

Document d'architecture logicielle

Version 1.5

Historique des révisions

| Date | Version | Description | Auteur |
|------------|---------|--|----------------|
| 2023-03-21 | 1.0 | Introduction, vue des cas d'utilisation et vue de déploiement. | Simon Bachand |
| 2023-03-21 | 1.1 | Diagramme de séquences | Mounir Lammali |
| 2023-03-21 | 1.2 | Diagramme de logique (packaging) | Étienne Perron |
| 2023-04-17 | 1.3 | Sprint 3 - révision des diagrammes de séquences | Mounir Lammali |
| 2023-04-19 | 1.4 | Sprint 3 - révision du diagramme de vue logique | Étienne Perron |
| 2023-04-19 | 1.5 | Sprint 3 - révision du diagramme de déploiement | Sidney Gharib |

Table des matières

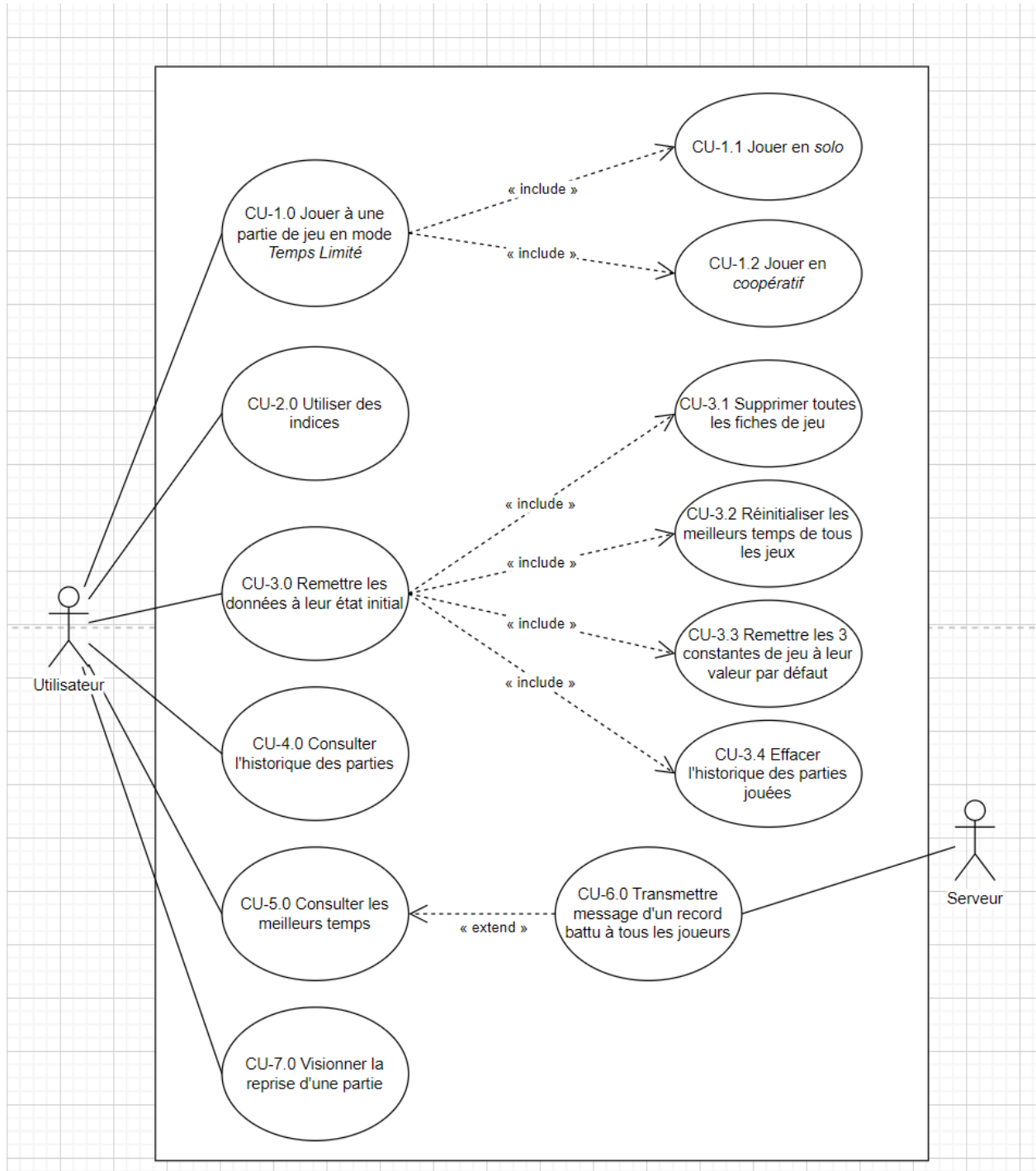
| | |
|-------------------------------------|-----------|
| 1. Introduction | 4 |
| 2. Vue des cas d'utilisation | 5 |
| 3. Vue des processus | 6 |
| 4. Vue logique | 8 |
| 5. Vue de déploiement | 14 |

