Crépin Aurélia, Levrat Gael, Ribierre Paul, Sagon Nicolas Veys Arthur

CPE Groupe E

10/11/2014

Dossier de Conception

Projet : Les trésors de Java

#### Objet du document :

Ce document vise à décrire la conception du logiciel fonction par fonction

#### Responsabilité du document :

|  |  |  |
| --- | --- | --- |
| Rédacteurs | Vérificateurs | Validateur |
| CREPIN Aurélia - Expert Environnement et Qualité | CREPIN Aurélia - Expert Environnement et Qualité | Comité de projet |
| LEVRAT Gaël - Expert  Modélisation | RIBIERRE Paul - Chef de Projet |  |
| RIBIERRE Paul - Chef de Projet  (Gestion et Management de projet) |  |  |
| SAGON Nicolas - Expert Programmation Objet |  |  |
| VEYS Arthur - Expert Technique et Environnement |  |  |

#### Historique des modifications et validation du document :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Parties modifiées | Nature du  changement | Statut |
| V0 | 04/11/2014 |  | Création | Obsolète |
| V1 | 11/11/2014 |  | Modification | Applicable |

#### Diffusion du document :

|  |  |
| --- | --- |
| Entités | Noms |
| Comité de projet | CREPIN Aurélia  LEVRAT Gaël  RIBIERRE Paul  SAGON Nicolas  VEYS Arthur |
| Examinateur CPE | TROUILLOT Xavier |

#### État du document : Applicable

Table des matières

[1. Recueil des besoins 3](#_Toc403479470)

[1.1. Définition du besoin 3](#_Toc403479471)

[1.2. Définition des acteurs 3](#_Toc403479472)

[1.3. Définition des cas d’utilisation 3](#_Toc403479473)

[1.3.1. Acteur Joueur 4](#_Toc403479474)

[1.3.2. Acteur Maitre du jeu 7](#_Toc403479475)

[2. Analyse 9](#_Toc403479476)

[2.1. Diagramme de classe simplifié 9](#_Toc403479477)

[2.2. Diagrammes de séquence détaillés 10](#_Toc403479478)

[2.2.1. Attaquer 10](#_Toc403479479)

[2.2.2. Charger Partie 11](#_Toc403479480)

[2.2.3. Création de partie 11](#_Toc403479481)

[2.2.4. Débarquer 12](#_Toc403479482)

[2.2.5. Débuter Partie 12](#_Toc403479483)

[2.2.6. Passer le tour 13](#_Toc403479484)

[2.2.7. Quitter Partie 13](#_Toc403479485)

[2.2.8. Réparer 14](#_Toc403479486)

[2.2.9. Riposter 15](#_Toc403479487)

[2.2.10. Sauvegarder 16](#_Toc403479488)

[2.2.11. Se Déplacer 17](#_Toc403479489)

[3. Conception 18](#_Toc403479490)

[3.1. Diagrammes de séquence IHM 18](#_Toc403479491)

[3.1.1. Attaquer 18](#_Toc403479492)

[3.1.2. Charger Partie 19](#_Toc403479493)

[3.1.3. Créer Partie 19](#_Toc403479494)

[3.1.4. Débuter partie 19](#_Toc403479495)

[3.1.5. Quitter Partie 20](#_Toc403479496)

[3.1.6. Sauvegarder Partie 20](#_Toc403479497)

[3.1.7. Se Déplacer 20](#_Toc403479498)

[3.2. Diagramme de classe détaillé 21](#_Toc403479499)

# Recueil des besoins

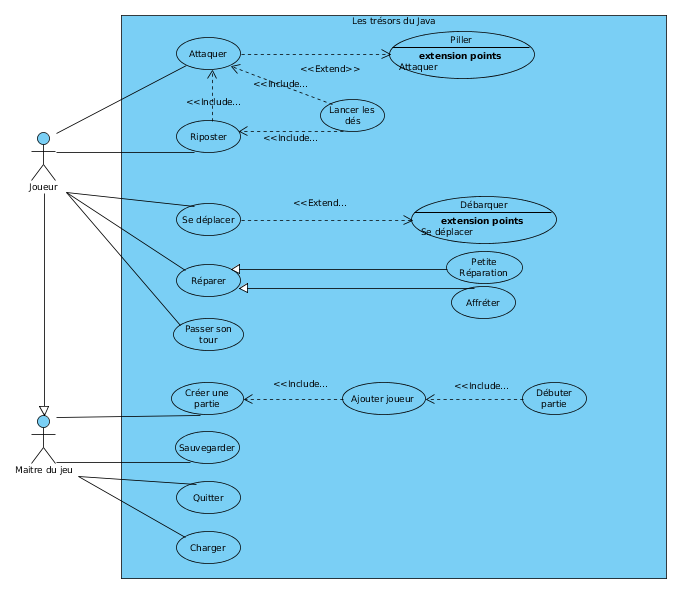
## Définition du besoin

Pour ce projet, le système s’appelle “Les trésors de java”. Ce document doit définir le comportement de ce système, qui a pour but de faire appliquer l’ensemble des règles du jeu et de guider les joueurs dans le déroulement de la partie.

## Définition des acteurs

Ce système ce compose de deux acteurs, un acteur principal qui sera le joueur : il pourra faire toutes les actions pour jouer une partie. Le système comportera aussi un acteur secondaire qui est le maitre du jeu. Cet acteur pourra lancer une partie et la gérer. Le maitre du jeu peut devenir un joueur une fois qu’il a rempli son rôle.

## Définition des cas d’utilisation

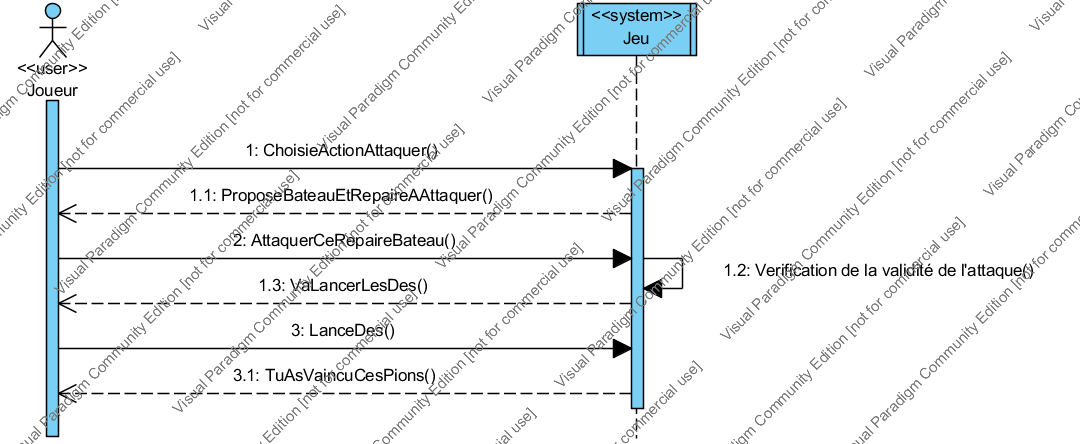


1. Attaquer

### Acteur Joueur

#### Attaquer

Cas d’utilisation: Le joueur veut attaquer un bateau ou un repaire. Le système lui demande de préciser quel endroit attaquer, puis lui demande de lancer les dés. Le système calcule et renvoi le résultat de l’attaque.



#### Riposter

Cas d’utilisation: Le joueur a la possibilité de riposter. Le système lui demande de lancer les dés et calcule le résultat du combat.



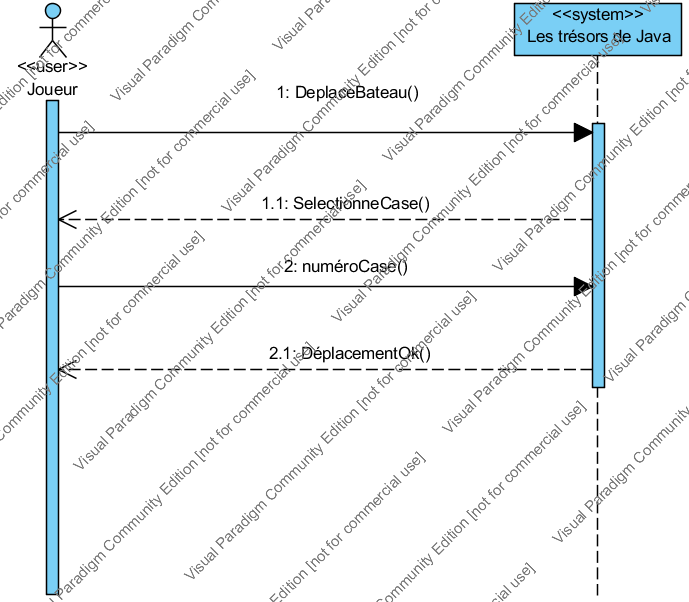
#### Lancer les dés

Cas d’utilisation: Le joueur lance les dés.



#### Se déplacer

Cas d’utilisation: Lors d’un déplacement le joueur à la possibilité de sélectionner une case “mer” à proximité de son bateau. Une fois que le joueur a choisi la case, le bateau se déplace vers cette nouvelle case.



