaires sont affichés.

comments

Leave your comment here...

POST

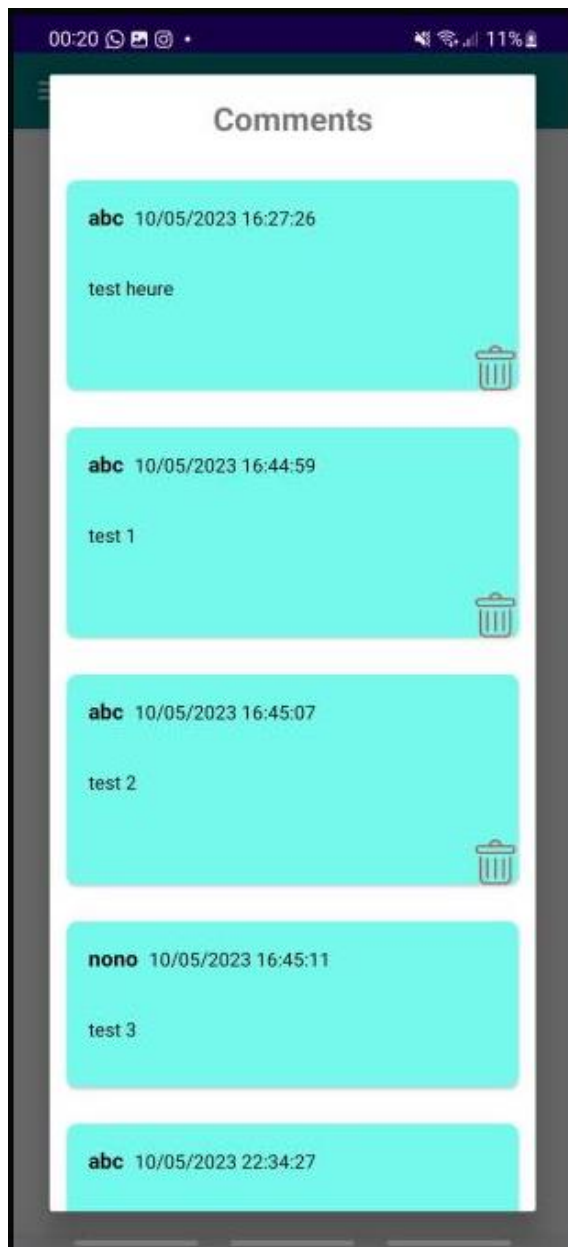
nono : 10/05/2023 16:45:11 :
magnifique 😊😊

abc : 10/05/2023 22:34:27 : color
comment

nono : 10/05/2023 16:45:11 : test 3

[More comments](#)

Sur cette page dédiée aux commentaires, chaque commentaire affiche l'auteur, la date et le contenu du commentaire. De plus, si l'utilisateur est l'auteur d'un commentaire, un logo de poubelle est affiché à côté du commentaire, lui permettant de le supprimer s'il le souhaite.



Pour implémenter cette fonctionnalité, j'ai utilisé les fonctionnalités de stockage de données de Firestore. Au début, j'avais envisagé de stocker tous les commentaires dans une seule chaîne de caractères (String) et de les récupérer en effectuant un parsing de cette chaîne. Cependant, après avoir exploré la documentation de Firestore, j'ai découvert qu'il était possible d'ajouter des attributs de type liste, ce qui m'a permis d'adopter une approche plus efficace.

Ainsi, chaque centre sportif contient une liste de commentaires. Chaque fois qu'un utilisateur ajoute un nouveau commentaire, il est ajouté à cette liste. Cette approche facilite la récupération des commentaires et permet également la suppression des commentaires.

▼ comments

- 0 "abc : 10/05/2023 16:27:26 : test heure "
- 1 "abc : 10/05/2023 16:44:59 : test 1"
- 2 "abc : 10/05/2023 16:45:07 : test 2"
- 3 "nono : 10/05/2023 16:45:11 : test 3"
- 4 "abc : 10/05/2023 22:34:27 : color comment "
- 5 "nono : 10/05/2023 16:45:11 : magnifique 🙌🙌"

