
INFO-F209 : PROJET D'ANNÉE

WIZARD POKER

SOFTWARE REQUIREMENT DOCUMENT

Amin Abdelkefi - *matricule*
Lucie Borremans - *matricule*
Corentin Candeur - 000411488
Jonas Descurieux - 000382867
Rémy Detobel - 000408013
Maxime Dewit - *matricule*
Carlos Requena López - 000410031
Raphaël Slagmolen - 000377852

18 décembre 2015

Table des matières

1	Introduction	1
1.1	But du projet	1
1.2	Glossaire	1
1.3	Historique du document	1
2	Besoins de l'utilisateur	1
2.1	Exigences fonctionnelles	1
2.1.1	Créer un compte lié à un pseudo	1
2.1.2	Se connecter à son compte	2
2.1.3	Consulter sa collection de cartes	2
2.1.4	Gestion de decks	2
2.1.5	Gérer une liste d'amis	2
2.1.6	Discuter avec ses amis	2
2.1.7	Proposer une partie à un ami	2
2.1.8	Trouver un adversaire pour un duel	2
2.1.9	Consulter le classement	3
2.2	Exigences non fonctionnelles	3
2.2.1	Ergonomie	3
2.2.2	Design	3
2.2.3	S'amuser	4
2.3	Exigences de domaine	4
3	Besoins du système	4
3.1	Exigences fonctionnelles	4
3.2	Exigences non fonctionnelles	4
3.3	Design et fonctionnement du systeme	4

1 Introduction

1.1 But du projet

Dans le cadre du projet d'analyse et méthode et de système d'exploitation, il est demandé d'implémenter un jeu de carte en ligne basé sur une architecture client/serveur en C/C++.

Ce projet devra fournir un jeu de carte symétrique où deux joueurs pourront s'affronter et gagner des points pour un classement général. Chaque joueur disposera d'une collection de cartes à partir de laquelle il pourra construire des decks. Ces decks contiendront des cartes de type créature qui peuvent attaquer et des cartes de type sort qui lanceront des événements spéciaux.

Le projet requière la mise en place d'une architectures client/serveur avec le client qui sert principalement d'interface de jeu (graphique ou console) et les calculs qui seront effectué côté serveur. Il est donc important de développer un serveur robuste, efficace et rapide qui soit en mesure de gérer plusieurs parties simultanément tout en fournissant des services tels que le classement général, un service de compte et de login ainsi qu'un service de messagerie instantané entre deux joueurs. Pour cela, nous seront ammené à développer une application capable de gérer un maximum de cas de figure pour garantir la continuité et la stabilité de ces différent services. La gestion d'erreurs devra être telle que cette erreur ne perturbera aucune autre partie ni aucun des autres services.

Le Wizard Poker sera disponible à n'importe quel joueurs enregistré pour jouer contre d'autres joueurs et essayer de gagner des places dans le classement générale. Il permettra aussi à de nouveaux joueurs de consulter ce classement et de s'enregistrer en tant que nouveau joueur. Dans ce cas, le joueur recevra ses premières cartes et sera invité à confectionner son premier deck avant d'affronter son premier adversaire.

1.2 Glossaire

Glossaire

Collection Ensemble qui contient toutes les cartes disponibles et connues par le jeu

Deck Un deck est un ensemble de 20 cartes

example1 description1

example2 description2

Tchat Messagerie instantanée permettant à 2 personnes de s'échanger des messages écrit

1.3 Historique du document

2 Besoins de l'utilisateur

2.1 Exigences fonctionnelles

2.1.1 Créer un compte lié à un pseudo

Tout visiteur se verra offerte l'opportunité de se créer un compte lié à un pseudo (si ce dernier est disponible) afin de pouvoir profiter pleinement du jeu.

Version	Auteur	Date modification	Description des changements
NEXT			
NEXT			
v0.3	Rémy et Lucie	11/12/15 16h20	Ajout de diagrammes, ajout de "Exigences du domaine"
v0.2	Amin Abdelkefi	10/12/15 23 :45	Ajout de la partie "Exigences fonctionnelles"
v0.1	Jonas et Raphaël	10/12/15 13 :28	Ajout de la partie "But du projet"
v0.0	Carlos Requena	26/11/15 22 :30	Mis en place. Diagrammes réalisés en groupe le 25/11/15 ajoutés

TABLE 1 – Changements document

2.1.2 Se connecter à son compte

Le joueur devra se connecter à son compte s'il souhaite effectuer les différentes actions répertoriées ci-dessous.

