

Document d'architecture logicielle

Version 2.4

Historique des révisions

Date	Version	Description	Auteur
2023-03-14	1.1	Introduction	Zied Kaabi
2023-03-16	1.2	diagramme de paquetage Game	Zied Kaabi
2023-03-19	1.3	diagramme CU globale + indice de Jeu	Zied Kaabi
2023-03-20	1.4	diagramme CU T.L + reprise Video + Historique de Jeu	Zied Kaabi
2023-03-20	1.5	diagramme CU remise des donnees a leur état initial	Zied Kaabi
2023-03-21	1.5	diagramme de déploiement	Ahmed El Shami
2023-03-21	1.6	diagrammes de paquetage	Skander Hannachi
2023-03-21	1.7	diagramme CU constantes de Jeu + Historique de Jeu + message global	Zied Kaabi
2023-03-21	1.8	Paquetage de sélection (client & serveur)	Radwan Rahman
2023-03-21	1.9	Diagrammes de classes des paquetages	Radwan Rahman
2023-04-15	1.10	mise à jour diagramme CU pour sprint 3 (post correction)	Zied Kaabi
2023-04-17	2.1	Corrections apportées aux diag. de paquetages et classes	Radwan Rahman
2023-04-19	2.2	mise à jour des diagrammes de séquences	Zied Kaabi
2023-04-19	2.3	mise à jour des diagrammes de séquences (TL + indice Jeu)	Rayan Fellah
2023-04-20	2.4	mise à jour diagramme de déploiement	Ahmed El Shami

Table des matières

1. Introduction.....	4
2. Vue des cas d'utilisation.....	4
3. Vue des processus.....	13
4. Vue logique.....	23
DIAGRAMMES DE PAQUETAGES ET DE CLASSES.....	23
Paquetages du côté client.....	23
Paquetages du côté serveur.....	33
5. Vue de déploiement.....	39
5.1 Diagramme de déploiement.....	39

Document d'architecture logicielle

1. Introduction

Le document fournit une vue d'ensemble de l'architecture logicielle de notre application web, divisée en quatre sections principales : la vue des cas d'utilisation, la vue des processus, la vue logique et la vue de déploiement.

La première section présente les cas d'utilisation pertinents du modèle, en mettant en évidence les fonctionnalités du Sprint 3 ainsi que certaines fonctionnalités des Sprints 1 et 2. Les diagrammes de cas d'utilisation décrivent les interactions entre les clients et le système pour chaque fonctionnalité.

La deuxième section décrit les processus significatifs du système sous forme d'un diagramme de séquence. Elle met l'accent sur les fonctionnalités du Sprint 3 en identifiant les principaux processus. Les aspects pertinents des Sprints 1 et 2 seront inclus dans cette section si nécessaire.

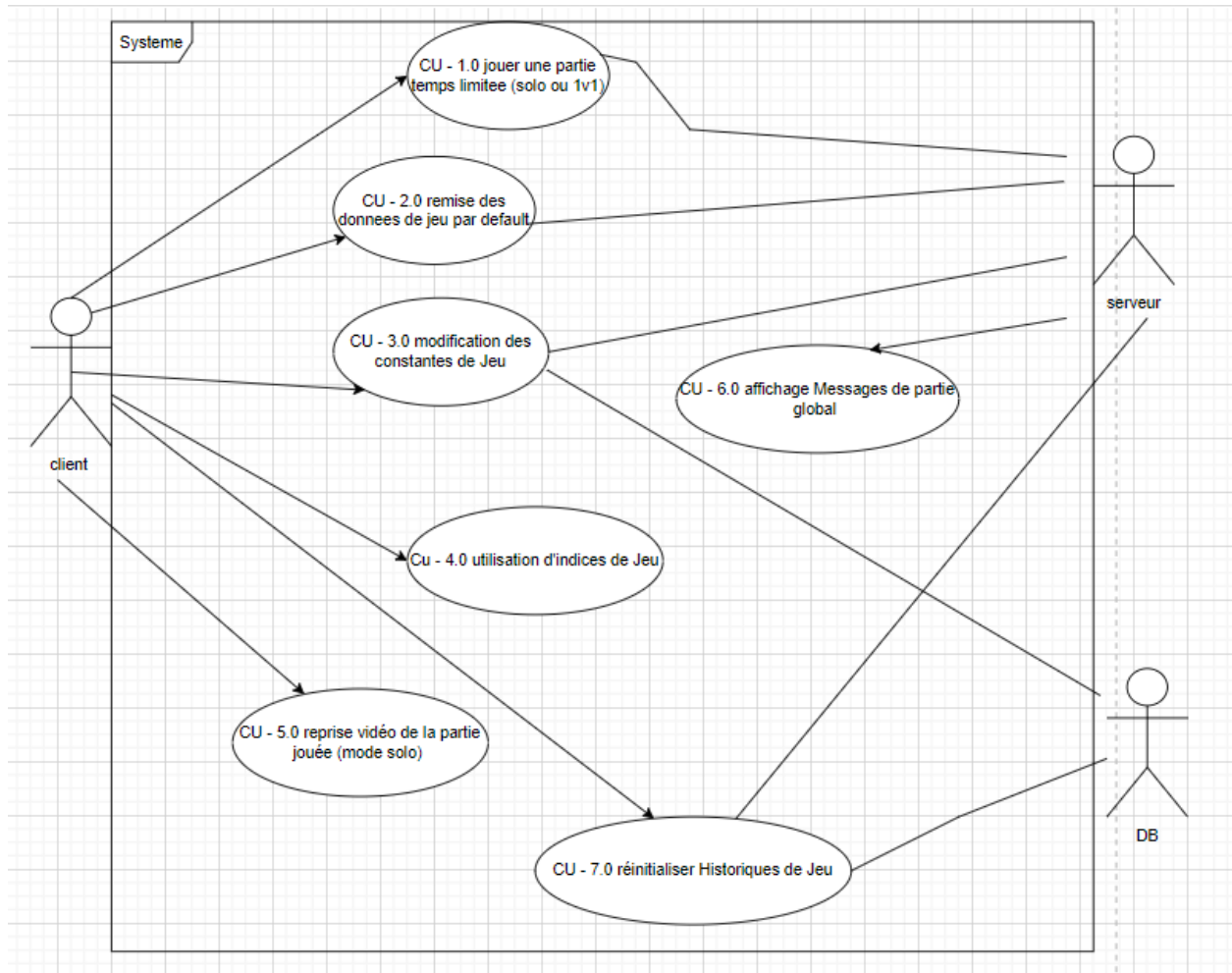
La troisième section montre les parties architecturalement significatives de notre modèle de conception, en utilisant des diagrammes de paquetages et de classes. Un tableau répertoriant tous les paquetages jugés pertinents dans notre modèle de conception architecturale est également inclus.

Enfin, la dernière section décrit la configuration de déploiement des différentes parties du système, en identifiant les nœuds physiques et les interconnexions utilisées, y compris les protocoles de communication utilisés.

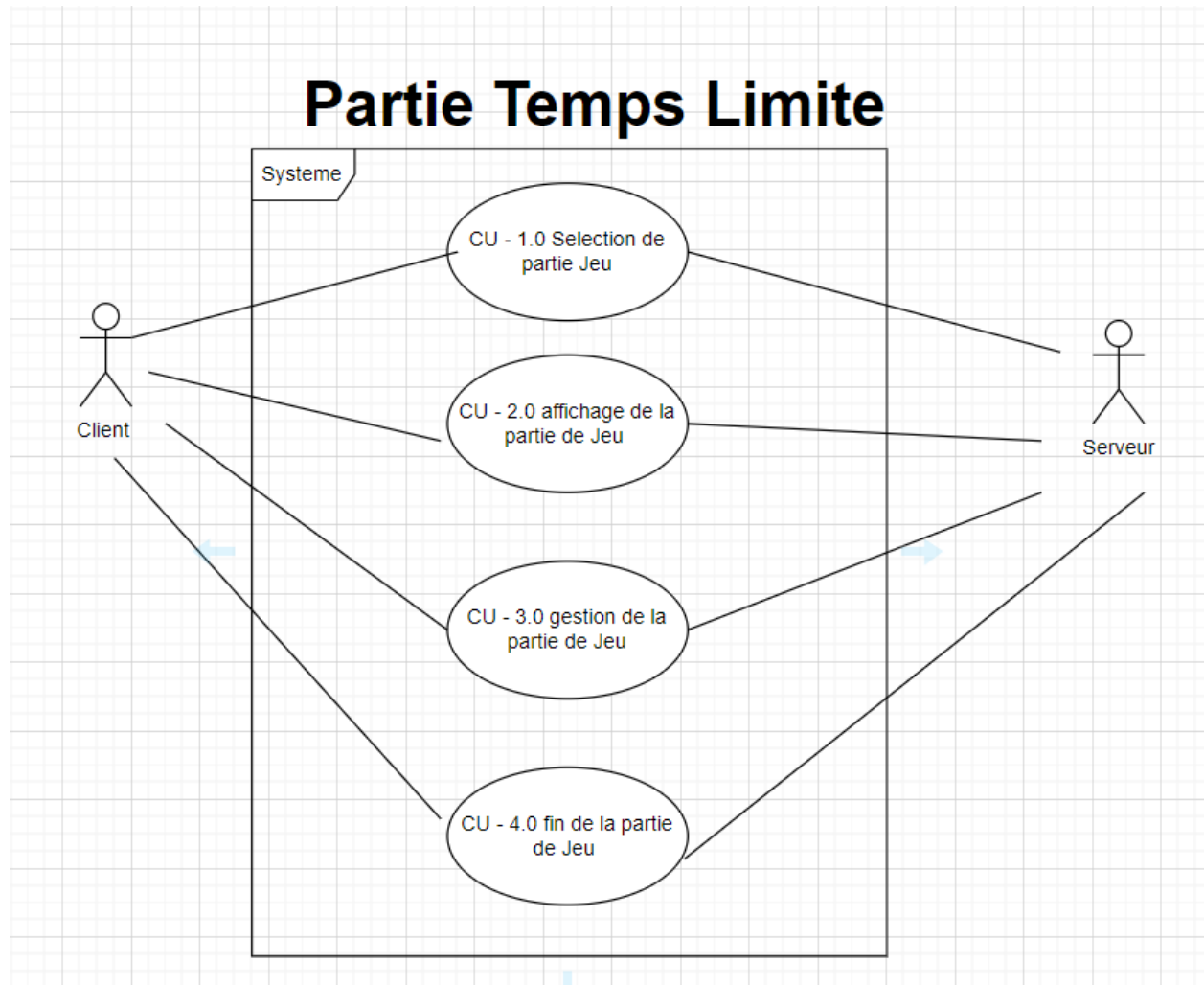
2. Vue des cas d'utilisation

Nous avons décidé de faire un diagramme de CU principale englobant tous les Cas d'Utilisation concernant les fonctionnalités du sprint 3 et des diagrammes pour chacun des cas d'utilisation présent dans le diagramme principale

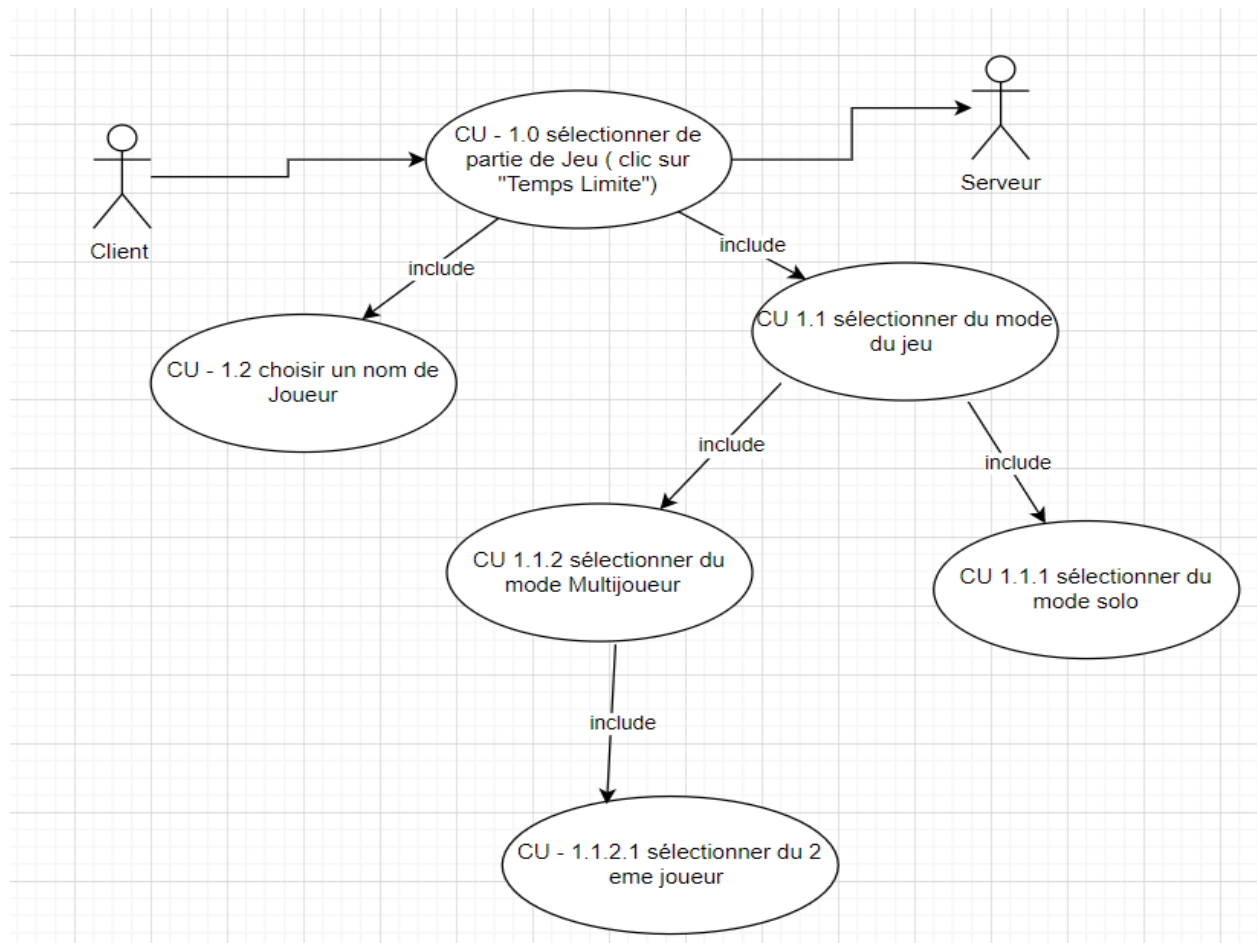
Diagramme principal (sprint 3) :