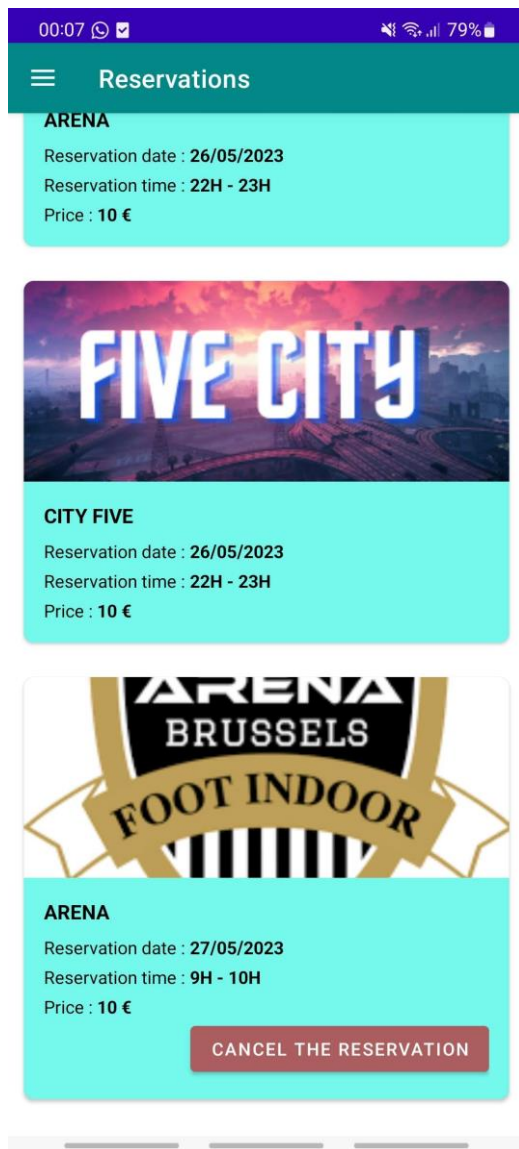
3.6. Gérer ses réservations

La fonctionnalité "Gérer ses réservations" permet à l'utilisateur de consulter toutes ses réservations et de les gérer, notamment en les supprimant si nécessaire.

Lorsqu'un utilisateur accède à cette fonctionnalité, il voit une liste de toutes ses réservations passées et futures. Chaque réservation affiche des détails tels que le centre sportif, la date, l'heure de réservation et le prix. De plus, les réservations sont affichées dans l'ordre (par date).

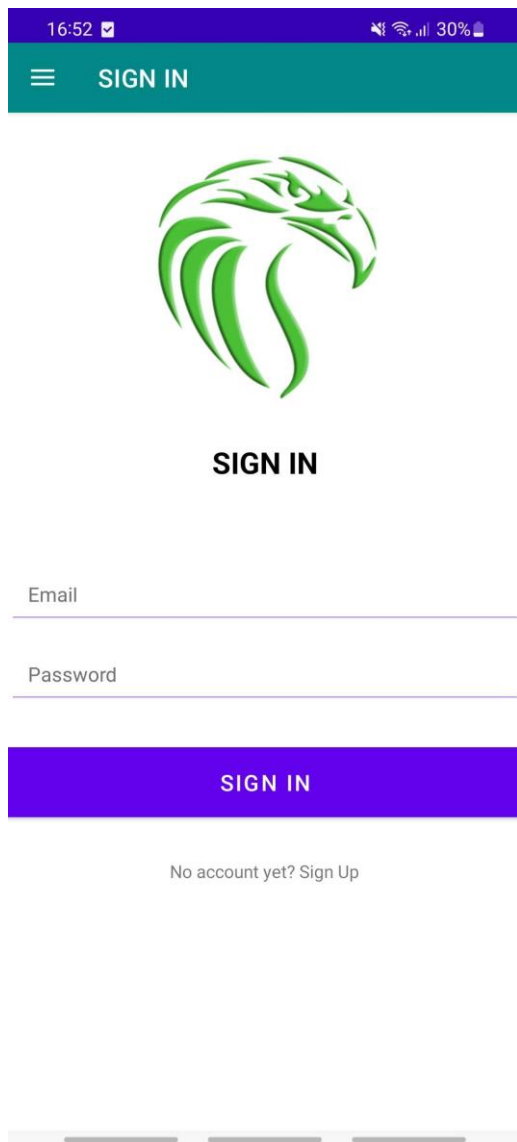
Cependant, il y a une limitation importante dans la gestion des réservations : toutes les réservations qui ont été effectuées dans les dernières 24 heures ne peuvent pas être annulées. Cette mesure vise à éviter les annulations de dernière minute qui pourraient causer des inconvénients aux centres sportifs. Un bouton « cancel reservation » est affichés pour les réservations qui peuvent être annuler.

