Laboratory Activity No. 4	
Introduction to GUI Development using Pycharm	
Course Code: CPE009B	**Program:** Computer Engineering
Course Title: Object-Oriented Programming	**Date Performed: October 16, 2024**
Section: CPE21S1	**Date Submitted: October 17, 2024**
Name(s): GASPAR, AARON ROWEN O.	**Instructor: Ms. Maria Rizette Sayo**

6. Supplementary Activity

registration.py

```python
from tkinter import *
from tkinter import messagebox

3 usages
class Registration:
    def __init__(self, win):
        self.Lbl1 = Label(win, fg="Green", bg="Dark Grey", text="Account Registration System", font=("Times New Roman", 22))
        self.Lbl1.place(x=30, y=30)

        self.Lbl2 = Label(win, text="First Name : ", bg="Dark Grey", font=("Times New Roman", 10), fg="Green")
        self.Lbl2.place(x=74, y=100)
        self.Entry1 = Entry(win, bd=3)
        self.Entry1.place(x=200, y=100)

        self.Lbl3 = Label(win, text="Surname : ", bg="Dark Grey", font=("Times New Roman", 10), fg="Green")
        self.Lbl3.place(x=74, y=140)
        self.Entry2 = Entry(win, bd=3)
        self.Entry2.place(x=200, y=140)

        self.Lbl4 = Label(win, text="Username : ", bg="Dark Grey", font=("Times New Roman", 10), fg="Green")
        self.Lbl4.place(x=74, y=180)
        self.Entry3 = Entry(win, bd=3)
        self.Entry3.place(x=200, y=180)
```

