

Jeu des 8 dames - Projet de développement sur smartphones

Alexy Lafosse - Yasin Lahrouri - Arthur Gillier

I - Introduction	3
1 - Présentation	3
2 - Outil collaboratif	3
II - Conception	3
1 - Diagramme de classe	3
2 - Diagramme des cas d'utilisation	4
III - Implémentation	6
1 - Stockage des solutions	6
2 - Fonctionnalités d'aides	6
2.1 - Affichage des dames attaquées	6
2.2 - Affichage du pourcentage de victoire	7
2.3 - Placement de dames	8
3 - Fonctionnalités de suppression	8
4 - Pop-up	9
4.1 - Pop-up classique	9
4.2 - Pop-up retirable	10
5 - Fonctionnalités esthétiques	11
6 - Design	12
6.1 - Logo	12
6.2 - Boutons	13
6.3 - Animations	13
6 - Musique	14
IV - Détails techniques	14
1 - Classe Material & Interface CosmeticListener	14
2 - Classe BoutonListener	15
V - Répartition du travail	15
VI - Conclusion	16

I - Introduction

1 - Présentation

L'objectif du jeu des huit dames est de placer huit dames d'un jeu d'échecs sur un échiquier de 8×8 cases sans que les dames ne puissent se menacer mutuellement, conformément aux règles du jeu d'échecs (la couleur des pièces étant ignorée). Par conséquent, deux dames ne devraient jamais partager la même rangée, colonne, ou diagonale. Il existe 12 solutions uniques, pour un total de 92 solutions en tenant compte de transformations telles que des rotations ou des réflexions.

2 - Outil collaboratif

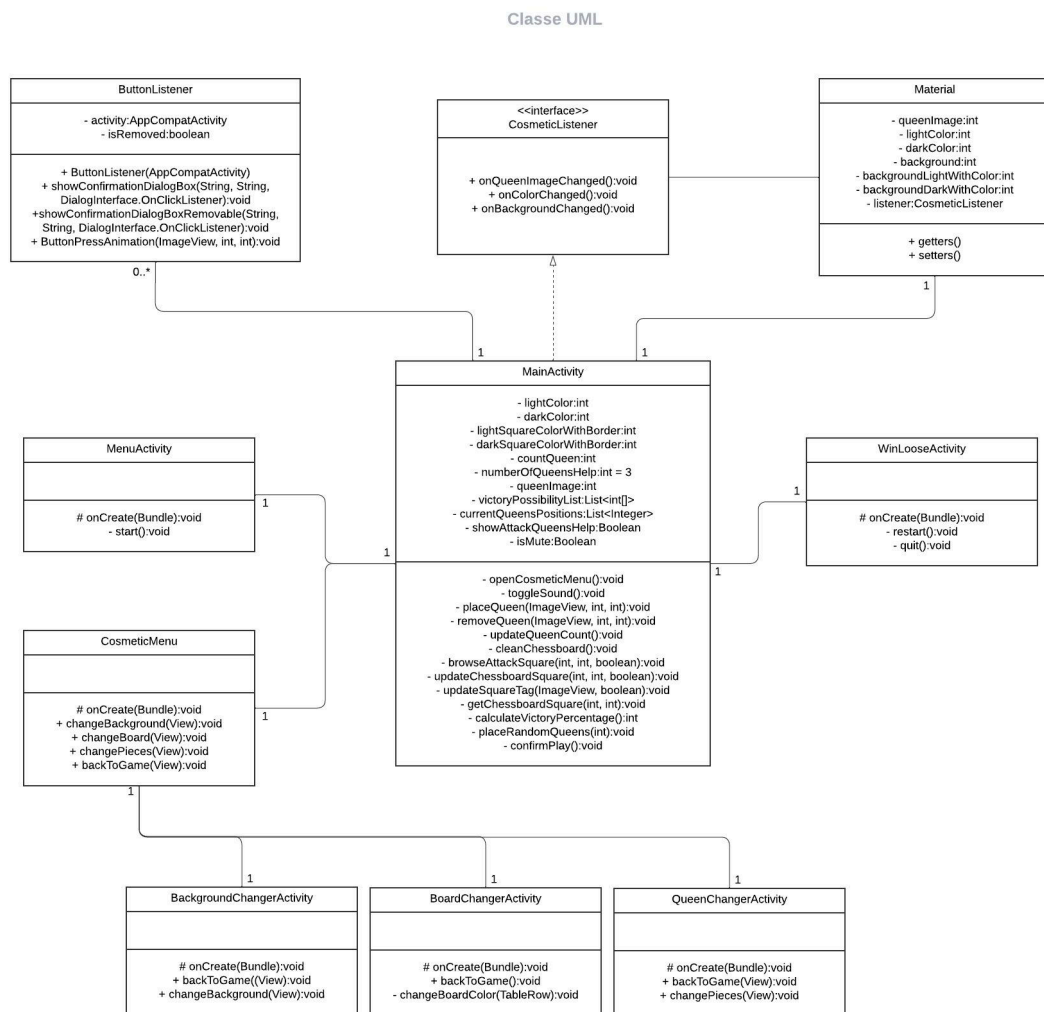
Afin de rester organisés et pour pouvoir travailler de façon propre nous avons opté pour la création d'un répertoire Gitlab sur le Gitlab universitaire, celui-ci nous a permis de travailler indépendamment sur nos parties et d'avancer sur plusieurs fonctionnalités en même temps.

