

Dossier projet ISN

La Bataille Navale



Groupe :

Savary-Derelle Victor.....Terminale S
Crépin Stéphane.....Terminale S
Delplanque Thibaut.....Terminale S

Introduction.....	3
 I. Présentation du projet	
1. L'origine.....	3
2. Le but.....	3
 II. Démarche collaborative	
1. Choix des outils.....	4
2. Répartition des tâches.....	4
 III. Réalisation du projet	
1. Présentation de mes fonctions.....	5
2. Difficultés rencontrées.....	7
 IV. Enjeux du projet	
1. Plan juridique.....	8
2. plan sociétal.....	8
 V. Bilan et perspectives	
1. Perspectives pour améliorer le projet.....	9
2. Ce que ce projet m'a apporté.....	9
 VI. Annexes	
1. Sources.....	10
2. Codes commentés.....	11

Introduction :

Il y a quelques mois nous avons dû choisir un projet à réaliser en groupe. J'ai choisis de me mettre en groupe avec Thibaut Delplanque et Stéphane Crépin car je les connais bien et que je savais qu'ils allaient travailler sérieusement. Nous nous sommes ensuite concerté pour choisir un projet à réaliser.

I. Présentation du projet :

1. L'origine

Nous avons donc choisis comme projet, la Bataille Navale, cela nous a intéressé notamment car c'est un jeu auquel presque tout le monde a déjà joué ou au moins connaît, et auquel nous avons tous beaucoup joué étant plus petit avec nos amis.

2. Le but

Notre but était de réaliser une bataille navale fonctionnelle avec la possibilité de jouer seul ou en réseaux. Avec une interface pratique et quelques options si nécessaire.

II. Démarche collaborative :

1. Choix des outils

Avant de commencer à programmer nous avons choisi d'utiliser python et notamment la bibliothèque Tkinter et PIL pour réaliser ce projet. Nous avons choisi python et Tkinter car nous avons souvent utilisé ces deux outils durant l'année en ISN.

2. Répartition des tâches

Nous avons réparti les tâches en fonctions des compétences de chacun. Etant le seul n'ayant jamais fait d'interface j'ai donc été chargé de réaliser le menu principal de la bataille navale. L'interface devait permettre de lancer le jeu final composer des plateaux et des différentes fonctions de mes partenaires.

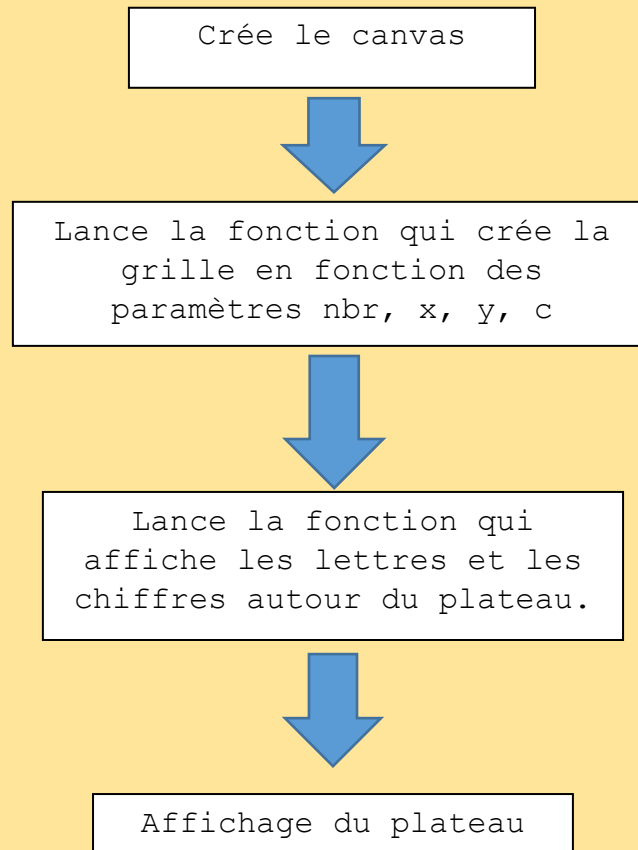
	Stéphane	Thibaut	Victor
Tâches	<ul style="list-style-type: none">-affichage et positionnement des bateaux-IA pour jouer en mode un joueur-Gestion du son	<ul style="list-style-type: none">-Gestion du plateau numérique-Design et placement aléatoire des bateaux-Interface en jeu-mise en réseau	<ul style="list-style-type: none">-création du plateau avec affichages des coordonnées autour-création de l'interface et de toutes ses options-déroulement du jeu
Outils	<ul style="list-style-type: none">-Tkinter-PIL-Pygame	<ul style="list-style-type: none">-paint.net-PIL-Tkinter	<ul style="list-style-type: none">-Tkinter