Document d'architecture logicielle

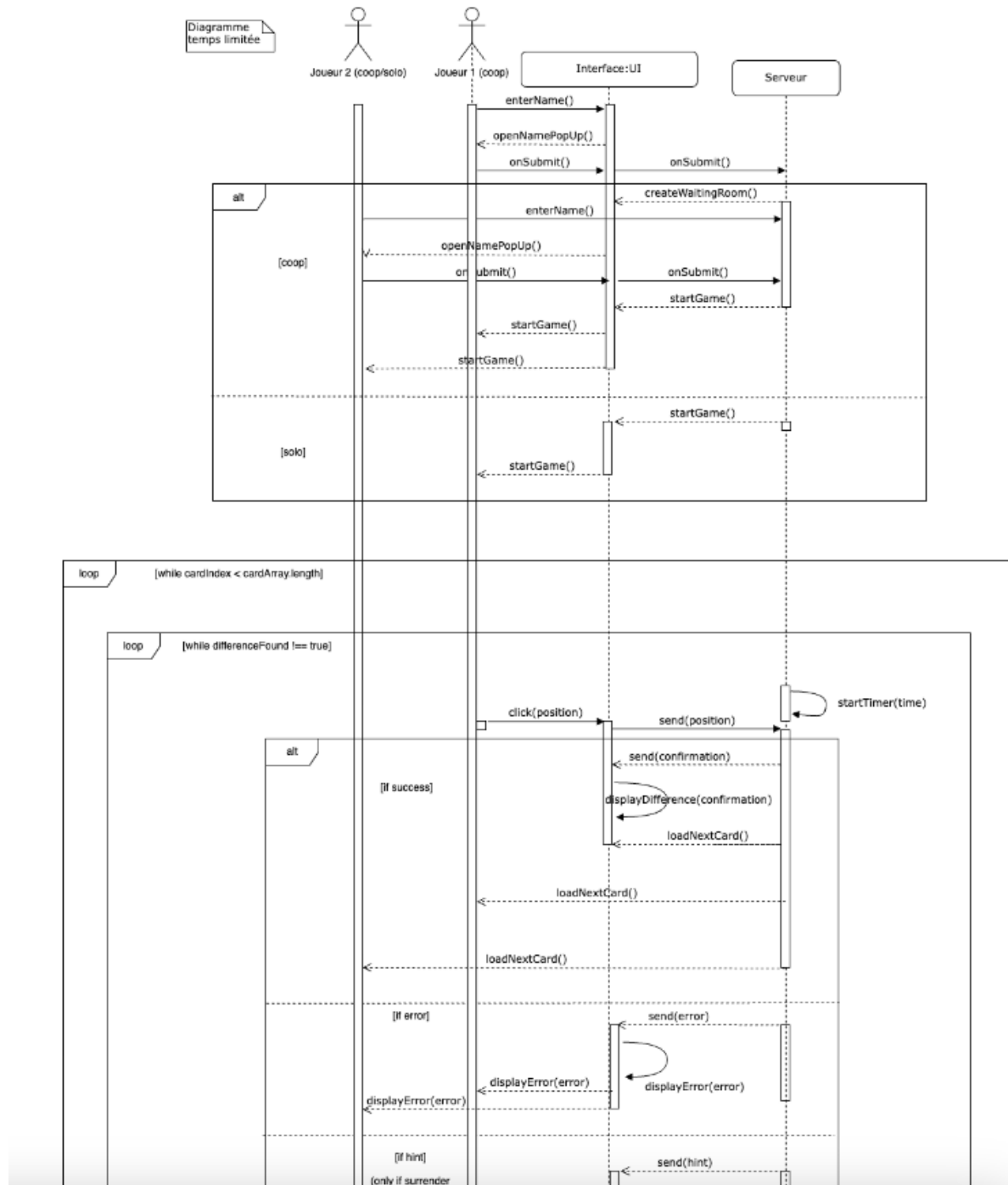
1. Introduction

Le présent document d'architecture logicielle décrit les différentes vues du système que nous avons conçu. Cette architecture est basée sur une approche modulaire qui permet une flexibilité accrue et une évolutivité du système. Les vues décrites dans ce document sont la vue des cas d'utilisation, la vue des processus, la vue logique et la vue de déploiement. Chacune de ces vues fournit une perspective unique sur le système et permet de mieux comprendre son fonctionnement. La section *Vue des cas d'utilisation* mettra en avant les différents cas d'utilisation du système, en se concentrant sur les fonctionnalités clés que les utilisateurs pourront utiliser. La section *Vue des processus*, quant à elle, décrira les différents processus impliqués dans le fonctionnement du système, en identifiant les étapes clés de chaque processus. La section *Vue logique* présentera la structure logique du système, en décrivant la manière dont les différents éléments du système interagissent entre eux. Enfin, la section *Vue de déploiement* fournira une vue détaillée de la manière dont le système est déployé, en identifiant les différents éléments faisant part du système et de quelle manière lesdits éléments communiquent entre eux.

