

# VISUALIGUE

## Livrable #2

### Équipe « Ganbatte »

Samuel Beaudoin (SABEA93)

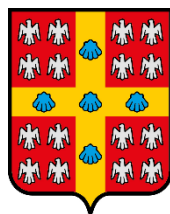
Michaël Dodier (MIDOD1)

Samy Harani (SAHAR48)

(IFT-2007) Analyse et conception des systèmes orientés  
objets

Travail présenté à Martin Savoie

Département d'informatique et de génie logiciel  
Université Laval  
Automne 2016



## Table des matières

Diagrammes de classe de conception.....	1
Diagrammes de package (architecture logique) .....	3
Diagrammes de séquence de conception .....	5
#1 - Conversion des coordonnées d'un point .....	5
#2 - Ajout d'un joueur à l'aide de la souris .....	5
#3 - Détecter l'endroit d'un clic .....	6
#4 - Interactions de l'édition en mode image par image .....	7
#5 - Interactions de l'édition d'un jeu en mode temps réel .....	8
#6 - Visionnage des jeux.....	9
Diagramme de Gantt (actualisé) .....	10
Annexes.....	11
#1 Vision.....	11
Introduction .....	11
Énoncé du problème.....	11
Description des parties prenantes .....	11
Problèmes et objectifs clés des parties prenantes .....	12
Objectifs niveau-utilisateur .....	12
Sommaires des bénéfices.....	12
Sommaire des caractéristiques du système.....	13
Autres exigences .....	13
Exigences de qualité.....	13
Exigences de performance.....	14
#2 Cas d'utilisations .....	14
Créer un type de sport.....	14
Détails .....	14
Diagramme de séquences.....	15
Ajouter un joueur.....	15
Détails .....	15
Diagramme de séquences.....	16
Déplacer un joueur .....	17
Détails .....	17

Diagramme de séquences.....	17
Créer le déplacement d'un joueur .....	18
Détails .....	18
Diagramme de séquences.....	19
Définir l'action d'un joueur .....	20
Détails .....	20
Diagramme de séquences.....	20
Supprimer un joueur .....	21
Détails .....	21
Diagramme de séquences.....	21
Changer l'orientation d'un joueur .....	22
Détails .....	22
Diagramme de séquences.....	22
Changer de mode de création.....	22
Détails .....	22
Diagramme de séquences.....	23
Créer ou modifier un obstacle .....	24
Détails .....	24
Diagramme de séquences.....	25
Ajouter un obstacle au jeu .....	26
Détails .....	26
Diagramme de séquences.....	26
Supprimer un obstacle du jeu .....	27
Détails .....	27
Diagramme de séquences.....	27
Définir le rôle d'un joueur.....	28
Détails .....	28
Diagramme de séquences.....	28
Donner ou bouger l'accessoire de sport d'un joueur .....	29
Détails .....	29
Diagramme de séquences.....	30
Créer un nouveau jeu.....	31

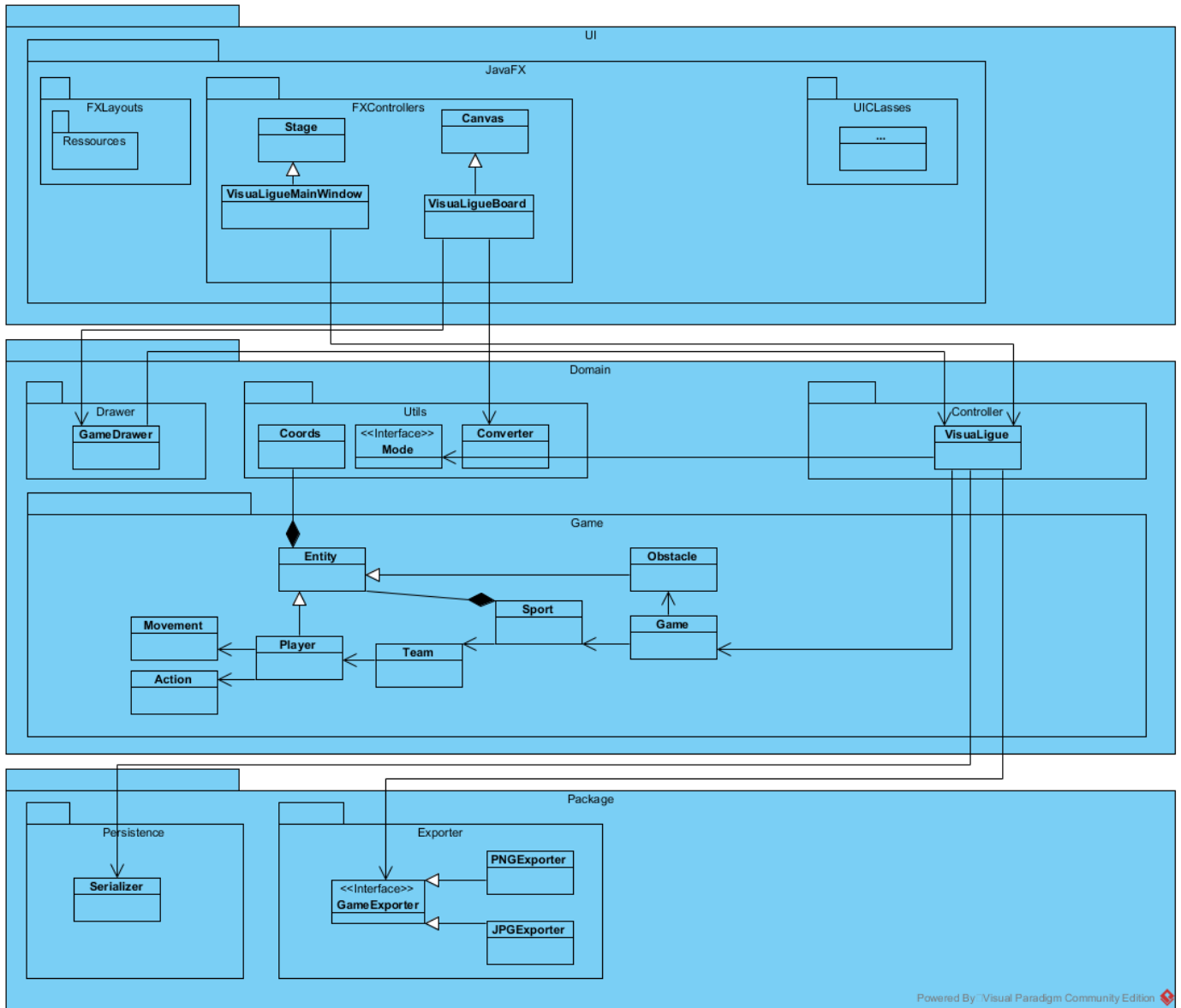
Détails .....	31
Diagramme de séquences .....	32
Visionner une partie.....	33
Détails .....	33
Diagramme de séquences .....	34
Ouvrir une partie existante .....	35
Détails .....	35
Diagramme de séquences .....	36
Exporter une partie .....	37
Détails .....	37
Diagramme de séquences .....	37
Sauvegarder une partie.....	38
Détails .....	38
Diagramme de séquences .....	38
Afficher ou cacher les rôles d'un ou des joueurs .....	39
Détails .....	39
Diagramme de séquences .....	39
Démarrer ou mettre en pause une partie.....	40
Détails .....	40
Diagramme de séquences .....	40
Avancer ou reculer l'exécution d'une partie.....	40
Détails .....	40
Diagramme de séquences .....	40
Avancer ou reculer l'exécution d'une partie par bond de temps .....	41
Détails .....	41
Diagramme de séquences .....	41
#3 Modèle du domaine .....	41
#4 Glossaire.....	42

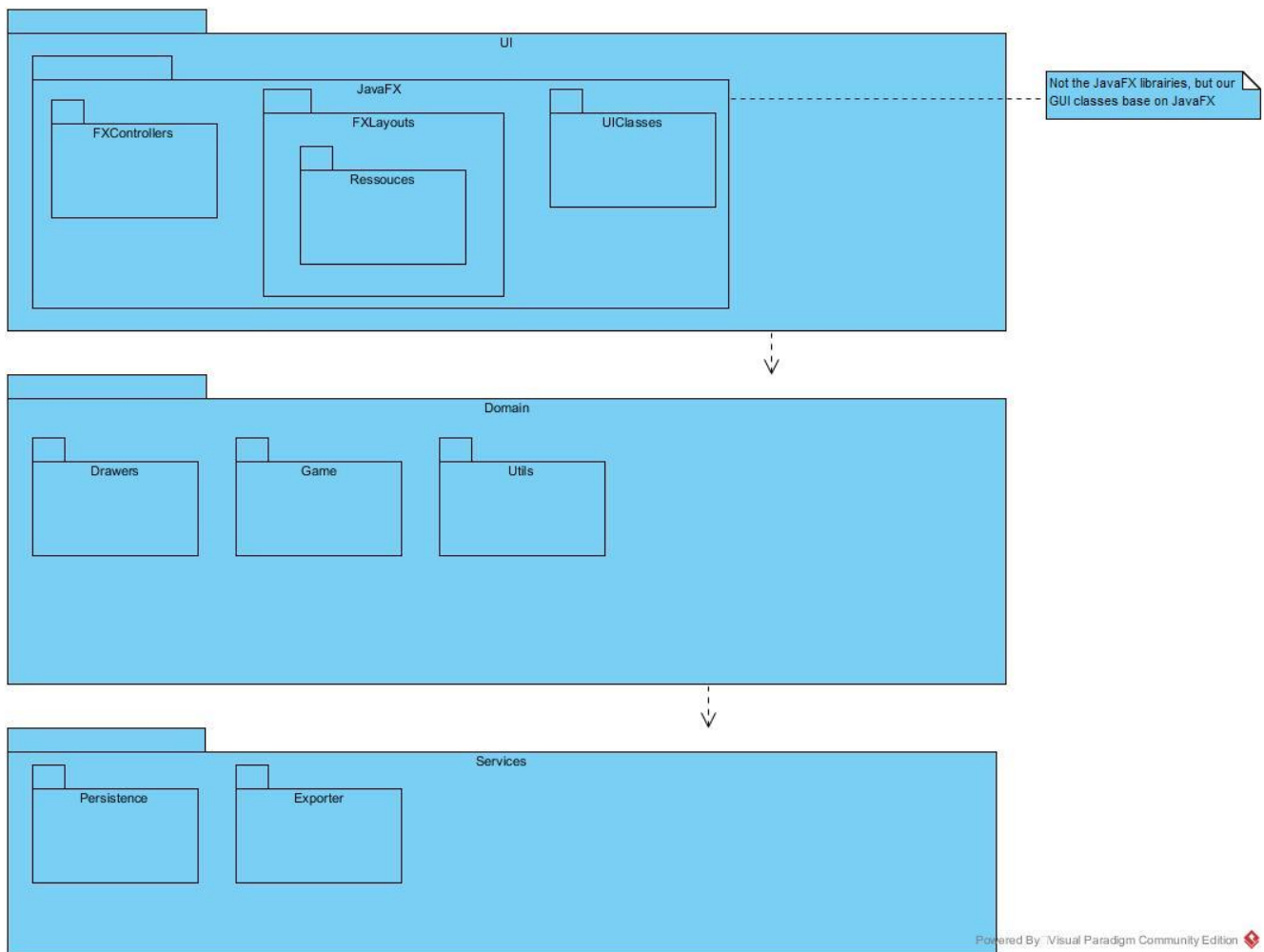
## 1 | Page



VisualLigue sera notre classe d'entrée qui gèrera toutes les fonctionnalités du domaine du réseau de notre application pour garder une forte cohésion. L'interface communiquera avec lui et celui-ci redistribuera les tâches à la bonne classe. Game classe importante, aura le rôle de gérer et connaître les sports et tout ce qui s'en suit. Par exemple, tous les éléments reliés au jeu passent par elle, que ce soit les joueurs, les accessoires ou les obstacles. La classe sport quant à elle permet notamment de gérer les équipes ainsi que la possibilité d'ajouter ou de supprimer des joueurs sur un terrain.

## Diagrammes de package (architecture logique)





La première couche est constituée de l'interface utilisateur. L'affichage de l'application se fera à l'aide de Java FX, donc elle a été représentée dans l'interface utilisateur.

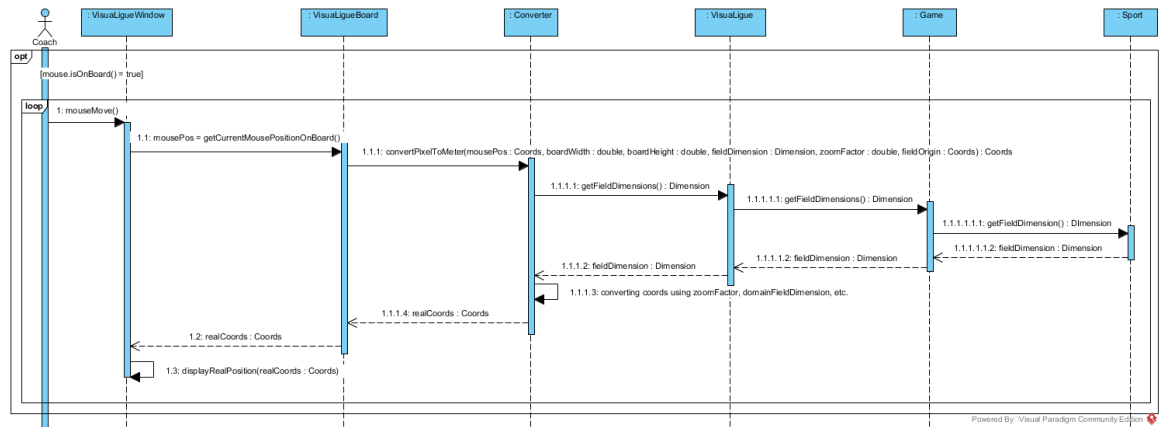
La deuxième couche comporte des éléments reliés au domaine du VisualLigue, c'est-à-dire les stratégies. On peut donc y retrouver les joueurs, les équipes, les jeux, les obstacles etc.

Dans la dernière couche, on retrouve l'exportation des fichiers (qui permettra d'exporter des stratégies au format .png ou .jpg) ainsi que la sérialisation à des fins de sauvegarde.

