

Conception du jeu de société Ave César

Dans le cadre du module :
Conception Orienté Objet et UML

Réalisé par :

- Rania AIT CHABANE
- Najat WADDIZ
- Ilias TADLAOUI
- Anass MEHDAOUI

Master MIAGE
Promotion 2022/2023

Table des matières

1) Introduction :	4
2) Présentation du jeu :	4
3) Les diagrammes de conception :	4
1) Diagramme de cas d'utilisation	4
2) Diagramme de classes	6
3) Diagramme d'États-transitions	7
4) Diagramme de séquence	10
5) Diagramme d'activité	13
6) Diagramme de paquetage	23
7) Diagramme de déploiement	23
4) Diagramme de Gantt :	25
5) Conclusion :	25

Table des figures

Figure 1 Diagramme de cas d'utilisation	5
Figure 2 : Diagramme de classes et OCL.....	7
Figure 3: Diagramme Tour.....	7
Figure 4 : Diagramme Course	8
Figure 5 : Diagramme Tournoi.....	8
Figure 6 : Diagramme Pièce.....	9
Figure 7: Diagramme Case.....	9
Figure 8 : Diagramme Joueur.....	10
Figure 9 : Diagramme de séquence	12
Figure 10 : Diagramme d'activité Inscription	13
Figure 11 : Diagramme d'activité Connexion.....	14
Figure 12 : Diagramme d'activité création tournoi	15
Figure 13 : Diagramme d'activité Répondre à une invitation.....	16
Figure 14 : Diagramme d'activité distribution des outils de jeu	17
Figure 15 : Diagramme d'activité piocher carte de positionnement	18
Figure 16 : Diagramme d'activité gestion des cartes pendant le jeu.....	19
Figure 17 : Diagramme d'activité se déplacer	20
Figure 18 : Diagramme d'activité saluer César.....	21
Figure 19 : Diagramme d'activité qualifier/disqualifier joueur	22
Figure 20 : Diagramme de paquetage	23
Figure 21 : Diagramme de déploiement	24
Figure 22 : Diagramme de Gantt.....	25

1) Introduction :

Dans le cadre du module Conception Orienté Objet et UML, il nous a été demandé de concevoir un jeu de société en ligne.

L'idée derrière ce projet est de bien manipuler les différents diagrammes vus en cours, et de les appliquer à un cas réel.

Pour cela, nous avons choisi le jeu Ave César.

2) Présentation du jeu :

Trois à six conducteurs de char s'affrontent au cours d'une course de chars qui se déroule dans les arènes du très célèbre Circus Maximus.

Les joueurs doivent utiliser astucieusement leurs cartes pour avancer dans leur course. Celui qui, au bout de trois tours, aura réussi à passer le premier la ligne d'arrivée, sera déclaré gagnant et récoltera le plus de lauriers (de points).

Enfin, celui qui au bout de plusieurs courses (nous en proposons quatre) aura gagné le plus de lauriers, sera déclaré vainqueur du tournoi : tous les honneurs lui reviendront de droit !

3) Les diagrammes de conception :

1) Diagramme de cas d'utilisation

Pour la conception du jeu, nous avons 2 acteurs, le joueur et le système, chacun ayant un certain nombre de fonctionnalités qu'ils font au cours du jeu.

Diagramme de cas d'utilisation

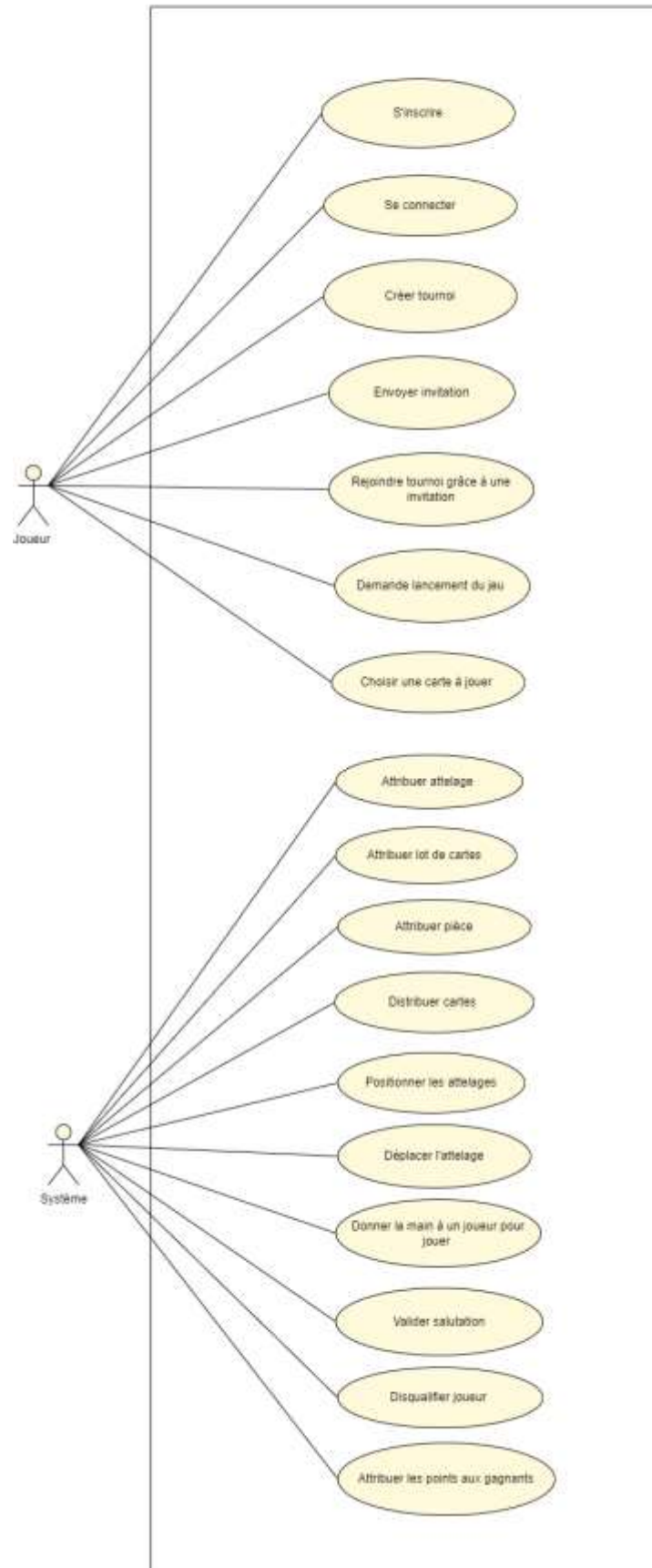


Figure 1 Diagramme de cas d'utilisation

2) Diagramme de classes

Nous avons commencé par définir une classe tournoi qui comporte 4 courses, c'est le nombre de courses recommandé, et une course qui comporte elle-même 3 tours.

Le joueur participe à un tournoi, le système lui affecte 24 cartes au départ nécessaire pour effectuer les 3 tours, et il a constamment 3 cartes en main. Cependant à la fin du troisième tour, le joueur aura moins de cartes jusqu'à ne plus en avoir, d'où la cardinalité 0..3 entre joueur et carte pour l'association 'en main'.

Le joueur a également une pièce avec un attribut booléen remise, qui a au départ à faux et devient vrai lorsque le joueur salue César.

L'attribut rang_course de la classe Joueur permet de savoir le rang du joueur lors de la course précédente et l'attribut position permet de connaître sa position en temps réel pendant la course (si en tête de liste ou plus en arrière)

Le joueur déplace un attelage dans les cases de la piste.

Une case est définie par un numéro de case, un état (libre ou pas), et une liste de cases dans lesquels on peut se déplacer à partir de la case en question.

Nous avons également un plateau de jeu qui contient des cases.

Note sur le mode de jeu : notre application ne propose que le mode réseau, qui est un mode multijoueur distant car, les joueurs ne doivent pas pouvoir voir les cartes des autres joueurs. Le mode local n'est donc pas possible.

