import time  
import pyttsx3  
import pygame  
import os  
import sys  
import threading  
import multiprocessing  
  
#initialize the pyttsx3 engine  
def speak(text):  
 global is\_talking  
 engine = pyttsx3.init('sapi5')  
 rate = engine.getProperty('rate')  
 engine.setProperty('rate', 190)  
 voices = engine.getProperty('voices')  
 engine.setProperty('voice', voices[0].id)  
 is\_talking = True  
 print("A.S.I.A:" +text+"\n")  
 engine.say(text)  
 engine.runAndWait()  
 is\_talking = False  
  
  
# Initialize Pygame  
pygame.init()  
  
# Screen dimensions  
SCREEN\_WIDTH = 400  
SCREEN\_HEIGHT = 300  
  
# Load GIF frames from a folder  
def load\_gif\_frames(folder\_path):  
 frames = []  
 for filename in sorted(os.listdir(folder\_path)):  
 if filename.endswith(".gif"):  
 frame = pygame.image.load(os.path.join(folder\_path, filename))  
 frames.append(frame)  
 return frames  
  
# Load GIF frames for talking and not talking states  
talking\_frames = load\_gif\_frames(r"C:\Users\HP\Documents\Python\_Scripts\PythonProject\A.S.I.A\ezgif-1-9482c10dc3-gif-im") # Replace with your folder path  
not\_talking\_frames = load\_gif\_frames(r"C:\Users\HP\Documents\Python\_Scripts\PythonProject\A.S.I.A\ezgif-1-74c4a079b6-gif-im") # Replace with your folder path  
  
# Frame rate for GIF  
FPS = 10  
clock = pygame.time.Clock()  
  
# Animation state  
is\_talking = False  
  
def pygame\_animation():  
 global is\_talking  
 frame\_index = 0  
 running = True  
  
 # Screen setup  
 screen = pygame.display.set\_mode((SCREEN\_WIDTH, SCREEN\_HEIGHT))  
 pygame.display.set\_caption("GIF Animation")  
  
 while running:  
 for event in pygame.event.get():  
 if event.type == pygame.QUIT:  
 running = False  
 elif event.type == pygame.KEYDOWN:  
 if event.key == pygame.K\_SPACE:  
 is\_talking = not is\_talking  
  
 # Select frames based on talking state  
 if is\_talking:  
 current\_frames = talking\_frames  
 else:  
 current\_frames = not\_talking\_frames  
  
 # Display the current frame  
 screen.fill((255, 255, 255))  
 screen.blit(current\_frames[frame\_index], (0, 0))  
  
 # Update frame index  
 frame\_index = (frame\_index + 1) % len(current\_frames)  
  
 # Update display  
 pygame.display.flip()  
  
 # Cap the frame rate  
 clock.tick(FPS)  
  
 pygame.quit()  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 # Start the Pygame animation in a separate thread  
 pygame\_thread = threading.Thread(target=pygame\_animation)  
 pygame\_thread.start()

speak("Hello, boss")  
  
  
time.sleep(5)