Projet Développement

Mobile II 2023

Une image contenant Graphique, cercle, art, symbole

Description générée automatiquement

Réalisé par Popadiuc Claudiu

Supervisé par Monsieur Riggio, Jonathan

**Table des matières**

[1) Introduction 3](#_Toc150112334)

[2) Description des technologie utilisées 4](#_Toc150112335)

[3) Sujet d’application 5](#_Toc150112336)

[4) Fonctionnalités 6](#_Toc150112337)

[**4.1)** **Inscription / Connexion / Fire Base** 6](#_Toc150112338)

[**4.2)** **Accueil & Jeux** 8](#_Toc150112339)

[**4.2.3) Quiz** 9](#_Toc150112340)

[**4.2.3) Hangman** 10](#_Toc150112341)

[Analyse 16](#_Toc150112342)

[Limitations et développement futur 18](#_Toc150112343)

[Conclusion 20](#_Toc150112344)

# **Introduction**

Dans le cadre du cours de Mobile II, il nous a été demandé de réaliser une application mobile qui sera développée avec React Native en utilisant le Java Script. Pour ma part, j'ai réalisé une application de gestion de plusieurs mini jeux tel que Flappy Bird, Morpion, Hangman et un quizz.

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.

# **2) Description des technologie utilisées**

Pour la réalisation de ce projet, j'ai opté pour une approche innovante en utilisant exclusivement React Native comme environnement de développement. React Native est une technologie de pointe qui a révolutionné la création d'applications mobiles multiplateformes. Cet outil m'a permis de combiner l'efficacité du développement multiplateforme avec des performances dignes d'applications natives.

L'un des aspects les plus remarquables de ce projet réside dans l'utilisation du langage JavaScript. JavaScript est un langage de programmation extrêmement polyvalent et populaire, qui s'est imposé comme le choix prédominant pour la création d'une vaste gamme d'applications, que ce soit pour les applications de bureau, les applications mobiles, ou les applications web. Cette polyvalence m'a offert une grande flexibilité et a facilité l'intégration de fonctionnalités complexes au sein de mon application.

# **3) Sujet d’application**

L'application que je développe est une plateforme de divertissement proposant une variété de mini-jeux passionnants tels que Flappy Bird, Morpion, Hangman et un quiz. Chacun de ces jeux sera hébergé au sein de l'application.

Les utilisateurs auront la possibilité de jouer à chacun des quatre jeux. Le jeu du Morpion se jouera en mode deux joueurs, tandis que les autres jeux seront des défis de score, enregistrant la performance de chaque utilisateur dans son propre profil.

Chaque utilisateur disposera d'un profil personnel, où ils pourront personnaliser leur nom d'utilisateur à tout moment. De plus, leur profil affichera leurs scores pour les trois autres jeux, permettant ainsi un suivi de leur progression.

