Projet Développement

Mobile 2023

Une image contenant texte, clipart

Description générée automatiquement

Réalisé par Popadiuc Claudiu

Supervisé par Monsieur Riggio, Jonathan

Table des matières

[Introduction 3](#_Toc134744141)

[Description des technologie utilisées 4](#_Toc134744142)

[Sujet d’application 5](#_Toc134744143)

[Fonctionnalités 6](#_Toc134744144)

[Analyse 19](#_Toc134744145)

[Limitations et développement futur 21](#_Toc134744146)

[Play Store 22](#_Toc134744147)

[Conclusion 23](#_Toc134744148)

# **Introduction**

Dans le cadre du cours de Mobile, il nous a été demandé de réaliser une application mobile qui sera développée uniquement sur Android Studio en utilisant le langage Java. Pour ma part, j'ai réalisé une application de gestion des matchs de football destiné aux arbitres, comprenant un historique des matchs ainsi que les profils des joueurs.

Dans ce rapport, je vais commencer par une description des technologies utilisées ainsi que mes motivations. Ensuite, je rentrerai dans le vif du sujet en expliquant le sujet de mon application suivi par les fonctionnalités de mon projet, ainsi que tout ce qui a été implémenté d'un point de vue à la fois fonctionnel et technique, en mettant en avant les défis techniques et les solutions apportées. Les points d'intérêt concernant la structure de mon implémentation seront ensuite présentés sous forme de diagramme.

Pour donner suite à cela, il y aura une section UI/UX qui détaillera les points d'attention apportés à l'UI/UX, ainsi que la façon dont les besoins et les préférences des utilisateurs ont été pris en compte.

Je continuerai mon rapport avec les limites de mon application et je répondrai à des questions telles que dans quels cas d'utilisation mon application pourrait ne pas fonctionner comme prévu ? Y a-t-il des aspects techniques non traités ? Si j’avais plus de temps pour le projet, qu'aurais-je amélioré ?

Je finirai ensuite mon rapport avec une conclusion en résumant ce qui a été vu dans le rapport, ce que j'ai réussi à faire ou non durant le projet, et les apprentissages que j'en tire. Pour finir, je donnerai mon avis et mon ressenti sur le développement de mon application.

# **Description des technologie utilisées**

Pour réaliser ce projet, j'ai utilisé exclusivement Android Studio comme environnement de travail. Android Studio est un environnement de développement intégré (IDE) utilisé pour créer des applications mobiles pour la plate-forme Android. Les langages de programmation possibles sont Java et Kotlin. Android Studio est largement utilisé par les développeurs du monde entier pour créer des applications mobiles pour la plate-forme Android.

Pour mon projet, j'ai utilisé uniquement le langage Java. Java est un langage de programmation populaire utilisé pour créer une grande variété d'applications, notamment des applications de bureau, mobiles, et web. Il est orienté objet, portable et largement utilisé dans l'industrie pour sa fiabilité et sa sécurité.

Dans Android Studio, le code Java est associé au code XML qui est disponible lors de la création d'une activité. XML est un langage de balisage utilisé pour décrire la structure et le contenu des données de manière structurée et lisible par les machines. Il est largement utilisé dans le développement d'applications web et d'autres domaines pour échanger et stocker des données.

# **Sujet d’application**

De manière simple et structurée, voici les fonctionnalités de mon application :

Le but de mon application est de permettre à l'utilisateur d’avoir sa propre saison de football avec les joueurs qui y participent. L'application permettra de créer des matchs et d'enregistrer l'historique de tous les matchs précédents. Chaque joueur pourra avoir son propre profil avec des statistiques telles que le nombre de défaites ou de victoires.

Une notification sera envoyée pour le chronomètre qui décompte le temps restant lorsqu'un match est joué, de sorte que l'arbitre n'aura pas besoin de se connecter à l'application pour consulter le temps restant. Une autre notification sera envoyée à l'arbitre une fois que le match sera terminé.

L'application sera également compatible avec les tablettes. Il y aura également une API qui permettra d'indiquer la position des joueurs comme pour les joueurs professionnels (ceci est expliqué plus en détail dans la section sur l'API).

Une carte sera disponible lors de la création de chaque match par l'arbitre, qui pourra ajouter le lieu où le match est joué. Dans l'historique, l'emplacement exact où le match de football a été joué sera disponible.

Voici le fonctionnement de l'application :

