Gisma University of Applied Sciences

# Assessment Submission Form

| | |
|---|---|
| **Student Number** (If this is group work, please include the student numbers of all group participants) | GH1024695<br>Mahdieh Rajabi |
| **Assessment Title** | Build a GUI |
| **Module Code** | M604R |
| **Module Title** | Advance programming |
| **Module Tutor** | William Morrison |
| **Date Submitted** | 11.01.2024 |

**Declaration of Authorship**

I declare that all material in this assessment is my own work except where there is clear acknowledgement and appropriate reference to the work of others.

I fully understand that the unacknowledged inclusion of another person's writings or ideas or works in this work may be considered plagiarism and that, should a formal investigation process confirms the allegation, I would be subject to the penalties associated with plagiarism, as per GISMA Business School, University of Applied Sciences' regulations for academic misconduct.

Signed……Mahdieh Rajabi…….. Date ……11.01.2024……

Link: https://github.com/Mahdiehrajabi/Advance-Programming2

# Table of contents

# Introduction

As a new software developer, requested to build a GUI with require dataset. GUI (Graphical user interface) refers to a model that allows users to have interaction with software applications, computers with icon, logo, button, and menu.

According to assessment brief, I found a dataset that is related to most crowded airports which is about society and services. I downloaded it from Kaggle website and uploaded it to GitHub repository as well.

For carrying out this project I used PyCharm.

At first, I installed some related packages like pandas, tkinter etc.

With this command in terminal of PyCharm: pip install pandas.

And for tkinter is the same step.

Then I wanted to build a GUI and I used a function to design it.

```
#import libraries
import sys
import tkinter as tk
import csv
import matplotlib.pyplot as plt
import requests
from matplotlib.backends.backend_qt5agg import FigureCanvasQTAgg as FigureCanvas
from matplotlib.figure import Figure

```

At the beginning of the code, I used different libraries or packages with ‘import’.

For example:

Import os

With os module, may communicate with the operating system and carry out several operations on files, directories, environment variables, and etc.

Import pandas as pd

It is a popular library in python. Used pandas functions, classes, and objects throughout code. This is a commonly used practice to make sure code which more concise and readable as well.

Import tkinter as tk

The default Python library for building graphical user interfaces (GUIs) is called tkinter. By importing it as ‘tk’, ease the names that must write when utilizing tkinter elements and widgets, making it simpler to reference the library's elements in code.

used the ‘tk’ in code to access and interact with the classes and functions of the tkinter library.

Import csv

Imports a csv module used to parse tabular-like data structures, such as data in excel format, and these files are saved in. This file can be read and written using different classes provided by the module with the.csv extension.

Import matplotlib

This library's most crucial component, the Figure class, shows the complete figure or the top-level container that contains all of the drawn plotting elements, such as axes and labels.

and another library needed that called ‘requests’ to fetch the data from API. API is Application Programming Interface: set of protocols that allows to communicate different applications with together. (They are fundamental)

# Create GUI

```
# Create a window
root = tk.Tk()
root.title("Airport Analyzer")

data_selection_frame = tk.Frame(root)
data_selection_frame.pack(pady=10)

airport_loc_label = tk.Label(data_selection_frame, text="Select airport location:")
airport_loc_label.pack(side=tk.LEFT)

airport_loc_options = ["All", "Name", "TotalSeats", "Country Name"]
airport_loc_combobox = tk.ttk.Combobox(data_selection_frame, values=airport_loc_options)
airport_loc_combobox.pack(side=tk.LEFT)

# Button for visualization
generate_visualization_button = tk.Button(data_selection_frame, text="Generate Visualization", command=generate_visualization)
generate_visualization_button.pack()


visualization_frame = tk.Frame(root)
visualization_frame.pack(pady=20)

# Canvas for visualization
visualization_canvas = tk.Canvas(visualization_frame, width=500, height=400)
visualization_canvas.pack()

    visualization_canvas.delete('all')
    visualization_canvas.create_image(200, 150, image=visualization)

def airport_data():

    data = pd.read_csv('airport_volume_airport_locations.csv')
    return data

def create_visualization(data):

    pie_chart = plt.pie(data['locations'].value_counts())
    plt.title('Airport location')
    plt.show()

root.mainloop()
```

## And then fetch data from our dataset:

```
    def fetch(self, dataset):


        api_url = f"https://www.kaggle.com/datasets/zvr842/all-global-airports"
        response = requests.get(api_url)



    def plot_data(self, data):


        x_values = list(range(len(data)))
        y_values = [item['value'] for item in data]

        figure, ax = plt.subplots()
        ax.plot(x, y, label="Values")
        ax.set_xlabel("X")
        ax.set_ylabel("Y")
        ax.legend()

    def clear_data(self):
        self.airport_data = None
        self.airport_list_widget.clear()
        self.details_table.setRowCount(0)
        self.ax.clear()
        self.canvas.draw()
        self.update_status_bar('Data cleared successfully.')


        canvas = FigureCanvasTkAgg(fig, master=self.root)
        canvas_widget = canvas.get_tk_widget()
        canvas_widget.grid(row=2, column=0, columnspan=2, pady=10)

```

Data can be centrally stored on a server or database. To bring information from these centralized sources into the local application environment, it needs to be fetched.  Before the data is viewed or evaluated, preprocessing may be required. This could include data transformation and cleaning. Gathering data should come first before carrying out any necessary preparatory steps.

Next stage is data visualization that used matplotlib to create resizable visualizations (such as line charts and graphs) of the dataset using matplotlib.

# References:

Codemy.com. (2019, January 10). *Create graphical user interfaces with Python and TKINTEr* [Video]. YouTube. https://www.youtube.com/watch?v=yQSEXcf6s2I

Developedbyed. (2023, January 13). *Build a modern Python GUI Project | Step by step tutorial* [Video]. YouTube. https://www.youtube.com/watch?v=NI9LXzo0UY0

Python Simplified. (2022, June 15). *Create GUI App with Tkinter and SQLite - Step by Step Python Tutorial for Beginners* [Video]. YouTube. https://www.youtube.com/watch?v=5qOnzF7RsNA