Pour implémenter cette fonctionnalité, j'ai utilisé la fonction de stockage de données de Firestore pour enregistrer les réservations de chaque utilisateur. Lorsqu'un utilisateur souhaite supprimer une réservation, si elle peut être annulée, elle est supprimée de la base de données.



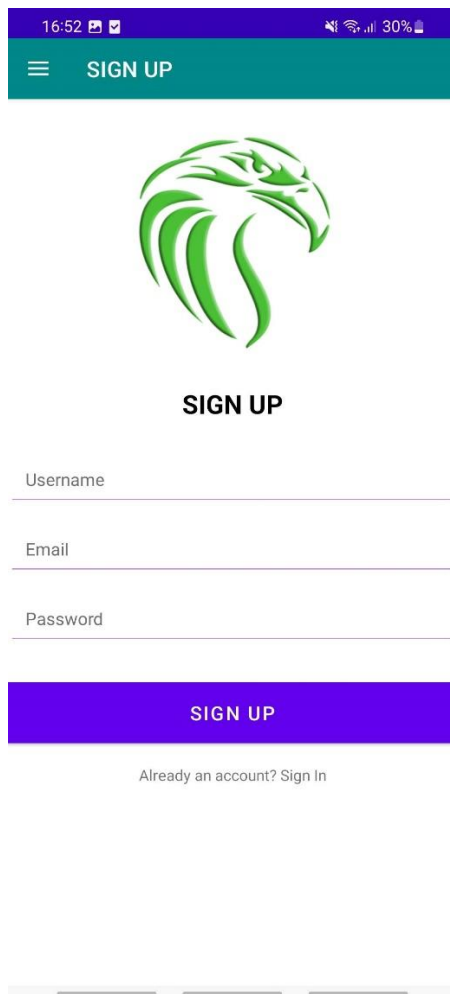
3.7. Connexion / inscription

La fonctionnalité de Connexion / Inscription permet aux utilisateurs d'accéder à l'application et de créer un compte s'ils n'en ont pas encore. J'ai utilisé Firestore pour le stockage des données et l'authentification pour mettre en œuvre cette fonctionnalité.




Lorsque l'utilisateur lance l'application, il peut accéder à une page de connexion où il peut saisir ses identifiants (adresse e-mail et mot de passe) s'il est déjà inscrit. S'il n'a pas de compte, il peut cliquer sur le bouton "Sign Up" pour accéder à une page d'inscription.

Dans la page d'inscription, l'utilisateur doit fournir ses informations personnelles telles que son nom, son adresse e-mail et son mot de passe. Une fois les informations saisies, elles sont envoyées à Firestore pour créer un compte utilisateur.



16:52 30%

≡ SIGN UP



SIGN UP

Username

Email

Password

SIGN UP

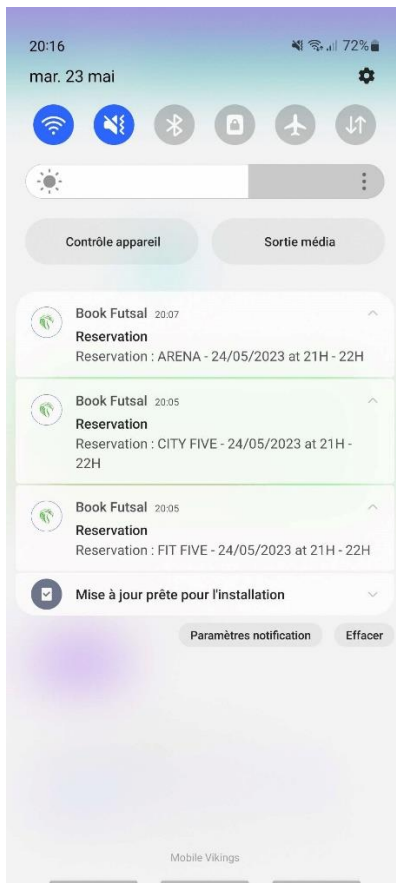
[Already an account? Sign In](#)

Lorsque l'utilisateur tente de se connecter ou de s'inscrire, une vérification est effectuée avec l'authentification Firestore pour confirmer que les informations fournies sont valides. Si les informations sont correctes, l'utilisateur est redirigé vers la page d'accueil de l'application où il peut commencer à utiliser toutes les fonctionnalités de l'application.

L'utilisation de Firestore pour le stockage des données permet de conserver en toute sécurité les informations utilisateur, telles que les informations de connexion et les détails du profil. L'authentification Firestore garantit que seuls les utilisateurs authentifiés peuvent accéder à toutes les fonctionnalités de l'application.