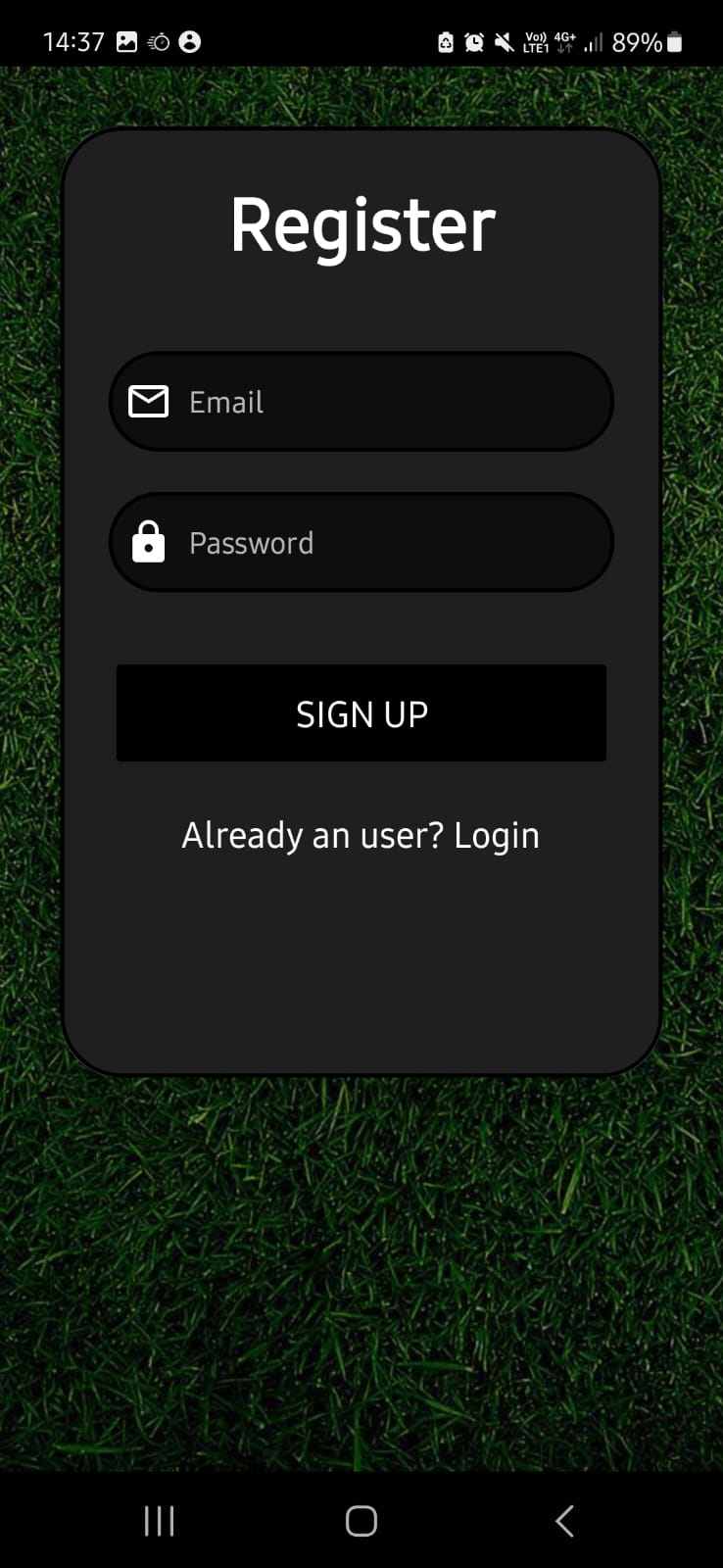
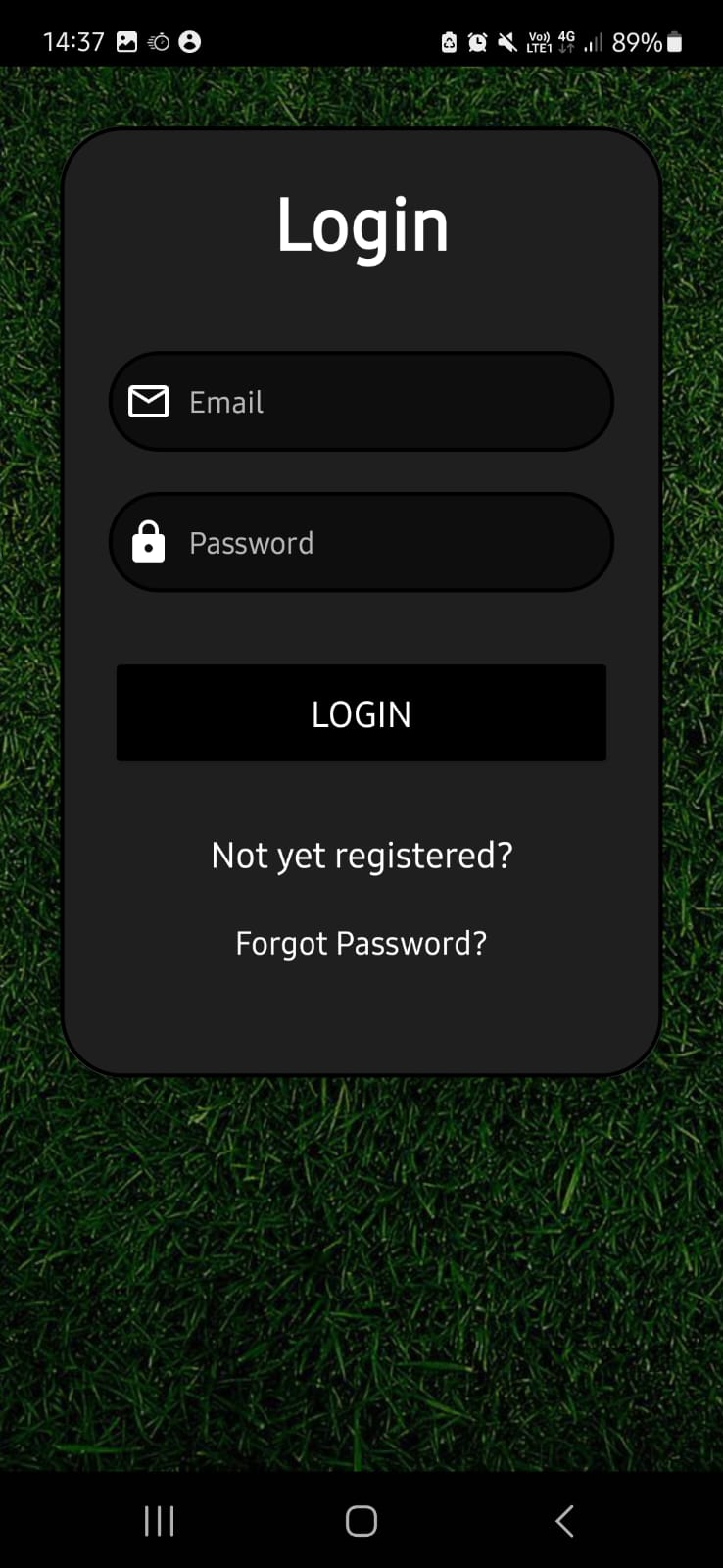
Un utilisateur poura s’inscire ou non dans l’application, Un utilisateur sera soit « Arbitre » soit « Joueur » S’il est arbitre, il pourra créer des profils pour les joueurs qui auront accès à l’application. L'arbitre pourra également créer des matchs qui seront enregistrés dans l'historique. La différence entre l’utilisateur « arbitre » et l’utilisateur « joueur » c’est que les joueurs ne pourront pas créer de matchs. Ils pourront seulement consulter l'historique des matchs ainsi que les profils des joueurs de la saison sans pouvoir modifier leurs statistiques. Seul l'arbitre pourra le faire.

# **Fonctionnalités**

**Le processus d’inscription/Connexion.**

Page d’acceuil : Page d’inscription : Page de connexion :

Une image contenant texte, arbre, plante, signe

Description générée automatiquement  

Sincèrement, je n'ai pas rencontré de difficultés pour implémenter le system de connexion et d’inscription. J'ai suivi un tutoriel sur YouTube qui m'a permis de comprendre et de développer cette fonctionnalité dont je vais parler plus tard. Au début, j'avoue que c'était compliqué et j'ai dû tester cela dans un nouveau projet avant de pouvoir l'implémenter dans mon projet actuel. Cependant, après quelques heures, j'ai réussi à faire fonctionner l'inscription et la connexion. Mais je vais rentrer dans les details pour cela.

Avant tout, il est important de préciser que mon application n'exige pas nécessairement une inscription. Cependant, si l'utilisateur décide de ne pas s'inscrire, certaines fonctionnalités de l'application lui seront inaccessibles (ce qui serait dommage, d'ailleurs). Par exemple, sur la page d'accueil, les boutons "Historique" et "Statistiques" ne seront disponibles qu'en étant connecté. Par conséquent, il sera obligatoire d'avoir un compte utilisateur pour profiter pleinement de l'application. Toutefois, si l'utilisateur souhaite réellement éviter de s'inscrire, il pourra créer un match qui ne pourra pas être sauvegardé.

Ainsi, si l'utilisateur désire d’avoir un avec un maximum de sensation sur l'application, il devra créer un compte. Comme vous pouvez le constater sur ma page d'inscription, il y a seulement deux champs à remplir : l'adresse e-mail et un mot de passe pour créer un compte. Sur mon application, c’est tres limité car l’application n’est pas poussée pour faire autre chose de plus complexe.

Pour mettre en place cette fonctionnalité, j'ai suivi un tutoriel disponible ici : <https://www.youtube.com/watch?v=TStttJRAPhE&t=650s>

J’ai dû utiliser Firebase qui est disponible directrement dans android studio dans les Tools, j’ai conencter mon projet avec Firebase, dans la partie authentification pour cette partie d’inscritpion avec mail et mots de passe :