III. Réalisation du projet :

1. Présentation de mes fonctions

- Le plateau :

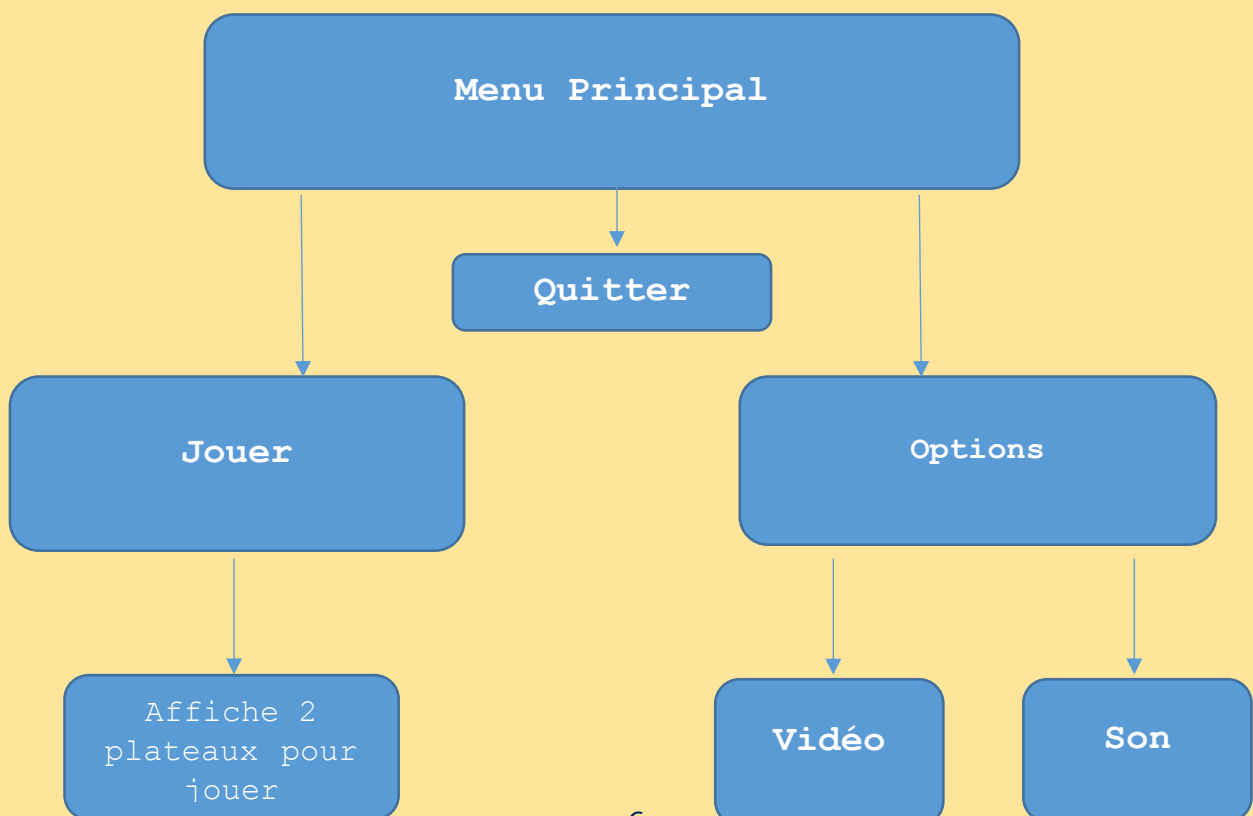
Pour créer le plateau j'ai tout d'abord créé une classe où sont définis les paramètres du canvas ainsi que les variables, **nbr** qui définit le nombre de lignes et de colonnes, **x**, **y** les coordonnées des cases et **c** qui correspond à la taille des cases en pixel.



- L'interface :

Mon but était de créer une interface graphique efficace, pas trop chargée en boutons ou en texte, qui permet de lancer le jeu et de régler les paramètres facilement.

