



UNIVERSITÉ D'AVIGNON
ET DES PAYS DE VAUCLUSE

C E N T R E
D'ENSEIGNEMENT
ET DE RECHERCHE
EN INFORMATIQUE



L3 Informatique IL

UE Projet de Programmation

Projet IL Rapport01

Nasreddine Kabbouch, Majda El khorji

13 octobre 2015

CERI - LIA
339 chemin des Meinajariès
BP 1228
84911 AVIGNON Cedex 9
France

Tél. +33 (0)4 90 84 35 00
Fax +33 (0)4 90 84 35 01
<http://ceri.univ-avignon.fr>

Table des matières

1	Présentation de la composante Moteur Graphique	3
2	Description des interactions avec les autres composantes	3
3	Formalisation des interactions	3

1 Présentation de la composante Moteur Graphique

Un moteur graphique, c'est un programme qui affiche la version "graphique" des éléments du jeu. Il charge des modèles, joue des animations, applique des textures etc... Tout en coordination avec des éléments tels que des personnages, tout ça, ainsi que leurs événements associés. Le moteur graphique d'un jeu vidéo constitue la couche externe de ce dernier, celle avec laquelle l'utilisateur interagit.

Le moteur graphique est un ensemble de commandes affichées à l'écran de l'ordinateur et permettant de piloter un logiciel sans saisie de lignes de commandes. En bref le moteur graphique gère l'affichage du jeu.

2 Description des interactions avec les autres composantes

Un moteur graphique c'est ce qui affiche les éléments à l'écran. Qui décrit aussi ces trucs comme des éléments et pas une bouillie de n'importe quoi. Ensuite le moteur physique prend ces éléments pour les faire se déplacer et interagir entre eux (rebonds, collisions) comme en vrai.

- les interfaces physiques, celles qui permettent de commander les actions dans le jeu (input).
- les interfaces graphiques, celles qui rendent compte d'éléments du jeu (output).
- L'interface a besoin du moteur graphique pour dessiner ces objets.
- L'interface a besoin du moteur réseau pour recevoir les commandes des joueurs clients, afin de les transmettre au moteur physique.

3 Formalisation des interactions

Les différentes classes qu'on a besoin de les utiliser :

Class Graphique : Présentation graphique du Snake qui est toujours en transactions, qui reçoit les commandes et dessine le résultat en appelant la classe PositionDirection aux utilisateurs.

Class PositionDirection : sert à présenter les points parcourus par le snake. Chaque point/position admet aussi une direction.

les Différentes Méthodes qui seront implémenter :

MoveSnake() : permet d'avoir la position du Snake.

GetData() : nous permet de récupérer les commandes.

DrawSnake() : Dessine les objets.

TrajectSnake() : Dessiner le trajet du Snake.

DetectCollision() : détection des collisions.