BAUD Adrien  
CHAMPAGNON Adrien  
DEBURE Margaux  
GREYL Robin  
RAYBAUD Vincent  
ROUSSEL Diego  
VARREL Lilian

**Projet Informatique 1**

**Création d’un jeu de Quarto**

**-**

**Document technique**

A l’attention de M. HADDAD

Table des matières

[Présentation du projet 3](#_Toc420429605)

[Diagramme des modules 3](#_Toc420429606)

[Partie Moteur 4](#_Toc420429607)

[Partie Graphique 5](#_Toc420429608)

[Partie IA 5](#_Toc420429609)

[Annexes 6](#_Toc420429610)

[Annexe 1 : PERT 6](#_Toc420429611)

[Annexe 2 : GANTT 7](#_Toc420429612)

# Présentation du projet

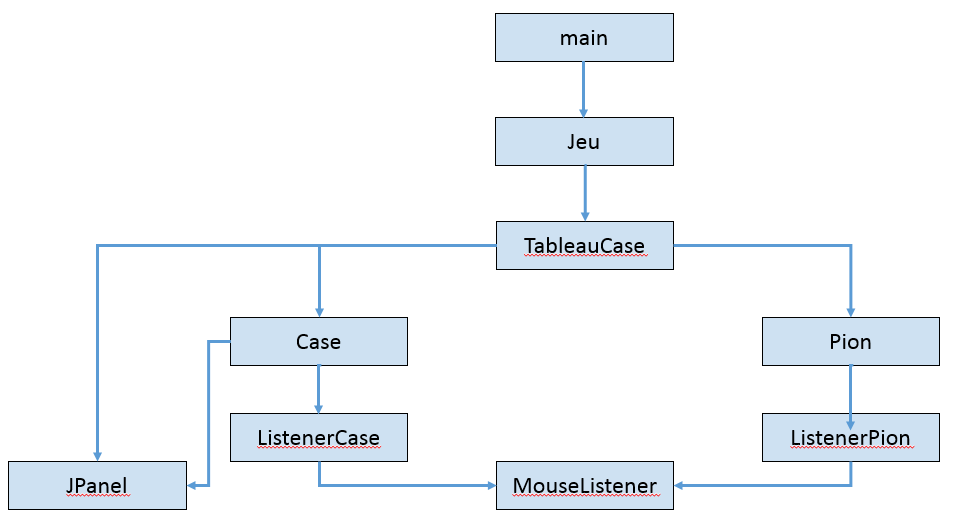
L’objectif de notre projet était de créer un jeu de quarto. Un jeu de quarto se compose d’un plateau de 16 cases et de 16 pions munis de 4 caractéristiques :

* Une couleur (clair / foncé)
* Une hauteur (petit / grand)
* Un sommet (troué / plein)
* Une forme (rond / carré)

Le but du jeu est d’aligner 4 pions possédant au moins une caractéristique commune. Pour se faire, à tour de rôle, un joueur donne une pièce à l’autre joueur qu’il devra placer sur le plateau.

Pour réaliser notre projet, nous avons séparé notre groupe en 3 équipes, après avoir suivi une phase d’apprentissage commune à tous. Nous avons une équipe dédiée à la création du moteur du jeu, une à la partie graphique, et une dernière pour l’IA.

# Diagramme des modules



# Partie Moteur

Pour réaliser notre projet nous avons été amené à imaginer et écrire différentes classes et leurs fonctions. En voici un résumé :

* Main

On crée la fenêtre du jeu en définissant sa taille, sa localisation, …

Cette classe contient seulement la fonction de création de la fenêtre

* TableauCase

On gère nos deux tableaux (jeu et réserve) qui sont composés de plusieurs cases.

Cette classe contient des fonctions pour obtenir une case en entrant ses coordonnées.

* Case

Les cases sont présentes sur le plateau de jeu et dans la réserve, elles sont vides ou contiennent un pion, elles peuvent être sélectionnées.

Cette classe contient principalement une fonction pour savoir si une case est sélectionnée.

* ListenerCase

Permet de gérer les cliques sur des cases.

* Pion

On regarde les caractéristiques d’un pion (carré, rond, plein, creux, …) et on les dessine. Ces pions sont contenus dans des cases.

