

Projet Informatique 1^{ère} Année
PRO 3600

Jeu XKube

MOREAU Romain
LANTIGNY Valentin
ROUX Basile
FOISSY Thomas

Enseignant responsable : **SIMATIC Michel**

24 Février 2023

Sommaire

1	Introduction.....	3
2	Cahier des charges.....	4
3	Développement.....	5
3.1	Analyse du problème et spécification fonctionnelle.....	5
4	Conclusion.....	6
5	Bibliographie.....	7
6	Annexes.....	8
6.1	Annexe 1 : Gestion de projet.....	8
6.2	Annexe 2 : Code source.....	8
6.3	Annexe 3 :	9

1 Introduction

L'objectif de notre projet est de développer une application pour mobile. Cette application se basera sur un jeu de logique inspiré du Morpion et de la Pipopipette. Ce dernier se joue normalement sur papier avec 2 joueurs qui s'affrontent. On représente initialement une grille de 4x4 cases dont on ne trace que les coins. On doit alors avoir 25 points disposés sur 5 lignes et 5 colonnes. Chacun leur tour, les joueurs possèdent 2 coups qu'ils devront utiliser en reliant 2 points l'un avec l'autre pour former une barre horizontale ou verticale. Ce sont les joueurs qui tracent la grille petit à petit.

Lorsqu'une des cases voit ces 4 côtés être tracés, l'un des joueurs peut dépenser l'un de ces coups pour tracer une croix de sa couleur sur cette case. L'objectif sera d'être le premier à aligner 3 croix de sa couleur en ligne, en colonne ou en diagonal comme au morpion.

D'apparence simpliste, il s'avère enfaîte très compliqué de trouver une stratégie gagnante dans 100% des cas, car tenter d'anticiper un coup fait augmenter le nombre de possibilité de manière exponentielle comme au échec. Par ailleurs, une partie ne ressemble jamais à une autre et le jeu semble symétrique pour les 2 joueurs : le premier qui joue n'a pas plus de chance de gagner que le deuxième.

En développant ce jeu sur mobile, l'objectif serait de pouvoir faire une partie rapidement avec d'autres joueurs. Cela faciliterait la mise en place du jeu : plus besoin de feuille et de crayon.