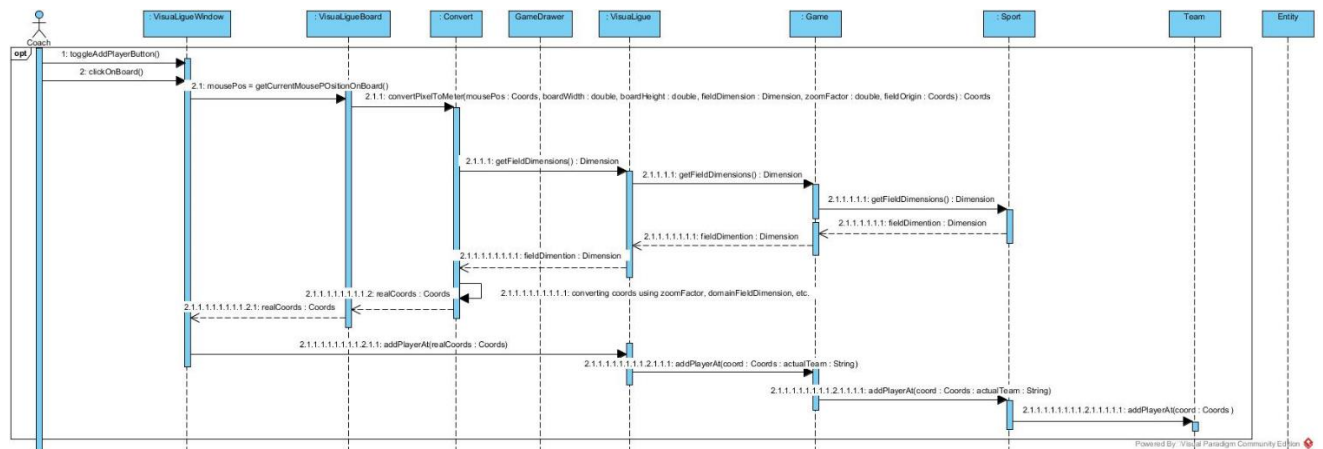


# Diagrammes de séquence de conception

## #1 - Conversion des coordonnées d'un point

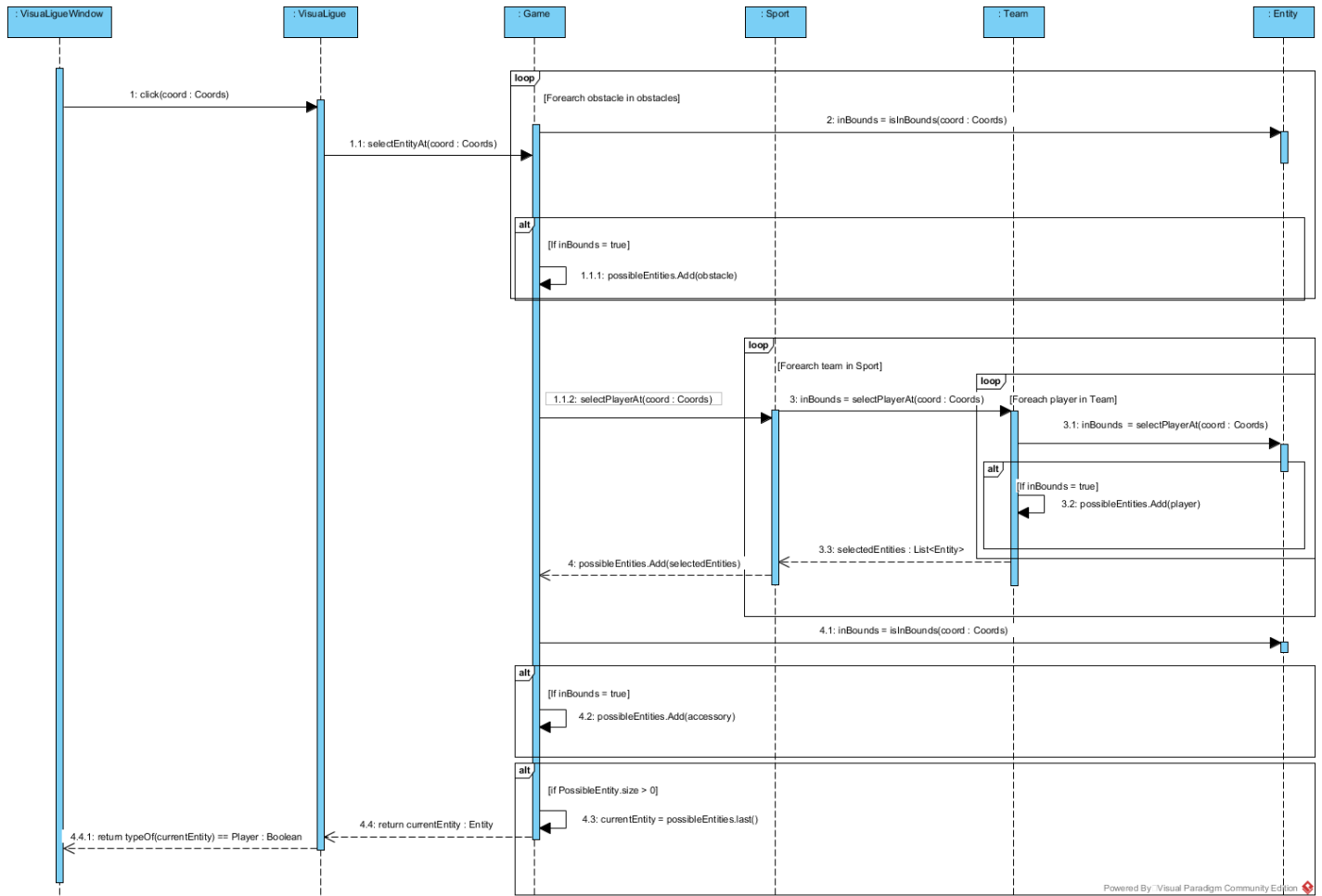


## #2 - Ajout d'un joueur à l'aide de la souris



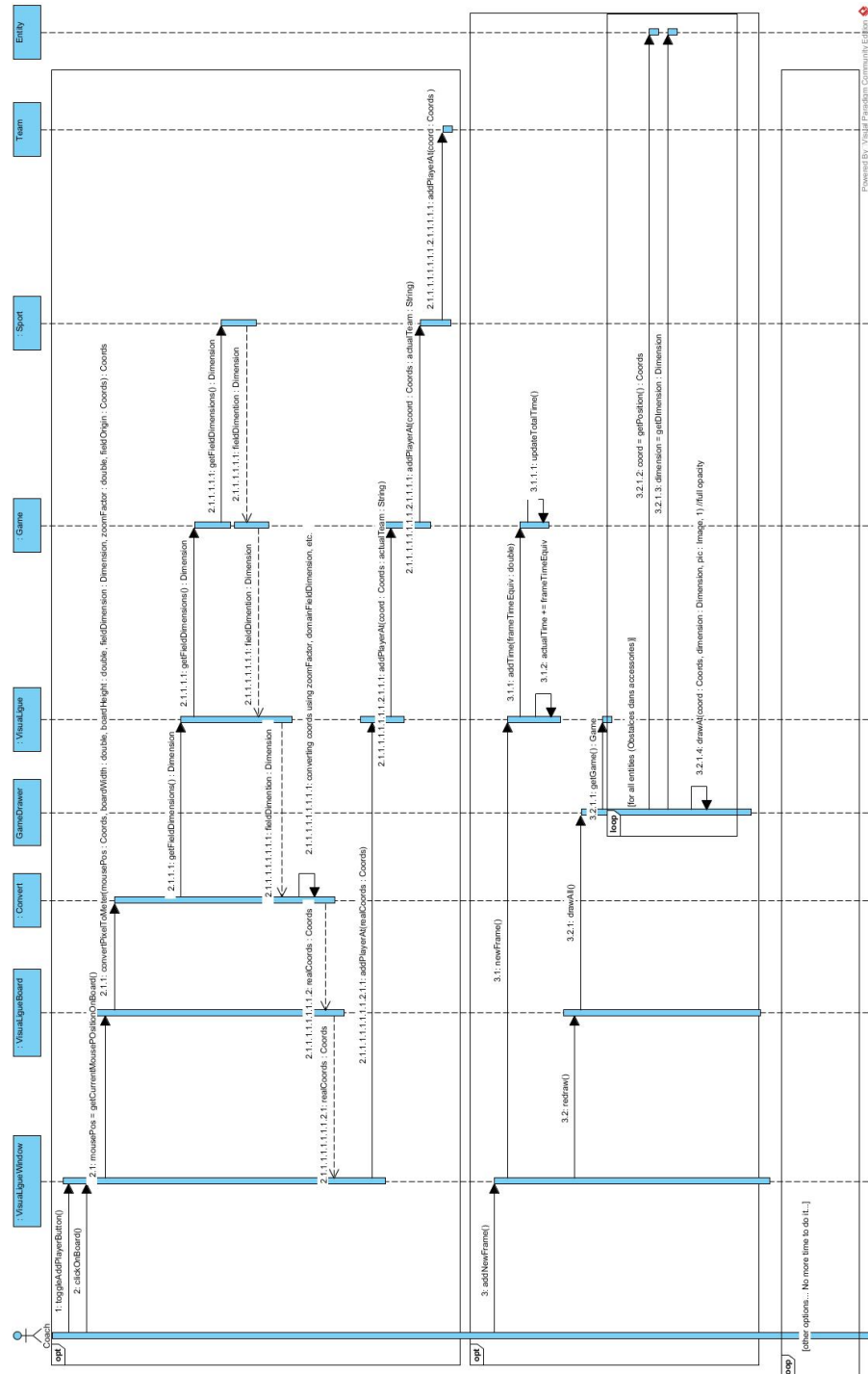
1. L'entraîneur appuie sur le toggle button afin d'activer l'option d'ajouter un joueur
2. La position du clic est enregistrée lorsque l'entraîneur clique sur l'interface où il désire placer son joueur
3. Le contrôle VisualLigue transmet l'information à Game qui ajoutera le joueur à la position souhaitée

### #3 - Détecter l'endroit d'un clic

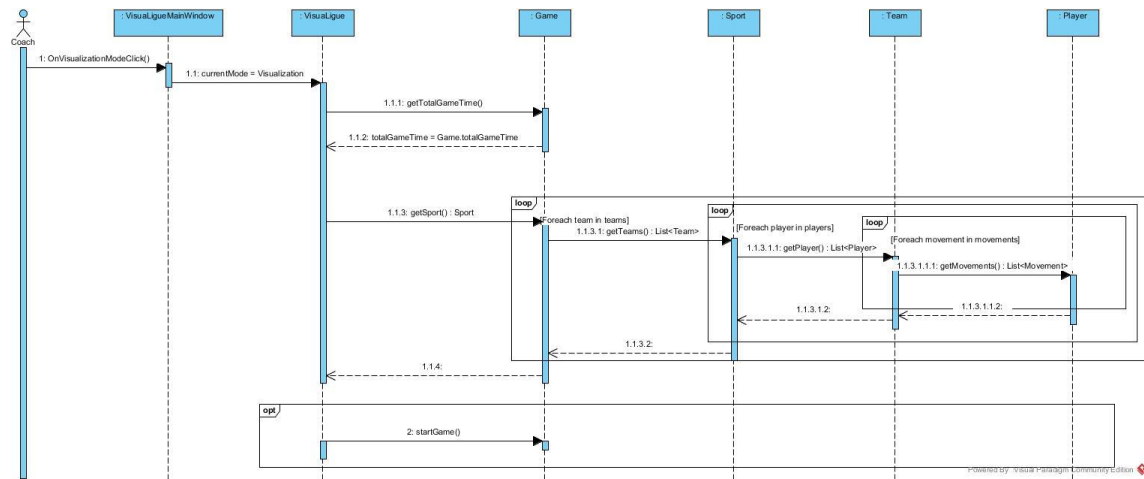


Lorsqu'un clic est effectué sur l'interface, VisualLigue vérifie si à la position du clic il y a soit un joueur, soit un obstacle soit un accessoire. Étant donné qu'une position (x,y) peut contenir plusieurs éléments l'un par-dessus l'autre (par exemple un joueur par-dessus un ballon), le dernier élément ayant été ajouté sur le terrain est retourné. En 1.1, on vérifie pour tous les obstacles s'il y en a un qui correspond à la position du clic, si oui on l'ajoute aux entités. Pour 1.1.2, on vérifie pour chaque joueur de toutes les teams s'il y a un joueur qui correspond à la position du clic, si oui, on l'ajoute aux entités et enfin, on vérifie si l'accessoire et à la position du clic puis on retourne la dernière entité sur lequel on a cliqué.

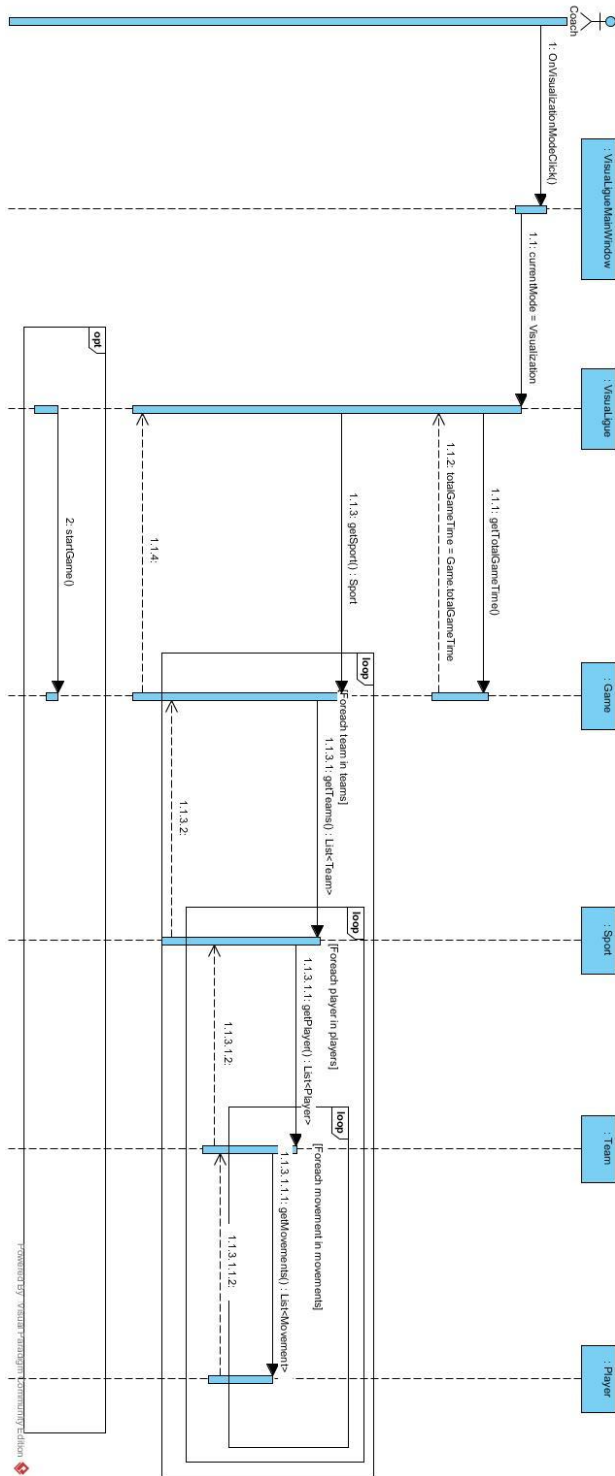
## #4 - Interactions de l'édition en mode image par image