2.1.3 Consulter sa collection de cartes

Le joueur peut consulter l'ensemble des cartes présentes dans sa collection. Pour cela, il devra au préalable s'être connecté.

2.1.4 Gestion de decks

Un deck est un ensemble de cartes qui sera utilisé par le joueur lors d'un duel. Plusieurs actions peuvent être effectuées sur les decks telle qu'en supprimer un ou plusieurs, les consulter, les modifier ou tout simplement en créer un nouveau.

2.1.5 Gérer une liste d'amis

Chaque joueur aura une liste d'amis avec lesquels il pourra interagir s'il le souhaite. Il pourra aussi ajouter un amis à sa liste ou en supprimer.

2.1.6 Discuter avec ses amis

Le joueur pourra, à tout moment, parler avec un ou plusieurs de ses amis actuellement connectés. Pour cela, un tchat sera mis à disposition du joueur. Le joueur pourra donc communiquer avec ses amis par message à tout moment.

2.1.7 Proposer une partie à un ami

Si le joueur le souhaite, il peut proposer une partie avec un de ses amis connectés si ce dernier accepte cette demande.

2.1.8 Trouver un adversaire pour un duel

Lorsqu'un joueur voudra lancer une partie, le jeu trouvera automatiquement un adversaire dans la liste des personnes connectées. Le choix est totalement aléatoire.

