

# Rapport 01 Projet IL

Roux Quentin & Fageot Pierre

October 11, 2015

## 1 Présentation de la composante Interface utilisateur

Cette composante permet à l'utilisateur de modifier les paramètres d'une partie et de voir les résultats via des options. Une de ces options permet de gérer les statistiques des joueurs. L'interface utilisateur sert également de lien avec le moteur physique pour transmettre les commandes utilisées par les utilisateurs. Ce lien est représenté logiquement par un code source.

## 2 Description des interactions

Il existe une faible interaction entre les différentes composantes, faible, mais pas négligeable:

- Sans interface le moteur physique ne peut pas avoir accès aux commandes des utilisateurs.
- Le moteur physique met à jour les données pour l'interface.
- Le moteur graphique envoie les informations nécessaires à l'interface pour dessiner les objets.
- Notre composante transmet au moteur réseau les mises à jour effectuées sur le moteur physique et le moteur réseau à son tour met à jour le serveur de l'utilisateur.
- Les commandes des joueurs sont transmises à l'interface par le moteur réseau.

## 3 Formalisation

Définition des classe générales utilisés par les différentes composantes:

La classe Object sera défini par des coordonnées, dans un tableau, de l'objet (zone de jeu). La classe Player sera défini par les joueurs ainsi que leurs statistiques.

Voici les méthodes que l'on devra implémenter :

- Send.Commande(). Permet d'envoyer les commandes utilisés par un joueur.

- `Get_Commande()`. Permet de récupérer les commande utilisés par les utilisateurs.
- `Get_Data()`. Récupère les mise à jours.
- `Send_Data()`. Envoie les mise à jour.
- `Draw_Object()`. Dessine les objets.
- `Display_Menu()`. Permet l'affichage des fenêtres fixes.
- `Display_Result()`. Affiche les résultats des parties.
- `Display_Stats()`. Affiche les statistiques des joueurs.