Cette classe contient des fonctions pour savoir les attributs des pions (isgrand, isplein, …), savoir où il est placé et si il est sélectionné.

* ListenerPion

Permet de gérer les cliques sur des pions.

* Jeu

C’est dans cette classe que le jeu est initialisé (on place les pions dans la réserve) et que le jeu se déroule et se termine.

Cette classe contient principalement une fonction de fin de jeu, une fonction pour savoir si un ensemble de 4 pions forme une combinaison gagnante, une fonction pour savoir quel joueur joue en ce moment et une fonction pour savoir si un élément est sélectionné.

# Partie Graphique

Dans un premier temps, nous nous sommes intéressés au point de vue du joueur sur le plateau. Nous avions d’abords pensé qu’une vue isométrique permettrait de rendre un effet 3D assez plaisant sur le jeu mais nous avons très vite compris que c’est angle obstruais la vision sur les case les plus reculé du plateau. Ainsi, nous avons changé le point de vu, nous avons choisi une vue de face un peu surélevé de façon à ce qu’une pièce ne cache pas les case non joué situés derrière elle.

Ensuite, nous avons commencé à dessiner les pièces du jeu sur un Logiciel de dessin 3D. Pour ce qui est des dimensions nous avons trouvé sur internet des dimensions ce qui nous a permis de réalisé de joli pièce correctement proportionnées.

Enfin nous avons intégré les 16 différentes images dans le programme du jeu. Pour cela nous avons utilisé des classes de la librairie standard Java permettant le traitement d’image :

* Java.awt.Image
* javax.swing.ImageIcon

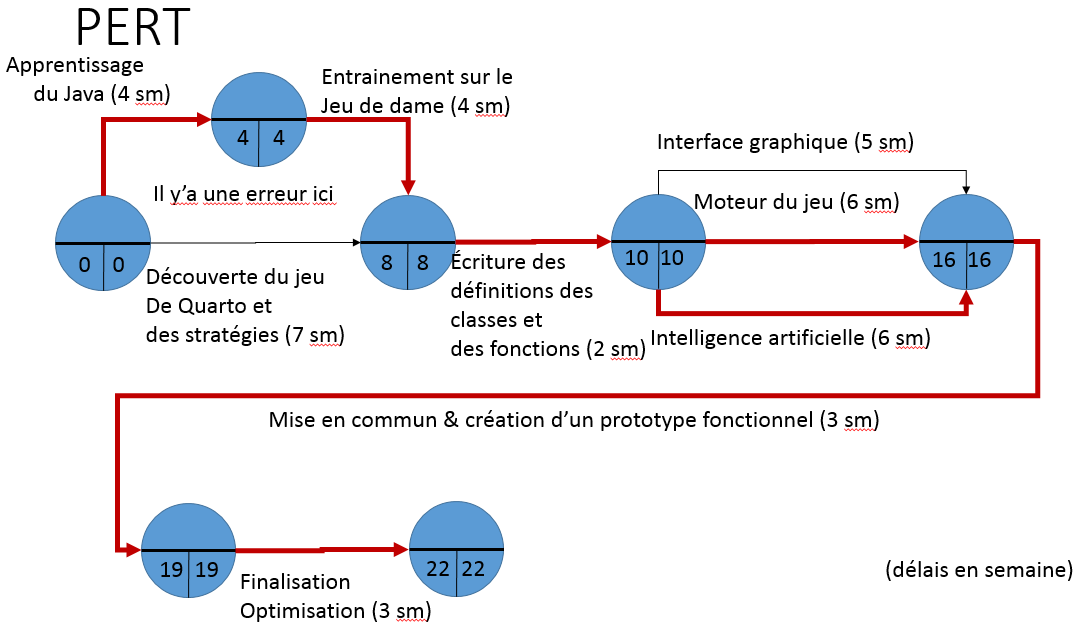
Nous avons donc utilisé quelques fonctions pour s’adapter au code de l’équipe de moteur du jeu :

* donneImage() : permet d’associé une image à un Pion
* paintComponent(Graphics g) : utilisée de différentes façon pour afficher tous les éléments présents dans la fenêtre de jeu.

# Partie IA

# Annexes

## Annexe 1 : PERT



## Annexe 2 : GANTT

