**DECORONAVISEUR**

Guide technique

Avril 2020

Version 1.0

G7

Stacey Cnudde

### Table des matières

[Table des matières 2](#_Toc38468796)

[1. Introduction 2](#_Toc38468797)

[2. Concept et fonctionnalités 2](#_Toc38468798)

[2.1. Modes de jeux 3](#_Toc38468799)

[Mode Normale 3](#_Toc38468800)

[Mode Propagation 3](#_Toc38468801)

[Mode Apocalypse 3](#_Toc38468802)

[3. Diagramme de classe 3](#_Toc38468803)

[3.1. Classes et méthodes 3](#_Toc38468804)

[4. Etude algorithmique 3](#_Toc38468805)

[5. Ressources : 3](#_Toc38468806)

[5.1. Graphisme 3](#_Toc38468807)

[6. Test 4](#_Toc38468808)

[6.1. Politique de test 4](#_Toc38468809)

[Test unitaire 4](#_Toc38468810)

[Test d’intégration 4](#_Toc38468811)

[Test utilisateur 5](#_Toc38468812)

[7. Annexe 5](#_Toc38468813)

[7.1. Codes 5](#_Toc38468814)

[7.2. Tests 5](#_Toc38468815)

# Introduction

Ce guide reprend les spécificités techniques du jeu blablabla

# Concept et fonctionnalités

Jeu de lutte contre le Coronavirus se basant sur le démineur. Il y a actuellement trois modes de jeu. Chacun des modes possède une mécanique différente.

## Inscription

Le joueur doit impérativement entrer son pseudo avant de lancer une partie. Il y est invité par un popup

## Difficulté

Niveau à choisir sous la forme de bouton radio

**Niveau Débutant :** - Lignes : 9 - Colonne : 9 - Mines : 10

**Niveau Intermédiaire :** - Lignes : 16 - Colonne : 16 - Mines : 40

**Niveau Expert : - Lignes :** 16 - Colonne : 30 - Mines : 99

## Options

Nouvelle partie : crée une grille en fonctione des paramètres entré par le joueur

Règle du jeu : Affiche les règles du jeu

Mode de jeu : Permet de choisir le mode de jeu (Normal / propagation / Apocalypse)

Quitter : Quitter le jeu

## Scoring.

Il existe différents types de compteurs ( nombre de mine restante / timer incrémental / nombre de coups joué). Ceux-ci servent de base pour le calcul du score final. Ce score est enregistré dans un fichier texte avec d’autre informations.

Exemple :

|| Pseudo || Mode || score ||

-----------------------------------------------------------------------------------------

||DesmetTePete || Apocalypse || 99999 ||

||Staceylicone85B || Normal || 2 ||

||MalaiseBaleze || Propagation || 777 ||

## Modes de jeux

### Mode Normale

\*\*expliquer le fonctionnement du mode normal

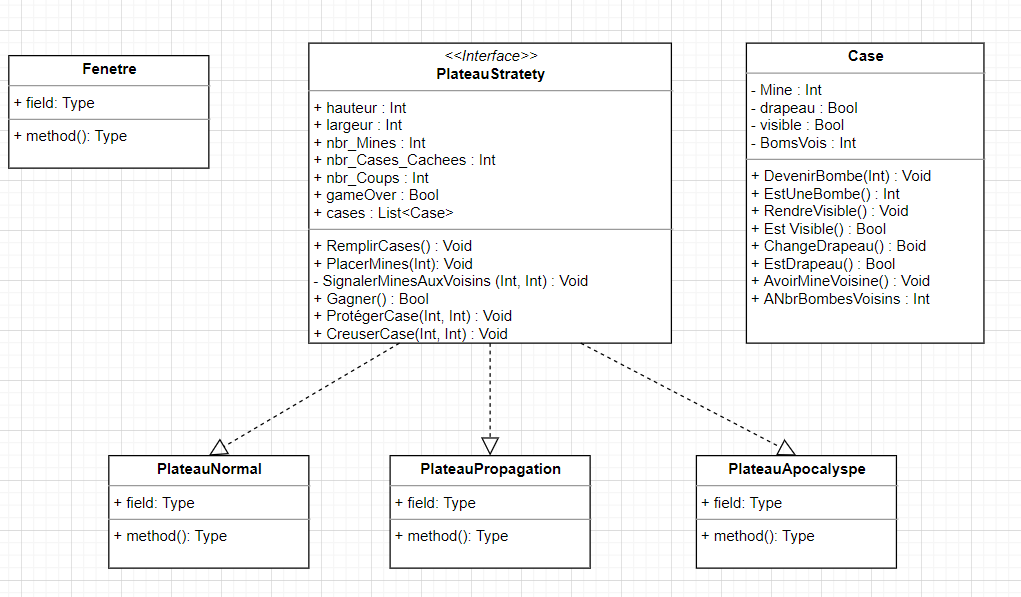
### Mode Propagation

\*\*expliquer le fonctionnement du mode propagation

### Mode Apocalypse

\*\*expliquer le fonctionnement du mode apocalypse

# Diagramme de classe



## Classes et méthodes

\*\* j’ai peur que ce soit redondant avec le diagramme de classe si on liste toutes les classe mais

Certain codes méritent surement de figurer dedans (mise en évidence )

\*\*pour le reste voir annexe

# Etude algorithmique

Etude Algorithmique du démineur

Lancement du jeu :

1. Invitation au joueur à se connecter
2. Demander à l’utilisateur de choisir son mode de jeu et de sa difficulté
3. Génération de la grille
   1. Création du tableau avec les cases
      1. Toutes les cases sont cachées
      2. Toutes les cases n’ont pas de mines
      3. Le nombre de voisins qui ont une mine est à 0
   2. Placement des mines
      1. Choix aléatoire d’une case
      2. Tant que la case choisie est minée, on rechoisit la case
      3. On mine la case choisie
      4. On signale aux voisins que la case est minée
      5. On boucle tant que le nombre de mines n’est pas atteint
      6. Si c’est une partie apocalypse
         1. 20% de Mines Propagation
         2. 20% de Mines Timer+
         3. 20% de Mines Coup+
         4. Le reste de Mines Léthale
4. Affichage de la grille

Partie normale :

1. On demande à l’utilisateur de choisir une case et s’il veut la creuser ou la marquer.
   1. S’il la marque
      1. On place un drapeau s’il n’y en a pas
      2. On retire de drapeau s’il y en a un
   2. S’il la creuse
      1. Si elle est visible ou qu’il y a un drapeau, on ne fait rien
      2. Sinon, on la creuse
         1. On la rend visible
         2. Si elle est minée, la partie est perdue
         3. Sinon, on affiche le nombre de voisins qui ont une mine
         4. Si aucun voisin n’a de mines, on creuse les cases voisines.
      3. On regarde si le nombre de case cachées restantes égale celui des mines
         1. Si oui, la partie est gagnée
         2. Sinon, la partie continue

Partie Propagation

1. On demande à l’utilisateur de choisir une case et s’il veut la creuser ou la marquer.
   1. S’il la marque
      1. On place un drapeau s’il n’y en a pas
      2. On retire de drapeau s’il y en a un
   2. S’il la creuse
      1. Si elle est visible ou qu’il y a un drapeau, on ne fait rien
      2. Sinon, on la creuse
         1. On la rend visible
         2. Si elle est minée, la partie est perdue
         3. Sinon, on affiche le nombre de voisins qui ont une mine
         4. Si aucun voisin n’a de mines, on creuse les cases voisines.
      3. On regarde si le nombre de case cachées restantes égale celui des mines
         1. Si oui, la partie est gagnée
         2. Sinon, la partie continue
   3. Tous les 5 coups
      1. On regarde le nombre de cases cachées restantes moins le nombre de mines et on rajoute 1/20 de mines
         1. Choix aléatoire d’une case
         2. Tant que la case choisie est minée ou visible, on rechoisit la case
         3. On mine la case choisie
         4. On signale aux voisins que la case est minée
         5. On boucle tant que le nombre de mines n’est pas atteint

Partie Apocalypse

1. On demande à l’utilisateur de choisir une case et s’il veut la creuser ou la marquer.
   1. S’il la marque
      1. On place un drapeau s’il n’y en a pas
      2. On retire de drapeau s’il y en a un
   2. S’il la creuse
      1. Si elle est visible ou qu’il y a un drapeau, on ne fait rien
      2. Sinon, on la creuse
         1. On la rend visible
         2. Si elle est minée
            1. Si c’est une mine létale, la partie est perdue
            2. Si c’est une mine Timer+, 10 secondes sont ajoutées au compteur
            3. Si c’est une mine Coup+, 5 coups sont ajoutés au compteur coup
            4. Si c’est une mine Propagation, on rajoute 1 mines de chaque type
         3. Sinon, on affiche le nombre de voisins qui ont une mine, de n’importe quel type
         4. Si aucun voisin n’a de mines, on creuse les cases voisines.
      3. On regarde si le nombre de case cachées restantes égale celui des mines
         1. Si oui, la partie est gagnée
         2. Sinon, la partie continue

# Ressources :

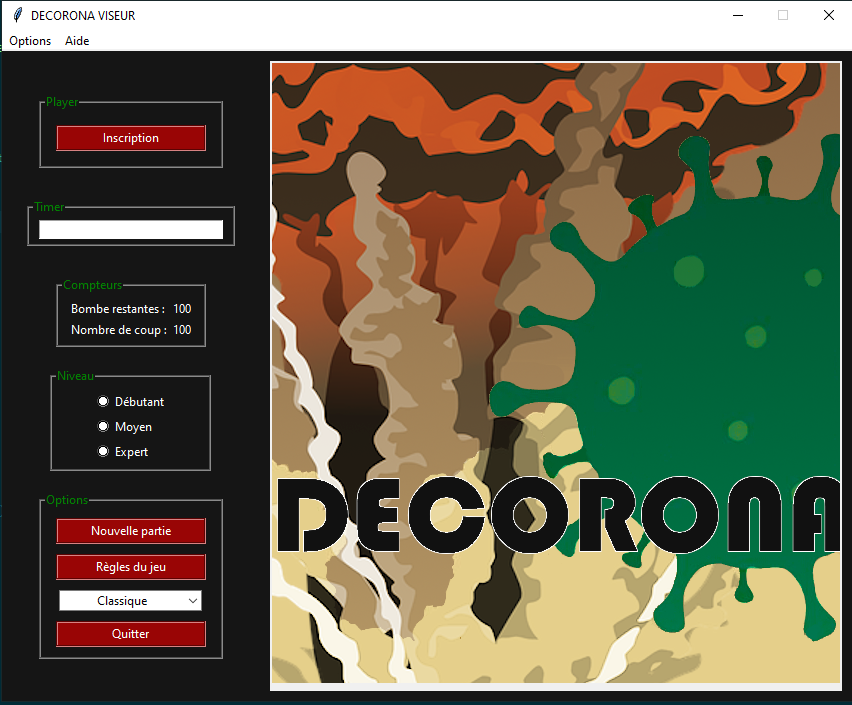
## Logiciels

Python 3.8.2 

\*\* si on utilise des librairies en plus

## Graphisme

### Vue ingame



### Icones du jeu

|  |  |  |  |
| --- | --- | --- | --- |
| virusA.png | virusB.png | virusC.png | virusD.png |
| casePleine.png | caseVide.png | Docteur.png |  |

### Affiches

 **

# Test

## Politique de test

1.[Test]développement module(s).

2.[Test]test module(s).

3.[Test]mise à jour du rapport de bug.

4.[Dev]Correction Bug.

5.[Dev]Mise à jour de correction dans Rapport bug.

6.[Test] suivit et documenter.

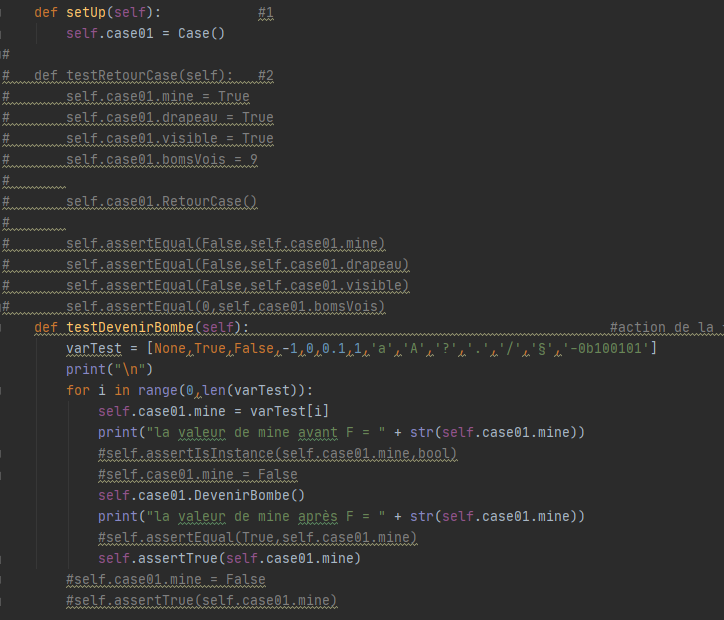
\*\*Cibler les fichiers sources des test ( voir : Testunitaire.xlsx / test\_integration\_composant.xlsx / Test\_Systeme.xlsx)

\*\*Unittest est utilisé pour les test unitaire et intégration.

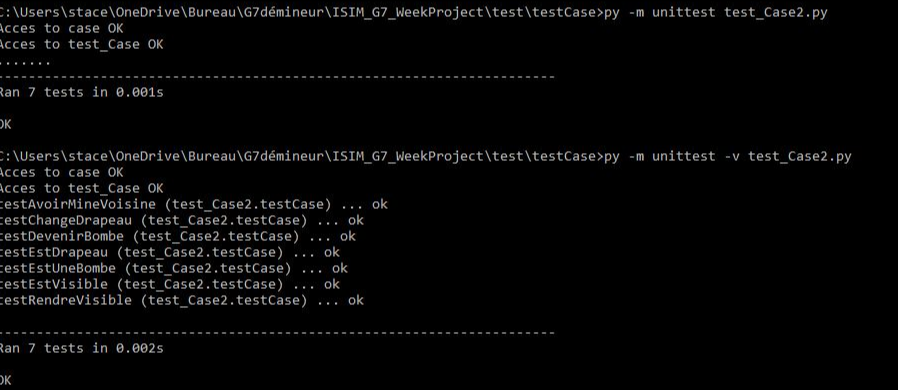
Pour les tests complémentaires nous nous mettons en mode utilisateur lambda pour vérifier les problèmes éventuels

### Test unitaire

\*\*exemple classe de test unitaire (voir : \test\testCase)



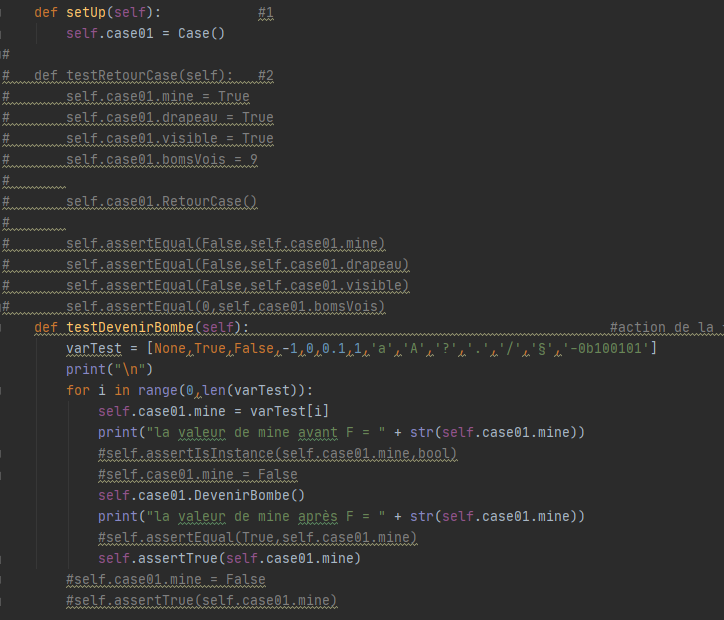
\*\*exemple résultat test unitaire



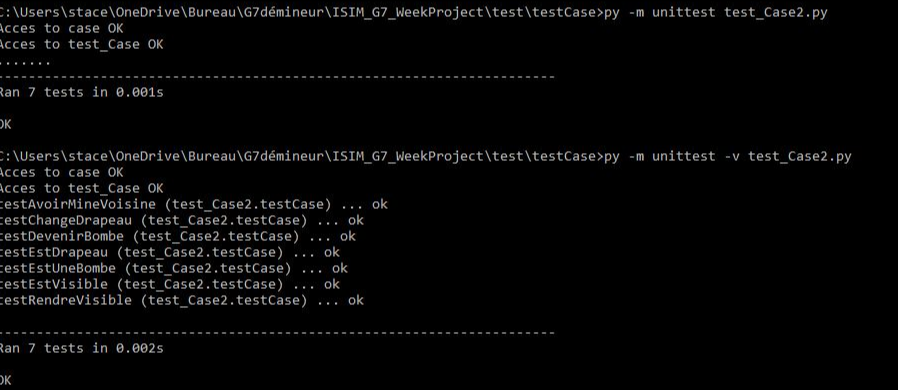
\*\*suite en annexe de document

### Test d’intégration

\*\*exemple classe test d’intégration (voir : \test\testCase)



\*\*exemple resultat test d’intégration



\*\* reste en annexe

### Test utilisateur

\*\* exemple rapport test utilisateur

# Annexe

## Codes

\*\* codes annexe

## Tests

\*\*tests annexe