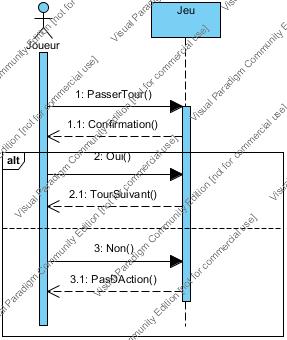
#### Débarquer

Cas d’utilisation: Lors d’un retour au port, le joueur débarque ses trésors pour augmenter son score.



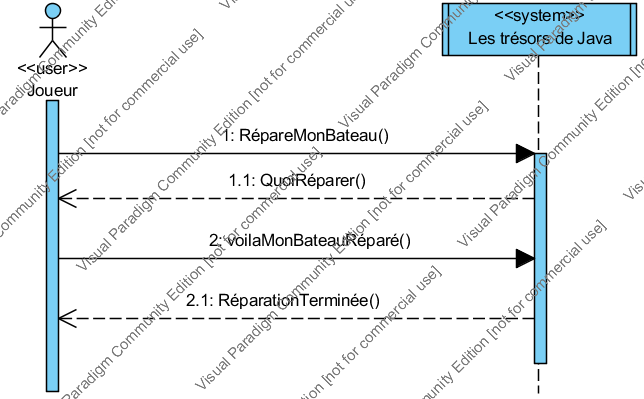
#### Passer tour

Cas d’utilisation : le joueur souhaite passer son tour.



#### Réparer

Cas d’utilisation: Le joueur veut réparer son navire. Le système lui demande ce qu’il veut ajouter ou modifier au navire  Le joueur répond et le système effectue les modifications.



#### Piller

Cas d’utilisation: Quand le joueur décide de piller, il récupère les trésors du bateau ennemi. Le bateau de l’acteur est par la suite réorganisé pour pouvoir transporter les trésors.



### Acteur Maitre du jeu

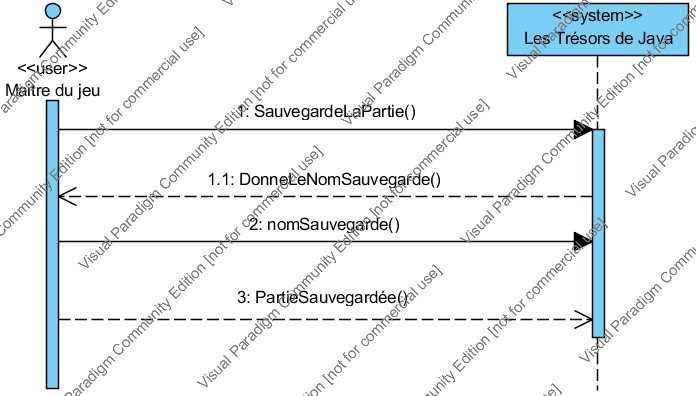
#### Créer partie

Cas d’utilisation: un maitre du jeu ajoute le nombre de joueurs, leurs noms et couleurs avant le début de la partie.



#### Sauvegarder

Cas d’utilisation: A n’importe quel moment, un maitre du jeu peut sauvegarder la partie.



#### Quitter le jeu

Cas d’utilisation: un maitre du jeu quitte le jeu. le système lui propose de sauvegarder sa partie en cours.



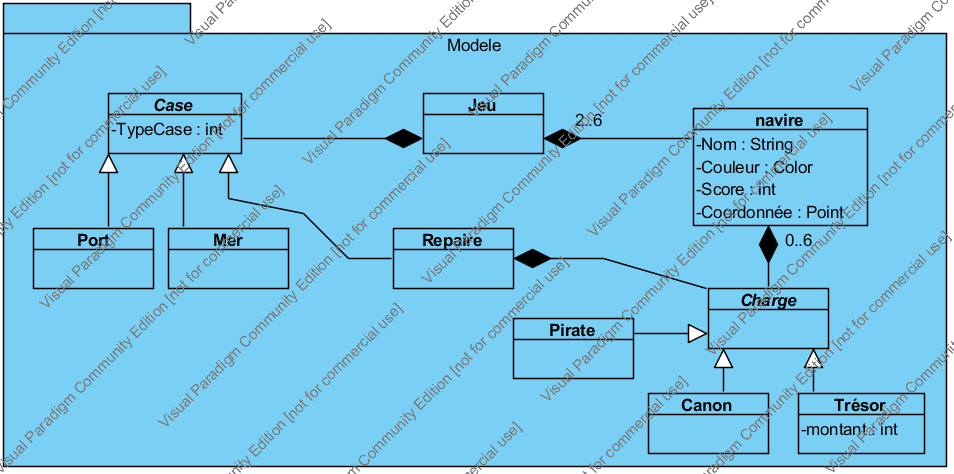
#### Débuter partie

Cas d’utilisation: un maitre du jeu lance la partie. Le système initialise le jeu.



