

Projet d'étude – Azul Rapport intermédiaire

Ralph Mohdad- Nicolas Quatelas - Mouhamed Gando

Diallo –Thomas Choubrac- Thibault Canavaggio

Table des matières

1.	Analyse des besoins.....	1
a.	Diagramme de cas d'utilisation	2
2.	Réalisation du projet	3
a.	Diagramme de classe du projet	4
b.	Diagramme d'activité	5
c.	Diagrammes de séquence	6
3.	Conclusion	8
a.	Fonctionnalités traitées.....	8
b.	Analyse de la solution choisie.....	8
c.	Bilan sur l'organisation du découpage	9

1. Analyse des besoins

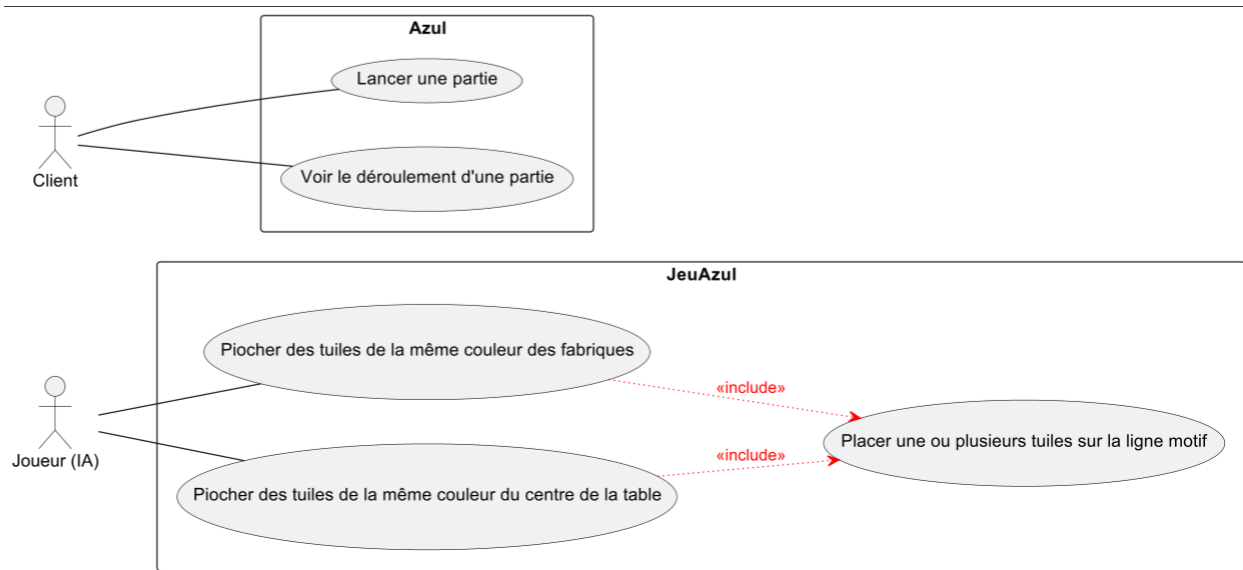
Le but de ce projet est d'accomplir la réalisation d'un jeu électronique basé sur le jeu de plateau Azul. Le jeu sera joué par une IA qui simulera les interactions qu'un joueur pourrait effectuer.

Glossaire des termes techniques du jeu :