2. Vue des cas d'utilisation



3. Vue des processus



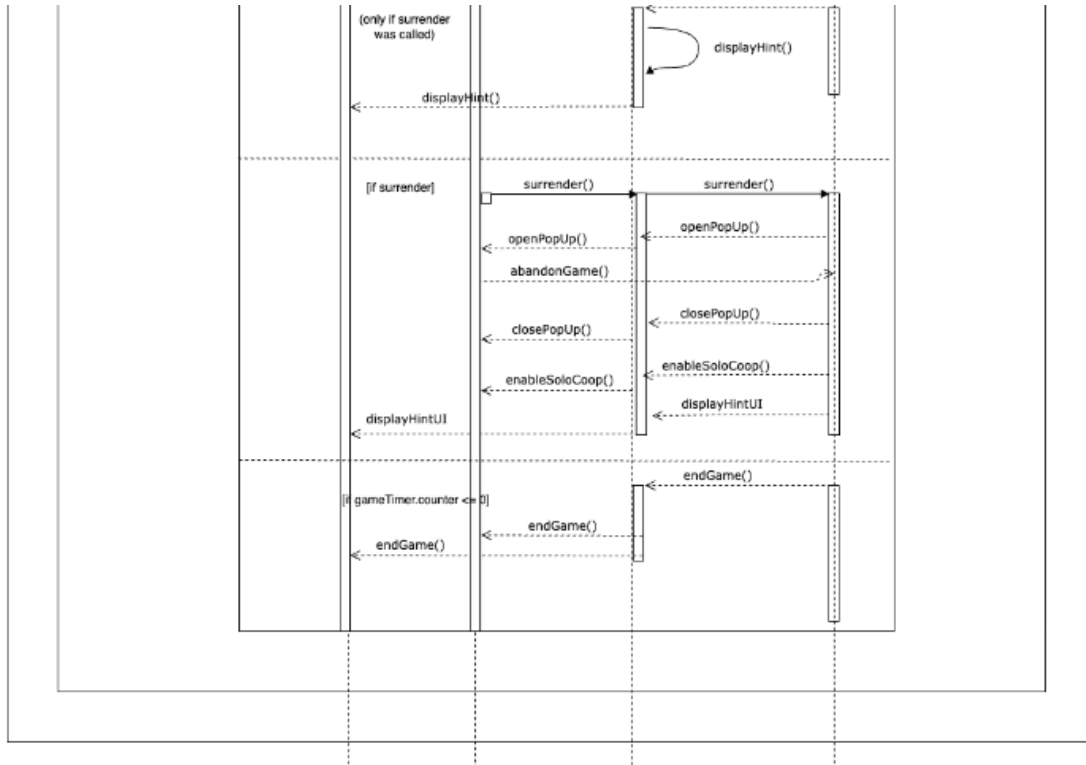
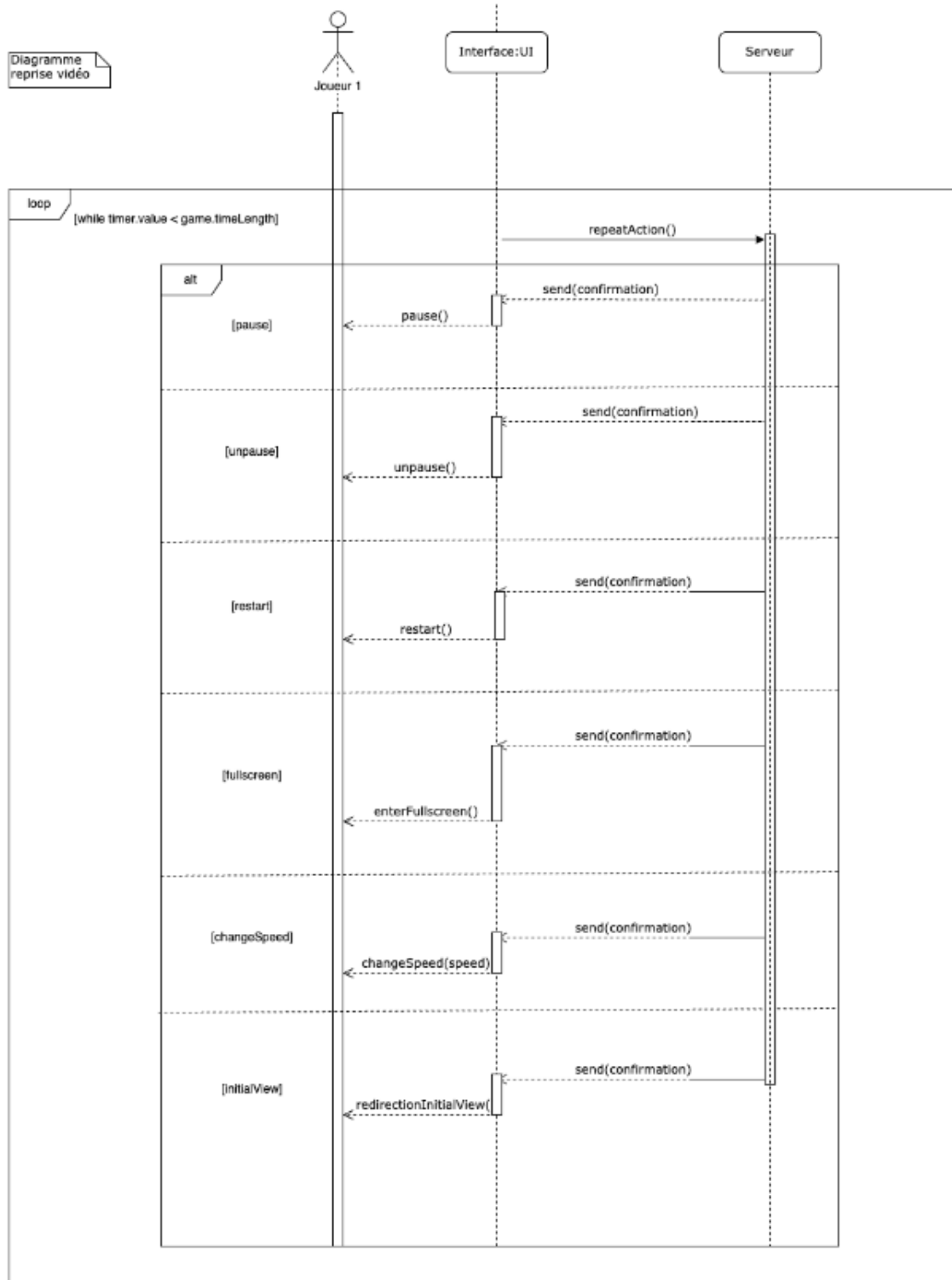
