**FINAL PROJECT**

**Name**: Monica Campoverde

# Assignment:

# Write a GUI-based program designed to collect customer feedback. The application allows users to select from three options: Inquiry, Complaint, or Congratulations, and then enter their feedback.

# **1 (required) Readme Documentation**

Introduction

The Customer Service Application is a GUI-based application designed to collect customer feedback. The application allows users to select from three options: Inquiry, Complaint, or Congratulations, and then enter their feedback.

# **2 (required) Source Code of All files (.py)**

"""

Author: Monica Campoverde

Date written: February -March, 2025

Project: Customer Service App

This secure app will enable the users to register inquiries, congratulations and complaints

"""

#Import libraries

import tkinter as tk #Import Tkinter to create graphical user interfaces

from tkinter import ttk, messagebox #Import message box from Tkinter to display message boxes

from PIL import Image, ImageTk #Import Image and ImageTK from PIL

# Define a class called CustomerServiceApp

class CustomerServiceApp:

    # Define an \_\_init\_\_ method, which is a special method that is called when an object is created from the class

    def \_\_init\_\_(self):

        """

        Initializes the Customer Service Application.

        Creates the login window with username and password entry fields,

        and a login button.

        """

        # Create a window with the title "Login"

        self.login\_window = tk.Tk()  # Create the login window

        self.login\_window.title("Login")  # Set the title of the login window

        self.login\_window.configure(bg='lightgreen')  # Set the background color of the login window

        # Create a label widget with the text "Customer Service App" in bold and large font

        self.title\_label = tk.Label(self.login\_window, text="Customer Service App", bg='lightgreen', font=('Arial', 20, 'bold'))  # Create the title label

        self.title\_label.pack()  # Add the title label to the login window

        # Create a label widget with the text "Username:" and add it to the login window

        self.username\_label = tk.Label(self.login\_window, text="Username:", bg='lightgreen')  # Create the username label

        self.username\_label.pack()  # Add the username label to the login window

        # Create an entry widget for the username and add it to the login window

        self.username\_entry = tk.Entry(self.login\_window)  # Create the username entry field

        self.username\_entry.pack()  # Add the username entry field to the login window

        # Create a label widget with the text "Password:" and add it to the login window

        self.password\_label = tk.Label(self.login\_window, text="Password:", bg='lightgreen')  # Create the password label

        self.password\_label.pack()  # Add the password label to the login window

        # Create an entry widget for the password and add it to the login window

        self.password\_entry = tk.Entry(self.login\_window, show="\*")  # Create the password entry field

        self.password\_entry.pack()  # Add the password entry field to the login window

        # Create a button widget with the text "Login" and add it to the login window

        self.login\_button = tk.Button(self.login\_window, text="Login", command=self.check\_credentials)  # Create the login button

        self.login\_button.pack()  # Add the login button to the login window

    # Define a method called check\_credentials

    def check\_credentials(self):

        """

        Checks the username and password credentials.

        If the credentials are valid, creates the main window with

        registration options. Otherwise, displays an error message.

        """

        # Get the username from the username entry widget

        username = self.username\_entry.get()  # Get the username from the entry field

        # Get the password from the password entry widget

        password = self.password\_entry.get()  # Get the password from the entry field

        # Check if the username and password are correct

        if username == "admin" and password == "password":  # Check if the credentials are valid

            # Destroy the login window

            self.login\_window.destroy()  # Close the login window

            # Create a new window with the title "Customer Service App"

            self.main\_window = tk.Tk()  # Create the main window

            self.main\_window.title("Customer Service App")  # Set the title of the main window

            self.main\_window.configure(bg='lightgreen')  # Set the background color of the main window

            self.main\_window.geometry("800x600")  # Set the size of the main window

            # Load the images

            self.image1 = ImageTk.PhotoImage(Image.open("C:/Users/mcampoverde/Documents/SDEV140 Software Development/Project/image1.jpg").resize((150, 150), Image.LANCZOS))  # Load the first image

            self.image2 = ImageTk.PhotoImage(Image.open("C:/Users/mcampoverde/Documents/SDEV140 Software Development/Project/image2.jpg").resize((150, 150), Image.LANCZOS))  # Load the second image

            self.image3 = ImageTk.PhotoImage(Image.open("C:/Users/mcampoverde/Documents/SDEV140 Software Development/Project/image3.jpg").resize((150, 150), Image.LANCZOS))  # Load the third image

            # Create a label widget to explain the user and add it to the main window

            self.options\_label = tk.Label(self.main\_window, text="Please select one of the following options:", bg='lightgreen', font=('Arial', 14, 'bold'))  # Create the options label

            self.options\_label.pack()  # Add the options label to the main window

            # Create a frame to hold the buttons

            self.button\_frame = tk.Frame(self.main\_window, bg='lightgreen')  # Create the button frame

            self.button\_frame.pack()  # Add the button frame to the main window

            # Create a button widget with the text "Inquiry" and add it to the button frame

            self.inquiry\_button = tk.Button(self.button\_frame, text="Inquiry", image=self.image1, compound=tk.TOP, command=self.inquiry\_callback)  # Create the inquiry button

            self.inquiry\_button.image = self.image1  # Keep a reference to the image

            self.inquiry\_button.pack(side=tk.LEFT)  # Add the inquiry button to the button frame