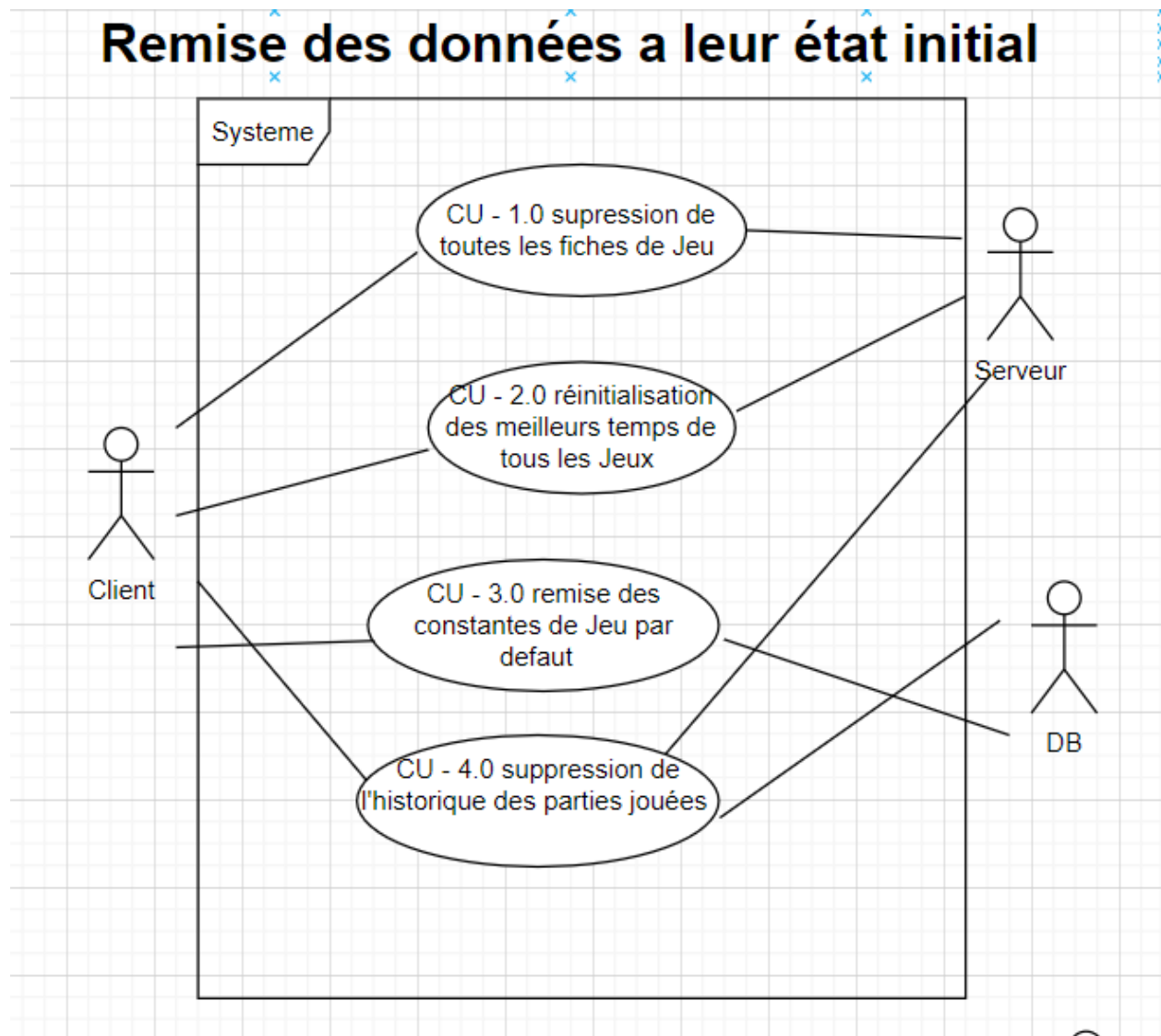
1-Jouer une partie T.L :



CU- 1.0 selection de partie de Jeu :

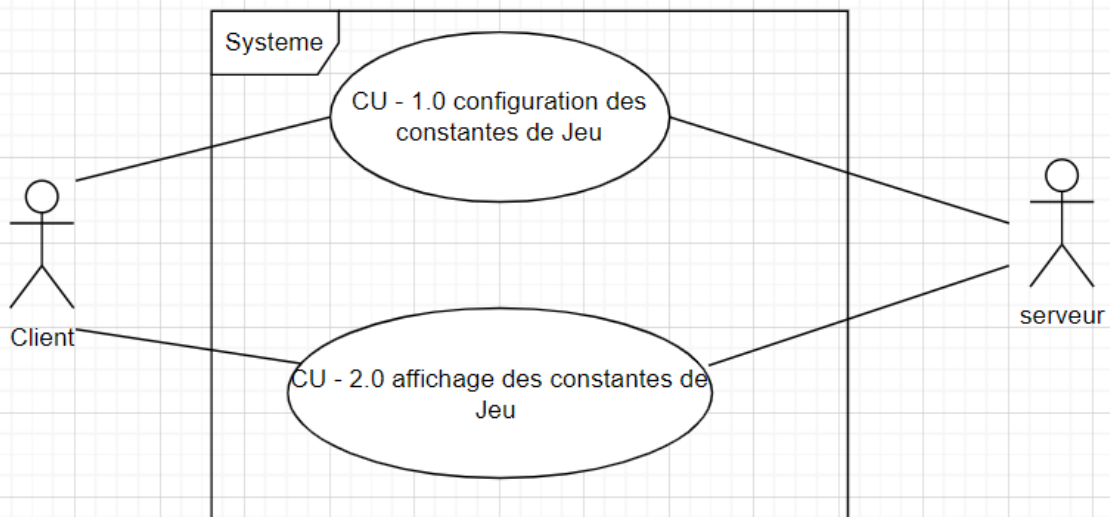


2- Remise des donnees a leur état initial :

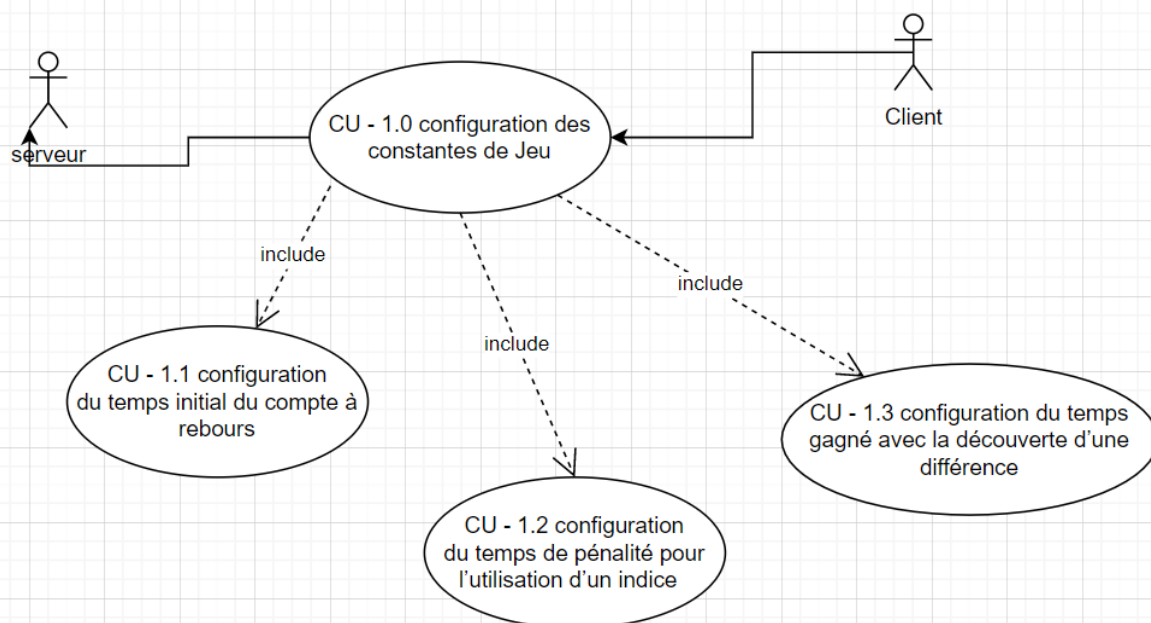


3 - Modification des constantes de Jeu :

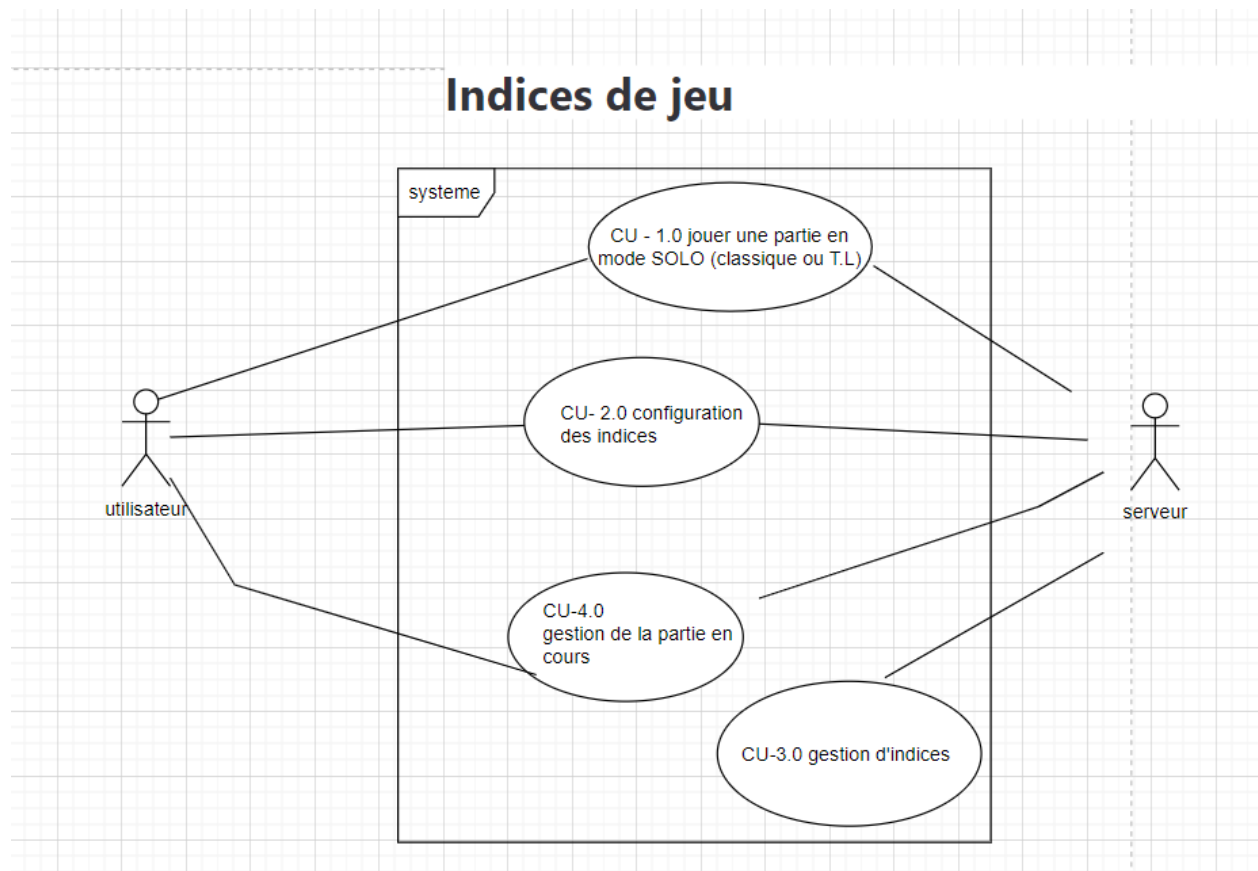
Modification des constantes de Jeu



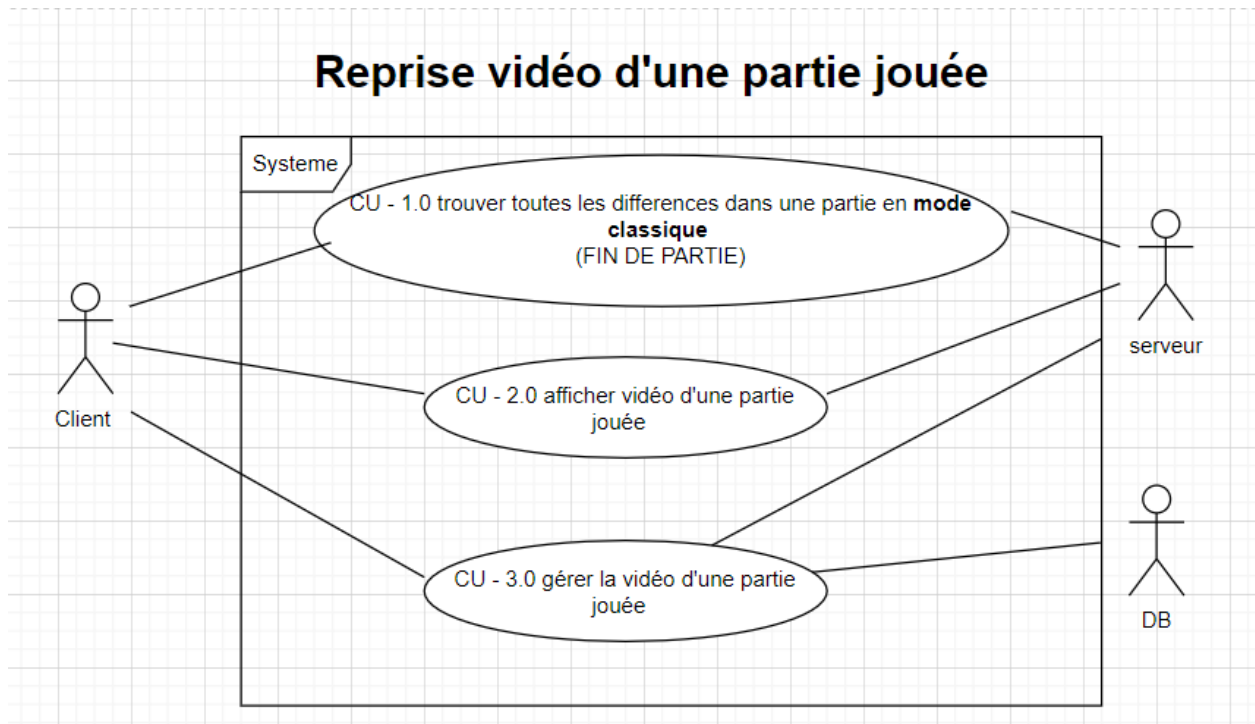
CU - 1.0 configuration des constantes de Jeu :