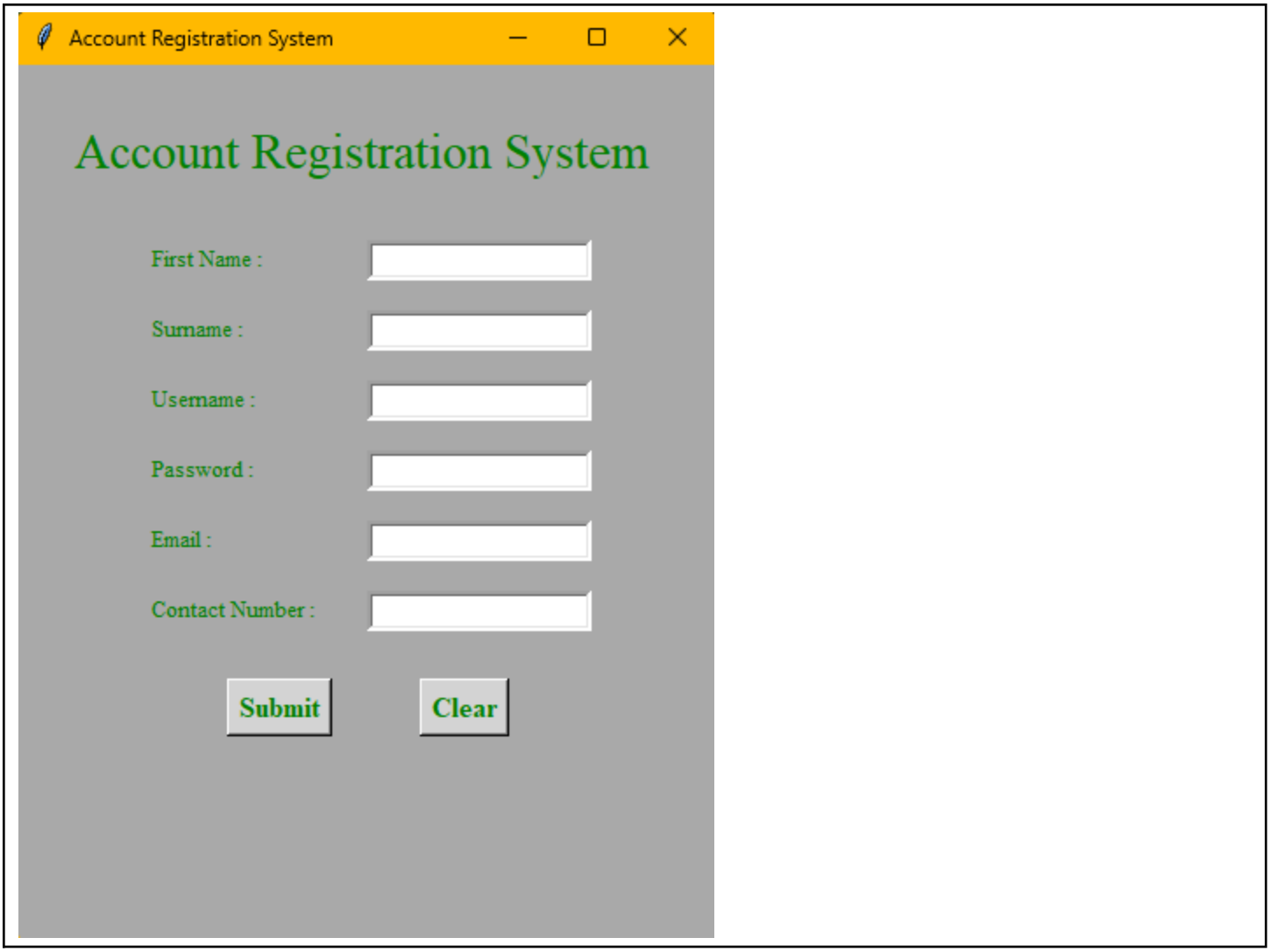
```
        self.Lbl5 = Label(win, text="Password : ", bg="Dark Grey", font=("Times New Roman", 10), fg="Green")
        self.Lbl5.place(x=74, y=220)
        self.Entry4 = Entry(win, bd=3, show="*")
        self.Entry4.place(x=200, y=220)

        self.Lbl6 = Label(win, text="Email : ", bg="Dark Grey", font=("Times New Roman", 10), fg="Green")
        self.Lbl6.place(x=74, y=260)
        self.Entry5 = Entry(win, bd=3)
        self.Entry5.place(x=200, y=260)

        self.Lbl7 = Label(win, text="Contact Number : ", bg="Dark Grey", font=("Times New Roman", 10), fg="Green")
        self.Lbl7.place(x=74, y=300)
        self.Entry6 = Entry(win, bd=3)
        self.Entry6.place(x=200, y=300)

        self.Button1 = Button(win, fg="Green", text="Submit", command=self.submit, bg="Light Grey", font=("Times New Roman bold", 12))
        self.Button1.place(x=120, y=350)
        self.Button2 = Button(win, fg="Green", text="Clear", command=self.clear, bg="Light Grey", font=("Times New Roman bold", 12))
        self.Button2.place(x=230, y=350)

        win.config(bg="Dark Grey")
```