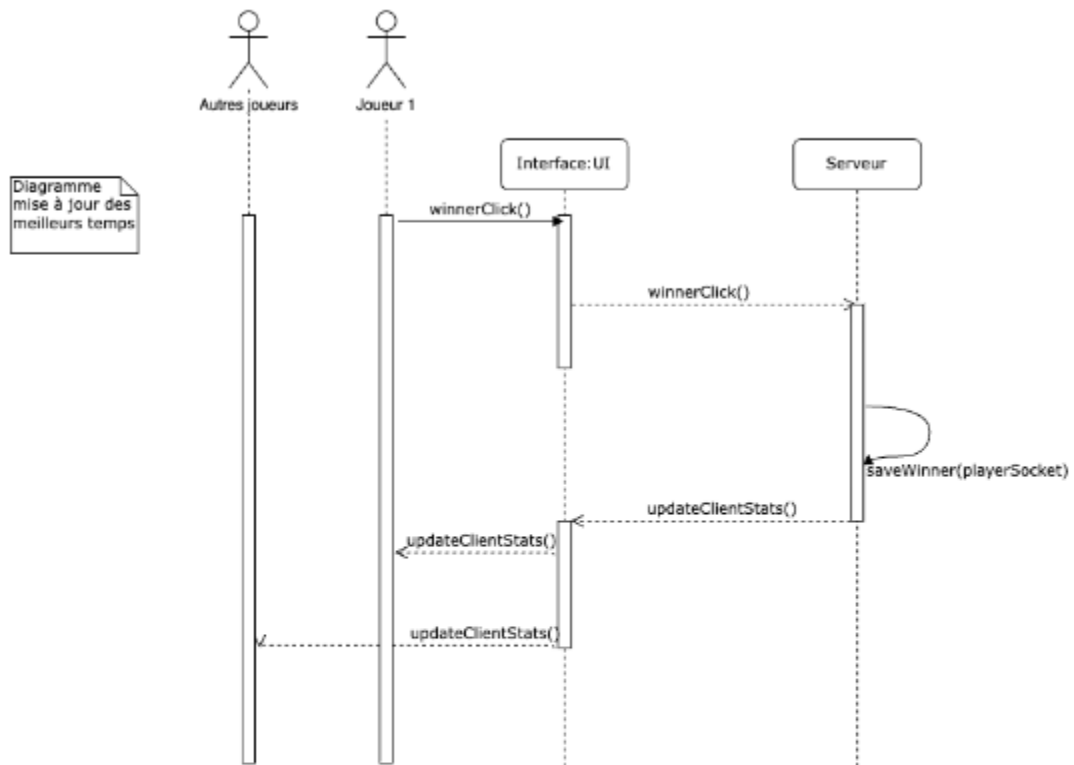
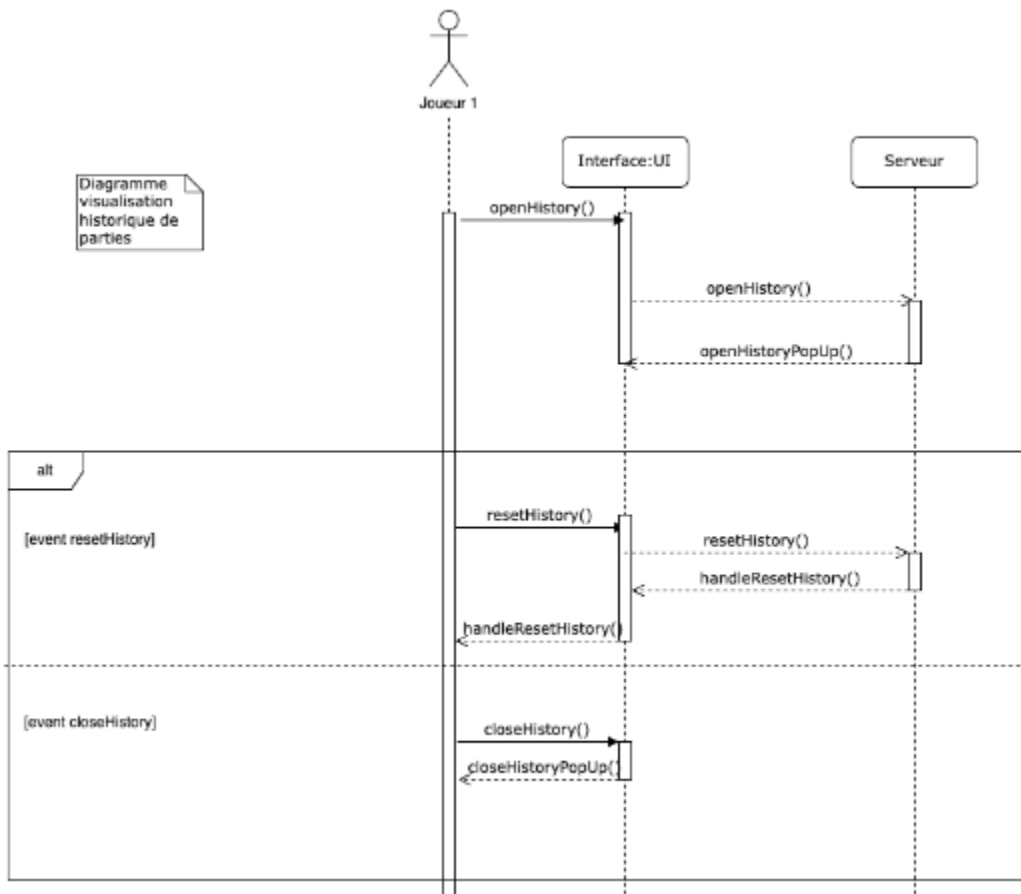
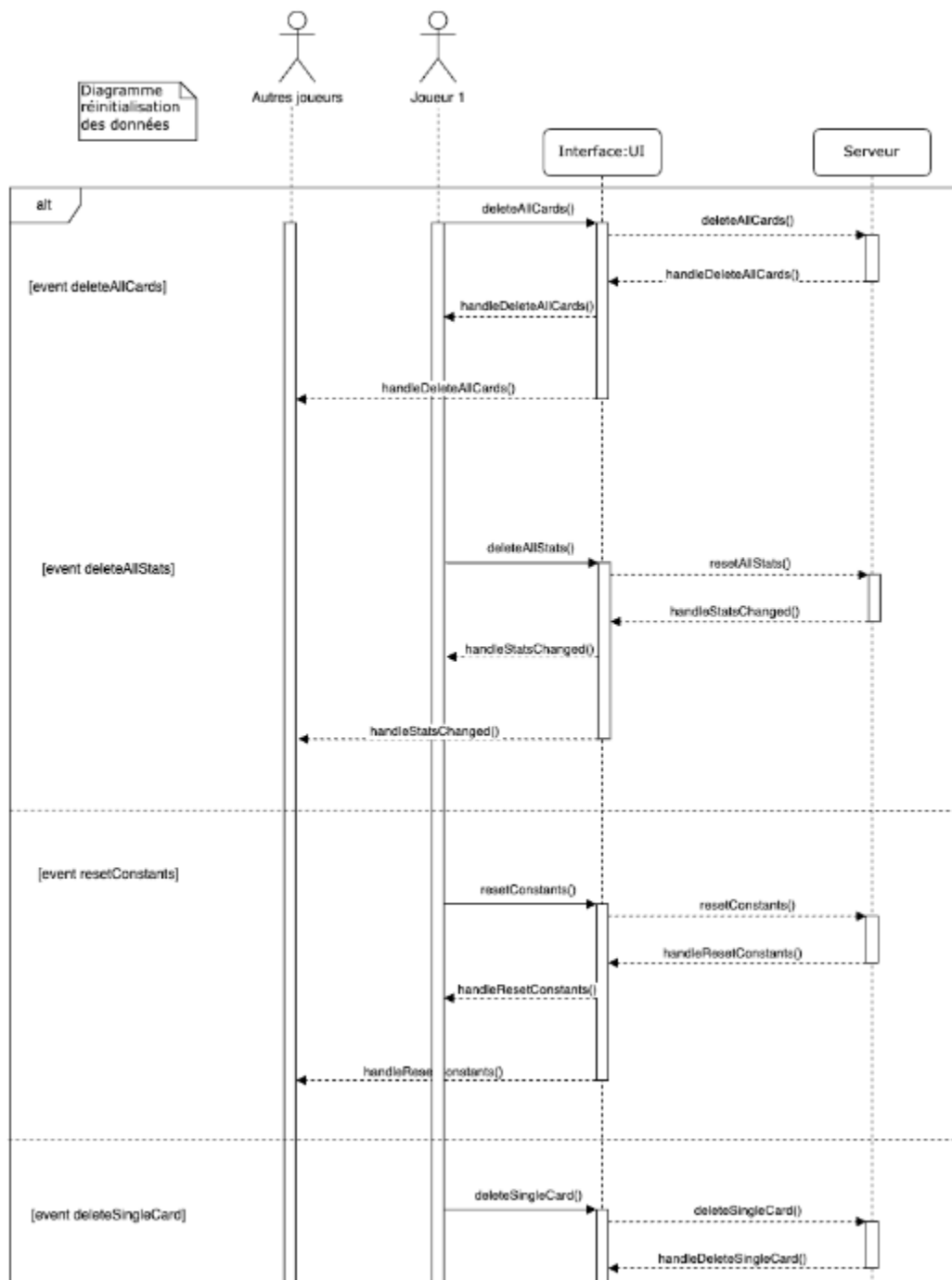