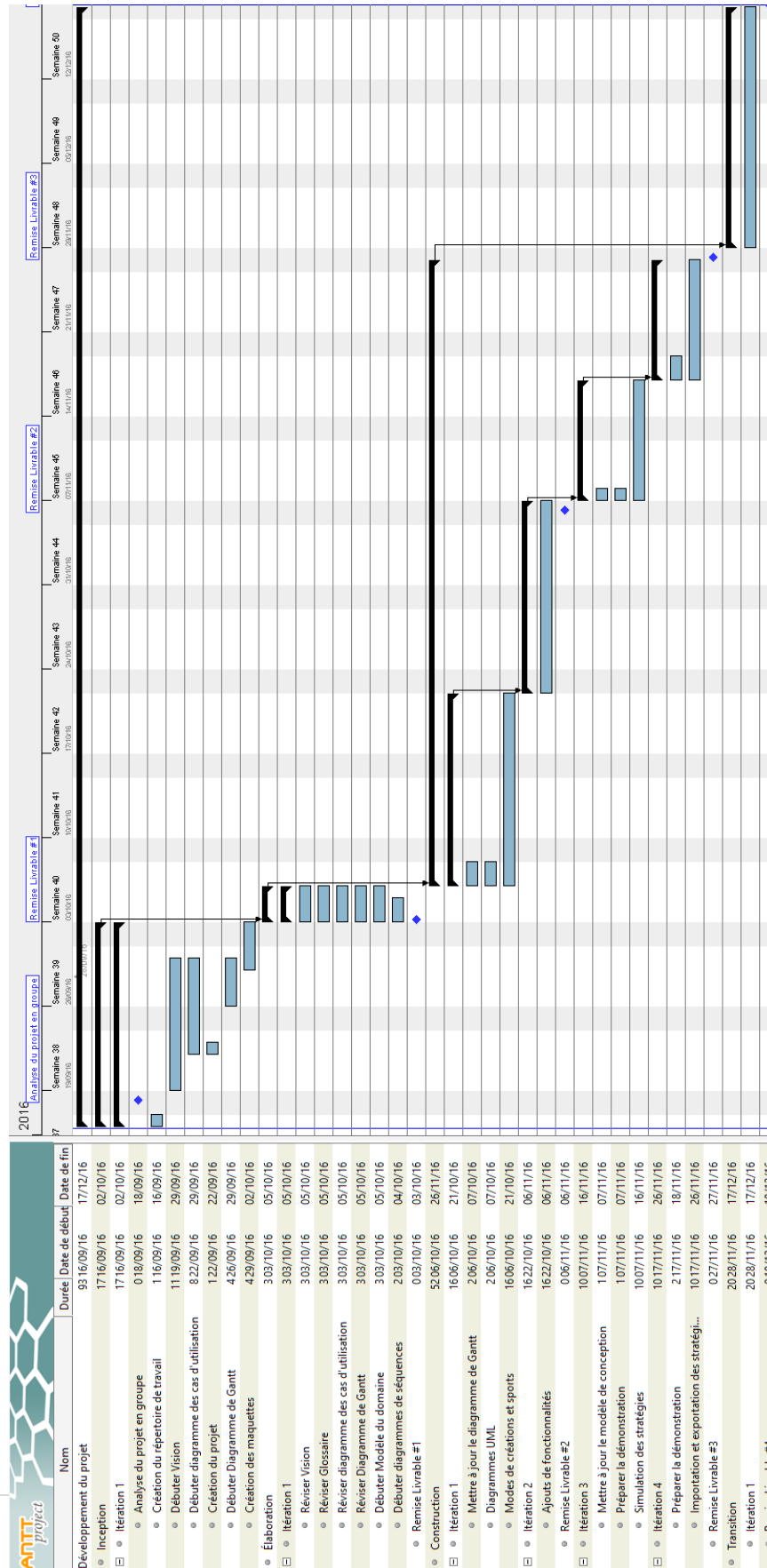
## #5 - Interactions de l'édition d'un jeu en mode temps réel



## #6 - Visionnage des jeux



## Diagramme de Gantt (actualisé)



## Annexes

### #1 Vision

#### Introduction

L'AEMQ (Association des entraîneurs mineurs) désire mettre à jour les outils utilisés par les entraîneurs qu'elle emploie. Présentement, ces derniers utilisent un tableau afin d'y effectuer des dessins pour expliquer aux membres de leur équipe les jeux à utiliser en fonction des situations qui se présentent. Il n'est malheureusement pas aisé pour l'entraîneur d'enseigner efficacement ses jeux à ses élèves à cause du support visuel désuet.

Suite à une rencontre avec l'AEMQ, notre start-up envisage donc de développer Visualigue, une application qui permettrait d'afficher les jeux de façon dynamique et en temps réel sur un écran plat, en plus d'améliorer nettement la qualité des enseignements, cette dernière aura quelques options supplémentaires non réalisables sur un support papier.

#### Énoncé du problème

Lorsque les entraîneurs de l'AEMQ essaient d'expliquer les jeux à appliquer à leurs élèves, ils n'ont pas toujours de la facilité à comprendre. C'est dans l'optique de régler ces problèmes que Visualigue a été pensé, le fait de faciliter et de dynamiser les explications augmenterait les performances des joueurs tout en rendant moindre la difficulté aux entraîneurs de donner leur cours.

Ce rapport permettra de trouver des solutions afin de pallier le problème énoncé.

#### Description des parties prenantes

Il existe trois parties prenantes :

1. Les entraîneurs : étant donné que ce sont eux qui interagiront directement avec l'application afin de prodiguer un enseignement meilleur
2. Les élèves : bien qu'ils aient moins d'intérêt que les entraîneurs, ces derniers restent tout de même concernés par le projet, les élèves pourraient démontrer de meilleurs résultats
3. L'AEMQ : en tant qu'association et mandateur, ils sont directement visés par l'application. Elle sera possiblement utilisée par tous leurs entraîneurs, quelle que

soit la discipline sportive (tant que ce soit un sport d'équipe où les joueurs interagissent avec un ballon, une balle ou une rondelle).

### Problèmes et objectifs clés des parties prenantes

L'AEMQ constate que les outils utilisés par les entraîneurs ne sont pas toujours efficaces. L'objectif de haut niveau de l'AEMQ est de simplifier l'enseignement dispensé par les entraîneurs tout en permettant de faciliter l'apprentissage des élèves à l'aide d'une application dynamique et interactive.

### Objectifs niveau-utilisateur

<i>Objectifs</i>	<i>Priorité</i>
<i>Objectif haut niveau</i>	Prioritaire
<i>Afficher clairement et dynamiquement les jeux sur un écran</i>	Haute
<i>Permettre aux entraîneurs de naviguer entre plusieurs modes de création</i>	Haute
<i>Sauvegarder un jeu</i>	Haute
<i>Application multisports</i>	Haute
<i>Modifier les paramètres de visionnement de la simulation des jeux</i>	Moyen
<i>Redémarrer un jeu</i>	Moyen

### Sommaires des bénéfices

<i>Caractéristiques</i>	<i>Bénéfices</i>
<i>L'application offre toutes les fonctionnalités nécessaires afin d'afficher les jeux sur un écran</i>	Permet aux entraîneurs de dispenser leurs cours de manière optimale et dynamique
<i>Interface sobre et optimisée</i>	Permet aux élèves une meilleure compréhension des jeux à appliquer
<i>L'application contient tous les éléments nécessaires pour configurer les paramètres d'un jeu (position des joueurs, modes de jeu, paramètres de visionnement)</i>	Visualiser un jeu avec une interface conviviale (déplacements des joueurs, etc.)



<i>L'application permet de sauvegarder les jeux</i>	Retrouver un ancien jeu
<i>L'application fonctionne également avec d'autres sports en équipe</i>	Tous les entraîneurs de l'AEMQ pourront en bénéficier, peu importe le sport qu'ils enseignent

### Sommaire des caractéristiques du système

- Création de jeu ainsi que la sauvegarde de ces derniers
- Les jeux doivent être accessibles dans l'application au chargement
- Liste de jeux présents avec aperçu/titre
- 2 modes de création pour les jeux :
  - Mode image par image
  - Mode en temps réel
- Modifier les paramètres de visionnement de la simulation des jeux (débuter, pause, reculer, avancer)
- Application multisports
- Annuler/rétablir
- Exporter les jeux
- Zoomer/dézoomer
- Options relatives aux joueurs (coordonnées, afficher/cacher les rôles, etc.)
- Ajout d'obstacles

### Autres exigences

#### Exigences de qualité

Afin que les entraîneurs et les élèves aient la meilleure expérience possible lors de l'utilisation de l'application, il est primordial que cette dernière ait les attributs suivants :

- La gestion des positions des joueurs doit se faire de manière intuitive étant donné que les entraîneurs passeront une grande partie de leur temps à le faire, les boutons et menus permettront de comprendre promptement chacune des fonctions de l'interface.
- Ergonomie sobre et efficace : l'interface graphique doit afficher clairement le jeu en cours ainsi que les jeux. Sur le jeu, tous les joueurs doivent être clairement affichés tout comme la position de la souris. Les utilisateurs de l'application ne

doivent pas se poser de questions quant aux différentes options, tout doit être clair.

- Le visionnement du jeu devra être fluide, reculer, avancer ou encore mettre pause doit se faire de façon naturelle

#### Exigences de performance

- L'exportation des jeux dans un format image doit être rapide
- Générer un jeu doit être rapide
- Sauvegarder un jeu doit être rapide

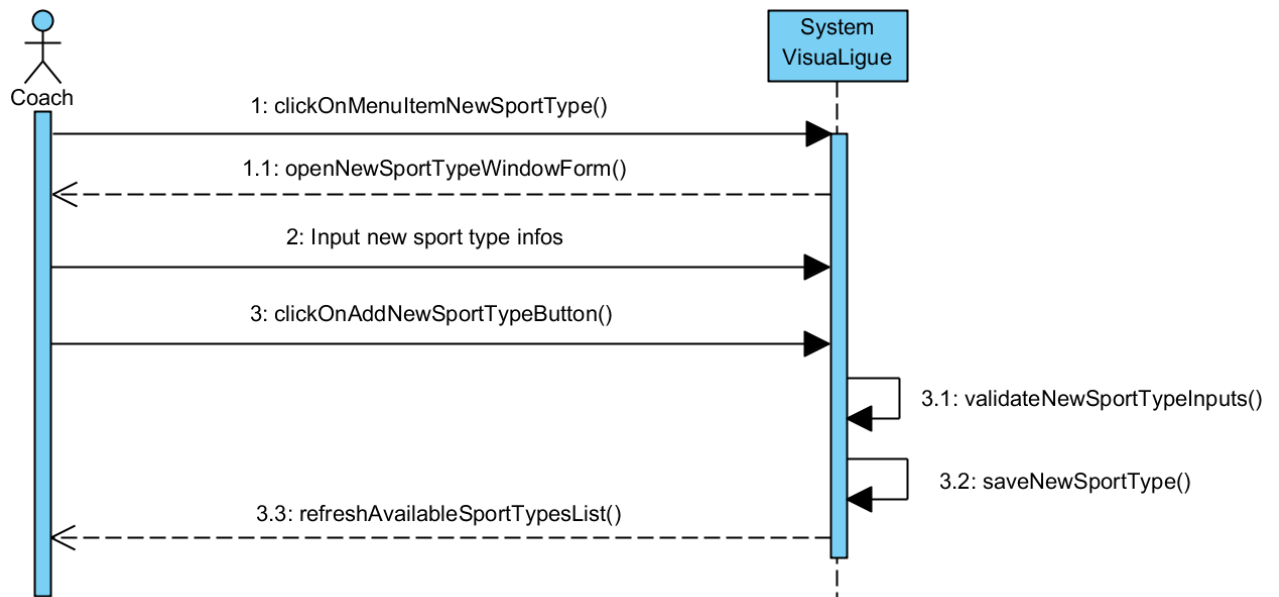
## #2 Cas d'utilisations

### Créer un type de sport

#### Détails

Cas d'utilisation :	Créer un type de sport
Système :	Visualigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	L'application doit être adaptable pour pouvoir l'utiliser avec d'autres sports d'équipe où les joueurs interagissent avec un ballon, une balle ou une rondelle
Préconditions :	Aucune
Garanties en cas de succès :	Lors de la création d'un nouveau jeu, ce nouveau sport apparaît maintenant dans la liste des sports disponibles.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Nouveau », et ensuite « Sport » dans le menu.</li><li>2. Une interface qui permet d'ajouter une image, de représenter le terrain, de définir les lignes sur le terrain, de choisir les dimensions du terrain, de définir le nombre des joueurs, leurs catégories des joueurs apparaît.</li><li>3. L'entraîneur définit tous ces paramètres, choisit un nom pour le sport et clique sur « Ajouter ».</li><li>4. Visualigue valide tous les champs saisis.</li><li>5. Visualigue enregistre le nouveau sport.</li><li>6. Visualigue affiche le nouveau sport dans la liste des sports disponibles.</li></ol>
Scénarios alternatifs :	Aucun

## Diagramme de séquences

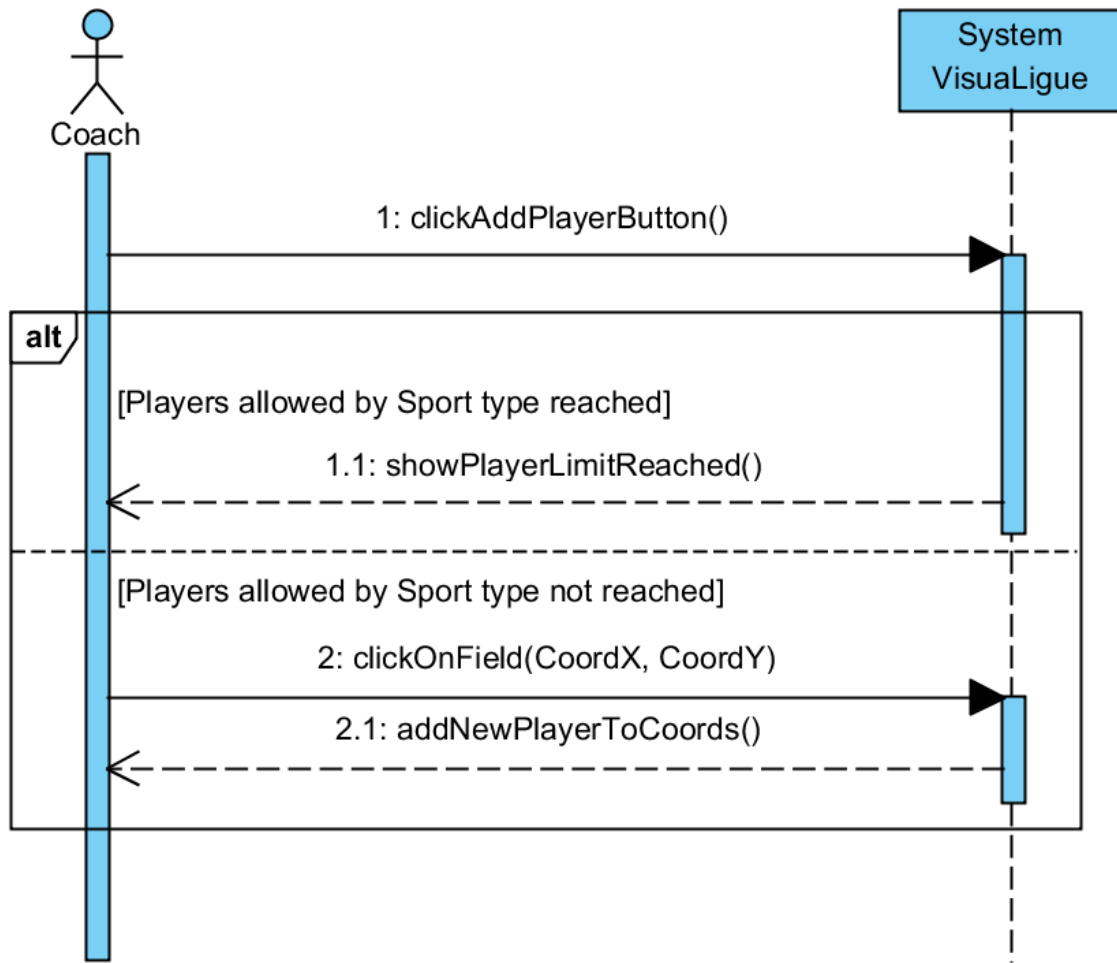


## Ajouter un joueur

### Détails

Cas d'utilisation :	Ajouter un joueur
Système :	VisualLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut ajouter un joueur dans le jeu.
Préconditions :	L'entraîneur est en train de créer un jeu et le type de sport doit être choisi.
Garanties en cas de succès :	On voit un nouveau joueur dans le jeu.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur le bouton « Ajouter un joueur » dans le menu latéral.</li><li>2. VisualLigue vérifie si le nombre de joueur permis par le sport est atteint.</li><li>3. La limite n'est pas atteinte alors un nouveau joueur apparaît sur le terrain du jeu.</li></ol>
Scénarios alternatifs :	<ol style="list-style-type: none"><li>3. La limite de joueur est atteinte.</li><li>4. VisualLigue afficher un message d'erreur à l'entraîneur</li><li>5. Aucun nouveau joueur s'affiche sur le terrain</li></ol>