3.8. Notifications

La fonctionnalité de Notifications permet d'envoyer aux utilisateurs des rappels de réservation 24 heures avant le moment prévu pour la réservation. J'ai mis en place cette fonctionnalité en récupérant la date et l'heure de la réservation, puis en soustrayant 24 heures à cette date pour générer la notification.



Pour réaliser cette fonctionnalité, j'ai exploré différentes documentations et sources telles que Stack Overflow afin de trouver des solutions adaptées. Une fois que j'ai obtenu la date et l'heure de la réservation, j'ai utilisé des bibliothèques spécifiques d'Android Studio pour gérer les notifications.

Lorsque l'utilisateur effectue une réservation, la date et l'heure sont enregistrées dans la base de données. À partir de cette information, j'ai utilisé des fonctions de manipulation de dates pour soustraire 24 heures à la date de réservation. Une fois cette nouvelle date calculée, j'ai programmé l'envoi de la notification à ce moment précis.

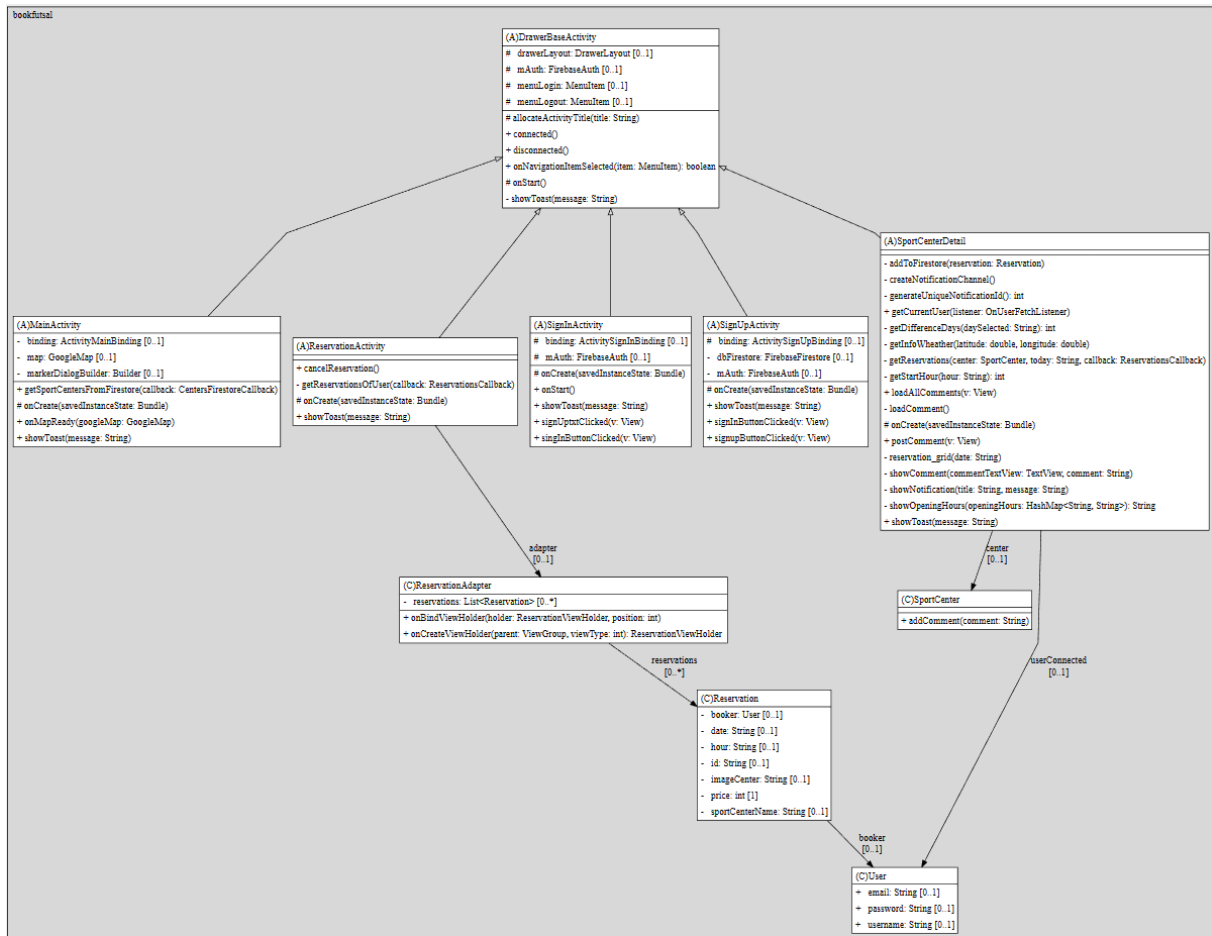
La notification de rappel est affichée sur l'appareil de l'utilisateur 24 heures avant le moment de la réservation. Elle contient des informations pertinentes telles que le centre sportif réservé, la date et l'heure de la réservation. L'utilisateur peut ainsi être informé à l'avance de sa réservation et se préparer en conséquence.