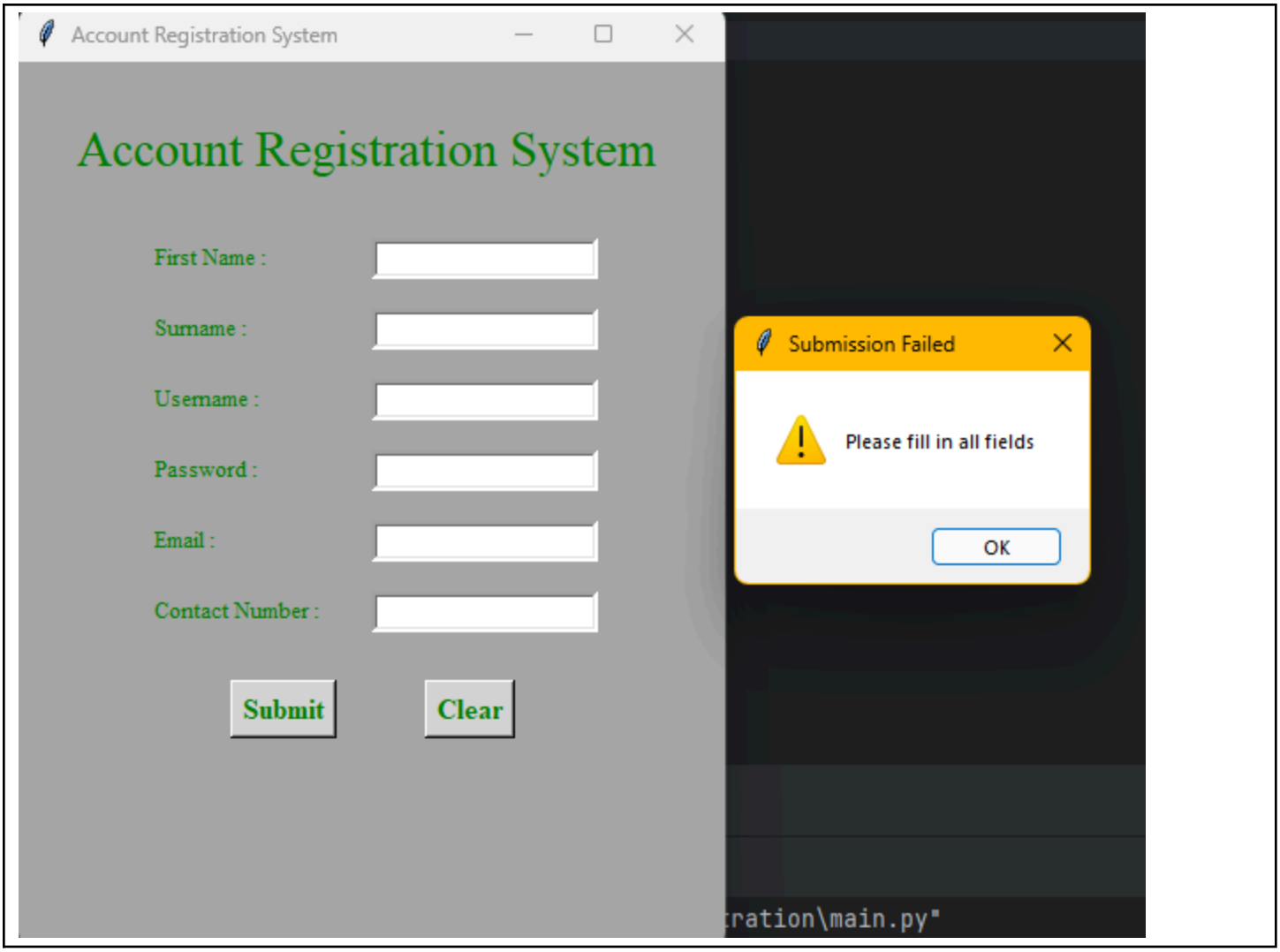
```
    1 usage
    def submit(self):
        first_name = self.Entry1.get()
        surname = self.Entry2.get()
        username = self.Entry3.get()
        password = self.Entry4.get()
        email = self.Entry5.get()
        contact_number = self.Entry6.get()

        if all([first_name, surname, username, password, email, contact_number]):
            messagebox.showinfo( title: "Submission Successful",  message: "Registration Submitted Successfully!")
        else:
            messagebox.showwarning( title: "Submission Failed",  message: "Please fill in all fields")

    1 usage
    def clear(self):
        self.Entry1.delete( first: 0, END)
        self.Entry2.delete( first: 0, END)
        self.Entry3.delete( first: 0, END)
        self.Entry4.delete( first: 0, END)
        self.Entry5.delete( first: 0, END)
        self.Entry6.delete( first: 0, END)
```

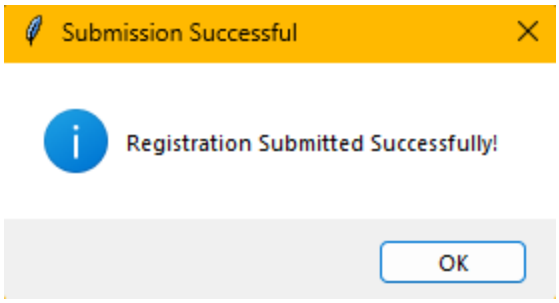
```
if __name__ == "__main__":
    window = Tk()
    app = Registration(window)
    window.geometry("400x500+720+250")
    window.title("Account Registration System")
    window.mainloop()
```

main.py

```
from registration import Registration
import tkinter as tk

if __name__ == "__main__":
    window = tk.Tk()
    app = Registration(window)
    window.geometry("400x500+720+250")
    window.title("Account Registration System")
    window.mainloop()
```