J'ai donc commencé par créer une fenêtre que j'ai intitulée « Bataille Navale ». Dans cette fenêtre j'ai créé un canvas qui me servira de fond. Une fonction crée dans ce canvas des frames qui sont en quelque sorte le squelette de l'interface. Une autre fonction crée dans ces frames des boutons permettant de lancer le jeu, d'ouvrir le menu des options, ou de quitter le jeu. Chacun de ces boutons lance des fonctions différentes. Le bouton jeu, par exemple, supprime la fenêtre existante et recrée une nouvelle fenêtre dans laquelle il lance le programme final qui permet de jouer à la bataille navale. Le bouton options, lui, supprime et recrée un canvas et affiche de nouvelles frames contenant les boutons vidéo et son. J'ai aussi créé un menubar qui permet à l'utilisateur de pouvoir revenir au menu, quitter le jeu ou de lire les règles du jeu depuis n'importe quelle fenêtre de l'interface. Une fois l'interface relativement bien structurée j'ai affiché une image de fond sur le canvas et j'ai coloré les boutons et les textes pour que l'interface soit plus agréable à utiliser.



2. Difficultés rencontrées

- Difficultés du plateau :

Réaliser le plateau ne m'a pas poser énormément de problème excepter au moment de la mise en commun j'ai dû recommencer mes fonctions avec les variables nbr, x, y et c.

- Difficultés de l'interface :

Pour l'interface j'ai eu de nombreux problèmes d'interaction entre les différentes fenêtres que je créé. Certaine frames ce créé dans les mauvaises fenêtres. J'ai eu aussi des problèmes pour positionner l'image de fond car j'avais tout d'abord crée une frame1, au lieu du canvas, et même après de nombreuses recherches je n'ai pas réussi à mettre d'image dans une frame. J'ai donc dû créer un canvas et ensuite déplacer les frames de la frame1 au canvas.

Pour créer le menubar j'ai eu quelques difficultés pour l'afficher sur chaque fenêtre et ensuite pour le rendre fonctionnelle car les boutons ne fonctionnait pas ou ne s'affichait pas.

IV. Enjeux du projet

1. Plan juridique

Sur un plan juridique notre projet peut être soumis, si monétiser, à des droits d'auteurs. Il peut aussi y avoir des litiges car nous sommes trois à posséder les droits de ce programme si l'un de nous désire le monétiser et les autres refusent cela peut poser problème.

2. Plan sociétal

Ce jeu permet de divertir les gens en leur faisant oublier leurs quotidiens le temps d'une partie. Or dans notre société le divertissement est très important, notamment à cause des guerres et des problèmes de société qui prennent de plus en plus d'importance. Cela peut permettre de rencontrer de nouvelles personnes et de se faire de nouvelles relations.

V. Bilan et perspectives

1. Perspectives pour améliorer le projet

Notre programme est fonctionnel mais pas complet, en effet nous pouvons encore y ajouter plusieurs améliorations. Notamment au niveau des options. Car notre programme ne permet pas, en l'état, à l'utilisateur de modifier les paramètres car pour les options vidéo il n'y a pas encore de modification possible à partir des options. Et pour le son on ne peut pas régler le volume ou choisir entre différentes musiques. De plus la possibilité de jouer en réseau n'est pas disponible car c'est une notion très difficile à appréhender et encore plus à mettre en œuvre. Nous pourrions améliorer l'aspect graphique avec plus d'options vidéo et en améliorant le design des bateaux et du plateau, améliorer la jouabilité avec la possibilité de jouer en réseau. Nous pourrions aussi créer un menu pop-up qui s'afficherait à la pression d'une touche, comme échappe, ce qui serait plus ergonomique et plus beau que le menubar existant.

2. Ce que ce projet m'a apporté

En réalisant ce projet j'ai appris à mieux travailler en équipe. Je pense bien maîtriser la bibliothèque tkinter et le fonctionnement et la structure d'une interface graphique. J'ai aussi appris à me servir des classes, outils indispensables pour des projets de tailles importantes. J'ai aussi découvert les menubar qui peuvent se montrer très utiles.