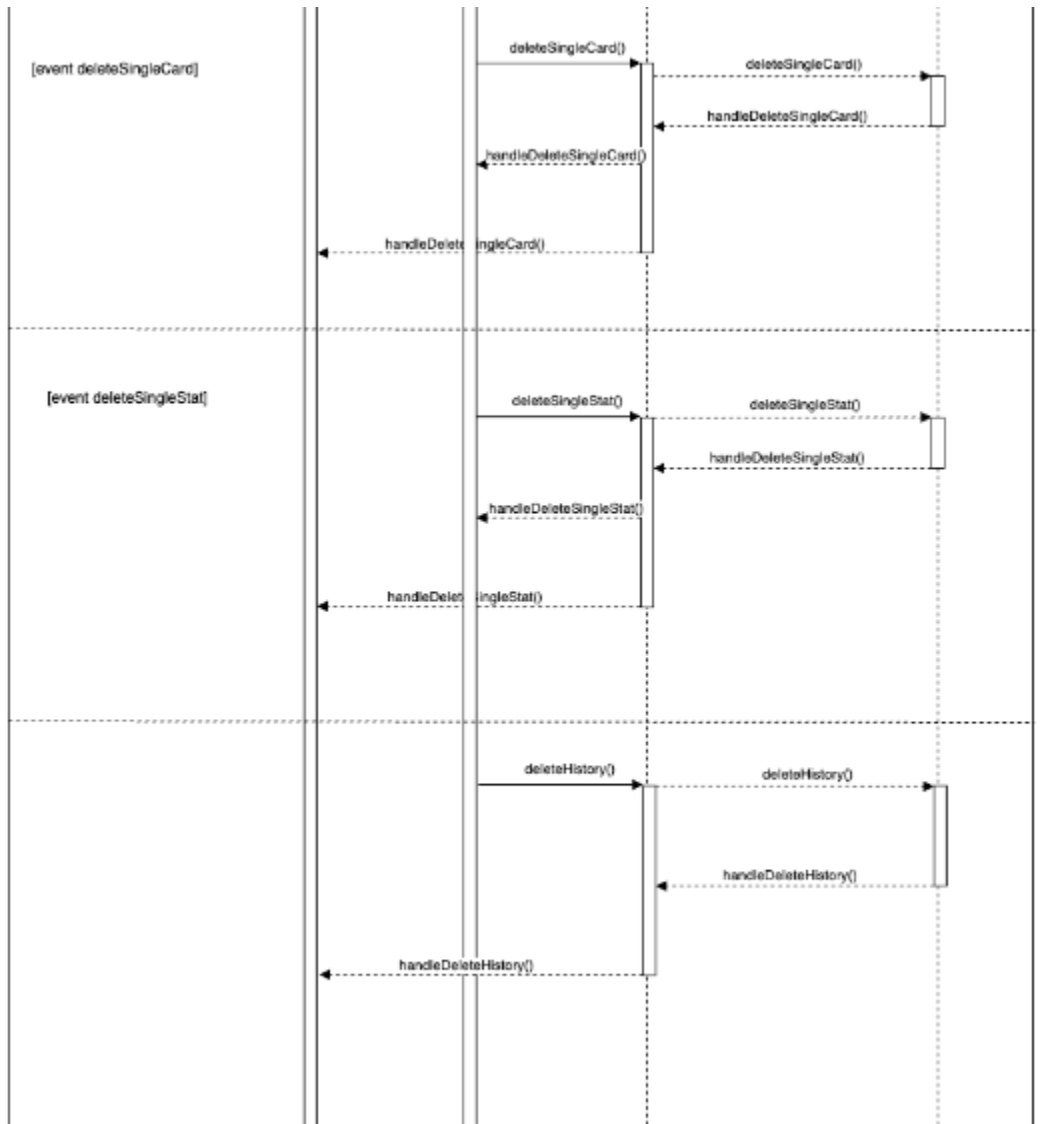
Diagramme
reprise vidéo



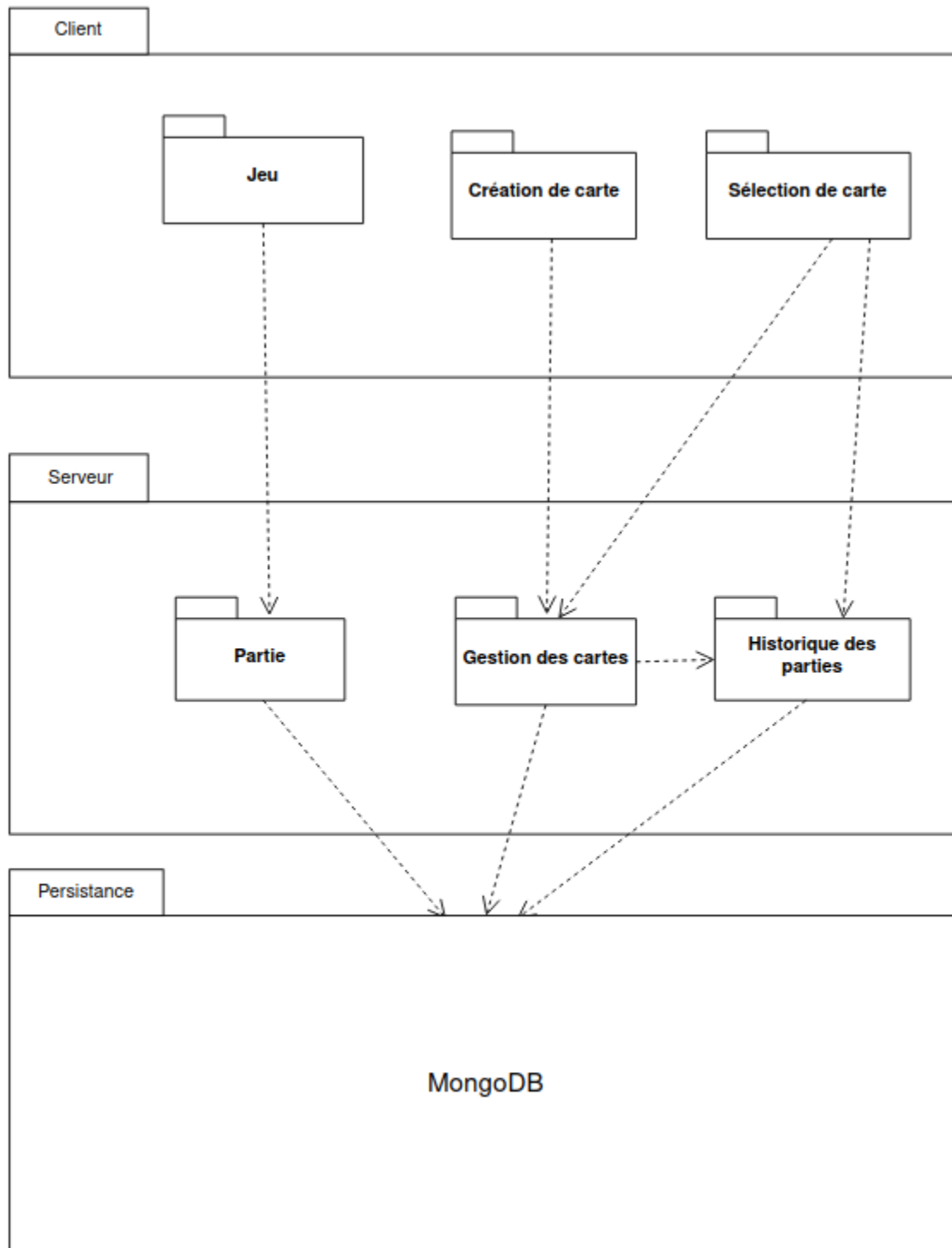








4. Vue logique



Client

Le paquetage “Client” est responsable de tout ce qui a trait à la vue du jeu. On y trouve tous les éléments qui ont un impact sur ce que l'utilisateur pourra observer sur son écran en plus de la logique qui entoure ces éléments. Toutefois, la logique d'application ne s'y trouve pas. Les paquetages qu'on y trouve communiquent avec le serveur lorsqu'il n'est plus question de visuel mais bien de logique.