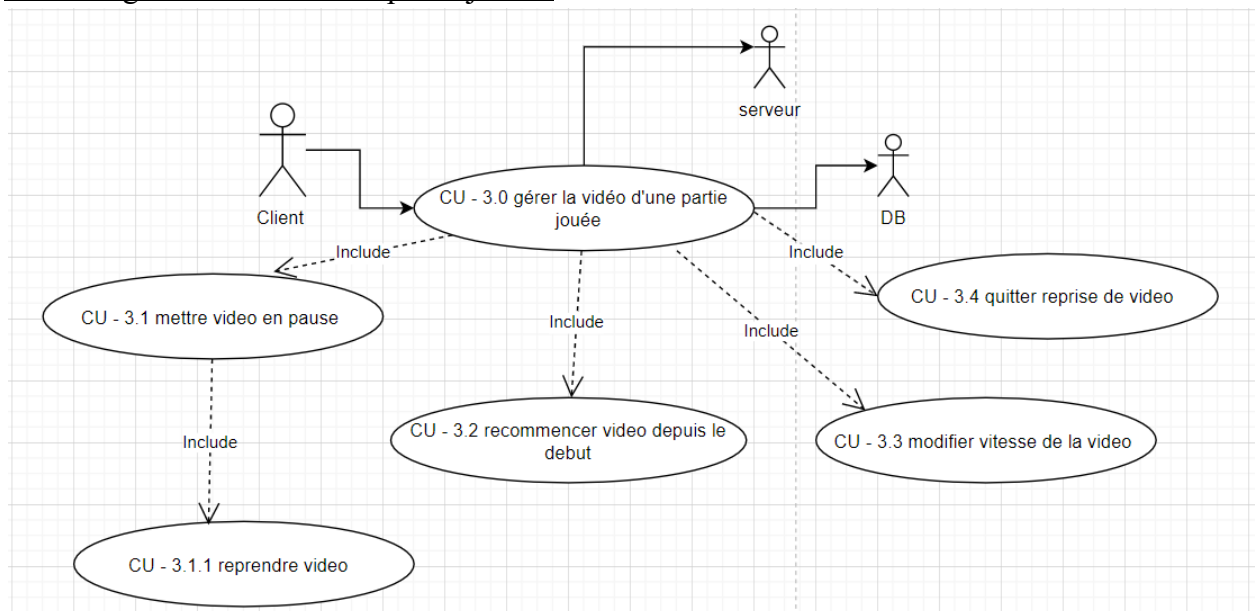
4 - utilisation d'indice de Jeu



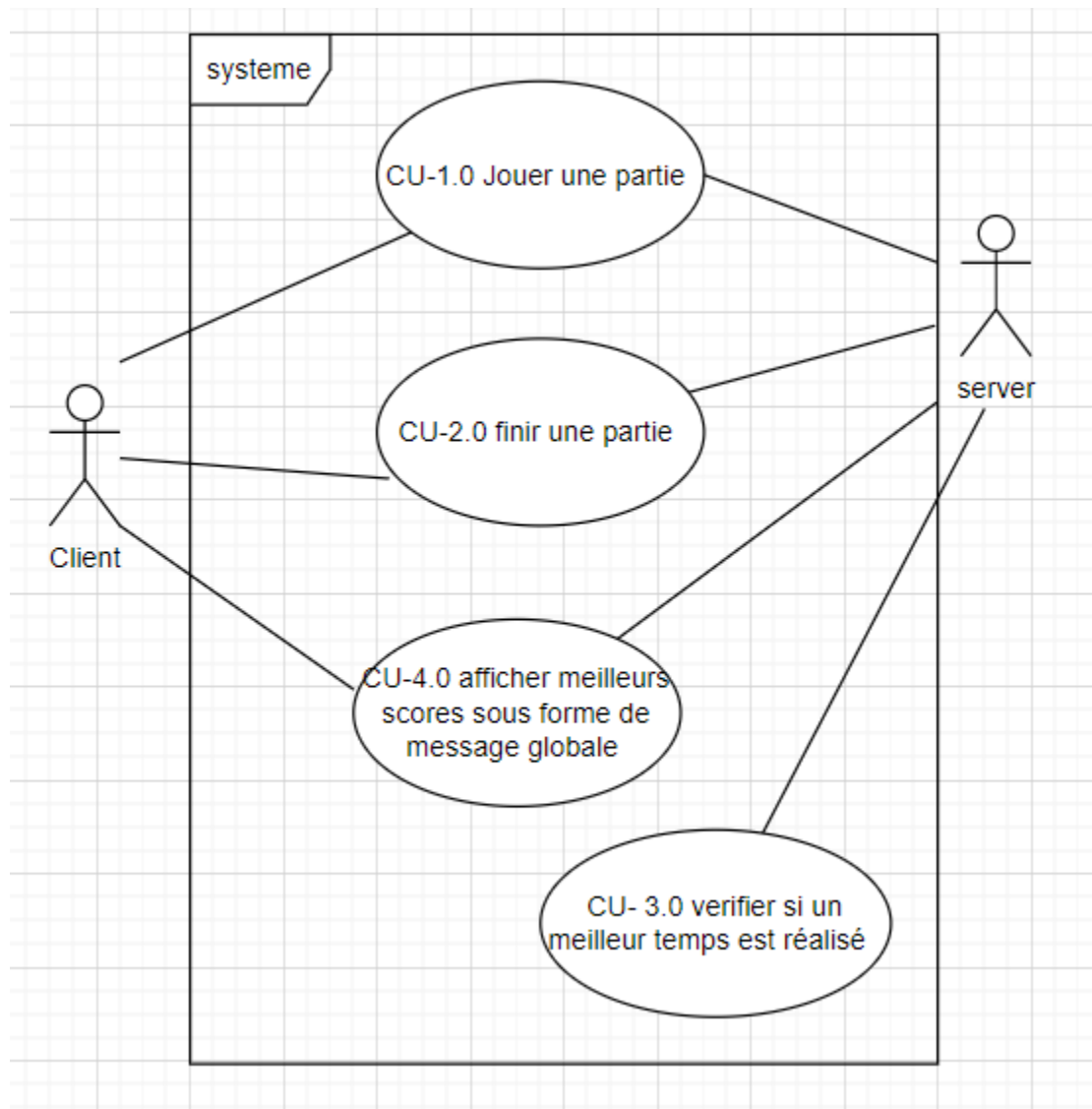
5-Reprise vidéo d'une partie jouée :



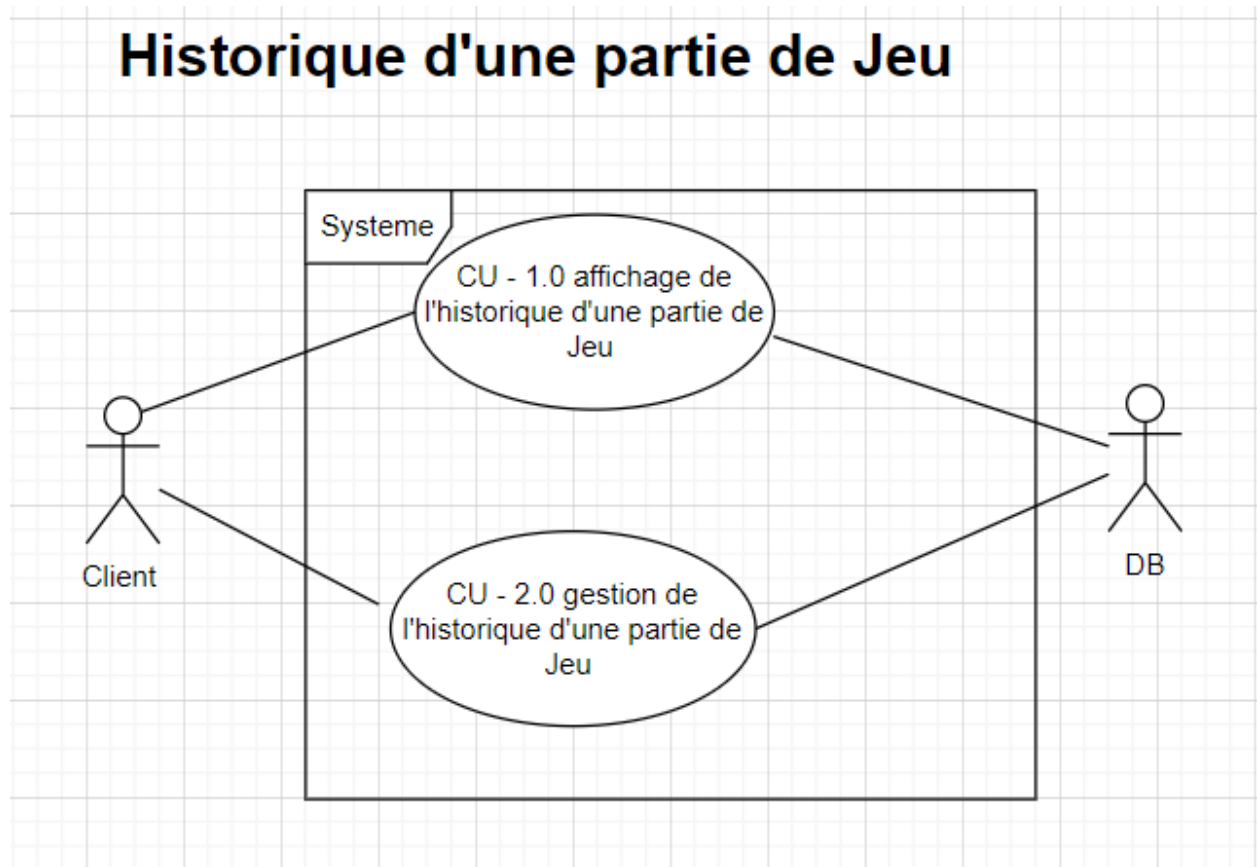
CU - 3.0 gérer la vidéo d'une partie jouée :



6 - affichage de message globale d'une partie :



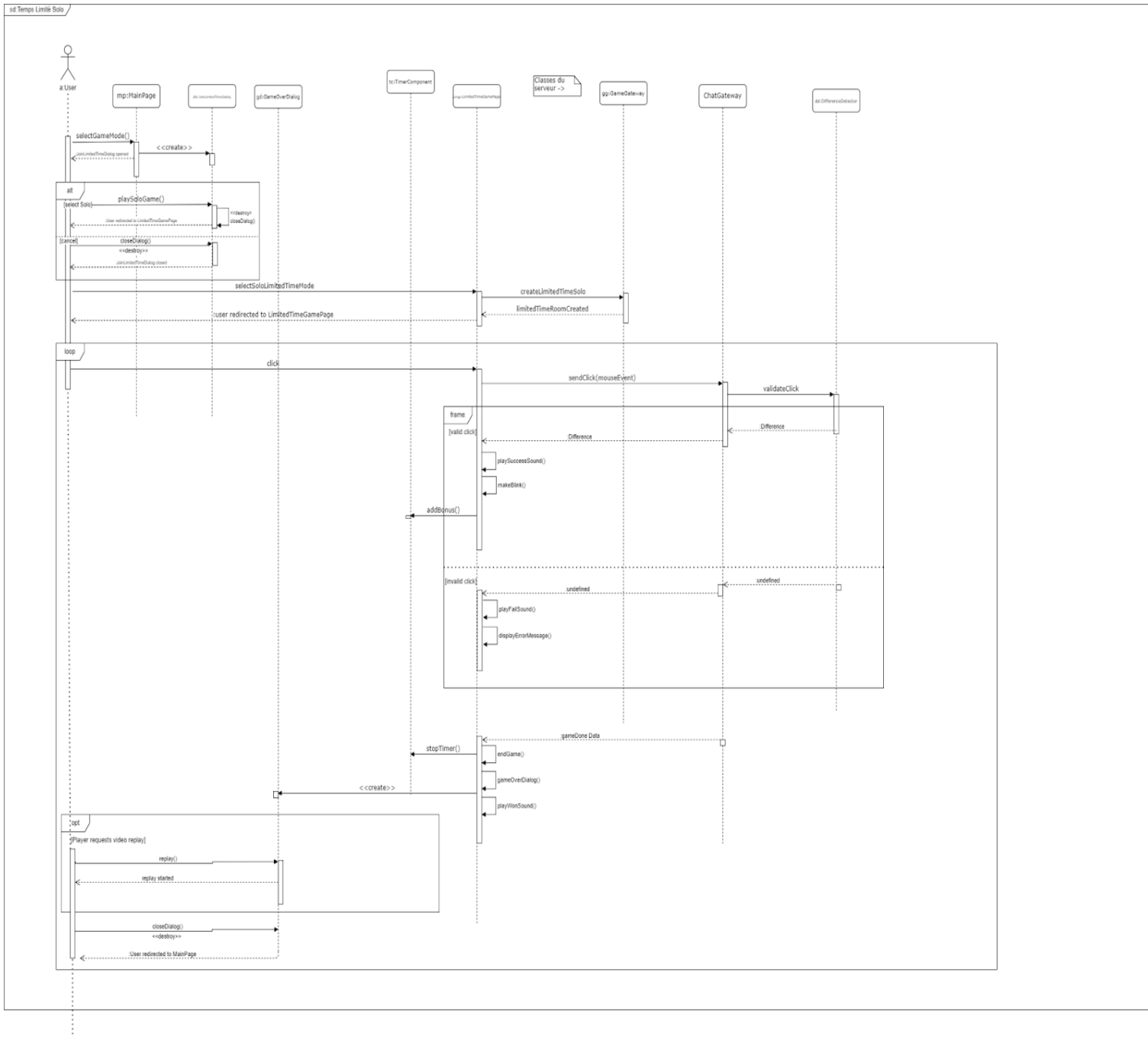
7- Historique d'une partie de Jeu :