VI. Annexes

1. Sources :

<https://openclassrooms.com/>

<http://apprendre-python.com/page-tkinter-interface-graphique-python-tutoriel>

http://fsincere.free.fr/isn/python/cours_python_tkinter.php

<http://mcclnews.free.fr/python/pygktutfr/ch-MenuWidget.html>

<http://zetcode.com/gui/tkinter/menustoolbars/>

<https://openclassrooms.com/forum/sujet/tkinter-fermeture-de-fenetre-66319>

<http://infohost.nmt.edu/tcc/help/pubs/tkinter/web/relief.html>

http://www.tutorialspoint.com/python/tk_relief.htm

http://www.tutorialspoint.com/python/tk_cursors.htm

<http://infohost.nmt.edu/tcc/help/pubs/tkinter/web/cursors.html>

http://sebsauvage.net/python/gui/index_fr.html#class

2. Codes commentés :

Plateau :

```
from tkinter import *

class platotest:

    """Cette classe défini la taille des cases et le plateau
    est ensuite créer en fonction de la taille des cases.

    Ensuite affiche les lettres et les chiffres autour du
    plateau. """

    def fourretout(self,can,nbr,x,y,c):

        """fonction principale qui paramètre le plateau"""

        self.can=can

        self.grille(10,50,50,50)

        self.can.bind('<ButtonRelease-1>', self.souris)

    def blabla(self,nbr,x,y,c):

        """écrit les lettres et les chiffres autour des 2
        grilles"""

        lettre='ABCDEFGHIJK'

        for loop in range (nbr):

            self.can.create_text(x+c/2,y+c*loop+c*(1.5),text=loop+1,font=(
            'Purisa'))

            self.can.create_text(x+c*loop+c*1.5,y+c*0.5,text=lettre[loop],
            font=('Purisa'))

    def grille(self,nbr,x,y,c):
```

```

        """crée les cases en fonction des parametre de la
grille"""

        for l in range(nbr):
            self.can.create_rectangle(x+l*c,y,x+c+l*c,y+c)
            for k in range(nbr):

self.can.create_rectangle(x+l*c,y+k*c,x+c+l*c,y+c+k*c)
                self.blabla(nbr,x-c,y-c,c)

        def souris(self,event):

            """change la couleur des cases lorsque l'on click
dessus"""

            click =
self.can.find_closest(self.can.canvasx(event.x),
self.can.canvasy(event.y))

            if self.can.itemcget(click, 'fill'):
                self.can.itemconfigure(click, fill='green')

root=Tk()
can = Canvas(root, width =8000, height=8000)
can.pack()
objet = platotest()
objet.fourretout(can,10,50,50,70)
root.mainloop()

```

Interface :

```

from tkinter import *
from final import*

#####
#####

#Menu Principale#

#####
#####

```

```

class interface:

    """cette classe gère le menu principale contenant les
options de son et de
video et permet de lancer le jeu"""

    def fenprincip(self):
        """crée la fenetre principale de l'interface"""
        self.fen=Tk()
        self.fen.title("Bataille Navale")
        self.can= Canvas(self.fen, width=0, height=0)
        self.fond = PhotoImage(file="testimage2.gif")
        self.can.create_image(500, 300, image=self.fond)
        self.can.pack(padx=0, pady=0)
        self.frame()
        self.menu()
        self.Menubarre()
        self.fen.mainloop()

    def frame(self):
        """définit les frames,le squelette, de l'interface"""
        #self.frame1=Frame(self.fen,bg='#682626')
        #self.frame1.pack(side=LEFT, padx=10, pady=10)

        self.frame2=Frame(self.can)
        self.frame2.pack(side=BOTTOM,padx=300,pady=100)

        self.frame3=Frame(self.can,bg='purple')
        self.frame3.pack(side=TOP,padx=0,pady=50)

    def menu(self):
        """crée les boutons permettant l'accès aux différentes
fonctionnalités

```

```

        du jeu et le titre dans les frames"""

        self.b=Button(self.frame2, text ='Jouer', height=1
,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30
bold",relief=GROOVE,
cursor="boat",command=self.play).pack(padx=0, pady=0)

        self.b1=Button(self.frame2, text ='Multijouer',
height=1 ,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant
30 bold",relief=GROOVE, cursor="boat").pack(padx=0,pady=0)

        self.b2=Button(self.frame2, text ='Options',
height=1,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30
bold",relief=GROOVE,
cursor="boat",command=self.option).pack(padx=0,pady=0)

        self.b3=Button(self.frame2, text ='Quitter', height=1
,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30
bold",relief=GROOVE,
cursor="boat",command=self.fen.destroy).pack(padx=0,pady=0)

        self.label=Label(self.frame3,text="Bataille
Navale!",bg='purple',fg='pink',font="Elephant 80
bold",relief=RAISED).pack(side=LEFT,padx=5,pady=5)

    def play(self):

        """détruit le canvas contenant le menu et affiche le
plateau pour jouer"""

        self.can.destroy()

        test=jeu(self.fen)

        test.mainloop()

        self.can=Canvas(self.fen,width=1440,height=900,bg
='#56739A')

        self.can.pack()

        self.plat_1=plato(self.can,62,180,225,10)

#####
#####

#OPTIONS#