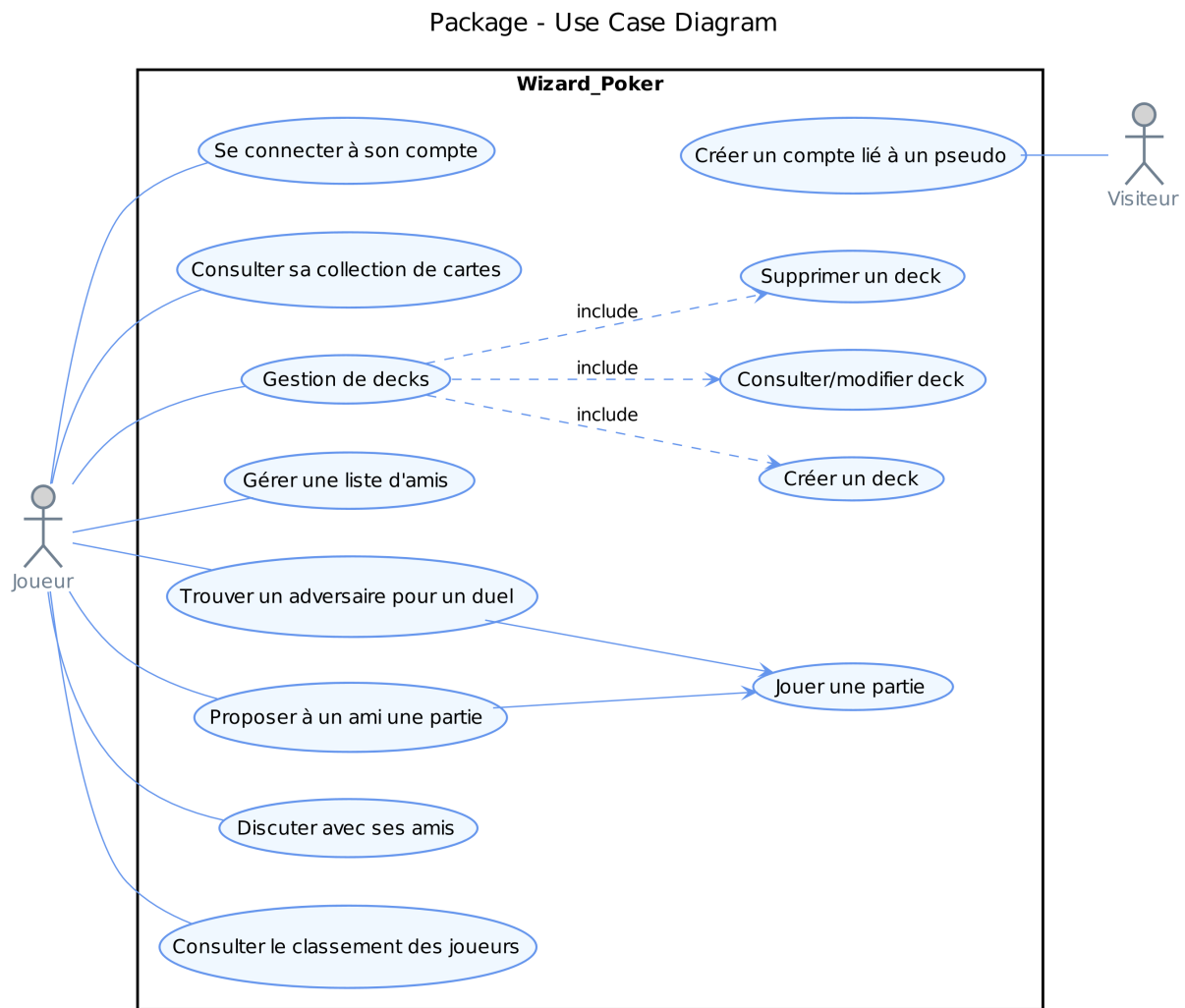


FIGURE 1 – Use Case Diagram du jeu en question

2.1.9 Consulter le classement

Le classement va permettre de classer les joueurs en fonction de leur nombre de victoires et de leurs défaites. Ce classement se basera donc sur le ratio victoire/défaites.

2.2 Exigences non fonctionnelles

2.2.1 Ergonomie

Créer un jeu qui sera simple d'utilisation et facilement compréhensible pour un utilisateur n'ayant jamais joué à un jeu de ce genre.

Pour cela, on peut rendre l'interface intuitive à l'aide de "boutons" décrivant exactement leurs utilités ou encore à l'aide d'un rapide tutoriel après la création du compte expliquant les différentes caractéristiques du jeu.

2.2.2 Design

Le design est l'apparence qu'aura l'interface graphique. Comme pour le point précédent, une belle interface va permettre à l'utilisateur de facilement voir les différentes possibilités offertes par le jeu en plus d'apprécier l'esthétique.

Le design est un "domaine" (vague) c'est-à-dire qu'il définit l'apparence du jeu mais aussi les différents sons et/ou musiques.

2.2.3 S'amuser

Qu'est-ce qu'un bon jeu ?

Un bon jeu est un jeu attractif, beau, facile à comprendre mais avant tout, un bon jeu est jeu où l'on s'amuse. Notre principal objectif sera de réaliser ce jeu et de permettre à tout les joueurs de s'amuser. (????? A changer si vous le souhaitez)

2.3 Exigences de domaine

Le domaine du jeux vidéo laisse beaucoup de liberté. En effet, chaque créateur essaye de créé son univers, de développer son concept. Cependant, tous les jeux doivent permettre à leur utilisateur de trouver du plaisir ou une certaines forme de satisfaction. Il faut également éviter que le joueur soit interrompu durant sa partie. Les exigences du domaine ne sont donc pas nombreuses mais compliqué à satisfaire complètement.

Pour résumé, un utilisateur attend de l'amusement et une stabilité acceptable (voir infaillible coté serveur). Nous nous concentrerons dans un premier temps (dans le cadre de ce projet) à ce deuxième point.

3 Besoins du système

3.1 Exigences fonctionnelles

Sequence diagram pour interaction ??

Le système doit interagir avec l'utilisateur, et lui proposer certaines actions, comme commencer un duel, gérer ses s, consulter le classement des joueurs, etc.

Pour les nouveaux joueurs, l'application doit offrir la possibilité de s'enregistrer, avec un pseudonyme et un mot de passe.

3.2 Exigences non fonctionnelles

Les exigences non fonctionnelles décrivent le comportement de l'application.

Le système doit être tout d'abord robuste et stable, c'est-à-dire gérer les erreurs et éviter les crashes et les déconnexions... Il doit par ailleurs être agréable d'utilisation pour l'utilisateur (ergonomie et esthétique/design) et garantir que celui-ci ne puisse pas tricher pendant une partie.