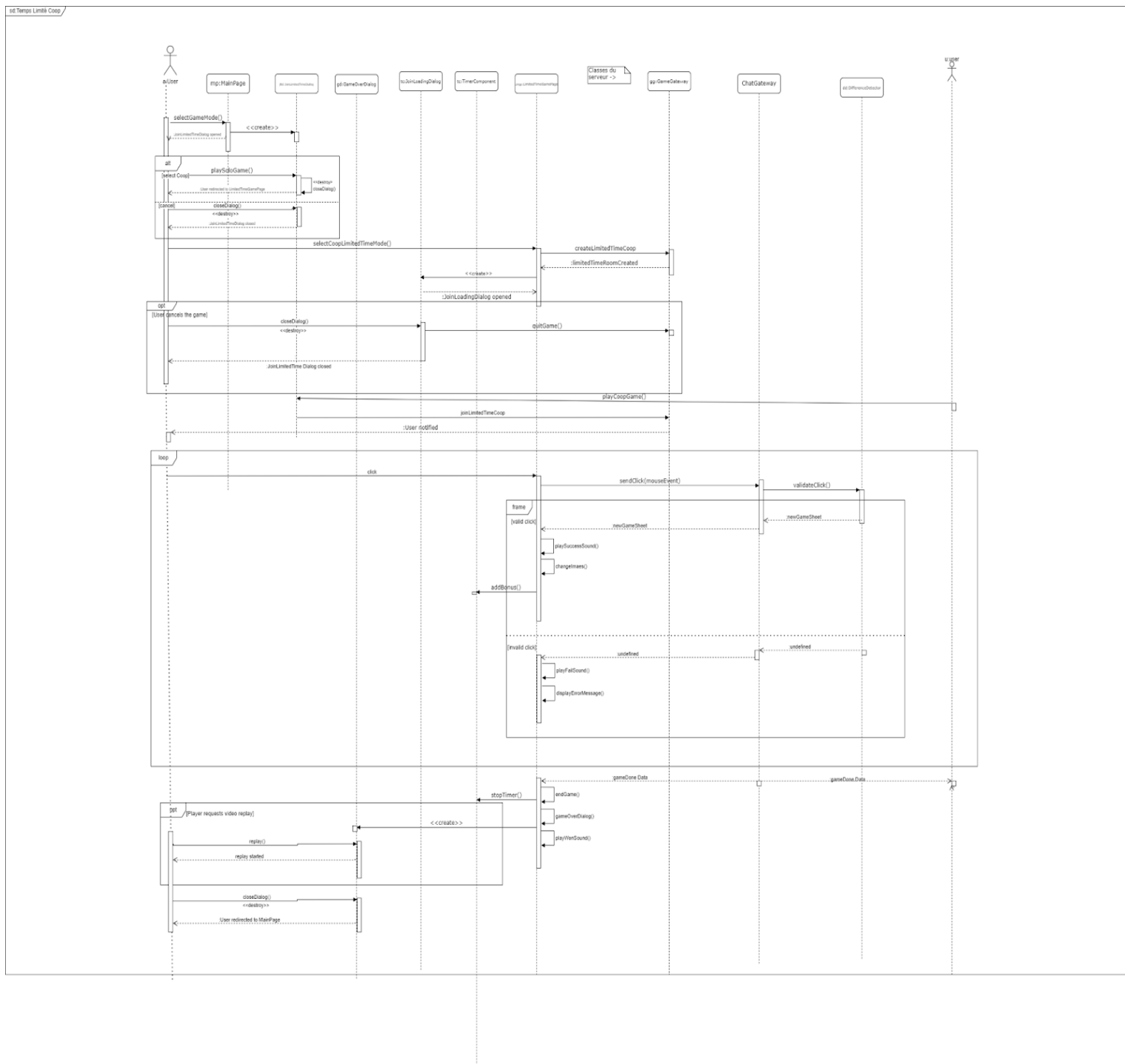
3. Vue des processus

1) Mode TL :

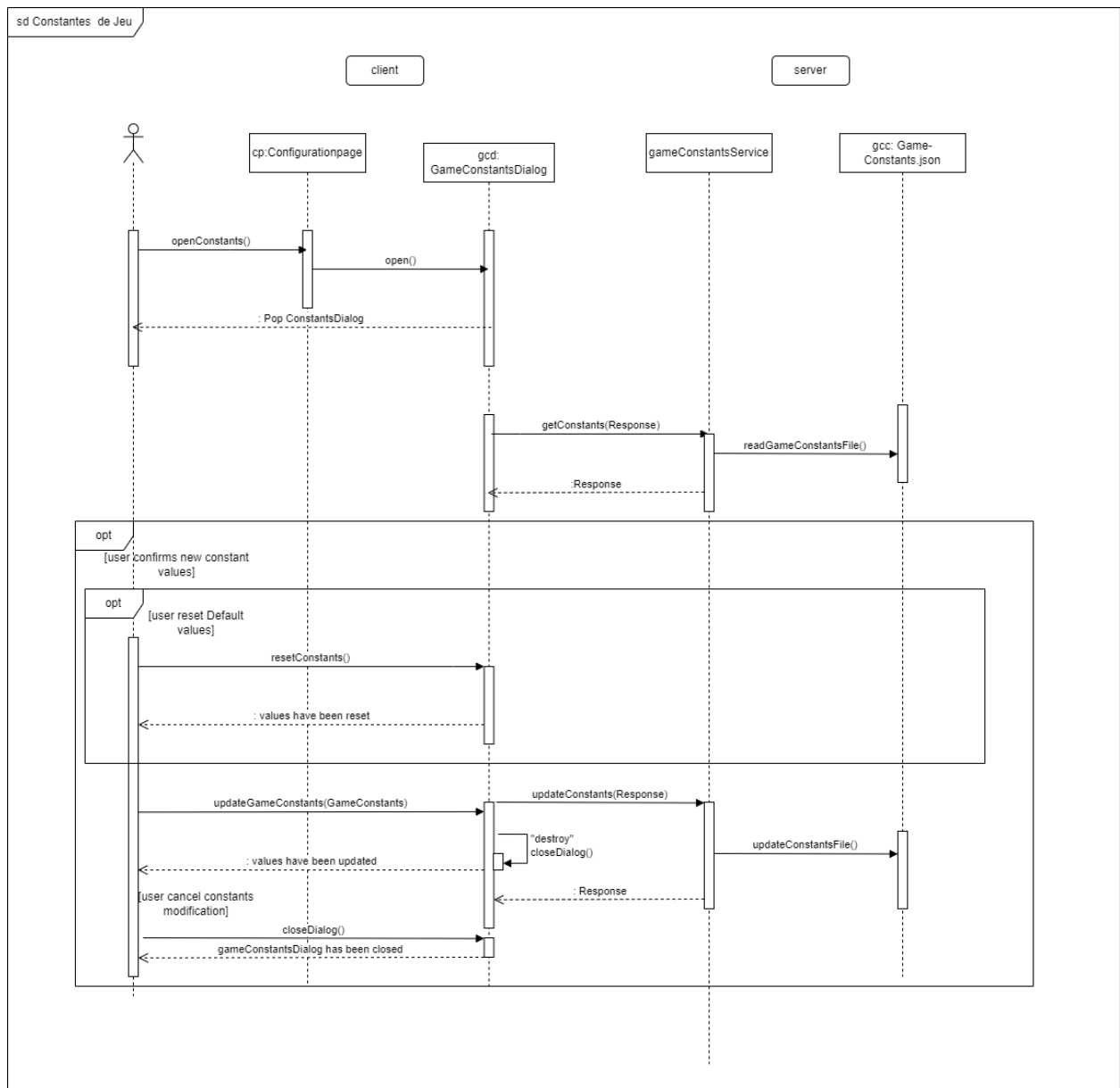
1.1) Mode TL Solo :



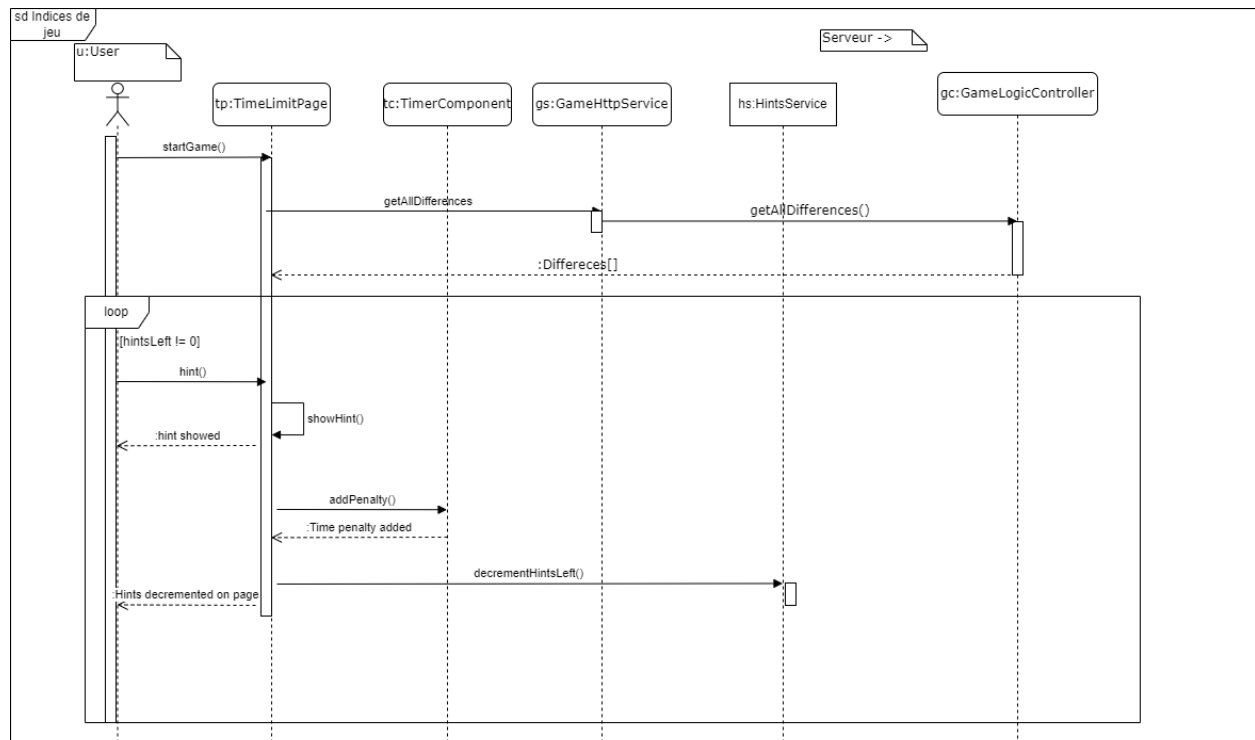
1.2) Mode TL Coop :



