

## Software Requirements Document: Tetris Royal

Chris Badi Badu (n°569082)      Anthony Van Der Wilt (n°569315)

Valentin Dias (n°593394)      Ismail Rhabouqi (n°584567)

Mohamed Boutaleb (n°566181)      Ikram Boutaleb (n°536046)

Oumaima Hamdach (n°568078)

December 2024

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Besoins utilisateurs: Fonctionnels</b>	<b>5</b>
2.1	Lancement du programme . . . . .	5
2.2	Menu de connexion . . . . .	5
2.2.1	Se connecter . . . . .	5
2.2.2	Créer un compte . . . . .	5
2.2.3	Quitter le programme . . . . .	5
2.3	Menu Principal . . . . .	6
2.3.1	Créer une partie . . . . .	6
2.3.2	Accéder à sa liste d'amis . . . . .	7
2.3.3	Consulter le classement . . . . .	8
2.3.4	Rejoindre une partie . . . . .	8
2.3.5	Déconnexion . . . . .	8
2.4	Partie de Tetris Royal . . . . .	8
2.4.1	Endless Mode . . . . .	8
2.4.2	Classic Mode . . . . .	9
2.4.3	Duel Mode . . . . .	9
2.4.4	Royal Competition Mode . . . . .	9
<b>3</b>	<b>Besoins utilisateurs: Non Fonctionnels</b>	<b>10</b>
<b>4</b>	<b>Besoins système : Fonctionnels</b>	<b>10</b>
4.1	Gestion des comptes . . . . .	10
4.1.1	Connexion à un compte . . . . .	10
4.1.2	Inscription d'un compte . . . . .	10
4.2	Gestion du Menu Principal . . . . .	11
4.2.1	Créer une partie . . . . .	11
4.2.2	Accéder à sa liste d'amis . . . . .	11
4.2.3	Rejoindre une partie . . . . .	11
4.2.4	Quitter le programme . . . . .	11
4.3	Gestion d'une partie . . . . .	11
4.3.1	Apparition des tétraminos . . . . .	11
4.3.2	Fin de la partie . . . . .	11
<b>5</b>	<b>Besoins système : Non Fonctionnels</b>	<b>12</b>
5.1	Système d'exploitation . . . . .	12
5.2	Language . . . . .	12
5.3	Réseaux . . . . .	12
5.4	Disponibilité . . . . .	12
5.5	Performances . . . . .	12
5.6	Capacité . . . . .	12
5.7	Sécurité . . . . .	12
<b>6</b>	<b>Design et fonctionnement du Système</b>	<b>13</b>

6.1	Diagrammes de Classe du serveur . . . . .	13
6.2	Diagrammes de Classe du serveur/database . . . . .	14
6.3	Diagrammes de Classe du client . . . . .	15
6.4	Fonctionnement du système . . . . .	16
6.4.1	Création d'un compte . . . . .	16
6.4.2	Connexion . . . . .	18
6.4.3	Cycle de jeu - mode Endless . . . . .	19
6.4.4	Cycle de jeu - mode Classic et Duel . . . . .	20
6.4.5	Cycle de jeu - mode Royal competition . . . . .	21
6.4.6	Chat entre les joueurs . . . . .	22
6.4.7	Création d'une partie . . . . .	23
6.4.8	L'envoi des malus - mode Classic et Duel . . . . .	24
6.4.9	L'envoi des malus - mode Royale . . . . .	25
<b>7</b>	<b>Historique</b>	<b>26</b>

# 1 Introduction

Le but de ce projet de groupe est de créer une application client-serveur permettant à un utilisateur de jouer au jeu *Tetris Royal*. L'utilisateur aura le choix de jouer en local en multi-joueurs ou en solo. Le concept de base du jeu est assez simple à comprendre. En effet, *Tetris* se joue sur une grille de 10x20 cases; des tétriminos apparaissent un par un dans la grille. Le but du joueur est de ne jamais remplir complètement la grille. Pour ce faire, il doit compléter une ou plusieurs lignes horizontales, qui disparaîtront grâce aux tériminos qui arrivent. Le joueur peut les déplacer horizontalement et les faire pivoter. Une fois qu'une ou plusieurs lignes sont détruites, tous les tériminos restants descendent d'un ou plusieurs rangs, selon le nombre de lignes détruites. L'utilisateur pourra également marquer des points pour chaque ligne complétée. Des combos apparaîtront lorsqu'au moins deux lignes sont complétées simultanément. (Quatre lignes terminées en un seul coup sont appelées un *Tetris*)

Cette application propose de nombreuses autres fonctionnalités en plus du jeu classique de *Tetris*. L'utilisateur pourra non seulement affronter d'autres joueurs ou jouer en solo dans différents modes de jeu plus palpitants les uns que les autres, mais il pourra également discuter avec ses amis ou observer leurs parties et consulter le classement des meilleurs joueurs !

## 2 Besoins utilisateurs: Fonctionnels

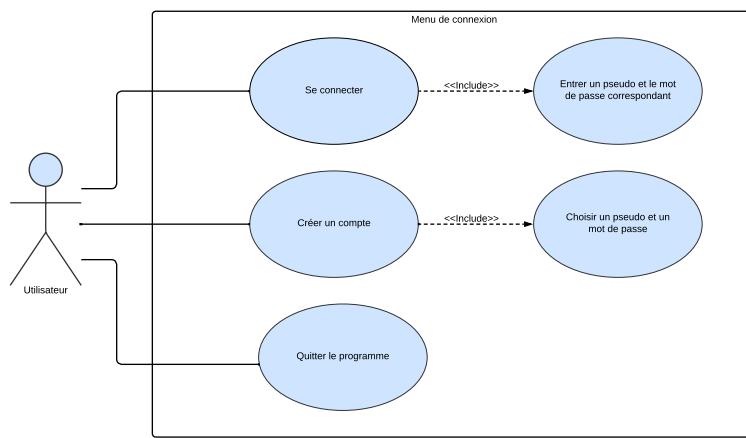
### 2.1 Lancement du programme

L'utilisateur a le choix de lancer le programme sous sa forme terminal ou sa forme d'application.

### 2.2 Menu de connexion

Comme représenté par la figure 1, lorsque l'utilisateur se trouve dans le menu de connexion, il aura le choix entre : se connecter, créer un compte ou quitter le programme.

Figure 1: Use Case - Menu de connexion



#### 2.2.1 Se connecter

Si l'utilisateur possède déjà un compte, il peut se connecter à l'application en saisissant son pseudo et son mot de passe correspondant. Si le pseudo saisi n'existe pas ou si le mot de passe est incorrect, un message d'erreur s'affichera et l'utilisateur pourra recommencer l'opération.