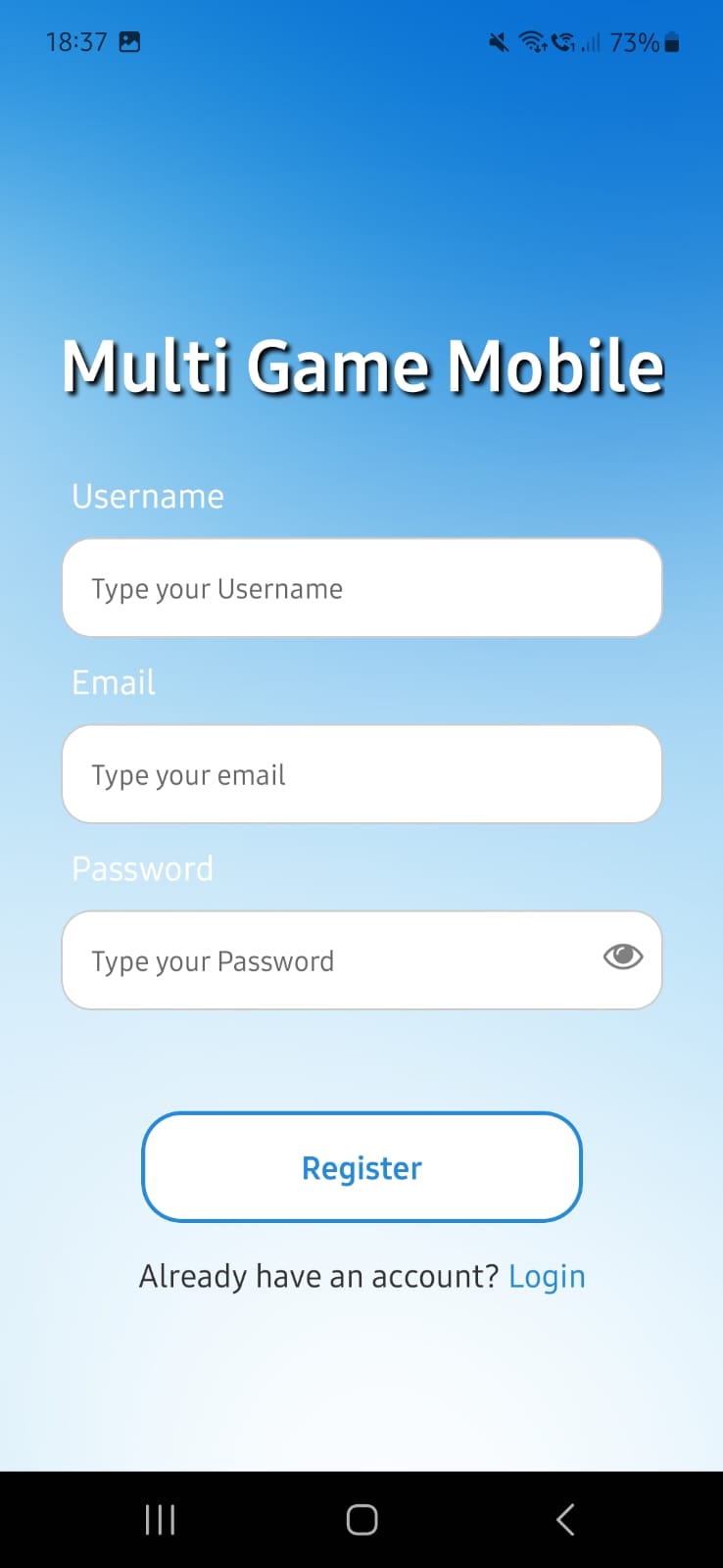
L'application comprendra une section d'informations détaillant les règles et mécanismes de chaque jeu, facilitant ainsi la prise en main pour les joueurs.

Une page de paramètres sera également disponible, permettant aux utilisateurs de contrôler la musique de fond et d'envoyer des commentaires et suggestions directement aux développeurs, favorisant ainsi une expérience utilisateur améliorée.

Enfin, l'application inclura une page de classements pour chaque jeu, affichant les scores et les classements de tous les joueurs inscrits sur la plateforme. Cela encouragera la compétition saine et la motivation pour les joueurs afin d'améliorer leurs performances.

# **4) Fonctionnalités**

## **Inscription / Connexion / Fire Base**

 Une image contenant texte, capture d’écran, Police, Système d’exploitation

Description générée automatiquement

Voici à quoi ressemble visuellement le processus d'inscription et de connexion. Je ne vais pas m'attarder trop longtemps sur cette fonctionnalité, car nous l'avons déjà abordée dans Android Studio. J'ai réussi à la mettre en place en suivant un tutoriel YouTube : <https://www.youtube.com/watch?v=ql4J6SpLXZA>

Ensuite, j'ai personnalisé ces fonctionnalités pour les intégrer à mon application. J'avais besoin d'un nom d'utilisateur pour identifier les joueurs, ainsi que leur pseudo, afin de pouvoir afficher tous les utilisateurs de l'application sur ma page de classement. Ces informations sont enregistrées sur Firebase, où l'authentification gère le courrier électronique et le mot de passe, tandis que le nom d'utilisateur est enregistré dans Firestore sous le champ "profiles". Lors de la création d'un compte les scores élevés des jeux sont initialisés à zéro.