```

```
#####  
#####
```

```
    def option(self):  
        """détruit le canvas contenant le menu et crée de  
nouvelles frames ou  
sont créer de nouveaux boutons"""  
        self.can.destroy()  
        self.can= Canvas(self.fen, width=0, height=0)  
        self.fond = PhotoImage(file="testimage2.gif")  
        self.can.create_image(500, 300, image=self.fond)  
        self.can.pack(padx=0, pady=0)  
        self.frames()  
        self.button()  
        self.Menubarre()  
        self.fen.mainloop()  
  
    def frames(self):  
        """créer les frames du menu option"""  
        self.Frame1=Frame(self.can)  
        self.Frame1.pack(side=LEFT, padx=300, pady=225)  
  
        self.Frame2=Frame(self.Frame1)  
        self.Frame2.pack(side=BOTTOM, padx=0, pady=0)  
  
    def button(self):  
        """créer les différent boutons des options"""  
        self.B1=Button(self.Frame2, text ='Son', height=1  
,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30  
bold",relief=GROOVE,  
cursor="boat",command=self.son).pack(padx=0, pady=0)  
  
        self.B2=Button(self.Frame2, text ='Video',  
height=1,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30  
bold",relief=GROOVE,  
cursor="boat",command=self.video).pack(padx=0,pady=0)
```

```

        self.B3=Button(self.Frame2, text ='Menu', height=1
,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30
bold",relief=GROOVE,
cursor="boat",command=self.remenu).pack(padx=0,pady=0)

def remenu(self):

    """permet de revenir au menu,en détruisant la fenetre
déjà existante et
    en recréant la fenetre du menu"""
    self.fen.destroy()
    self.fen=Tk()
    self.fen.title("Bataille Navale")
    self.can= Canvas(self.fen, width=0, height=0)
    self.fond = PhotoImage(file="testimage2.gif")
    self.can.create_image(500,300, image=self.fond)
    self.can.pack(padx=0, pady=0)
    self.frame()
    self.menu()
    self.Menubarre()
    self.fen.mainloop()

#####
#####

#SON#

#####
#####

def son(self):

    """crée une nouvelle fenetre pour régler le son"""
    self.fen2=Tk()
    self.fen2.title("Son")
    self.can2= Canvas(self.fen2, width=0, height=0)
    #self.fond = PhotoImage(file="testimage2.gif")
    #self.can2.create_image(500,300, image=self.fond)

```



```

        self.can2.pack(padx=10, pady=10)

        self.frameson()

        self.boutons()

        self.fen2.mainloop()

def frameson(self):
    """crée les frames de la fenetre son"""
    self.frameson=Frame(self.can2)
    self.frameson.pack(side=BOTTOM,padx=100,pady=0)

    self.frameson1=Frame(self.can2)
    self.frameson1.pack(side=TOP,padx=50,pady=50)

def boutons(self):
    """crée les boutons pour mettre ou non du son et pour
    revenir sur les options"""
    self.BS=Button(self.frameson1, text ='OUI', height=1
, width=15, bg='#799ACC', fg='#7B6C0D', font="Elephant 30
bold", relief=GROOVE, cursor="boat").pack(padx=0, pady=0)

    self.BS1=Button(self.frameson1, text ='NON',
height=1, width=15, bg='#799ACC', fg='#7B6C0D', font="Elephant 30
bold", relief=GROOVE, cursor="boat").pack(padx=0, pady=0)

    self.BSR=Button(self.frameson1, text ='RETOUR',
height=1, width=15, bg='#799ACC', fg='#7B6C0D', font="Elephant 30
bold", relief=GROOVE,
cursor="boat", command=self.fen2.destroy).pack(padx=0, pady=0)

#####
#####

#Video#

#####
#####

def video(self):

