Ex. No. : 9 **Date:**

Register No.: 231701045 **Name: P. Sahaana**

Create an Application to Send and Receive Multimedia Messages (Text, Image, Audio)

AIM:

To develop a basic chat-like application that allows:

- Sending and receiving text, images, and audio
- Using local file selection and simulated messaging UI

Procedure:

1. Create a GUI using tkinter.
2. Enable sending:
   - Text via entry box
   - Images via file dialog and display
   - Audio via file selection and play
3. Use a scrollable chat window to simulate message exchange.
4. Optionally add threading for media playback.

Program:

```
import tkinter as tk
from tkinter import filedialog, scrolledtext
from PIL import Image, ImageTk
from playsound import playsound
import threading

def send_text():
    msg = text_entry.get()
    if msg:
        chat_box.insert(tk.END, f"You: {msg}\n")
        chat_box.yview(tk.END)
        text_entry.delete(0, tk.END)
```

```
def send_image():
    path = filedialog.askopenfilename(filetypes=[("Image Files", "*.jpg *.png
*.jpeg")])
    if path:
        chat_box.insert(tk.END, f"You sent an image: {path}\n")
        chat_box.yview(tk.END)

        img = Image.open(path)
        img.thumbnail((100, 100))
        img_tk = ImageTk.PhotoImage(img)

        # Display image in chat
        image_label = tk.Label(chat_box.window_create(tk.END), image=img_tk)
        image_label.image = img_tk  # Keep a reference
        chat_box.insert(tk.END, "\n")

def send_audio():
    path = filedialog.askopenfilename(filetypes=[("Audio Files", "*.mp3 *.wav")])
    if path:
        chat_box.insert(tk.END, f"You sent an audio message: {path}\n")
        chat_box.yview(tk.END)
        threading.Thread(target=lambda: playsound(path), daemon=True).start()

# GUI Setup
window = tk.Tk()
window.title("Multimedia Messaging App")
window.geometry("450x550")

# Chat Frame
chat_frame = tk.Frame(window)
chat_box = scrolledtext.ScrolledText(chat_frame, wrap=tk.WORD, width=55,
height=20, state='normal')
chat_box.pack()
chat_frame.pack(pady=10)

# Text Entry
```

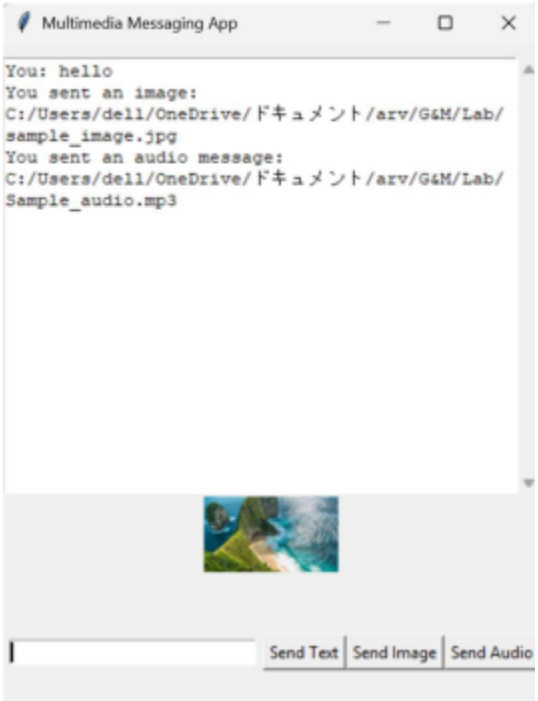
```
entry_frame = tk.Frame(window)
text_entry = tk.Entry(entry_frame, width=30)
text_entry.pack(side=tk.LEFT, padx=5)

btn_text = tk.Button(entry_frame, text="Send Text", command=send_text)
btn_text.pack(side=tk.LEFT)

entry_frame.pack(pady=5)

# Multimedia Buttons
btn_frame = tk.Frame(window)
btn_image = tk.Button(btn_frame, text="Send Image", command=send_image)
btn_audio = tk.Button(btn_frame, text="Send Audio", command=send_audio)
btn_image.pack(side=tk.LEFT, padx=5)
btn_audio.pack(side=tk.LEFT, padx=5)
btn_frame.pack()

# Start GUI
window.mainloop()
```



Result:

A multimedia messaging interface was successfully created with options to send text, image, and audio content. The application simulated messaging behavior in a local environment.