Account Registration System
Account Registration System
First Name :
Surname :
Username :
Password :
Email :
Contact Number :
Submit
Clear


Account Registration System
Account Registration System
First Name :
Surname :
Username :
Password :
Email :
Contact Number :
Submit
Clear
Submission Failed
Please fill in all fields
OK
ration\main.py"



7. Questions

1. What are the common GUI Applications that general end-users such as home users, students, and office employees use? (give at least 3 and describe each)

- Common GUI applications include the Microsoft Office Suite, web browsers like Google Chrome and Mozilla Firefox, and media players such as VLC Media Player and Windows Media Player. The Microsoft Office Suite, which includes Word, Excel, and PowerPoint, is used for creating documents, spreadsheets, and presentations. These applications offer a user-friendly interface with menus, toolbars, and icons that make it easy to perform various tasks. Web browsers allow users to access and navigate the internet, providing a graphical interface with tabs, bookmarks, and address bars, making it easy to browse websites, manage downloads, and use web applications. Media players are used to play audio and video files, offering controls for play, pause, stop, and volume, as well as features like playlists and media libraries.

2. Based on your answer in question 1, why do you think home users, students, and office employees use those GUI programs?

- Home users, students, and office employees use the Microsoft Office Suite because it is essential for creating and editing documents, managing data, and preparing presentations. These applications are widely recognized and compatible with many other software and systems. Web browsers are indispensable for accessing information, communicating via email, and using online services. They support a wide range of activities from research to entertainment. Media

players are popular for entertainment purposes, allowing users to watch videos and listen to music. They support various file formats and provide a seamless media experience.

3. How does Pycharm help developers in making GUI applications, what would be the difference if developers made GUI programs without GUI Frameworks such as Pycharm or Tkinter?

- PyCharm is an Integrated Development Environment (IDE) for Python that supports GUI development by providing features like code completion, debugging, and integration with version control systems. It simplifies the process of writing and testing GUI code. Developing GUI applications without frameworks like PyCharm or Tkinter would be more challenging and time-consuming. These frameworks provide pre-built components and tools that streamline the development process, whereas without them, developers would need to write more low-level code to handle GUI elements and events.

4. What are the different platforms a GUI program may be created and deployed on? (Three is required then state why might a program be created on that specific platform)

- Many GUI applications are developed for Windows due to its large user base and extensive support for various software. It is a common platform for business and personal use. macOS is known for its sleek design and strong performance, making it preferred for creative applications like graphic design and video editing. It offers a consistent user experience and robust security features. Linux is often used for development and server management, supporting a wide range of open-source GUI applications. It is favored for its flexibility, security, and cost-effectiveness.

5. What is the purpose of app = QApplication(sys.argv), ex = App(), and sys.exit(app.exec_())?

- The purpose of app = QApplication(sys.argv), ex = App(), and sys.exit(app.exec_()) is as follows: app = QApplication(sys.argv) initializes the application and handles command-line arguments, setting up the application environment and preparing it to run. ex = App() creates an instance of the main application window or widget, setting up the GUI elements and their layout. sys.exit(app.exec_()) starts the application's event loop, which waits for user interactions and events. The sys.exit() ensures a clean exit from the application when the event loop ends.

8. Conclusion

In summary, graphical user interface (GUI) applications like the Microsoft Office Suite, web browsers, and media players are essential tools for home users, students, and office employees due to their user-friendly interfaces and wide range of functionalities. These applications facilitate productivity, information access, and entertainment, making them indispensable in daily tasks. GUI programs can be developed and deployed on various platforms, including Windows, macOS, and Linux, each offering unique advantages such as user base size, design aesthetics, and flexibility. PyCharm, as an IDE, significantly aids developers in creating GUI applications by providing advanced features that streamline coding and debugging processes. Without such frameworks, developing GUI applications would be more complex and time-consuming, requiring more low-level coding. Overall, the use of GUI applications and development tools enhances efficiency and user experience across different platforms and user groups.

9. Assessment Rubric