## Diagramme de séquences

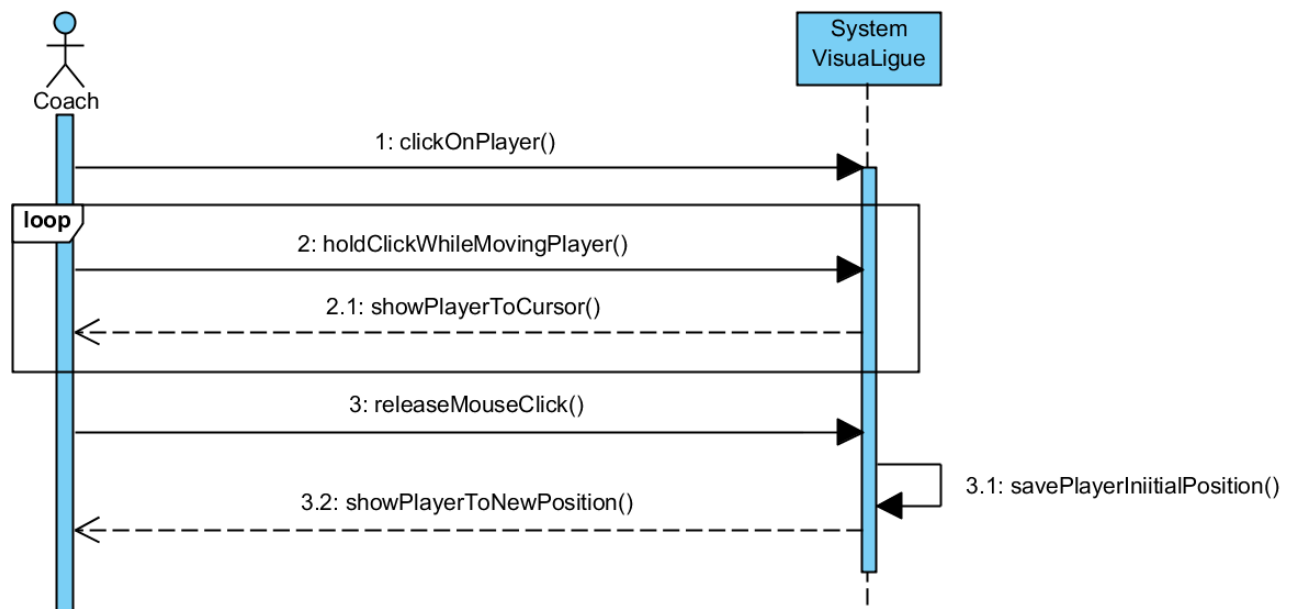


## Déplacer un joueur

### Détails

Cas d'utilisation :	Déplacer un joueur
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut changer la position initiale d'un joueur.
Préconditions :	Au moins un joueur doit être sur le terrain.
Garanties en cas de succès :	On voit maintenant le joueur à une nouvelle position initiale.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur déplace sa souris sur le joueur qu'il désire déplacer.</li><li>2. Il clique et maintient son curseur sur le joueur tout en effectuant le déplacement vers l'endroit initial souhaité.</li><li>3. VisuaLigue enregistre l'emplacement initial du joueur.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences

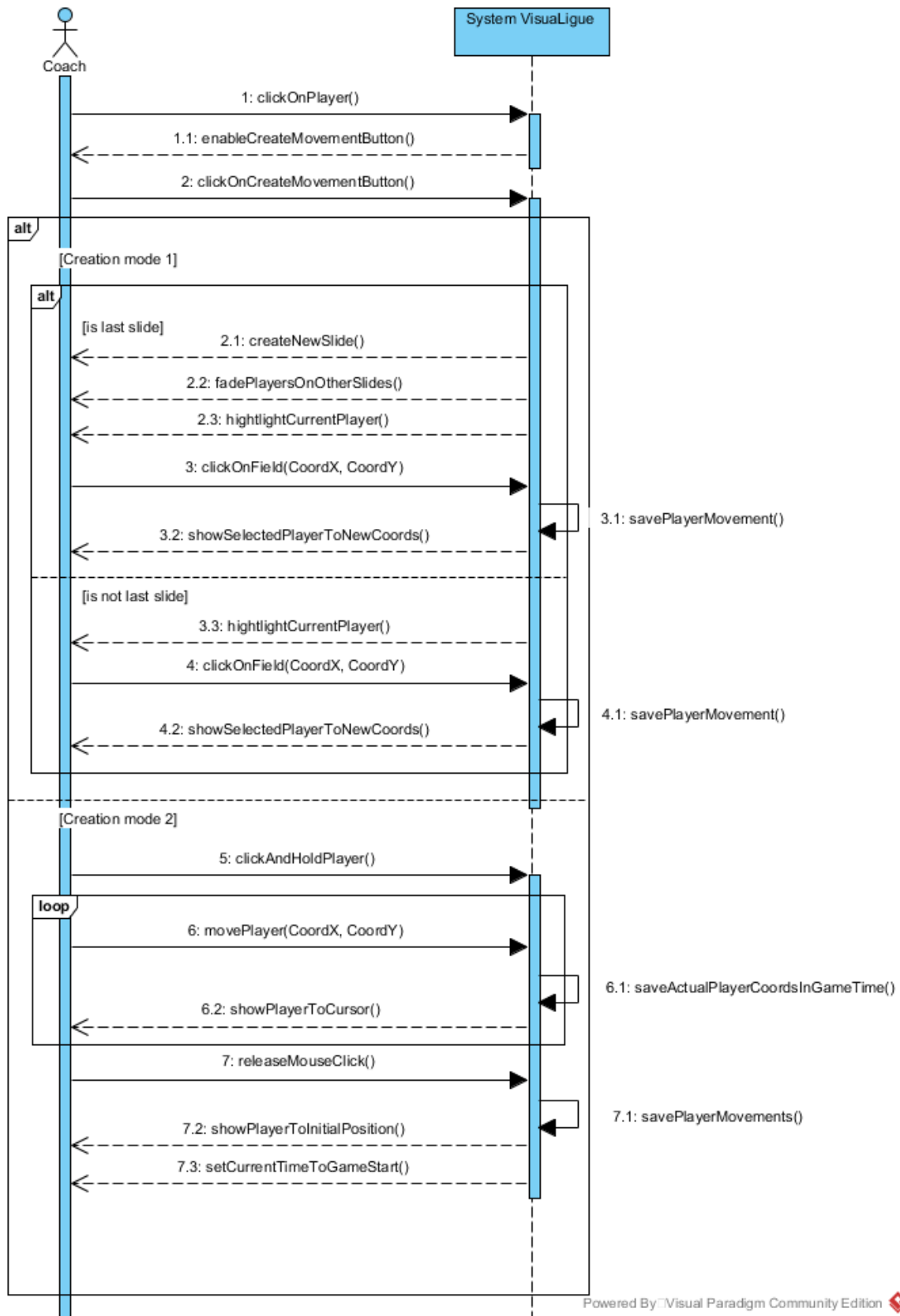


## Créer le déplacement d'un joueur

### Détails

Cas d'utilisation :	Créer le déplacement d'un joueur
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut créer le déplacement d'un joueur.
Préconditions :	L'entraîneur doit être en train de modifier un jeu (nouveau ou non). Le joueur que nous souhaitons déplacer est sur le terrain et le mode de création est choisi.
Garanties en cas de succès :	Le joueur est à sa nouvelle position (mode de création 1), ou on voit maintenant son déplacement (mode de création 2).
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur le joueur auquel il veut définir un déplacement.</li><li>2. L'entraîneur clique sur "Créer un déplacement" dans le menu latéral.</li><li>3. Il clique et maintient son curseur sur le joueur tout en effectuant le déplacement vers l'endroit souhaité (mode de création 1).</li><li>4. VisuaLigue affiche le joueur à son nouvel emplacement</li></ol>
Scénarios alternatifs :	<ol style="list-style-type: none"><li>3. Il clique et maintient son curseur sur le joueur tout en effectuant le déplacement complet d'un jeu en un temps réel</li><li>4. VisuaLigue enregistre le parcours du joueur.</li><li>5. VisuaLigue replace le joueur à l'emplacement de départ (pour faciliter la création des mouvements des autres joueurs) (mode de création 2).</li></ol>

## Diagramme de séquences



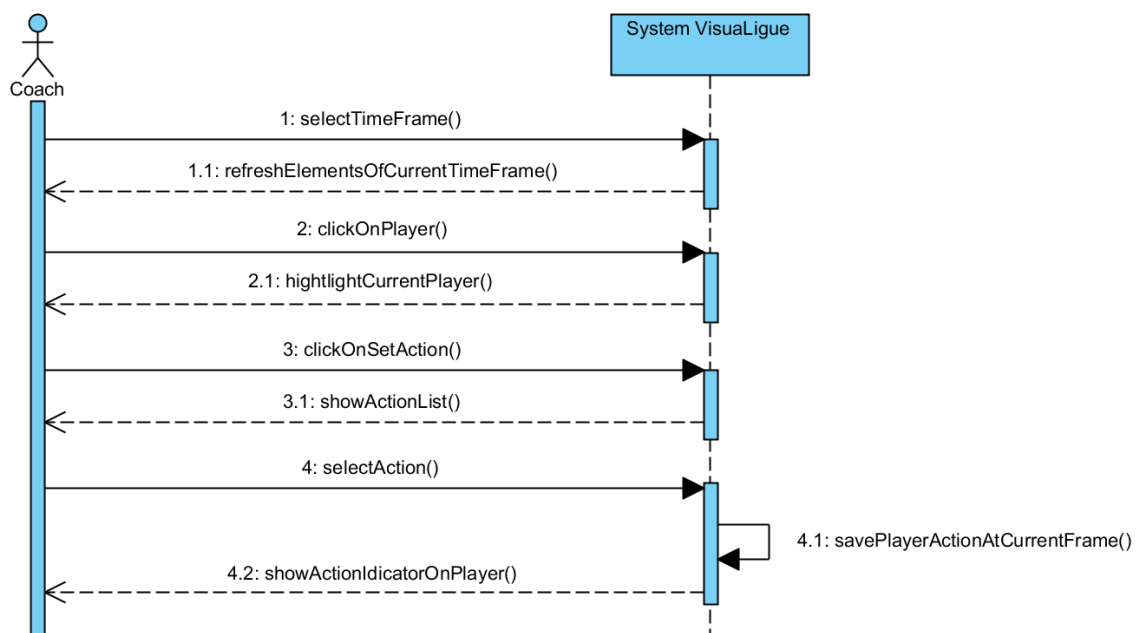
Powered By: Visual Paradigm Community Edition

## Définir l'action d'un joueur

### Détails

Cas d'utilisation :	Définir l'action d'un joueur.
Système :	VisualLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut définir l'action d'un joueur, par exemple faire une passe à un autre joueur, tenter de compter un but, intercepter la passe de l'adversaire, aller sur le banc, etc.
Préconditions :	Il doit avoir au moins un joueur sur le terrain. Préféablement celui dont nous voulons assigner une action.
Garanties en cas de succès :	Il y a une indication visuelle de l'action du joueur. Un trait bleu indiquera la direction de l'accessoire, un trait vert indiquera une tentative de faire un but, et un trait noir pointillé indiquera une tentative d'interception de passe d'un adversaire.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur choisit une image dans le jeu.</li><li>2. L'entraîneur choisit un joueur pour lequel il veut définir une action et il clique dessus.</li><li>3. L'entraîneur choisit dans une liste déroulante d'actions qu'est-ce que le joueur devrait faire comme action.</li><li>4. Si c'est le cas pour une telle action, l'entraîneur choisit de cliquer sur le joueur auquel il veut intercepter une passe, ou le but où il va tenter de compter, etc.</li><li>5. On voit un trait sur le jeu qui représente l'action du joueur.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences



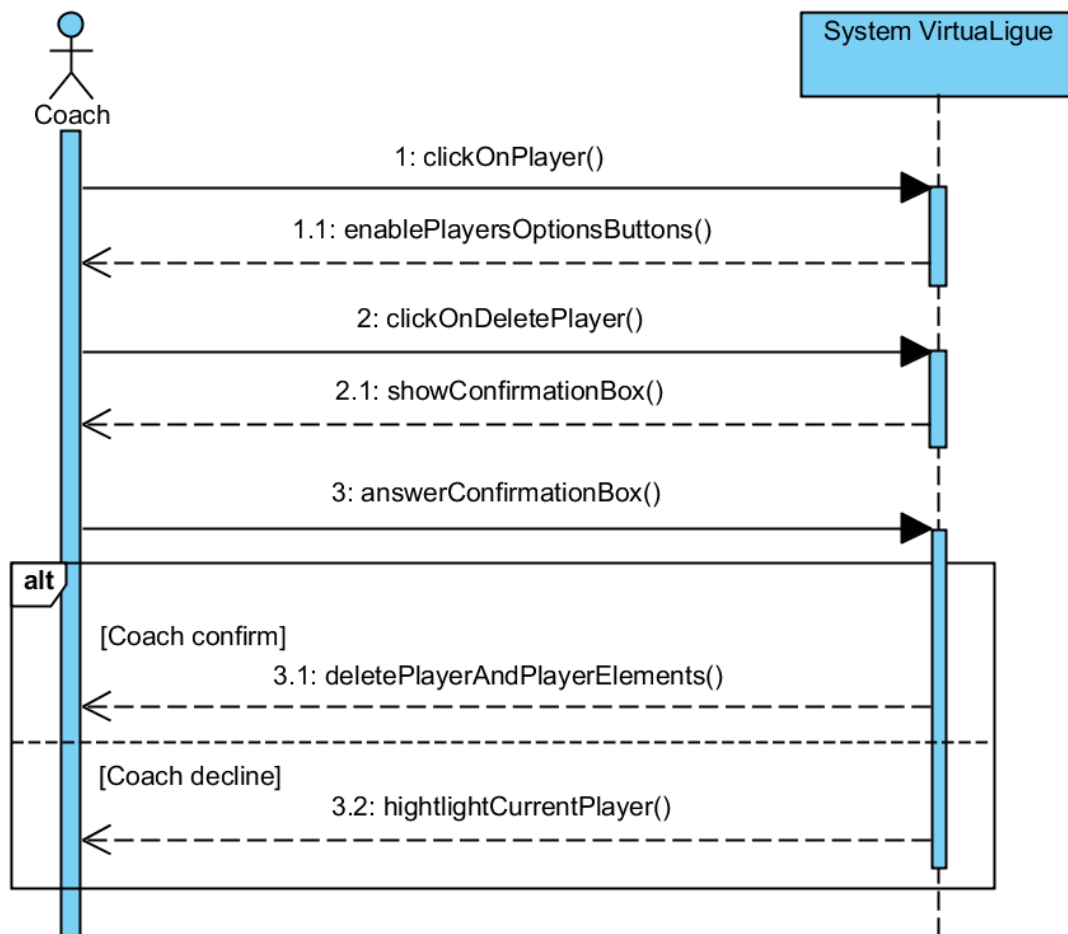


## Supprimer un joueur

### Détails

Cas d'utilisation :	Supprimer un joueur
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut supprimer un joueur dans le jeu.
Préconditions :	Il doit avoir au moins un joueur sur le terrain.
Garanties en cas de succès :	Il ne doit plus rester de signe d'existence du joueur dont nous avons souhaité supprimer.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur un joueur</li><li>2. L'entraîneur clique ensuite sur "Supprimer le joueur" dans le menu latéral de l'application.</li><li>3. VisuaLigue demande une confirmation de suppression.</li><li>4. L'entraîneur accepte.</li><li>5. VisuaLigue efface le joueur et tous les éléments qui y sont reliés.</li></ol>
Scénarios alternatifs :	<ol style="list-style-type: none"><li>4. L'entraîneur refuse.</li><li>5. VisuaLigue annule la suppression.</li></ol>