Serveur

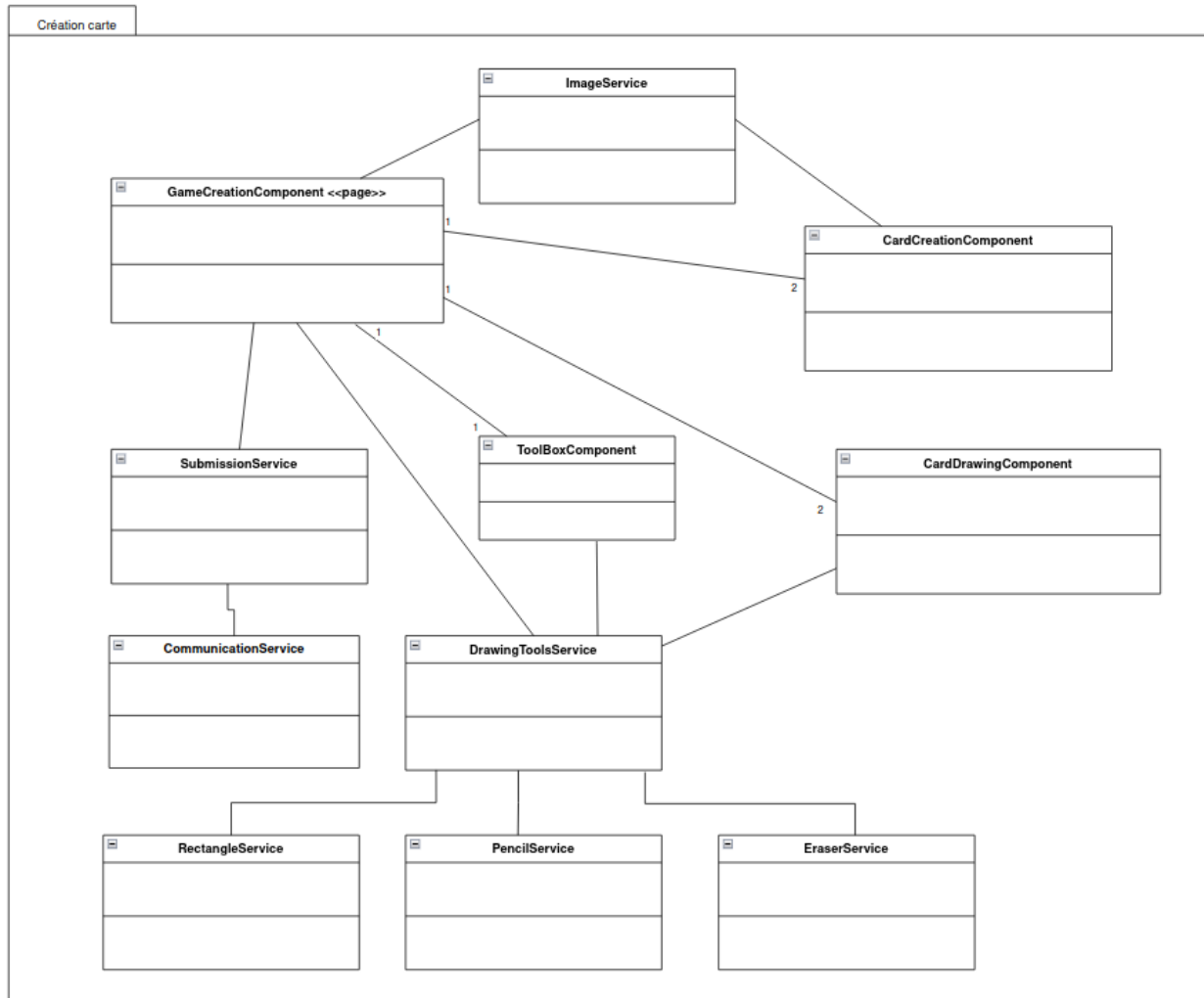
Le paquetage “Serveur” contient la logique d'application. C'est lui qui gère les parties, s'occupe de la gestion des cartes de jeu et de l'historique des parties. Ce paquetage communique avec la base de données pour les informations qui demandent de la persistance.

Persistance

Le paquetage “Persistance” contient les informations qui nécessitent d'être conservées à long terme, tels que les statistiques de meilleurs temps ainsi que l'historique des parties jouées.

Diagrammes de classe pour les paquetages du client:**Création carte**

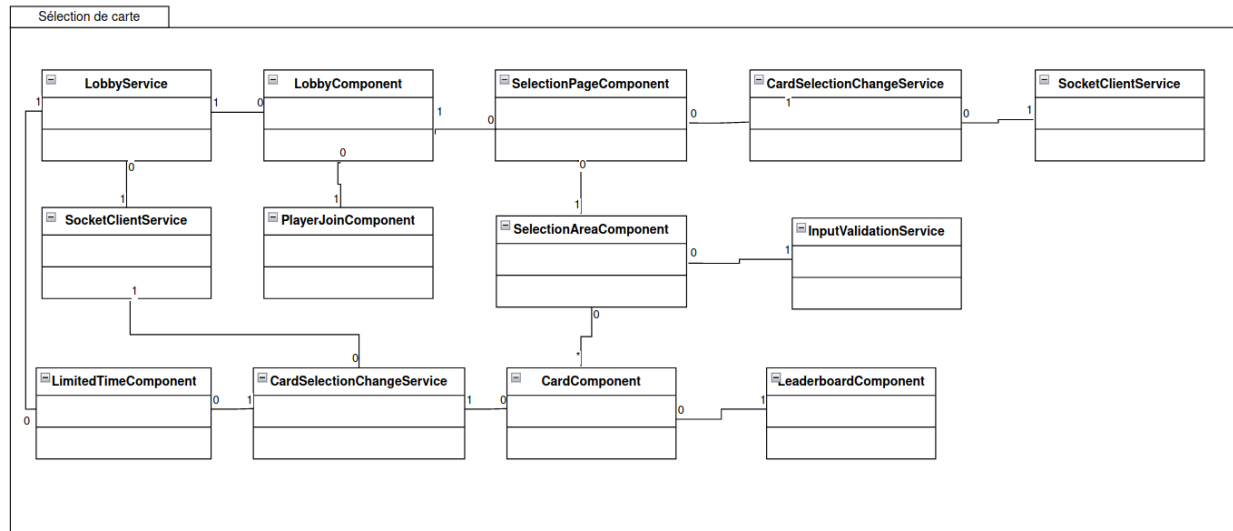
Le paquetage “Création carte” est responsable de la vue permettant à l'utilisateur de créer une nouvelle carte. Ceci implique l'ajout d'une carte à partir d'une image bitmap que le client possède ainsi que la modification de celle-ci au moyen des outils crayon et efface parmi d'autres. Ce service communique avec le serveur dynamique afin de s'assurer que la nouvelle carte respecte les critères de création et affiche au client l'image de différence. Le paquetage s'occupe aussi de communiquer avec le serveur dynamique pour lui demander d'ajouter la nouvelle carte à sa liste de cartes existantes