Une image contenant texte

Description générée automatiquement



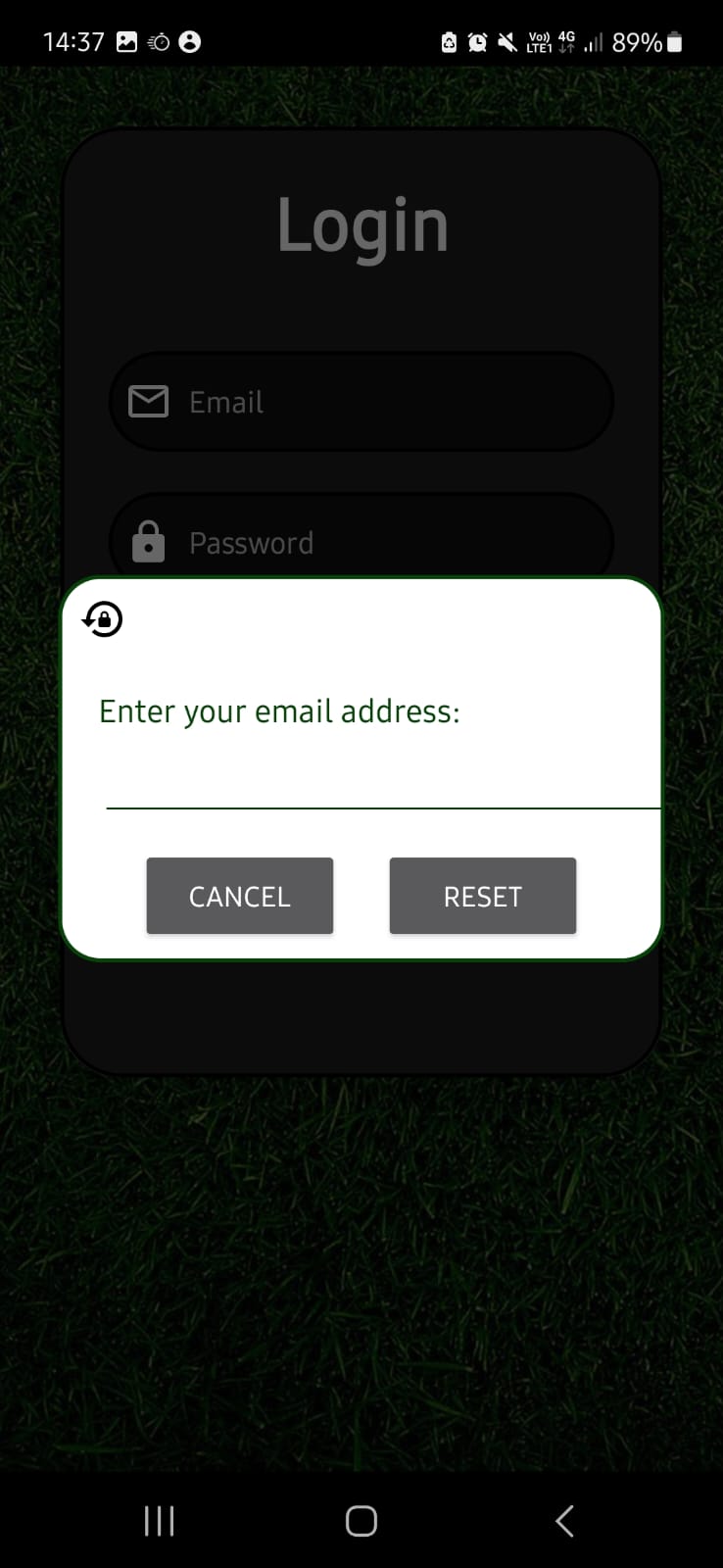
\*Afin de rester profesionnel, j’ai voulu respecter la vie privée de mes utilisateurs alors j’ai baré en bleu leur adresse mail.

Ce qui a été complique pour realiser cette fonctionalité c’est que j’aurais voulu utiliser la Real Time Database pour avoir mes utilisateurs dedans mais je n’ai pas réussi a le realiser, je suivis plusieurs tuto, j’ai couper mon proejt en plusieurs parties, mais je n’ai malheureusement pas réussi a faire cela, alors j’ai preferer rester sur le system de authentification qui est largement suffisant dans le cadre de mon application.

Pour rentrer dans les details, c’est lorsque j’appuie sur le bouton « Sign Up » que toute la partie de code se fera. Je vais d’abord verifier si le champs mail correspond bien à la strucutre mail avec « @ », je verifie aussi si le champs n’est pas vide ou alors si pour le mots de passe il ny’a pas au moin 8 caractere et si tout cela est bon alors on va ajouter un utilisateurs avec une adresse mail et un mots de passe dans le system de auhtentification.

Pour la connexion le principe sera legerement different car il verifiera si l’email et le mots de passe encoder fait bien parties des utilisateur disponible dans le system de authentification sinon il renvera un Toast.

Si l’utilisateur n’arrive plus a se connecter car il possiblement oublié sont mots de passe, un system de recuperation de mots de passe est disponible dans la page de connexion, a voir dans la page. Voici ce qui apparaitra lorsque j’appuyerais sur « Forgot password ? » :

 Une image contenant texte

Description générée automatiquement

Un email de récuperation de mots de passe sera alors envoyé, apres avoir tester cette fonctionalité, ça fonctionne bel et bien.

**Les fonctionnalités propres à mon application**

**Création de match**

TeamSelection : Team : Live :

 Une image contenant texte

Description générée automatiquement 

Voici le fonctionnement de la création de match :

Après avoir appuyer sur le bouton « New Game » qui est dans la page d’acceuil, on arrive sur la premiere page qui demande tout simplement quel type d’equipe desirer vous, si c’est random alors on arrivera sur la page Team, et dans cette page ci, on devra simplement selectionner des joueurs qui sont disponible dans mes statistiques, si on clique sur un des joueurs ca va l’ajouter dans le cadre « Players » et se suprimera de la liste des joueurs disponible pour indiquer a l’utilisateur qu’il a été selectionner et pour pas avoir de doublons, si l’utilisateur desir ajouter un joueur qui ne se trouve pas dans les statistiques pour ce match, alors il poura l’ajouter manuellement en appuyant sur le cadres « Players » et en ecrivant le nom du joueurs suivi d’une virgule comme indique juste au dessus du cadre. Une fois cela fait, l’utilisateur devra ajouter un temps de jeux pour chaque mi temps et pour la pause, qui ne pourra pas depasser 2 chiffres, donc une mi temps ne peux pas durer plus de 99 minutes, meme chose pour la pause, afin d’eviter des nombres abérant comme 45125 minutes pour une pause.

Si on n’ajoute pas au minimum 2 joueurs, et qu’on n’indique pas le temps de jeux pour chaque champ, alors le match ne peux pas commencer ce qui est logique, il n’ya aps de match sans temps de jeux et pas d’equipes sans au moin 1 joueurs dans chaque equipes.

Une fois tout cela bien rempli, nous arrivons sur la derniere partie qui est le Live du match, c’est ici que l’arbitre poura ajouter le nombre de goal en cliquant sur les balons de football pour chaque equipe, ou par exemple si les equipes qui ont été tirer au sort ne sont pas asse bien alors il y a le bouton « Refresh » en bas a guache qui permet de relancer le sort et faire 2 nouvelles equipes avec les joueurs qui on été choisi pour jouer le match, et remets les valeurs des goals a 0. Si tout est bon le match peux commencer en cliquant sur le bouton « Start » qui la est mis sur « Pause » car le match est en train de se dérouler et le temps défile déjà, et on peux mettre le chronometrer sur pause si on desire mettre sur pause le match.

Le match se deroule calmement, une fois que le match est terminé, il poura etre enregister en cliquant sur le bouton en bas a droite, qui demandera via un Builder s’il est sur d’enregistrer le match maintenant, car si l’utilisateur le souhaite il peux enregistrer un match avant que le temps de jeux soit terminée.

En haut a droite du Live il y a une icone qui permet de aller directement dans l’historique des matches joué, si l’utlisateur va dans cet historique le match en cours sera suprimer si il n’est pas enregistrer, alors un builder est aussi envoyer pour indiquer si il veux quiter le live sans enregistrer par contre si le match est enregistrer, le builder n’est pas envoyer.Il faut savoir que pour le system d’equipes « Selected » reprend le meme principe que pour le « Random «  sauf que ca ne va pas melanger mais, il y aura 2 champs dans la séléction, un pour chaque equipe et les joueurs pouront etre ajouter de la meme facon que pour le random.Pour le live du match « Selected » le bouton en bas a gauche ne permetera pas de re randomiser les equipes comme pour les equipes Random car les equipes seront déjà choisi par l’arbitre lui-même, donc pas besoin de faire ca, par contre si on appuie sur ce bouton les goals seront aussi remis à 0.