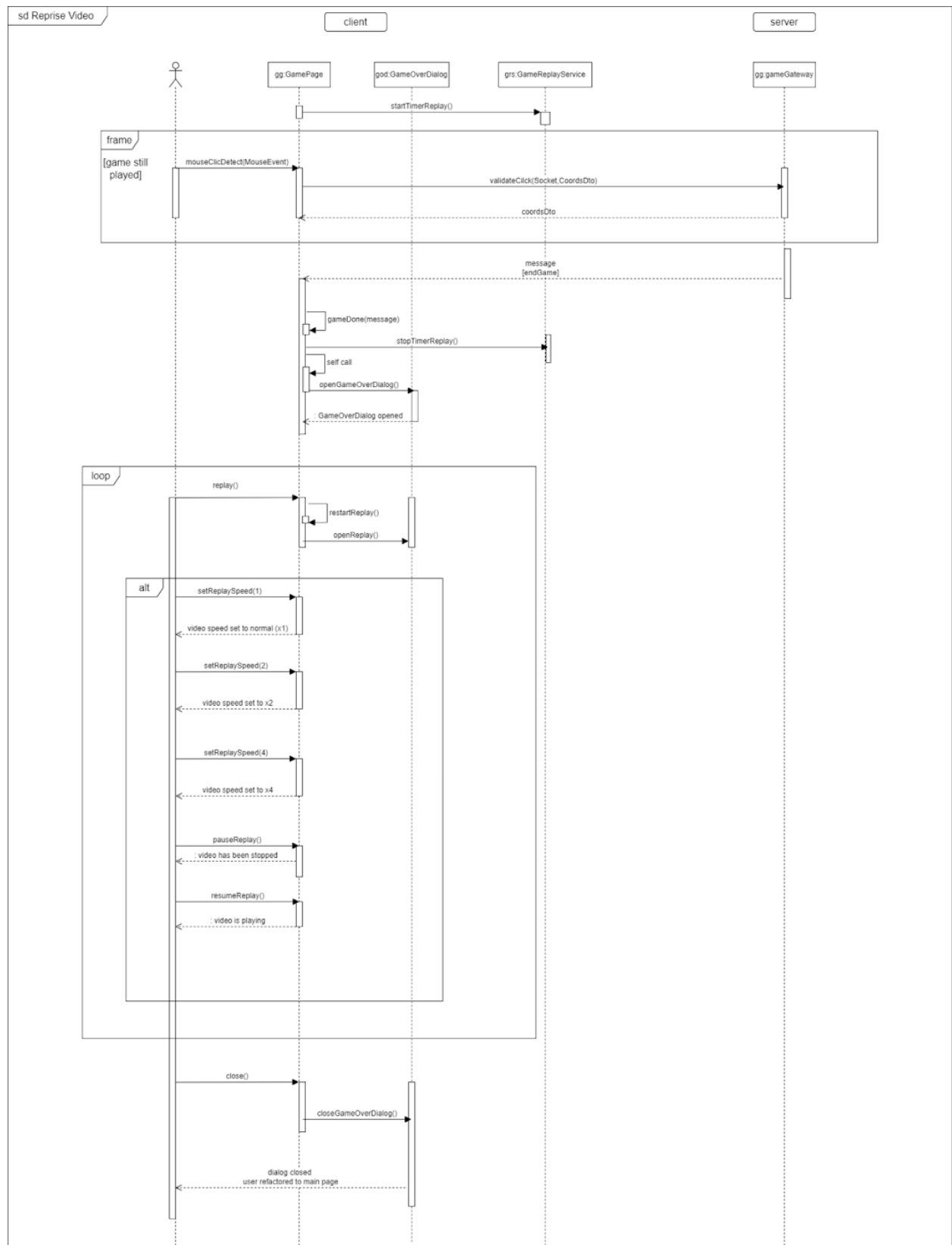
3) Constantes de Jeu :



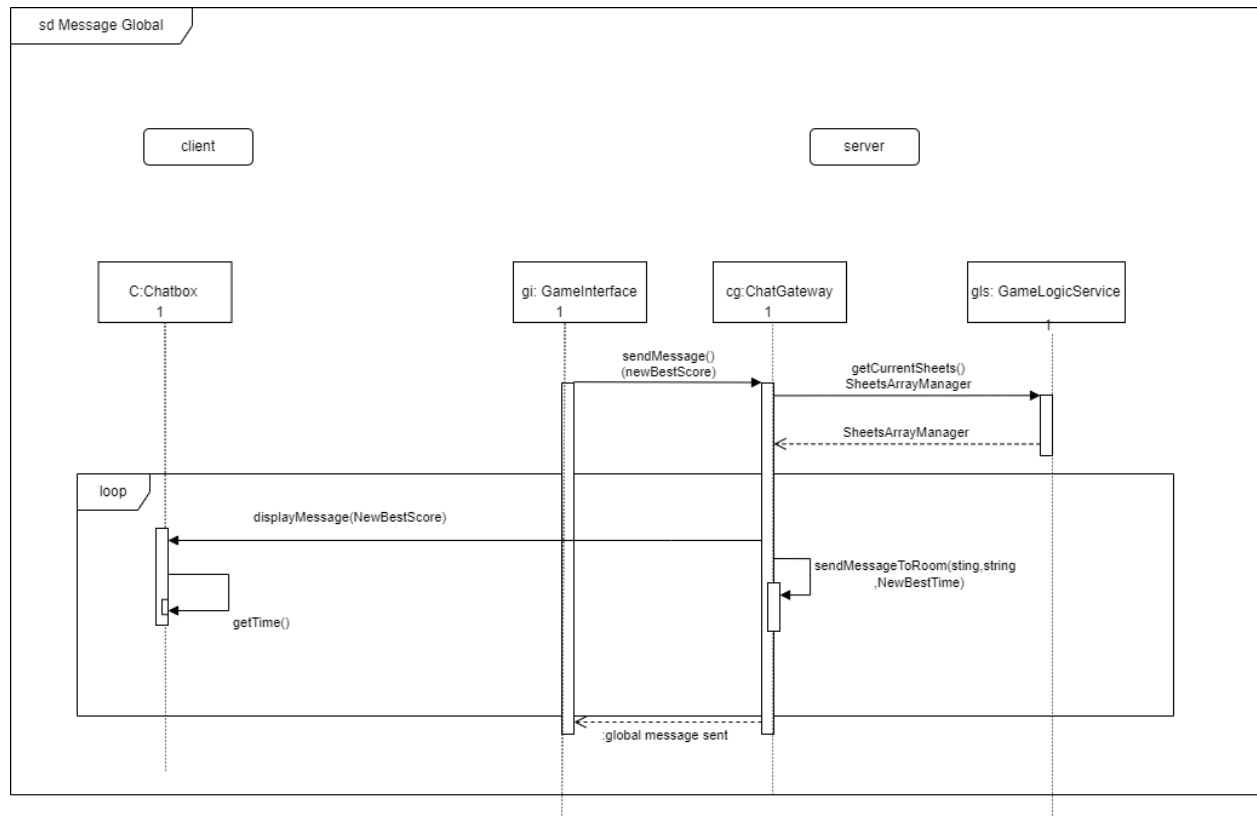
4) Utilisation d'indice de Jeu :



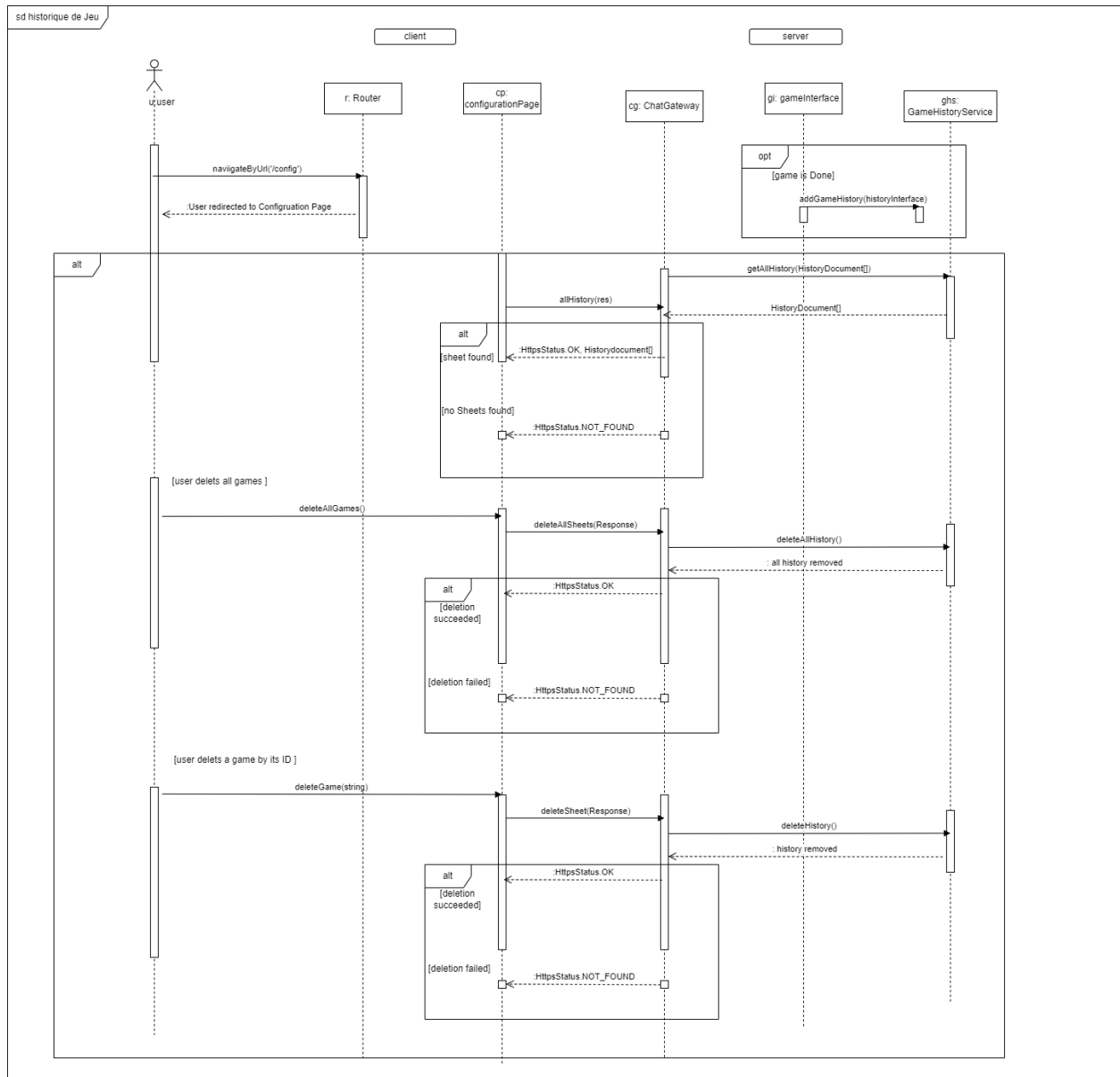
5) Reprise Video :



6) affichage de message global :

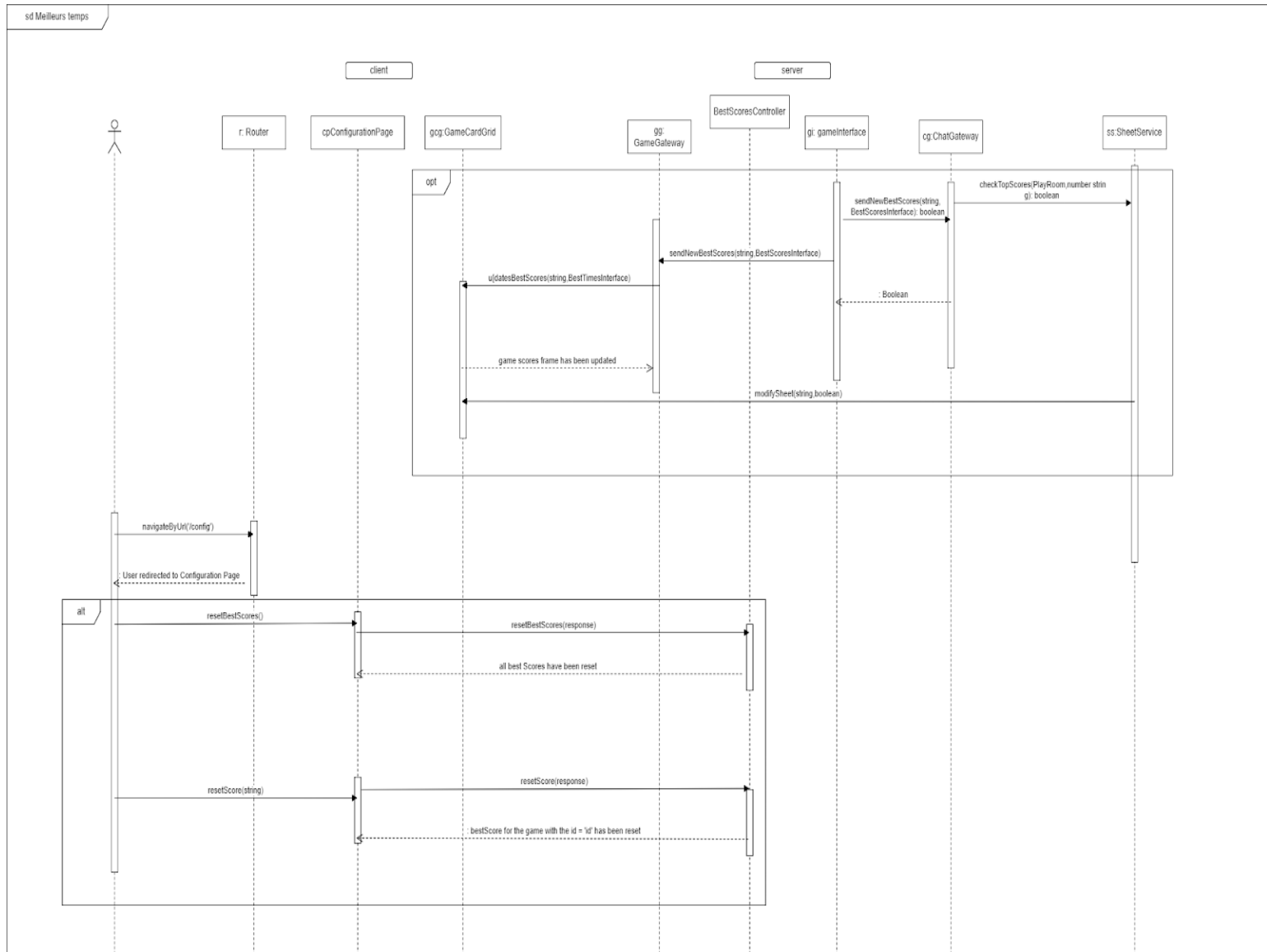


7) Historique d'une Partie de jeu :



Remise des données à leur état initial : cette fonctionnalité a été traité dans tous les diagrammes. On en traitera celui des meilleurs temps (sprint 2) qui utilise la remise des données.