#### 2.2.2 Créer un compte

Si l'utilisateur souhaite créer un compte, il peut le faire en saisissant un nouveau pseudo qui n'est pas déjà présent dans la base de données, ainsi qu'un nouveau mot de passe. Si le pseudo est déjà utilisé, un message d'erreur apparaîtra et l'utilisateur aura la possibilité de recommencer l'opération.

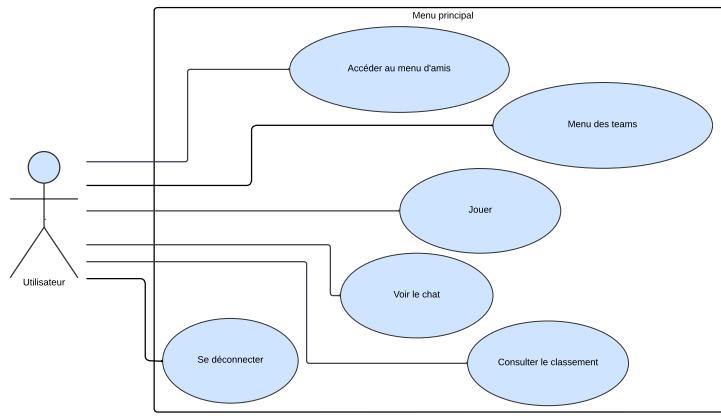
#### 2.2.3 Quitter le programme

L'utilisateur peut, s'il le souhaite, quitter le programme, qui se fermera sans problème.

## 2.3 Menu Principal

Comme représenté par la figure 2, une fois que l'utilisateur s'est connecté ou a créé un compte, il accède à la page du menu principal. Cette page constitue le point central de l'application. En effet, l'utilisateur pourra jouer (soit créer ou rejoindre une partie). Il pourra également aller dans le menu d'amis, visualiser le classement général des meilleurs joueurs, avoir accès au chat avec ses différents amis, et, enfin, se déconnecter de son compte actif.

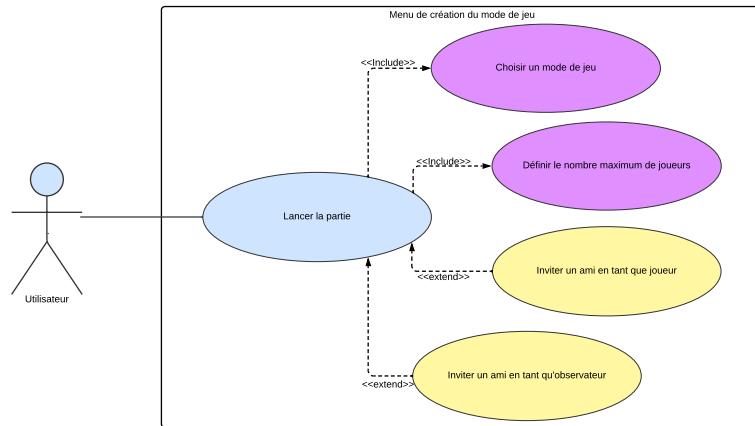
Figure 2: Use Case - Menu Principal



### 2.3.1 Créer une partie

L'utilisateur, s'il le souhaite, peut créer une partie. Pour ce faire, il devra configurer les paramètres de la partie. Il aura le choix du mode de jeu et du nombre maximum de joueurs. Il existe quatre modes de jeu différents : Endless, Classic, Duel et Royal Competition. Le nombre maximum de joueurs dépendra du mode de jeu sélectionné. Tous ces choix effectués se passent dans le lobby.

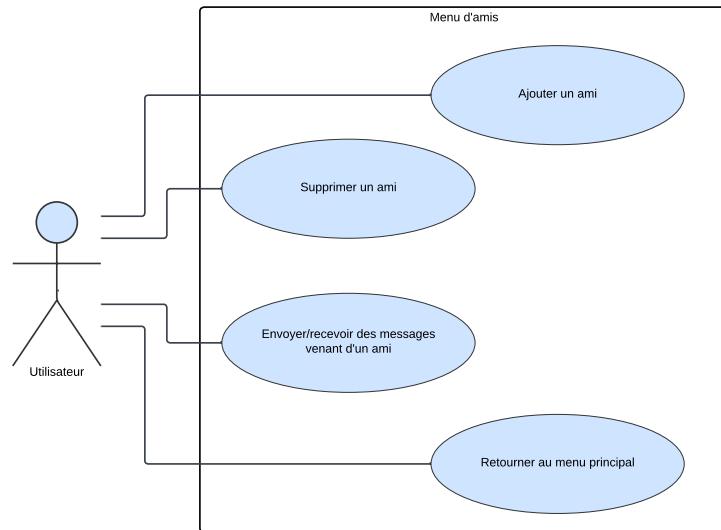
Figure 3: Use Case - Menu de création de parties



### 2.3.2 Accéder à sa liste d'amis

Dans le menu des amis, l'utilisateur aura la possibilité de gérer sa liste d'amis. Il pourra ajouter ou supprimer un ami et bien entendu, retourner au menu principal.

Figure 4: Use Case - Menu Amis



### 2.3.3 Consulter le classement

Le joueur pourra consulter le classement général des meilleurs joueurs du mode Endless. Ce classement sera mis à jour dès que l'utilisateur arrivera sur cette page.

### 2.3.4 Rejoindre une partie

Si l'utilisateur a reçu une invitation, il pourra rejoindre ou non la partie en tant que joueur ou spectateur, tout dépend du type d'invitation reçue.

### 2.3.5 Déconnexion

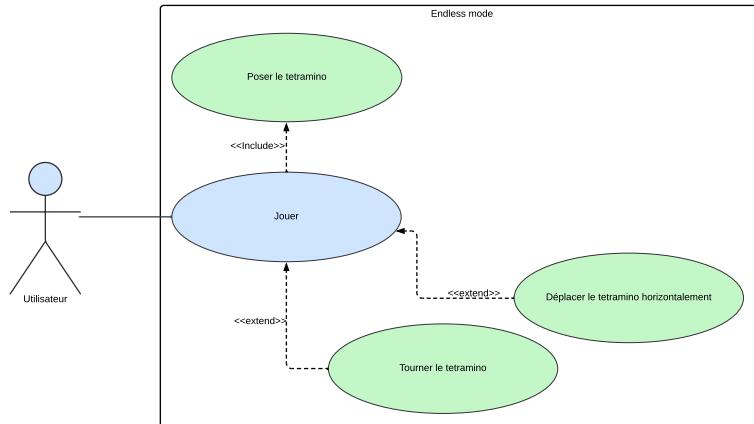
Se déconnecter.