<https://gitlab.univ-lr.fr/alafosse/jeu8dames>

II - Conception

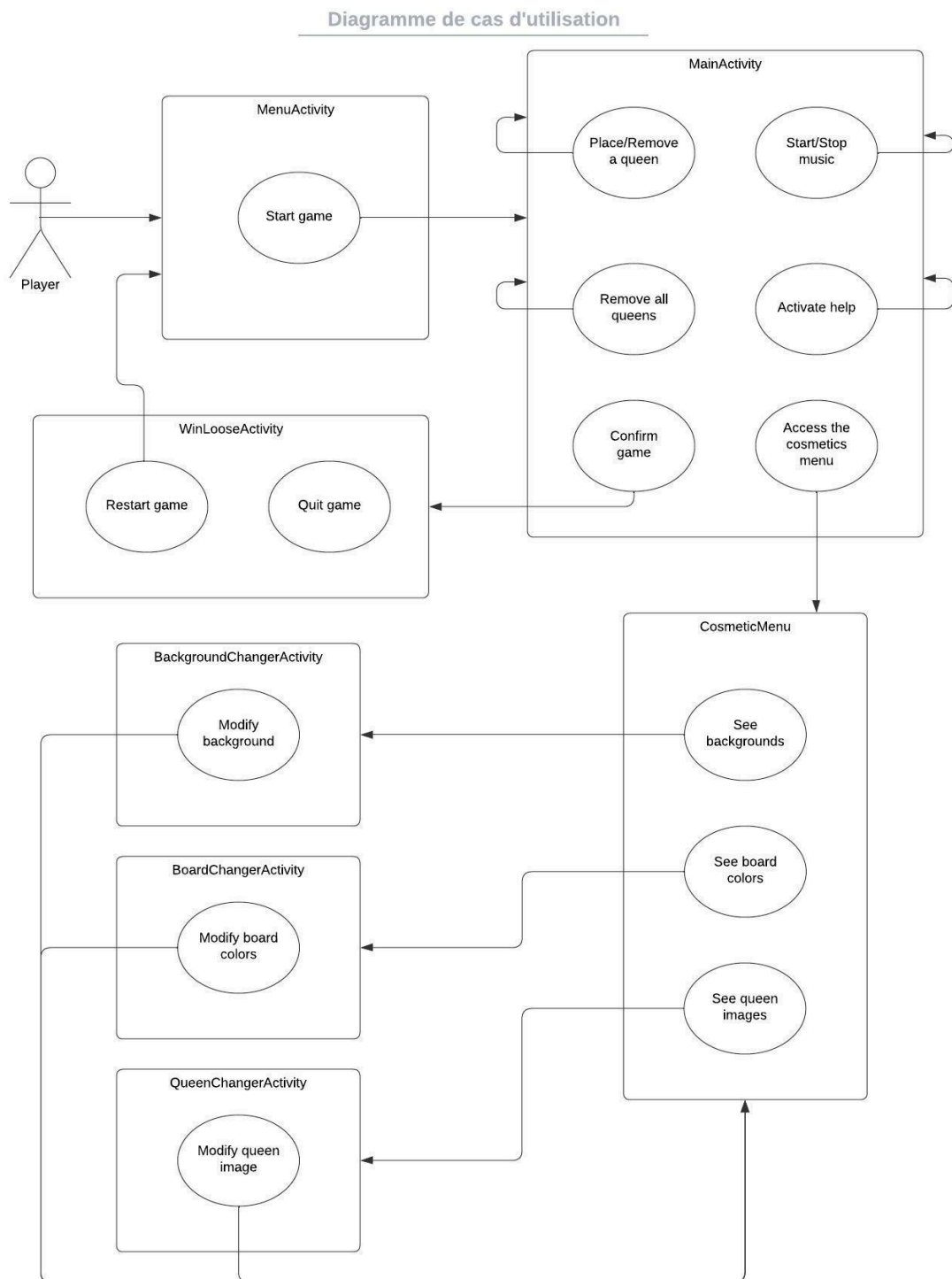
1 - Diagramme de classe

Nous avons conçu un diagramme de classe pour représenter les différentes classes de notre application.