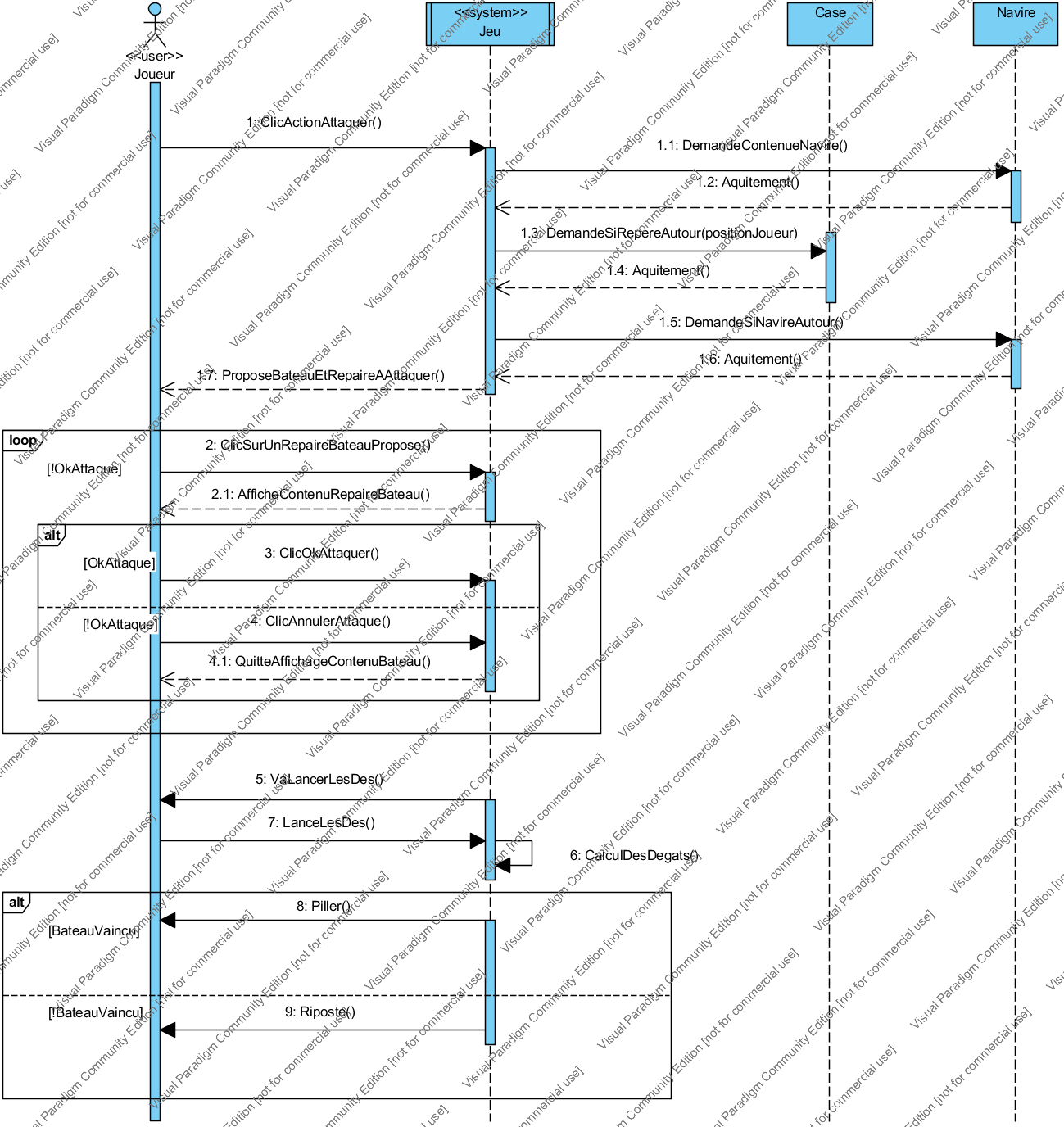
# Analyse

## Diagramme de classe simplifié : Modèle

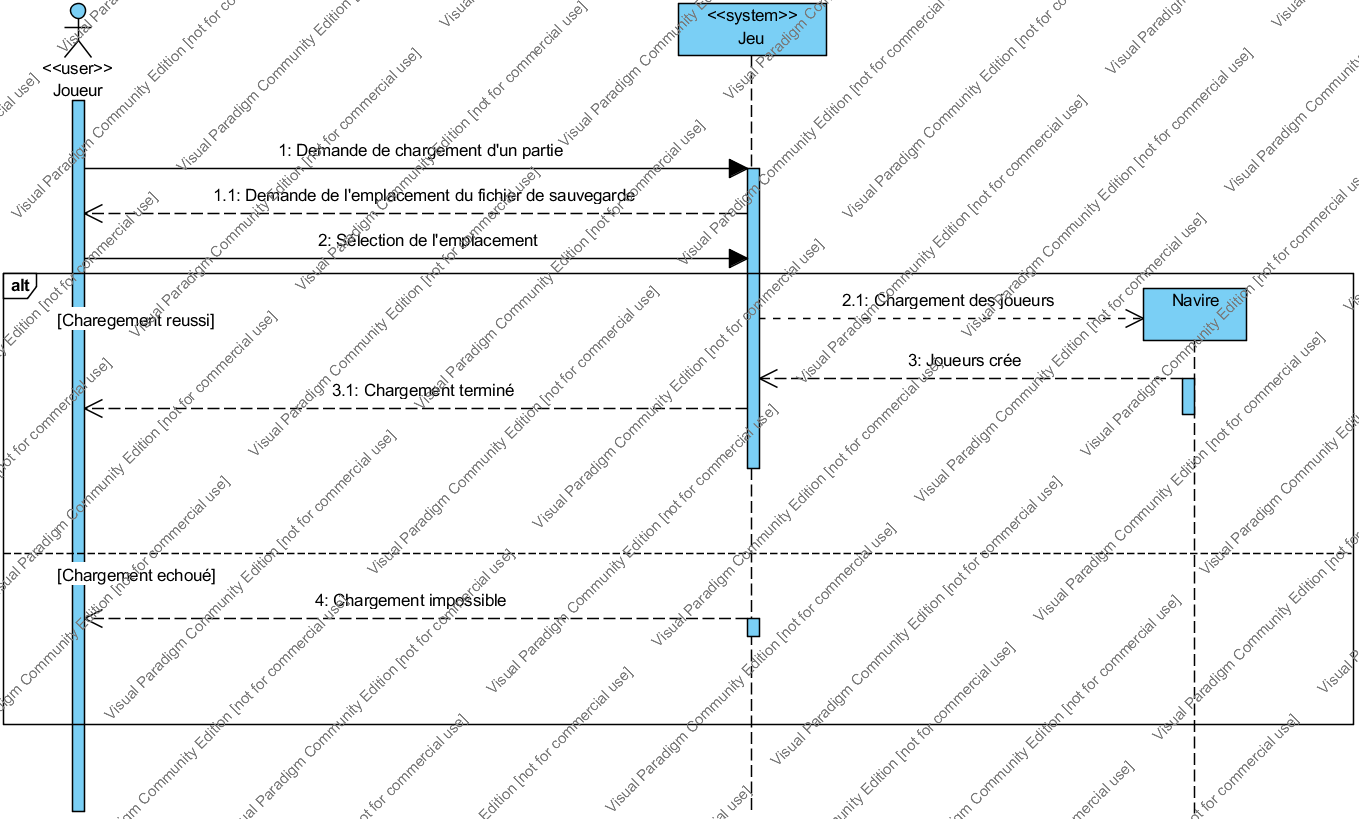


## Diagrammes de séquence détaillés

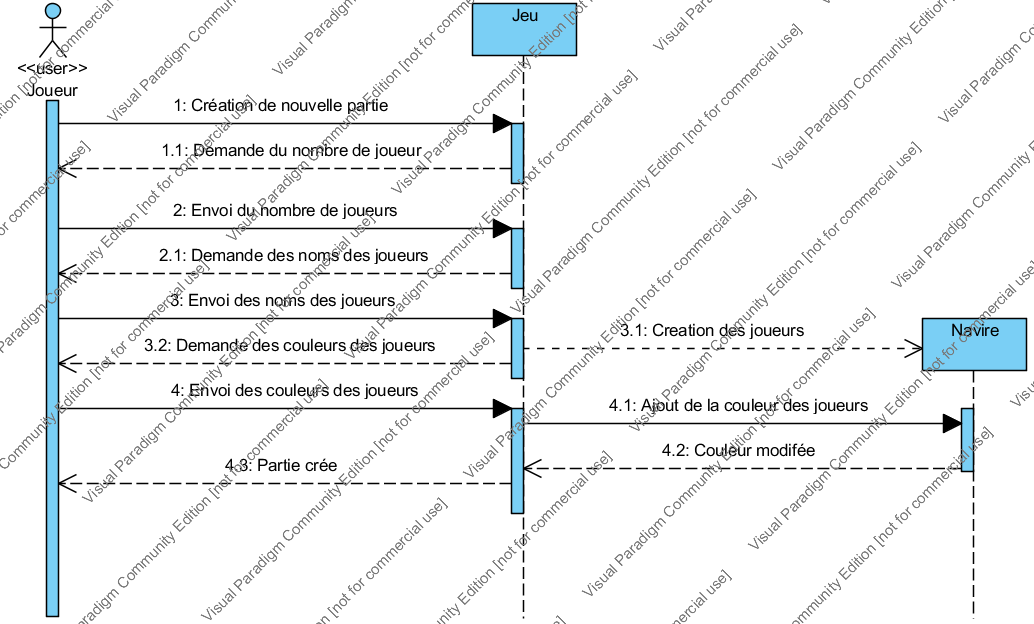
### Attaquer



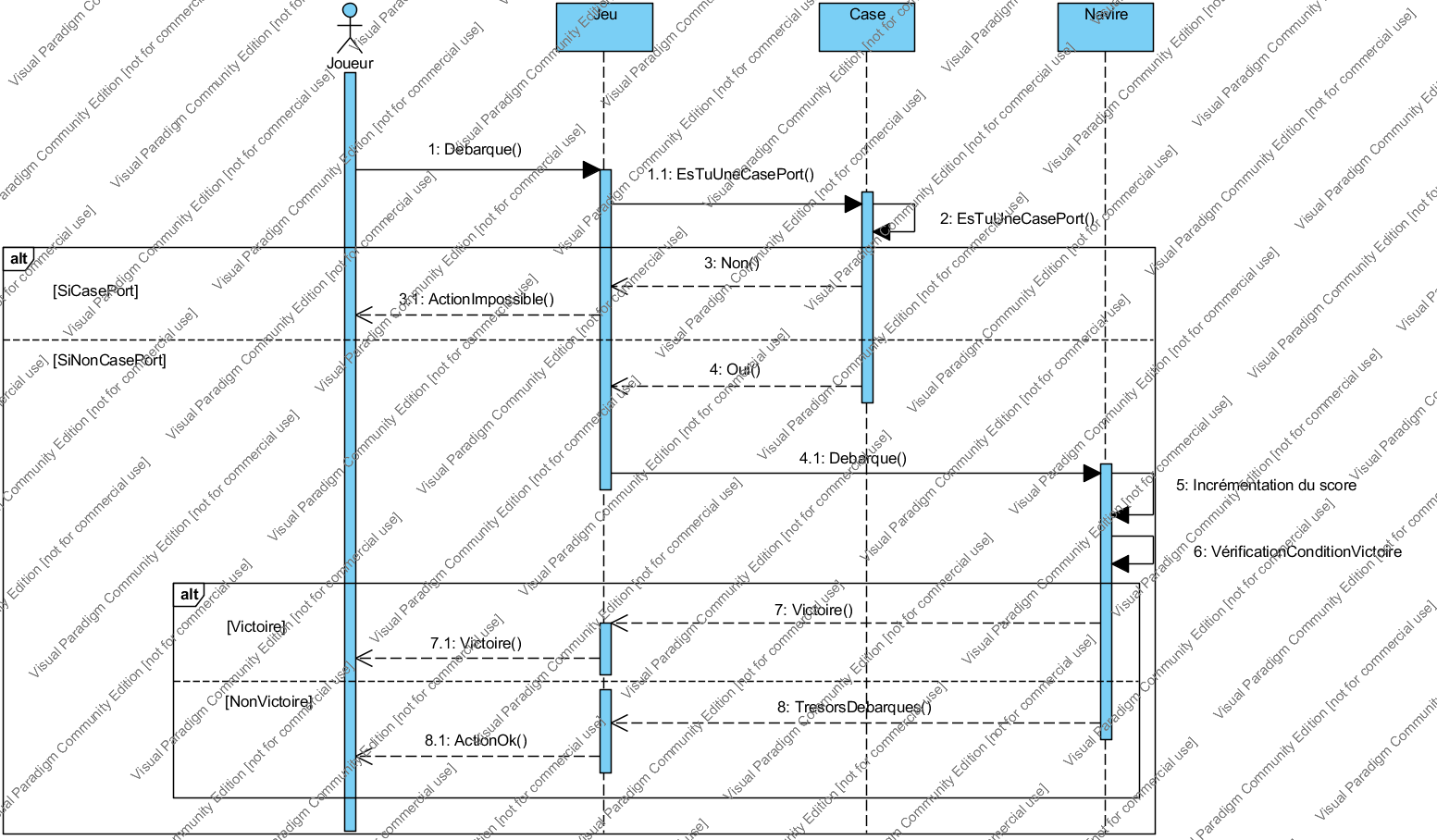