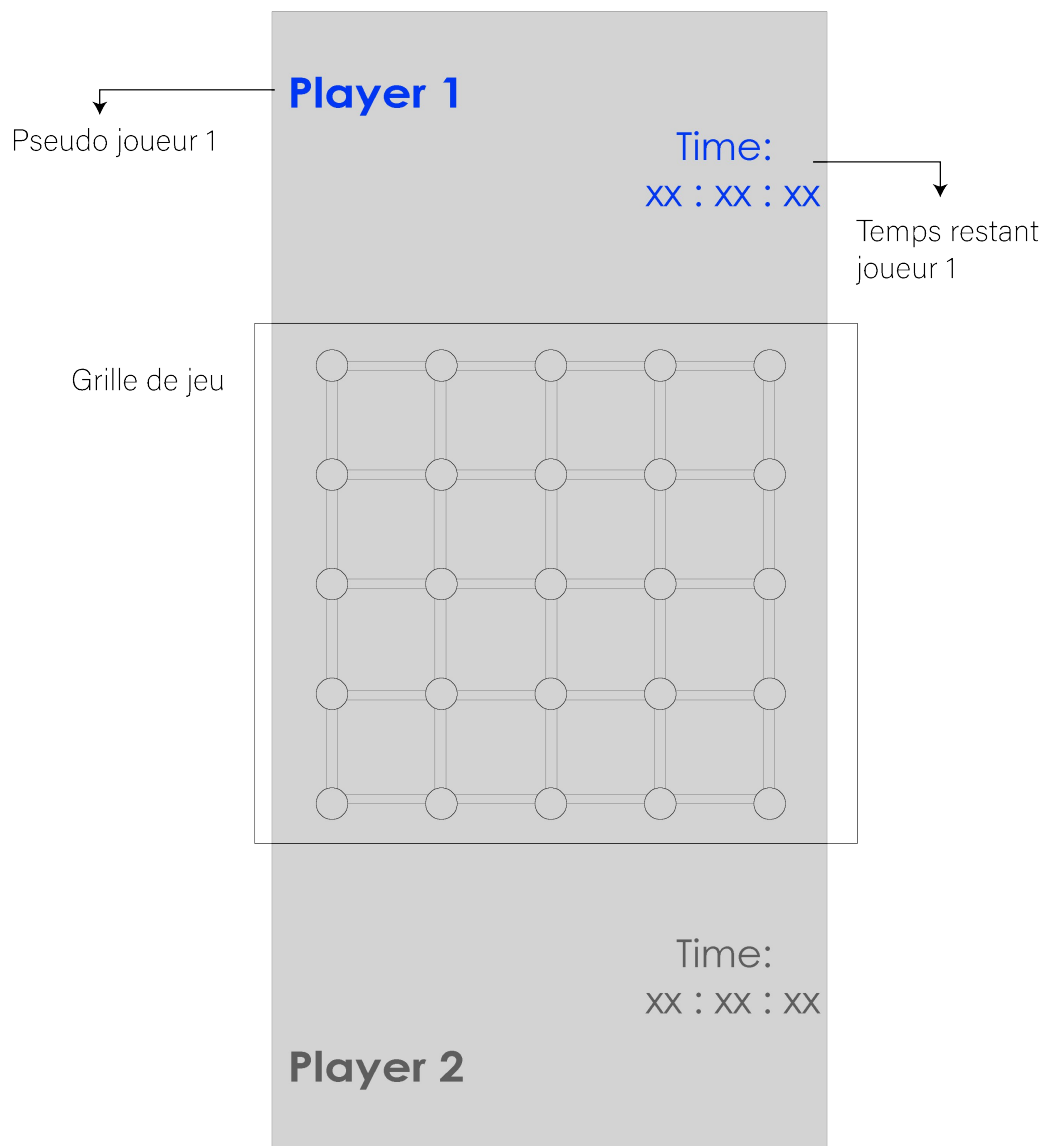
Par ailleurs, cette application qui nécessite peu de fonctionnalités sera une bonne occasion pour toucher à des outils et des domaines que nous n'avons pour l'instant jamais utiliser : Utiliser un IDE spécialisé pour les applications mobiles (Android studio), faire fonctionner un jeu multijoueur sur différents appareils, utiliser les réseaux de neurones.

2 Cahier des charges



Écran Principal de L'Application

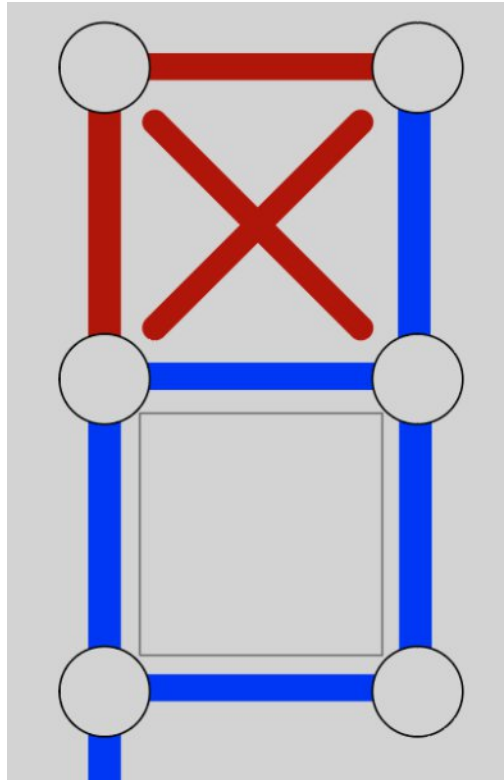
L'application s'ouvrirait sur un écran permettant de choisir son mode de jeu. On pourra jouer à 2 joueurs sur le même appareil, ou chacun sur son propre téléphone en connexion locale. Il y aura également possibilité de jouer en solo contre une IA, On retrouve également le Logo de l'application ainsi que plusieurs options : gérer la luminosité, le son, obtenir un tutoriel, changer la langue, ajouter un shop avec des accessoires pour les joueurs, changer l'apparence du jeu, diminuer la qualité des animations pour les petits portables.