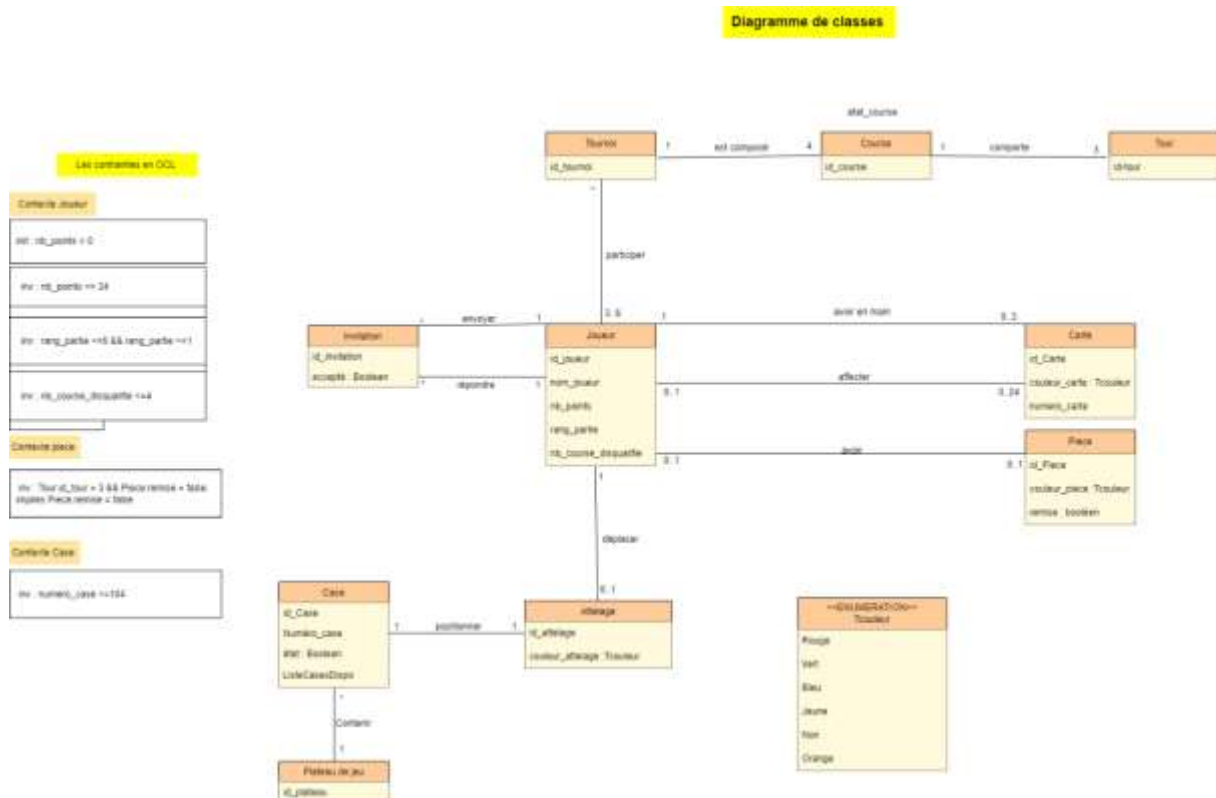


Figure 2 : Diagramme de classes et OCL

3) Diagramme d'États-transitions

Pour réaliser le jeu Ave César nous avons défini 7 diagrammes d'état transition pour les classes : Tournoi, Tour, Course, Pièce, Joueur et case.

Diagramme tour

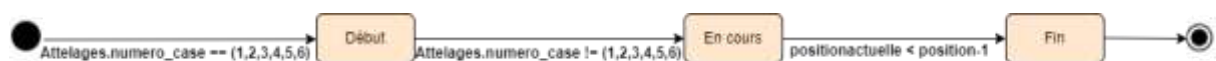


Figure 3: Diagramme Tour

Le diagramme d'état transition Tour est composé de trois états : Début, En cours et Fin. L'état initial de ce diagramme est Début quand tous les attelages sont positionnés dans la case 1, 2, 3, 4, 5 ou 6

Pour basculer de l'état initial à l'état en cours il faut que le numéro de case d'au moins un attelage soit différent de 1, 2, 3, 4, 5 ou 6.

Pour basculer de l'état En cours à l'état Fin il faut que le numéro de case ou les attelages sont positionnés soit inférieure au numéro de case de leur position précédente.

Diagramme Course



Figure 4 : Diagramme Course

Le diagramme d'état transition Course est composé aussi de trois états : Départ, En cours et Arrivée.

- La course est en état *Début* quand le numéro du tour est 1 et tous les attelages sont soit dans la position 1, 2, 3, 4, 5 ou 6
- Pour passer de l'état *Départ* à l'état *En cours* il faut que le numéro de tour soit inférieur à 3 et qu'il existe au moins un attelage qui est positionné dans une case dans le numéro est différent de 1, 2, 3, 4, 5, ou 6
- Pour passer de l'état *En cours* à l'état *Arrivé* il faut que le numéro de tour soit égale à 3 et le numéro de case ou les attelages sont positionnés est inférieur au numéro de case de la position précédente.

Diagramme tournoi



Figure 5 : Diagramme Tournoi

Le diagramme d'état-transition (Tournoi) est composé également de 3 états : Début, En cours et Fin.

- L'état de départ de tournoi est "*Début*" quand le numéro de la Course est égal à un (nous sommes dans la première course) et que tous les attelages sont positionnés soit dans la case 1, 2, 3, 4, 5 ou 6.
- Pour passer de l'état *Début* à l'état *En cours* il faut que le numéro de la course soit inférieur à 4 et qu'il existe au moins un attelage qui est positionné dans une case différente de 1,2,3,4,5 ou 6
- Pour passer de l'état *En cours* à l'état *Fin* il faut que le numéro de la course soit égale à 4 et le numéro de case ou les attelages sont positionnés est inférieur au numéro de case de la position précédente.

Diagramme Pièce

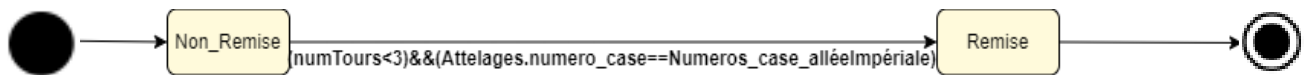


Figure 6 : Diagramme Pièce

Le diagramme d'état transition Pièce contient deux états : non remise et remise.

- La pièce est par défaut *Non remise* parce qu'au début du jeu aucun joueur n'a effectué aucun tour et donc aucun joueur n'a remis la pièce à César.
- Pour que la pièce puisse passer à l'état *Remise* il faut que l'attelage en question ait pu se mettre dans une des cases de l'allée impériale et que le nombre de tours soit inférieur

Diagramme Case

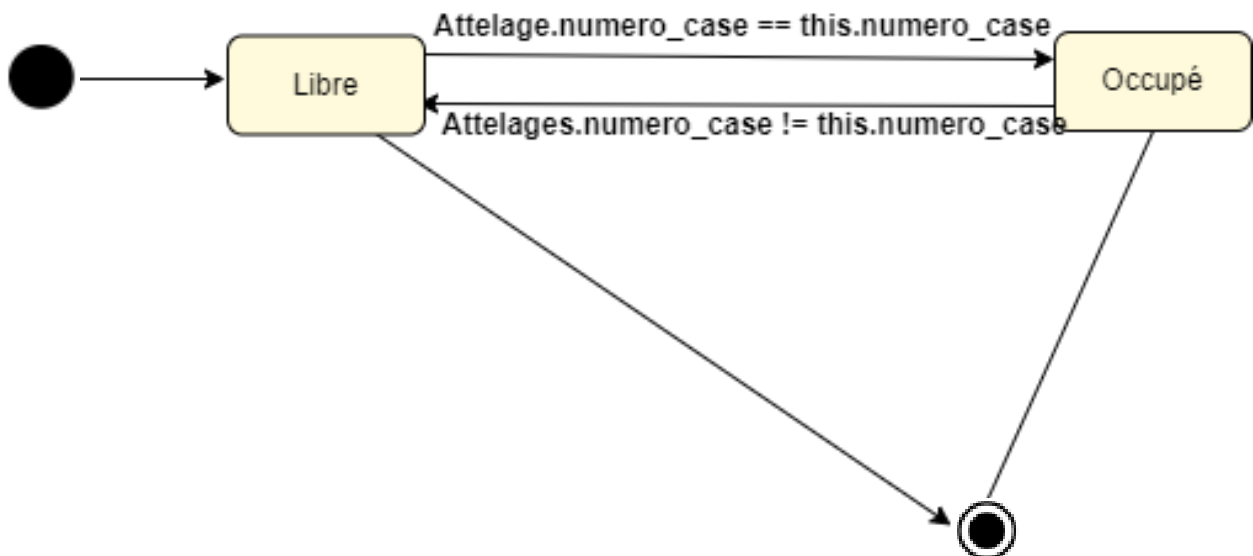


Figure 7: Diagramme Case

Le diagramme d'état transition case est composé de deux états : Libre et occupé.

- La case est d'état *libre* par défaut parce qu'au début du jeu aucun attelage n'est positionné dans aucune case
- Pour passer de l'état *libre* à l'état *Occupé* il faut qu'un attelage se positionne dans la case en question et pour pouvoir ensuite retourner à l'état libre il faut que le numéro de case actuelle de tous les attelages soit différent au numéro de case en question

Diagramme Joueur

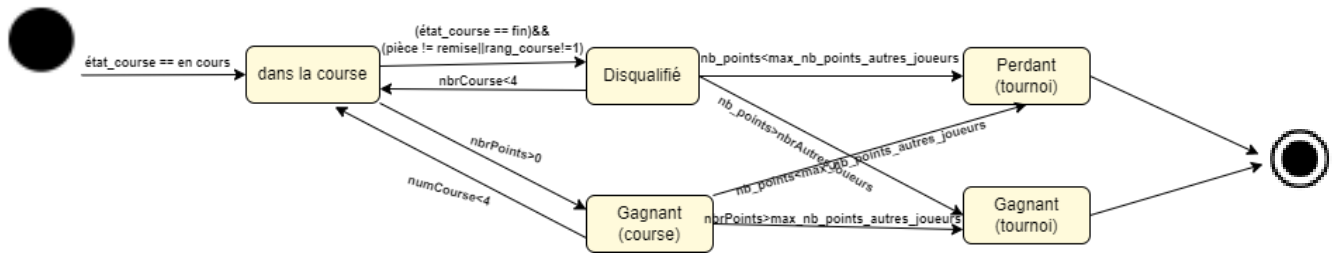


Figure 8 : Diagramme Joueur

Le diagramme Joueur contient 5 états : dans la course, Disqualifié, Gagnant de la course, perdant et gagnant du tournoi.

