

Rapport de projet de programmation

Jeu de la vie

Equipe

Marius Cahagne

Alexis N'gabala

Yann Pickern

Thibault Rolland

François Ngy



Introduction

Ce document a pour but de décrire le déroulement de notre projet : Jeu de la vie.
Il se découpera en deux parties.

Tout d'abord nous parlerons de l'aspect technique, le cahier des charges qui nous a été imposé et comment nous avons agi en conséquence, le fonctionnement de notre projet, ainsi que les difficultés que nous avons pu rencontrer.

Dans un second temps nous évoquerons la gestion de notre projet, la répartition des tâches ainsi que la gestion du temps qui nous a été fourni, et ce qu'il nous a appris d'un point de vue technique comme personnel.

Sommaire

I – Rapport technique

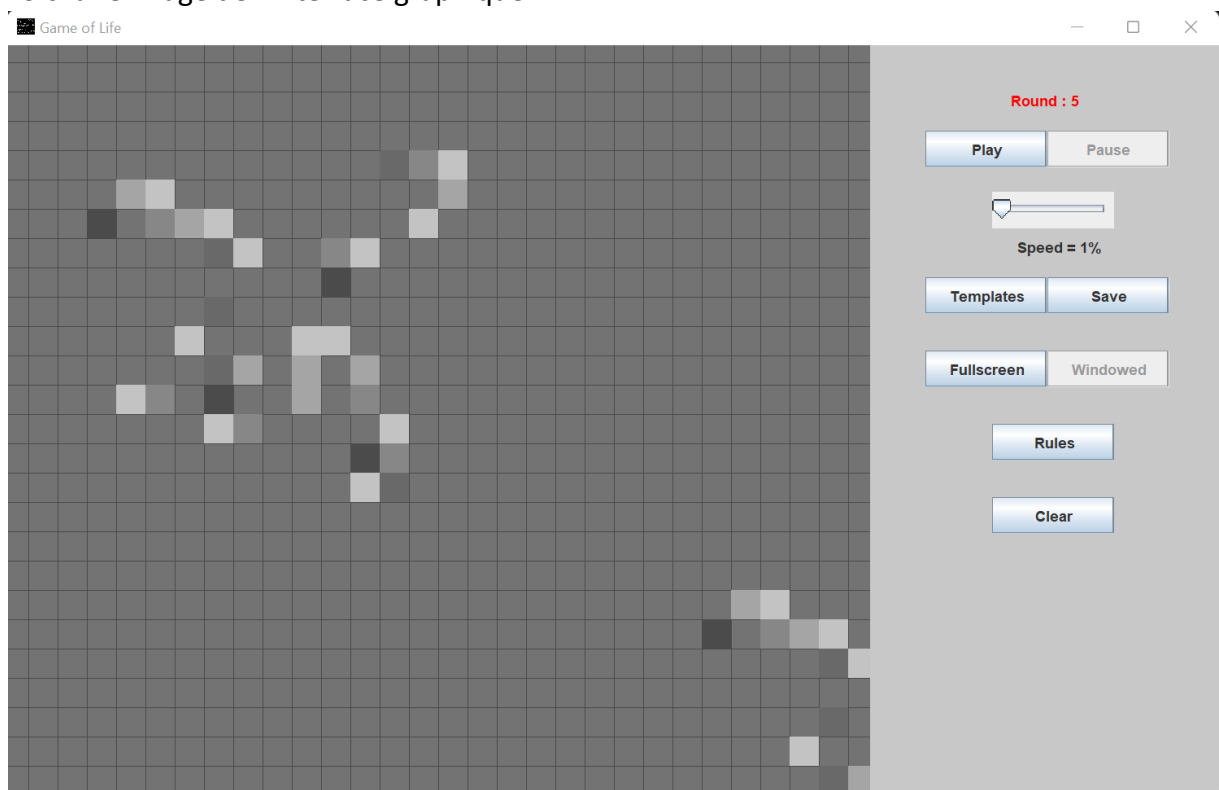
II - Gestion

I – Rapport technique

Ce projet nous a été proposé avec un cahier des charges prédéfini :

- Le plateau doit être infini.
- Toutes les règles du jeu doivent être implémentées et l'utilisateur peut ajouter ses propres règles via une interface. Parmi les règles de base :
 1. Toute cellule vivante qui a moins de deux voisines vivantes meurt
 2. Toute cellule vivante qui a deux ou trois voisines vivantes survit.
 3. Toute cellule qui a plus de trois voisines vivantes meurt.
 4. Toute cellule morte avec exactement trois voisines vivantes est régénérée.
- Le jeu doit pouvoir être joué via l'interface graphique.

Voici une image de l'interface graphique :



Comment avons-nous répondu à ces contraintes ?

- Concernant le plateau infini, nous utilisons une liste qui contient seulement les cellules vivantes. On peut zoomer et dézoomer pour afficher plus ou moins de cases, et on peut se déplacer sur la grille à l'aide de la souris.
- Concernant les règles
 1. Les règles de base du Jeu de la vie sont déjà implémentées sans que l'utilisateur n'ait besoin de faire quoi que ce soit. Néanmoins, un bouton « Rules » a été mis à disposition afin que l'utilisateur puisse modifier ses propres règles.

Ensuite, nous avons établi dans des listes d'entiers le nombre de cellules voisines vivantes nécessaire pour qu'une cellule vivante reste en vie et qu'une cellule morte prenne vie.

- Concernant la vue, le jeu se partage en deux principales parties : un plateau à gauche qui servira de grille de jeu, et un menu à droite.
Ce dernier comporte plusieurs boutons :
 1. Le bouton « Play » qui sert à faire démarrer l'animation du jeu de la vie.
 2. Le bouton « Pause » qui sert à mettre en pause l'animation.
 3. Un curseur de vitesse qui va, selon le choix de l'utilisateur, augmenter ou diminuer la vitesse de l'animation.
 4. Le bouton « Rules » qui ouvre un pop-up où l'utilisateur va pouvoir configurer ses propres règles.
 5. Le bouton « Clear » qui va remettre à zéro la partie en cours.
 6. Les boutons « Fullscreen » et « Windowed » qui vont respectivement passer en plein écran ou en fenêtre réduite le jeu.
 7. Le bouton « Templates » qui va donner quelques exemples de structures de cellules de départ à l'utilisateur.
 8. Le bouton « Save » qui sert à sauvegarder la partie en cours.

Comment notre projet fonctionne-t-il ?

Nous avons découpé notre projet en plusieurs packages :

- Le premier package model : Il regroupe tous les fichiers nécessaires au fonctionnement purement technique des diverses fonctionnalités du jeu. Comme par exemple les classes Cellule, ModelQuadTree, ou encore Regles.
- Un second package view : Il regroupe les fichiers nécessaires à la partie visuelle. Ces derniers sont gérés par le contrôleur.
- Un troisième package controller : C'est lui qui se charge de « relier » les éléments visuels à la partie technique, donc il fait la liaison entre les deux packages ci-dessus.

- Et un dernier package lanceur: il contient seulement le fichier Lanceur qui est le point d'entrée du projet et qui sert à compiler le jeu.

Vous pourrez retrouver le diagramme de classe du projet qui correspond au fichier suivant : **diagrammeClasses.png**. Avec dans l'ordre les différentes classes des packages : lanceur, controller, model, view.

II - Gestion

Nous nous sommes tous occupés d'une partie spécifique du projet.

Certains se sont occupés de l'aspect visuel, d'autres du modèle, du contrôleur, etc...

Nous avons plutôt bien géré notre temps, en effet, chaque semaine avec notre responsable de projet, nous faisons chacun un rapport des tâches effectuées durant la semaine et planifions les tâches à venir.

Cette méthode nous a permis de travailler régulièrement, nous avons eu assez de temps pour corriger certains bug ou encore rédiger ce rapport. Cependant, nous voulions optimiser le jeu en modifiant le système de stockage de données. Nous avons choisi d'implémenter un QuadTree d'après la recommandation de notre responsable de projet pour stocker les cellules vivantes, ce qui nous permet de fortement réduire le temps pour vérifier si oui ou non une cellule est présente dans le modèle.

D'un point de vue technique, ce projet nous a permis de bien maîtriser le travail d'équipe.

D'avantage que le projet du premier semestre qui était en binôme, ce projet de programmation par groupe de 5 nous forçait d'avantage à être organisé pour éviter tout conflit sur Git.