

“Aidez Mac Gyver à s'échapper !”

Projet n°3

Zoé Belleton • 28/06/2018





Le projet

une application “standalone”



Documentation de module



Algorithmique / Analyse logicielle



Conceptualisation / Structure (UML)



Environnement de développement (IDE)



Distribution du script (open source)

(pyGame) (pyCharm / pyLint) (cx_Freeze)

Plateformes

Linux , Windows ,
Windows CE, BeOS,
MacOS, Mac OS X,
FreeBSD, NetBSD,
OpenBSD, BSD/OS,
Solaris, IRIX, QNX.

Le code contient un
support pour
AmigaOS,
Dreamcast, Atari,
AIX, OSF/Tru64, RISC
OS, Symbian OS, and
OS/2, nokia, game
consoles like gp2x,
the One Laptop Per
Child (OLPC), and the
Orange Pi.

Pygame (www.pygame.org)

Set de modules dédié à la conception d'applications GUI

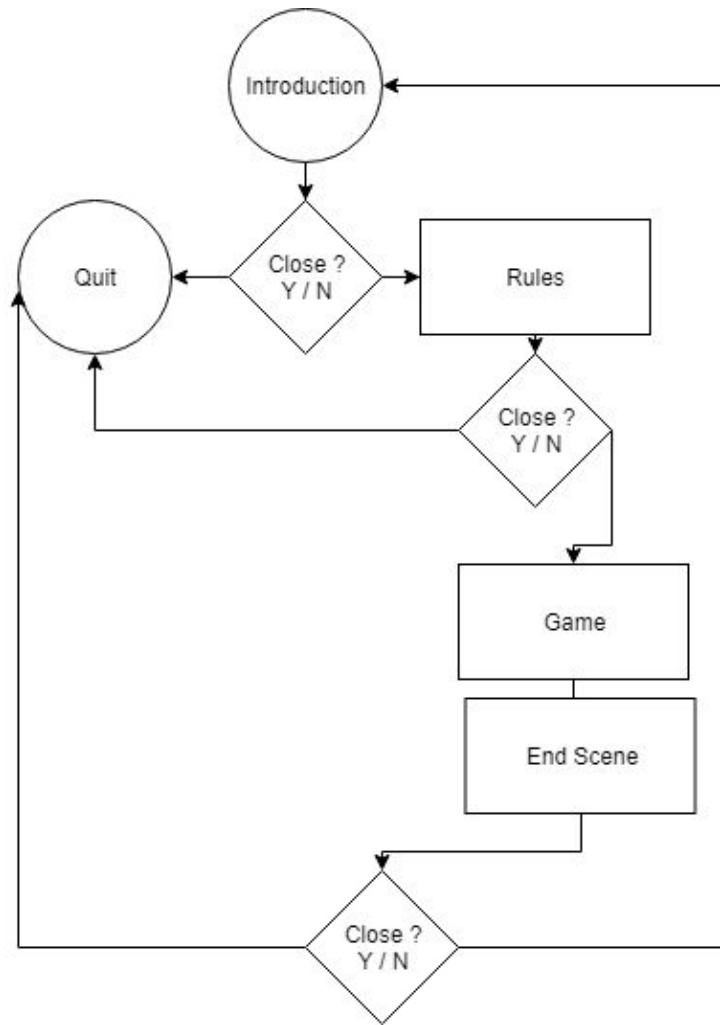


SDL*



- Libre & open-source
(sous licence GNU LGPL)
- Repose sur la SDL | Pas OpenGL
(conçu pour remplacer pySDL)
- Utilise des langages qui multiplie sa
vitesse d'exécution
(C (10 à 20 *) et Assembleur (~ 100*))
- Cross-platform





Game loop

Re démarrage facilité



Un seul niveau (cf : consigne)



Une variable “status”