## 2.4 Partie de Tetris Royal

### 2.4.1 Endless Mode

Ce mode de jeu correspond au mode de Tetris original. Le joueur joue seul et doit essayer de survivre le plus longtemps possible en obtenant le score le plus élevé. Plus le temps passe, plus les tétriminos tombent rapidement. La partie se termine dès qu'un tétrimino atteint le haut de la grille. Le score, ainsi que le pseudo du joueur, seront ensuite ajoutés au tableau de scores général.

Figure 5: Use Case - Endless Mode



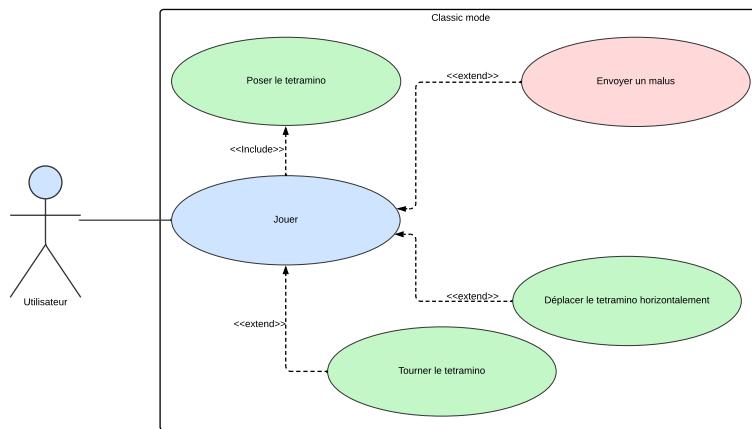
#### 2.4.2 Classic Mode

Le mode Classic est une variante du mode Endless. En effet, les règles de base du Tetris sont conservées, mais ce mode se joue de 3 à 9 personnes. Les joueurs ont la possibilité d'infliger des malus à leurs adversaires. Dès qu'un joueur réalise un combo (termine plusieurs lignes à la fois), il peut envoyer un malus à un autre joueur. Un malus consiste en l'apparition d'une ligne avec un bloc manquant en bas de la grille de l'adversaire. Deux lignes complétées correspondent à une ligne de malus pour l'adversaire, trois lignes complétées correspondent à deux lignes de malus, et enfin, pour quatre lignes complétées (un Tetris), ce sont quatre lignes de malus qui sont envoyées à l'adversaire ! Le dernier joueur encore en vie remporte la partie.

#### 2.4.3 Duel Mode

Le mode Duel est identique au mode Classic, sauf qu'au lieu de se jouer de 3 à 9 personnes, il se joue en 1 contre 1.

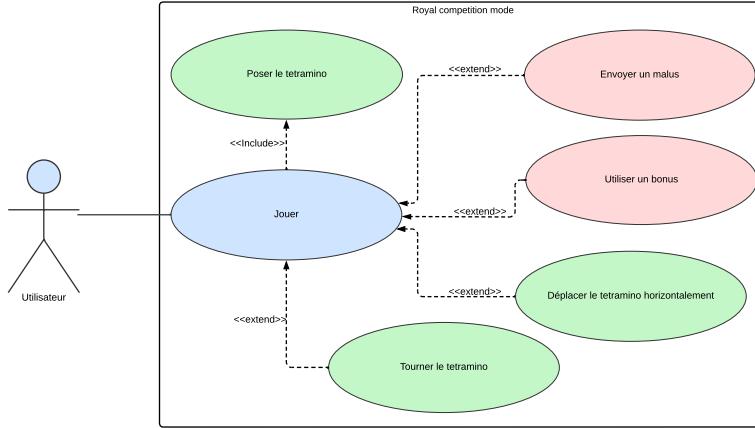
Figure 6: Use Case - Classic et Duel Mode



#### 2.4.4 Royal Competition Mode

Ce mode de jeu est une variante du mode Classic. En effet, il se joue également de 3 à 9 personnes, mais en plus des malus, des bonus sont également disponibles. Ces derniers ne sont pas envoyés automatiquement : lorsque le joueur accumule suffisamment d'énergie (en détruisant des blocs), il peut choisir d'envoyer un malus à un adversaire ou de s'accorder un bonus. Le dernier survivant remporte la partie.

Figure 7: Use Case - Royal Competition Mode



### 3 Besoins utilisateurs: Non Fonctionnels

Le jeu Tetris doit offrir une interface utilisateur intuitive et agréable, permettant à tous les joueurs, y compris les novices, de prendre en main rapidement et aisément les différentes fonctionnalités du jeu. Une navigation fluide et une manipulation réactive des tétriminos sont essentielles pour garantir une expérience de jeu optimale. Pour cela, chaque interaction doit être instantanée et sans latence, afin de maintenir une immersion totale, qu'il s'agisse de déplacer des pièces, de participer à des parties multijoueurs ou de naviguer dans les menus.

### 4 Besoins système : Fonctionnels

#### 4.1 Gestion des comptes

Les comptes créés seront enregistrés dans la base de données avec leurs informations correspondantes par le système. Chaque utilisateur se trouvera dans la base de données SQL possédant: un ID, un pseudonyme, un mot de passe, un meilleur score et un ID d'avatar.

##### 4.1.1 Connexion à un compte

Lorsqu'un utilisateur souhaite se connecter, le système vérifie si le pseudonyme et le mot de passe correspondent grâce aux informations enregistrées dans la base de données.

##### 4.1.2 Incription d'un compte

Lorsqu'un utilisateur souhaite créer un compte, le système vérifiera si le pseudonyme choisi n'est pas déjà utilisé. Si ce n'est pas le cas, il enregistrera le nouveau pseudonyme et le mot de passe dans la base de données.

## **4.2 Gestion du Menu Principal**

### **4.2.1 Crer une partie**

Le programme doit permettre à l'utilisateur de créer une partie en configurant divers paramètres, tels que le mode de jeu et le nombre de participants.

Lors de la création, l'utilisateur peut envoyer une invitation à ses amis. Cette invitation sera transmise du créateur au destinataire via le serveur.

### **4.2.2 Accéder à sa liste d'amis**

Le programme doit également permettre à l'utilisateur de gérer sa liste d'amis. Pour créer ou supprimer un "lien d'amitié" entre deux joueurs, une action non réciproque suffit. Si un utilisateur envoie une demande d'amitié à un autre utilisateur et que ce dernier l'accepte, le programme les ajoutera mutuellement en tant qu'amis. Si l'un des deux décide de supprimer l'autre de sa liste d'amis, le serveur cessera de les considérer comme amis.