- Pour que le Joueur soit dans l'état *Dans la course* il faut que l'état de la course soit en cours
- Pour passer de l'état *Dans la course* à l'état *Disqualifié* il faut que l'état de la course soit égale à fin et la pièce est non remise ou le joueur n'est pas le premier à franchir la ligne d'arrivée
- Pour que le joueur passe de l'état *Dans la course* à l'état *Gagnant* il faut que le nombre de ses points soit supérieur à 0
- Pour passer de de l'état *Disqualifié* à l'état *Dans la course* ou de l'état *Gagnant* à l'état *dans la course* il faut que le numéro de la course soit inférieur à 4
- Pour passer de *disqualifiant* ou *Gagnant* de la course à *Gagnant* il faut que le nombre de points du joueur en question soit supérieur au nombre de points de tous les autres joueurs
- Pour passer de *disqualifier* ou de *gagnant* de la course à *Perdant* il faut que le nombre de points du joueur en question soit inférieur au nombre de points d'au moins un autre joueur

4) Diagramme de séquence

Tout commence lorsque le joueur s'inscrit, puis se connecte.

Afin de jouer, il doit soit créer un tournoi, ou être invité à un tournoi.

Une fois que les joueurs invités se connectent, le jeu est lancé, et le système distribue les attelages, les pièces et les cartes. Concernant le positionnement, nous avons légèrement modifié la règle de positionnement de départ, le système distribue des cartes de valeurs de 1 à n, n étant le nombre de joueurs.

De cette manière, le positionnement des attelages sera aléatoire et il n'y aura pas de conflit de cartes de même valeur pour plusieurs joueurs.

Ensuite, le jeu commence, le joueur choisit une carte à jouer, si il y a un chemin possible avec le numéro de carte choisi, alors le système déplace l'attelage de joueur et donne la main ou joueur suivant, sinon le joueur passe son tour au suivant directement.

Concernant la salutation de César, il suffit que le joueur s'arrête sur l'allée impériale, que ce soit fait dans le premier ou second tour et qu'il ait une pièce pour que la salutation soit comptée.

Si c'est le troisième tour et que la pièce n'est pas remise alors le joueur est disqualifié.

Les points gagnés dans le premier tour déterminent le positionnement de départ pour le tour d'après.

Diagramme de séquence

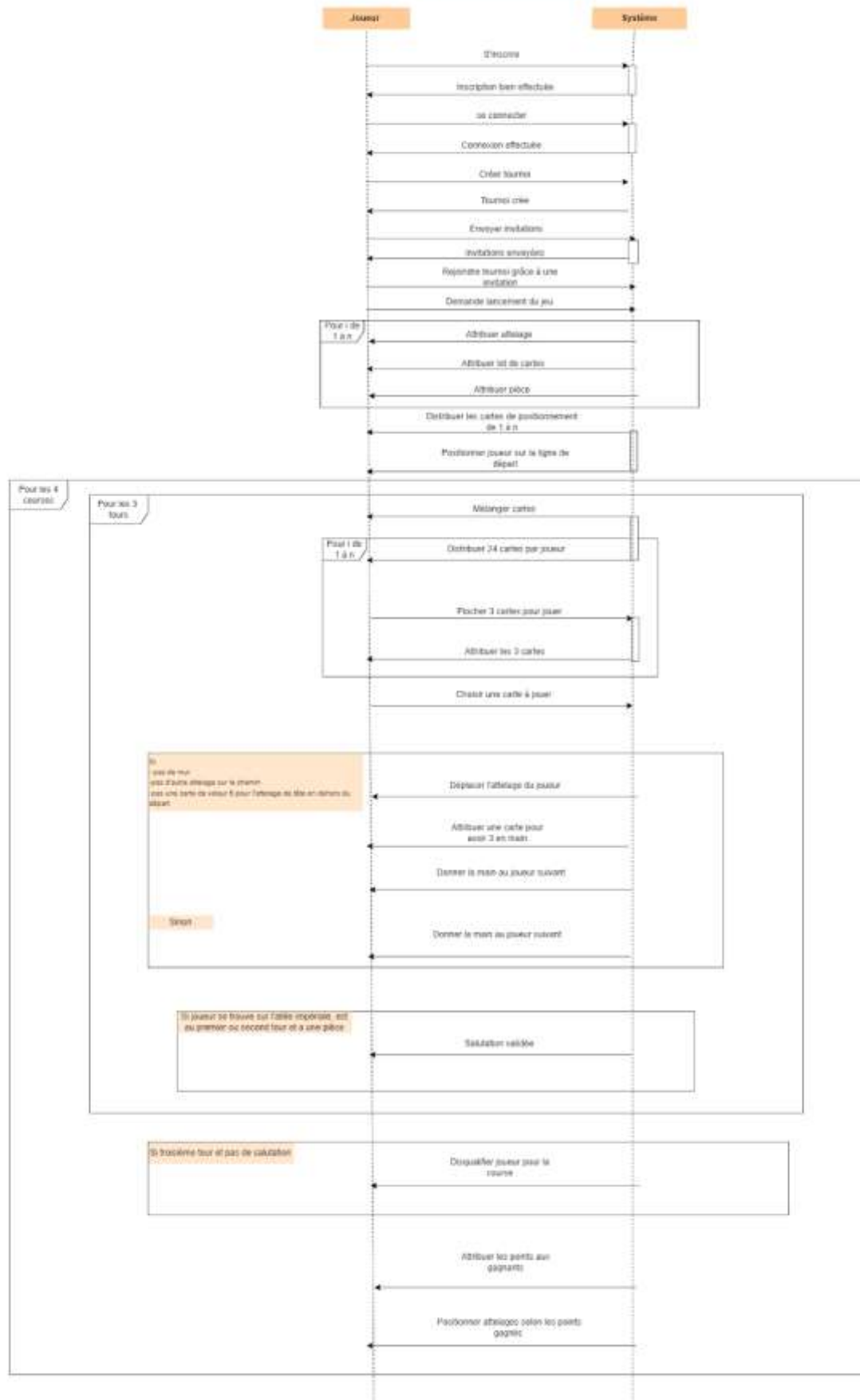


Figure 9 : Diagramme de séquence

5) Diagramme d'activité

Le diagramme d'activité fournit une vue détaillée du comportement d'un système en décrivant la séquence d'actions d'un processus. C'est pour cela que nous avons commencé par identifier les diagrammes nécessaires pour bien modéliser notre application, qui sont :