Donc en résumé, robuste, stable, sûr, et excessivement amusant. Des heures de franche rigolade, en toute sécurité, sont assurées à tout utilisateur du jeu.

3.3 Design et fonctionnement du systeme

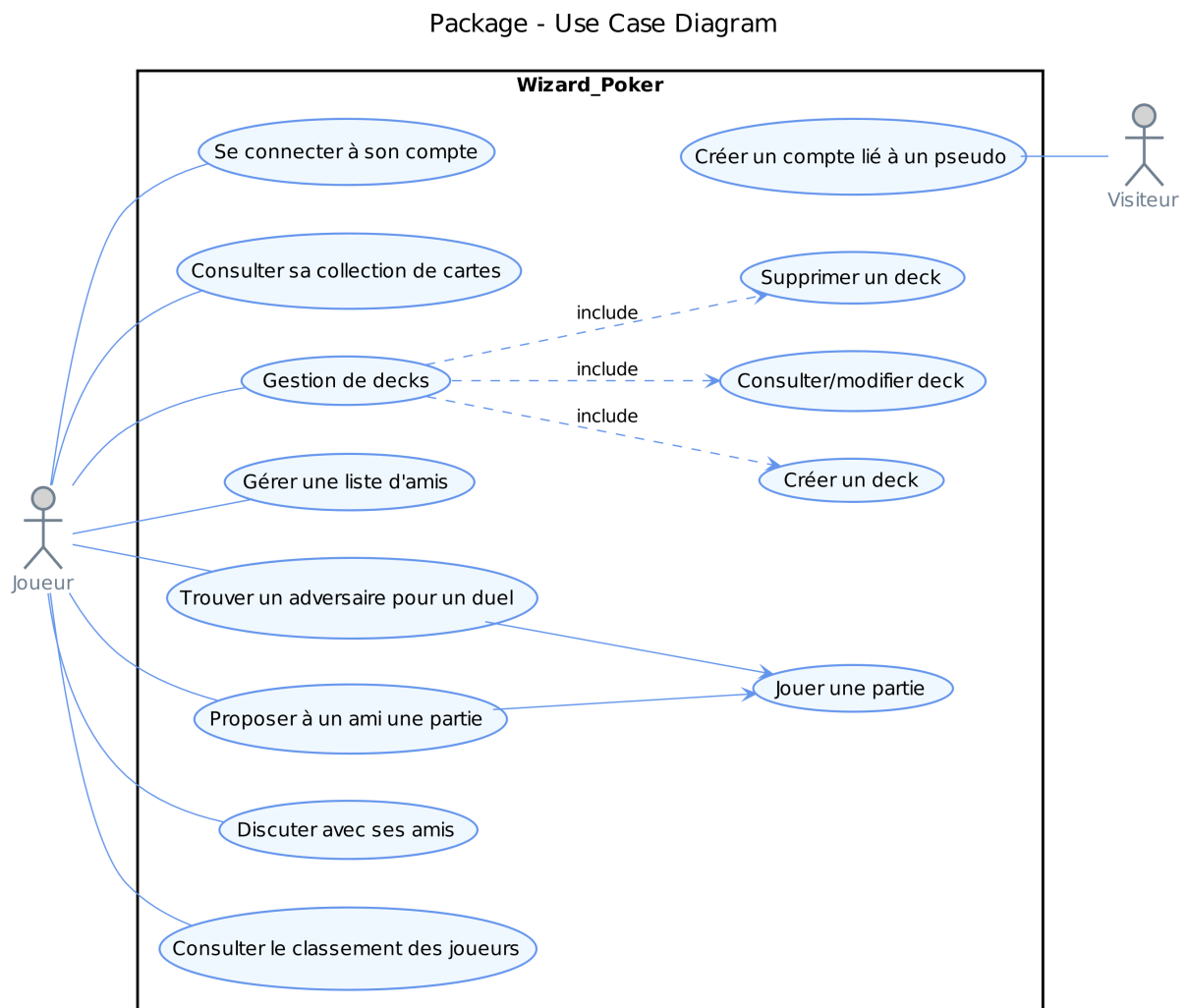


FIGURE 2 – Use Case Diagram du jeu en question

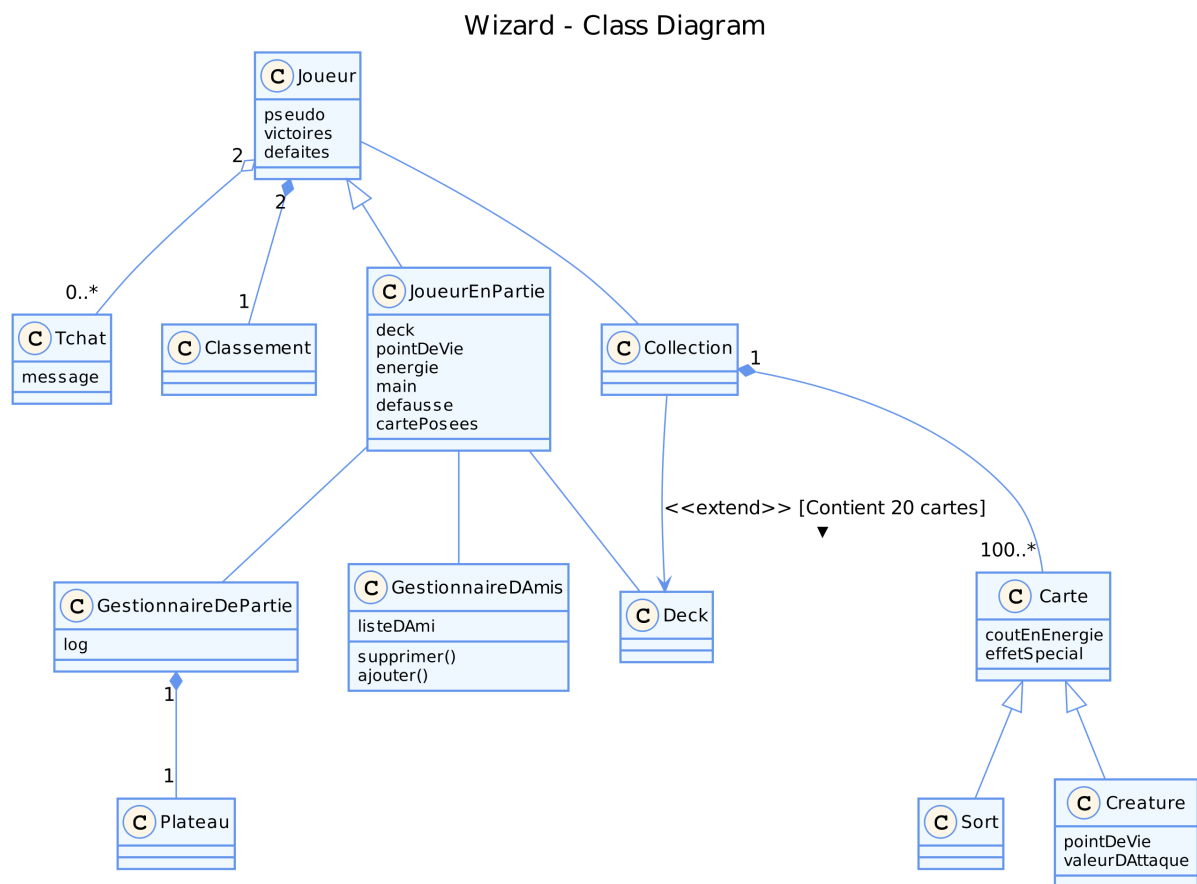


FIGURE 3 – Diagramme des classes