8) Meilleurs temps :



4. Vue logique

DIAGRAMMES DE PAQUETAGES ET DE CLASSES

Paquetages du côté client

paquetage de jeu :

GamePlay
ce paquetage se charge de la vue et la logique pour les différents modes de jeu, il communique avec le serveur via le service SocketClientService afin de recevoir les événements liés au jeu notamment le nombre de différences trouvées et leurs rétroaction visuelle et sonore ainsi que les clics émis par l' utilisateur et leur validité

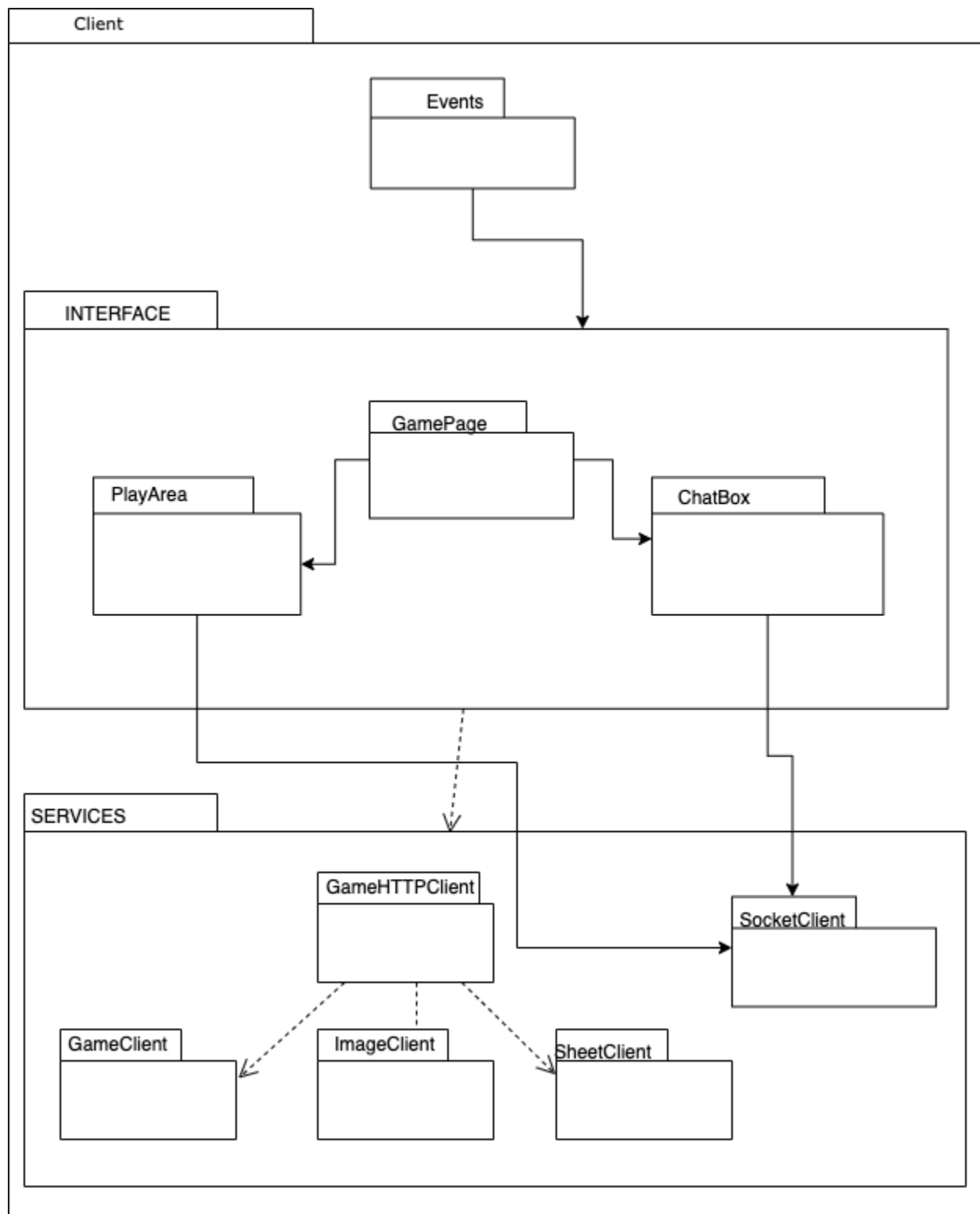
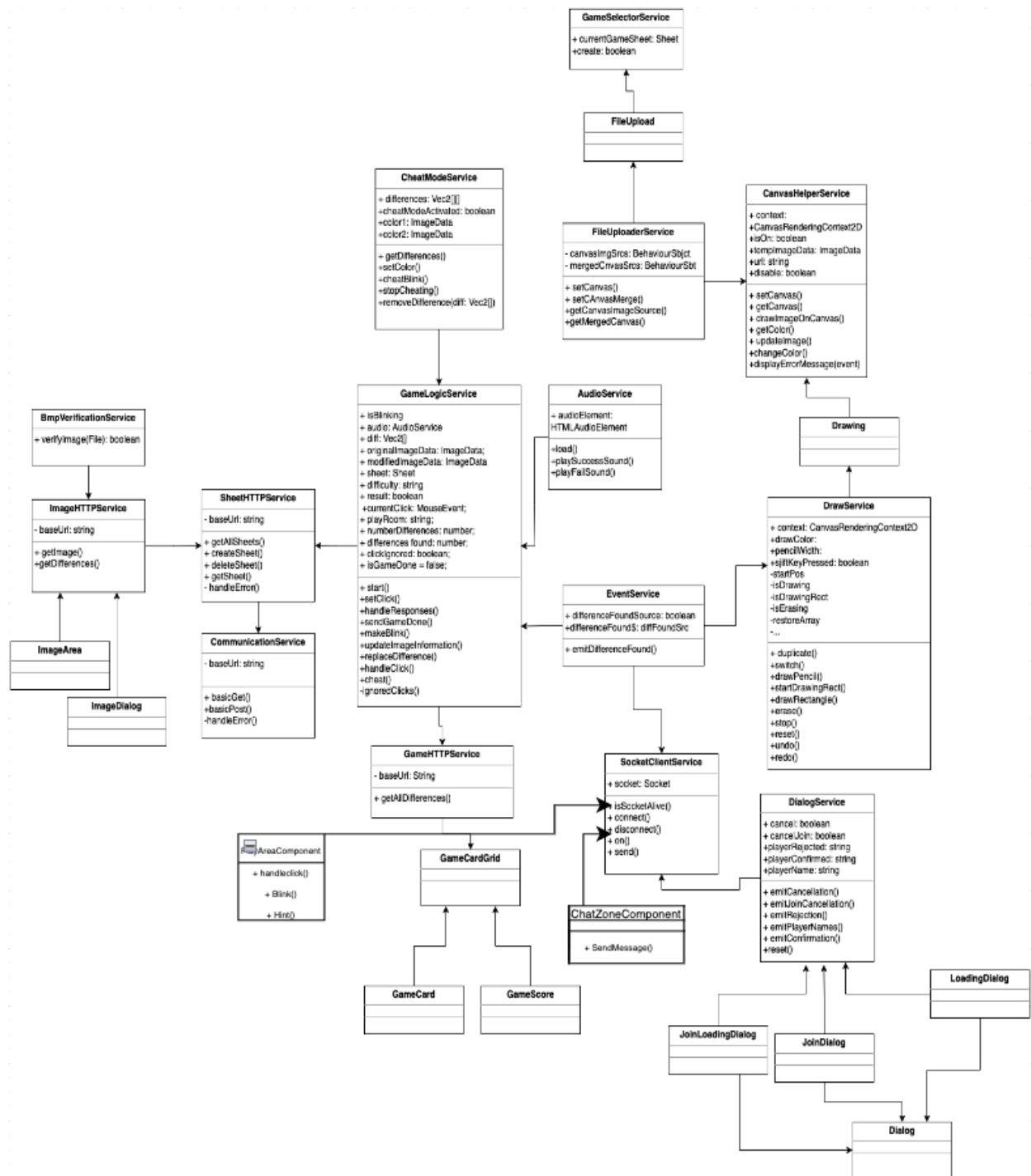


Figure 1.0 - Diagramme de Paquetage Client GamePlay



paquetage de configuration :

GameConfiguration

ce paquetage se charge de la vue et la logique pour la configuration du jeu, que ça soit pour modifier ou créer une nouvelle fiche de jeu. Il communique avec le sheets http pour qu'il puisse à son tour, communiquer avec le serveur pour appliquer la configuration demandée.
--

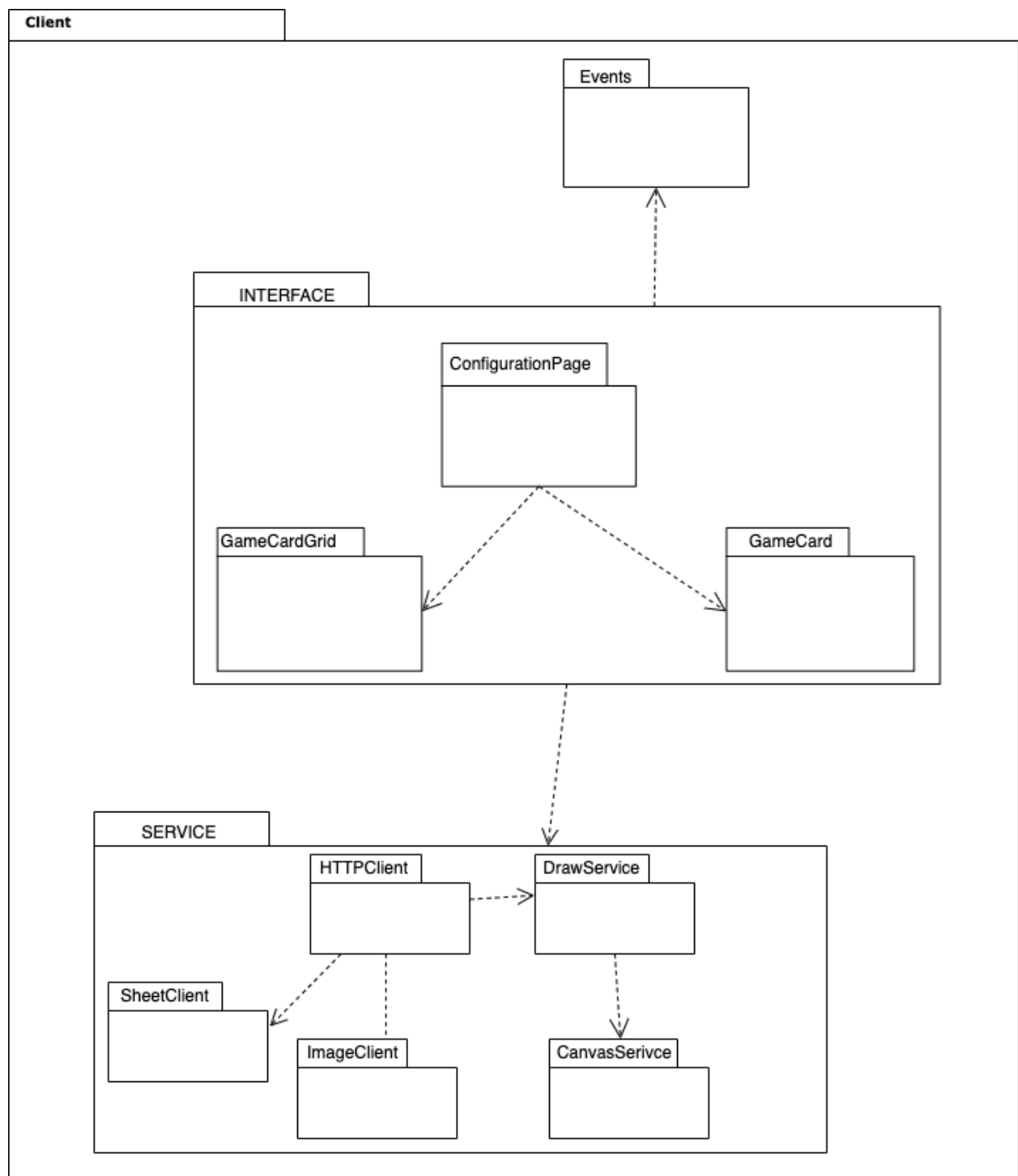
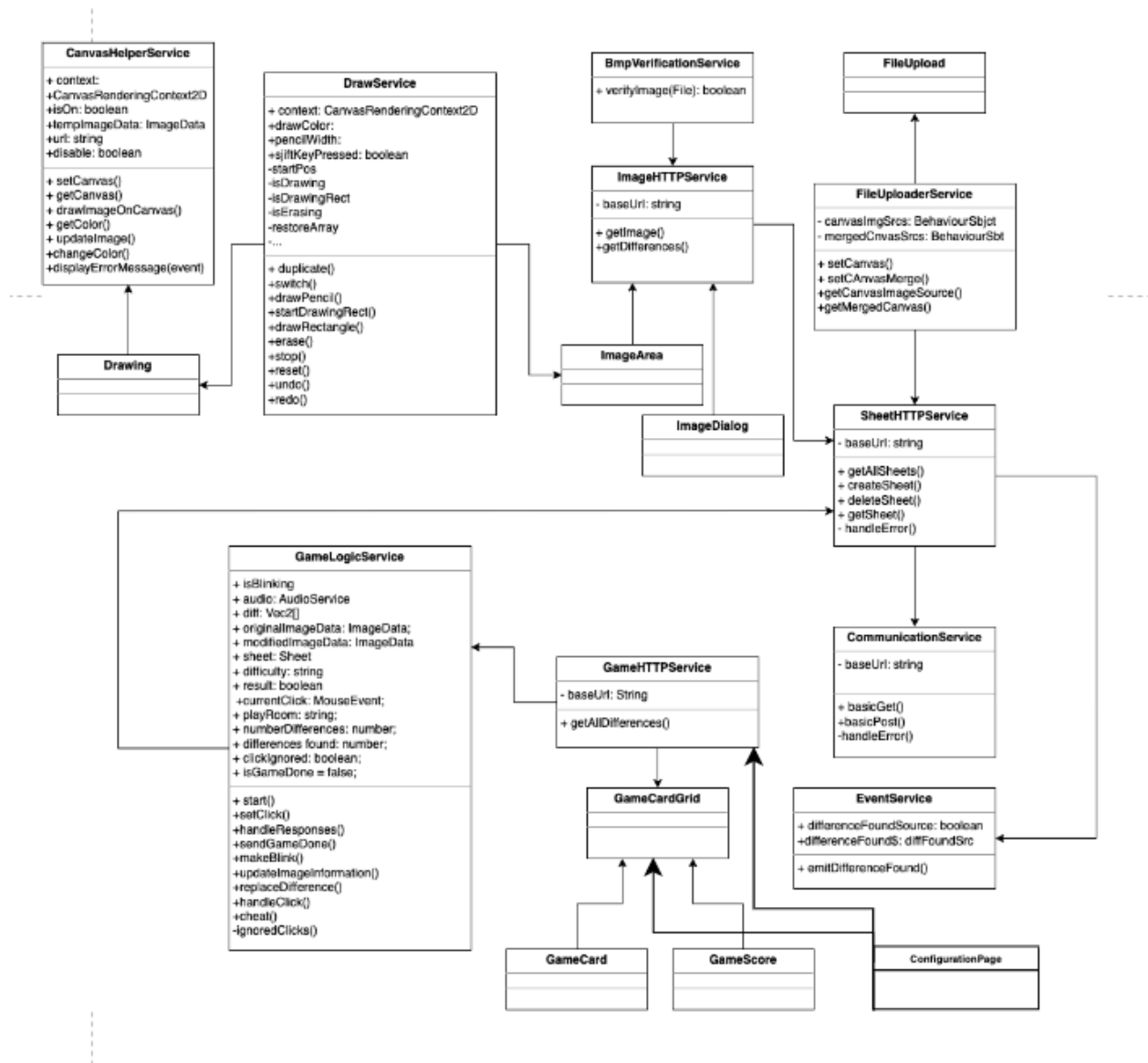