2 - Diagramme des cas d'utilisation

Nous avons également réalisé un diagramme des cas d'utilisation de notre application, il représente les actions réalisables par un joueur.



III - Implémentation

1 - Stockage des solutions

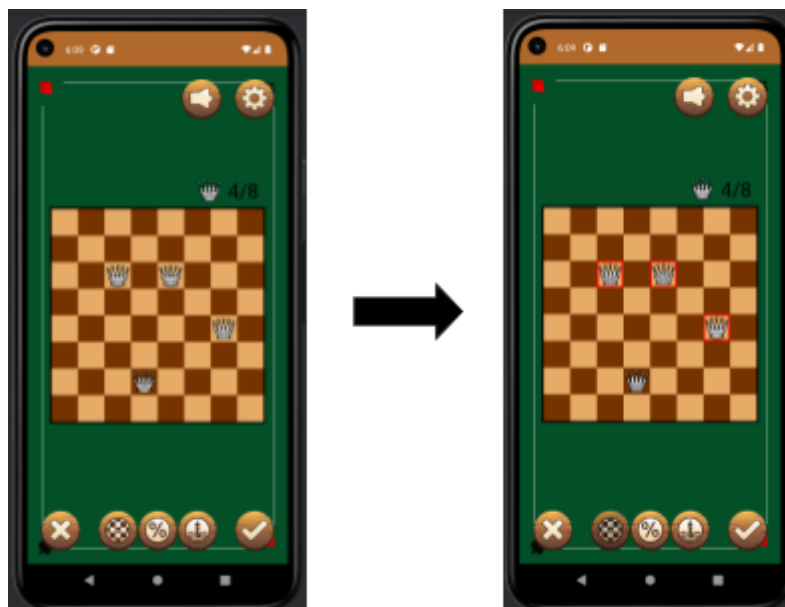
Nous avons décidé de stocker les 92 solutions possibles du jeu dans une liste de tableaux d'entiers de taille 8. L'index d'un tableau représente la colonne de l'échiquier (0 à 7) et les 8 valeurs stockées représentent les lignes. Grâce à cette méthode, on peut stocker une solution avec uniquement un tableau de taille 8. Les 12 solutions uniques ont été enregistrées manuellement et les transformations issus des 12 uniques dans une boucle.

2 - Fonctionnalités d'aides

Nous avons mis en place un système d'aide pour aider à la complétion du jeu.

2.1 - Affichage des dames attaquées

Le premier bouton d'aide consiste à afficher avec une bordure rouge les dames qui s'attaquent mutuellement. Il permet au joueur de savoir dès qu'il fait une erreur, avec cette aide, le joueur ne validera normalement jamais une réponse fausse. L'aide ne peut s'activer qu'une seule fois, donc lors du clic, un pop-up de confirmation s'affiche pour demander la confirmation d'activation de l'aide. Après le premier clic, le joueur ne peut plus cliquer sur le bouton et il devient plus sombre.



2.2 - Affichage du pourcentage de victoire

Le deuxième bouton d'aide consiste à afficher le pourcentage de victoires selon la disposition des dames placées.



