Int-213_Python Project

On

Capstone Supervisor Allocation Portal

School of Computer Science and Engineering

Capstone Supervisor Allocation Portal For LPU Students Using Python

** Submitted To **Navpreet Rupal** (Faculty INT 213) On 10/Nov/2022**

Submitted By	**R**oll Number	**R**eg. Number
ABHISHEK KUMAR	**36**	**12115093**
SANJAY KUMAR NAYAK	11	12115352
SAKSHAM BAJPAI	60	12115837

Name –> Abhishek Kumar

Reg. no. –> 12115093

Section -> K21QT

Roll no. –> A36

Subject -> Python(project)

Teacher -> Er. Navpreet Rupal

Topic -> Capstone Supervisor Allocation Portal for LPU Students

College -> Lovely Professional University

Acknowledgement

I *Abhishek Kumar* and my teammates _Sanjay Kumar Nayak _and _ SAKSHAM BAJPAI _have taken efforts to complete this project. However, it would not have been possible without the kind support and help of my professor _ **Navpreet Rupal** (Faculty INT-213).

I would like to extend my sincere thanks to all those who helped with the project & also for their support in completing the project. This project helped me and my team members hone our Python programming skills and learn how to use programming in real life. There were a lot of online resources that I referred to while preparing the project.

** Summary**

The project titled '**Capstone supervisor allocation portal for LPU students using python**' is a GUI approach to create a software that helps in allocating supervisors to various registered students for their capstone project.

INTRODUCTION

In this project we have made a **Capstone Supervisor Allocation Portal**, the technology that we have used is solely **Python Tkinter**. _ Tkinter_ is the standard GUI (Graphical User Interface) library for *Python*. _Python _when combined with _Tkinter _provides a fast and easy way to create GUI applications. _Tkinter _provides a powerful object-oriented interface to the Tk GUI toolkit. This project has been divided into two segments: -

1. ****Student Segment** **which helps new students register for the supervisor allocation and pre-registered students to login and check the name of their supervisor and their basic details.

2. ****Supervisor Segment **** which helps new supervisors to register for his availability and pre-registered supervisors to check the name of allocated students under them and their basic details.