1. Inscription

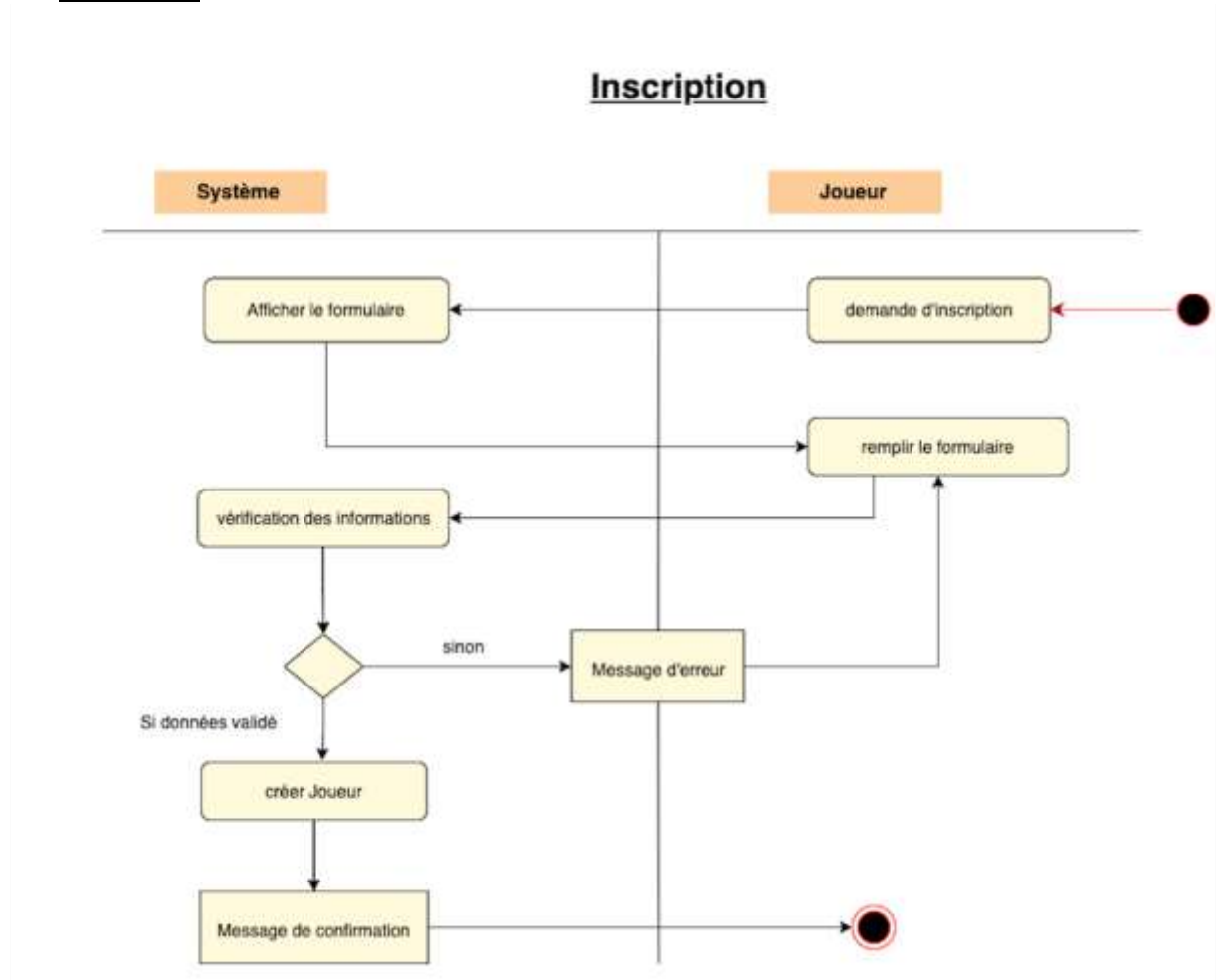


Figure 10 : Diagramme d'activité Inscription

Lors de l'inscription, le joueur fait la demande, puis il doit remplir un formulaire qui doit être validé par le système en respectant les contraintes, comme (le format de l'adresse email, mot de passe conforme, etc.). Sinon, le joueur reçoit un message d'erreur et doit modifier le formulaire pour que l'activité soit validée.

2. Connexion

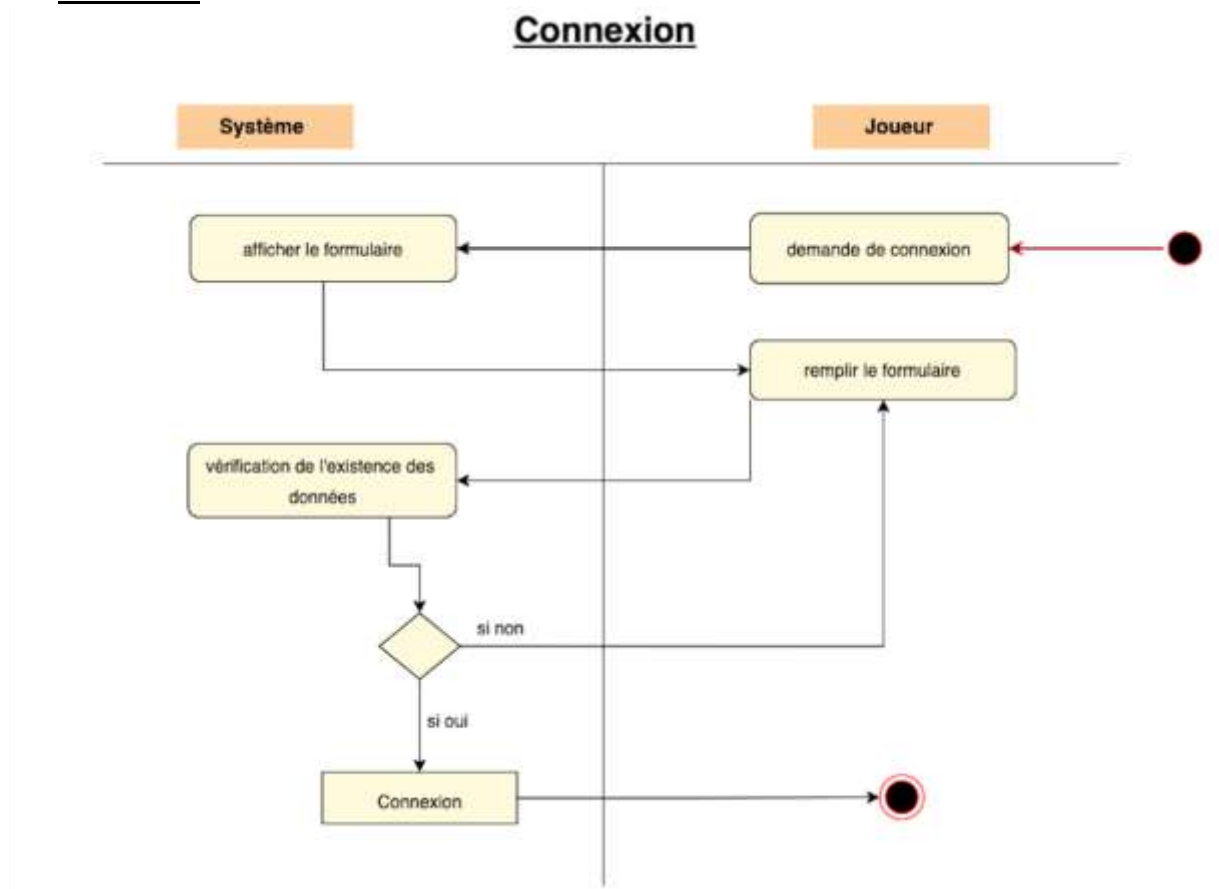


Figure 11 : Diagramme d'activité Connexion

Dans cette partie, il faut vérifier que les informations saisies par le joueur pour se connecter existent dans la base de données, sinon l'action ne pourra pas être validée.

3. Création de tournoi

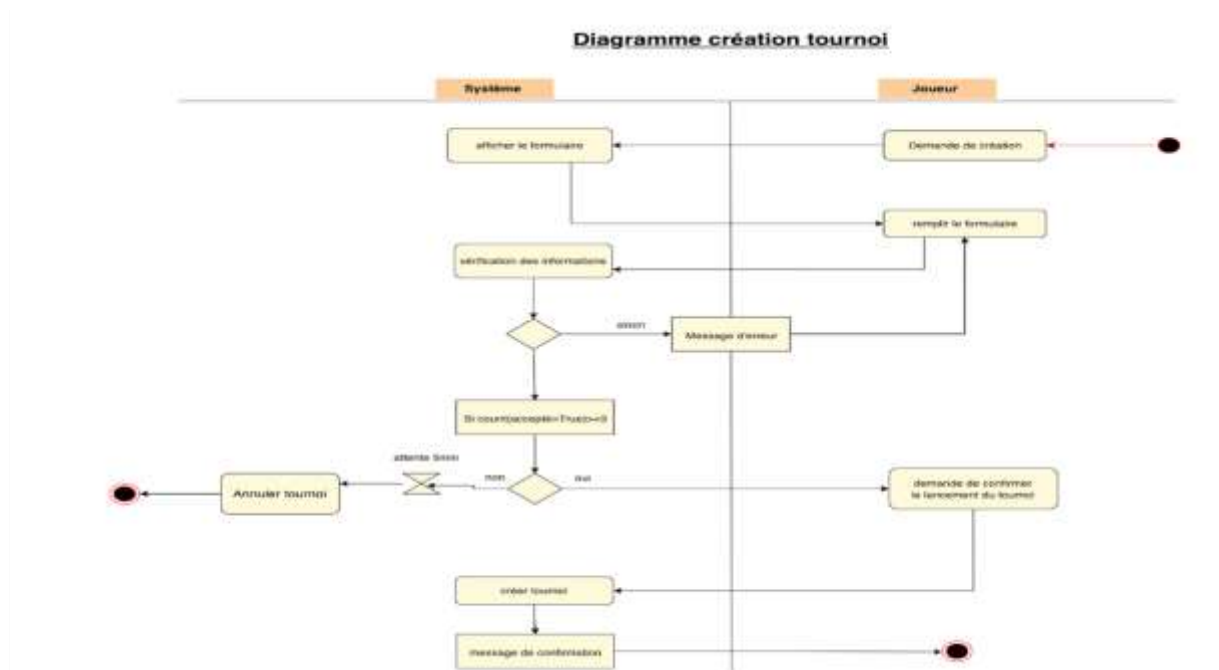


Figure 12 : Diagramme d'activité création tournoi

Après la phase de connexion, il faut normalement jouer et pour cela, un joueur doit créer un tournoi et inviter des personnes à rejoindre le salon. Et donc il y a des contraintes à vérifier, qui sont que le nombre de joueurs doit être compris entre 3 et 6, ainsi que le système doit vérifier que l'ID du joueur que l'on souhaite inviter existe dans la base, sinon un message d'erreur va s'afficher.

Si les invitations ont été envoyées, il faut attendre qu'au moins trois joueurs acceptent l'invitation pour pouvoir lancer le tournoi, sinon il y aura un délai de cinq minutes d'attente. Dépassé ce délai, le tournoi sera directement annulé.