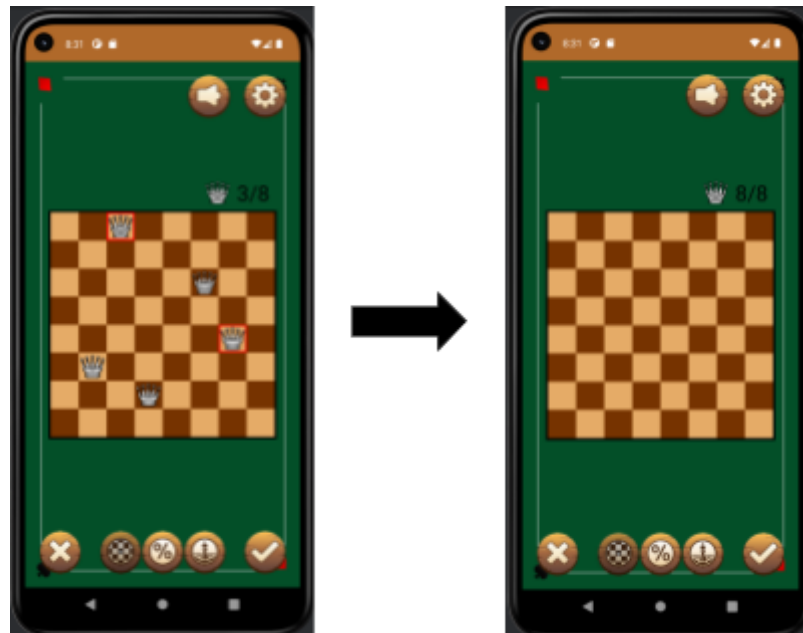
2.3 - Placement de dames

La troisième et dernière aide consiste à placer 3 dames aléatoirement selon une des 92 solutions possibles.



3 - Fonctionnalités de suppression

Nous avons également implémenté un bouton permettant de retirer toutes les dames placées d'un seul coup. Cette fonctionnalité permet à l'utilisateur de recommencer directement lorsqu'il a placé plusieurs dames. Elle rend l'expérience utilisateur plus agréable.



4 - Pop-up

Nous avons mis en place deux sortes de pop-up lors de l'appui de certains boutons qui exécutent des actions rédhibitoires telle que la suppression de toutes les dames placées ou encore de l'activation d'une aide non désactivable.

4.1 - Pop-up classique

Le premier type de pop-up est un pop-up normal qui est présent sur les boutons dont l'action est effectuée qu'une seule fois, comme la confirmation du jeu.



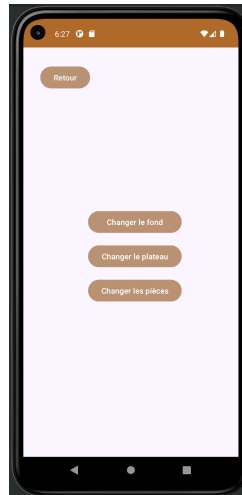
4.2 - Pop-up retirable

Le deuxième type de pop-up est un pop-up retirable. Il possède une checkbox "Ne plus afficher" qui permet si elle est cochée de ne plus afficher ce pop-up aux prochains clics sur le même bouton. Il est utilisé pour les boutons réutilisables comme le bouton de suppression de toutes les dames placées.

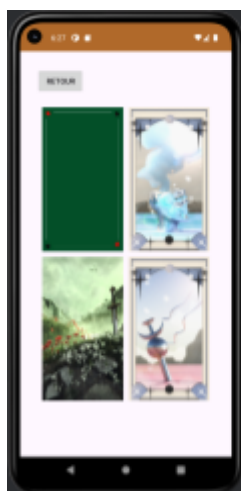


5 - Fonctionnalités esthétiques

L'ajout de paramètres esthétiques à un jeu de plateau peut améliorer l'expérience globale des joueurs en renforçant l'immersion, en permettant la personnalisation, en renouvelant l'intérêt et en différenciant le jeu sur le marché. Pour cela nous proposons aux utilisateurs 3 modes de personnalisations :



- Changement de fond d'écran : L'utilisateur a le choix entre 3 fond en plus du fond principal, ils ont tous été réalisés par [mhiro](#), un jeune artiste français, sauf le fond principal qui a été réalisé par Arthur.
- Changement de couleurs du plateau : L'utilisateur peut choisir entre 4 combinaisons de couleurs permettant une personnalisation accrue directe du jeu.
- Changement des pièces : L'utilisateur a le choix entre 4 styles de dames différentes.



6 - Design

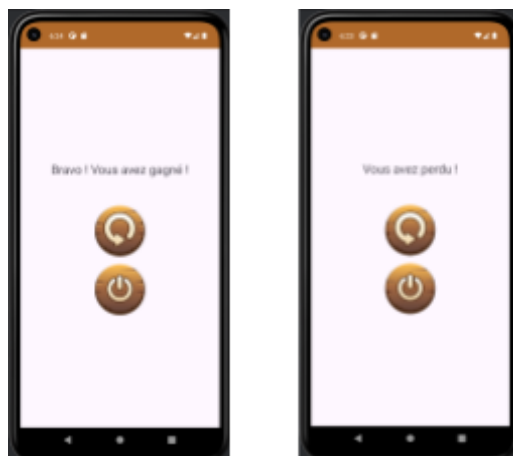
6.1 - Logo

Pour créer notre logo, nous avons utilisé l'intelligence artificielle pour obtenir un résultat unique et cohérent avec notre thème esthétique. Nous avons utilisé l'IA DALL-E en lui fournissant des directives pour créer un design « boisé » et le nom du jeu, ce qui correspondait parfaitement à l'image que nous souhaitions projeter pour notre projet.