Toutes les actions d'ajout ou de suppression d'amis sont gérées par le serveur, qui communiquera avec la base de données. Chaque ami d'un utilisateur se trouve dans la base de données.

### **4.2.3 Rejoindre une partie**

Le serveur permettra aux utilisateurs ayant reçu une invitation de rejoindre une partie, soit en tant que joueurs, soit en tant qu'observateurs. Il sera également chargé de déterminer le type d'invitation reçue.

### **4.2.4 Quitter le programme**

Le programme se ferme lorsque l'utilisateur quitte l'application.

## **4.3 Gestion d'une partie**

Le programme fait en sorte qu'un utilisateur ne voit que son plan de jeu, son score, son prochain tétramino et la grille des autres joueurs.

### **4.3.1 Apparition des tétraminos**

Le système fera apparaître des tétraminos de manière aléatoire pour l'utilisateur. Le téramino actuel se trouvera dans la grille, tandis que le prochain téramino sera affiché à côté de l'espace de jeu. Ils apparaîtront en haut de la grille et tomberont progressivement un à un.

### **4.3.2 Fin de la partie**

Lorsque le programme détecte une fin de partie, tous les joueurs recevront un message indiquant s'ils ont gagné ou perdu ainsi que leurs scores respectifs.

## **5 Besoins système : Non Fonctionnels**

### **5.1 Système d'exploitation**

Le jeu sera jouable sur le système d'exploitation Linux avec une optimisation pour la distribution Ubuntu. Bien qu'il soit également jouable sur d'autres distributions Linux, nous ne pouvons garantir le bon fonctionnement sur ces dernières.

### **5.2 Language**

Le jeu sera principalement développé en C++ en respectant les normes les plus récentes du langage.

### **5.3 Réseaux**

Le réseaux communiquera via des sockets permettant ainsi de gérer l'aspect multijoueur. Certains modes de jeu ne permettant qu'un nombre limité de joueurs dans la même partie, il sera donc nécessaire de limiter la création de socket en fonction du mode de jeu et de son nombre de joueurs autorisés par partie.

### **5.4 Disponibilité**

Le jeu étant en multijoueur, il ne sera jouable à plusieurs que si le serveur est en ligne.

### **5.5 Performances**

Le code sera optimisé pour être le plus robuste et performant possible afin de garantir la meilleure expérience utilisateur. Les temps de réponse seront donc de l'ordre de la milliseconde.

### **5.6 Capacité**

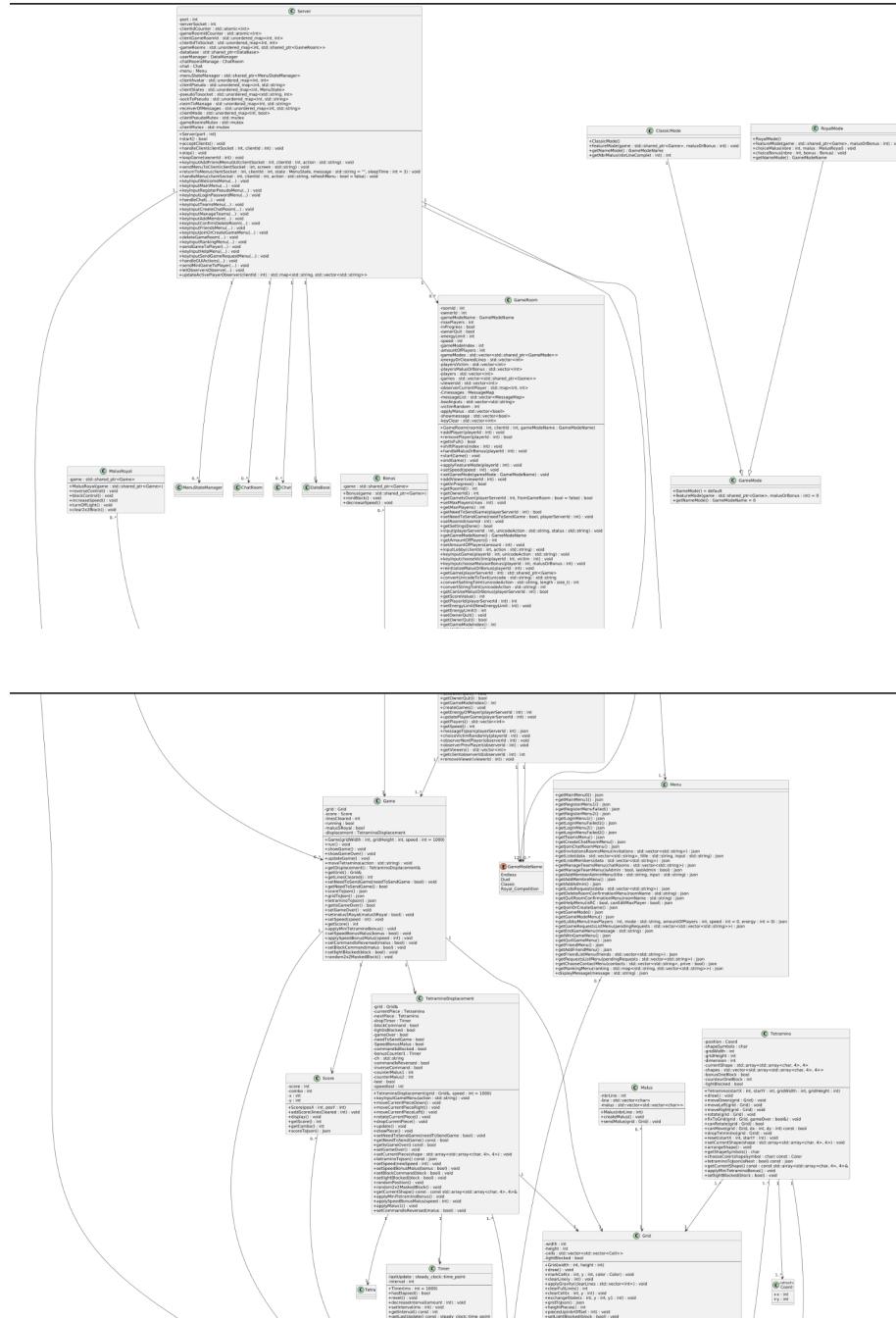
Le jeu devra supporter jusqu'à 9 joueurs actifs dans une même partie ainsi qu'un nombre "illimité" de spectateurs. Par conséquent, l'espace de stockage requis pour chaque partie sera donc de l'ordre du Megabyte (Mo).

