

Compte rendu Interface graphique **Visionneuse d'Images**

L'Interface :

Bonjour, voici mon compte-rendu concernant ma visionneuse d'images. Tout d'abord, j'ai réalisé cette interface avec la bibliothèque Tkinter ce qui m'a permis de m'aider et de récupérer du code du logiciel de traitement de texte fait en classe. Ma Visionneuse ouvre des fichiers png, gif et jpg. Son interface est simple d'utilisation et intuitive. Elle permet d'ouvrir un fichier préalablement choisi par l'utilisateur, de zoomer et dézoomer sur celle-ci et bien sûr d'un bouton pour quitter mon interface graphique. Les boutons sont tous regroupés dans un bandeau à gauche de l'interface et organisés entre eux de manière intuitive. En effet, l'utilisateur utilise les boutons du haut vers le bas, il commence par sélectionner un fichier et fini par quitter l'application. J'ai choisi un style sobre et épuré pour mon Interface en fonction des différentes contraintes imposées par la bibliothèque Tkinter notamment pour le choix des boutons. J'ai malgré tout essayé de personnaliser mon Interface en ajoutant une icône propre à celle-ci, en modifiant la couleur du bouton « Quitter » et en ajoutant des animations lorsque qu'un bouton est pressé. Concernant la réalisation de mon projet, j'ai principalement réutilisé du code de la précédente interface graphique en l'adaptant, utilisé une vidéo trouvée sur YouTube pour les bases de mon interface et utilisé une IA pour les boutons Zoom et Dezoom et pour le débogage. J'ai aussi utilisé quelques sites externes notamment pour la personnalisation.

Sources :

<https://www.youtube.com/watch?v=N4M4W7JPOL4&t=8s>

<https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQ32O-RRk-zKHqqVnWy4gJiLyhPptT63nY4mw&usqp=CAU>

<https://htmlcolorcodes.com/fr/>

<https://infoforall.fr/python/python-act110.html>

<https://python.doctor/page-tkinter-interface-graphique-python-tutoriel>

<https://waytolearnx.com/2020/06/button-tkinter-python-3.html>

Quelques prompts de l'IA Chatgpt les plus importants :

Voici comment vous pouvez ajouter un bouton pour zoomer sur l'image dans votre visionneuse en utilisant le module tkinter :

```
python Copy code

import tkinter as tk
from tkinter.filedialog import askopenfilename
from PIL import Image, ImageTk

def open_image():
    filepath = askopenfilename(filetypes=[("Fichiers image", "*.jpg;*.jpeg;*.png;*.gif")])
    if filepath:
        image = Image.open(filepath)
        image_tk = ImageTk.PhotoImage(image)
        label.config(image=image_tk)
        label.image = image_tk

def zoom_in():
    global image_tk
    if image_tk:
        new_width = int(label.winfo_width() * 1.1)
        new_height = int(label.winfo_height() * 1.1)
        resized_image = image.resize((new_width, new_height), Image.ANTIALIAS)
        image_tk = ImageTk.PhotoImage(resized_image)
        label.config(image=image_tk)
```

ChatGPT

L'erreur `NameError: name 'PhotoImage' is not defined` indique que Python ne reconnaît pas le nom `PhotoImage`. Cela est dû au fait que vous avez importé `PhotoImage` depuis `PIL`, mais vous ne l'avez pas utilisé correctement dans votre code.

Dans votre fonction `open_image()`, vous essayez de créer un objet `PhotoImage`, mais vous n'avez pas besoin de le faire car vous utilisez déjà `ImageTk.PhotoImage` de la bibliothèque PIL.

Pour corriger l'erreur, vous pouvez supprimer la ligne où vous essayez de créer un objet `PhotoImage` et utiliser directement `ImageTk.PhotoImage` pour charger votre image. Voici comment corriger votre fonction `open_image()` :

```
python Copy code

def open_image():
    global image, image_tk
    filepath = filedialog.askopenfilename(filetypes=[("Fichiers image", "*.jpg;*.jpeg;*.png;*.gif")])
    if filepath:
        image = Image.open(filepath)
        image_tk = ImageTk.PhotoImage(image)
        label.config(image=image_tk)
        label.image = image_tk
```

ChatGPT

Pour que l'image reste dans le cadre de la fenêtre et que sa taille s'adapte pour qu'elle soit entièrement visible, vous pouvez modifier votre fonction `open_image()` pour redimensionner l'image en fonction de la taille de la fenêtre. Voici comment vous pouvez le faire :

```
python Copy code

import tkinter as tk
from tkinter.filedialog import askopenfilename
from tkinter import filedialog
from PIL import Image, ImageTk

# Fonction pour ouvrir une image et la redimensionner pour l'afficher entièrement dans la fenêtre
def open_image():
    global image, image_tk
    filepath = filedialog.askopenfilename(filetypes=[("Fichiers image", "*.jpg;*.jpeg;*.png;*.gif")])
    if filepath:
        image = Image.open(filepath)
        window_width = window.winfo_width() - 20 # Largeur de la fenêtre moins un peu
        window_height = window.winfo_height() - 20 # Hauteur de la fenêtre moins un peu
        image.thumbnail((window_width, window_height))
        image_tk = ImageTk.PhotoImage(image)
        label.config(image=image_tk)
        label.image = image_tk
```

ChatGPT

Pour que l'image apparaisse dans la zone de texte, vous pouvez utiliser un widget `Text` pour afficher l'image au lieu d'un widget `Label`. Voici comment vous pouvez le faire :

```
python Copy code

import tkinter as tk
from tkinter.filedialog import askopenfilename
from tkinter import filedialog
from PIL import Image, ImageTk

def open_image():
    global image, image_tk
    filepath = filedialog.askopenfilename(filetypes=[("Fichiers image", "*.jpg;*.jpeg;*.png;*.gif")])
    if filepath:
        image = Image.open(filepath)
        image_tk = ImageTk.PhotoImage(image)
        text_area.image_create(tk.END, image=image_tk)

def zoom_plus():
    global image_tk
    if image_tk:
```