Meme si on ne peux pas le voir comme cela, mais il y a des animations entre les activiter par exemple on peux appuyer sur Random ou slider vers la gauche comme la fleche l’indique de meme pour Slected.

D’un point de vue thecnique tout cela n’est pas bien compliqué a comprendre, Pour la partie Random le champs d’equipe est un edit Text, pour les joueurs a selectionner c’est une listView avec comme Item tout les joueurs de pas base de donnée REAL TIME :

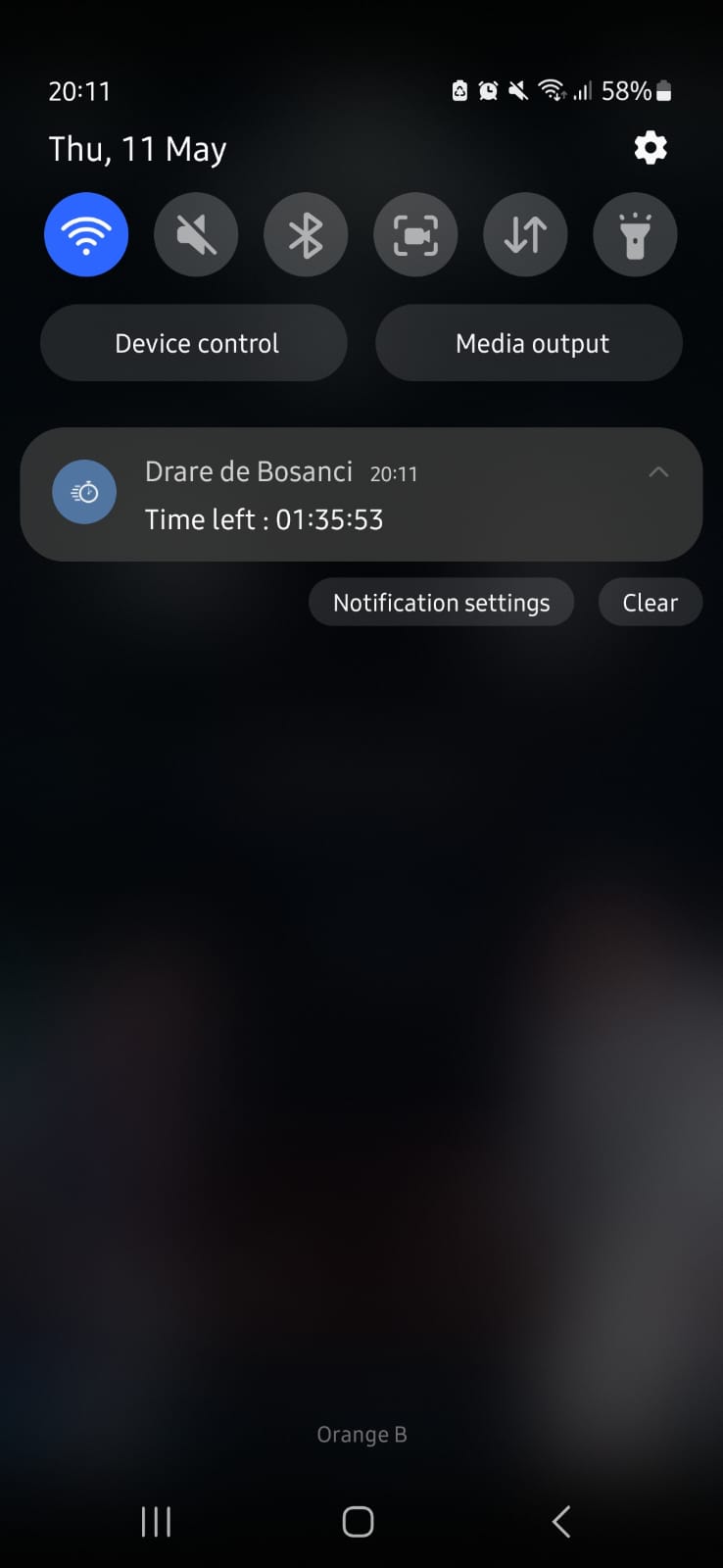
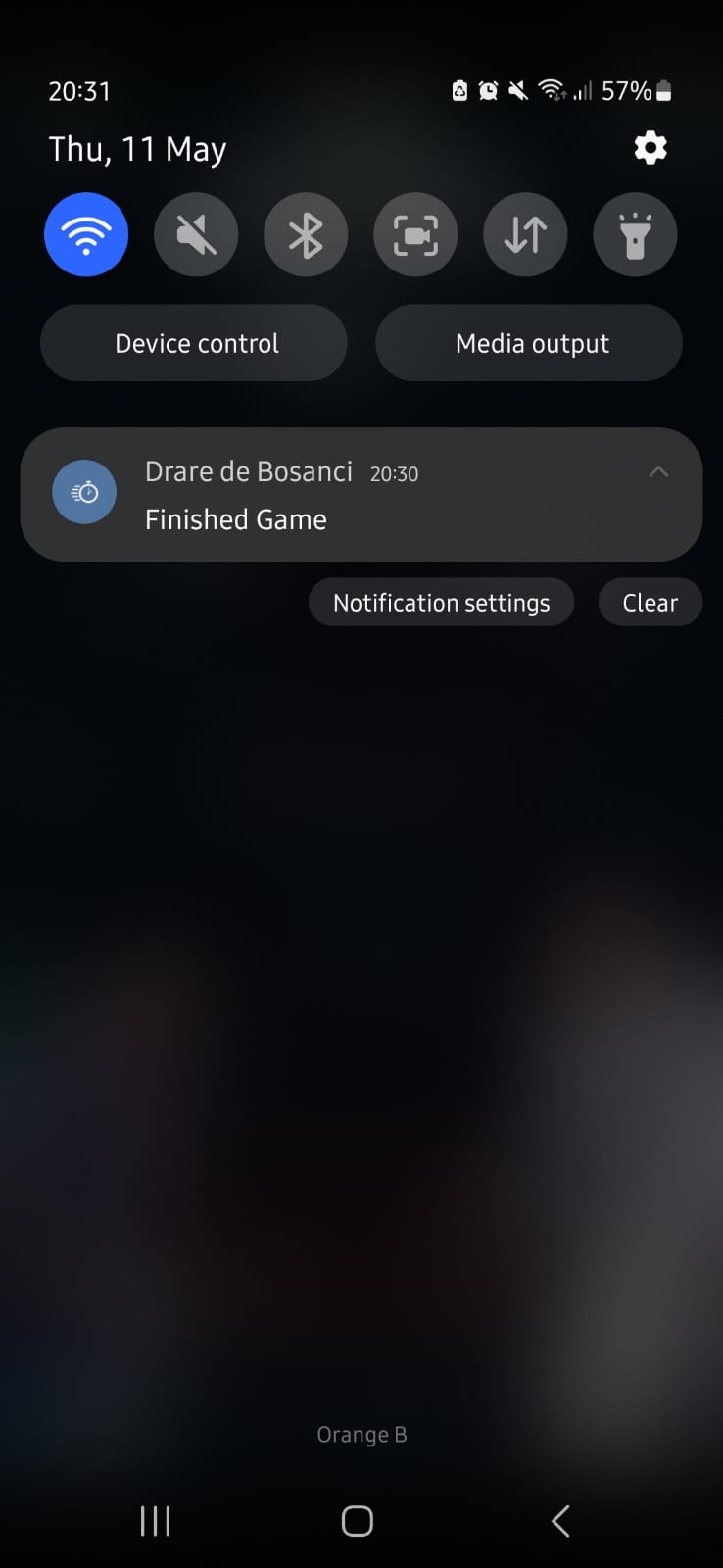
Une image contenant texte