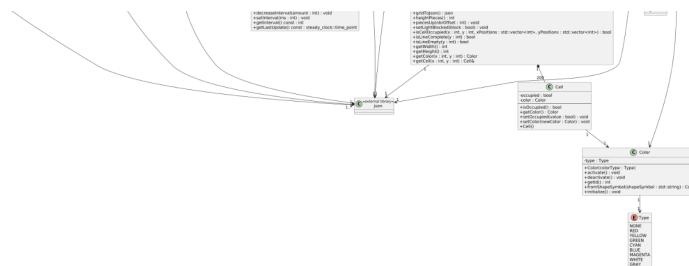
### **5.7 Sécurité**

Pour pouvoir lancer une partie, un joueur devra soit créer un compte soit se connecter à son compte via son pseudo et son mot de passe.

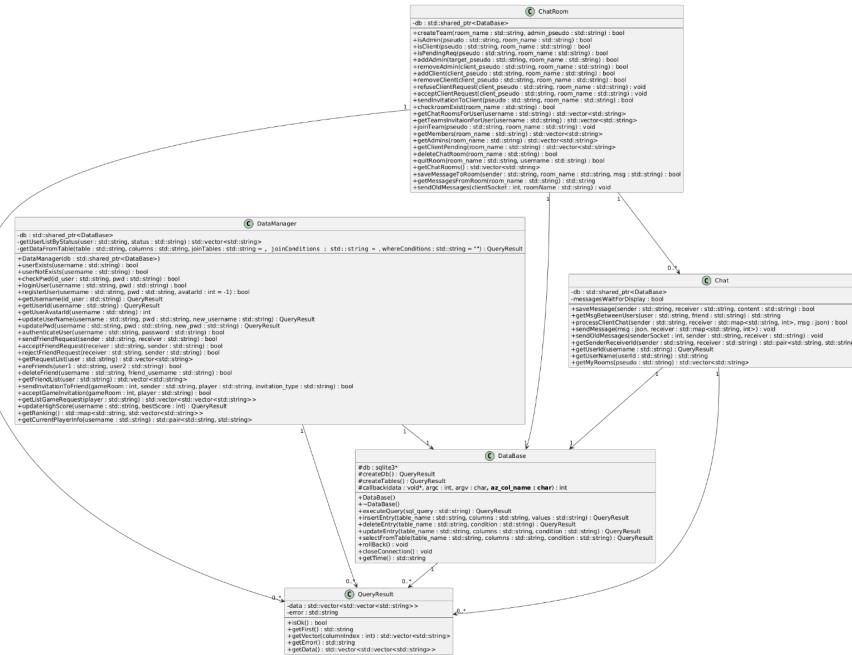
## 6 Design et fonctionnement du Système