Figure 2.0 - Diagramme de paquetage Client Configuration



paquetage de selection :

GameSelection
<p>Ce paquetage est responsable de la sélection de la partie de jeu et de la redirection du joueur vers la page de jeu appropriée en fonction du type de partie classique choisi par le joueur (1v1 ou solo). Il envoie également les informations nécessaires à la vue de jeu pour s'adapter correctement au mode de jeu choisi et pour afficher les informations requises. Cette communication avec le serveur implique l'envoi du mode de jeu choisi, de l'Id du jeu et du nom du joueur. En outre, ce module gère également tous les aspects liés à l'attente des joueurs pour une partie en 1v1 mode coopératif .</p>

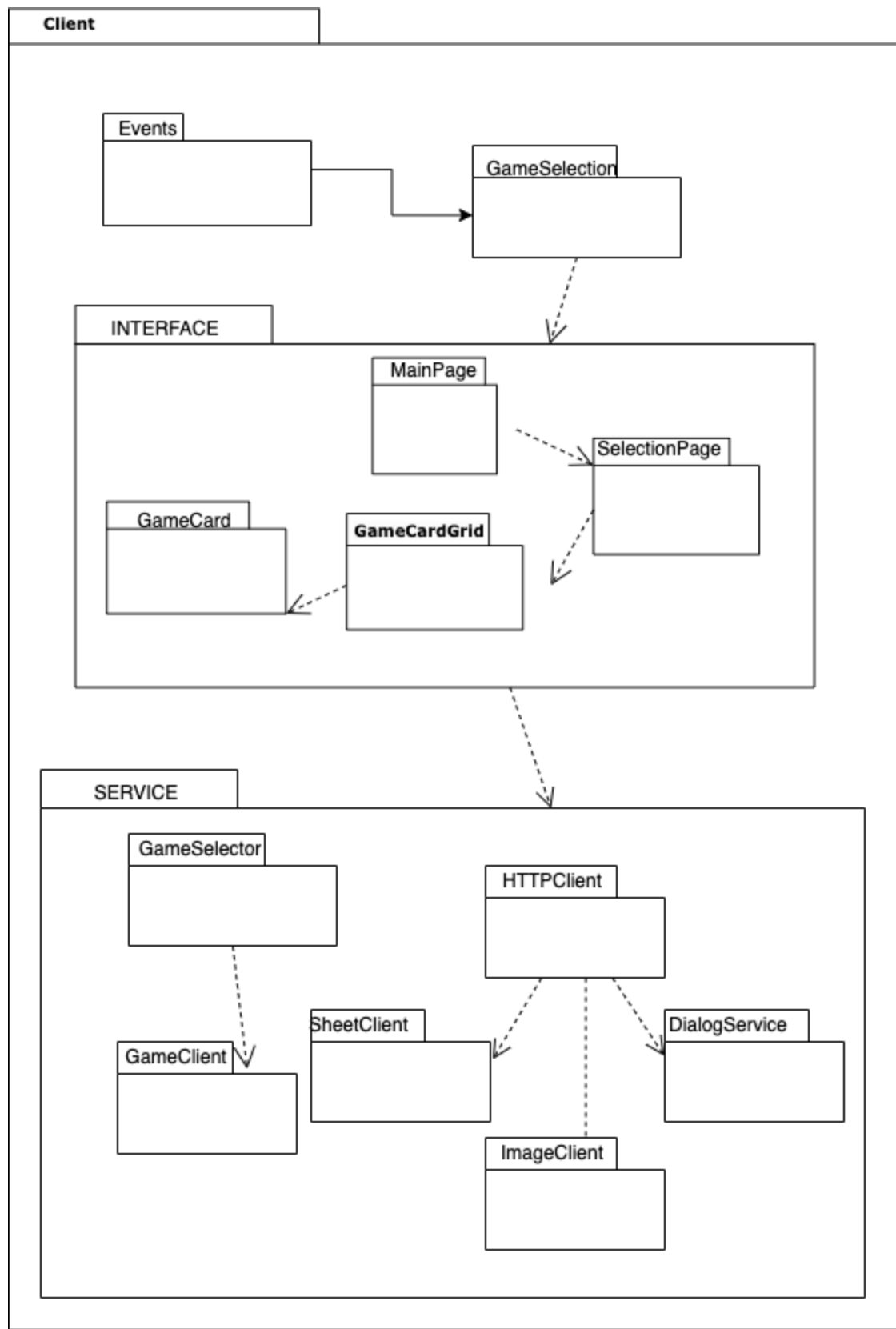
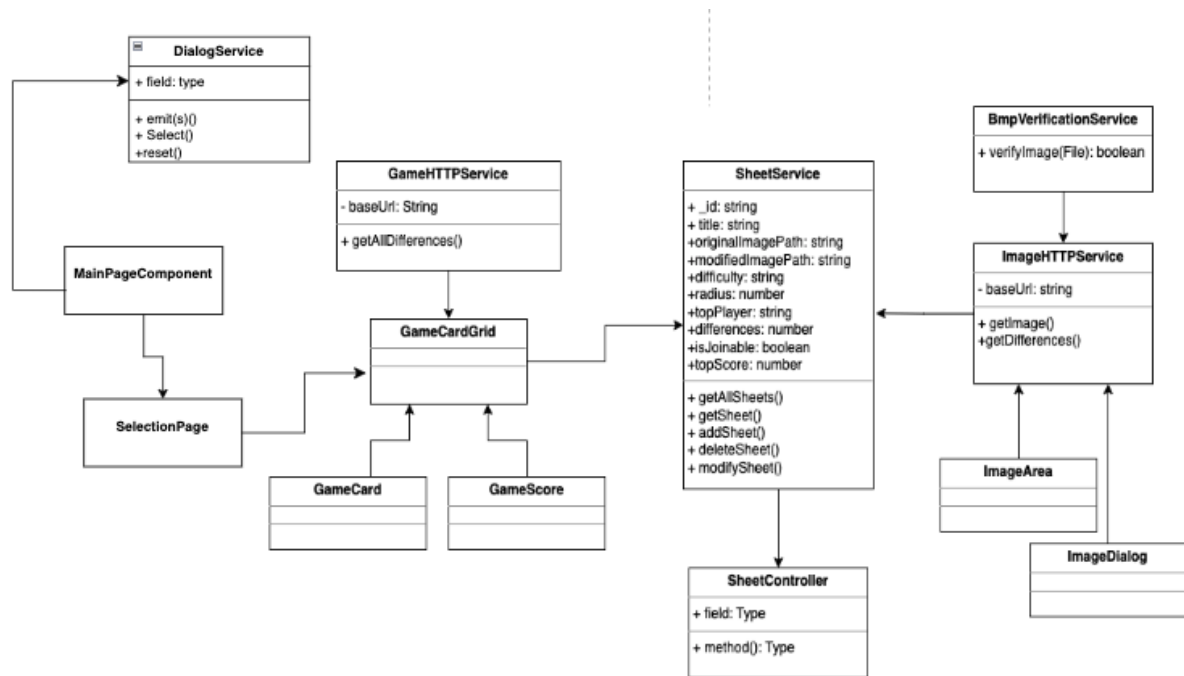
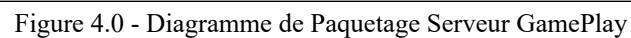


Figure 3.0 - Diagramme de Paquetage Client Selection



paquetage de jeu :

Ce paquetage s'occupe de faire la deuxième moitié du travail qui est initié par le `SocketClientService`. En effet, cette partie s'occupe de la partie serveur. Il a comme but de répondre aux requêtes du client. On utilise des websockets, le `ChatGateway`, pour qu'il puisse communiquer avec les services que nous avons sur le côté serveur. Ces services communiquent avec la base de données.



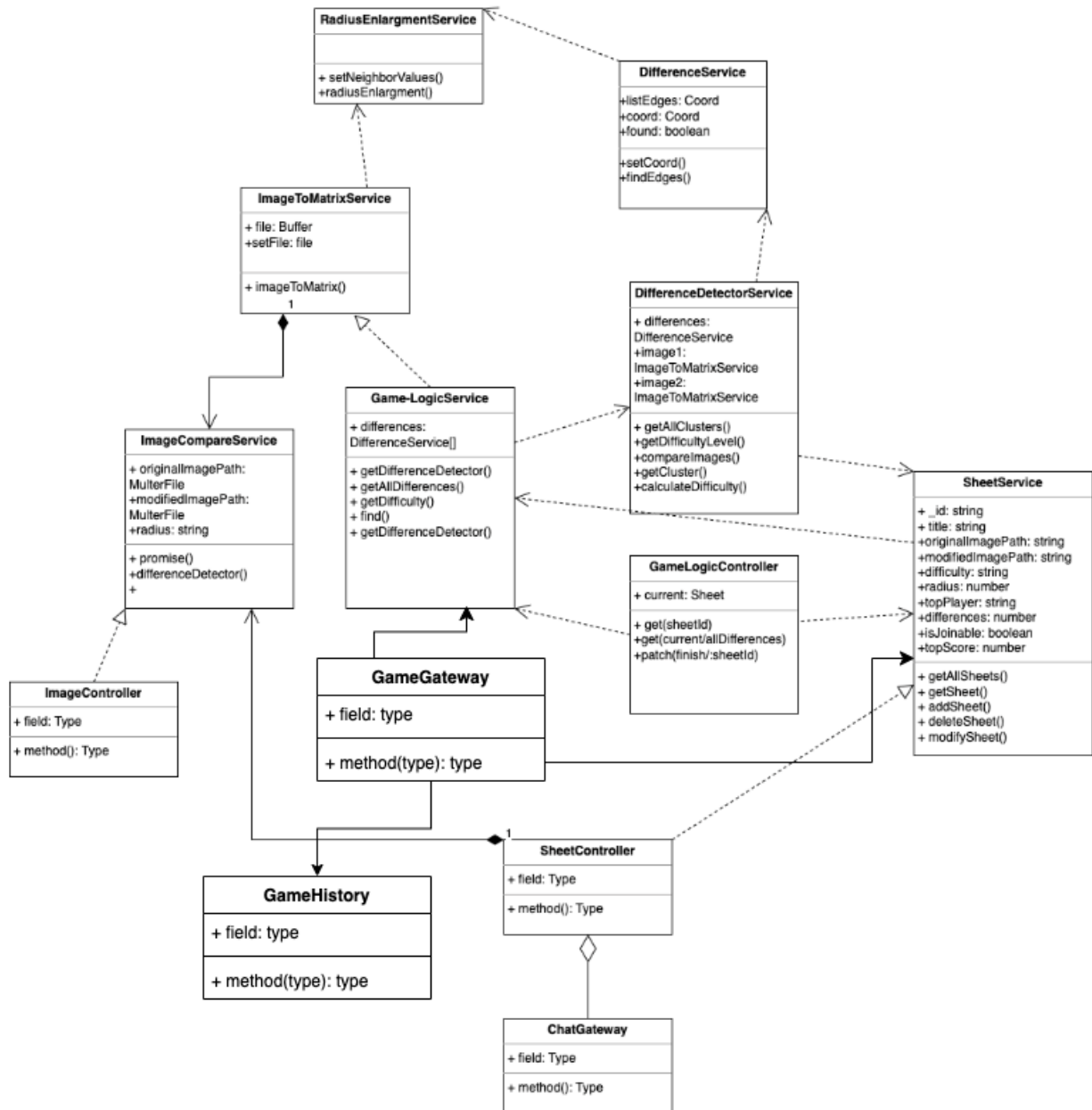


Figure 8 - Diagramme de classe Serveur Gameplay

paquetage de configuration :

GameConfiguration

Ce paquetage s'occupe de faire la deuxième moitié du travail qui est initié par le Sheet HTTPService. En effet, cette partie s'occupe de la partie serveur des implications de la Game Configuration. Il a comme but de répondre aux requêtes du client. Grâce aux controllers, ils permettent d'avoir les jeux sur la page de configuration, puisqu'ils appellent les fonctions des services. En bref, la configuration côté serveur s'assure d'avoir les game sheets et amène toute modifications et établissent les modifications sur la database.