### Charger Partie



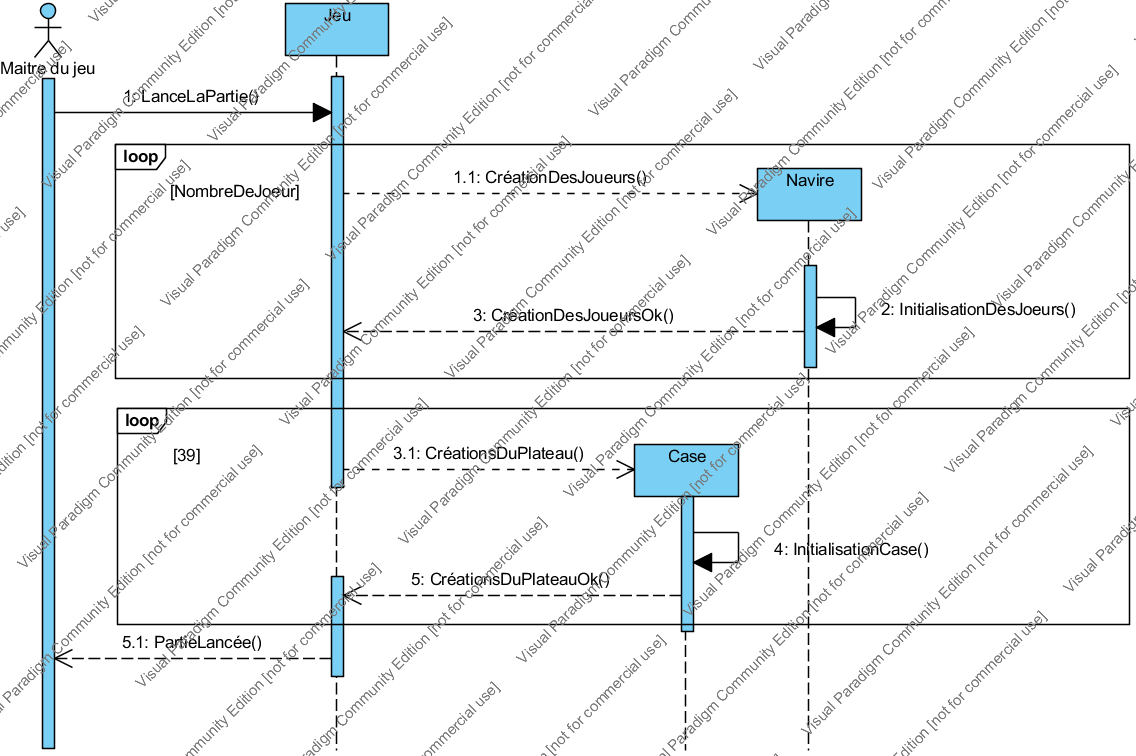
### Création de partie



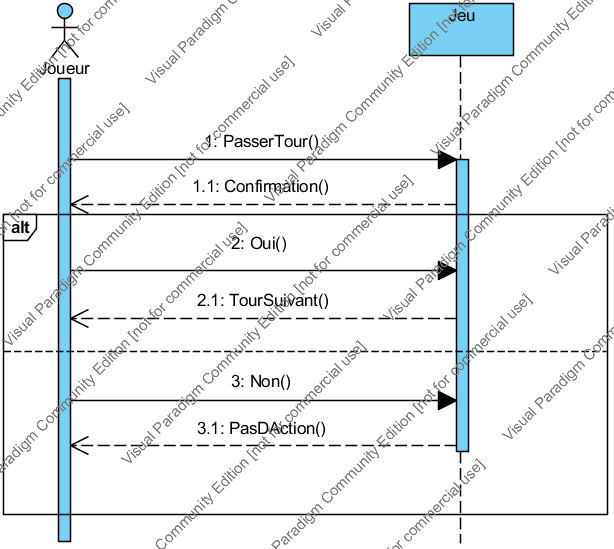
### Débarquer



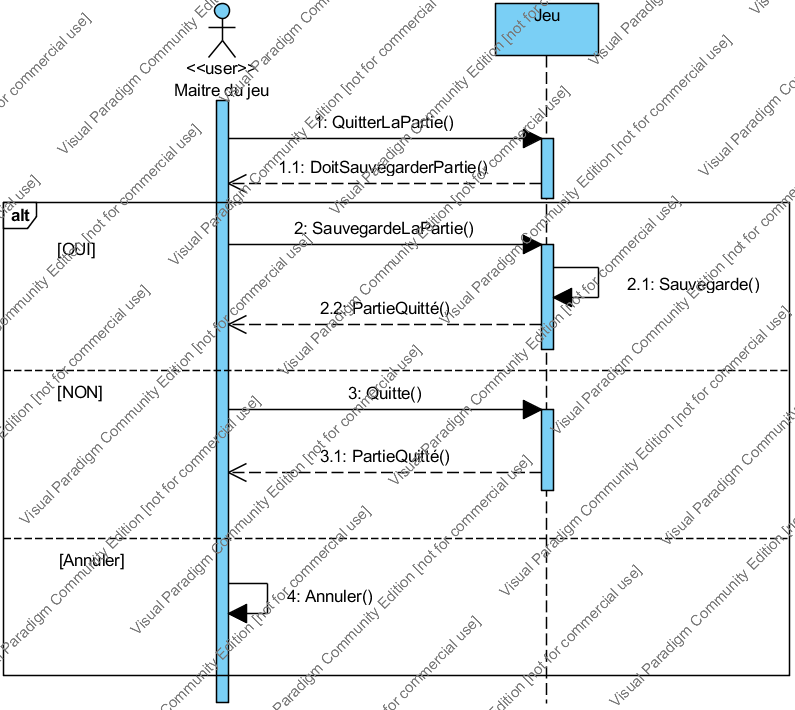
### Débuter Partie



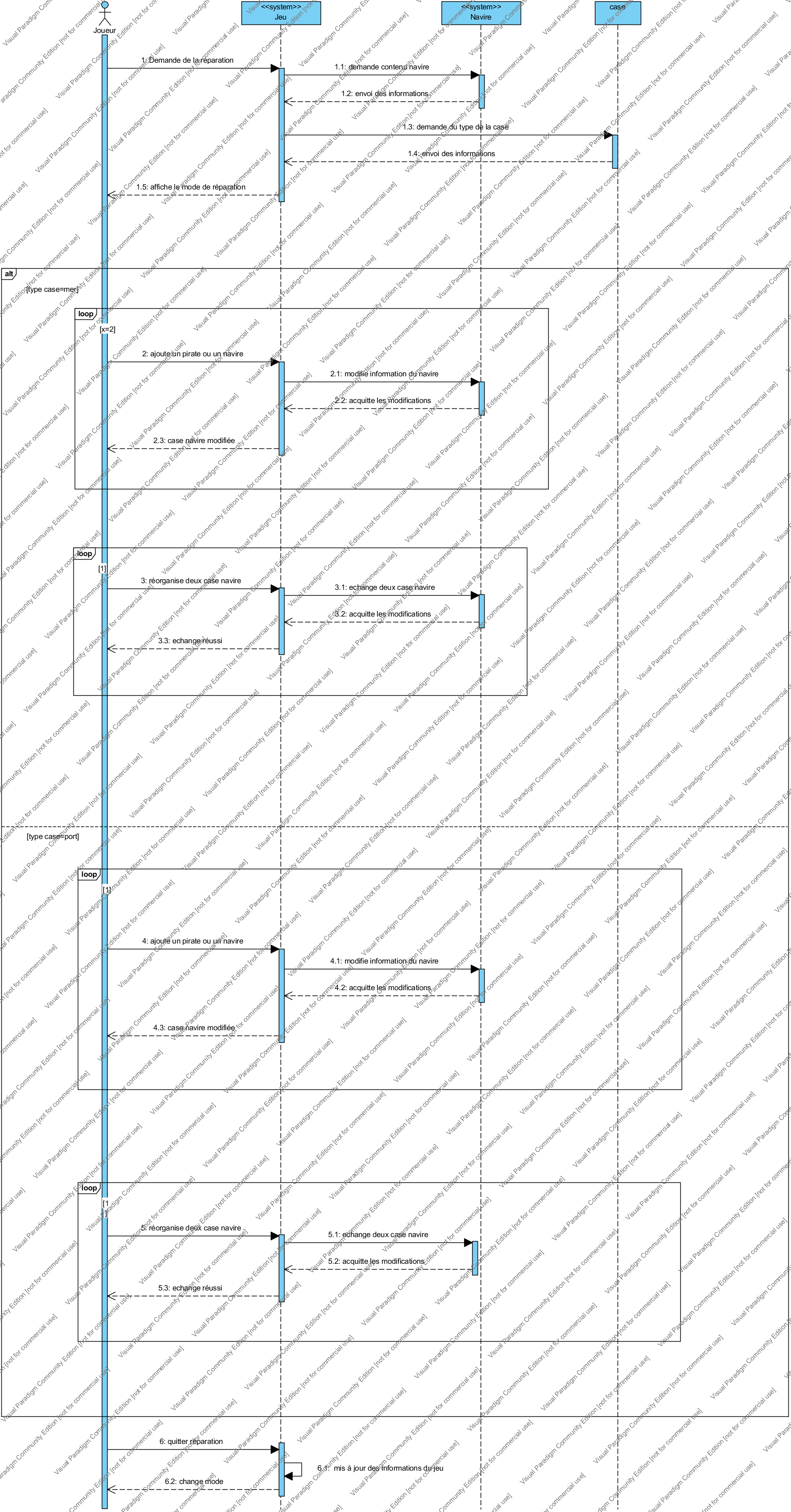