La mise en place de cette fonctionnalité de notifications a présenté quelques défis techniques, notamment la manipulation des dates et l'heure du début de la réservation (par exemple, de 8h à 9h, récupérer uniquement 8h). Cependant, grâce à des recherches approfondies et l'aide un peu de chatGPT (cette requête sera montrée à la fin de ce rapport), j'ai réussi à résoudre ces problèmes et à fournir aux utilisateurs une fonctionnalité pratique de rappel de réservation.

4 Analyse

Dans cette section d'analyse, je vais mettre en évidence les points d'intérêt concernant la structure de mon implémentation pour l'application Book Futsal. Je vais utiliser des diagrammes pour illustrer et expliquer ces aspects importants. Chaque diagramme sera accompagné d'un commentaire détaillé pour faciliter la compréhension.

4.1. Diagramme de classe



Le diagramme de classe représente la structure de mon code et les interactions entre les différentes entités de mon application Book Futsal.

Au sommet de la hiérarchie, nous avons l'activité mère appelée « DrawerBaseActivity » qui contient le menu de navigation. Cette activité est héritée par toutes les autres activités de l'application, ce qui évite la duplication de code et assure la présence du menu de navigation dans toutes les parties de l'application. De plus, cette activité permet également de gérer la connexion de l'utilisateur en vérifiant s'il est déjà connecté ou non, évitant ainsi de se reconnecter à chaque utilisation de l'application.

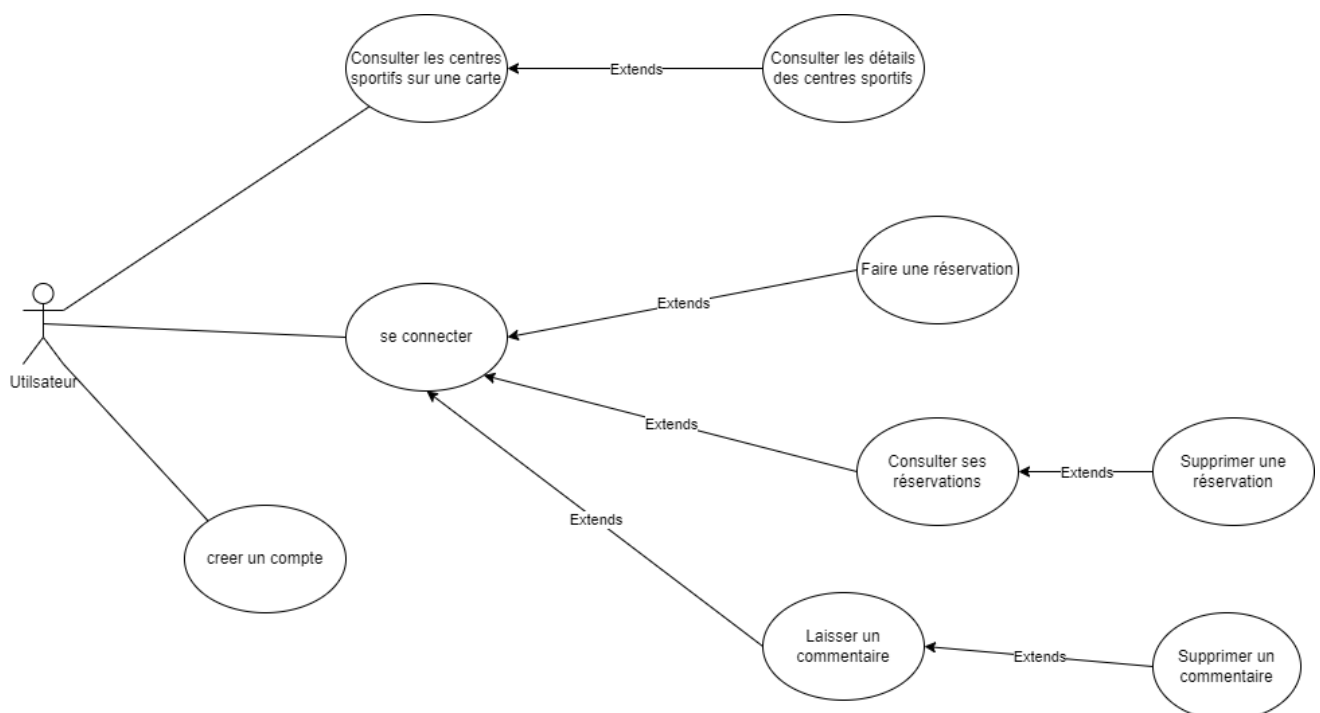
Ensuite, nous avons l'activité « MainActivity », qui représente la page d'accueil de l'application. Cette activité affiche la carte avec tous les centres sportifs disponibles.