### Diagramme de séquences

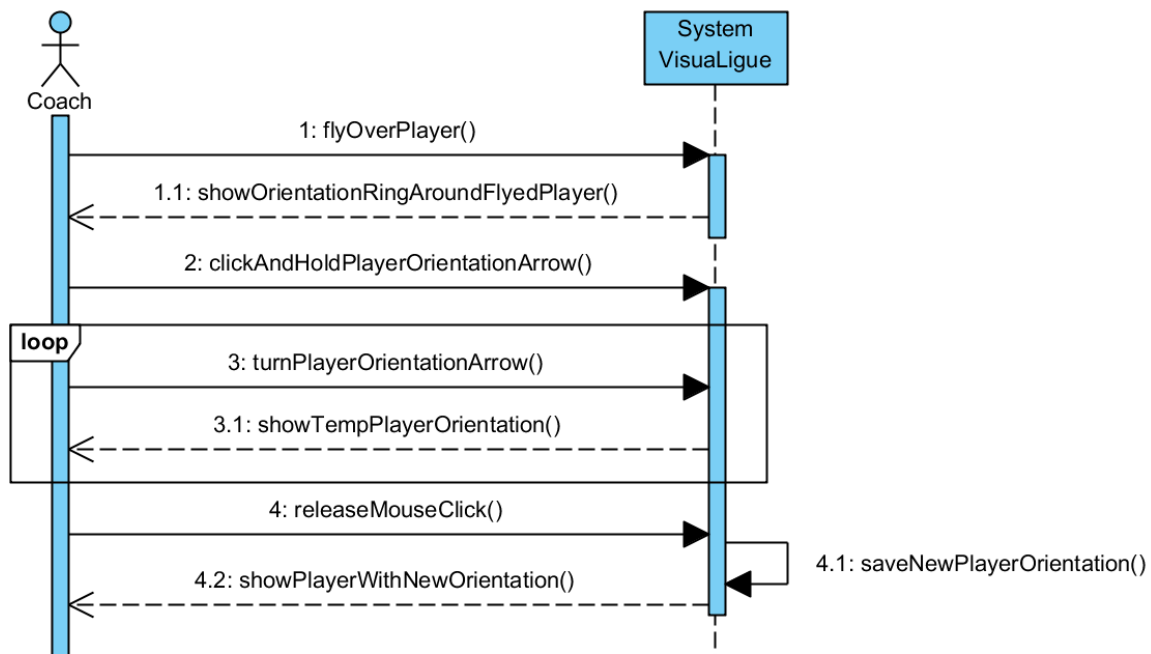


## Changer l'orientation d'un joueur

### Détails

Cas d'utilisation :	Changer l'orientation d'un joueur
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut changer l'orientation d'un joueur sur le jeu.
Préconditions :	Il doit y avoir un jeu ouvert avec au moins un joueur.
Garanties en cas de succès :	On voit que l'orientation du joueur est changée.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur survole un joueur</li><li>2. VisuaLigue affiche un cercle autour du joueur survolé avec son orientation actuelle représenté par une flèche.</li><li>3. L'entraîneur peut maintenant cliquer et maintenir la flèche du cercle du joueur pour lui définir une nouvelle orientation.</li><li>4. L'entraîneur relâche la flèche.</li><li>5. La nouvelle orientation du joueur est enregistrée.</li><li>6. VisuaLigue affiche le joueur avec la nouvelle orientation.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences



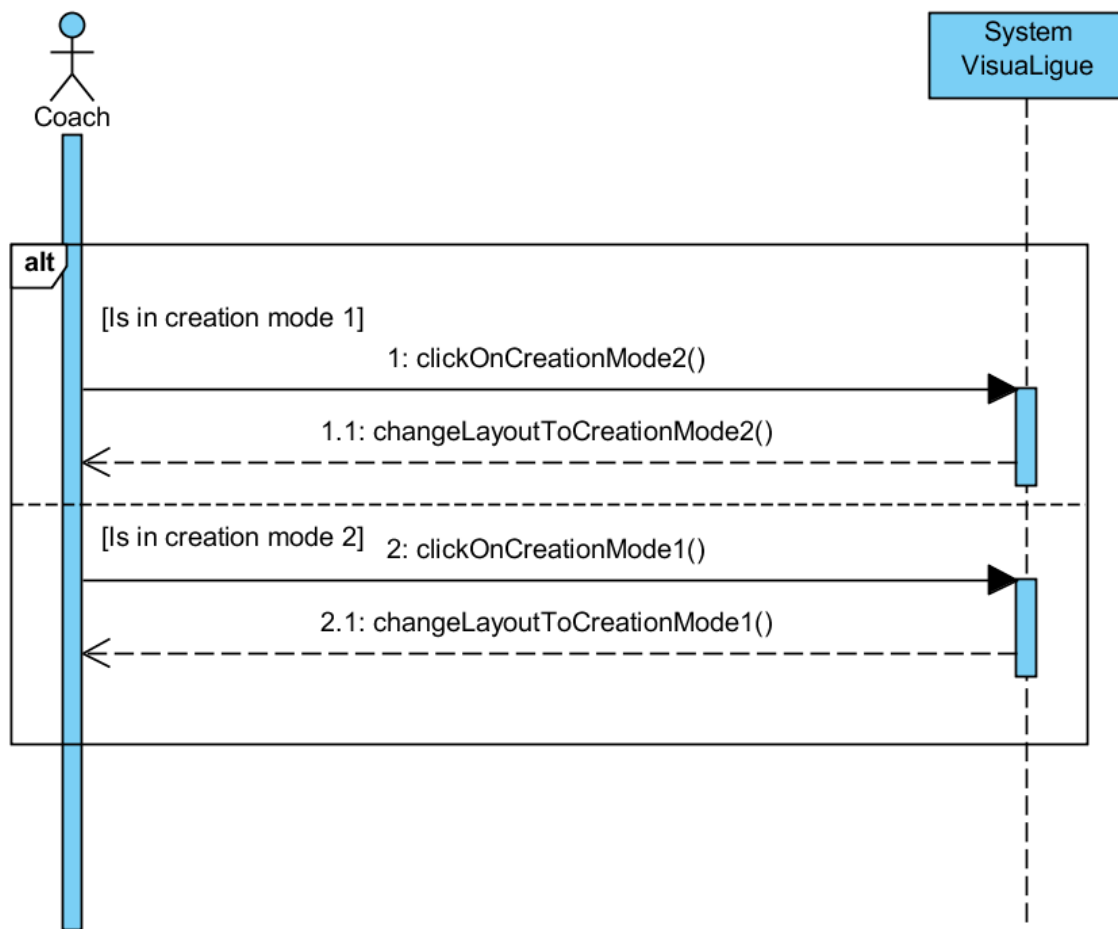
## Changer de mode de création

### Détails

Cas d'utilisation :	Changer de mode de création
Système :	VisuaLigue
Acteur(s) :	Entraîneur

Parties prenantes et intérêts :	Entraîneur : Veut changer le mode de création du jeu.
Préconditions :	Aucune
Garanties en cas de succès :	Le mode de création est changé et est affiché dans la zone d'information.
Scénario principal :	<p>L'entraîneur est en mode de création 1 (création par images) et souhaite aller en mode de création 2 (création en temps réel) :</p> <ol style="list-style-type: none"> <li>1. L'entraîneur clique sur « Mode de création » dans le menu et « Mode de création 2 ».</li> <li>2. VisuaLigue change d'affichage et la façon dont les données du domaine sont affichées.</li> </ol>
Scénarios alternatifs :	<p>L'entraîneur est en mode de création 2 (création en temps réel) et souhaite aller en mode de création 1 (création par images) :</p> <ol style="list-style-type: none"> <li>1. L'entraîneur clique sur "Mode de création" dans le menu et « Mode de création 1 ».</li> <li>2. VisuaLigue change d'affichage et la façon dont les données du domaine sont affichées.</li> </ol>

### Diagramme de séquences

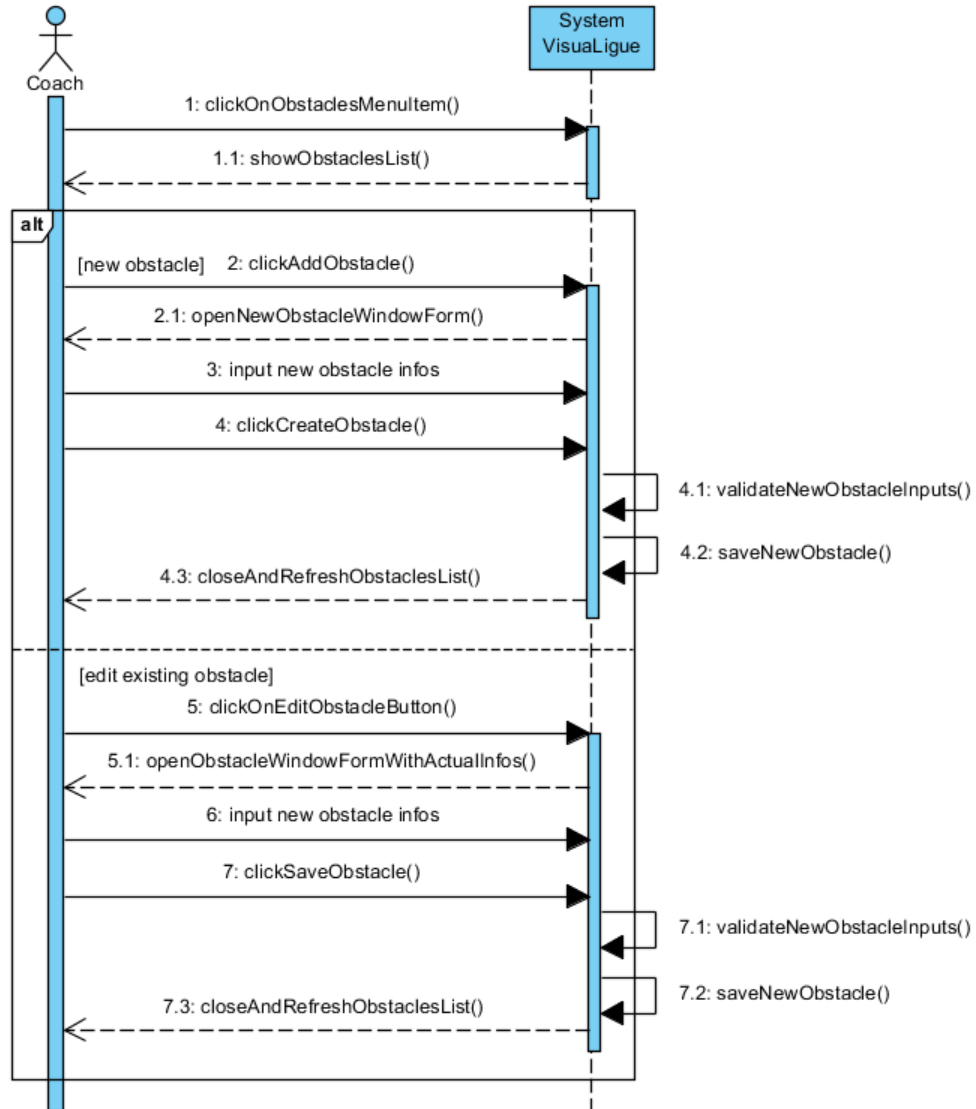


## Créer ou modifier un obstacle

### Détails

Cas d'utilisation :	Créer ou modifier un obstacle
Système :	Visualigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut créer ou modifier des obstacles du système.
Préconditions :	Aucune
Garanties en cas de succès :	On voit maintenant le nouvel obstacle dans la liste des obstacles disponibles.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur "Obstacles" dans le menu.</li><li>2. Visualigue ouvre une fenêtre contenant la liste des obstacles disponibles.</li><li>3. L'entraîneur veut ajouter un nouveau type d'obstacle donc il clique sur le bouton "+".</li><li>4. Visualigue ouvre une fenêtre avec les éléments à définir pour l'obstacle.</li><li>5. L'entraîneur nomme l'obstacle et choisit une image sur son ordinateur qui sera l'obstacle.</li><li>6. Visualigue valide les informations saisies et enregistre l'obstacle et ferme la fenêtre.</li><li>7. Visualigue affiche le nouvel obstacle dans la liste des obstacles.</li></ol>
Scénarios alternatifs :	<p>L'entraîneur veut modifier l'obstacle :</p> <ol style="list-style-type: none"><li>3. L'entraîneur veut modifier un obstacle existant donc il clique sur l'obstacle à modifier dans la liste.</li><li>4. Visualigue ouvre la fenêtre avec les éléments de l'obstacle pouvant être modifiés.</li><li>5. L'entraîneur apporte les modifications souhaitées et clique sur « Enregistrer ».</li><li>6. Visualigue valide les informations saisies et enregistre l'obstacle et ferme la fenêtre.</li><li>7. Visualigue affiche l'obstacle modifié dans la liste des obstacles.</li></ol>

## Diagramme de séquences

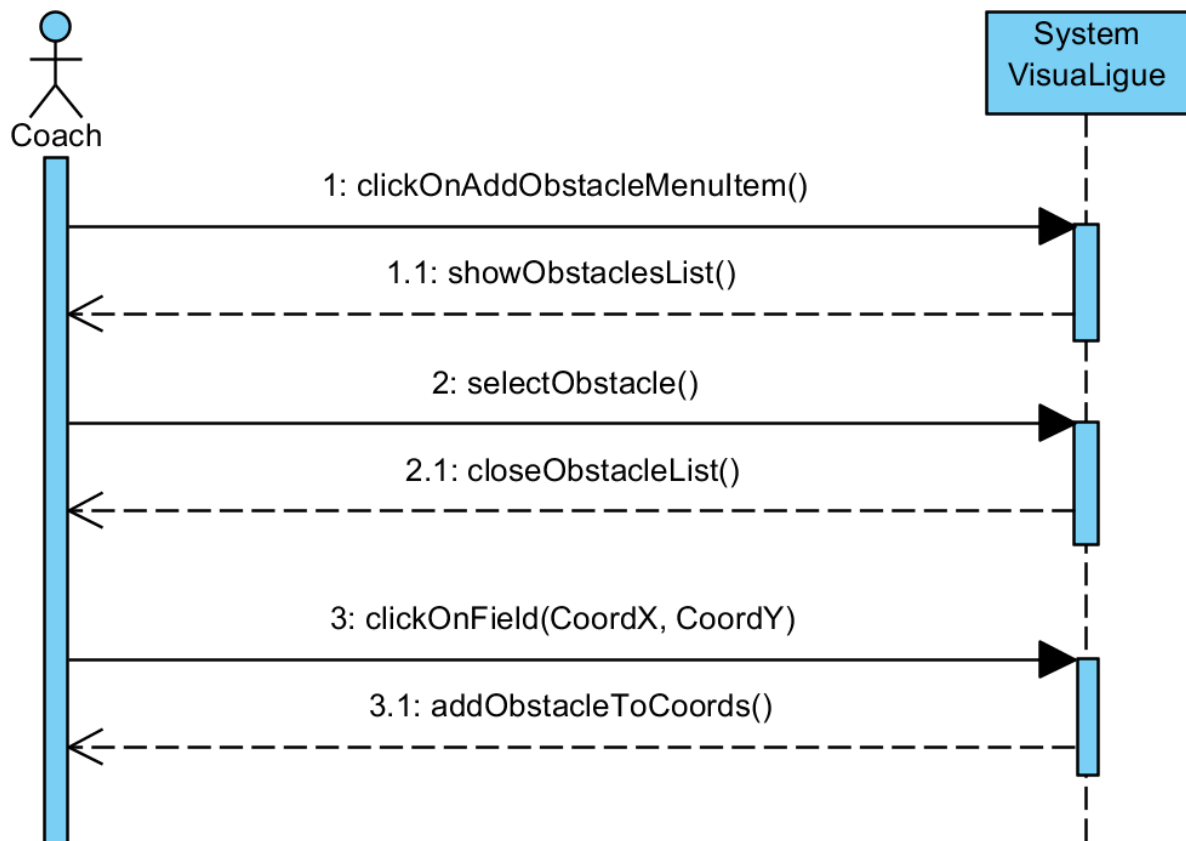


## Ajouter un obstacle au jeu

### Détails

Cas d'utilisation :	Ajouter un obstacle au jeu
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut ajouter un obstacle dans le jeu.
Préconditions :	Il doit y avoir un jeu d'ouvert.
Garanties en cas de succès :	On voit maintenant le nouvel obstacle sur le terrain du jeu à l'endroit désiré.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Ajouter un obstacle » dans le menu latéral.</li><li>2. Il choisit l'obstacle parmi les obstacles disponibles qu'il a ajouté au programme.</li><li>3. Il clique sur le terrain à l'endroit désiré pour insérer cet obstacle.</li><li>4. VisuaLigue affiche l'obstacle désiré à l'endroit cliqué.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences

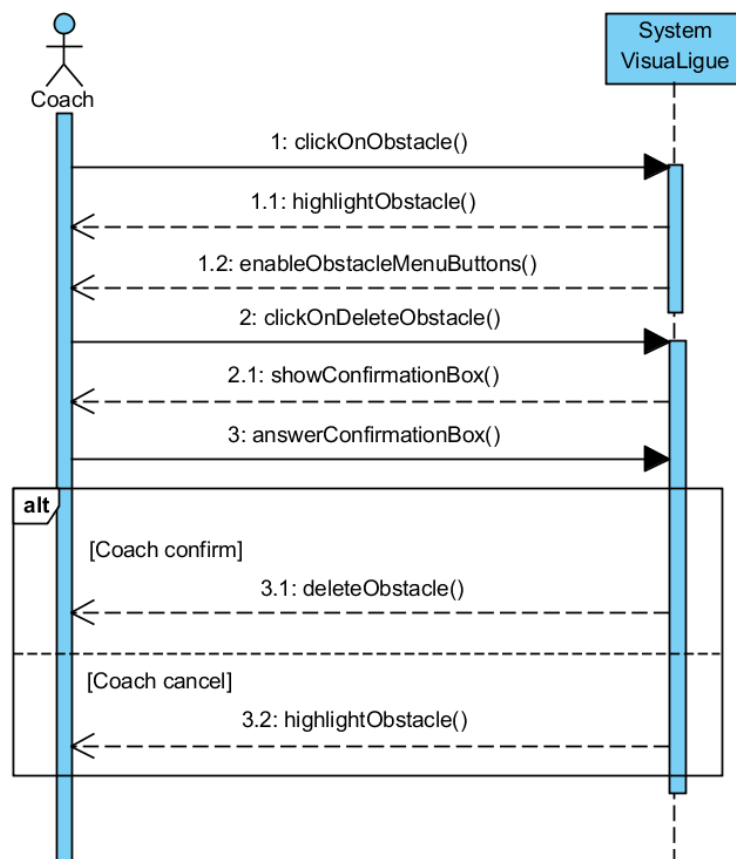


## Supprimer un obstacle du jeu

### Détails

Cas d'utilisation :	Supprimer un obstacle du jeu
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut supprimer un obstacle du jeu.
Préconditions :	Il doit y avoir un jeu d'ouvert. Un obstacle doit se trouver sur le terrain
Garanties en cas de succès :	L'obstacle cible ne doit plus se trouver sur le terrain du jeu
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur l'obstacle sur le terrain.</li><li>2. VisuaLigue affiche le bouton « X » (supprimer en rouge dans le menu latéral) comme disponible.</li><li>3. L'entraîneur clique sur ce bouton.</li><li>4. VisuaLigue demande une confirmation de suppression.</li><li>5. L'entraîneur confirme.</li><li>6. VisuaLigue supprime l'obstacle du terrain de jeu et ne s'affiche plus.</li></ol>
Scénarios alternatifs :	<ol style="list-style-type: none"><li>5. L'entraîneur annule la suppression.</li><li>6. VisuaLigue ne supprime pas l'obstacle du jeu et il reste sur le terrain du jeu.</li></ol>

### Diagramme de séquences

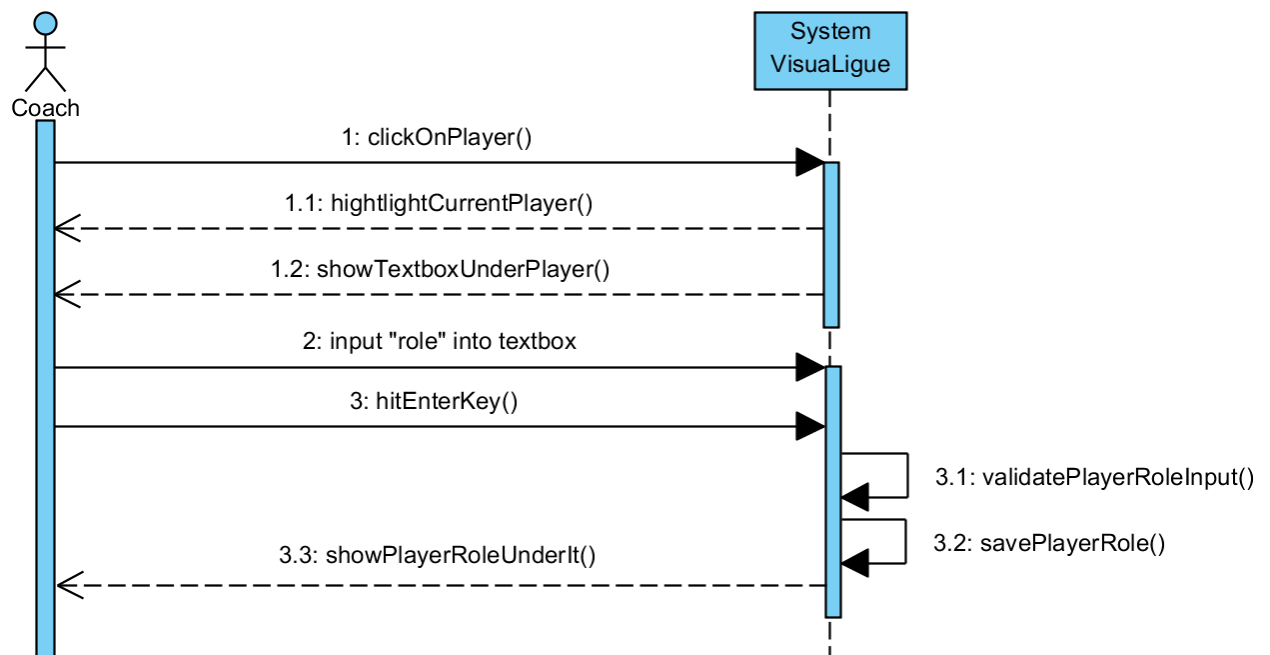


## Définir le rôle d'un joueur

### Détails

Cas d'utilisation :	Définir le rôle d'un joueur
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut choisir le rôle d'un joueur sur le terrain.
Préconditions :	Il doit y avoir un jeu ouvert et au moins un joueur sur le terrain.
Garanties en cas de succès :	On voit maintenant le rôle écrit sous le joueur sur le terrain.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur un joueur.</li><li>2. L'entraîneur écrit le rôle dans les informations du joueur.</li><li>3. VisuaLigue valide le rôle et l'enregistre au joueur.</li><li>4. VisuaLigue affiche ou non le rôle à l'écran en fonction des paramètres de l'application.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences



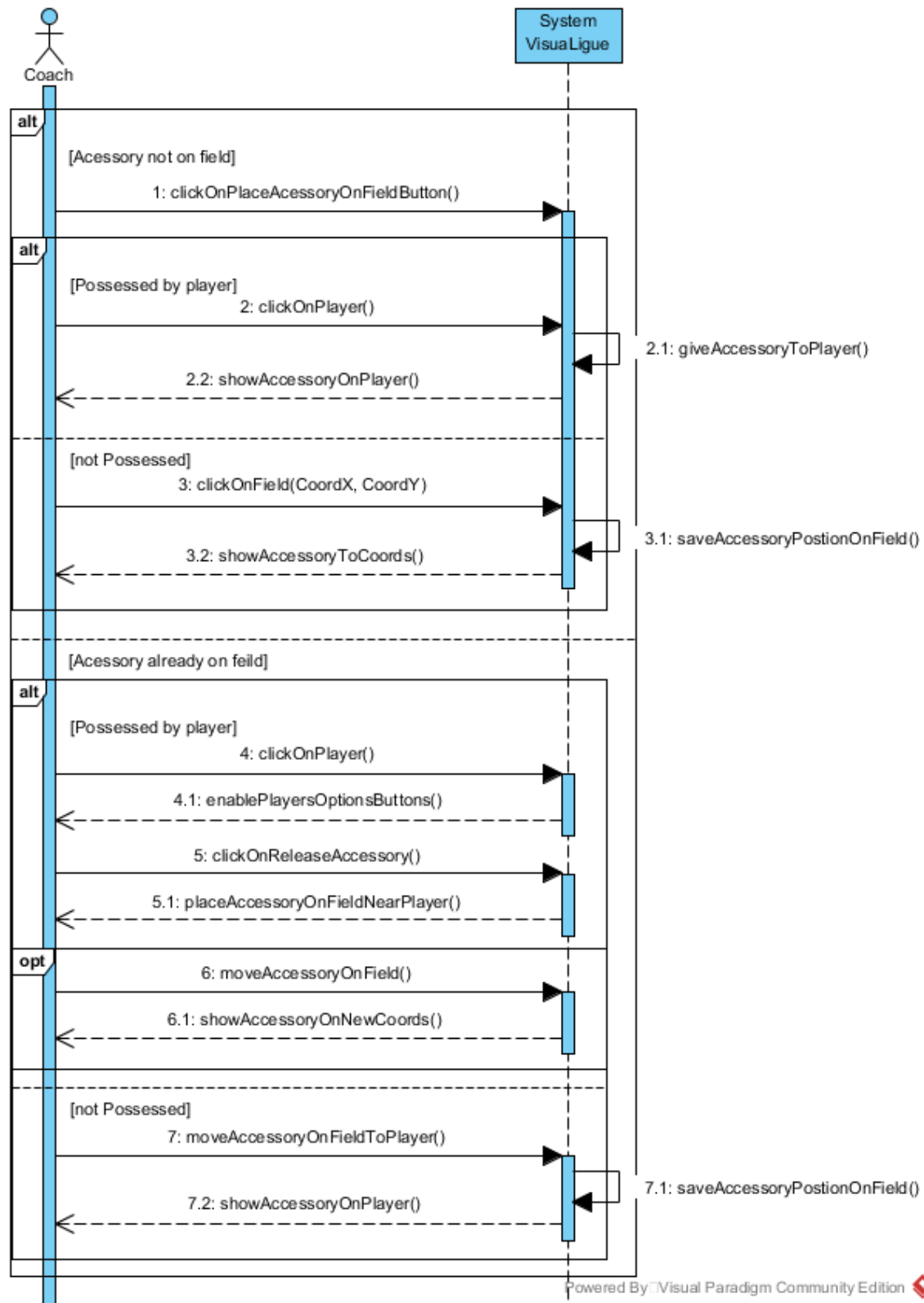


## Donner ou bouger l'accessoire de sport d'un joueur

### Détails

Cas d'utilisation :	Donner ou bouger l'accessoire de sport d'un joueur
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut donner, changer ou bouger l'accessoire de sport du joueur (balle, ballon, rondelle, etc.).
Préconditions :	Il doit y avoir un jeu ouvert avec au moins un joueur sur le terrain.
Garanties en cas de succès :	On voit maintenant l'accessoire de sport au nouvel emplacement souhaité
Scénario principal :	<p>L'accessoire n'est pas encore dans le jeu :</p> <ol style="list-style-type: none"><li>1. L'entraîneur clique sur le bouton « Accessoire de sport » dans le menu latéral.</li><li>2. Il clique ensuite sur un joueur à qui il donne cet accessoire.</li><li>3. VisuaLigue affiche l'image de l'accessoire de sport dans le coin du joueur pour afficher qu'il le possède.</li></ol>
Scénarios alternatifs :	<p>L'accessoire n'est pas encore dans le jeu :</p> <ol style="list-style-type: none"><li>1. L'entraîneur clique sur le bouton « Accessoire de sport » dans le menu latéral.</li><li>2. Il clique ensuite à l'endroit d'accessoire sur le terrain.</li><li>3. VisuaLigue affiche l'image de l'accessoire de sport à l'endroit souhaité sur le terrain</li></ol> <p>L'accessoire est déjà sur le terrain (libre).</p> <ol style="list-style-type: none"><li>1. L'entraîneur clique et maintient l'accessoire de sport et la déplace vers un joueur.</li><li>2. VisuaLigue affiche l'image de l'accessoire de sport dans le coin du joueur pour afficher qu'il le possède.</li></ol> <p>L'accessoire est déjà sur le terrain (possédé par un joueur)</p> <ol style="list-style-type: none"><li>1. L'entraîneur clique sur une fois sur le joueur.</li><li>2. VisuaLigue affiche le menu pour un joueur dans le menu latéral.</li><li>3. L'entraîneur clique sur "relâcher l'accessoire".</li><li>4. VisuaLigue place l'accessoire sur le terrain près du joueur.</li><li>5. L'entraîneur clique et maintient l'accessoire et le déplace à un nouvel endroit</li><li>6. VisuaLigue affiche l'image de l'accessoire de sport à l'endroit souhaité sur le terrain</li></ol>