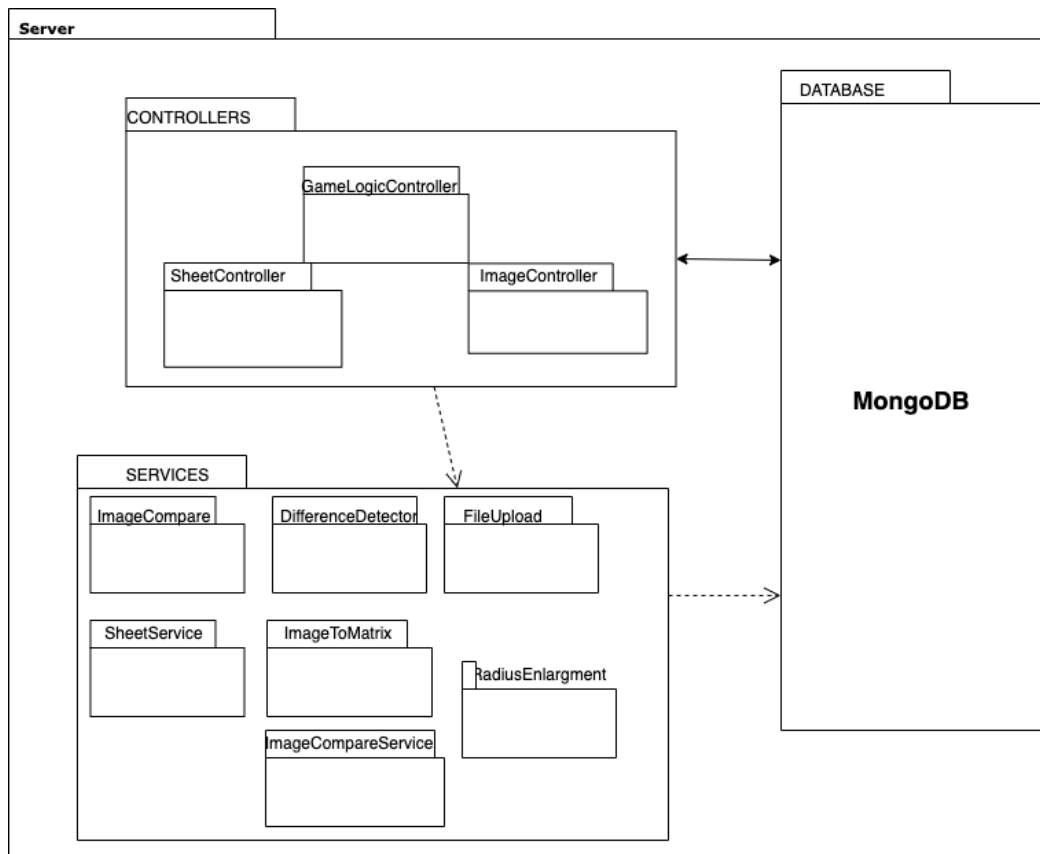
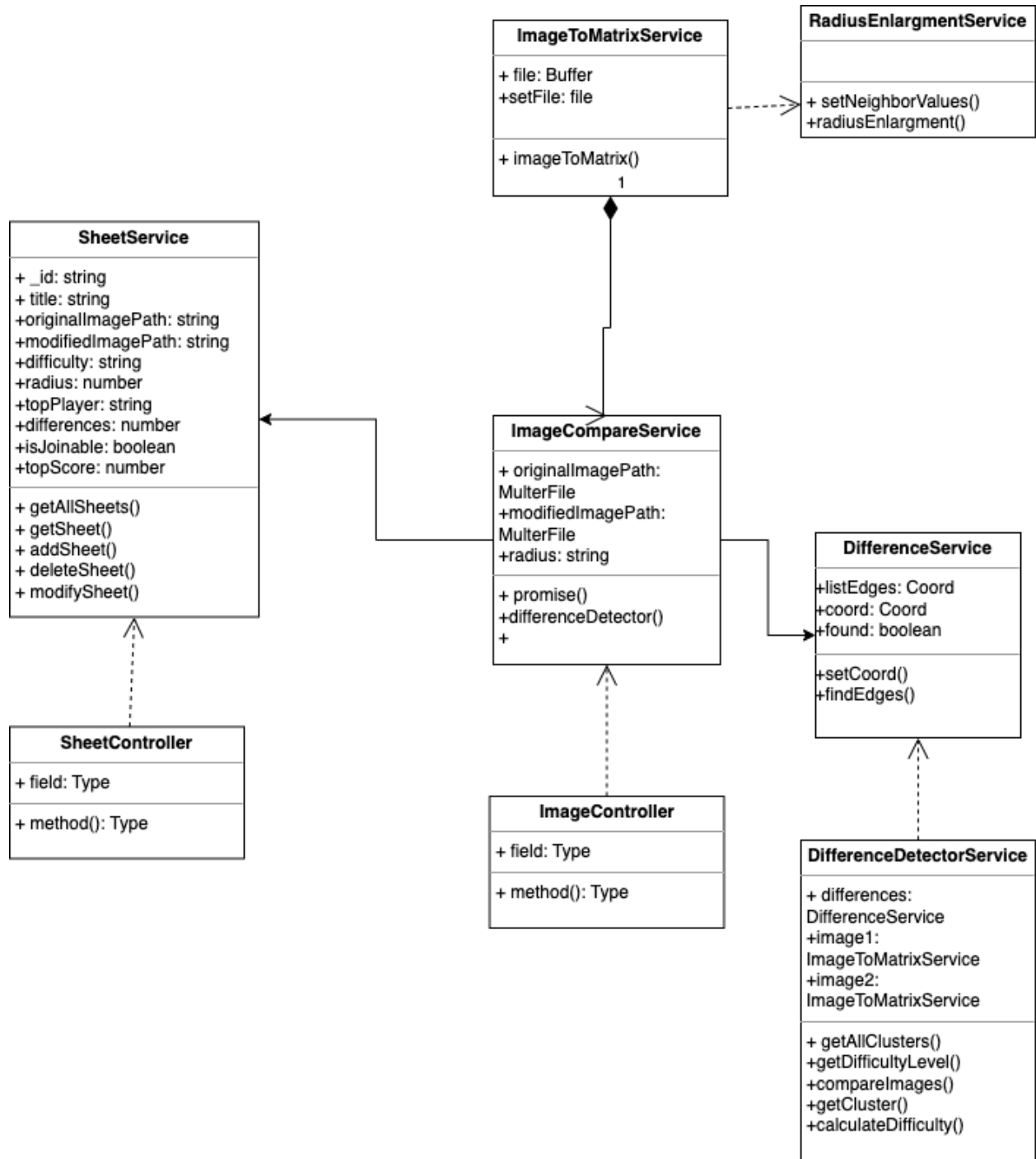


Figure7 - Diagramme de Paquetage Seveur Configuration



paquetage de Selection :

GameSelection

Ce paquetage est responsable de faire la deuxième moitié du travail du GameSelection. Elle s'assure, grâce aux controllers qui communiquent avec les services, que nous avons pris les sheets de la database pour le bon mode. On a aussi des fonctionnalités comme le clavardage qui utilise le websocket.

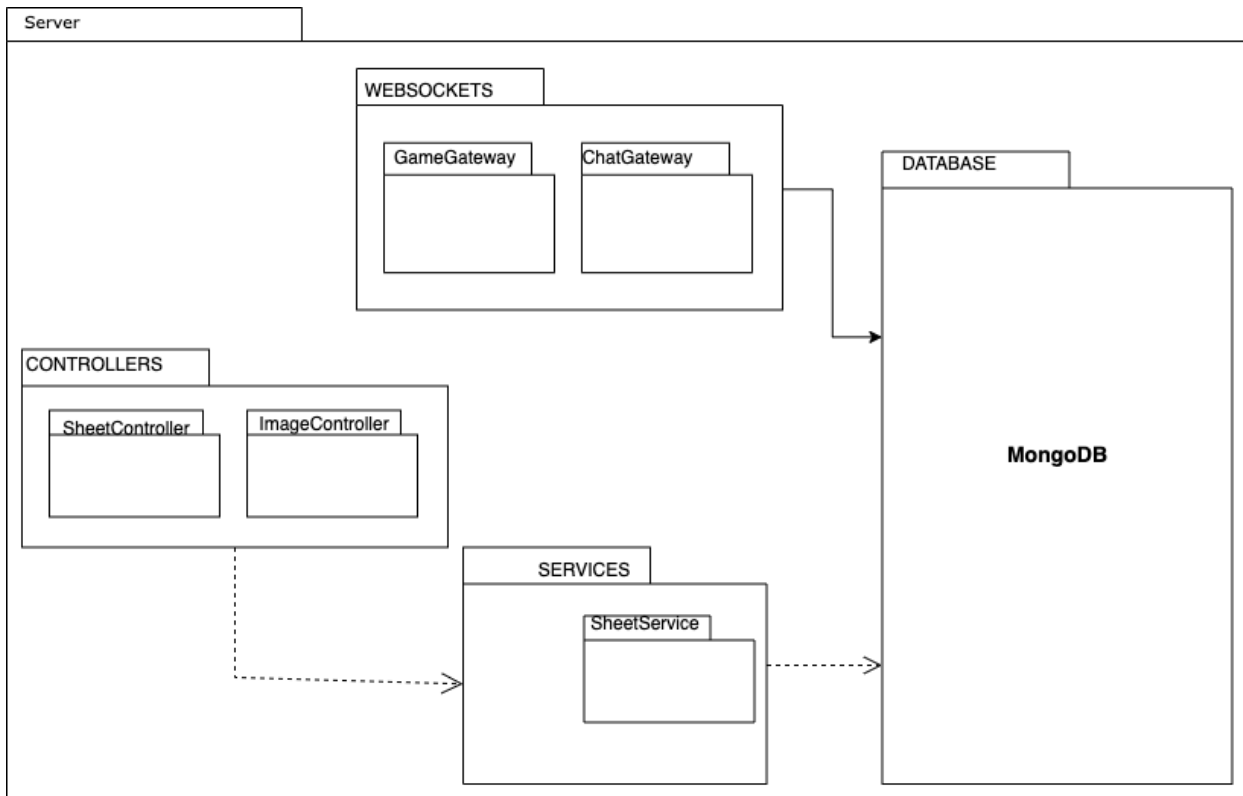
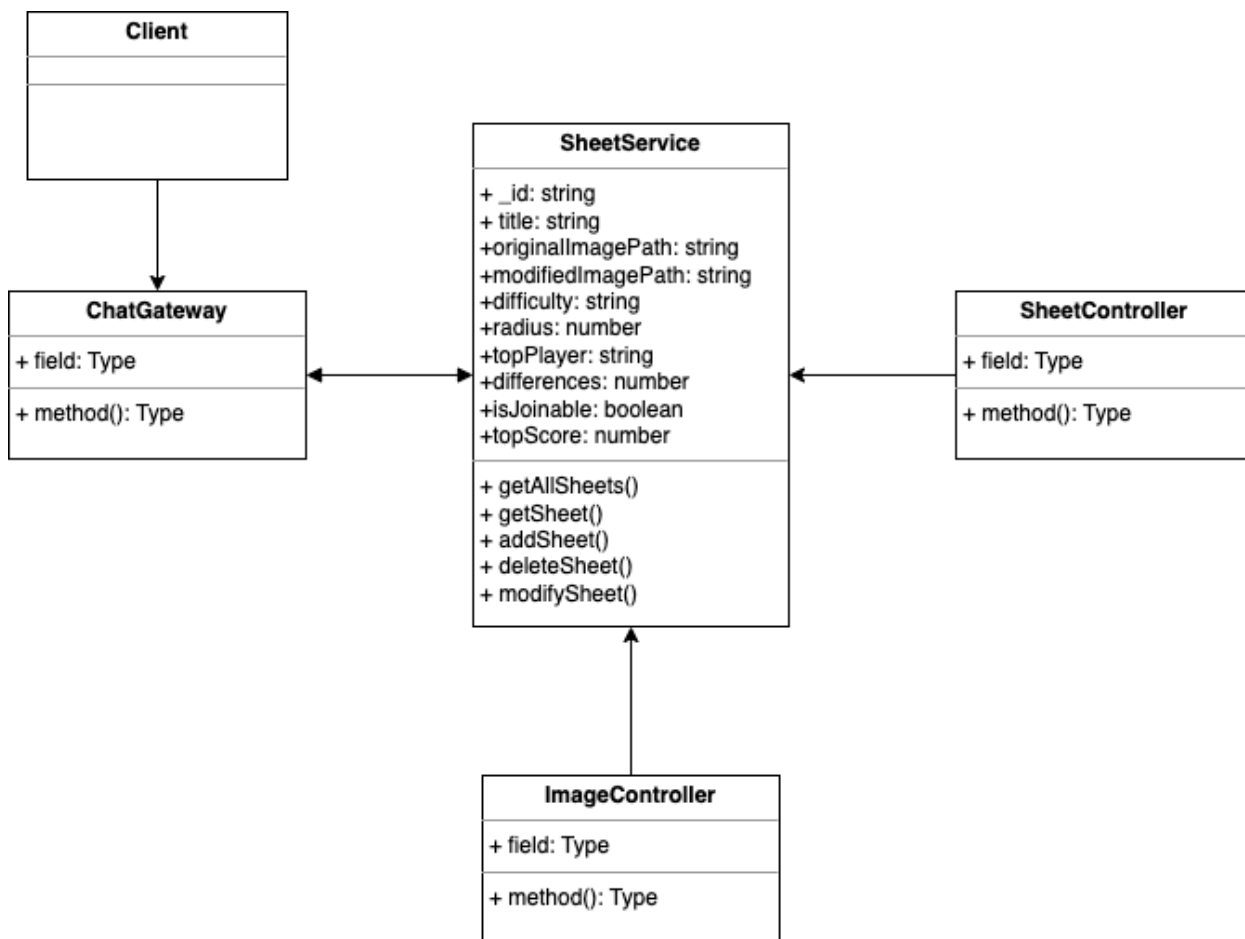


Figure 9 - Diagramme de Paquetage Serveur Selection



5. Vue de déploiement

5.1 Diagramme de déploiement

