

# Le Pendu

## Répartition des modules

Nous avons choisi de traiter le Pendu en mode graphique.

Nous avons décidé de travailler sur 3 programmes chacun pour gagner en efficacité : Mehdi se charge des 3 premiers, Clément des 3 derniers. A la fin de notre travail en autonomie, nous nous réunirons pour expliquer à l'autre comment on s'y est pris et rectifier les éventuelles erreurs. Bien entendu, dans un premier temps, à chaque ligne de code que nous écrivons, nous demandons au programme d'afficher la fenêtre et le contenu des variables dans la console pour vérifier que tout fonctionne bien.

## Approche globale

C'est le programme `mvc_controller` qui va lancer le jeu. Il appelle tout d'abord le module `root` qui crée une fenêtre. Ensuite, c'est `mvc_model` qui prend la relève en représentant l'univers dans lequel s'inscrit l'application (notamment en récupérant et triant les mots du fichier json) : `mvc_model` est le cerveau du jeu. Enfin, c'est `mvc_view` qui s'exécute en modifiant la partie visible de l'interface graphique (avec les modules `frames`, `display`, `command` et `alphabet`). Ainsi, une fois les programmes `Root`, `Frames`, `Vocabulaire`, `Mot`, `Display` et `Draw` terminés, le jeu devrait parfaitement fonctionner.

## Root

J'étais dans l'impossibilité de réaliser ce programme seulement avec ce qu'on a appris. Je me suis donc renseigné sur les classes et Tkinter grâce à la python docs en ligne.

### **Algorithme :**

Importer le module `tkinter`

Créer la fonction `create_main` qui crée un objet fenêtre graphique des dimensions souhaitées (en fonction de la taille de l'écran) en renvoyant un objet fenêtre graphique de classe `'tkinter.Tk'`.

- Créer un objet fenêtre graphique main\_gui de type tk.Tk
- Créer une variable w qui contient la mesure de la largeur de l'écran
- Créer une variable h qui contient la mesure de la hauteur de l'écran
- Redimensionner la fenêtre main\_gui en fonction de la taille de l'écran (donc en fonction de w et h)
- Rendre la fenêtre main\_gui non redimensionnable
- Ajouter un titre à la fenêtre main\_gui
- Ajouter une icône "icone.ico" à la fenêtre main\_gui
- Afficher la fenêtre main\_gui

Dans le programme principal

- Créer la variable main\_windows qui appelle la fonction create\_main
- Vérifier que main\_windows est bien devenue une fenêtre graphique de type tk.Tk
- Afficher les objets

## Frames

### **Algorithme :**

Importer le module tkinter

Importer les options LEFT, BOTH, RAISED, TOP de Frame et Pack

Créer la fonction create\_frame, prenant main\_gui de classe 'tkinter.Tk' comme argument, qui crée les zones d'affichages en renvoyant la liste des Frames.

- Mesurer les dimensions de la fenêtre

- Créer une variable w qui contient la mesure de la largeur de la fenêtre

- Créer une variable h qui contient la mesure de la hauteur de la fenêtre

- Créer une variable bleu qui référence une frame bleue en relief

- mesurant la largeur et un tiers de la hauteur de la fenêtre

- La placer en haut de la fenêtre

- Créer une variable vert qui référence une frame verte en relief mesurant la moitié de la largeur et trois demi de la hauteur de la fenêtre

- La placer à gauche de la fenêtre

- Créer une variable rouge qui référence une frame rouge en relief

- La placer à droite de la fenêtre

- Créer une variable gris qui référence une frame grise en relief

- La placer à droite de la fenêtre

- Créer une variable violet qui référence une frame violette en relief

La placer à droite de la fenêtre  
Renvoyer la liste des frames

Dans le programme principal  
Créer une fenêtre graphique main\_windows  
La redimensionner  
Créer la variable frames\_list qui appelle la fonction create\_frame avec  
pour argument main\_windows  
Vérifier que frames\_list est bien une liste de frames de type list  
Afficher les objets

## Vocabulaire

Je me suis renseigné sur les méthodes spécifiques aux fichiers json.

### **Algorithme :**

Importer le module json

Créer une fonction open\_json, prenant comme argument "cible" le nom du  
fichier à ouvrir, qui ouvre un fichier json et renvoie une liste de mots.

Ouvrir et fermer le fichier json en mode lecture

Récupérer le contenu du fichier dans une variable rep à l'aide de  
la méthode load

Renvoyer la liste de mots de rep

Dans le programme principal :

Affecter le fichier words.json à une variable fichier\_json

Créer la variable words qui appelle la fonction open\_json avec pour  
argument fichier\_json

Vérifier que words est bien une liste de mots de type list

Afficher la liste de mots

## Mot

## Display

Draw