L'activité « ReservationActivity » est responsable de la gestion des réservations des utilisateurs. Elle fait appel à un adaptateur « ReservationAdapter » qui permet d'afficher toutes les réservations sous forme de liste. Cet adaptateur utilise le modèle « Reservation » pour stocker les informations de chaque réservation, et ce modèle à son tour utilise le modèle « User » pour enregistrer les informations sur l'utilisateur qui a effectué la réservation.

Les activités « SignInActivity » et « SignUpActivity » permettent aux utilisateurs de se connecter et de s'inscrire.

Enfin, l'activité « SportCenterDetail » contient toutes les informations détaillées d'un centre sportif spécifique. Elle utilise le modèle « SportCenter » pour récupérer les informations relatives à ce centre spécifique, et également le modèle « User » pour stocker des informations liées à une réservation.

4.2. Diagramme de cas d'utilisation



Le diagramme de cas d'utilisation permet de visualiser tout ce que l'utilisateur peut faire du point de vue fonctionnel.

Lorsque l'utilisateur lance l'application, il a accès à certaines fonctionnalités limitées. Pour profiter pleinement de toutes les fonctionnalités offertes par l'application, il doit se connecter. Sans la connexion, il peut consulter les centres sportifs disponibles et accéder à des informations détaillées sur chaque centre

sportif, telles que les horaires, les places disponibles, les commentaires des autres utilisateurs, les tarifs, etc

Une fois connecté, l'utilisateur a la possibilité d'effectuer des réservations dans un centre sportif. Après avoir effectué une réservation, il peut consulter la liste de ses réservations pour voir les détails et gérer celles-ci. Si nécessaire, il peut supprimer une réservation existante. Il peut également ajouter des commentaires et les supprimer s'il le souhaite. s

5 UI/UX

L'interface utilisateur et l'expérience utilisateur sont des aspects essentiels lors du développement d'une application. Dans le cadre de ce projet, j'ai accordé une attention particulière à l'UI/UX afin de fournir aux utilisateurs une expérience agréable. Voici les points d'attention que j'ai pris en compte :

5.1. Design visuel attrayant

J'ai veillé à ce que l'application présente un design visuel attrayant en utilisant une palette de couleurs cohérente, des icônes claires et une disposition soignée des éléments. J'ai également utilisé des polices lisibles pour améliorer la lisibilité du contenu.

Le design visuel a été amélioré grâce aux commentaires et aux retours des utilisateurs lors du speed testing réalisé en classe. En tenant compte de leurs suggestions, j'ai apporté des modifications pour améliorer l'expérience utilisateur. Voici quelques exemples :

- Page de connexion :

Avant :



SIGN IN

Email

Password

SIGN IN

No account yet? [Sign Up](#)



Après :



Email

Mot de passe

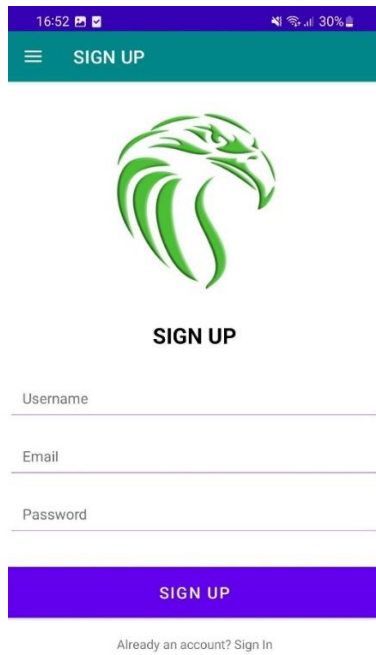
SIGN IN

No account yet? [Sign Up](#)




- Page d'inscription :

Avant :