Sélection de carte

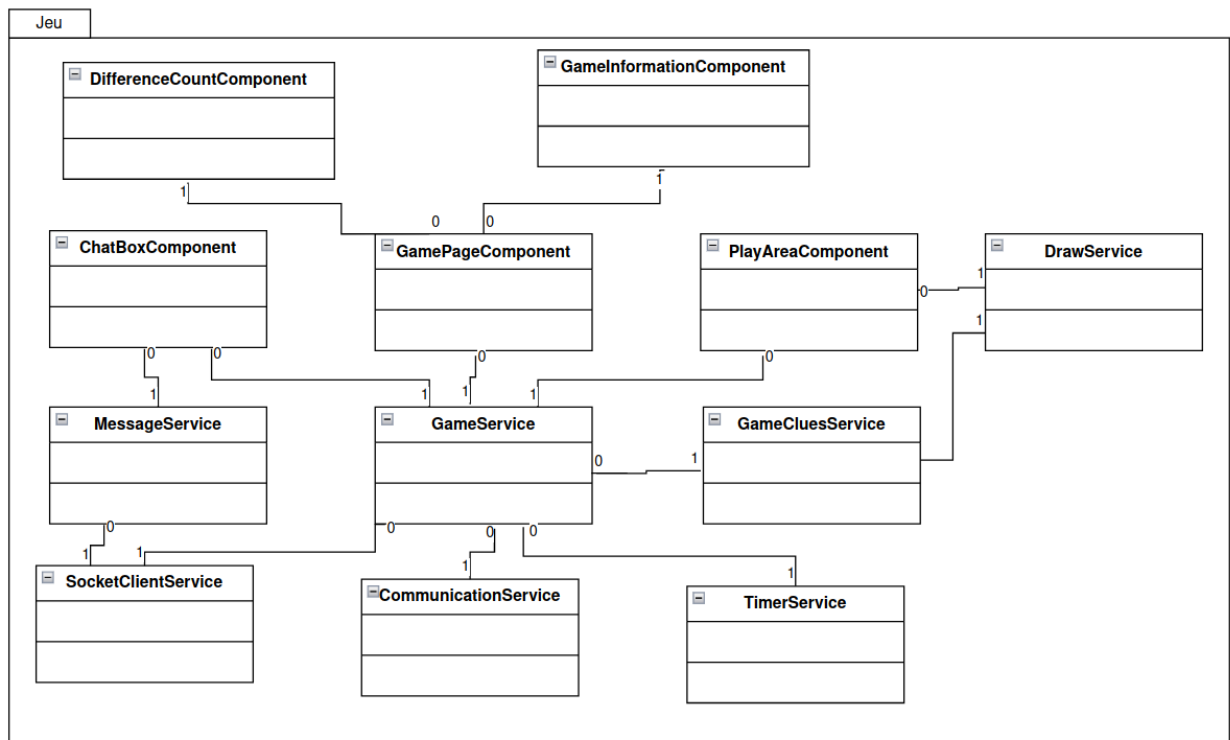
Le paquetage “Sélection de carte” s’occupe d’afficher à l’utilisateur les cartes de jeu disponibles. Lorsque celui-ci sélectionne une carte pour jouer, le paquetage prend les mesures nécessaires pour amener le joueur en jeu. Ces mesures impliquent, entre-autres, la demande de nom d’utilisateur et le démarrage de la partie en communiquant avec le serveur. Lorsque le joueur souhaite jouer en mode temps limité, la sélection de carte est aléatoire. Ce paquetage garde contact avec le serveur dynamique afin d’afficher la bonne information selon si un joueur a créé une salle d’attente sur la carte ou pas encore. Ses communications avec le serveur lui permettent aussi de déterminer si les cartes affichées existent toujours et si les statistiques de meilleurs temps sont à jour. Finalement, ce paquetage réutilise la logique

d'affichage pour donner accès aux joueurs à des options de configuration sur les cartes, telles que la suppression de carte.



Jeu

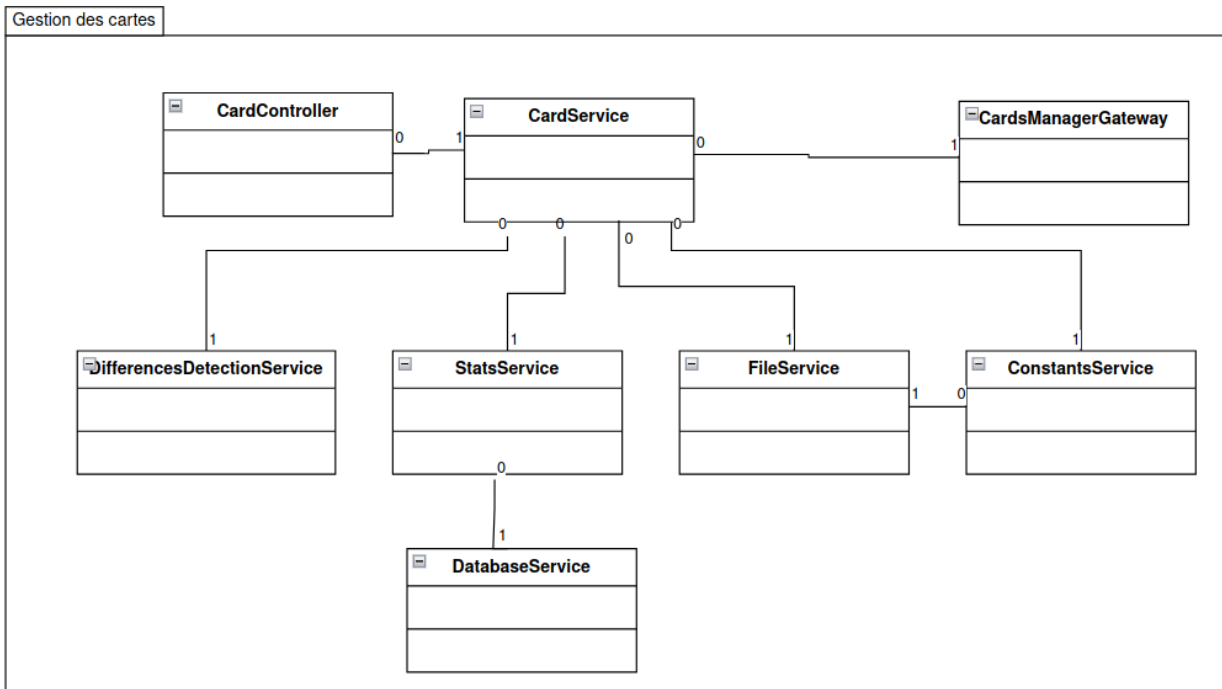
Le paquetage “Jeu” s’occupe de l’affichage des parties aux joueurs. Il garde une communication constante avec le serveur dynamique pour l’informer des dernières modifications et obtenir les résultats de celles-ci afin de garder l’interface des joueurs à jour. Ce paquetage permet aussi d’afficher le chat de aux joueurs afin que ceux-ci communiquent ensemble et puissent être informés des messages système. Lorsqu’en jeu, on affiche des informations intéressantes aux joueurs tels que le nombre de différences et la valeur de la minuterie. Finalement, c’est ce paquetage qui récupère les demandes d’indices des utilisateurs, les envoie au serveur, puis affiche la réponse de celui-ci au client.