Écran Lors du Lancement du Jeu

Une fois une partie lancée, l'écran de jeu ressemblera à ceci avec les emplacements pour cliquer pré-dessinés, les pseudos des 2 joueurs et un chrono pour éviter que la partie s'éternise. Le joueur qui doit jouer est celui dont le pseudo est en couleur.

Le fond n'est pas encore défini mais devrait également indiquer par sa couleur, quel joueur doit jouer.



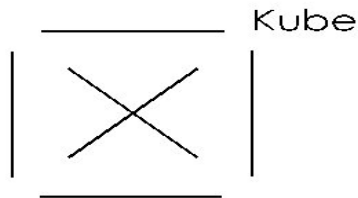
Aspect après quelques coups des joueurs

Chaque joueur aura une couleur arbitraire qui lui sera attribué : par défaut bleu ou rouge. Chacun de ses coups sera représenté avec cette couleur. En particulier, le dernier coup joué par un joueur sera d'une couleur légèrement plus clair pour qu'on puisse clairement identifier quels coups ont été joué au dernier tour.

L'objectif est que le jeu se déroule selon les règles du jeu de base : 2 coups par joueur, interdiction de joué une croix si les 4 côtés ne sont pas remplis, victoire du joueur qui aligne 3 croix peu importe la direction.

3 Développement

3.1 Analyse du problème liste des fonctionnalités



Écran Principal de L'Application

Nous avons pu implémenter les 3 modes de jeux (Local, Multijoueur et IA) que l'on peut lancer en appuyant sur le bouton associé.

- Le mode local permet à 2 personnes de jouer sur le même appareil en appuyant chacun leur tour sur l'écran.

- Le mode Multijoueur redirige vers un écran d'attente et lance une partie lorsqu'un 2ème joueur demande également à jouer en multijoueur.

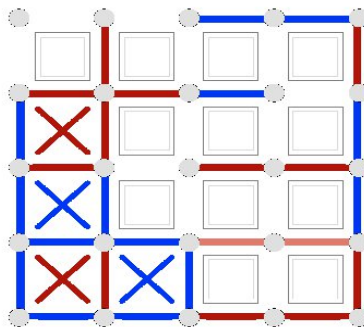
- Le mode IA se joue en solo : le joueur commence et l'IA joue derrière lui en essayant toujours de gagner.

Dans tous les cas, la partie se termine quand un joueur remplit une condition de victoire ou que

le chrono de son adversaire arrive à 0. Un écran apparaît, félicitant le joueur concerné. On peut alors revenir à l'écran d'accueil en cliquant une nouvelle fois sur l'écran.

Nous n'avons toute fois pas créé les options initialement annoncées faute de temps et de priorité. A la place 2 autre les remplacent : La possibilité de modifier le temps du chronomètre et l'activation d'une option : le DLC qui ajoute une modification au mode de jeu de base.

1 50 6



2 29 7

Écran durant la partie

Les règles du jeu sur papier ont bien été retranscrit dans l'application. Les conditions de victoires sont toutes implémentées (alignement et temps épuisé).

4 Conclusion

Les premières heures consacrées aux projets ont permis d'accomplir plusieurs objectifs.

En premier lieu, une conception préliminaire a permis de définir notre interface de jeu, ainsi que les spécialités nécessaire pour son bon fonctionnement. Dans un second temps, le développement du code source permet d'avoir une application fonctionnelle : le jeu se lance sur Android, et fonctionne correctement (alternance des couleurs en fonction du joueur, etc.). Enfin, l'interface et les graphismes ont été conçus, permettant d'avoir par la suite une application plus esthétiques.

Pour les prochaines étapes, il reste à concevoir et à mettre en place une connexion pour jouer en multijoueur ainsi que la formation d'une intelligence artificielle afin de jouer seul. Une fois ces différentes étapes accomplis, il restera l'implémentation des outils divers assurant une meilleure expérience pour le joueur.