Description générée automatiquement

Comme vous pouvez le voir ce sont tout les joueurs de mes statistiques, y seront ajouter dans ma list, et chaque item pourra etre cliquer afin d’etre repris dans l’edit text, ma liste a un Listener afin d’ecouter lorsque une modification a la liste est realsier et modifier la liste, donc quand j’appuie sur l’item il va suprimer le nom de la personne et grace a l’ecouteur, cela va fonctionenr sans problemes, enfin pour chaque item selectionner le text de l’item est simplement ajouter a l’edit text suivi d’une virgule, pourquoi ? car ensuite je vais mettre ce qui’il dans l’edit text dans une liste et la virgule me servira a diviser le nombres et les noms des joueur, cela sera donc fait lorsque on appuyera sur « Start », il mettra tout les joueurs dans une liste, et cette liste sera melanger grace Collection.Shuffle et ensuite on va diviser tout simplement en 2 listes la liste de base il y aura donc une liste avec les joueurs random par equipes . Le meme principe est applique pour les equipe predefini mais il n’yaura pas de random, ce sera directement le champs equipe 1 avec sa propre liste et de meme pour l’equipe 2, chaque equipe aura sa propre liste. Et une fois que le match commence chaque liste sera simplement afficher dans un TextView qui est celui en dessous du temps, la ou se trouve les noms, dans la page Live dans le grand cadre, ainsi chaque joueur sera affiche dans son equipe et comme vous pouvez le voir dans le screen si il y a un nombre impaire cava quand meme créer des equipes avec 3 et 4 joueurs et pas faire des 2 et 6 joueurs, ce qui est pratique pour garder des equipes equitable si c’est random.

Pour les champs du temps, j’ai ajouter une contraite qui permet d’ajouter que 2 chiffres maximum grace a un ecouteur de temps qui verifie a chaque fois que un chiffre est ajouter dans le champs la longeur de la chaine qui doit etre inferieur a 2, j’ai egalement ajouter un input de type chiffre dans ces champs la, ca veux dire que les lettres ne seront pas permise dans ces champs la.

**Notifications**

Mon system de notification est tres simple, une notification est envoyée lorsque l'arbitre a créé un match et qu’on arrive dans la page du Live. Un chronomètre est alors décompté dans la notification. Lorsque le match est terminé, l'arbitre reçoit une notification indiquant que le match est fini dans les screens ci-dessus il est 20h31 car j’ai juste enlever le chrono de 1h35 pour pas avoir 1h35 d’attente pour faire le screen de la notification que le match est fini, je dit cela simplement pour que vous sachier que rien n’est truqué, j’ai aussi ajouté le fait que si on appuie sur le bouton pause, le chronometre se mets en pause aussi dans la notification que dans la l’affichage de l’application.

Par contre dit comme ça, en effet on dirait ca aurait pris 1h pour le faire mais c’est l’une des fonctionalite obligatoire qui m’a pris le plus de temps car j’ai un plusieurs problemes, Tout d’aboir il fallait afficher le chronomètre dans la notification, celui que j’avais mi dans le textView en grand, et il fallait reussir a reprendre la valeur qui était dans ce textView la, et la reprendre et l’envoyer dans une notification, et une fois que cela a fonctionner, les problemes arrivait.

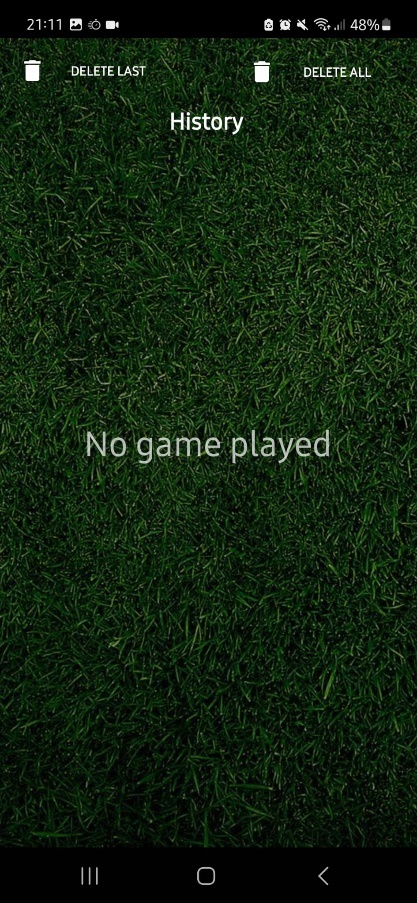
C'était très compliqué et cela m'a pris beaucoup de temps. En effet, lorsque le téléphone reçoit une notification, c'est simple, mais pour que la notification change à priori chaque seconde, il fallait envoyer une notification chaque seconde. Cependant, cela générait un grand nombre de notifications non-stop. Pour bien contextualiser, ça veut dire que j’envoyais une notification des que le chronometre changait donc vu que il changait toute les secondes alors chaque secondes une notification était envoyer, ce qui peux deranger enormement l’utilisateur

Finalement, j'ai réussi à résoudre ce problème avec une méthode appelée "onTick". Cette méthode était appelée à chaque fois que le minuteur était décrémenté, ce qui permettait de mettre à jour l'affichage de la notification sans la renvoyer sur le telephone ni faire vibrer ou sonner le téléphone à chaque seconde, la methode fonctionne de facon a seulement changer le text qui est dans la notification, et donc a chaque seconde la notification allait afficher le temps restant, mais ce qu’il fallait surtout bien faire, c’est que il fallait pas recuperer le TextView avec le temps chaque secondes car c’est ce que je faisait et donc des que y avait un changement dans le textview avec un ecouteur de temps, alors je renvoyais une notification. J’ai donc dû changer cela afin que la notification elle-même recupere ces valeurs afin que ce soit la notification qui gere la decrementation du temp et qui est ensuite similaire pour le TextView pour afficher le temps restant dans le Live.

J’ai aussi eu quelque souci avec la pause du chronometre, car j’avais essayer d’implementer une pause au cas ou le match devrait se mettre en pause, pour faire cela j’ai du faire plusieurs calculs, car j’avais un probleme du style que quand j’appuiais sur pause ca terminaint le chronometre et ca affichait la valeur du dernier chrono et si je redemarais alors ca redemarait le chronometre completement, ou alors j’avais des problemes du style que j’avais plusieurs chronometre en meme temps je parle dans la section des notification de la pause car le meme probleme de l’affichage de la page était dans la notification.Finalement j’ai compris qu’il fallait stocker la derniere valeur du chronometre et tout simplement reprendre a partir de cette valeur le chronometre lorsque on le redemarait.

**Historique de match**

Une image contenant texte

Description générée automatiquement  

Dans mon applications j’ai un system d’historique des matches crée, celui-ci sert a avoir un historique de tout les matches qui ont été enregistrer prealablement dans la creation de match, la page va tout simplement afficher pour chaque match enregistrer le numero de la partie avec le mail de la personne qui a créer le match, la date du matches, les joueurs des 2 equipes, le score final du match et le temps de jeux pour chaque mi temps et pause. Cela va s’afficher du plus recent au plus ancien dans une scrollView. Il y a notament 2 bouton, celui a gauche qui permet de suprimer seulement le dernier match jouer qui sera le plus recent et le bouton de droite qui permetera de suprimer tout les matches de l’historique, il faut savoir que pendant que la page charge, car elle recuperer des donnes de la base de donner il y a un textView qui affiche : « loading… » et si il n’ya aucun match il affichera ensuite « No game played » ou allors les matches existant.