Wizard - Activity Diagram

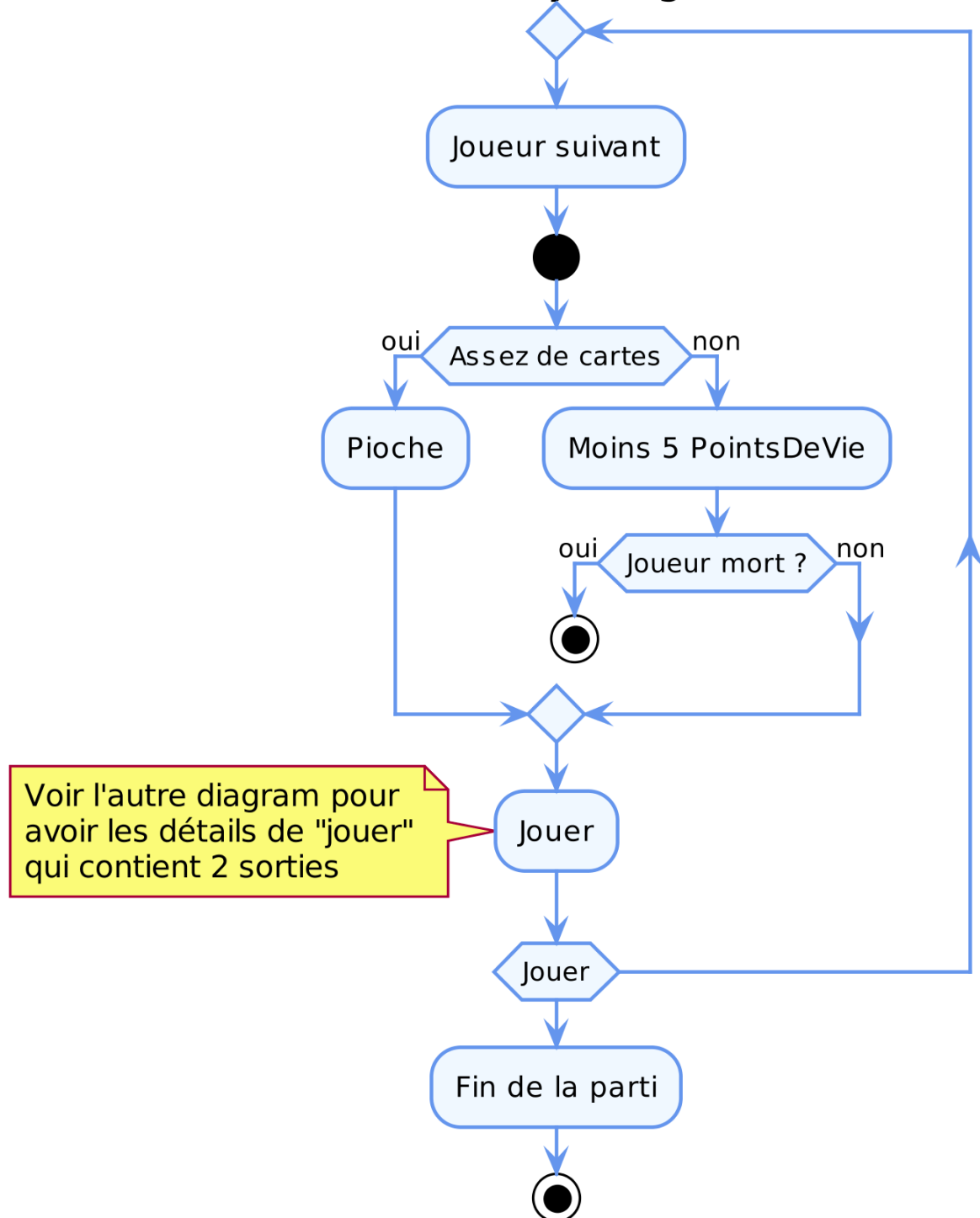


FIGURE 4 – Diagramme d'activité

Wizard - Activity Diagram (fin de tour de jeu d'un joueur)