## Diagramme de séquences

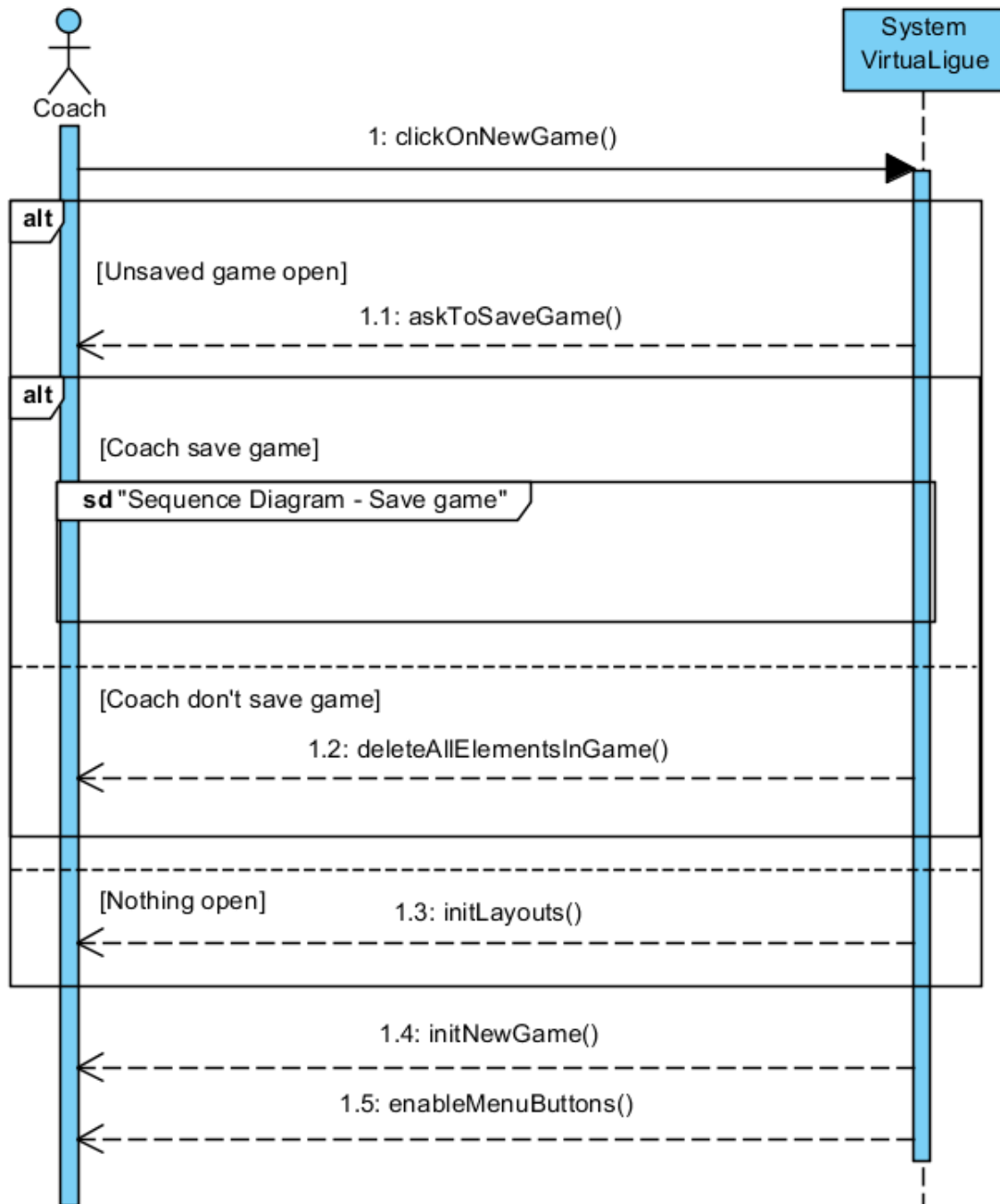


## Créer un nouveau jeu

### Détails

Cas d'utilisation :	Créer un nouveau jeu
Système :	VisualLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut créer un nouveau jeu.
Préconditions :	Aucune
Garanties en cas de succès :	L'entraîneur voit maintenant l'interface mise à zéro. (Aucun joueur, aucun obstacle, etc.)
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Nouveau », et puis « Jeu » dans le menu.</li><li>2. VisualLigue vérifie si un jeu non sauvegardé est en cours d'édition.</li><li>3. Le jeu ne comporte aucune modification ou aucun jeu n'était ouvert donc le système initialise un nouveau jeu</li><li>4. VisualLigue active les boutons du menu latéral pour l'édition.</li></ol>
Scénarios alternatifs :	<p>Dans le cas que l'entraîneur ne sauvegarde pas :</p> <ol style="list-style-type: none"><li>3. Un jeu non sauvegardé est en cours d'édition. Le système demande à l'entraîneur s'il veut sauvegarder le jeu.</li><li>4. L'entraîneur refuse de sauvegarder.</li><li>5. VisualLigue oublie les modifications et initialise un nouveau jeu.</li><li>6. VisualLigue active les boutons du menu latéral pour l'édition.</li></ol> <p>Dans le cas que l'entraîneur sauvegarde :</p> <ol style="list-style-type: none"><li>4. L'entraîneur veut sauvegarder.</li><li>5. VisualLigue ouvre une fenêtre et demande le nom du jeu.</li><li>6. L'entraîneur saisit le nom</li><li>7. VisualLigue valide le nom puis enregistre un fichier .vl (fichier VisualLigue)</li><li>8. Le système initialise un nouveau jeu.</li><li>9. VisualLigue active les boutons du menu latéral pour l'édition.</li></ol>

## Diagramme de séquences

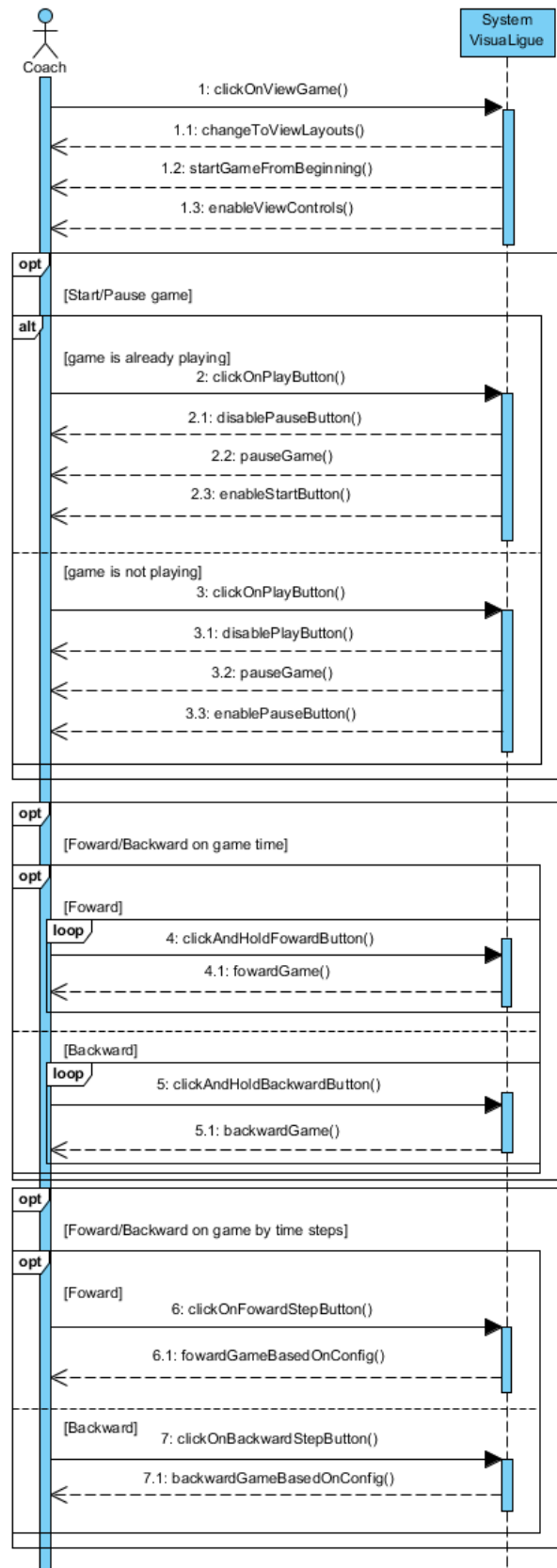


## Visionner une partie

### Détails

Cas d'utilisation :	Visionner un jeu
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut visualiser un jeu existant ou en cours d'édition.
Préconditions :	Un jeu est ouvert et au moins un déplacement pour un joueur est défini.
Garanties en cas de succès :	Les joueurs se déplacent automatiquement sur le terrain comme défini par l'entraîneur.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur le bouton « Visionner le jeu » dans le menu du bas.</li><li>2. VisuaLigue désactive tous les boutons d'édition du jeu et démarre l'exécution à partir du début.</li><li>3. VisuaLigue active les boutons permettant d'accélérer ou reculer dans le jeu, deux autres boutons pour faire des sauts en plus d'une zone de texte pour modifier la vitesse des sauts.</li></ol>
Scénarios alternatifs :	Aucun

## Diagramme de séquences

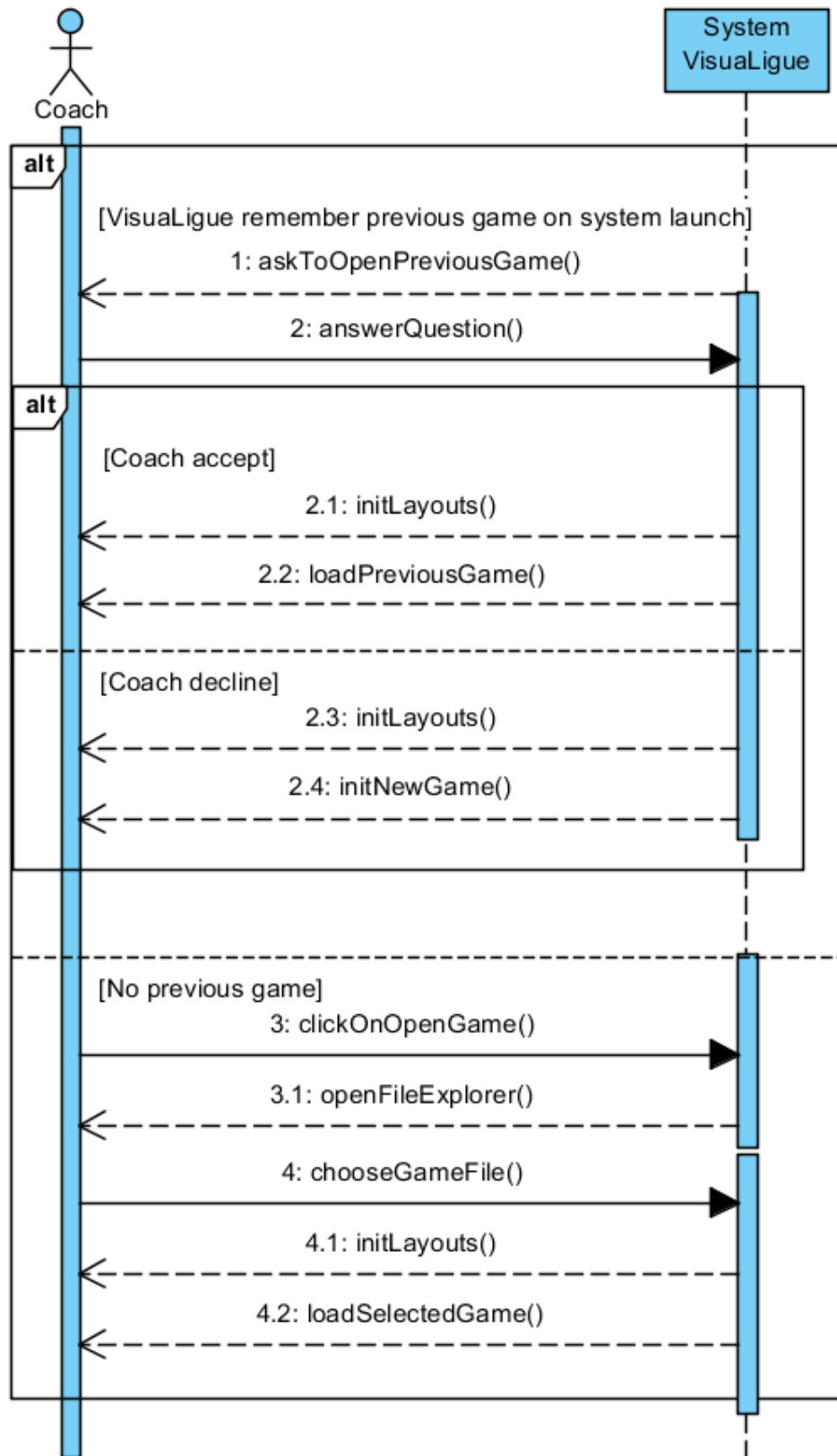


## Ouvrir une partie existante

### Détails

Cas d'utilisation :	Ouvrir un jeu existant
Système :	VisualLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut charger un jeu existant qu'il a déjà créé et sauvegardé.
Préconditions :	Un fichier .vl doit être disponible sur l'ordinateur de l'entraîneur.
Garanties en cas de succès :	Le programme affiche maintenant les éléments du jeu que l'utilisateur a chargé.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Fichier » puis « Ouvrir » dans le menu.</li><li>2. VisualLigue ouvre l'explorateur de fichiers.</li><li>3. L'entraîneur choisit le fichier .vl souhaité.</li><li>4. VisualLigue affiche une animation indiquant « Chargement en cours... ».</li><li>5. VisualLigue charge les éléments du jeu à ouvrir et récupère les configurations liées (mode de création, vitesse des bons de temps, etc.)</li></ol>
Scénarios alternatifs :	<p>Le programme a déjà sauvegardé un jeu avant sa fermeture :</p> <ol style="list-style-type: none"><li>1. À l'ouverture de VisualLigue, le système demande à l'entraîneur s'il souhaite charger le dernier jeu connu.</li><li>2. L'entraîneur accepte.</li><li>3. VisualLigue charge les éléments du jeu à ouvrir et récupère les configurations liées (mode de création, vitesse des bons de temps, etc.)</li></ol> <p>L'entraîneur refuse :</p> <ol style="list-style-type: none"><li>2. L'entraîneur refuse.</li><li>3. VisualLigue initialise un nouveau jeu.</li><li>4. VisualLigue active les boutons du menu latéral pour l'édition.</li></ol>

## Diagramme de séquences



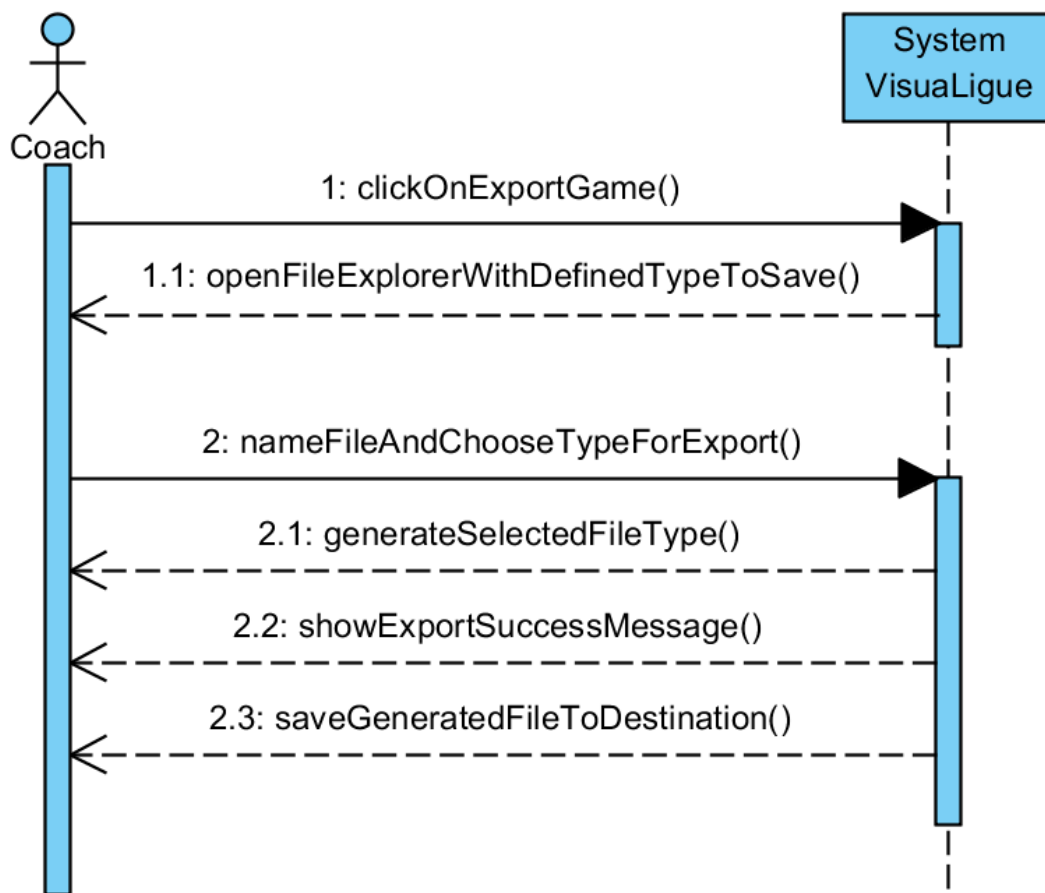


## Exporter une partie

### Détails

Cas d'utilisation :	Exporter un jeu
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut exporter un jeu dans un format d'image tel que PNG, ou JPEG.
Préconditions :	Un jeu est ouvert et au moins un déplacement pour un joueur est défini.
Garanties en cas de succès :	L'interface du programme donne une confirmation qui indique que le jeu a bien été exporté vers le type d'image souhaité.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Fichier », et puis « Exporter » dans le menu.</li><li>2. VisuaLigue ouvre l'explorateur de fichier qui offre les types d'exportation possibles pour la sauvegarde.</li><li>3. L'entraîneur choisit l'emplacement et le type d'image à exporter et confirme.</li><li>4. VisuaLigue génère une image avec une représentation des déplacements des joueurs vers le type d'image souhaité.</li><li>5. VisuaLigue affiche un message de succès pour l'exportation.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences

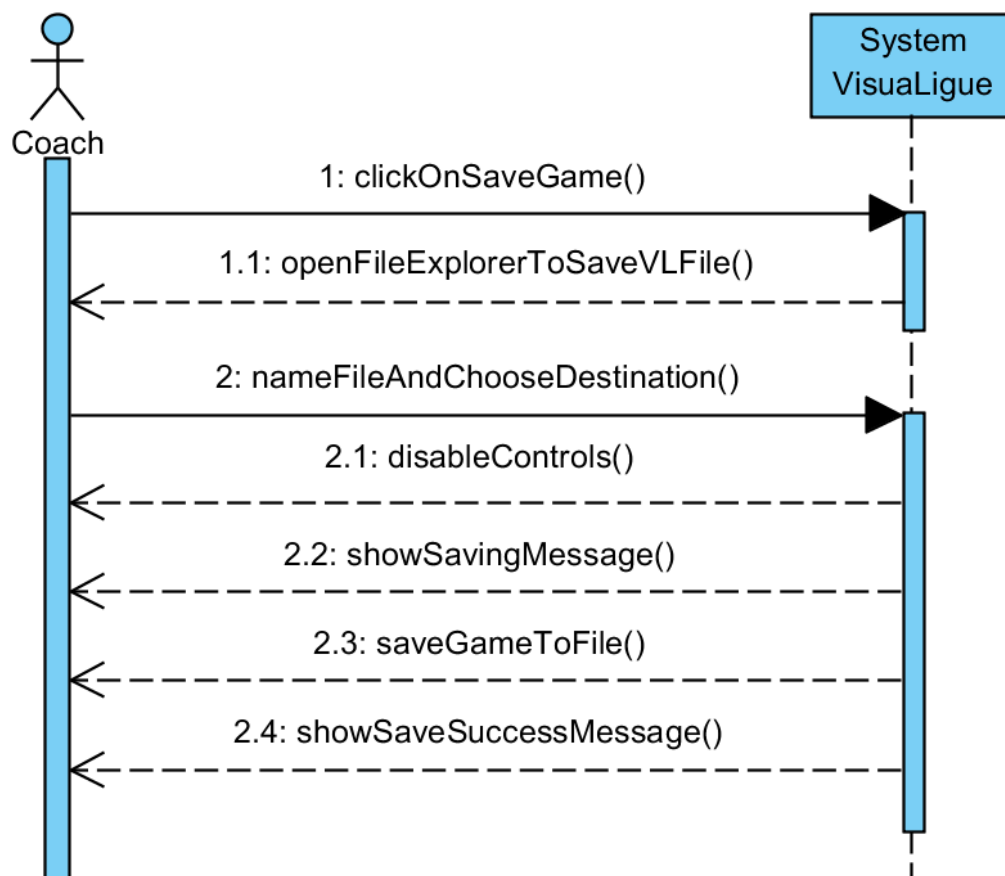


## Sauvegarder une partie

### Détails

Cas d'utilisation :	Sauvegarder un jeu
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut pouvoir sauvegarder un jeu pour pouvoir le visionner ou modifier plus tard.
Préconditions :	Un jeu est ouvert et il y a au moins une modification de fait.
Garanties en cas de succès :	Le système affiche un message de succès de l'enregistrement et le fichier est bien sur l'ordinateur de l'entraîneur.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Fichier » puis « Sauvegarder ».</li><li>2. VisuaLigue ouvre l'explorateur de fichier afin de choisir l'endroit pour enregistrer le fichier .vl.</li><li>3. L'entraîneur nomme le fichier, choisit l'emplacement puis clique sur "Enregistrer".</li><li>4. VisuaLigue bloque les contrôles et affiche "Enregistrement en cours...".</li><li>5. VisuaLigue effectue l'enregistrement des éléments du jeu dans le fichier.</li><li>6. VisuaLigue affiche un message de succès et les contrôles sont réactivés.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences

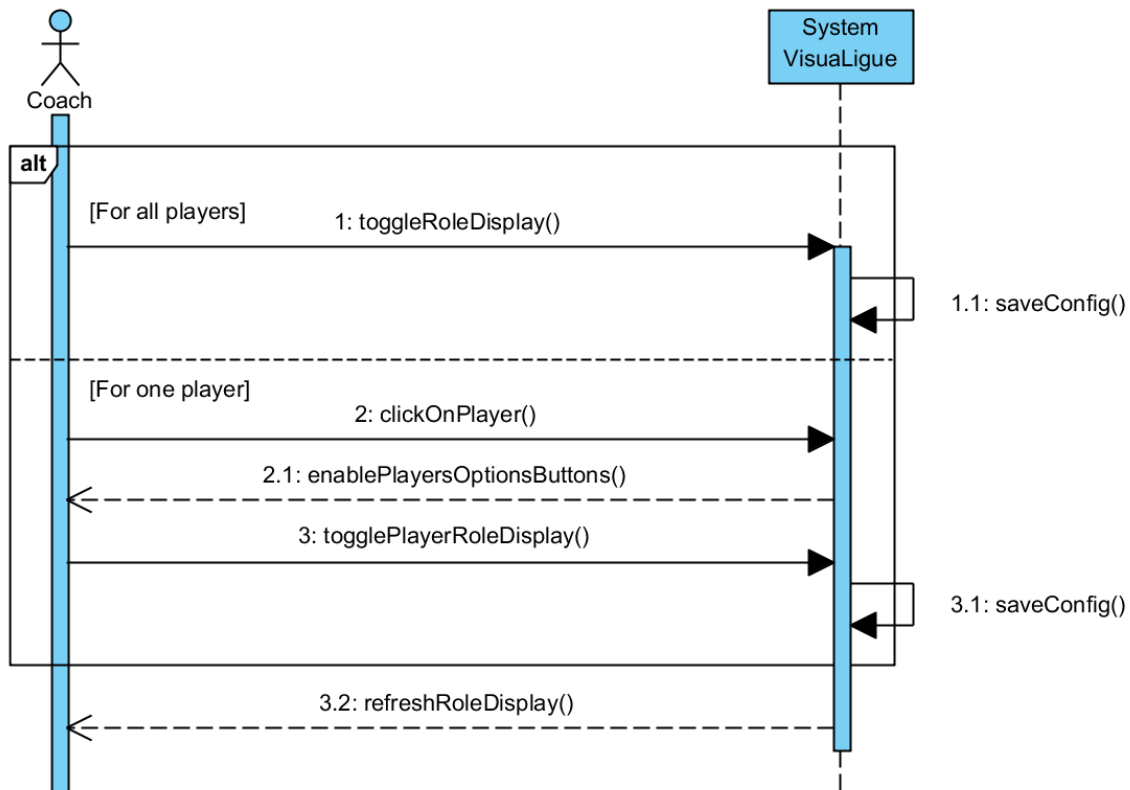


## Afficher ou cacher les rôles d'un ou des joueurs

### Détails

Cas d'utilisation :	Afficher ou cacher les rôles d'un ou des joueurs
Système :	VisualLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut pouvoir afficher ou cacher le rôle d'un joueur ou de tous les joueurs sur le terrain.
Préconditions :	Un jeu est ouvert et il y a au moins un joueur de placé dans le jeu. (Lorsqu'il s'agit de cacher le rôle de seulement un joueur)
Garanties en cas de succès :	Le ou les rôles ne sont plus affichés dans l'écran de jeu
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Affichage » puis « Afficher ou cacher les rôles ».</li><li>2. VisualLigue enregistre la nouvelle configuration d'affichage.</li><li>3. VisualLigue ajuste l'affichage en conséquence.</li></ol>
Scénarios alternatifs :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur un joueur.</li><li>2. VisualLigue active le bouton « afficher ou cacher le rôle » du menu latéral.</li><li>3. L'entraîneur clique sur ce bouton.</li><li>4. VisualLigue enregistre la nouvelle configuration d'affiche pour le joueur cible.</li><li>5. VisualLigue ajuste l'affichage en conséquence.</li></ol>

### Diagramme de séquences



## Démarrer ou mettre en pause une partie

### Détails

Cas d'utilisation :	Démarrer ou mettre en pause un jeu
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut pouvoir démarrer ou mettre en pause le visionnement d'un jeu.
Préconditions :	Il y a un jeu chargé dans le programme. (Ou un nouveau jeu avec au moins un joueur avec un déplacement)
Garanties en cas de succès :	Si les joueurs sont initialement immobiles, les joueurs sont maintenant en déplacement. S'ils étaient en déplacement, ils sont maintenant immobiles à nouveau.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur « Visionner » dans le menu.</li><li>2. VisuaLigue bascule en mode « visionnement » d'un jeu.</li><li>3. VisuaLigue active les contrôles de visionnements et désactive les contrôles d'édition.</li><li>4. L'entraîneur clique sur « Jouer ».</li><li>5. VisuaLigue désactive le bouton « Jouer » et active le bouton « Pause »</li><li>6. VisuaLigue débute le jeu et déplace les joueurs en fonction de leurs déplacements.</li></ol>

### Diagramme de séquences

(Voir l'option dans le diagramme de séquences de « Visionner une partie »)

## Avancer ou reculer l'exécution d'une partie

### Détails

Cas d'utilisation :	Avancer ou reculer l'exécution d'un jeu
Système :	VisuaLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut avancer ou reculer l'exécution d'un jeu qu'il est en train de visionner (avec changement de vitesse).
Préconditions :	Il y a un jeu d'ouvert et est en mode « visionnement ». Le jeu doit être en exécution. (Il doit être en train de visionner le jeu).
Garanties en cas de succès :	L'exécution se voit accéléré ou ralenti comme souhaité.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique une fois sur le bouton &gt;&gt; (avancer) ou &lt;&lt; (reculer)</li><li>2. VisuaLigue avance ou recule l'exécution du jeu d'un temps constant configurable dans les options.</li></ol>
Scénarios alternatifs :	L'exécution du jeu est déjà un peu plus vite (avant ou arrière) : <ol style="list-style-type: none"><li>1. L'entraîneur clique à nouveau le même bouton pour avancer ou reculer.</li><li>2. VisuaLigue augmente la vitesse d'avancement ou de reculement d'un temps constant configurable dans les options.</li></ol>

### Diagramme de séquences

(Voir l'option dans le diagramme de séquences de « Visionner une partie »)

## Avancer ou reculer l'exécution d'une partie par bond de temps

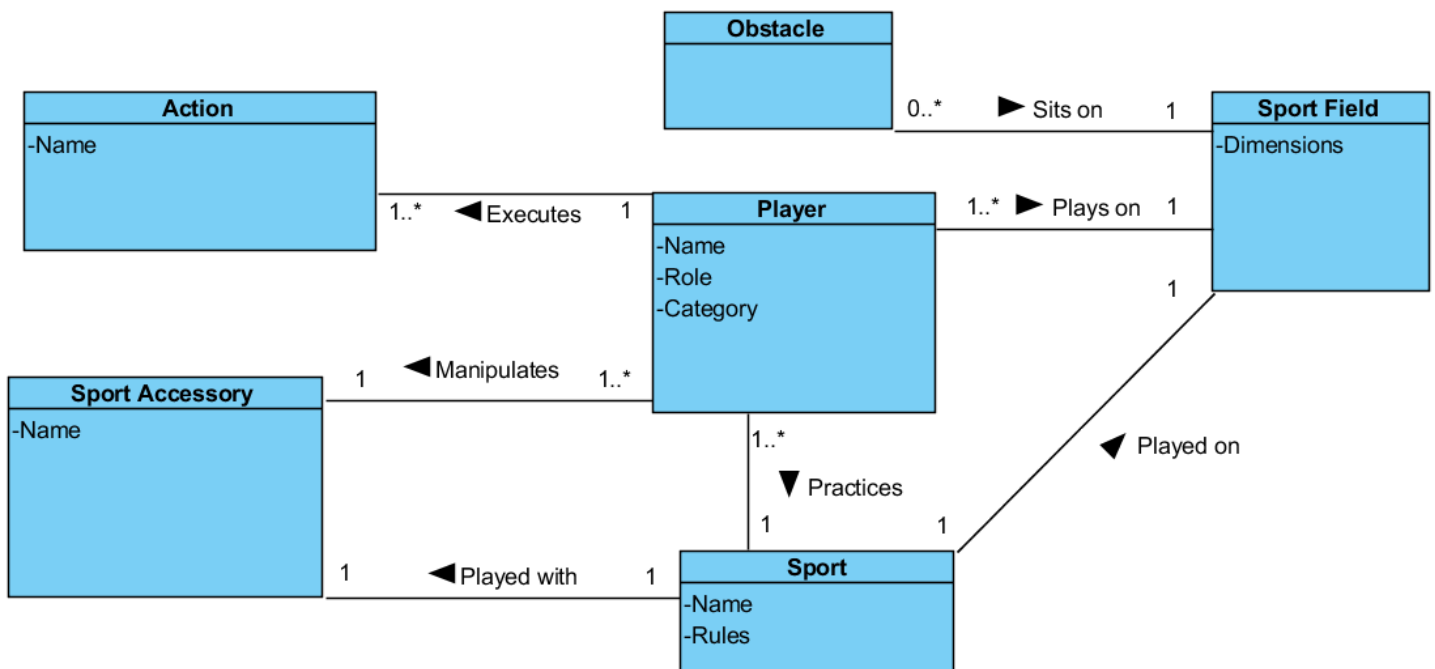
### Détails

Cas d'utilisation :	Avancer ou reculer l'exécution d'un jeu par bond de temps
Système :	VisualLigue
Acteur(s) :	Entraîneur
Parties prenantes et intérêts :	Entraîneur : Veut avancer ou reculer l'exécution d'un jeu par bond de temps prédéfini (ex. : un bon de 5 secondes)
Préconditions :	Il y a un jeu chargé dans le programme en mode « visionnement ».
Garanties en cas de succès :	Le jeu est déplacé dans le temps des déplacements.
Scénario principal :	<ol style="list-style-type: none"><li>1. L'entraîneur clique sur &gt;&gt;  (avancer X sec.) ou  &lt;&lt; (reculer X sec.)</li><li>2. VisualLigue avance ou recule X sec. (Configurable dans les options de l'application) et mets l'exécution sur pause s'il était démarré.</li></ol>
Scénarios alternatifs :	Aucun

### Diagramme de séquences

(Voir l'option dans le diagramme de séquences de « Visionner une partie »)

## #3 Modèle du domaine



#### #4 Glossaire

<i>Terme</i>	<i>Définition</i>	<i>Informations</i>	<i>Alias</i>
<i>Jeu</i>	Art de diriger et de coordonner des actions pour atteindre un objectif.	Les jeux sont appliqués par les élèves lors d'un jeu.	
<i>Position</i>	Place précise occupée dans l'espace par quelqu'un.	Chaque joueur a une position précise sur le jeu (coordonnées).	
<i>Obstacle</i>	Tout objet qui s'interpose, qui se trouve sur le trajet de quelque chose.	Des obstacles pourront être ajoutés sur le jeu tels que des cônes.	
<i>Déplacement</i>	Fait de changer de position.	Les joueurs peuvent être déplacés à des fins de jeu.	
<i>Orientation</i>	Détermination d'un emplacement par rapport aux points cardinaux.	Position vers laquelle se déplace un joueur.	
<i>PNG, JPEG, BMP,</i>	Formats d'images.	Les jeux peuvent	

<i>TIFF, GIF</i>		être exportés en formats d'images, plusieurs formats sont disponibles.
<i>Rôle</i>	Fonction remplie par quelqu'un.	Les joueurs peuvent avoir un rôle différent, par ex. attaquant, défensif, chef d'équipe, etc.