### 6.1 Diagrammes de Classe du serveur

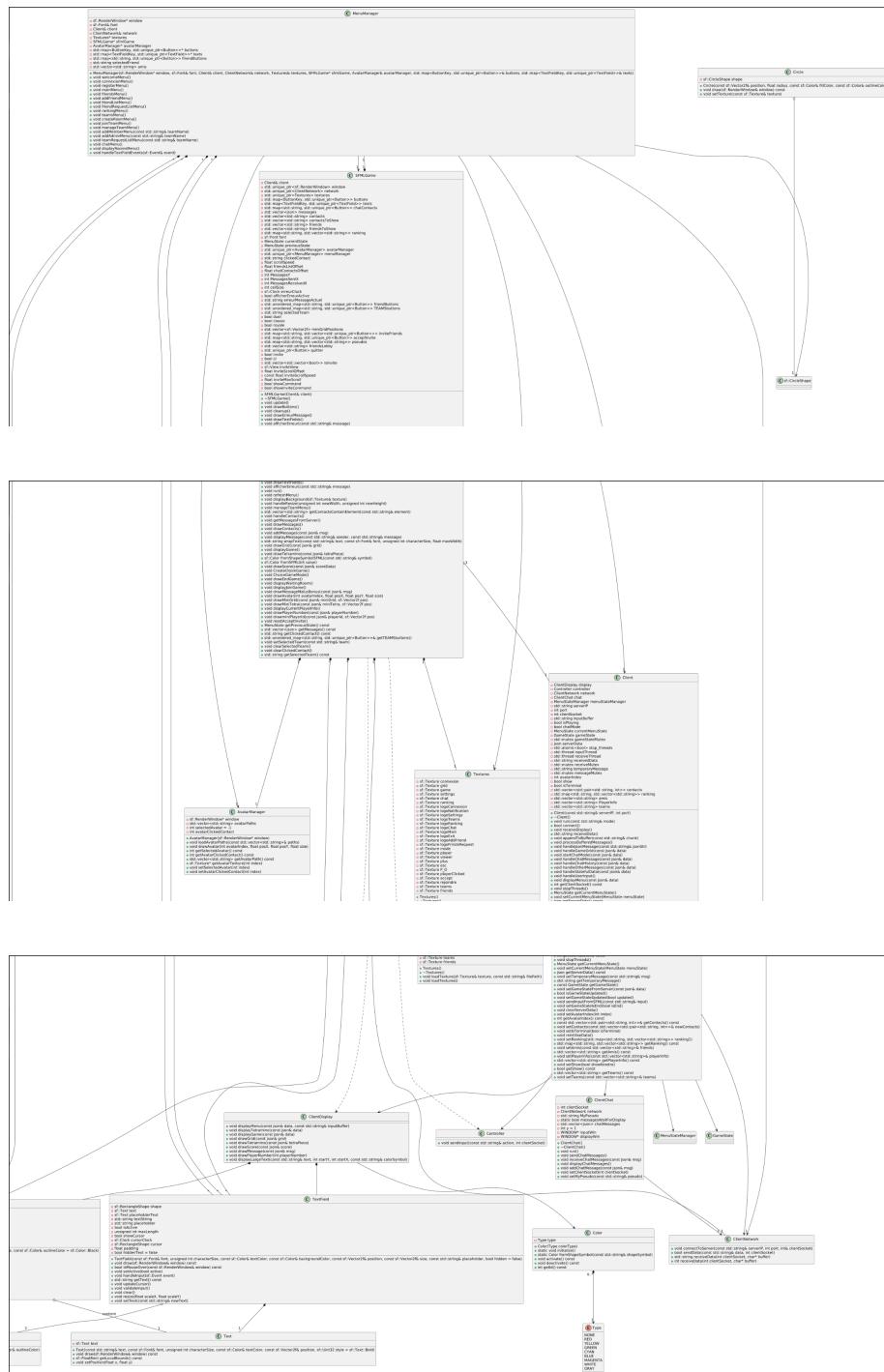


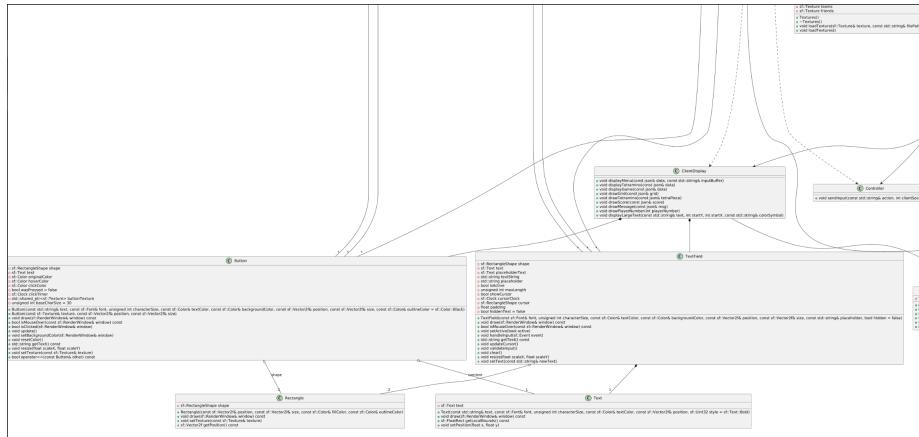


## 6.2 Diagrammes de Classe du serveur/database



### 6.3 Diagrammes de Classe du client





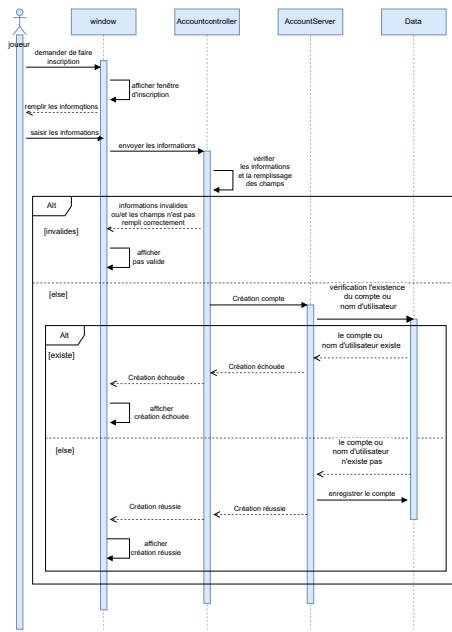
## 6.4 Fonctionnement du système

Dans cette partie nous allons présenter les diagrammes des séquences et d'activités qui illustrent le fonctionnement du système et comment ses composants interagissent entre eux.

### 6.4.1 Crédation d'un compte

Lorsque le joueur veut créer un compte, il va remplir les champs obligatoires (nom d'utilisateur, mot de passe) qui sont montrés par la fenêtre (window). Après que le joueur saisit les informations, le contrôleur va vérifier si tous les champs sont remplis et si le joueur a bien saisi le nom d'utilisateurs et le mot de passe qui respectent le format général imposé par le système. Ensuite si les données sont valides, le serveur va faire une deuxième vérification qui concerne l'existence du compte ou le nom d'utilisateur, si le compte ou le nom d'utilisateur ne sont pas déjà disponibles sur la base de donnée (Data), les données vont être enregistrées. au cas où les données ne sont pas valides ou l'inscription est échouée, une message d'erreur va être affichée sur la fenêtre.

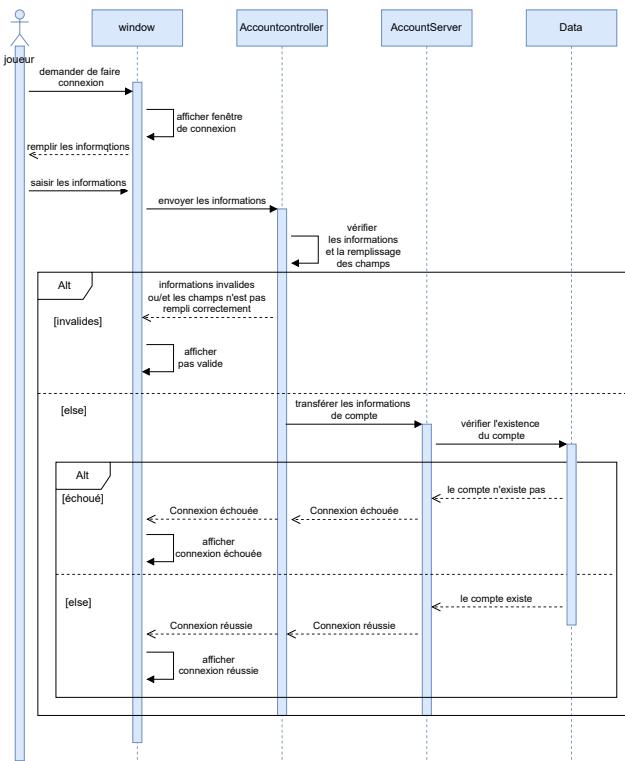
Figure 8: Diagramme de séquence - Inscription



### 6.4.2 Connexion

Pour faire une connexion le système va suivre le même scénario de l'inscription jusqu'à le server reçoit les données entrées par le joueur. Le server va faire une recherche sur la base de données, si le compte existe et le mot de passe est correcte, le joueur peut se connecter. Dans le cas contraire un message d'erreur va être afficher pour informer le joueur que la connexion est échouée.