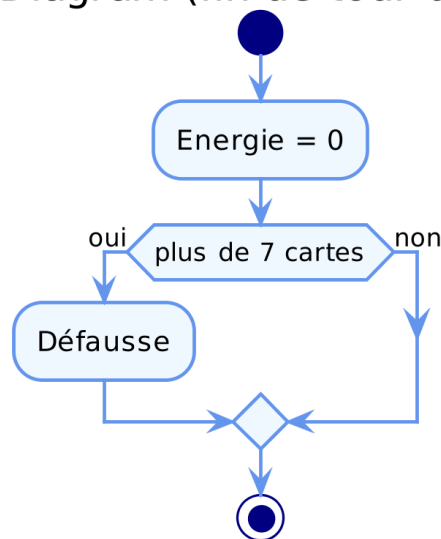


FIGURE 5 – Diagramme d'activité 2

Wizard - Sequence Diagram

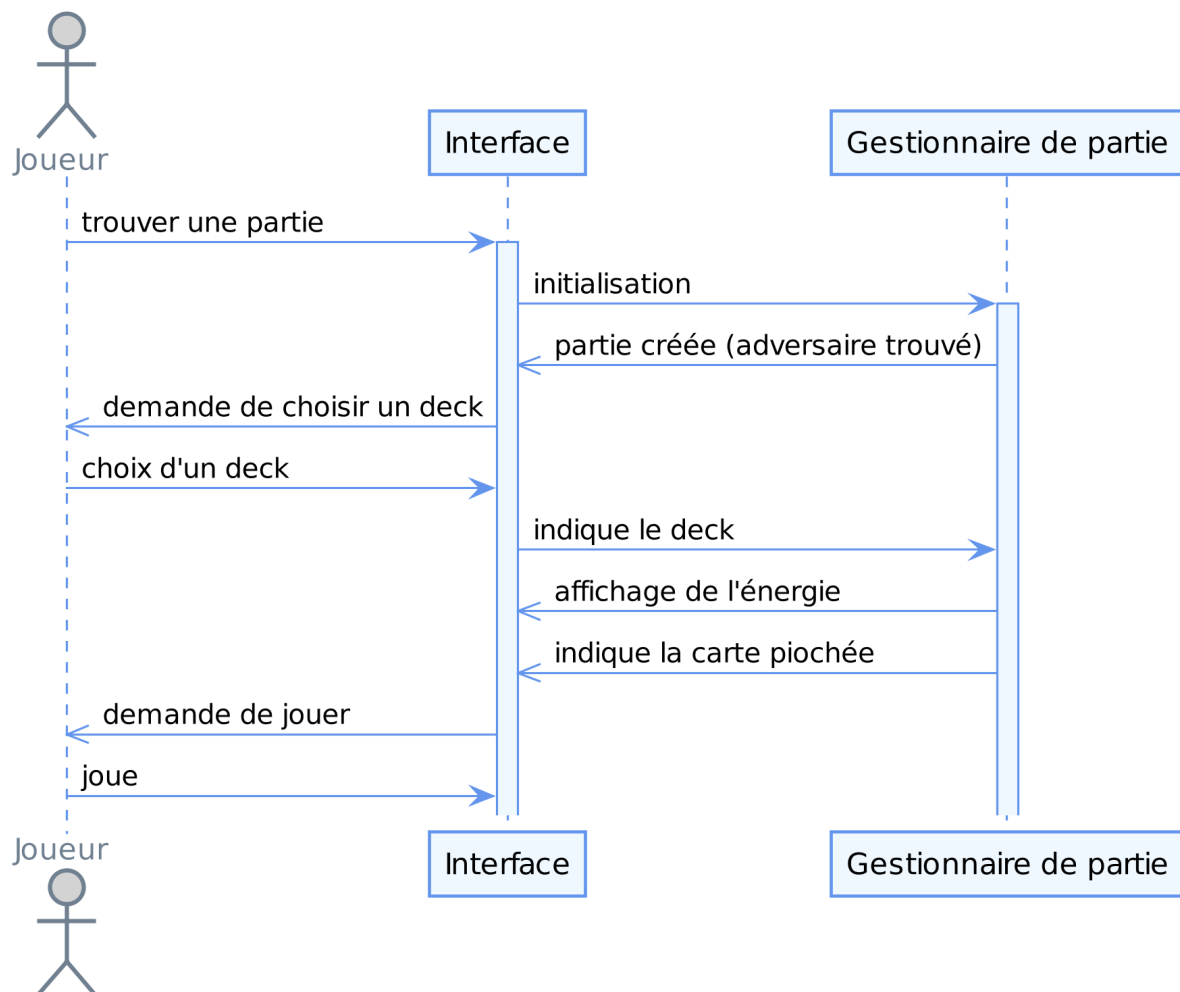


FIGURE 6 – Diagramme de séquence

Index

deck, 4