4. Répondre à une invitation

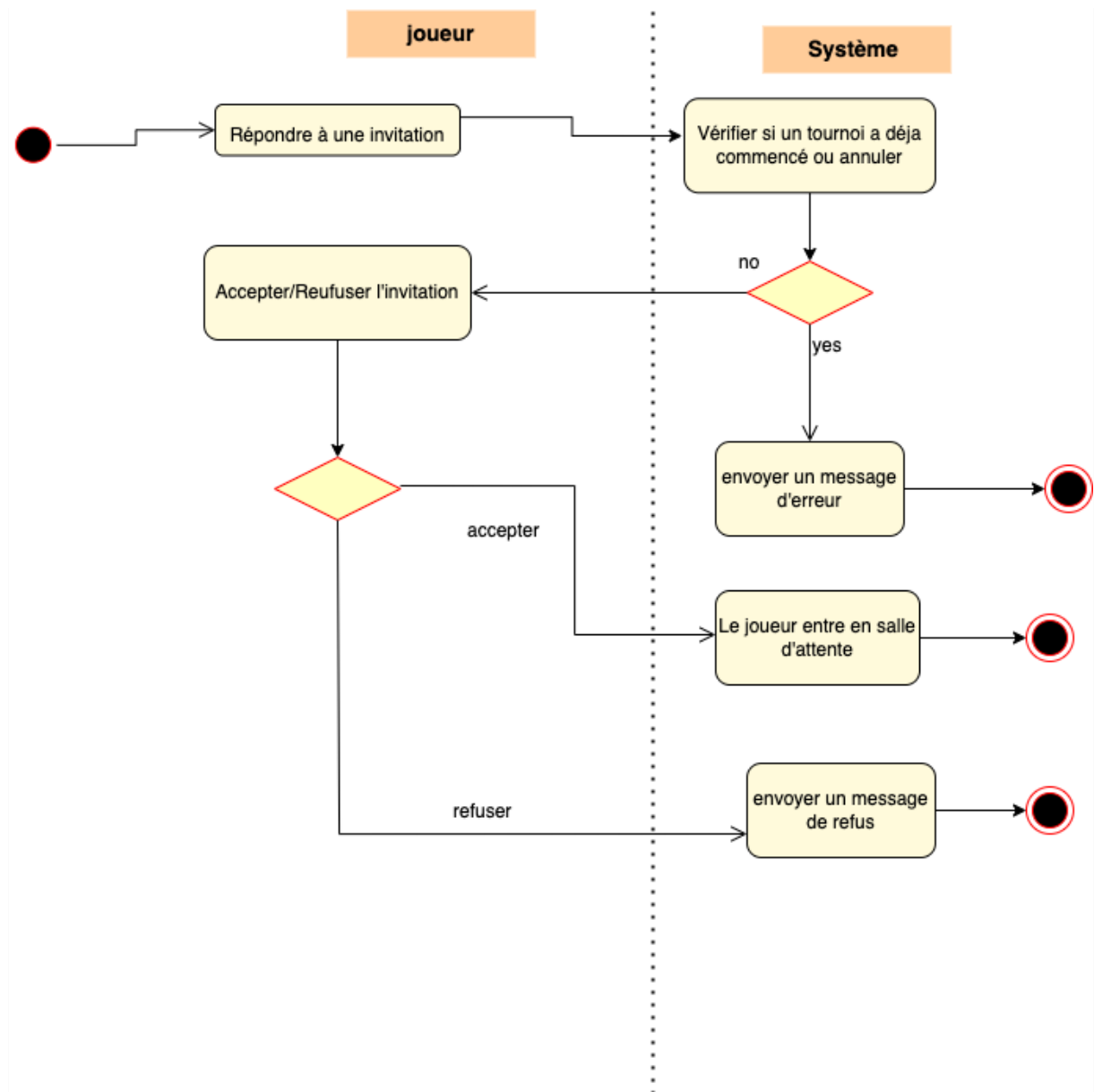


Figure 13 : Diagramme d'activité Répondre à une invitation

Pour répondre à une invitation, on vérifie d'abord si le tournoi choisi a déjà commencé ou a été annulé, si ce n'est pas le cas, le joueur peut accepter ou refuser la demande d'invitation.

5. Distribution d'outils de jeu

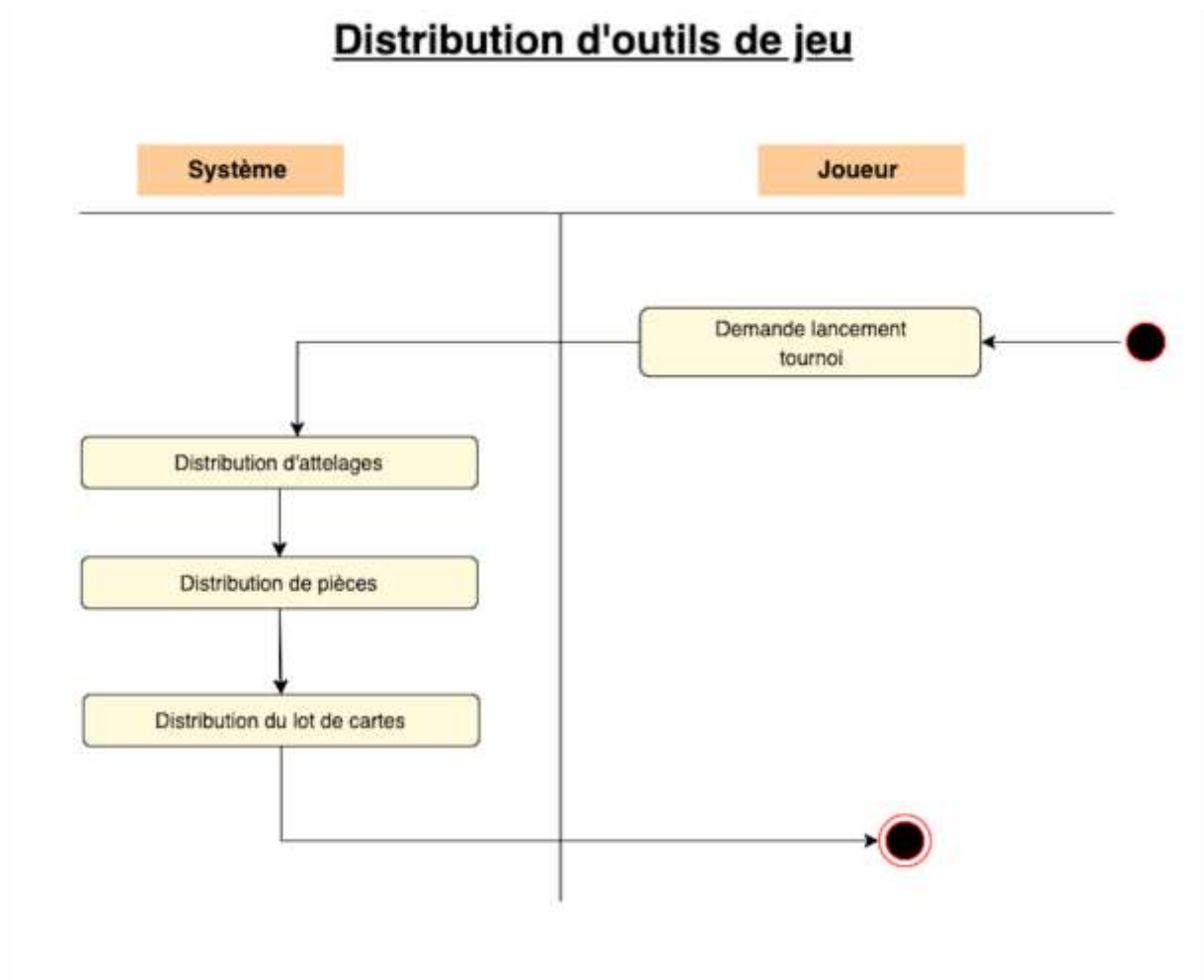


Figure 14 : Diagramme d'activité distribution des outils de jeu

Une fois le tournoi créé, le système affecte aléatoirement pour chaque joueur une couleur et distribue les outils pour jouer, qui sont (attelages, pièces, paquet de 24 cartes) en respectant les couleurs affectées.