Terme	Définition
Le plateau de jeu	Représente la surface sur laquelle seront placés les éléments du jeu au cours de la partie.
Tuile	Éléments du jeu qui sera utilisés à chaque tour de jeu. Elles permettent d'inscrire des points. Il existe différentes couleurs de tuiles : <ul style="list-style-type: none">- Rouge- Bleue- Noir- Jaune- Blanc- Vide (permet d'initialiser)
Le sac	Le sac contient 100 tuiles, 20 de chaque couleur. Le sac permet de remplir les fabriques.
Fabrique	La fabrique permet de prendre un groupe de tuile de la même couleur. La fabrique est remplie avec 4 tuiles venant du Sac.
Centre de la table	Lorsque le joueur pioche une tuile d'une des fabriques les tuiles qui n'ont pas été piochées sont placées au centre de la table.
Ligne Plancher	Toutes les tuiles placées sur la ligne plancher font perdre des points lors de la phase de décompte des points.
Ligne motif	Permet de placer les tuiles récupérées depuis la fabrique ou centre sur l'une des 5 lignes horizontales du plateau du joueur.
Mur	Chaque joueur peut prendre une tuile sur chaque ligne motif complète et la placer sur la même ligne correspondant à la même couleur du mur.
Tour de jeu	Tous les joueurs ont pioché une fois et ont posé une ou plusieurs tuiles.

Manche	Plusieurs tours jusqu'à que les fabriques et le centre soit vides, les fabriques obtiennent de nouvelles tuiles.
Coupole	Stocker l'excédent de tuiles de la ligne motif.

a. Diagramme de cas d'utilisation



Le joueur doit pouvoir :

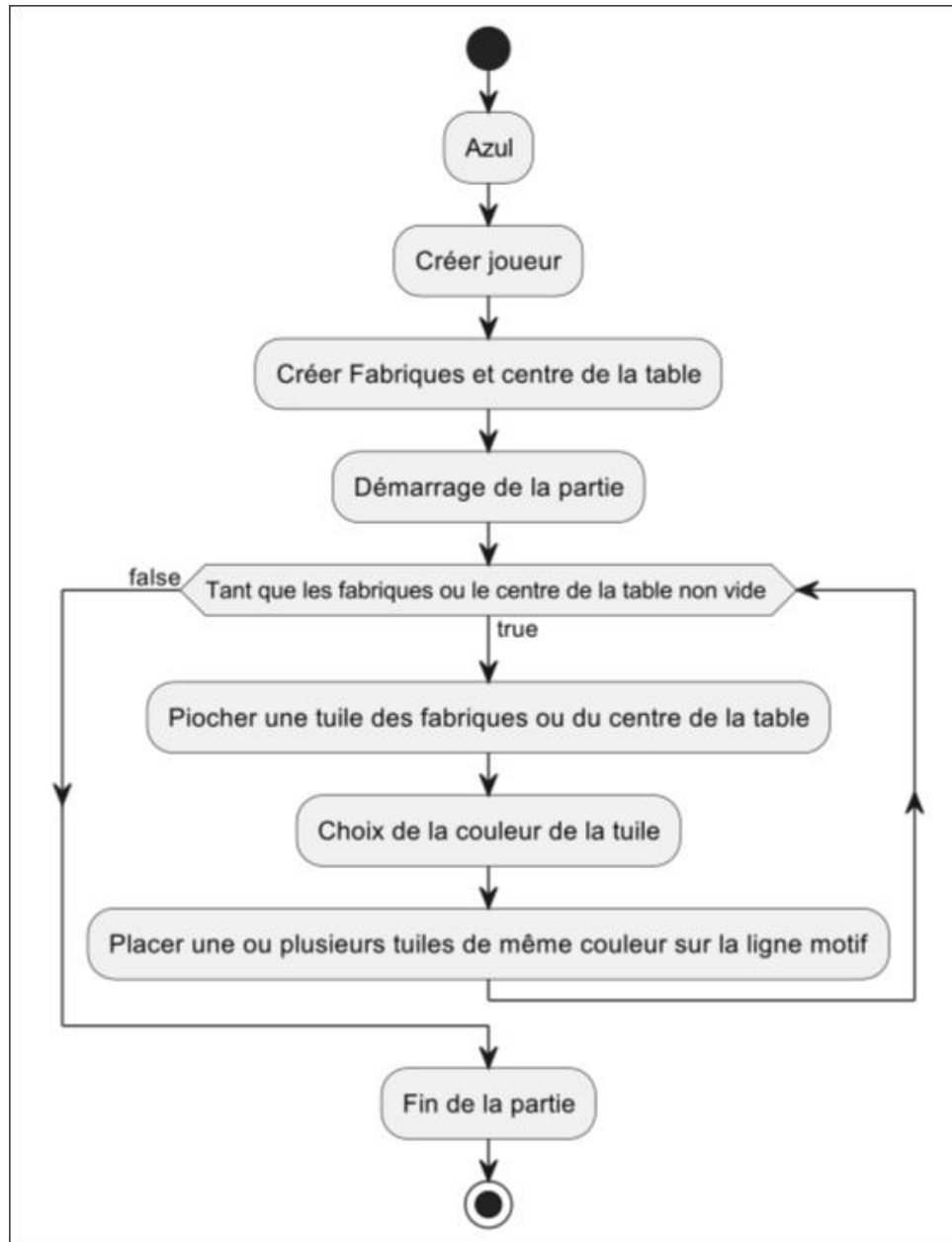
Piocher une ou plusieurs tuiles de même couleur depuis les fabriques ou le centre et les placer sur la ligne motif.

