

PROGRAMMATION AVANCEE

RAPPORT DE PROJET



MASTER 1 – MIAGE

Monsieur Fabrice HUET

Executive Summary

The project :

The goal of this project is to build a virtual fighting game of the « RobotWar » type.

In a battle arena in 2D, seen from above, robots clash, managed by a relatively basic AI. The behavior and the graphics of robots is decided by plugins. A robot is active if its life has not reached 0 and the winner is the last active robot. Each robot is associated with a quantity of energy and each action consumes a part of it. The energy goes up regularly until it reaches the maximum value.

Le projet :

Le but de ce projet est de construire un jeu de combat virtuel de type « RobotWar ».

Dans une arène de combat en 2D, vue de dessus, des robots s'affrontent, gérés par une IA relativement basique. Le comportement ainsi que le graphisme des robots est décidé par des plugins. Un robot est actif tant que sa vie n'a pas atteint 0 et le gagnant est le dernier robot actif. À chaque robot est associé une quantité d'énergie et chaque action consomme une partie de celle-ci. L'énergie remonte régulièrement tant qu'elle n'a pas atteint la valeur maximale.

Sommaire

PARTIE 1. PRÉSENTATION DES MEMBRES DU GROUPE.....	3
DATAO Chen.....	3
FONTAINE Gaël.....	3
ROMAN Geoffrey	4
PARTIE 2. PROCEDURE A SUIVRE POUR TESTER LE PROJET	4
PARTIE 3. CALENDRIER DES GRANDES ETAPES	4
PARTIE 4. REALISATION DES CINQ POINTS DU PROJET.....	4
Fonctionnalité	5
Mécanisme de gestion de plugins et de chargement dynamique.....	5
Persistance (sauvegarde de l'état des plugins)	5
Modularité et dépendances.....	5
Documentation / gestion du projet	5
PARTIE 5. PROCEDURE POUR CREER UN NOUVEAU PLUGIN	5
PARTIE 6. EXEMPLES DE PLUGINS	5

PARTIE 1. PRÉSENTATION DES MEMBRES DU GROUPE

DATAO Chen

FONTAINE Gaël

Mon travail au sein du groupe et dans le projet en général est divisé en plusieurs parties. Premièrement, la structure et la découpe du projet « maven » en 3 modules distinct (moteur, annotation et plugin). J'ai également pu rédiger 1 plugin d'attaque et 1 plugin de graphisme sur le projet, « *AttaqueSimple* » qui permet à deux Robot de s'affronter et « *NomRobot* » qui place le nom des robots au-dessus de ces derniers en cours de partie.

De plus, mon rôle aura été de créer la classe « Launch » qui permet de lancer une partie et de dérouler un scénario quant à la mort des robots. J'ai également participé au moteur du jeu avec la classe « *FrameWithMenu* » que j'ai modifié plusieurs fois afin d'optimiser le fonctionnement du jeu.

Enfin, la dernière partie concrète aura été de rédiger les tests unitaires de certaines classes permettant de tester les classes centrales du jeu. Pour conclure, le projet m'a permis de mettre en pratique toutes les connaissances accumulées au cours du semestre de M1 MIAGE et de m'améliorer dans le travail d'équipe.

ROMAN Geoffrey

Mon travail durant ce projet a été, de créer un classLoader personnalisé permettant de charger des classes dans un dossier spécifique, ou dans un JAR. J'ai aussi créé une classe Repository, permettant de charger toutes les classes dans le projet, ayant une annotation spécifique à un des trois types de plugins.

De plus, je me suis chargé de faire la classe FrameWithMenu, qui permet d'afficher la fenêtre du jeu avec les différents robots qui se déplacent, attaque... La fenêtre est composée d'un menu permettant soit de sauvegarder/charger une partie, soit d'utiliser un plugin de type graphique.

Je me suis aussi occupé de réaliser la classe ChargementPlugin, qui permet, en passant en paramètre la classe du plugin ainsi que la FrameWithMenu dans laquelle le plugin doit être chargé, d'appeler la méthode d'action du plugin (draw, déplacement, ou attaque)

J'ai aussi fait le système de sauvegarde/chargement de partie, grâce à la partie sérialisation vu en cours.

En ce qui concerne la partie plugin, ma tâche a été de faire le plugin de déplacement simple

PARTIE 2. PROCEDURE A SUIVRE POUR TESTER LE PROJET

PARTIE 3. CALENDRIER DES GRANDES ETAPES

PARTIE 4. REALISATION DES CINQ POINTS DU PROJET

Fonctionnalité

Mécanisme de gestion de plugins et de chargement dynamique

Persistence (sauvegarde de l'état des plugins)

Modularité et dépendances

Documentation / gestion du projet

PARTIE 5. PROCEDURE POUR CREER UN NOUVEAU PLUGIN

PARTIE 6. EXEMPLES DE PLUGINS