La logique du niveau dans une classe dédiée



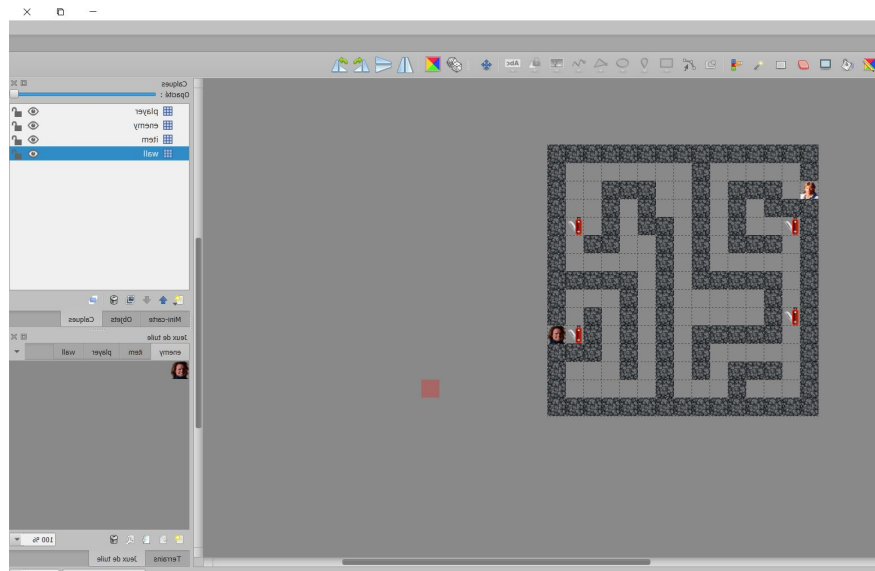
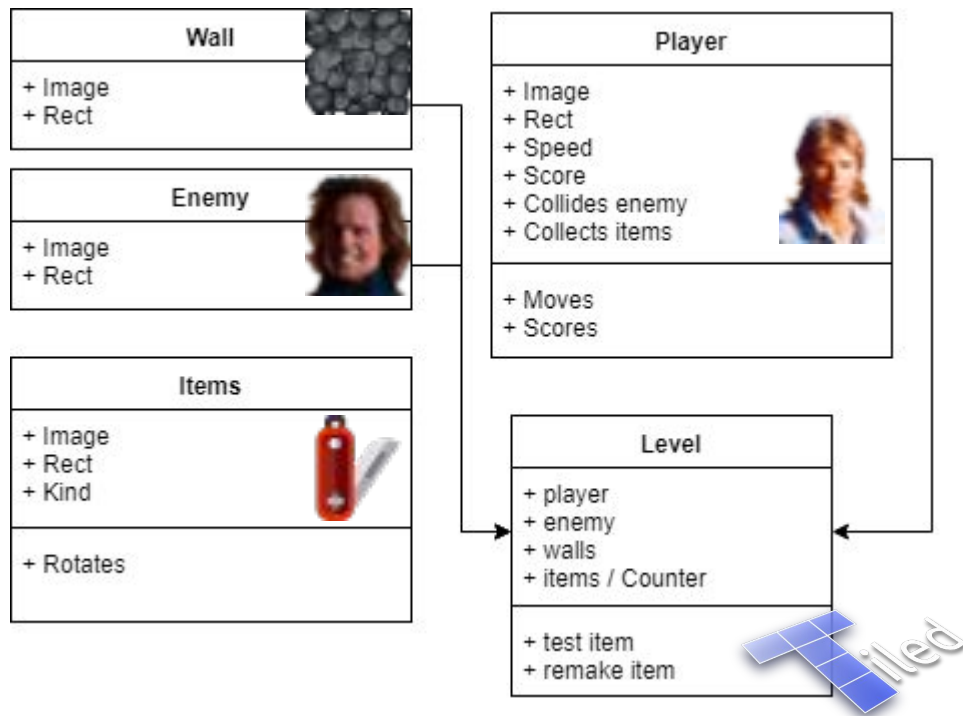
“intro” / “rules” / “game” / “end”



Diffusion du score ! (score.txt)

Level design

Une classe pour chaque élément du niveau



Tiled / PyTMX

(chargeur de carte)

[illegible]

```
for row in range(15):
    for col in range(15):
        # Walls in tmx file
        wall = level.get_tile_image(row, col, wall_tiles)
        if wall is not None:
            self.wall = Wall((row, col))
            self.walls.append(self.wall)
```



```
def walking_in_maze(self, keys):  
    """  
    Player is evolving in the maze  
    :param keys: Pressed keys  
    """  
  
    # Check if the game is paused  
    if not self.pause:  
  
        # up  
        if keys[K_UP]:  
  
            # Check the player position before moves  
            if self.player.rect.y > 0:  
                self.player.move_up()  
  
            # Prevent player colliding walls  
            for self.wall in self.walls:  
                if self.player.rect.colliderect(self.wall.rect):  
                    # Going back  
                    self.player.move_down()
```

Défi

Déplacement dans le labyrinthe

- Pause ?
- Direction ?
- Obstacle ?
- Résultat !

Démonstration



Pistes d'améliorations

1. Multi level

(Branche Master du dépôt Git)

2. Inventaire

(Lister les points au lieu de les compter)

3. Vue à la première personne

(Afficher dans un deuxième écran ce qui est visible par le joueur)