16:52 30%

≡ SIGN UP



SIGN UP

Username

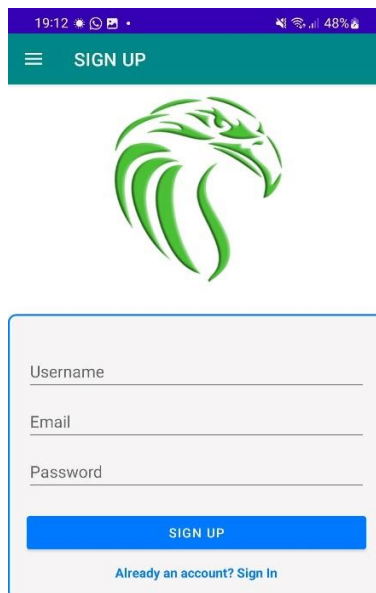
Email

Password

SIGN UP


[Already an account? Sign In](#)

Après :



19:12 48%

≡ SIGN UP



Username

Email

Password

SIGN UP

[Already an account? Sign In](#)

5.2. Navigation facile

J'ai mis en place un menu de navigation accessible depuis toutes les activités de l'application. Cela permet aux utilisateurs de naviguer facilement entre les différentes fonctionnalités sans perdre leur orientation.

5.3. Feedback utilisateur

J'ai intégré des mécanismes de feedback utilisateur pour informer les utilisateurs des actions en cours.

Par exemple, lorsqu'un utilisateur effectue une réservation avec succès, un message de confirmation s'affiche pour lui indiquer que sa réservation a été enregistrée. De plus, une notification de rappel de réservation est envoyée aux utilisateurs 24 heures avant la date de réservation pour les aider à se souvenir.

De même, lors de l'inscription et de la connexion, des messages d'indication sont affichés en cas de mauvaises saisies ou en cas de mauvaises données.

5.4. Adaptabilité aux différents appareils

J'ai pris en compte la responsivité de l'application en la testant sur différents appareils mobiles et en m'assurant que l'interface s'adapte de manière fluide aux différentes tailles d'écran. Cela garantit une expérience utilisateur cohérente, quel que soit l'appareil utilisé.

5.5. Possibilité d'ajouter des commentaires

J'ai intégré une fonctionnalité permettant aux utilisateurs d'ajouter des commentaires pour partager leurs expériences et opinions sur les centres sportifs. Cette fonctionnalité offre aux utilisateurs la possibilité d'exprimer leurs avis et d'interagir avec d'autres utilisateurs de l'application.

5.6. Conclusion

J'ai fait tester l'application à mon entourage et recueilli le feedback lors d'un speed testing en classe. Ces retours m'ont permis d'ajuster l'UI/UX de l'application en prenant en compte les besoins et les préférences des utilisateurs. J'ai ainsi amélioré la disposition des fonctionnalités, les éléments de design et la facilité d'utilisation, offrant ainsi une meilleure expérience utilisateur.

6 Limitations et développement futur

Bien que mon application ait été développée avec soin et ait été testée avec succès, il y a encore quelques limitations et points de progrès qui méritent d'être mentionnés.

6.1. La géolocalisation

Actuellement, l'application ne prend pas en charge la géolocalisation pour permettre à l'utilisateur de trouver automatiquement les centres sportifs les plus proches de sa position. Cela peut rendre la recherche plus efficace, surtout pour les utilisateurs qui ne sont pas familiers avec la région.

6.2. Ajout d'un nouveau centre sportif

Pour ajouter un nouveau centre sportif, il faut aller dans la base de données et l'ajouter manuellement. Il serait intéressant d'ajouter une section dédiée pour l'admin qui permet directement d'ajouter un nouveau centre sportif.

6.3. Gestion des horaires spéciaux

Actuellement, l'application ne prend pas en compte les horaires spéciaux des centres sportifs, tels que les jours fériés ou les événements spéciaux. L'ajout d'une fonctionnalité permettant aux administrateurs de gérer ces horaires spéciaux et de les afficher correctement dans l'application améliorerait la précision des informations pour les utilisateurs.

6.4. Amélioration de la sécurité

Bien que l'authentification des utilisateurs soit déjà mise en place, il est important de continuer à renforcer la sécurité de l'application. Cela pourrait inclure l'implémentation de mesures de protection supplémentaires, telle que la vérification en deux étapes.

7 Conclusion

Dans le cadre de ce projet, j'ai développé une application de réservation de terrains de football avec diverses fonctionnalités pour améliorer l'expérience des utilisateurs. J'ai réussi à mettre en place les fonctionnalités principales telles que la visualisation des centres sportifs sur une carte, la réservation de terrains, les commentaires, la gestion des utilisateurs et les notifications de rappel de réservation.