```

```

        """crée une nouvelle fenetre pour régler les options
video"""

        self.fen3=Tk()
        self.fen3.title("Video")
        self.can3= Canvas(self.fen3, width=300, height=225)
        #self.fond = PhotoImage(file="testimage2.gif")
        #self.can3.create_image(500,300, image=self.fond)
        self.can3.pack(padx=10, pady=10)
        self.framevideo()
        self.boutonV()
        self.fen3.mainloop()

def framevideo(self):
    """crée les frames de la fenetre video"""
    self.framevideo=Frame(self.can3)
    self.framevideo.pack(side=BOTTOM,padx=100,pady=0)

    self.framevideo1=Frame(self.can3)
    self.framevideo1.pack(side=TOP,padx=50,pady=50)

def boutonV(self):
    """crée les boutons pour modifier l'interface et pour
revenir sur les options"""

    self.BV=Button(self.framevideo1, text ='Interface
Surprise', height=1 ,width=15,
bg='#799ACC',fg='#7B6C0D',font="Elephant 30
bold",relief=GROOVE, cursor="boat").pack(padx=0, pady=0)

    self.BV1=Button(self.framevideo1, text ='classique',
height=1,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30
bold",relief=GROOVE, cursor="boat").pack(padx=0,pady=0)

    self.BVR=Button(self.framevideo1, text ='RETOUR',
height=1,width=15, bg='#799ACC',fg='#7B6C0D',font="Elephant 30

```

```

bold",relief=GROOVE,
cursor="boat",command=self.fen3.destroy).pack(padx=0,pady=0)

def randominter(self):

colorB=['#00FF37','#FF009E','#0077FF','#FFEF00','#6F00FF','#FF
7C00']

colorF=['#4000FF','#00F3FF','#0E721D','#7D0101','#651078','#0D
6E5E']

imagebg=[testimage2.gif]

#####
#####

#Menubar#

#####
#####

def Menubarre(self):

    """crée u menubar qui permet de quitter,de revenir au
menu ou de voir

    les règles du jeu depuis n'importe quelle fenetre """

    self.menubar = Menu(self.fen)

    self.menu1 = Menu(self.menubar, tearoff=0)

    self.menu1.add_command(label="Menu",
command=self.remenu)

    self.menu1.add_command(label="Quitter",
command=self.fen.destroy)

    self.menu1.add_separator()

    self.menubar.add_cascade(label="Menu",
menu=self.menu1)

    self.fen.config(menu=self.menubar)


    self.menu2 = Menu(self.menubar, tearoff=0)

    self.menu2.add_command(label="Règles",
command=self.regles)

```

```

        self.menu1.add_separator()

        self.menubar.add_cascade(label="Aide",
menu=self.menu2)

def regles(self):
    """crée une fenetre pour afficher les règles du jeu"""
    self.fen2=Tk()
    self.fen2.title("Règles")
    self.can2= Canvas(self.fen2, width=0, height=0)
    self.can2.pack(padx=0, pady=0)
    self.Labels()
    self.fen2.mainloop()

def Labels(self):
    """crée le label contenant les règles et un bouton
permettant de revenir au jeu"""

    self.label=Label(self.can2,text="1-placer vos bateaux
de manière stratégique sur votre plateau\n\n2-Jouer chacun
votre tour en cliquant sur une case du plateau adverse pour
tirer",bg='Black',fg='White',font="Arial
20",relief=RAISED).pack(side=LEFT,padx=50,pady=50)

    self.B6=Button(self.can2, text ='Retour au jeu',
height=1 ,width=10, bg='#799ACC',fg='#7B6C0D',font="Elephant
30",relief=GROOVE, cursor="boat",command=self.fen2.destroy)

    self.B6.pack(side=LEFT,padx=10,pady=10)

#####
#####

#####
#####

class plato:
    """initialise le plateau et la taille des cases"""

```

```

def __init__(self, can, c, x, y, nbr):
    self.c=c
    self.can=can
    self.grille(can, nbr, x, y, c)

def grille(self, can, nbr, x, y, c):
    """crée les cases en fonction des parametres de la
    grille"""
    for l in range(nbr):

can.create_rectangle(x+l*c, y, x+c+l*c, y+c, fill='blue')
        for k in range(nbr):

can.create_rectangle(x+l*c, y+k*c, x+c+l*c, y+c+k*c, fill='#56739A
')

objet = interface()
objet.fenprincip()

```