5 Bibliographie

6 Annexes

6.1 Annexe 1 : Gestion de projet

6.1.1 Planning prévisionnel

PLAN DE CHARGES PRÉVISIONNEL

Description de l'activité	Charge en %	Charge en H	Charge en H / Participant			
			Romain	Thomas	Basile	Valentin
Total	100%	211	53	51	53	54
Gestion de projets	22%	46	12	11	11	12
Réunion de lancement	2%	4	1	1	1	1
Planning prévisionnel et Suivi d'activités	2%	4	1	1	1	1
Réunions de suivi	11%	24	6	6	6	6
Rédaction des rapports	3%	6	2	1	1	2
Mise en place de Git et gestion de l'espace	4%	8	2	2	2	2
Spécification	4%	8	2	2	2	2
Définition des fonctionnalités	4%	8	2	2	2	2
Conception préliminaire	11%	24	6	6	6	6
Définition du type d'application	2%	4	1	1	1	1
Définition du langage et des outils nécessaires	2%	4	1	1	1	1
Définition des fonctionnalités	6%	12	3	3	3	3
Définition des graphismes	2%	4	1	1	1	1
Conception détaillée	21%	45	12	11	11	11
Définition des classes	2%	4	1	1	1	1
Définition des méthodes	8%	16	4	4	4	4
Essais	6%	12	3	3	3	3
Auto-formation	4%	8	2	2	2	2
Interfaces et graphismes	2%	5	2	1	1	1
Code (jeu fonctionnel)	16%	33	9	9	9	9
Codage des classes	4%	8	2	2	2	2
Codage des méthodes	10%	21	6	6	6	6
Essais	2%	4	1	1	1	1

Intégration des fonctionnalités	17%	36	8	8	10	10
Implémentation de l'IA	9%	20	5	5	5	5
Implémentation du multijoueur	5%	10	1	1	4	4
Implémentations diverses	3%	6	2	2	1	1
Soutenance	8%	16	4	4	4	4
Préparation de la soutenance	6%	12	3	3	3	3
Soutenance	2%	4	1	1	1	1

6.1.2 Suivi d'activités

SUIVI D'ACTIVITÉS (Charge Consommée)

Description de l'activité	Charge en %	Charge en H	Charge en H / Participant			
			Romain	Thoma s	Basile	Valenti n
Total	43%	91	22	20	24	25
Gestion de projets	30%	27	7	7	6	7
Réunion de lancement	4%	4	1	1	1	1
Planning prévisionnel et Suivi d'activités	4%	4	1	1	1	1
Réunions de suivi	13%	12	3	3	3	3
Rédaction des rapports	3%	3	1	1	0	1
Mise en place de Git et gestion de l'espace	4%	4	1	1	1	1
Spécification	9%	8	2	2	2	2
Définition des fonctionnalités	9%	8	2	2	2	2
Conception préliminaire	26%	24	6	6	6	6
Définition du type d'application	4%	4	1	1	1	1
Définition du langage et des outils nécessaires	4%	4	1	1	1	1
Définition des fonctionnalités	13%	12	3	3	3	3
Définition des graphismes	4%	4	1	1	1	1
Conception détaillée	34%	31	6	5	10	10
Définition des classes	7%	6	1	1	2	2
Définition des méthodes	9%	8	1	1	3	3
Essais	4%	4	1	1	1	1
Auto-formation	11%	10	2	2	3	3
Interfaces et graphismes	3%	3	1	0	1	1
Code (jeu fonctionnel)	0%	0	0	0	0	0

Codage des classes	0%	0	0	0	0	0
Codage des méthodes	0%	0	0	0	0	0
Essais	0%	0	0	0	0	0
Intégration des fonctionnalités	0%	0	1	0	0	0
Implémentation de l'IA	0%	0	0	0	0	0
Implémentation du multijoueur	0%	0	0	0	0	0
Implémentations diverses	0%	0	1	0	0	0
Soutenance	0%	0	0	0	0	0
Préparation de la soutenance	0%	0	0	0	0	0
Soutenance	0%	0	0	0	0	0

6.2 Annexe 2 : code source

Lien vers le gitlab avec le code source :

<https://gitlab.com/Racoon-r/Projet-dev>