Diagrammes de classe pour les paquetages du serveur:

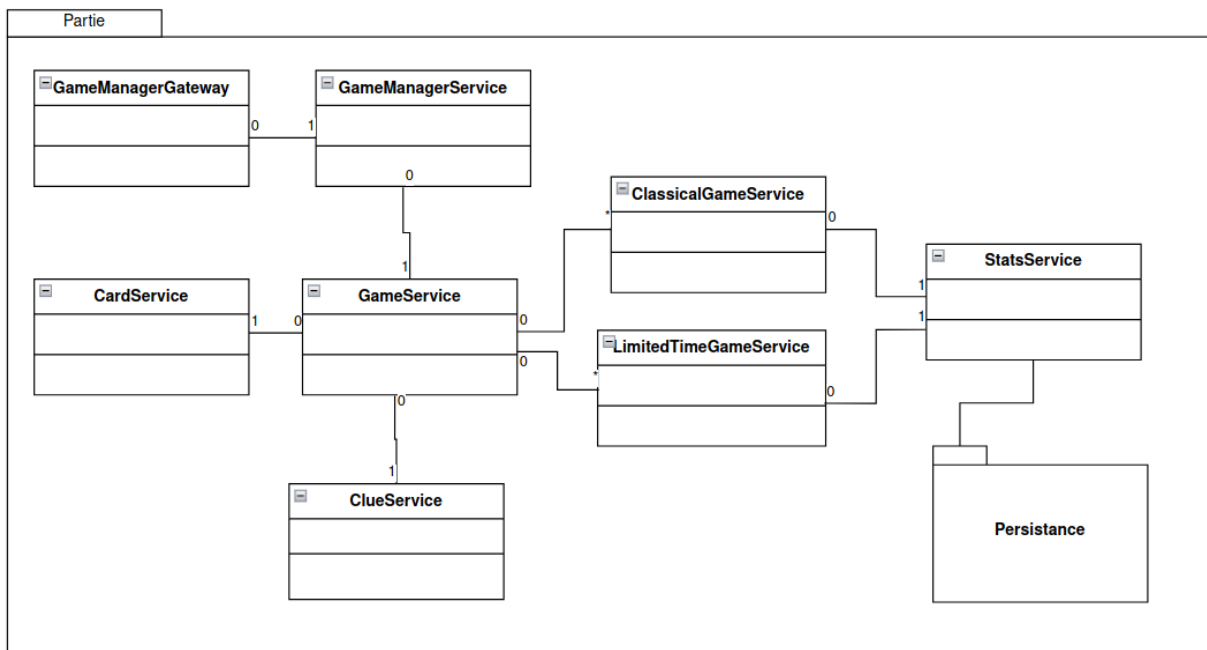
Gestion des cartes

Le paquetage “Gestion des cartes” s’occupe de toutes les demandes d’accès, de modification ou de création à l’égard des cartes de jeu. Ce paquetage contient la logique permettant par exemple de calculer les différences entre deux images, modifier les statistiques d’une carte, modifier les constantes de jeu appliquées aux cartes, supprimer une carte et récupérer une carte en entier.



Partie

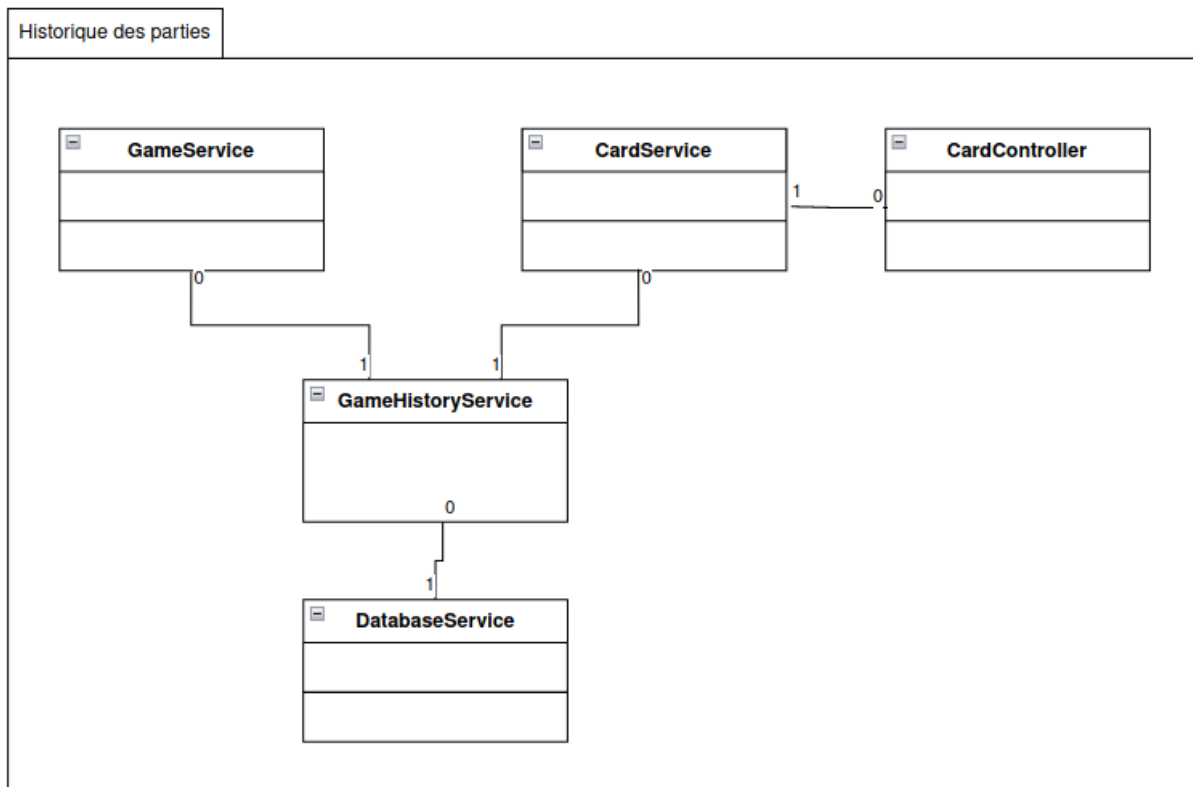
Le paquetage “Partie” contient toute la logique de jeu. C’est celui-ci qui détermine si le clique d’un joueur correspond à une différence ou non et si la partie commence ou termine. Lorsque la partie est terminée, ce paquetage contacte le service de carte pour que celui-ci enregistre les nouvelles statistiques. Les connexions entre ce paquetage et les clients se font avec le protocole websocket. La logique de ce paquetage fait aussi la distinction des parties de type “Temps Limitée” de celles de type “Classique”



Historique des parties

Le paquetage “Historique des parties” contient la logique permettant l’enregistrement et l’accès à l’historique des parties. Le paquetage contient l’historique sur disque et l’accès à celui-ci depuis un client se fait par HTTP au moyen du “CardController”. Autrement, les nouvelles données sont

enregistrées à partir du “GameService” qui connaît les informations de fin de partie.



5. Vue de déploiement