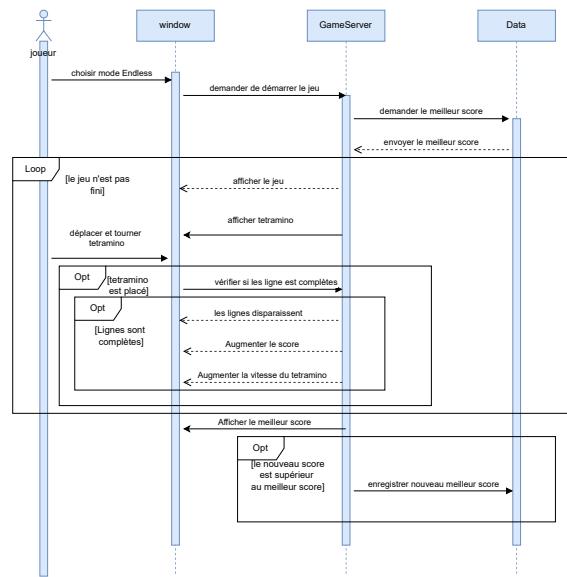
Figure 9: Diagramme de séquence - Connexion



### 6.4.3 Cycle de jeu - mode Endless

Si le joueur décide de créer une partie et il choisit le mode Endless, et puis la fenêtre affiche la grille du jeu avec un score à zéro et avec un tetramino qui apparaît en haut de la grille et qui tombe progressivement. Avant que le tetramino arrive à la base de la grille, le joueur peut le déplacer et le tourner. Après chaque placement du tetramino le GameServer vérifie chaque fois s'il y a une ou plusieurs lignes complètes, si c'est le cas, le score va être augmenter et également la vitesse des tetraminos qui vont arriver après. le même scénario se répète jusqu'à un tetramino touche le haut de la grille et le jeu se termine par l'affichage de le score obtenu dans ce partie et s'il est supérieur au meilleur score, le GameServer l'enregistre dans la base de donnée comme le nouveau meilleur score.

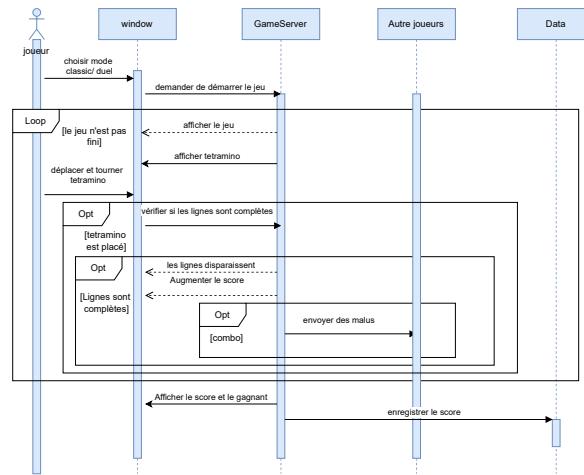
Figure 10: Diagramme de séquence - mode Endless



#### 6.4.4 Cycle de jeu - mode Classic et Duel

Le lancement et l'affichage de jeu en mode Classic ou Duel suivent les mêmes étapes que mode Endless sauf que le GameServer ne récupère pas le meilleur score parce qu'il n'existe pas dans ces modes. Alors il y a plusieurs joueurs qui jouent en concurrence, chaque joueur possède sa propre grille avec ses propres tetraminos, lorsqu'un joueur place un tetramino, le GameServer vérifie s'il y a plus d'une ligne qui est complète, si c'est le cas il va augmenter le score de joueur et si en plus il y a un combo le GameServer va envoyer des malus à un des adversaires. A la fin du jeu, la fenêtre affiche le gagnant avec les scores et le Gameserver enregistre ces données dans la base de données.

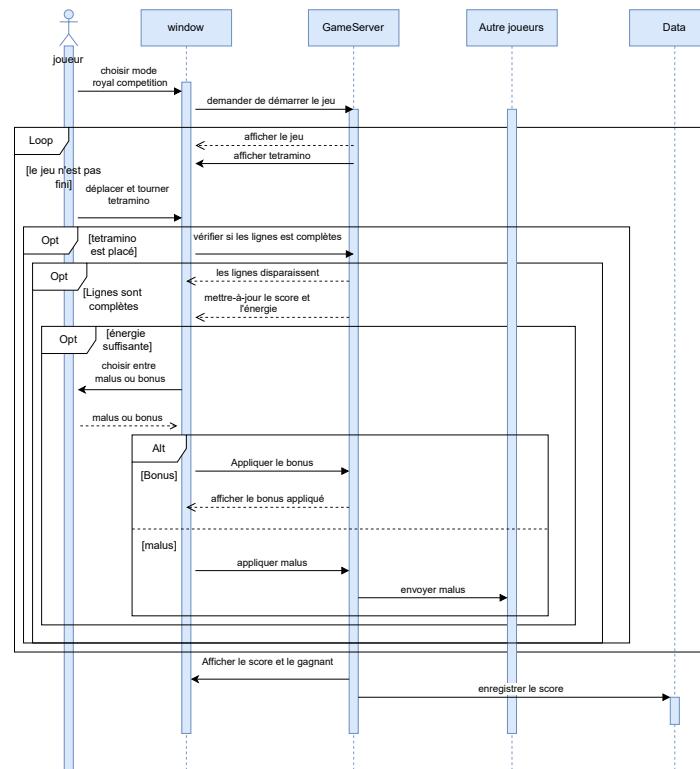
Figure 11: Diagramme de séquence - mode Classic et Duel



#### 6.4.5 Cycle de jeu - mode Royal competition

Ce diagramme est presque le même que du mode classic et duel, La seul différence c'est le moyen d'obtenir des malus. Chaque fois le joueur complète des lignes le GameServer met-à-jour le score et l'énergie de joueur. S'il a recueilli assez d'énergie, la fenêtre demande à lui de choisir entre malus ou bonus, si le joueur choisit de bénéficier d'un bonus le GameServer va l'appliquer, au cas contraire le GameServer va envoyer les malus à l'adversaire choisi par le joueur.

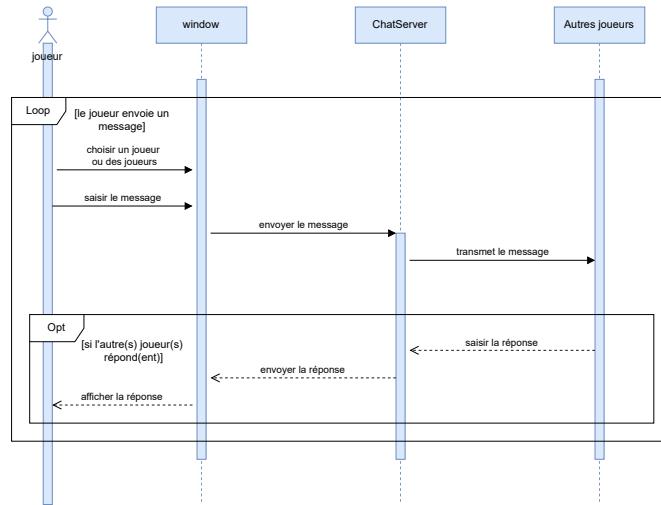
Figure 12: Diagramme de séquence - mode Royal competition



#### 6.4.6 Chat entre les joueurs

Depuis la fenêtre le joueur choisit un ou plusieurs joueurs, et puis il saisit le message. La fenêtre prend ce message et l'envoie au ChatServer qui le transmet après aux destinataires. Si ces derniers répondent, le ChatServer transmet les réponses et la fenêtre les affichent.