6. Piocher carte de positionnement

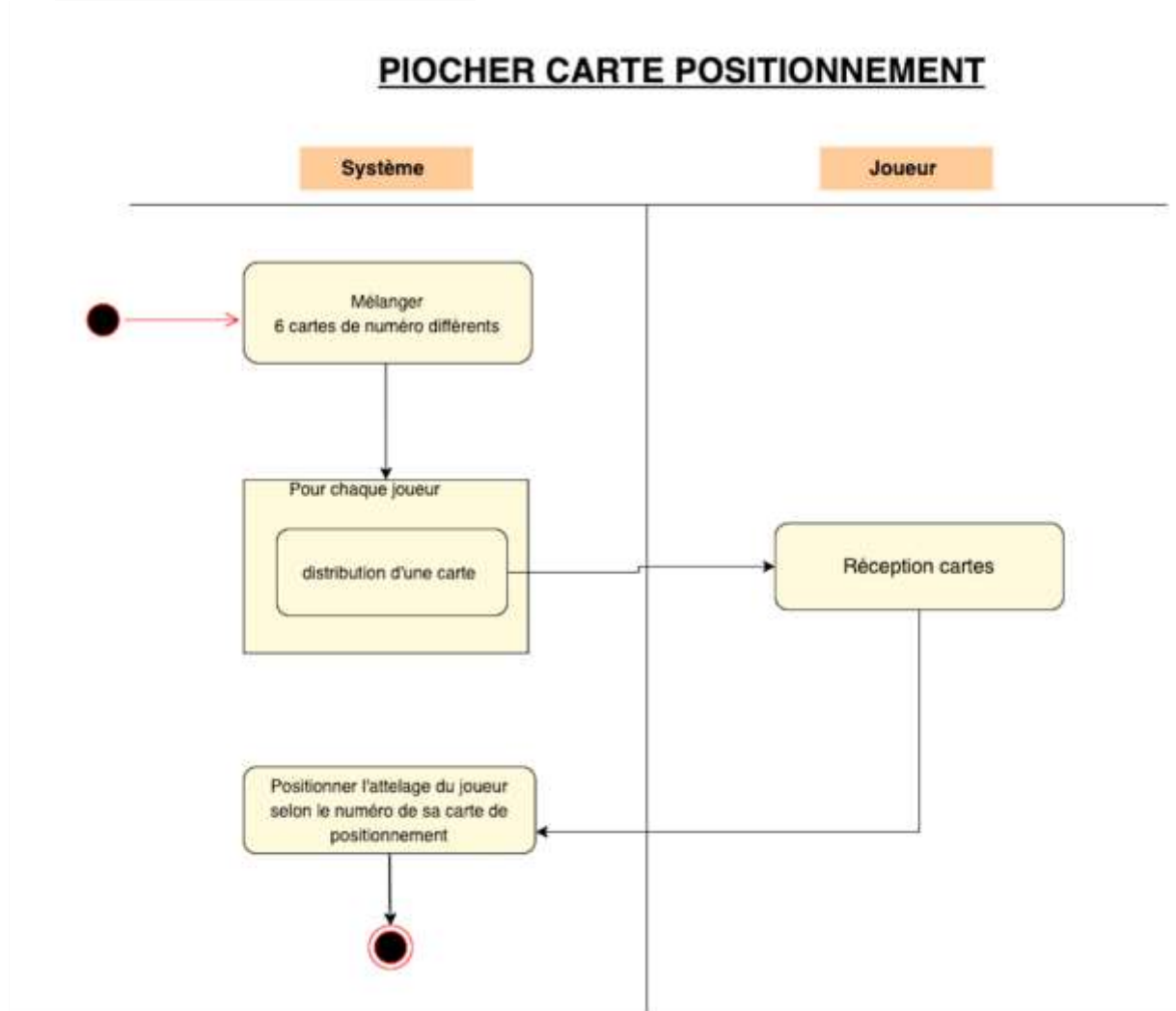


Figure 15 : Diagramme d'activité piocher carte de positionnement

Afin de donner à chaque participant un ordre de jeu, le système va prendre n cartes avec des numéros différents qui doivent être distribués aléatoirement aux n joueurs et les positionner sur la ligne de départ en respectant le numéro dont ils ont été affectés.

7. Gestion de carte pendant le jeu

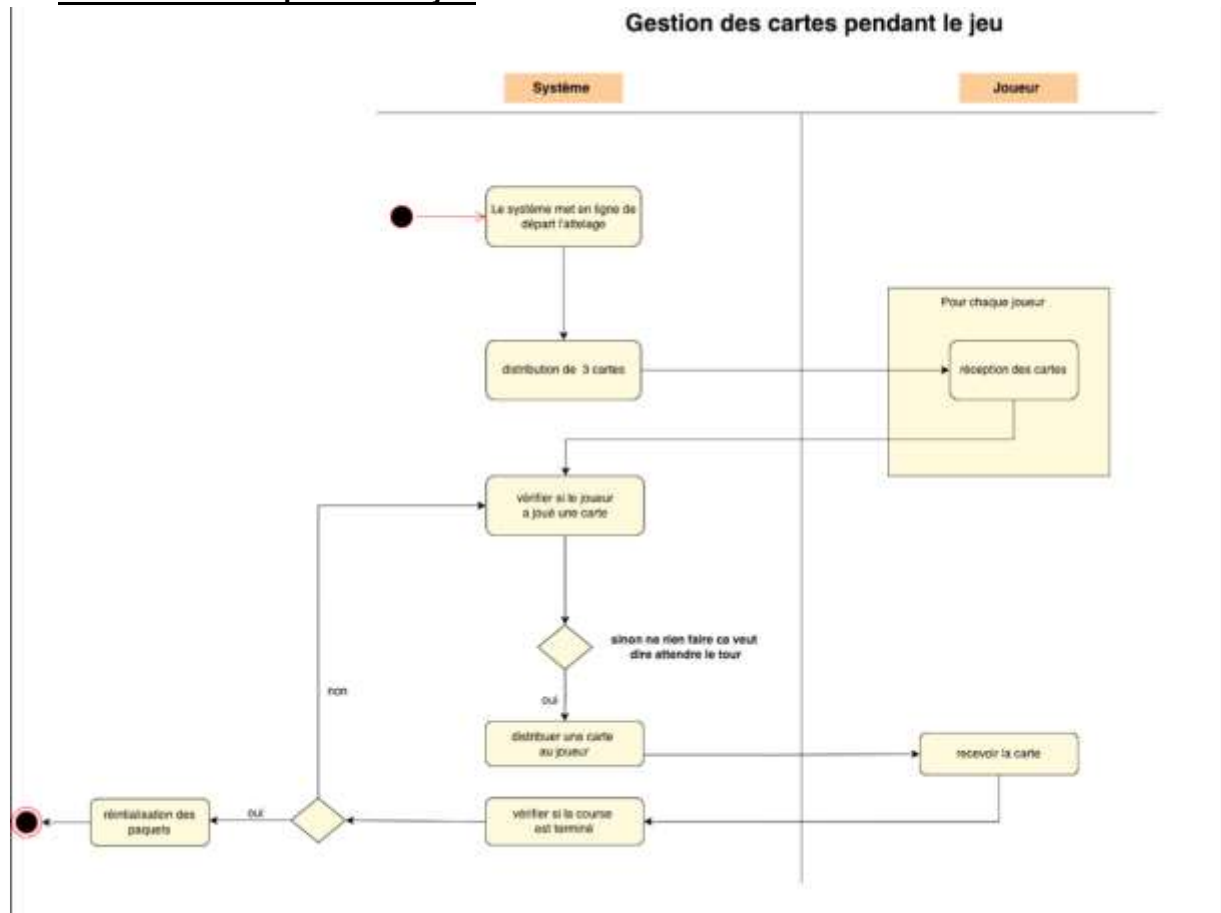


Figure 16 : Diagramme d'activité gestion des cartes pendant le jeu

Le système distribue 3 cartes du paquet de chaque joueur une fois qu'ils ont un numéro de positionnement.

Ensuite, à chaque fois que le joueur joue, le système lui donne une carte pour s'assurer qu'il a toujours 3 cartes en main jusqu'à ce que la course soit terminée et, dans ce cas, le système réinitialise le paquet de chaque joueur.

8. Se déplacer

Pour se déplacer d'une case à une autre, on vérifie d'abord la règle que l'attelage de tête n'est pas autorisé à avancer de six cases à la fois (au cas où une carte de valeur 6 est tirée) sauf au départ. Après on vérifie si un autre attelage se trouve dans une case \leq +numero (carte Choisie), n étant la position actuelle. Ensuite, on vérifie si un mur se trouve en dessous ou en dessous dans les cases suivantes. Si ce n'est pas le cas, le joueur avance du nombre de cases indiqué par la valeur de la carte choisie dans le chemin choisi, sinon on passe au joueur suivant.