Voici un aperçu de l'authentification : Une image contenant texte, capture d’écran, nombre, document

Description générée automatiquement

Voici un aperçu du Firestore :Une image contenant texte, logiciel, Page web, Site web

Description générée automatiquement

## **Jeux**



Une fois le processus d'inscription ou de connexion est réalisé, nous accédons à la première page, qui constitue la page principale de l'application. Cette page affiche les quatre jeux disponibles : Flappy Bird, le Quiz, le Hangman et le Tic-Tac-Toe. De plus, elle comprend une barre de recherche permettant de trouver un jeu parmi une liste, notamment lorsque celle-ci compterais plus de dix éléments. Comme vous pouvez le constater, une barre de navigation est également disponible pour faciliter la navigation dans l'application qui a été réalisé avec ce tuto : <https://www.youtube.com/watch?v=ePwSuyAcEvs>

Je vais maintenant expliquer un à un les jeux ci-dessus.

### **4.2.1) Quiz**

Une image contenant texte, capture d’écran, multimédia, logiciel

Description générée automatiquement Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Voici à quoi ressemblerait le jeu "Quiz". Au début, nous avons le choix entre trois thèmes prédéfinis, car j'ai créé des fichiers JSON regroupant plusieurs dizaines de questions pour chaque thème. Lorsqu'on sélectionne l'un des thèmes, le jeu mélange les questions en utilisant un tableau contenant toutes les questions. Une fois le mélange terminé, les quatre réponses possibles sont affichées sous forme de boutons pour faciliter la sélection.

Un chronomètre de 30 secondes démarre après chaque question, affiché à l'écran. Lorsque le temps s'écoule, un "useEffect" retire une vie à l'utilisateur. Il y a en tout 3 vies, et on peut les perdre de deux manières : en laissant le chronomètre atteindre 0 ou en choisissant une mauvaise réponse. En revanche, si l'on trouve la bonne réponse, on reçoit un certain nombre de points en fonction du temps écoulé. Par exemple, en répondant instantanément, on obtient 30 points, mais si la réponse arrive après 10 secondes, on n'en reçoit que 20. Ainsi, plus on répond rapidement, plus on gagne de points.

Lorsque l'utilisateur épuise ses 3 vies, le jeu le renvoie à la page précédente avec les choix de quiz et met à jour le meilleur score du joueur s'il l'a dépassé. Ce score est également mis à jour dans le classement global de l'application, que nous explorerons plus tard.

### **4.2.2) Hangman**

Une image contenant texte, horloge, capture d’écran

Description générée automatiquement

Voici à quoi ressemble le jeu du "Bonhomme Pendu". Lorsque vous cliquez sur le jeu, vous arrivez directement sur cette page. Vous pouvez voir un bouton en forme de lampe, qui est une icône "indice", et qui affiche une lettre du mot caché. Ce mot caché est choisi parmi une série de plus de 100 mots liés à l'informatique inclus par défaut dans le code. J'ai placé tous ces mots dans un tableau de chaînes de caractères, et au hasard, je sélectionne l'un de ces mots comme le mot à deviner lors de la partie. En fonction du nombre de lettres dans le mot choisi, nous affichons des espaces " \_ " au milieu de l'écran pour représenter les lettres cachées.

Le joueur a 7 tentatives pour deviner le mot, sinon il risque de perdre 3 points. En cas de victoire, il gagne 5 points, ce qui s'ajoute au meilleur score de l'utilisateur, tout comme dans le jeu "Quiz". Comme vous pouvez le voir, il y a un alphabet affiché, ce sont des boutons ajoutés par défaut, car il n'y a pas d'autres lettres nécessaires. J'ai choisi cette approche pour rendre l'expérience utilisateur plus agréable que d'avoir un clavier complet.

Une fois que le joueur clique sur l'un des 26 boutons de l'alphabet, le programme vérifie si la lettre sélectionnée est présente dans le mot caché. Si c'est le cas, le programme met à jour le mot en remplaçant les espaces " \_ " par la lettre correcte, et le bouton correspondant devient inactif et gris. Si la lettre est incorrecte, le mot reste tel quel, et rien ne se passe en ce qui concerne le mot. Le jeu s'arrête lorsque le joueur décide d'arrêter de jouer, ce qui signifie qu'il peut continuer indéfiniment en gagnant des points par groupes de 5.