6.2 - Boutons

Pour la création des boutons de notre projet, nous avons choisi d'explorer les ressources disponibles sur Freepik.com, en raison de sa variété et de la qualité des éléments visuels proposés. Après avoir sélectionné les boutons correspondant le mieux à notre esthétique, nous les avons retravaillés sur Figma. Cette étape nous a permis d'apporter des ajustements et des personnalisations pour garantir une bonne harmonisation de notre projet. Nous avons notamment modifié les couleurs, les formes pour maintenir une cohérence esthétique avec notre identité visuelle globale. En adoptant cette approche, nous avons pu obtenir des boutons à la fois fonctionnels et esthétiquement en accord avec notre vision créative.



6.3 - Animations

Nous avons créé une animation pour l'appui sur les boutons afin de rendre l'application plus esthétique. Cette animation crée un effet d'écrasement du bouton et rend le clic du joueur plus réaliste.



6 - Musique

Nous avons intégré une musique libre de droits dans notre application en utilisant la classe MediaPlayer fournie par l'API Android. Cette musique a été sélectionnée avec soin pour correspondre à l'atmosphère et au thème de notre projet. Nous avons inclus un bouton de contrôle qui permet aux utilisateurs de couper ou de réactiver la musique à tout moment pendant leur partie (voir animation ci-dessus). Grâce à l'utilisation de la classe MediaPlayer, nous avons pu gérer facilement la lecture audio et offrir une expérience sonore personnalisable.

IV - Détails techniques

1 - Classe Material & Interface CosmeticListener

Cette classe contient à la fois des attributs et des méthodes statiques. Son rôle est de fournir de manière dynamique des informations sur les cosmétiques à toutes les autres classes qui en ont besoin, et de les mettre à jour en temps réel grâce à un observateur. Dès que le joueur modifie une caractéristique, l'activité principale se met automatiquement à jour. Cette classe est étroitement liée à l'interface CosmeticListener, qui permet à toute classe qui l'implémente de définir les actions à entreprendre lorsqu'une caractéristique du joueur est modifiée, principalement pour mettre à jour les informations nécessaires.

2 - Classe BoutonListener

Cette classe permet l'affichage des pop-ups, ainsi que l'exécution de l'animation de clic d'un bouton. Chaque bouton est associé à une instance de BoutonListener.

V - Répartition du travail

Lors de la conception de l'application nous avons divisé le travail en plusieurs branches afin de travailler plus rapidement et efficacement, nous avons donc réparti en fonction de nos facilités et de nos connaissances.

Arthur a principalement dirigé les aspects esthétiques de l'application, en mettant en place des fonctionnalités permettant au joueur de personnaliser son expérience de jeu selon ses préférences. Il a également développé des composants graphiques pour améliorer l'expérience utilisateur de l'application.

Alexy s'est principalement concentré sur la classe MainActivity, qui abrite les fonctions principales de gestion du jeu. En plus de cela, il a travaillé sur le développement des fonctionnalités d'aide et sur l'animation des boutons de l'interface. Il a également pris en charge la gestion de Git et des fusions de branches.

Yasin a principalement été responsable des mises en page de l'application en concevant entre autres les designs des boutons et du logo. En outre, il a intégré une fonctionnalité musicale dans le jeu pour enrichir l'expérience utilisateur.

VI - Conclusion

Pour conclure, ce projet nous a énormément apporté, que ce soit du côté développement sur Smartphone et plus précisément Android, mais aussi en termes d'organisation et gestion de travail, ainsi qu'en travail sur l'expérience utilisateur et l'interface utilisateur.

Nous avons pu complètement apprendre l'utilisation de Java à travers une application Android en suivant les cours donnés, en utilisant les ressources proposées sur internet et en s'aidant d'IA génératives telles que ChatGPT.

Enfin pour la partie design UX/UI nous avons quelques connaissances de part nos expériences extérieures et en nous documentant sur les différentes façon de faire nous avons pu développer une UX/UI assez convenable pour une application de ce type.