Cependant, certaines fonctionnalités n'ont pas été implémentées comme la géolocalisation, la gestion des horaires spéciaux et l'ajout automatique de nouveaux centres sportifs qui pourraient être développés à l'avenir pour améliorer encore plus l'application.

Ce projet m'a permis d'acquérir de précieuses compétences en développement mobiles, en utilisant Android Studio et les technologies associées telles que Firebase pour la gestion des données. J'ai également appris à résoudre des problèmes techniques, à effectuer des recherches approfondies et à tirer parti des ressources disponibles pour surmonter les obstacles rencontrés lors du développement de l'application.

Enfin, bien que le projet présente certaines limitations et des possibilités de développement futur, je suis satisfait du résultat obtenu. J'ai pu mettre en place des fonctionnalités clés, fournissant ainsi aux utilisateurs une application fonctionnelle et pratique pour la réservation de terrains de football. Ce projet m'a permis d'augmenter mes compétences en développement mobile et ma capacité à résoudre des problèmes techniques. Je suis convaincu que les connaissances et l'expérience acquises seront bénéfiques pour mes futurs projets de développement d'applications.

8 Sources

Météo api : <https://openweathermap.org/api>

Utilisation de l'api météo : <https://www.youtube.com/watch?v=Tic83VMltHg>

Documentation firestore : <https://firebase.google.com/docs/firestore?hl=fr>

Notification :

<https://developer.android.com/develop/ui/views/notifications/build-notification>

Connexion / inscription : <https://www.youtube.com/watch?v=QAKq8UBv4GI>

Api maps:

<https://console.cloud.google.com/welcome?hl=fr&pli=1&project=pivotal-pattern-381922>

Menu de navigation : <https://www.youtube.com/watch?v=pRieCkF1Yts&t=122s>

Adapter : <https://www.youtube.com/watch?v=gGFvbvkZiMs>

Avoir l'utilisateur connecté :

<https://developer.android.com/guide/topics/location/transitions?hl=fr>

Stack Overflow : <https://stackoverflow.com>

ViewBinding : <https://developer.android.com/topic/libraries/view-binding>

9 ChatGPT

- Requête pour avoir un déclencheur 24heures avant une date précise :

0

comment declencher une notification 24h avant une date spécifique sur android studio?



Bien sûr ! Voici le code correspondant à votre demande :



java

Copy code

```
// Récupérer la date et l'heure actuelles
Calendar calendar = Calendar.getInstance();
calendar.setTimeInMillis(System.currentTimeMillis());
calendar.set(Calendar.YEAR, Integer.parseInt(year)); // Année
calendar.set(Calendar.MONTH, Integer.parseInt(month) - 1); // Mois -1 car java
calendar.set(Calendar.DAY_OF_MONTH, Integer.parseInt(day)); // Jour

// Définir l'heure spécifique
calendar.set(Calendar.HOUR_OF_DAY, startHour); // Heure

// Soustraire 24 heures à la date et l'heure de la réservation
int minuteBeforeNotification = -1440; // 60 min * 24 => 1440 => 24h

calendar.set(Calendar.MINUTE, 0); // Réinitialiser les minutes à 0
calendar.add(Calendar.MINUTE, minuteBeforeNotification);

// Calculer la différence de temps entre l'heure actuelle et l'heure de la r
long timeDifference = calendar.getTimeInMillis() - System.currentTimeMillis();

if (timeDifference > 0) {
    // Créer un objet Handler pour afficher la notification 24 heures avant
    Handler handler = new Handler();
    handler.postDelayed(new Runnable() {
        @Override
        public void run() {
```

```

if (timeDifference > 0) {
    // Créer un objet Handler pour afficher la notification 24 heures avant
    Handler handler = new Handler();
    handler.postDelayed(new Runnable() {
        @Override
        public void run() {
            // Afficher la notification
            NotificationCompat.Builder builder = new NotificationCompat.Builder
                .setSmallIcon(R.drawable.ic_notification)
                .setContentTitle("Notification")
                .setContentText("Votre notification 24 heures avant la d
                .setPriority(NotificationCompat.PRIORITY_DEFAULT);

            NotificationManagerCompat notificationManager = NotificationMana
            notificationManager.notify(1, builder.build());
        }
    }, timeDifference);
}

```

N'oubliez pas de personnaliser ce code en fonction de votre cas d'utilisation spécifique et d'ajouter les fonctionnalités supplémentaires que vous souhaitez pour vos notifications.

