Jérémie Rigert

# Conway's Game of Life

Création du jeux en MVVM C#

Table des matières

[Conway's Game of Life 0](file:///C:\Users\suiss\Desktop\JeremieSoloProject\Trello\DescriptionFeature.docx#_Toc100090821)

[Description du projet 2](#_Toc100090822)

[Les règles du jeux 2](#_Toc100090823)

[Principe de naissance 2](#_Toc100090824)

[Principe de stase 2](#_Toc100090825)

[Principe de mort 2](#_Toc100090826)

[Échéancier 2](#_Toc100090827)

[Description des « features » 3](#_Toc100090828)

[1. Créer l’arborescence du projet 3](#_Toc100090829)

[2. Créer la vue du projet 3](#_Toc100090830)

[3. Créer un Canvas fonctionnel 3](#_Toc100090831)

[4. Ajouter les states d’un carré(mort ou vivant) 3](#_Toc100090832)

[5. Créer la fonctionnalité d’enregistrer le tableau 3](#_Toc100090833)

[6. Créer la fonctionnalité de charger un tableau 3](#_Toc100090834)

[7. Créer la fonctionnalité d’ajouter des formes 3](#_Toc100090835)

[8. Créer la fonctionnalité de faire un Canvas aléatoire 3](#_Toc100090836)

[9. Avoir l’information des carrés adjacents 3](#_Toc100090837)

[10. Faire le binding 3](#_Toc100090838)

[11. Création de la classe VueModele 3](#_Toc100090839)

[12. Ajouter les règles du jeux de la vie 3](#_Toc100090840)

[13. Ajouter la vitesse du jeux 3](#_Toc100090841)

[14. Faire la classe carré(cells) 3](#_Toc100090842)

[15. Besoin d’une classe cellsHelper? 3](#_Toc100090843)

# Description du projet

* Dans le cadre du cours de DÉVELOPPEMENT D'APPLICATIONS (AVANCÉ) nous devions faire soit le jeux de la vie de Conway ou un jeux de Snake.
* Dans mon cas j’ai choisi le jeux de Conway
* Le jeux a été créée en 1970
* C’est un jeux de simulation au sens mathématique, un automate cellulaire

# Les règles du jeux

## Principe de naissance

* Si une case est vide et que trois de ses voisines sont occupées, alors une naissance s’y produit.

## Principe de stase

* Si une case est occupée, la survie n’y est possible que si deux ou trois cases voisines sont occupées

## Principe de mort

* Si une case est entourée de 0 ou 1 voisine occupée, la case est vide à la génération suivante (mort par isolement).
* Si une case est entourée de 3 voisines occupées et plus, la case est vide à la génération suivante (mort par surpopulation).

# Échéancier

28 Avril 2022

# Description des « features »

## Créer l’arborescence du projet

## Créer la vue du projet

## Créer un Canvas fonctionnel

## Ajouter les states d’un carré(mort ou vivant)

## Créer la fonctionnalité d’enregistrer le tableau

## Créer la fonctionnalité de charger un tableau

## Créer la fonctionnalité d’ajouter des formes

## Créer la fonctionnalité de faire un Canvas aléatoire

## Avoir l’information des carrés adjacents

## Faire le binding

## Ajouter les CanExecute(si besoin)

## Création de la classe VueModele

## Ajouter les règles du jeux de la vie

## Ajouter la vitesse du jeux

## Faire la classe carré(cells)

## Besoin d’une classe cellsHelper?