            # Create a button widget with the text "Complaint" and add it to the button frame

            self.complaint\_button = tk.Button(self.button\_frame, text="Complaint", image=self.image2, compound=tk.TOP, command=self.complaint\_callback)  # Create the complaint button

            self.complaint\_button.image = self.image2  # Keep a reference to the image

            self.complaint\_button.pack(side=tk.LEFT)  # Add the complaint button to the button frame

            # Create a button widget with the text "Congratulations" and add it to the button frame

            self.congratulations\_button = tk.Button(self.button\_frame, text="Congratulations", image=self.image3, compound=tk.TOP, command=self.congratulations\_callback)  # Create the congratulations button

            self.congratulations\_button.image = self.image3  # Keep a reference to the image

            self.congratulations\_button.pack(side=tk.LEFT)  # Add the congratulations button to the button frame

            # Create a text entry widget for the user to enter their inquiry, complaint or congratulations

            self.entry = tk.Text(self.main\_window, height=10, width=40)  # Create the text entry field

            self.entry.pack()  # Add the text entry field to the main window

            # Create a button widget with the text "Save" and add it to the main window

            self.save\_button = tk.Button(self.main\_window, text="Save", command=self.save\_callback)  # Create the save button

            self.save\_button.pack()  # Add the save button to the main window

            # Create a label widget to display the thank you message

            self.thank\_you\_label = tk.Label(self.main\_window, text="", bg='lightgreen')  # Create the thank you label

            self.thank\_you\_label.pack()  # Add the thank you label to the main window

            # Create a button widget to Exit and add it to the main window

            self.exit\_button = tk.Button(self.main\_window, text="Exit", command=self.exit\_callback)  # Create the exit button

            self.exit\_button.pack()  # Add the exit button to the main window

            # Start the main loop

            self.main\_window.mainloop()  # Start the tkinter event loop to display the window and wait for the user interaction

        else:

            # Display an error message if the credentials are invalid

            messagebox.showerror("Error", "Invalid username or password")  # Display an error message

    # Define a method called inquiry\_callback

    def inquiry\_callback(self):

        """

        Sets the text of the entry widget to "Inquiry: "

        """

        # Set the text of the entry widget to "Inquiry: "

        self.entry.delete("1.0", "end")  # Clear the text entry field

        self.entry.insert("1.0", "Inquiry: ")  # Insert the inquiry text

    # Define a method called complaint\_callback

    def complaint\_callback(self):

        """

Sets the text of the entry widget to "Complaint: "

        """

        # Set the text of the entry widget to "Complaint: "

        self.entry.delete("1.0", "end")  # Clear the text entry field

        self.entry.insert("1.0", "Complaint: ")  # Insert the complaint text

    # Define a method called congratulations\_callback

    def congratulations\_callback(self):

        """

        Sets the text of the entry widget to "Congratulations: "

        """

        # Set the text of the entry widget to "Congratulations: "

        self.entry.delete("1.0", "end")  # Clear the text entry field

        self.entry.insert("1.0", "Congratulations: ")  # Insert the congratulations text

    # Define a method called save\_callback

    def save\_callback(self):

        """

        Saves the text from the entry widget to a file.

        """

        # Get the text from the entry widget

        text = self.entry.get("1.0", "end-1c")  # Get the text from the entry field

        # Check if the text is not empty

        if text:  # Check if the text is not empty

            # Save the text to a file

            with open("registers.txt", "a") as f:  # Open the file in append mode

                f.write(text + "\n")  # Write the text to the file

            # Display the thank you message

            self.thank\_you\_label.config(text="Thank you. Your feedback is very important!", font=('Arial', 18, 'bold'))  # Display the thank you message

        else:

            # Display an error message if the text is empty

            messagebox.showerror("Error", "Please enter some text.")  # Display an error message

    # Define a method called exit\_callback

    def exit\_callback(self):

        """

        Exits the application.

        """

        # Check if there is any unsaved text

        text = self.entry.get("1.0", "end-1c")  # Get the text from the entry field

        if text:  # Check if there is any unsaved text

            # Ask the user if they want to save the text before exiting

            if messagebox.askyesno("Confirm", "Do you want to save the text before exiting?"):  # Ask the user for confirmation

                # Save the text to a file

                with open("registers.txt", "a") as f:  # Open the file in append mode

                    f.write(text + "\n")  # Write the text to the file

        # Destroy the main window

        self.main\_window.destroy()  # Close the main window

if \_\_name\_\_ == "\_\_main\_\_":

    # Create an instance of the CustomerServiceApp class

    app = CustomerServiceApp()  # Create an instance of the CustomerServiceApp class

    # Start the main loop

    app.login\_window.mainloop()  # Start the tkinter event loop to display the window and wait for the user interaction

# 5 (required) Three Use Case Screen Shots

|  |  |  |
| --- | --- | --- |
| Example 1 - Inquiry | Example 2 – Complaint | Example 3 - Congratulations |
| A screenshot of a computer  AI-generated content may be incorrect. |  |  |

A screenshot of a computer

AI-generated content may be incorrect.

# 6 (Optional) GitHub URL

https://github.com/mcampoverd/Customer-Service.git