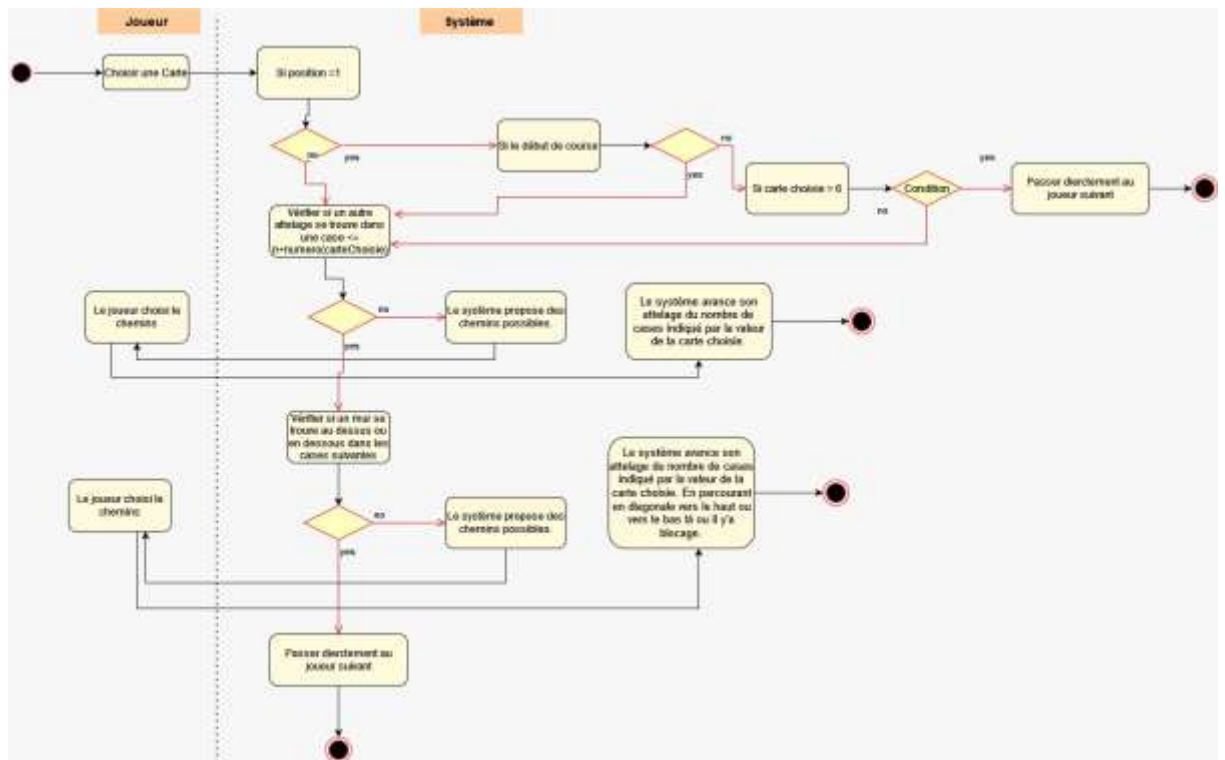


Figure 17 : Diagramme d'activité se déplacer

9. Saluer César

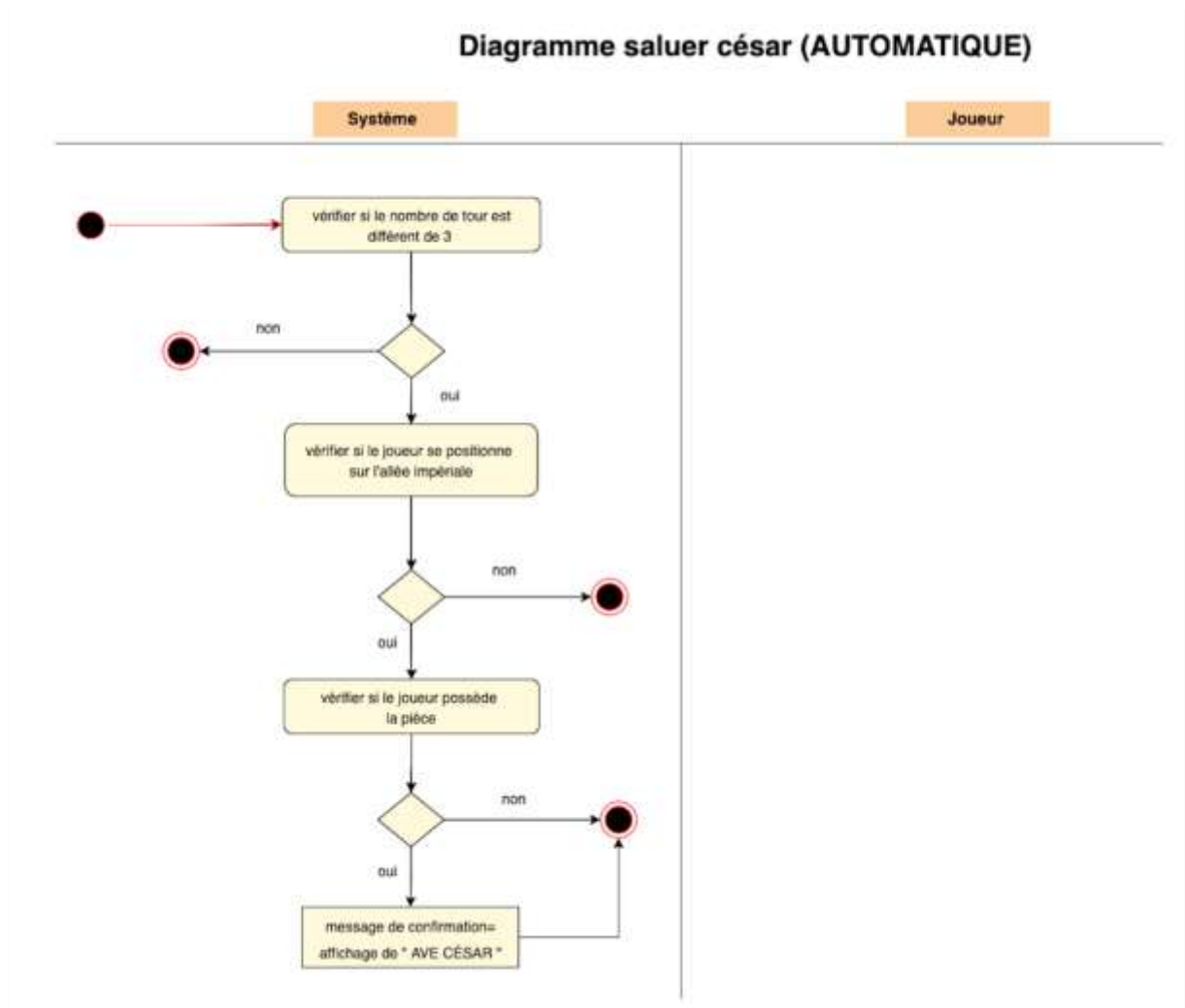


Figure 18 : Diagramme d'activité saluer César

Pendant la course, le système vérifie si le nombre de tours est différent de 3. Alors, il faut vérifier si le joueur se positionne sur l'allée impériale. Si c'est le cas, il faut que le joueur possède la pièce pour pouvoir valider sa salutation, car le joueur peut passer par l'allée impériale pendant les deux tours s'il le souhaite afin de bloquer le passage aux autres.

10. Qualifier ou disqualifier un joueur

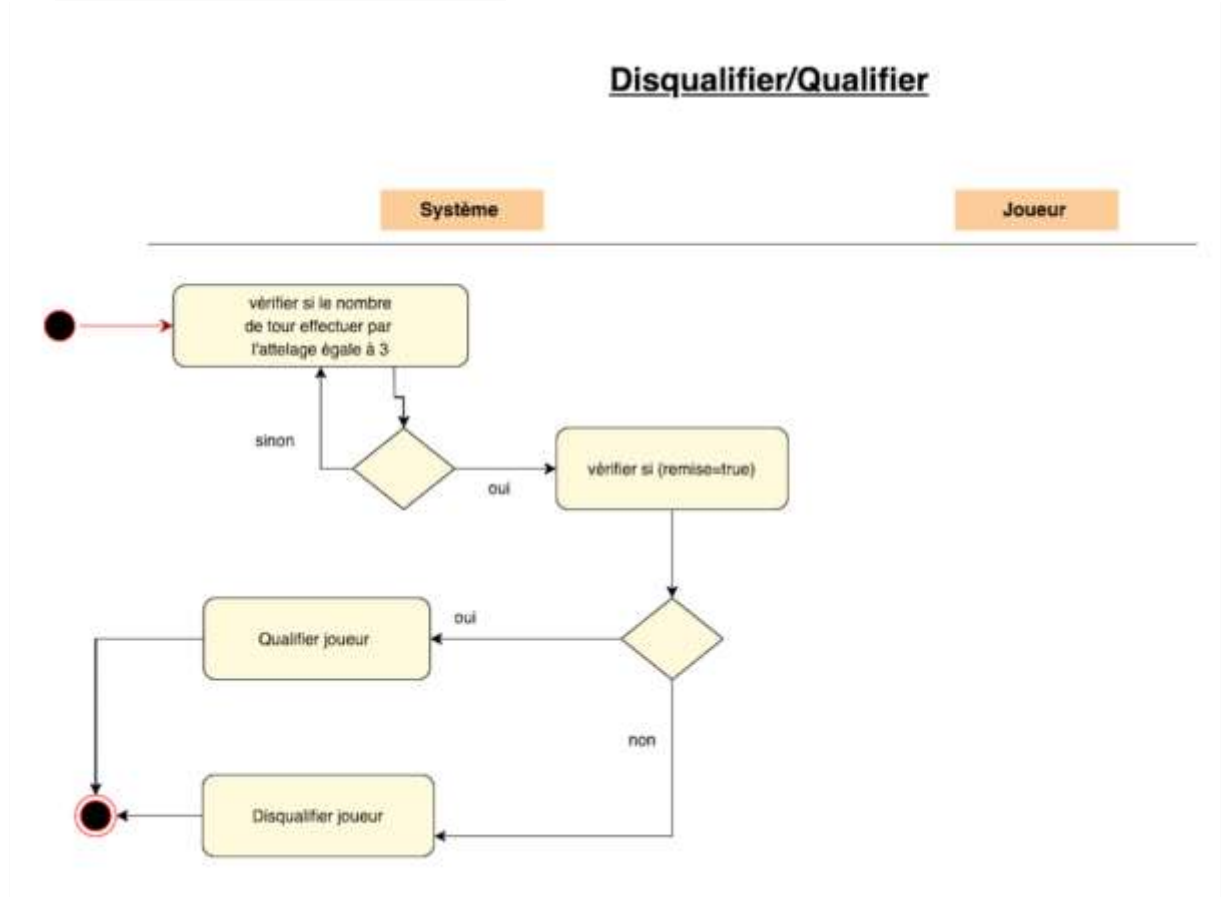


Figure 19 : Diagramme d'activité qualifier/disqualifier joueur

Pour pouvoir qualifier ou disqualifier un joueur, il faut vérifier pendant la course si le joueur est au dernier tour, c'est-à-dire le 3ème. Ensuite, s'il ne possède plus la pièce, on peut dire que le joueur est qualifié et peut avoir ses points normaux en fonction de sa position d'arrivée. Sinon, il est disqualifié et n'aura aucun point.