### Passer le tour

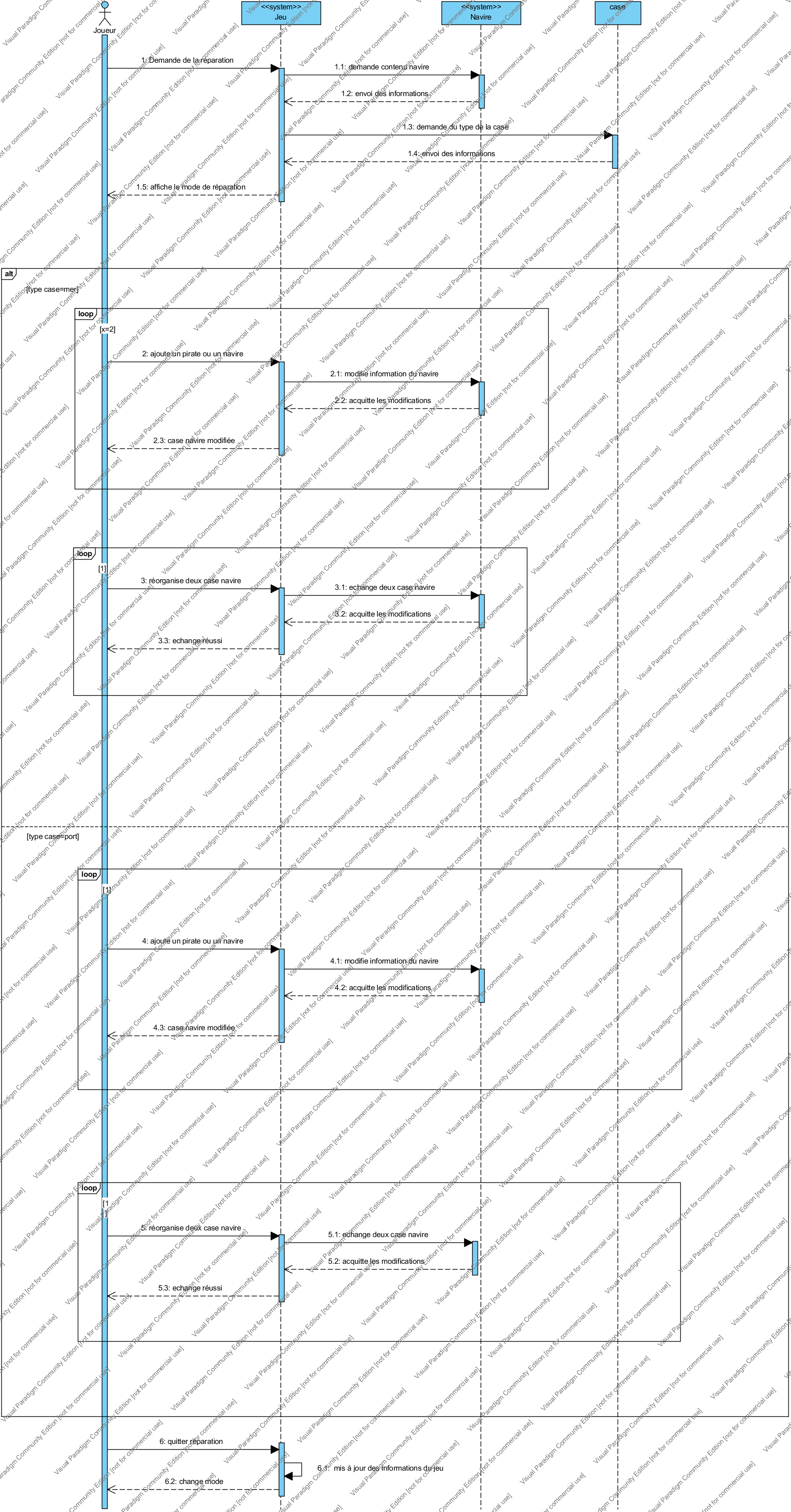


### Quitter Partie



### Réparer

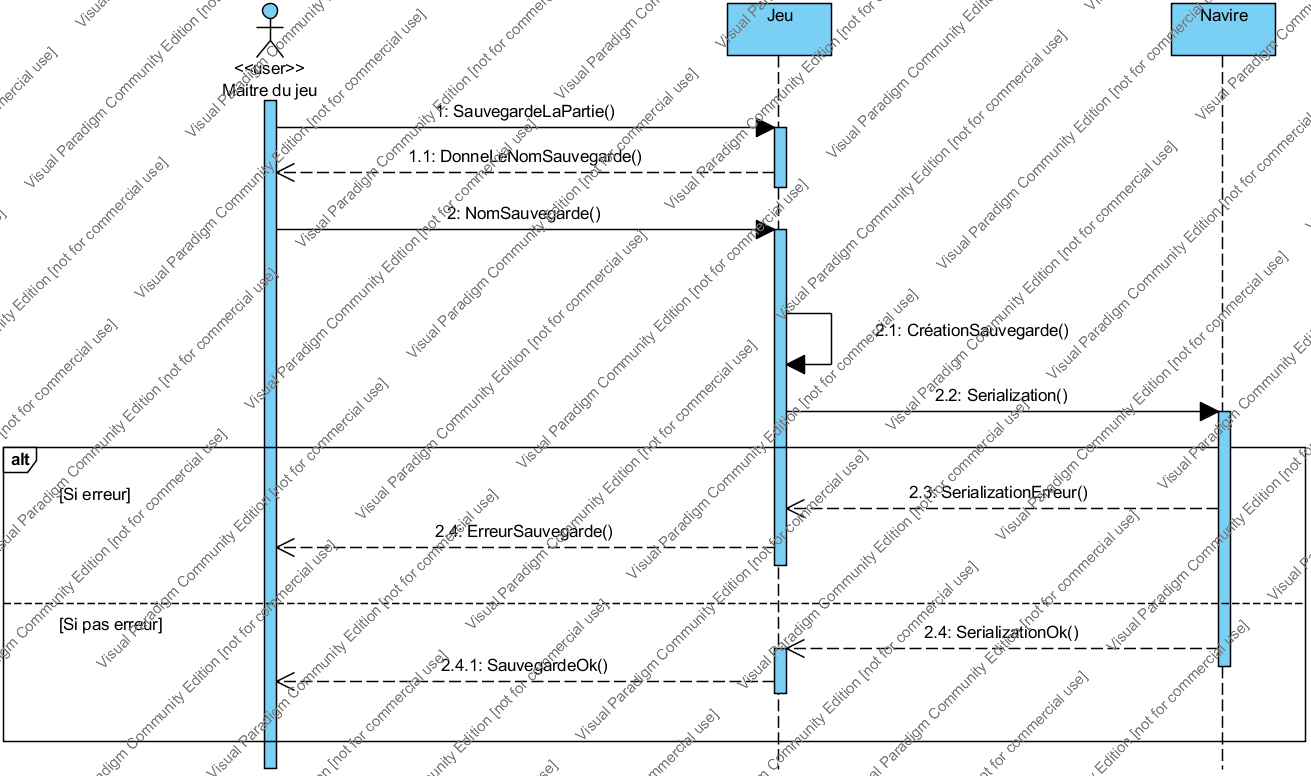




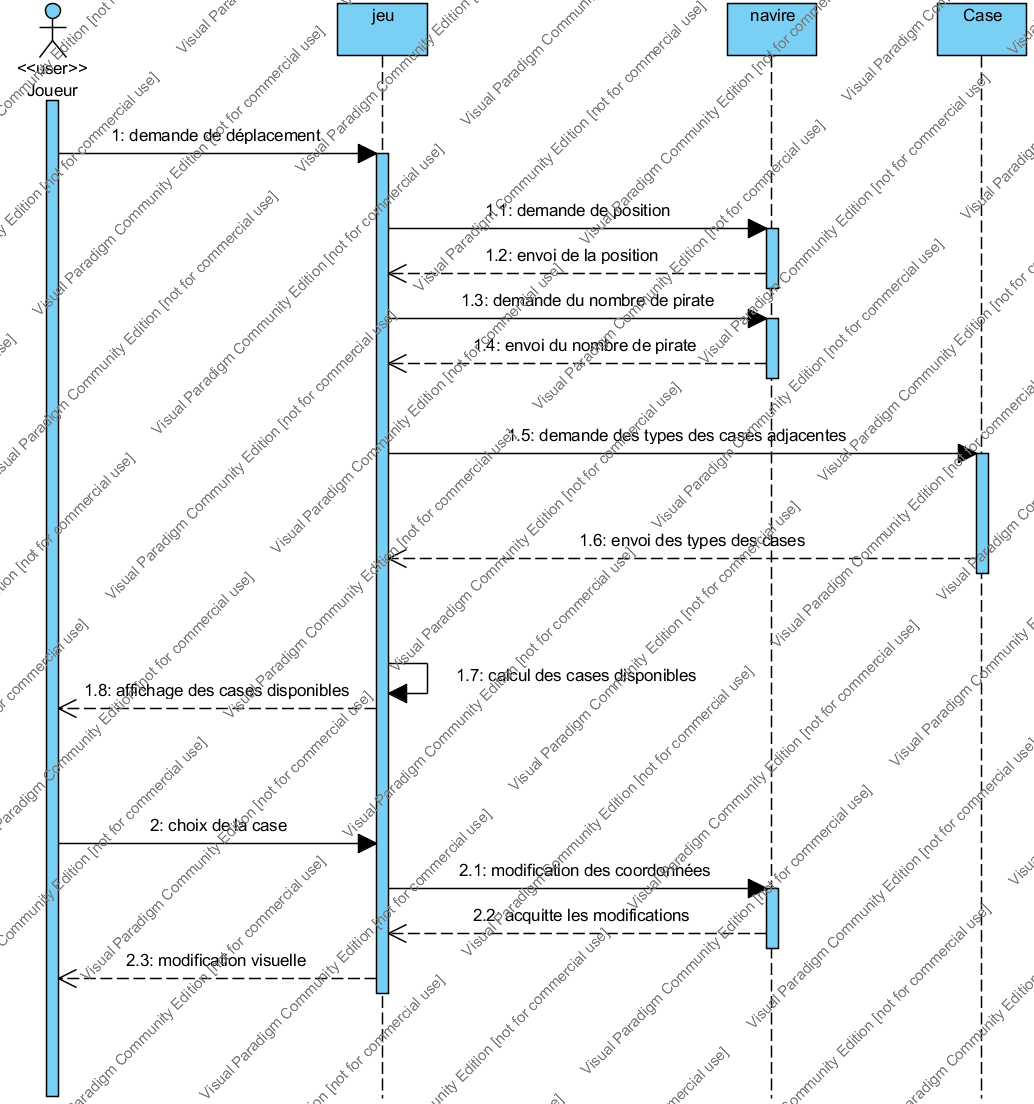
### Riposter



### Sauvegarder



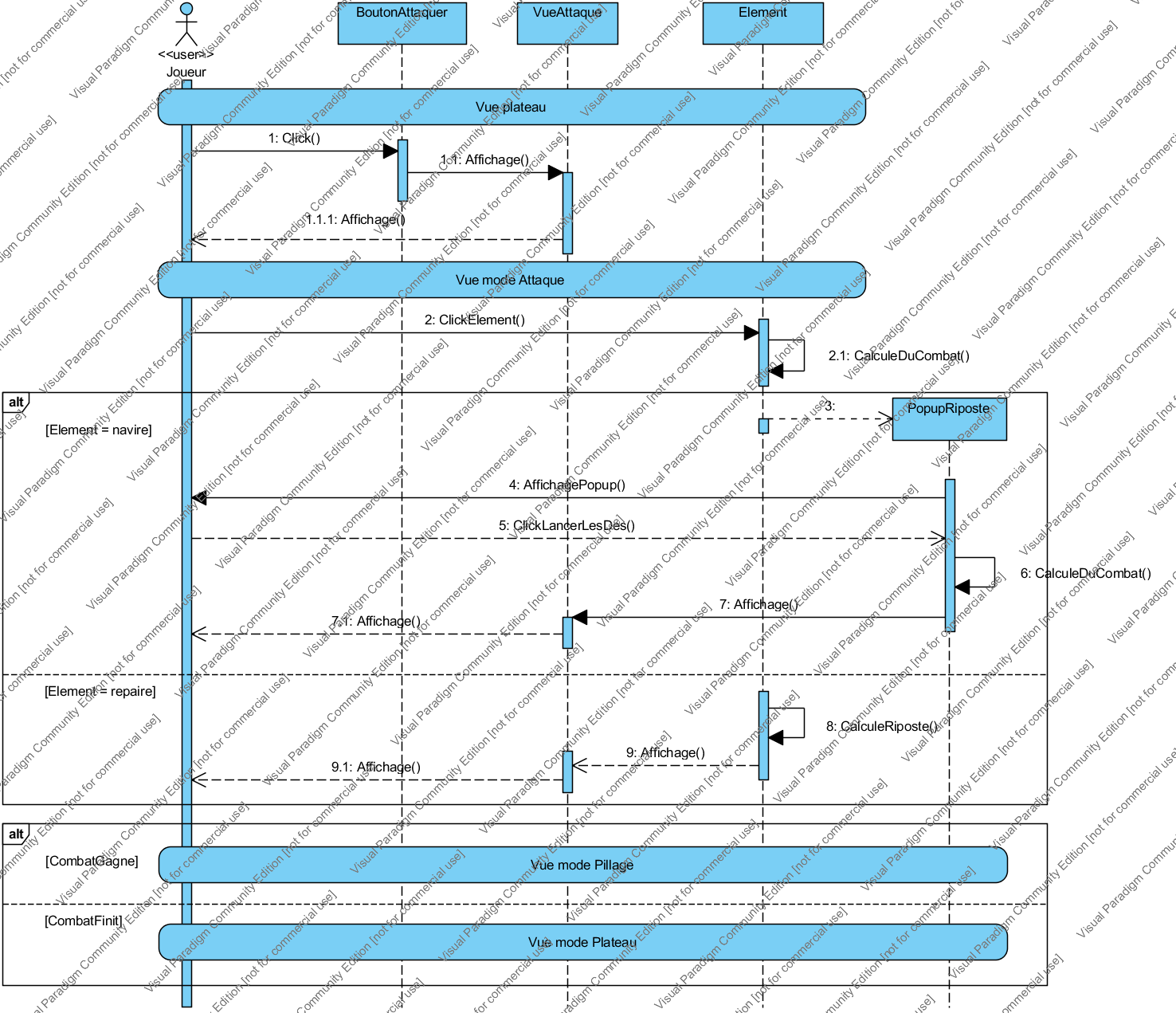
