OTP Generation with GUI

September 30, 2024

[ ]:

[ ]:
```
## OTP Generation with Graphical User Interface (GUI)
```

[ ]:
```
# Importing Libraries
import secrets
import string
import smtplib
import re
import time
import warnings
import tkinter as tk
from tkinter import messagebox
warnings.filterwarnings("ignore")


# Global variable
attempt= 3   # to limit otp attempts


# Generating 6 digit otp
def otp_generation():
    sequence = string.digits   # defining variable for 0-9
    otp = ''.join(secrets.choice(sequence) for i in range(6))  # 6 digit otp
    return otp


# Validating entered email id
def validate_email_id(receipient_email):
    email_pattern = r"^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\.[a-zA-Z0-9]+$" ␣
 ↪#Defining pattern
    if re.match(email_pattern, receipient_email):   # Matching condition
        return receipient_email
    else:
        messagebox.showerror("Invalid Email", "Please enter a valid Email ID.")
```

```
# Establishing Server Connection
def send_otp_by_email(receipient_email, otp):
    sender_email = "xxxxxxxxxx@gmail.com"
    sender_password = "xxxx xxxx xxxx xxxx"
    message = f"Subject: Your OTP Code\n\nYour OTP Code is {otp}."  # Message␣
 ↪subject and body

    try:
        server = smtplib.SMTP("smtp.gmail.com", 587)  # Client server and port␣
 ↪number
        server.starttls()                # Starting Transport Layer Security␣
 ↪encryption
        server.login(sender_email, sender_password)
        server.sendmail(sender_email, receipient_email, message)
        server.quit()     # Terminating the SMTP session

    except smtplib.SMTPAuthenticationError:
        messagebox.showerror("Error","Please check sender email and password.")
    except smtplib.SMTPConnectError:
        messagebox.showerror("Error","Please Check the server address and port.
 ↪")
    except Exception as e:
        messagebox.showerror("Error",f"An error occurred: {e}")




# Creating otp
def creating_otp():
    global OTP,start_time  # Defining global variable

    recepient = email_input.get() # Input from GUI
    email=validate_email_id(recepient)  # Validating entered Email Id
    if email:
        OTP = otp_generation()    # Otp from otp_generation function
        send_otp_by_email(email, OTP)
        start_time=time.time()
        messagebox.showinfo("Success", "OTP sent successfully !")
        messagebox.showinfo("Notice","You have 90 seconds")
```

```
# Verifying Otp and Defining Errors
def verifying_otp():
    global attempt, start_time   # global variable
    if attempt>0:
        try:
            entered_otp = otp_input.get()
            current_time=time.time()      # Starting timer of 90 seconds
            elapsed_time=int(current_time-start_time)

            if elapsed_time > 90:    # Checking for otp entry within 90 sec␣
↪timeframe
                messagebox.showerror("Error","\nYour Time is Up !!! Please␣
↪Regenerate OTP")
                otp_input.config(state='disabled')  # Disable further input␣
↪after time up
            elif entered_otp==OTP:                     # Comparing user I/P and␣
↪generated OTP
                messagebox.showinfo("Alert","OTP verified !!!")
                messagebox.showinfo("Success","Access Granted !!!")
            else:
                attempt-= 1
                if attempt > 0:
                    messagebox.showerror("Error", f"Invalid OTP. You have␣
↪{attempt} attempts remaining.")
                else:
                    messagebox.showerror("Error", "You have exhausted all␣
↪attempts.")
                    otp_input.config(state='disabled')  # Disable further input␣
↪after 3 failed attempts

        except Exception as e:
            messagebox.showerror("Error", f"An error occurred: {e}")
```

```
# Graphical User Interface Creating Labels, Buttons, Entry Fields
def create_GUI():
    global email_input, otp_input  # defining global variable
    wind = tk.Tk()
    wind.geometry("300x210")
    wind.title("OTP Verification System")

    tk.Label(wind, text="Enter your Email ID :").pack(pady=10)    # Creating␣
 ↪label for Email Id
    email_input = tk.Entry(wind, width=30)    # Creating I/P box for Email Id
    email_input.pack(pady=1)

    send_otp_button = tk.Button(wind,text="Send OTP",command=creating_otp)
    send_otp_button.pack(pady=8)        # Creating button to send otp

    tk.Label(wind, text="Enter the received OTP :").pack(pady=10) # Creating␣
 ↪label for otp
    otp_input = tk.Entry(wind, width=10)     # Creating I/P for Email Id
    otp_input.pack(pady=1)

    verify_otp_button = tk.Button(wind,text="Verify OTP",command=verifying_otp)
    verify_otp_button.pack(pady=8)    # Creating button to verify otp

    wind.mainloop()   #starts the GUI event loop


# Executing GUI
create_GUI()
```

[ ]: