

**Ministère de l'Économie, du Plan et de la Coopération**

\*\*\*\*\*

**Agence Nationale de la Statistique et de la Démographie (ANSD)**

\*\*\*\*\*

**Ecole Nationale de la Statistique et de l'Analyse Economique Pierre NDIAYE  
(ENSAE)**

**Projet Python**

**Thème**

**Conception d'un jeu de devinette**

**Présenté par :**

**Yatoute MINTOAMA,**

*Élève en ISEP 3*

**Enseignant :**

**Dr Pape A. MBOUP**

*PhD, Informaticien*

**I. Contexte du projet**

**II. Présentation et installation de Kivy**

**III. Lancement de l'application « Riddling game »**

**IV . Présentation de l'application « Riddling game »**

Ce projet s'inscrit dans le cadre du cours de programmation python. Il était question de concevoir un jeu de devinette qui consiste à demander au joueur de deviner un nombre choisi de manière aléatoire sur une plage de nombres entiers selon le niveau du jeu. Le joueur a au plus Une minute et 30 secondes pour deviner le nombre avec possibilité d'essayer plusieurs fois durant cet intervalle de temps. Ainsi Il gagne le jeu s'il trouve le nombre choisi dans cet intervalle de temps. Dans le contraire, il perd le jeu. Le jeu a été développer essentiellement avec la bibliothèque Kivy de Python.

## II. Présentation et Installation de Kivy



**Kivy** est une bibliothèque Python open source d'interface utilisateur graphique qui permet de développer des applications multiplateformes sur Windows, macOS, Android, iOS, Linux et

Raspberry-Pi. La version stable de Kivy peut être installée à partir du terminale avec la commande suivante :

- `python -m pip install "kivy[base]" kivy_examples --no-binary kivy`

Sur Anaconda, Kivy peut être installé avec son gestionnaire de paquets Conda en utilisant :

- `conda install kivy -c conda-forge`

### III. Lancement de l'application « Riddling game »

- Une fois l'installation de « **Kivy** » terminé, pour lancer l'application, il suffit d'exécuter le fichier « **main.py** » dans un éditeur de code python ou dans le terminale.
- Enfin, Il faut s'assurer que le fichier « **main.kv** », les images et les audio à charger soient dans le même dossier que le fichier « **main.py** ».

...

Le jeu de devinette ( Riddling game) est structuré en deux pages dont chacune d'elle s'identifie dans les programmes « **main.py** » et « **main.kv** » à une classe d'objet python héritée de la classe « **Screen** » du module « **uix.screenmanager** » :

- « **HomePage** » pour la page d'accueil ;
- « **GuessNumber** » pour la page du jeu.

Le gestionnaire d'écran « **SreenManager** » permet de transiter d'une page à l'autre.

# « Riddling game » : Page d'accueil

6



- La page d'accueil est structurée en deux parties (colonnes) principale comme illustre la figure précédente.
- Cette structuration s'est effectuée dans le fichier « **main.kv** » à l'aide de l'outil de disposition « **GridLayout** » qui permet d'organiser les widgets d'une page en une structure matricielle.
- Dans la deuxième colonnes, les boutons « **Level i** » permettent d'accéder à la page du jeu selon le niveau. Lorsqu'on clique sur l'un de ces trois boutons, la méthode « **level()** » de la classe « **HomePage** » est d'appelée appelée. Cette méthode utilise une variable globale « **n** » pour fixer le niveau du jeux. Ensuite la propriété « **current** » de la classe gestionnaire d'écran « **SreenManager** » permet d'accéder à la page du jeu



- Structure de la page d'accueil dans le programme main.kv

```
1 <HomePage>:
2     name : 'Home'
3     GridLayout:
4         cols:2
5         size_hint: (1, 1)
6
7         BoxLayout:
8             orientation: "vertical"
9             spacing: 0.005
10            space_x: self.size[0]/3
11            size_hint: (.8, 1)
12            canvas.before:
13                Color:
14                    rgba: (1, 1, 1, 1)
15                Rectangle:
16                    source: 'background0.PNG'
17                    size: root.width, root.height
18                    pos: self.pos
19
20            GridLayout:
21                cols:1
22                rows:4
23                size_hint: (1, 1)
24                canvas.before:
25                    Color:
26                        rgba: (1, 1, 1, 1)
27                    Rectangle:
28                        source: 'background.png'
```

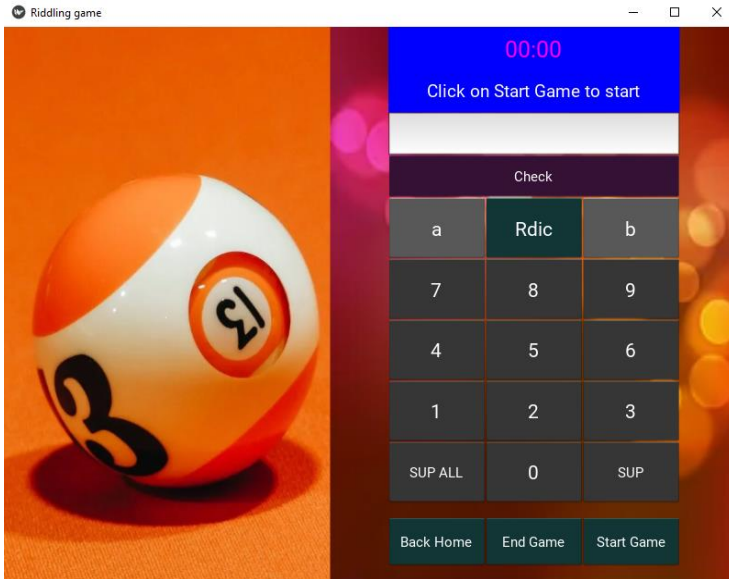
- Structure des boutons d'accès à la page du jeu

```
54      GridLayout:
55          cols: 3
56          rows: 1
57          size_hint: (1, 0.1)
58
59      Button:
60          id:lavel
61          size_hint: (1/3, .1/3)
62          font_size: self.size[0]/5
63          text: "Level 1"
64          on_press:
65              root.level(1)
66              app.root.current='GuessNumber'
67
68      Button:
69          size_hint: (1/3, .1/3)
70          font_size: self.size[0]/5
71          text: "Level 2"
72          on_press:
73              root.level(2)
74              app.root.current='GuessNumber'
75
76      Button:
77          size_hint: (1/3, .1/3)
78          font_size: self.size[0]/5
79          text: "Level 3"
80          on_press:
```

Line 69, Column 43

# « Riddling game » : Interface du jeu

10



- Tout comme la page d'accueil, l'outil de disposition « **GridLayout** » est utilisé pour structurer l'interface du jeu en deux grande parties comme illustre la figure précédente.
- Dans la deuxième partie des widgets de Label, de TextInput et des boutons sont insérés et structuré à l'aide de « **GridLayout** » et de « **BoxLayout** » qui organise les widgets soit de manière verticale, l'un au-dessus de l'autre, soit de manière horizontale, l'un après l'autre.
- Sur l'interface du jeu, excepté le bouton « **Back Home** » qui permet de revenir à la page d'accueil et le bouton « **Start Game** » qui permet de démarrer le jeu, aucun autre bouton ou widget n'est actif jusqu'à ce que le jeu démarre.
- Des sons ont été chargés avec la méthode « **SoundLoader** »

du module « **core.audio** » de Kivy.

- Une fois le jeu démarré, le joueur peut utiliser les touches de son clavier ou le clavier numérique pour entrer un nombre ;
- La touche « **Check** » permet de vérifier si le nombre entré correspond au nombre à deviner ;
- Si le nombre entré correspond au nombre à deviner, le joueur a gagné la partie du jeu ;
- Sinon, le widget label situé au-dessus du widget TextInput indique au joueur si le nombre entré est plus grand ou plus petit au nombre à deviner. Il indique à ce titre un nouvel intervalle  $[a, b]$  sur lequel se trouve le nombre à deviner.

## « Riddling game » : Interface du jeu

- ▶ La borne inférieure de cet intervalle correspond à la touche « **a** » du clavier numérique et la borne supérieure la touche « **b** ».
- ▶ « **Rdic** » est la touche de la recherche dichotomique : elle correspond à chaque instant du jeu au centre près de l'intervalle  $[a, b]$  qui contient toujours le nombre à deviner.
- ▶ Ainsi si le joueur utilise uniquement la touche « **Rdic** » pour jouer le jeu, le nombre d'essai pour trouver le nombre à deviner ne devrait pas dépassé  $\ln(N)$  ( $N=100$  pour le niveau 1, 250 pour le niveau 2 et 1000 pour le niveau 3).
- ▶ Le joueur doit pouvoir trouver le nombre à deviner dans 1 min 30 secondes au plus pour gagner le jeu. Sinon, il le perd.

- ▶ Il peut quitter le jeu à tout moment avec le bouton « **End Game** » ;
- ▶ En cours du jeu, un chrono indique au jours le temps restant pour que le jeu se termine ;
- ▶ La touche « **SUP** » permet respectivement de supprimer le dernier chiffre entré et « **SUP ALL** », tous les chiffres ;
- ▶ Enfin, il faut noter aucune entrée de caractère non numérique n'est autorisé.

**MERCI DE VOTRE  
AIMABLE ATTENTION**