### **4.2.3) Tic-Tac-Toe**

**Une image contenant capture d’écran, texte, ligne

Description générée automatiquement**

Voici à quoi ressemble le jeu "Tic-Tac-Toe". Il comporte une zone de jeu sous forme d'un tableau de 16 cases, que j'ai délibérément agrandi par rapport au jeu classique qui se joue généralement sur 9 cases, afin de rendre le jeu plus captivant. Contrairement au Tic-Tac-Toe traditionnel qui se joue en formant des paires de 3, dans cette version étendue, vous devez désormais former des paires de 4. Le jeu se joue à 2 joueurs et ne comporte pas de système de meilleurs scores, car il ne permet pas de gagner des points. Il s'agit du seul jeu de l'application à fonctionner de cette manière, nécessitant la présence de deux joueurs pour y jouer.

Lorsqu'un joueur clique sur l'une des cases, un "X" est placé, et si une autre case est cliquée après celle portant un "X", un "O" est placé. Chaque fois qu'une case est cliquée, une copie du tableau est générée pour refléter les derniers mouvements des joueurs. Cependant, une fois qu'une case a été cliquée, sa valeur ne peut plus être modifiée.

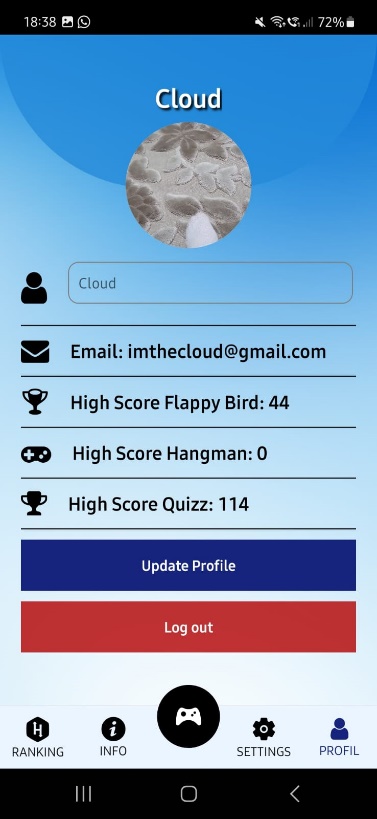
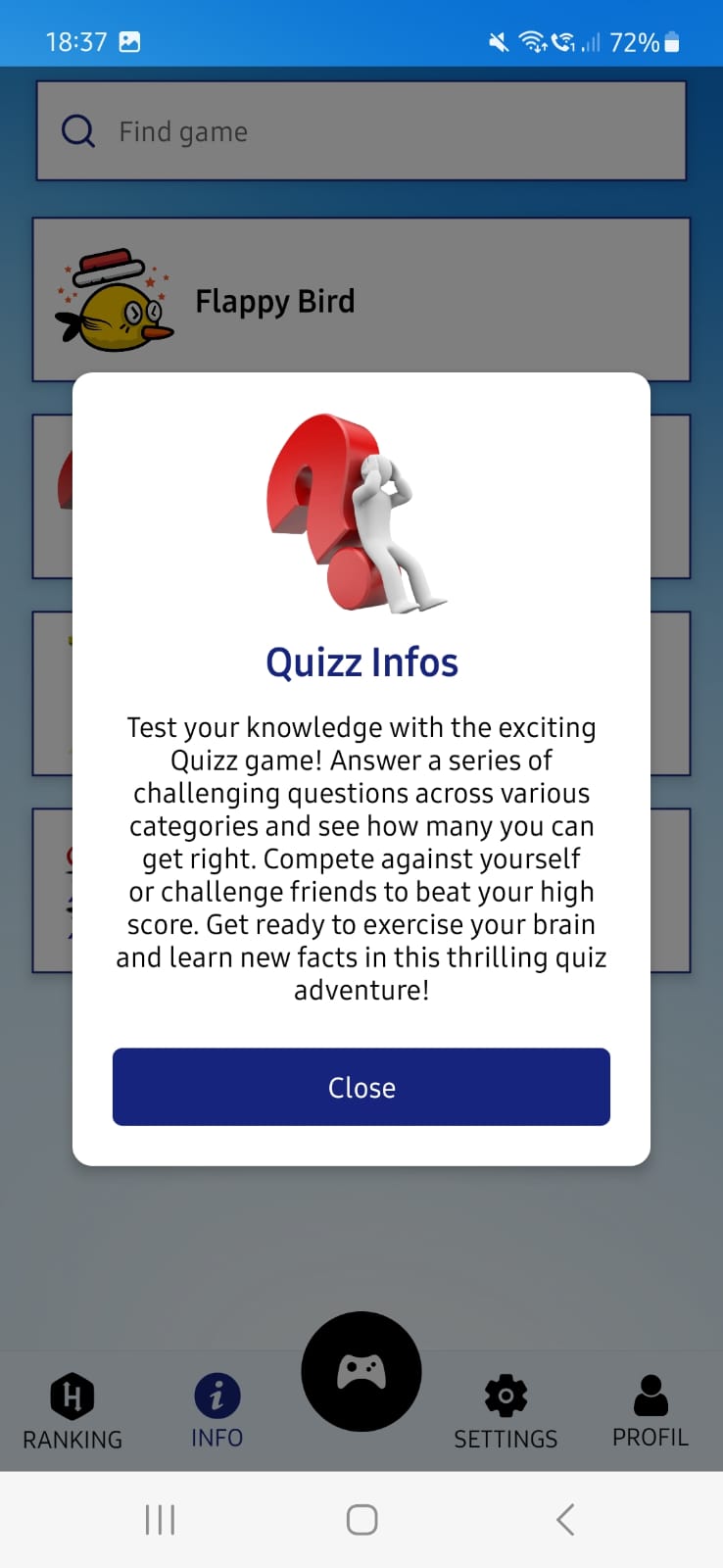
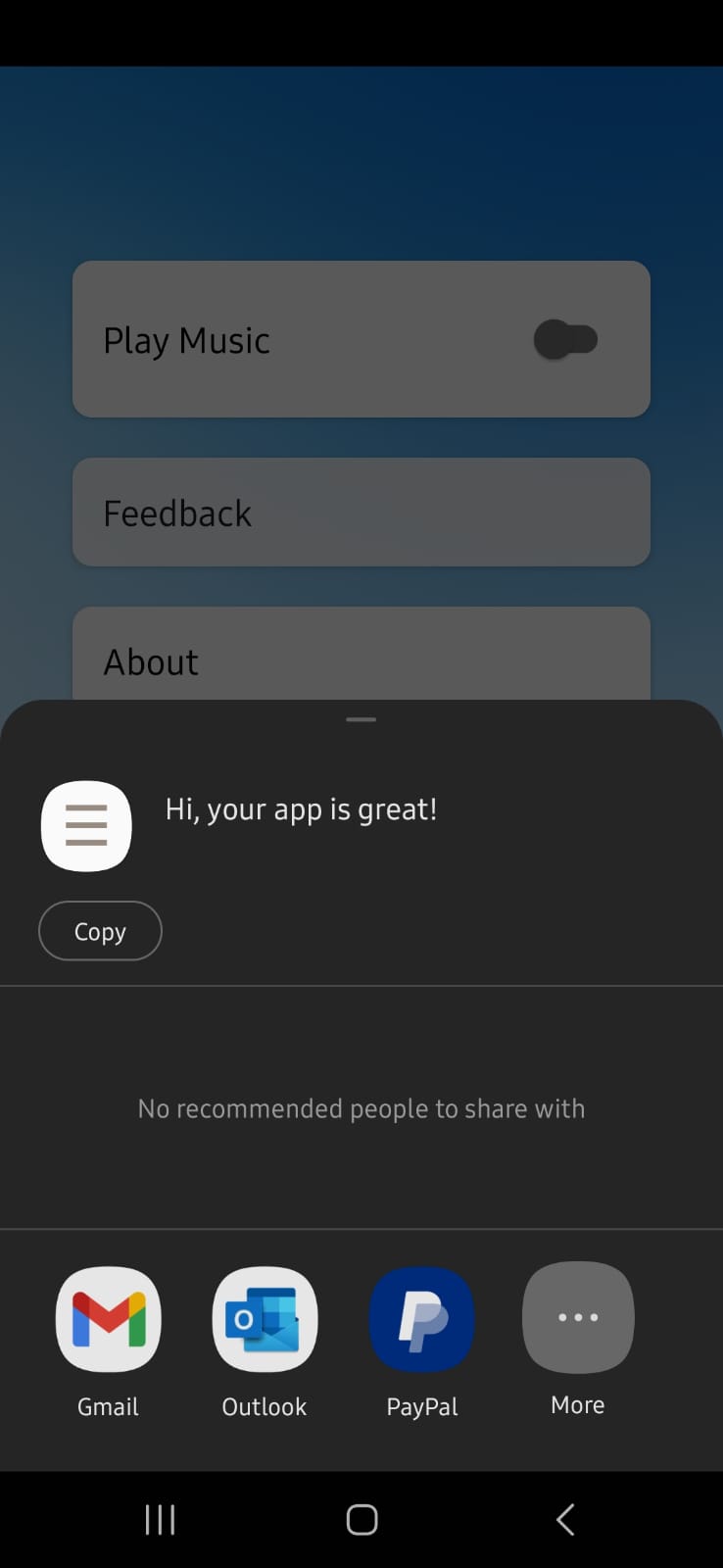
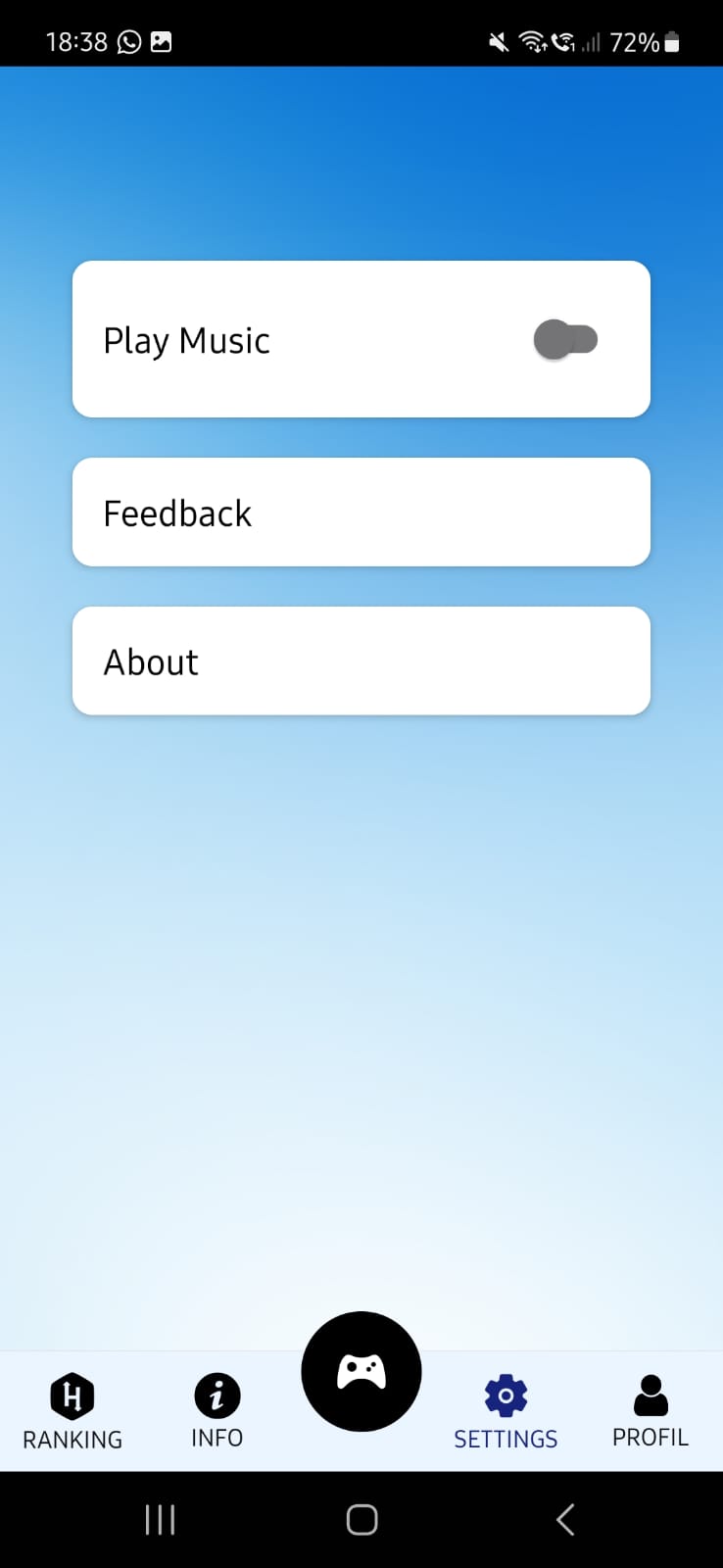
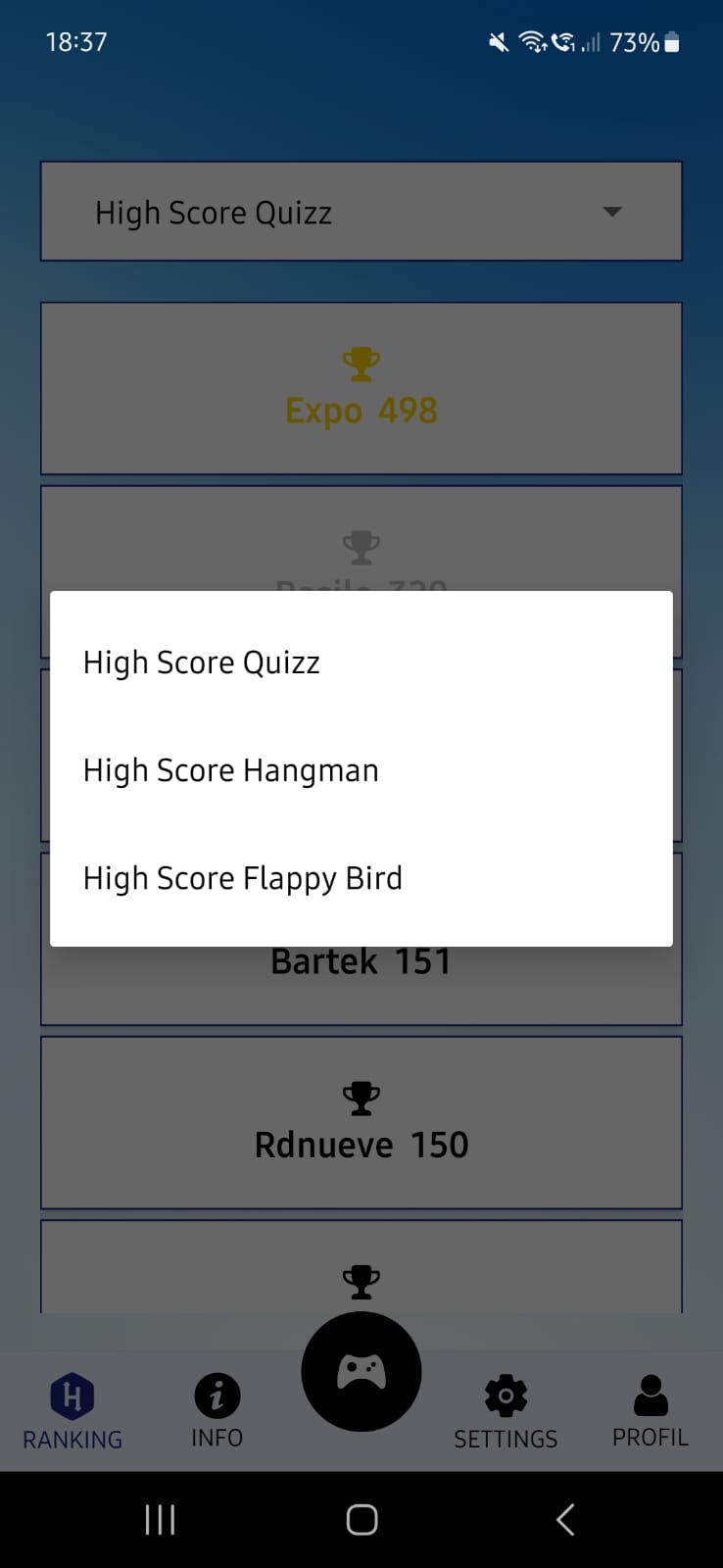
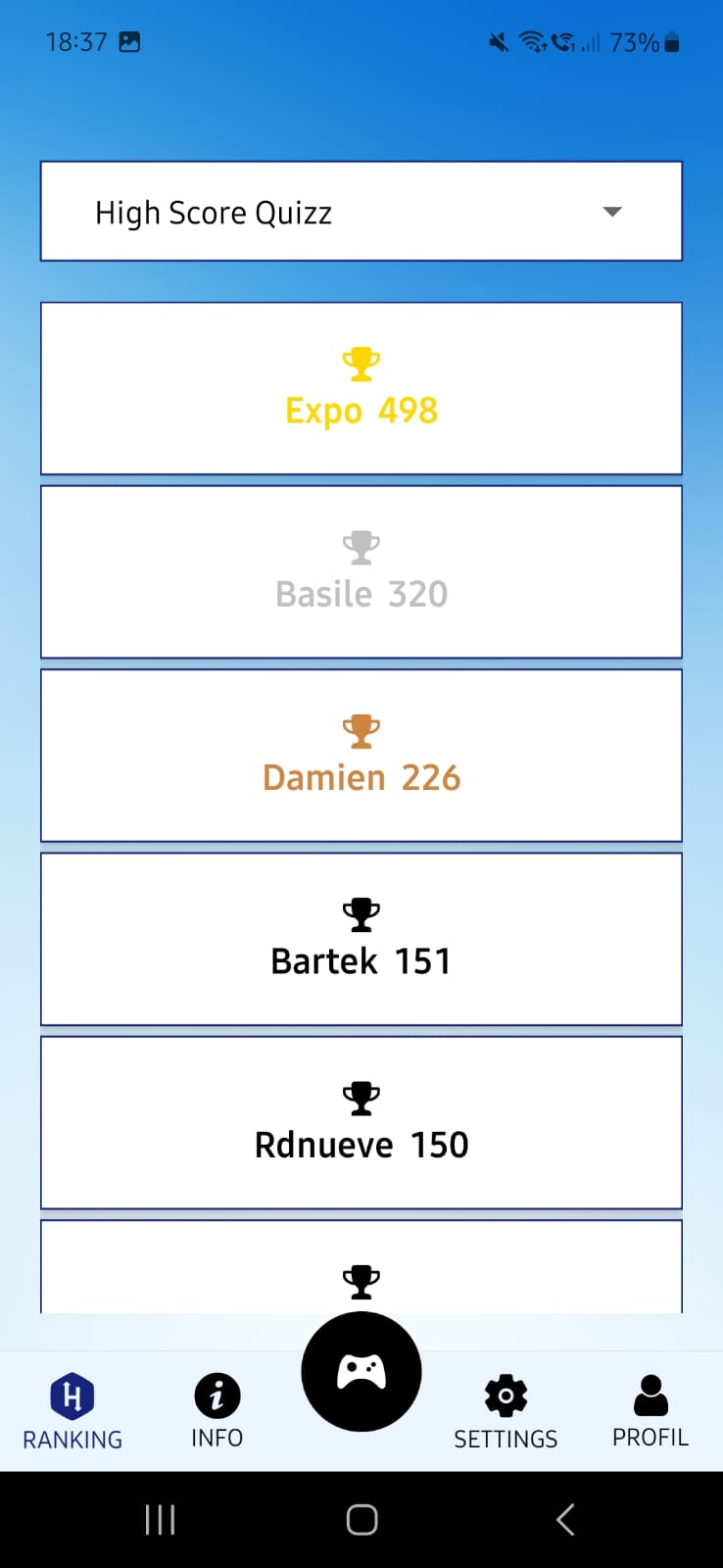
Pour déterminer la victoire, un "useEffect" vérifie si l'un des joueurs a gagné en examinant toutes les possibilités en termes de lignes, de colonnes et de diagonales. Si un joueur remporte la partie, le jeu se termine, et vous avez le choix de recommencer en cliquant sur "Restart" ou de changer de jeu.

### **4.2.4) Flappy Bird**

### 

Voici à quoi ressemble le jeu "Flappy Bird". Pour realsier ce jeux en 2D, il était un peu plus compliquer from scratch j’ai donc du suivre un tuto youtbe pour le réaliser : https://www.youtube.com/watch?v=zK2xYD4Nytw

J'ai également suivi le tutoriel Flappy Bird en adaptant le jeu selon mes besoins pour apporter un effort personnel, améliorant certains aspects fonctionnels tel qu’ajusté la vitesse du jeu au fil de la partie par exemple tous les +5 de score ( je n’ai pas encore réussi j’y travaille), et optimisé les sauts pour rendre le jeu plus viable. En bonus j’ai aussi amélioré certains aspects visuels tel que l'apparence de l'oiseau et des obstacles, ainsi que le fond du jeu.



**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.

# **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 !