Pour arriver a ce resultat j’ai du passer par plusieurs facon de faire, voici notament un ancien screen avec les attribut de chaque matche diviser en plusieurs texr view, cetait aberant car il fallait alle regarder dans chaque textView pour chaque match le score, le temps,etc.. du match en question :

Avant : Après :

Une image contenant texte

Description générée automatiquement Une image contenant texte

Description générée automatiquement

Avant toute chose il faut savoir que pour recupere les donner a afficher dans l’historique c’est fait dans le live du match en cours, lorsque je vais sauvegarder un match en cours il va ajouter dans la base de donner dans le nœud « Game » :



Je recupere donc toute les valeurs du match et je les ajoute dans un nouveau nœud incrementer, voici comment se retrouve les attribut de chaque match :

Une image contenant texte

Description générée automatiquement

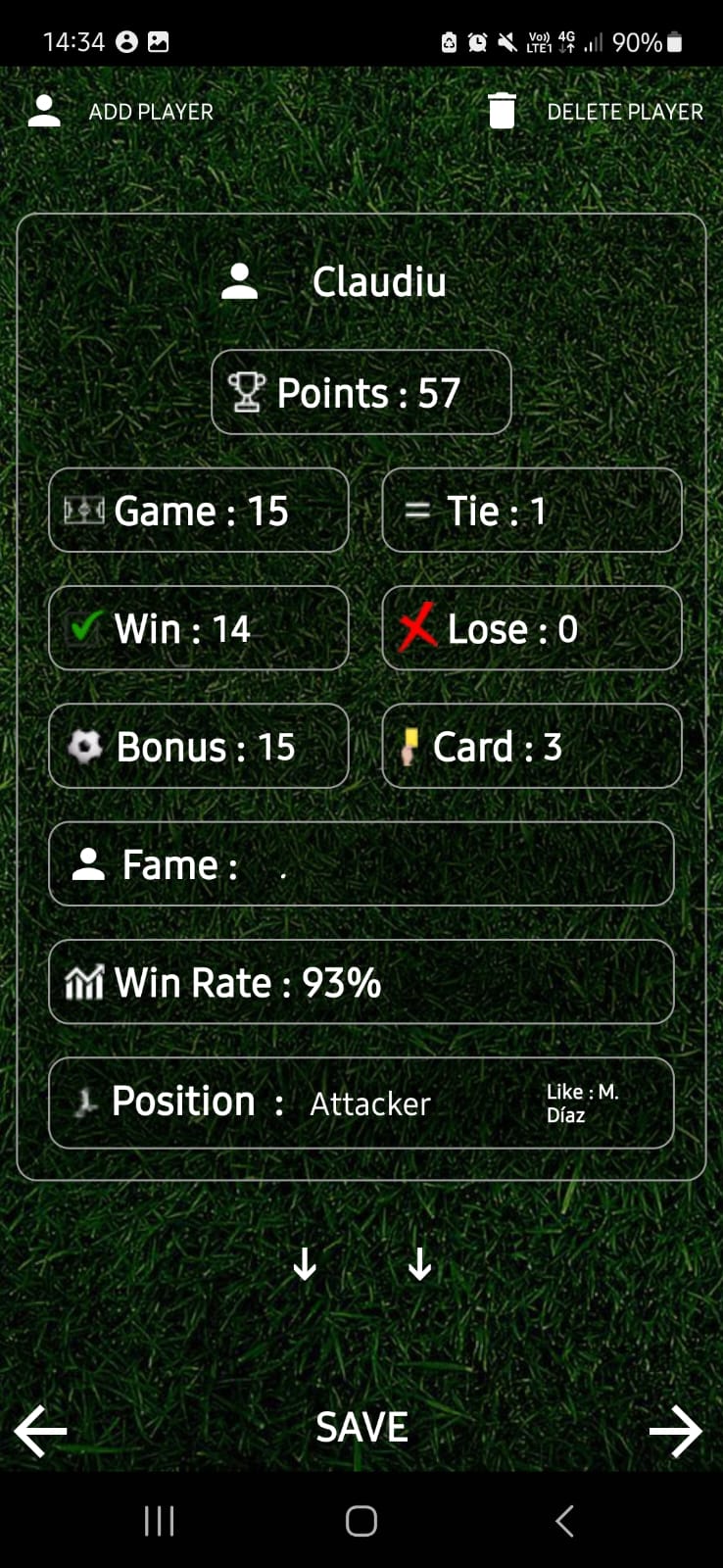
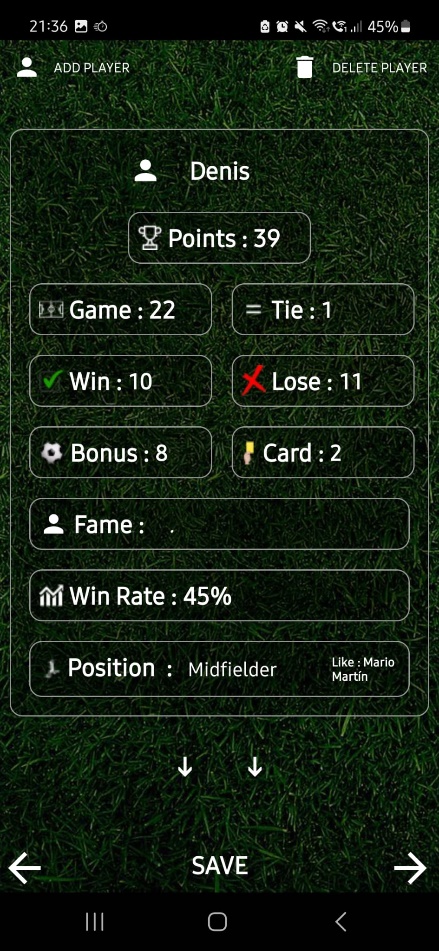
Pour continuer ce que je disait avant dans l’ancienne version (gauche) chaque cadre visible est en fait un TextView, et chaque TextView affiche une valeur récupérée de la base de données. Par exemple, pour l'enregistrement d'un match, les scores de l'équipe 1 et de l'équipe 2 sont stockés dans Firebase. Dans mon code, j'accède à la base de données pour récupérer ces données, puis je les affiche dans le TextView "Scores". C'est la même chose pour chacun des cadres affichés ci-dessus.

Au début, j'ai rencontré une difficulté avec l'affichage de l'heure et du score dans un ordre différent, ce qui signifiait que la partie 1 ne correspondait pas aux données affichées dans l'historique. J'ai pu résoudre ce problème en utilisant l'objet TreeMap, qui me permet de trier ma Map, qui parcourt les nœuds de ma base de données.

Et meme si j’ai reussi a regler ces soucis sous cette forme la d’affichage j’ai changé l’affichage en l’image de droite j’ai modifer cela pour arriver a recuprer seulement en une seul fois de facon synchrone chaque attribut de ma base de données grace au SnapShot, c’était une partie vraiment pas compliqué en terme de codage, je l’ai fait la premiere fois pour l’historique ensuite pour le reste de mon application c’était tres facil pour moi de faire des choses pareil meme si je recuperais et enregistrait d’autre choses. En tout cas c’était tres satisfaisant a voir que ca fonctionne.