Jouer une partie avec plusieurs joueurs.

Le client doit pouvoir :

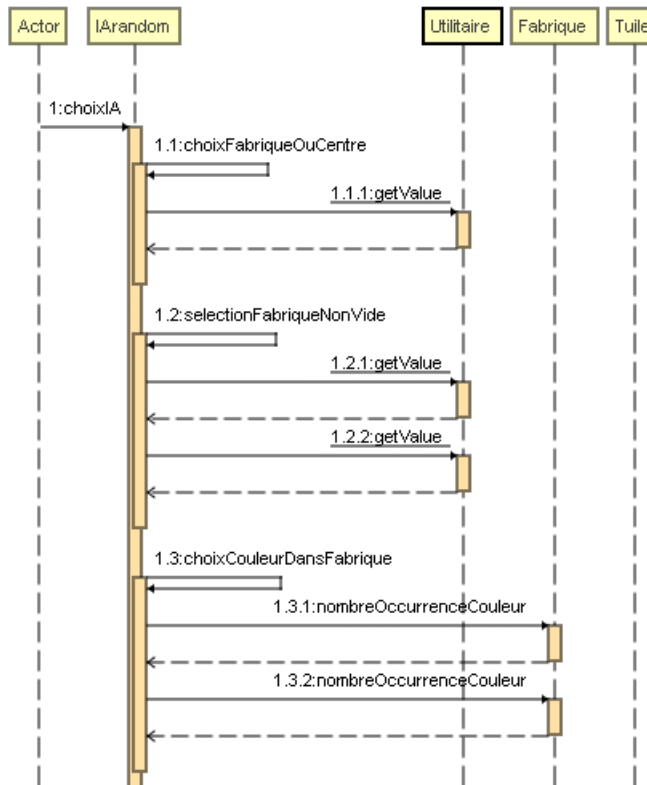
Avoir un aperçu du déroulement d'une partie et il doit pouvoir lancer une partie.