### Se Déplacer



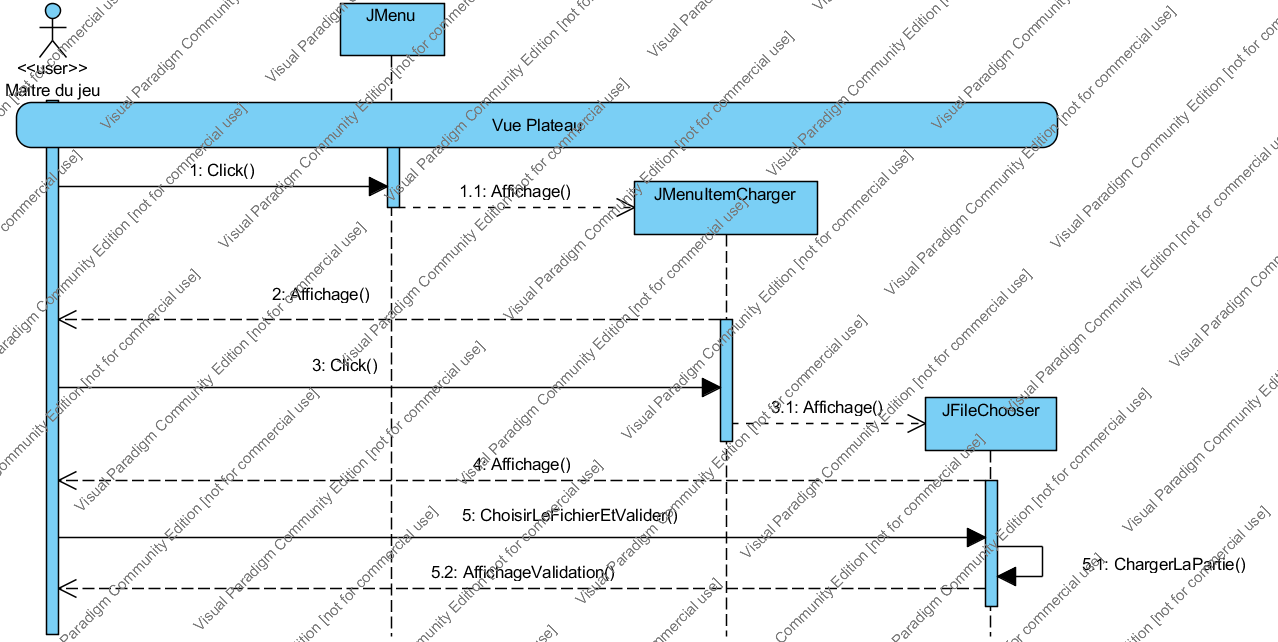
# Conception

## Diagrammes de séquence IHM

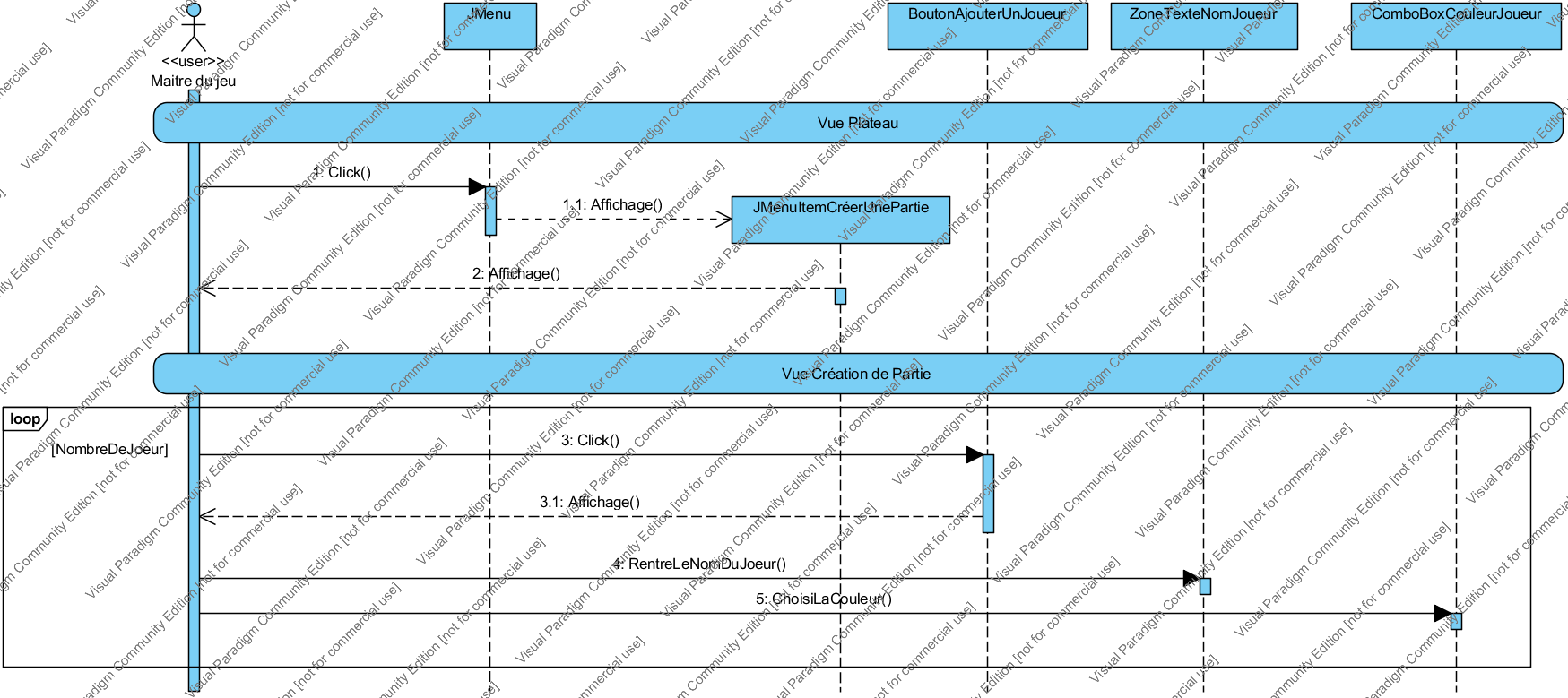
### Attaquer



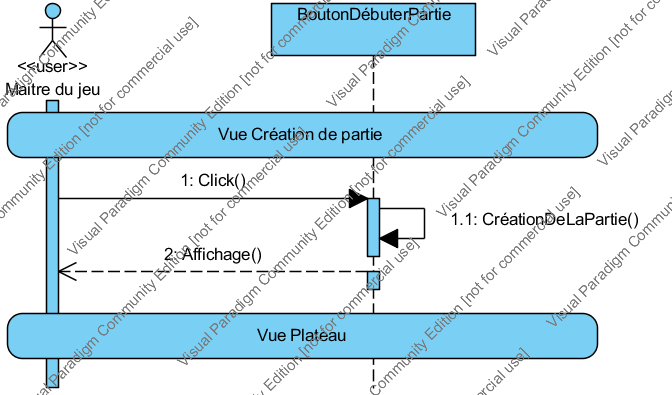