6) Diagramme de paquetage

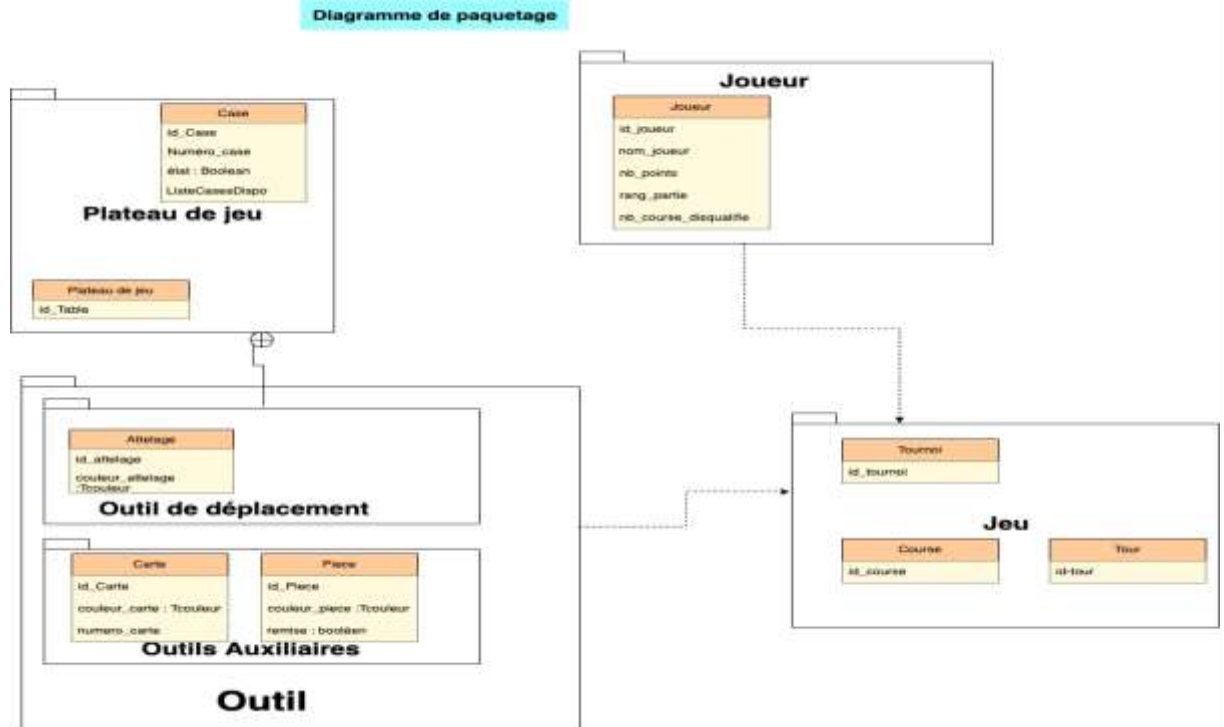


Figure 20 : Diagramme de paquetage

Ce diagramme permet de regrouper d'éléments UML apparentés, ils sont le plus souvent utilisés pour simplifier les diagrammes de classes complexes sous une forme visuelle bien ordonnée. C'est pour cela que nous avons regroupé chaque classe à l'intérieur du package qu'il convient. Nous avons également choisi de mettre une relation de contenance entre le package outil de déplacement et plateau de jeu puisque le plateau de jeu contient des attelages qui se déplacent dans des cases.

7) Diagramme de déploiement

Ce diagramme modélise l'architecture physique d'un système. Il affiche les relations entre les composants logiciels et matériels du système, et la distribution physique du traitement.

L'utilisateur se connecte à un navigateur internet dans un PC portable, après il peut s'inscrire ou se connecter dans notre application, pour jouer dans une interface qui est le plateau de jeu.

Le PC portable est lié avec le serveur grâce à un protocole TCP/IP, et ce serveur contient le paquetage plateau de jeu, inscription/connexion, jeu et les outils.

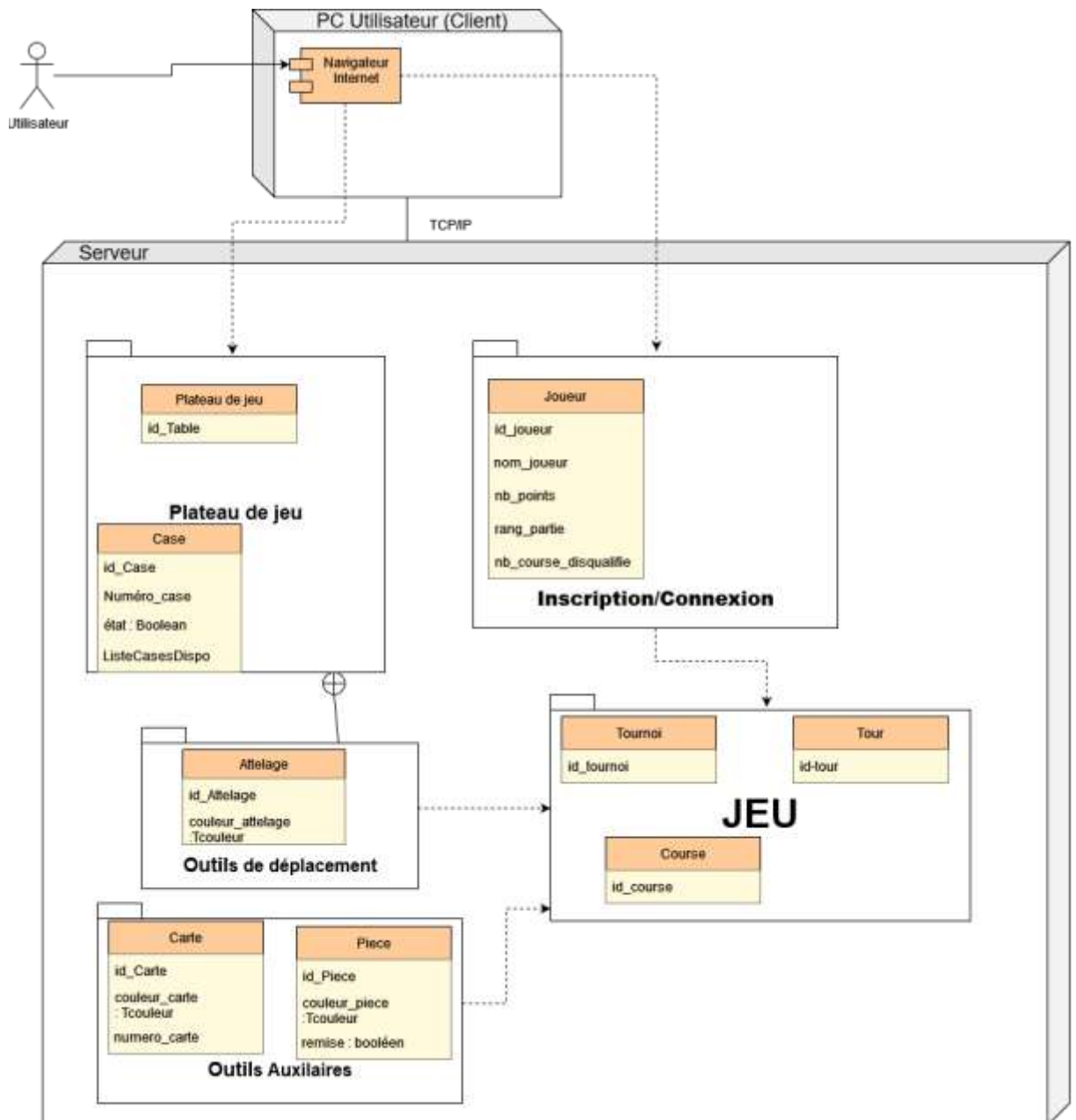


Figure 21 : Diagramme de déploiement

4) Diagramme de Gantt :



Figure 22 : Diagramme de Gantt

5) Conclusion :

En conclusion, ce projet a été bénéfique pour mettre en pratique les compétences théoriques acquises tout au long du semestre.
Nous avons également développé nos compétences relationnelles et notre esprit de travail en équipe.