b. Diagrammes d'activité

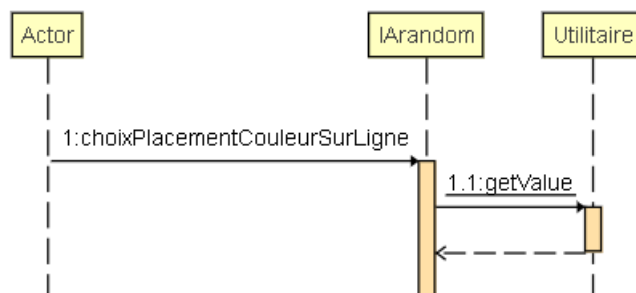


c. Diagrammes de séquence

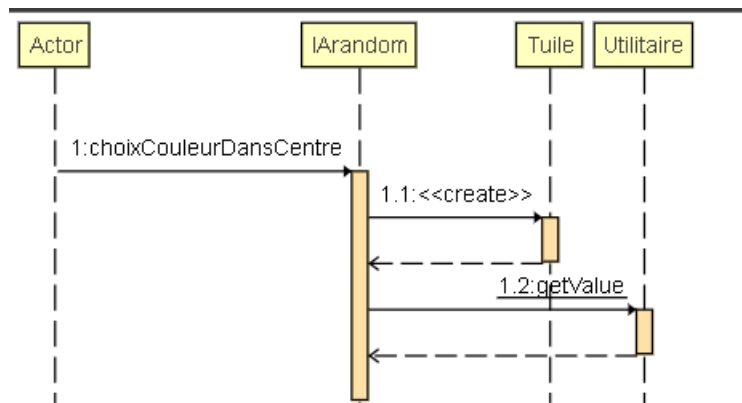
L'IA pioche une ou plusieurs tuiles de même couleur depuis les fabriques



L'IA place une ou plusieurs tuiles sur la ligne motif



L'IA pioche une ou plusieurs tuiles de même couleur depuis le centre de la table



3. Conclusion

a. Fonctionnalités traitées

Durant cette première phase de projet, nous avons implémenté les fonctionnalités de base du jeu.

Dans l'état actuel du jeu, le joueur peut :

Jouer plusieurs manches avec plusieurs joueurs.

Choisir des tuiles à partir d'une IA aléatoire.

Remplir les fabriques depuis la pioche.

Prendre plusieurs tuiles depuis une des fabriques ou bien le centre et placer sur les lignes de motifs

Nous n'avons pas encore traité les fonctionnalités requises par le client, à savoir les statistiques et le lancement de 500 parties. Cependant, le client peut déjà lancer plusieurs manches, car nous n'avons pas encore implémenté le système de coupole pour remplir la pioche à nouveau.

b. Analyse de la solution choisie

Avant de commencer la phase de développement, nous avons choisi de jouer au jeu afin de bien comprendre chacune des règles et spécificités. Bien comprendre le sujet est une étape clé dans le déroulement d'un projet.

Nous avons tout d'abord développé les éléments essentiels (itération 1) car il était nécessaire d'avoir cette base pour la développer lors des itérations suivantes.

Pour le développement du jeu nous avons choisi d'utiliser le français afin de faciliter la compréhension du code entre membres de projet et aussi pour le client de plus ça permet de gagner en efficacité.

Nous avons pris l'initiative de répartir notre code dans un nombre conséquent de package et de classe pour faciliter l'ajout de fonctions et la lisibilité des méthodes. Développer de cette façon nous permet de répartir plus facilement le travail et de mieux analyser les problèmes.

Cependant, nous avons eu tendance à tout mettre dans une seule classe et ne pas découper en sous-classe ce qui nous a poussé à revoir notre structuration durant l'itération 2 et 3.

c. Bilan sur l'organisation du découpage

Au départ, nous avons réalisé un premier découpage en 4 itérations que l'on a ensuite étalé en 8 itérations. Lors de la première itération, nous avons fait en sorte d'ajouter toutes les fonctionnalités nécessaires pour faire un jeu qui tourne pour le client.

Pendant la seconde itération, nous avons implémenter la fonctionnalité de pouvoir lancer une manche, nous avons aussi générer les fabriques et avons fait en sorte que l'IA pioche une couleur de tuile depuis la fabrique et la place sur la ligne motif.

Durant la troisième itération, nous avons fait un refactoring complet de notre code et avons isoler les choix de l'IA à part et avons implémenter la fonctionnalité joueurs qui par suite permet d'attribuer un niveau de IA à chaque joueur.

Après avoir rempli les trois premières livraisons, nous pouvons en conclure que notre découpage est correct, car nous ne sommes pas en retard et pas pour autant surchargés de travail.