### Charger Partie



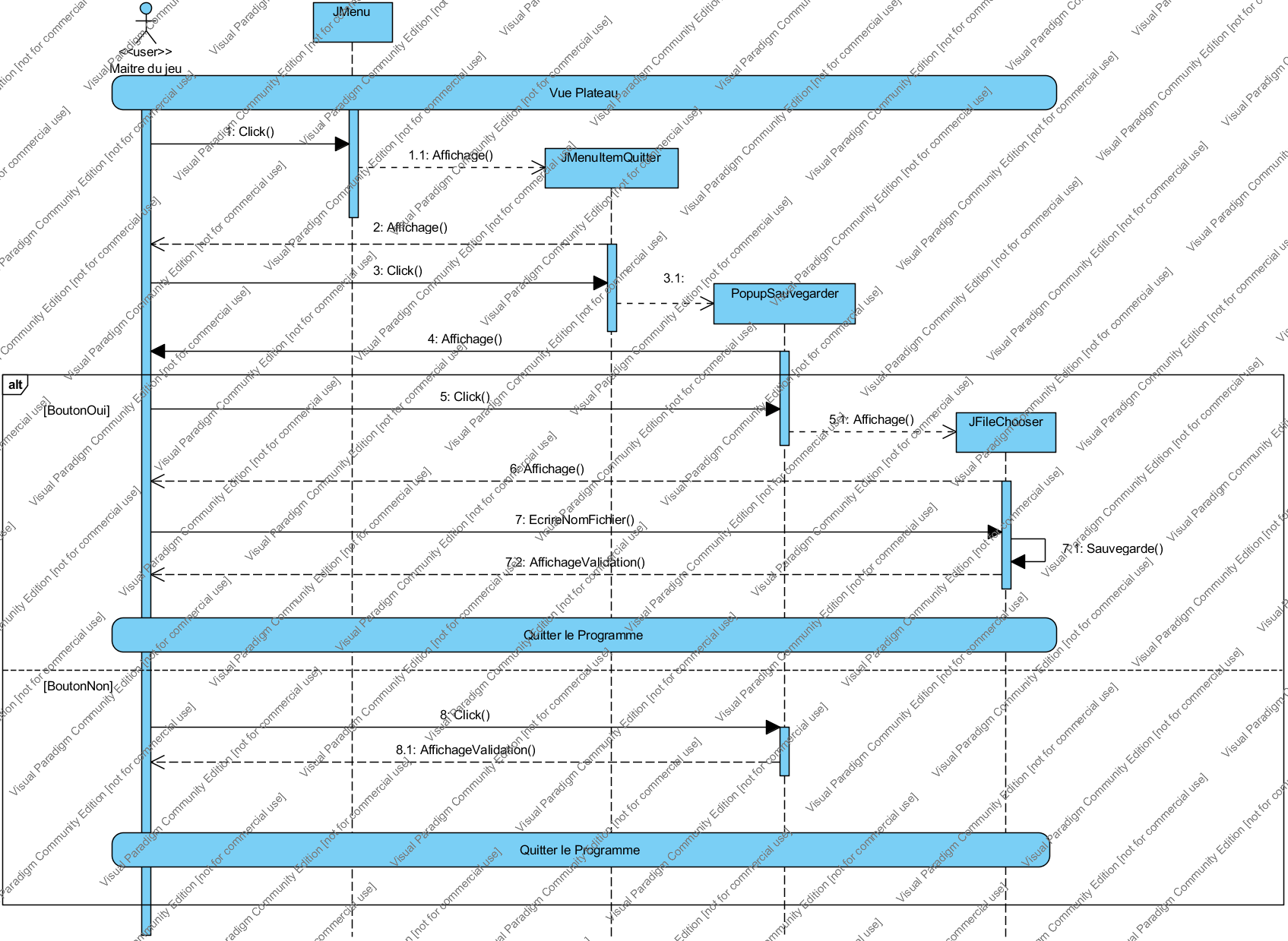
### Créer Partie



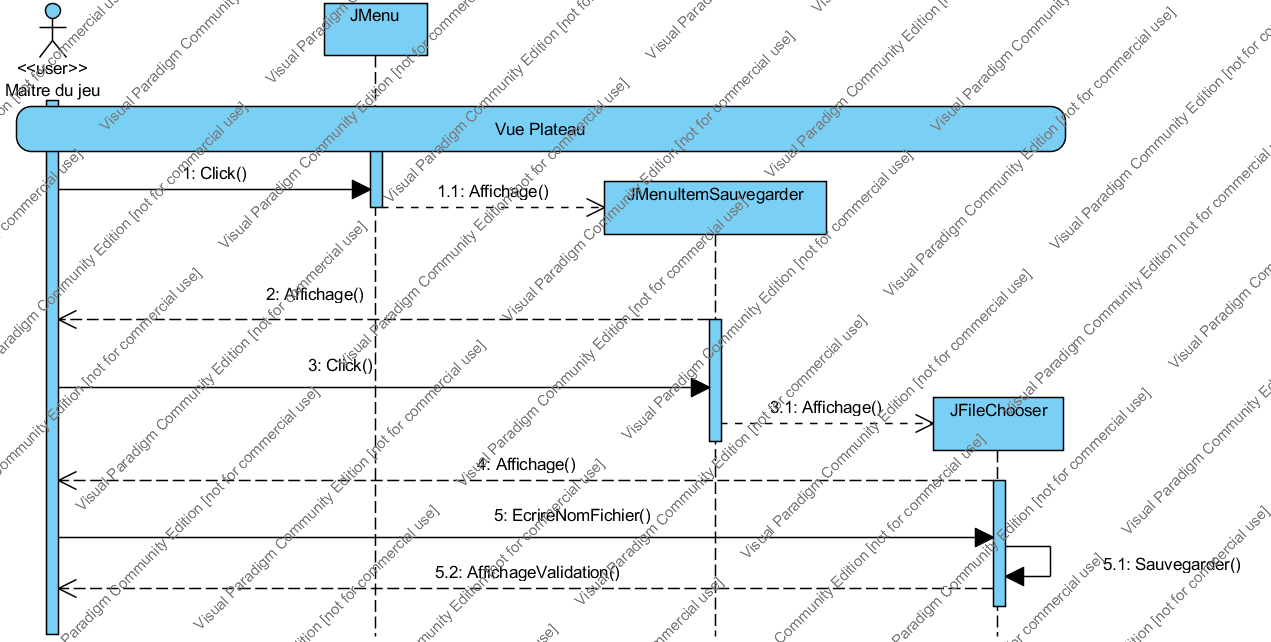
### Débuter partie



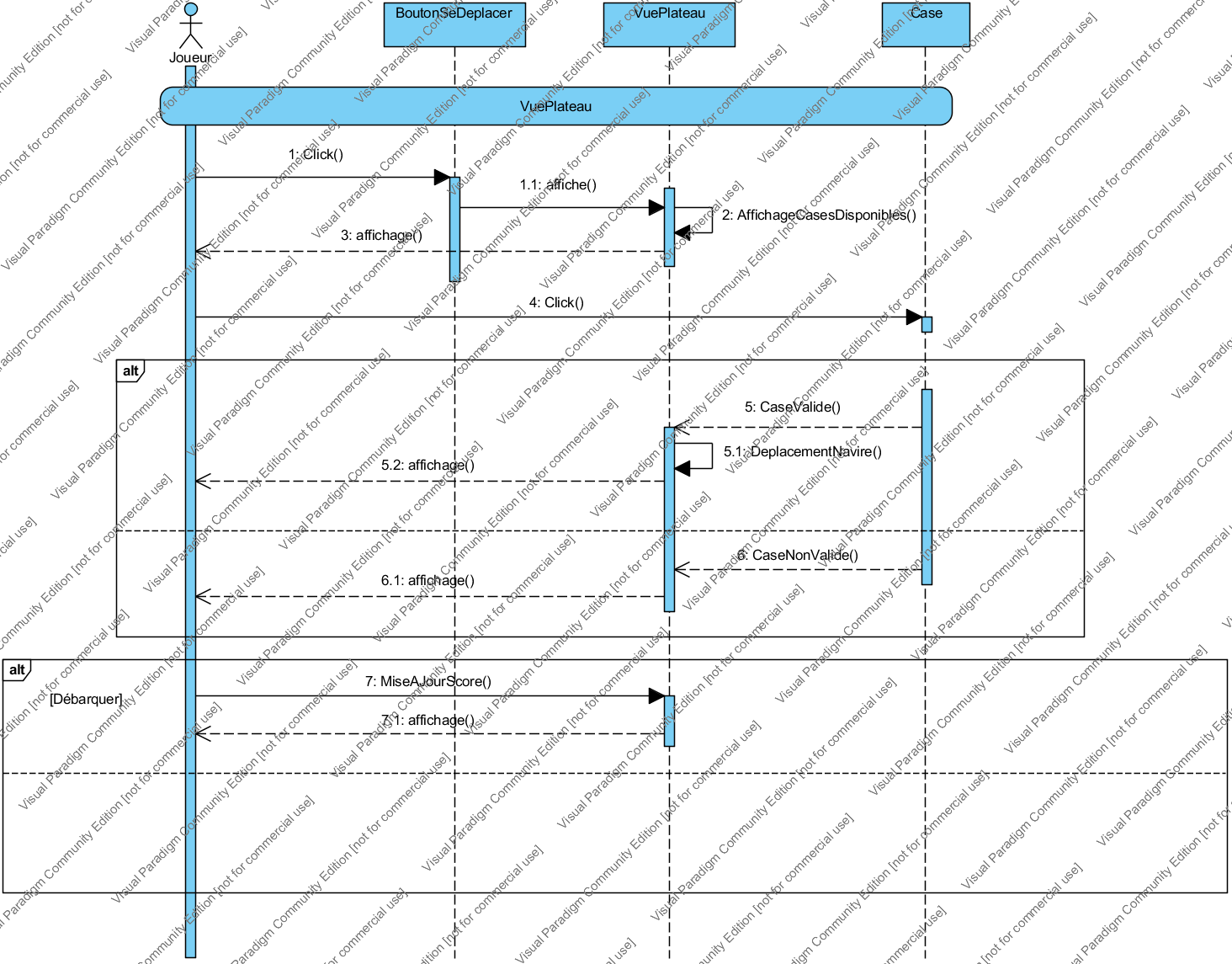
### Quitter Partie



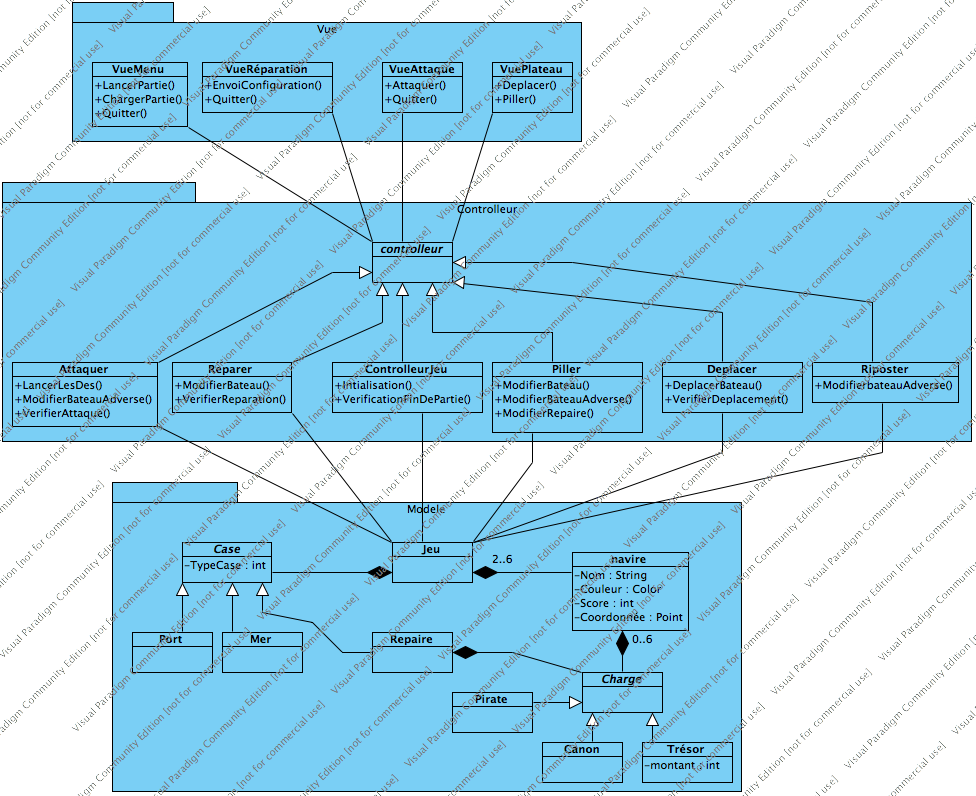
### Sauvegarder Partie



### Se Déplacer



## Diagramme de classe détaillé



# Implémentation

Faute de temps nous n’avons pas pu terminer tout ce que nous aurions aimé faire pour la conception de ce logiciel (notamment les diagrammes d’etats-transitions).