Ce qu’il en est des boutons, c’est que pour celui de tout suprimer, c’est simple, il recupere seulement le nœud « Game » de ma base de donnée et suprimer toutes les valeurs qui y sont dedans. Pour suprimer le dernier match, c’est plus compliquer car il fallait alle verifier quel est le match ajouter en dernier, donc il fallait regarder le numero de la game et en fonction de cela si c’est la derniere, tout simplement la suprimer, donc il y avait certaine condititon a faire en plus pour realiser cette fonctionalité.

**Statistiques des joueurs**

  Une image contenant diagramme

Description générée automatiquement

<https://developers.google.com/maps/documentation/android-sdk/start?hl=fr#:~:text=Open%20Android%20Studio%2C%20and%20click,Language%20to%20Java%20or%20Kotlin>

<https://console.cloud.google.com/welcome?project=tchoo-tchoo-tracher&hl=fr>

**UI/UX**

Cette section aborde en détail l'importance accordée à l'UI/UX, ainsi que la manière dont les besoins et préférences des utilisateurs ont été pris en compte, notamment lors de la séance de Speed Testing.

Avant d'aborder les retours du Speed Testing, j'avais déjà effectué des tests d'expérience de mon application auprès de certaines personnes de mon entourage. On m'a signalé quelques bugs d'affichage dans mon application, notamment dans le TextView du chronomètre qui n'affichait pas les unités de secondes sur les téléphones moins larges à mon avis. De plus, certaines images étaient décalées sur d'autres types de téléphones. J'ai donc apporté les modifications nécessaires et j'ai renvoyé l'application pour être testée à nouveau.

J'ai également rencontré un problème avec une personne qui ne recevait pas son mail de confirmation. Initialement, j'envoyais un mail de confirmation lors de l'inscription pour confirmer l'identité humaine de l'utilisateur et lui permettre de se connecter. La personne devait vérifier sa boîte mail et cliquer sur le lien reçu pour confirmer son compte. Cependant, une personne ne recevait tout simplement pas ce mail de confirmation, bien que son adresse mail soit correcte. Pour éviter ce type de problème à l'avenir, j'ai décidé de supprimer cette fonctionnalité, car elle n'était pas particulièrement utile dans mon application. À part ce problème, tout fonctionnait bien.

Lors d'une de nos itérations précédentes, vous m'avez fait remarquer que les fonds d'écran de l'application n'étaient pas adaptés, ce qui rendait le texte peu visible. J'ai donc pris en compte votre feedback et j'ai modifié presque tous les fonds d'écran de l'application, comme vous pouvez le voir dans les captures d'écran figurant dans le rapport.

Voici un résumé des retours reçus des testeurs via les formulaires Google lors de l'activité Speed Testing, ainsi que la manière dont ces retours ont été pris en compte dans l'implémentation de ma version finale de l'application :

* Plusieurs testeurs ont suggéré qu'il serait utile de savoir quand on clique sur un joueur de la liste des joueurs lors de l'ajout de joueurs aux équipes de football. J'ai donc modifié cela pour que, dès qu'un joueur est cliqué, il disparaisse de la liste afin qu'il ne soit plus cliquable.
* Lors de l'ajout du temps, j'ai limité le temps maximum à deux chiffres, c'est-à-dire qu'il n'y aura pas plus de 99 minutes pour une période de jeu.
* Les testeurs ne savaient pas où cliquer pour démarrer le match en direct, alors j'ai agrandi le bouton et ajouté un grand texte pour bien comprendre cela.
* Il a également été suggéré qu'il serait préférable de ne pas pouvoir commencer un match si les équipes ne sont pas encore déterminées pour le mode RANDOM. J'ai donc fait en sorte que le match ne commence pour la partie RANDOM que si l'on clique sur le grand bouton PLAY.
* L'un des problèmes rencontrés était que l'on pouvait quitter le live du match sans sauvegarder, ce qui supprimait complètement les données du live. Pour résoudre ce problème, j'ai ajouté un avertissement à l'utilisateur avant de quitter la page live, indiquant que s'il continue, cela risque de supprimer la session. S'il décide d'annuler, il revient à la page et peut sauvegarder le match avant de le quitter.
* Selon un testeur, il y avait trop de pop-up dans la partie météo, alors j'en ai enlevé.
* Un autre testeur a suggéré qu'il serait préférable d'indiquer les fonctionnalités de l'application et comment cela fonctionne avec des textes au-dessus des champs à compléter pour indiquer comment ils doivent être remplis et comment l'application fonctionne.

# **Analyse**

Une image contenant diagramme

Description générée automatiquement

**Explication du diagramme :**

Comme nous pouvons le constater, la première page que nous verrons sera en rouge, ce sera la page d'accueil qui permettra d'accéder à toutes les fonctionnalités. À partir de là, les utilisateurs pourront accéder au menu déroulant qui leur permettra de se connecter ou de s'inscrire, ou tout simplement utiliser les fonctionnalités de l'application, telles que la map, les pages de composition avec des schémas tels que 4-4-2 ou 4-3-3, ainsi que toutes les pages d'information comme "About me", "Rules" ou "Help".

Les fonctionnalités principales, telles que la création de matches, sont disponibles dans la section "Home". À partir de là, nous aurons le choix de l'équipe, et en fonction de ce choix, nous arriverons dans la diffusion en direct du match. À partir de là, nous pourrons accéder directement à l'historique de tous les matches, ou cela sera également possible depuis la page d'accueil. Il y aura également la possibilité de consulter les statistiques, où les performances de tous les joueurs de l'application seront affichées.

En résumé, la page d'accueil rouge est la porte d'entrée de toutes les fonctionnalités de l'application. Elle permettra aux utilisateurs de se connecter, de s'inscrire et d'utiliser les différentes fonctionnalités telles que la carte, les pages de composition et les informations générales. De plus, la section " Home " offrira la possibilité de créer des matches, de suivre leurs matchs en direct et de consulter l'historique des matches. Les statistiques des joueurs seront également disponibles pour une analyse approfondie.

# **Limitations et développement futur**

Voici quelques questions auxquelles je vais essayer de répondre au maximum pour mon application :

1. Dans quels cas d'utilisation mon application pourrait ne pas fonctionner ?

En ce qui concerne les crashs, normalement l'application ne devrait pas crasher. Cependant, il y a quelques points qui pourraient être extrêmement gênants, par exemple :