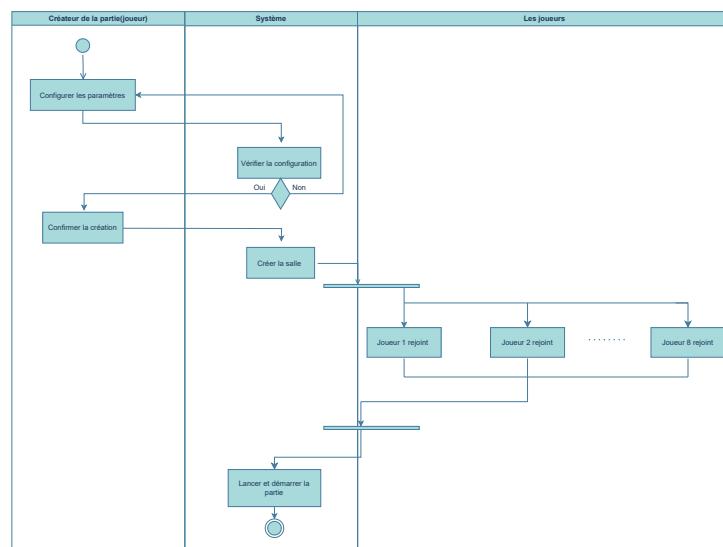
Figure 13: Diagramme de séquence - Chat



#### 6.4.7 Cration d'une partie

Ce diagramme dcrit le processus de cration d'une partie dans Tetris Royale : le crateur configure les paramtres, le systme vrifie leur validit, une salle est cree pour que les joueurs rejoignent et la partie ne dmarre que lorsque tous les joueurs ont rejoint la salle.

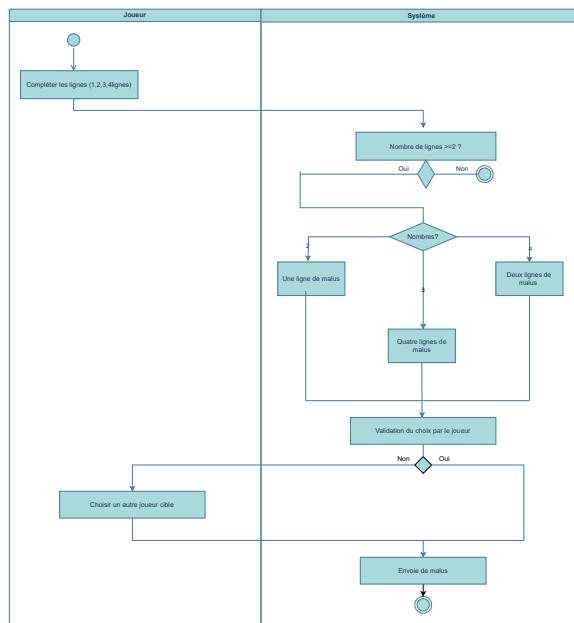
Figure 14: Diagramme d'activit - Cration d'une partie



#### 6.4.8 L'envoi des malus - mode Classic et Duel

Ce diagramme montre le mécanisme de malus dans le mode Classique et le mode Duel du Tetris Royale : lorsque le joueur complète des lignes, le système vérifie leur nombre. Si au moins 2 lignes sont complétées, des malus sont envoyés aux joueurs choisis par le système (1 ligne de malus pour 2 lignes complétées, 2 pour 3, et 4 pour 4). Le joueur valide le choix de joueur cible. Si le choix n'est pas validé, le joueur peut sélectionner une autre cible avant l'envoi du malus par le système.

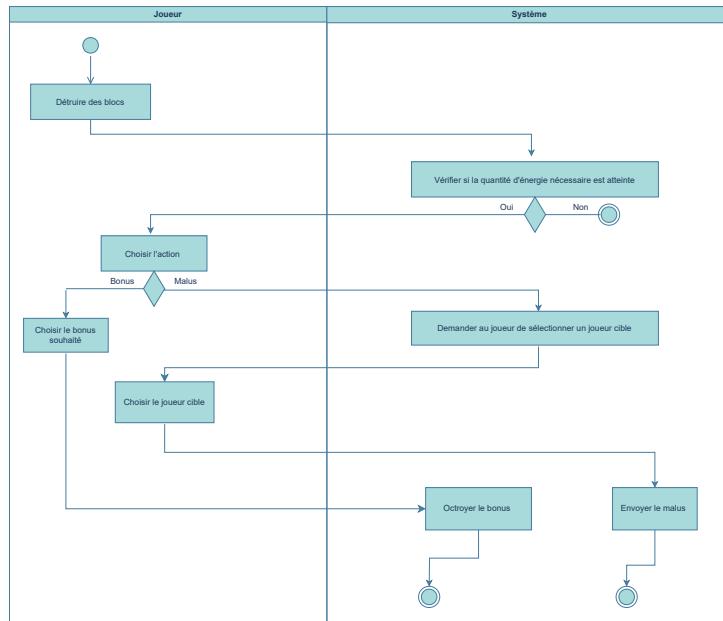
Figure 15: Diagramme d'activité - malus mode Classic et Duel



#### 6.4.9 L'envoi des malus - mode Royale

Ce diagramme décrit le mécanisme de malus dans le mode royale de Tetris Royale. Le joueur détruit des blocs pour accumuler de l'énergie. Si l'énergie requise est atteinte, il choisit une action : octroyer un bonus ou envoyer un malus.

Figure 16: Diagramme d'activité - malus mode Royale



## 7 Historique

Version	Modifications	Auteur	Date
0.1	Introduction	Anthony Van Der Wilt	10/11/2024
0.2	Besoins utilisateurs: Fonctionnels et Non Fonctionnels	Anthony Van Der Wilt	11/11/2024
0.2	Uses Cases	Valentin Dias	11/11/2024
0.3	Diagrammes de séquences	Oumaima Hamdach	21/11/2024
0.4	Besoins système: Fonctionnels	Anthony Van Der Wilt	23/11/2024
0.5	commenter les diagrammes de séquences	Oumaima Hamdach	28/11/2024
0.6	Diagrammes d'activité	Ikram Boutaleb	29/11/2024
0.7	commenter les diagrammes d'activités	Ikram Boutaleb	30/11/2024
0.8	Besoins système : Non Fonctionnels	Valentin Dias	04/12/2024
0.9	Diagramme de classe	Oumaima Hamdach Ikram Boutaleb	11/12/2024