* L'enregistrement des matches dans l'historique. Dans mon cas, il n'y aura probablement pas plus de 50 matches par saison, mais si l'on imagine seulement 1000 matches, il deviendrait extrêmement fastidieux de faire défiler 1000 matches dans l'historique. Il aurait été préférable d'ajouter une barre de recherche par équipe, score ou buts.
* Si l'on imagine 1000 matches dans l'historique, il y aurait également un problème lorsqu'on souhaite supprimer un match en particulier, par exemple le 750ème. Étant donné que je ne peux supprimer que le dernier match ou tous les matches, il serait impossible de supprimer uniquement le 750ème match.
* De même pour les statistiques, j'ai seulement 12 joueurs dans mes statistiques. Je peux en ajouter autant que je le souhaite, mais imaginons que j'en aie 100. Il serait alors nécessaire de chercher parmi tous les joueurs pour en retrouver un, il faudrait donc également une barre de recherche par prénom. Si nous avons 10 000 joueurs, le classement de 1 à 10 000 serait très long, il faudrait donc le diviser en fonction des joueurs.
* Dans mes statistiques, je peux ajouter un joueur avec un nom de 1000 caractères, et l'application l'acceptera probablement, mais cela causerait des problèmes d'affichage dans le choix des joueurs avec un spinner énorme.
* Des problèmes similaires peuvent se produire pour tous les champs de mes statistiques, par exemple, si je mets plus de 3 chiffres dans les cases avec des chiffres tels que les cartons jaunes, seuls les 3 premiers chiffres seront affichés, ce qui donnera des valeurs incorrectes.
* Pour l'attribut "fame", si vous écrivez 10 000 caractères, la case s'agrandira et vous ne verrez plus ce qui se trouve en dessous, vous devrez donc faire défiler énormément.
* Les points réagissent en fonction des champs des statistiques, j'ai également remarqué que cela se produit pour "fame", la case des points s'agrandit si de grandes valeurs sont ajoutées.
* Mes compositions sont limitées aux plus connues, le 4-4-2 et le 4-3-3, mais si l'on souhaite une autre formation, il est impossible d'en créer une autre sans modifier le code.
* Pour la météo, je ne peux pas choisir certaines villes qui ne sont pas proposées par l'API, et si je fais une faute d'orthographe, cela ne fonctionne pas du tout.
* Mon API est limitée à 100 requêtes par jour, car seulement 100 sont permises gratuitement. Par conséquent, elle ne fonctionnera que 100 fois par jour. Si 100 utilisateurs consultent les statistiques une seule fois par jour, l'API sera déjà épuisée. Il faudrait donc payer chaque mois pour avoir droit à plus de requêtes.
* En ce qui concerne les codes admin, si une personne pirate mon application et trouve les codes admin, elle pourrait supprimer toute ma base de données et faire ce qu'elle souhaite avec les données de mon application.
* Lors de la création d'un match, si l'on ajoute un joueur manuellement, il peut entrer autant de caractères qu'il le souhaite. Par exemple, si on en met 5000, cela détruirait complètement l'affichage de mes équipes dans le LIVE et cela ne ressemblerait plus à rien. J'ai limité le temps d'ajout à la mi-temps et à la pause à 2 chiffres, mais il reste tout de même absurde d'avoir un match avec 99 minutes par mi-temps et 99 minutes de pause.
* Il peut également y avoir un gros problème de récupération des données si l'on n'a pas de connexion Wi-Fi ou 4G. Dans ce cas, on ne peut presque pas utiliser l'application, car tout fonctionne en Wi-Fi, que ce soit l'enregistrement des matches, l'affichage de l'historique, les statistiques des joueurs, les classements, etc.

2) Y a-t-il des aspects techniques non traités ?

Non, toutes les fonctionnalités demandées ont été réalisées.

3) Si j'avais plus de temps pour le projet, qu'aurais-je amélioré ?

L'un des points les plus importants que j'aurais améliorés aurait été la possibilité de personnaliser l'application pour un usage autre que le mien. Ainsi, n'importe qui aurait pu créer une saison, et chaque saison aurait eu son code d'accès avec ses participants, ses matches, son historique, sa carte et ses statistiques, et tout ce qui va avec. Cela aurait permis d'avoir une véritable application utilisable par n'importe qui qui aime le football.

# **Play Store**

Avant d'aborder le sujet principal, je tiens à mentionner que j'ai essayé de mettre mon application sur le Google Play Store. J'ai consacré du temps à suivre un tutoriel sur YouTube :

https://www.youtube.com/watch?v=7h6aouRXpWM&t=947s.

Tout se déroulait bien lors de mes tests pour mettre mon application sur la plateforme, jusqu'à ce que j'arrive à l'étape d'upload de mes bundles et APK. Je les avais correctement préparés, mais j'ai rencontré un problème lors de l'ajout de mon application. Une erreur est apparue, indiquant que je ne pouvais pas importer l'application car mon package était "com.example". Selon ChatGpt, voici pourquoi cela n'est pas possible :

*Le Google Play Store a des règles strictes concernant les packages d'application, dans un souci de sécurité et pour éviter toute confusion ou conflit entre les applications. Le package d'application, généralement au format "com.example", sert à identifier de manière unique chaque application sur le Play Store. L'utilisation du package "com.example" est généralement réservée aux exemples de code ou aux démonstrations fournies par les développeurs pour aider les autres à comprendre comment créer des applications. Cependant, il n'est pas autorisé pour les applications réelles destinées à être publiées sur le Play Store.*

*Lorsque vous soumettez une application au Play Store, vous devez fournir un package d'application unique qui n'est pas déjà utilisé par une autre application existante. Cela garantit que chaque application a son identifiant unique et peut être téléchargée et mise à jour indépendamment. En utilisant un package d'application générique comme "com.example", il y aurait un risque de conflit avec d'autres applications qui pourraient utiliser le même package. Cela pourrait entraîner des problèmes de sécurité, des erreurs de mise à jour ou d'autres complications indésirables. Il est donc important de choisir un package d'application unique et spécifique pour votre application afin de pouvoir la publier en toute sécurité sur le Play Store et éviter tout problème potentiel avec d'autres applications existantes.*

Ainsi, l'idée est simplement de modifier le nom de mon package. Cependant, cela s'est avéré plus compliqué que prévu. Lorsque j'ai essayé de le faire, cela a provoqué de nombreux problèmes, l’application ne voulais plus run du tout. J'avais push trop rapidement la dernière version sur GitHub, ce qui a également affecté le fonctionnement de GitHub. J'ai passé une bonne partie d'une nuit à résoudre ce problème en testant différentes méthodes, et j'ai finalement réussi à le résoudre sans même savoir exactement comment. En tout cas, je ne changerai plus jamais le nom de mon package s'il y a autant de risques dans un projet d'envergure. Le package a en réalité de nombreuses dépendances qui ne permettaient pas de simplement modifier son nom.

J'ai donc abandonné l'idée de publier mon application sur le Google Play Store. J'aurais au moins essayé.

# **Conclusion**

Je commence ma conclusion en exprimant ma fierté pour ce projet. Je pense que c'est la création la plus importante que j'ai réalisée en informatique. J'ai attendu longtemps le cours sur le développement mobile, depuis mes débuts en informatique.

Ce qui me passionnait le plus, c'était de réaliser l'application que j'ai crée, et j'ai pu le faire grâce à vous. En effet, il est parfois difficile de découvrir seul une nouvelle plateforme comme Android Studio, c'est pourquoi votre cours m'a été d'une grande aide.

Au cours de ce projet, j'ai pu apprendre énormément de choses, notamment en ce qui concerne le développement mobile. J'ai également approfondi mes connaissances du langage Java, notamment toutes les liaisons possibles avec le XML et tout ce qui concerne l'affichage. Je me rends compte que j'apprécie davantage le front-end que le back-end, et ce projet m'a permis de progresser considérablement dans ce domaine. J'ai découvert de nombreuses astuces qui m'ont permis d'améliorer l'interface graphique au fil des jours.

J'attends avec impatience de suivre le cours de Mobile 2 afin de poursuivre le développement mobile et d'apprendre encore plus de choses. Si possible, j'aimerais utiliser une plateforme comme React pour créer des applications pour Apple, car d'après mes recherches, c'est ce qui est recommandé.

Je tiens à vous remercier sincèrement d'avoir accepté ce sujet et de m'avoir supervisé tout au long de ce projet. Je ne dis pas cela pour vous impressionner, car je sais que j'ai réalisé toutes les fonctionnalités et, sans arrogance, je suis convaincu que je vais réussir. Cependant, je tiens à souligner que vous m'avez énormément aidé dans ce projet, et sans votre soutien, je n'aurais jamais pu aller aussi loin. Les itérations ont été d'une grande aide, car vous me guidiez et me donniez des conseils pour améliorer mon application, en particulier lors des sessions de test de vitesse, qui m'ont permis de recueillir plusieurs retours constructifs.

Je termine ce rapport en vous donnant un conseil :

Continuez à vous intéresser aux élèves comme vous l'avez fait jusqu'à présent !