BASIC MODULE DIVISION

Opening Student Window

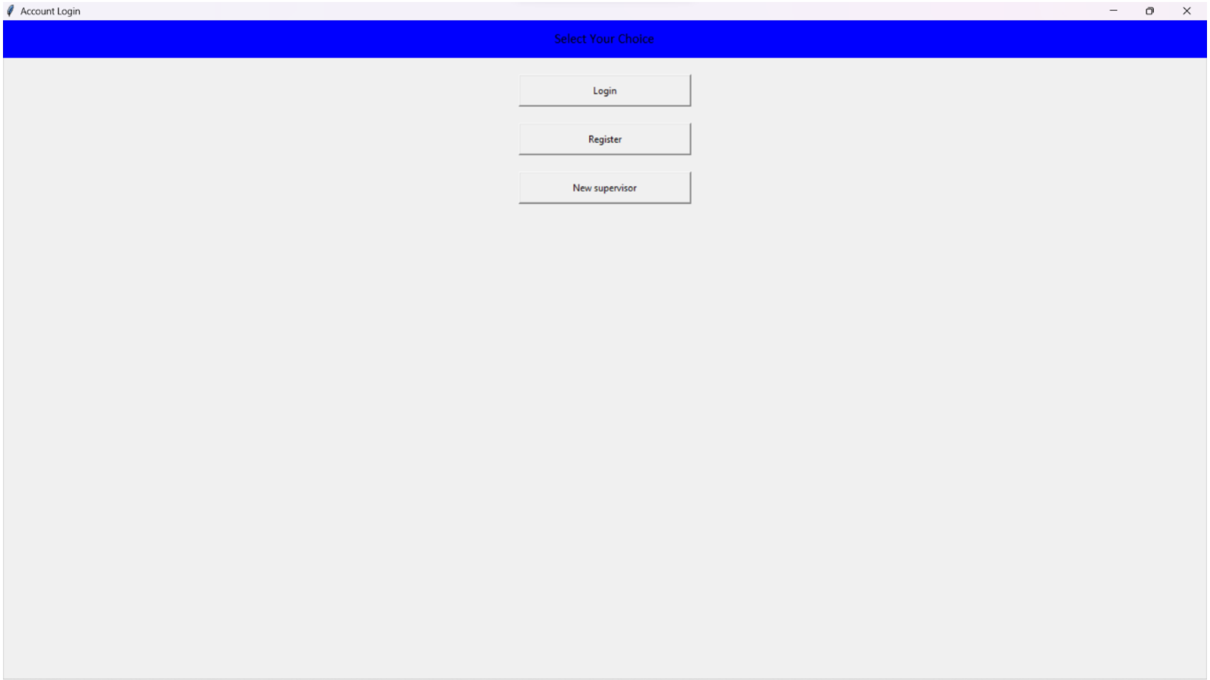
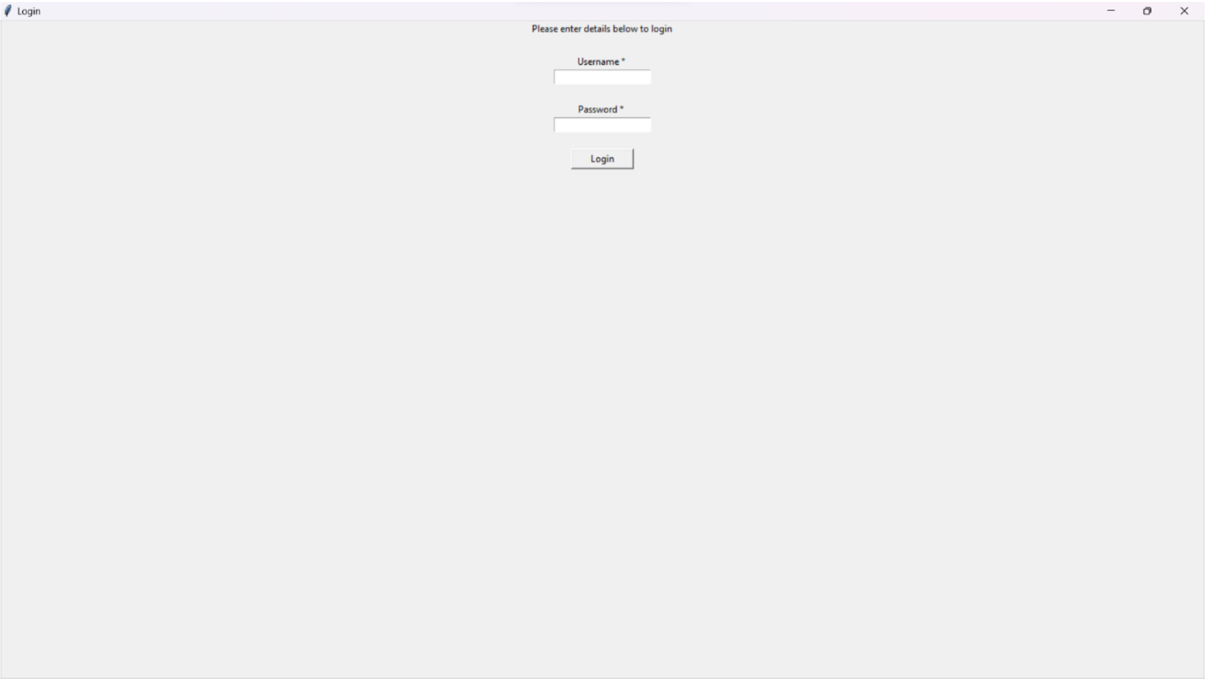


Fig 1

This is the first window that opens for the students. It has three buttons the first one is _the login _button, when clicked it opens the login window. The second button is the *Register* button, when this button is clicked it opens a window for new registrations. The third is the _New Supervisor _button, which takes the student to the _login _window and when the student is logged in the student is taken to the form window which needs to be filled to the new supervisor.

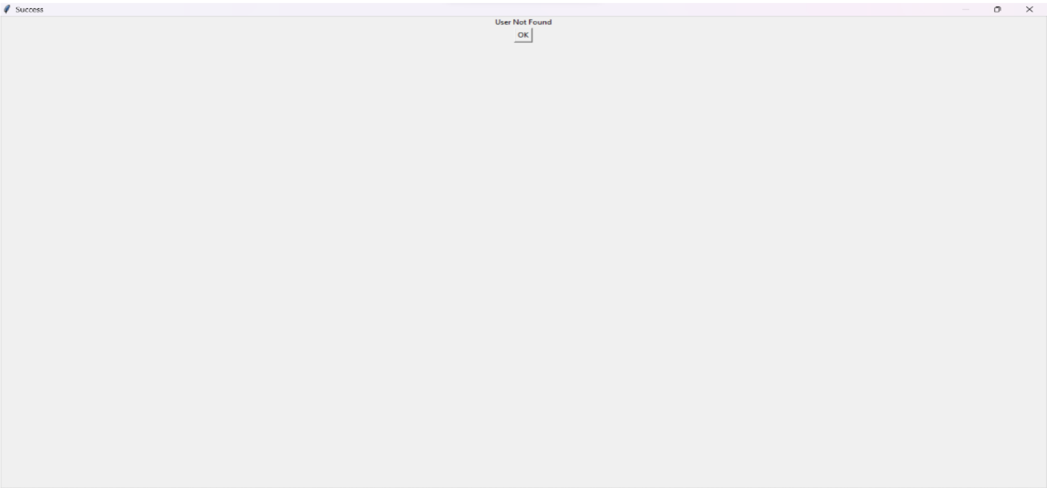
Student Login Window



Please enter details below to login

Fig 2

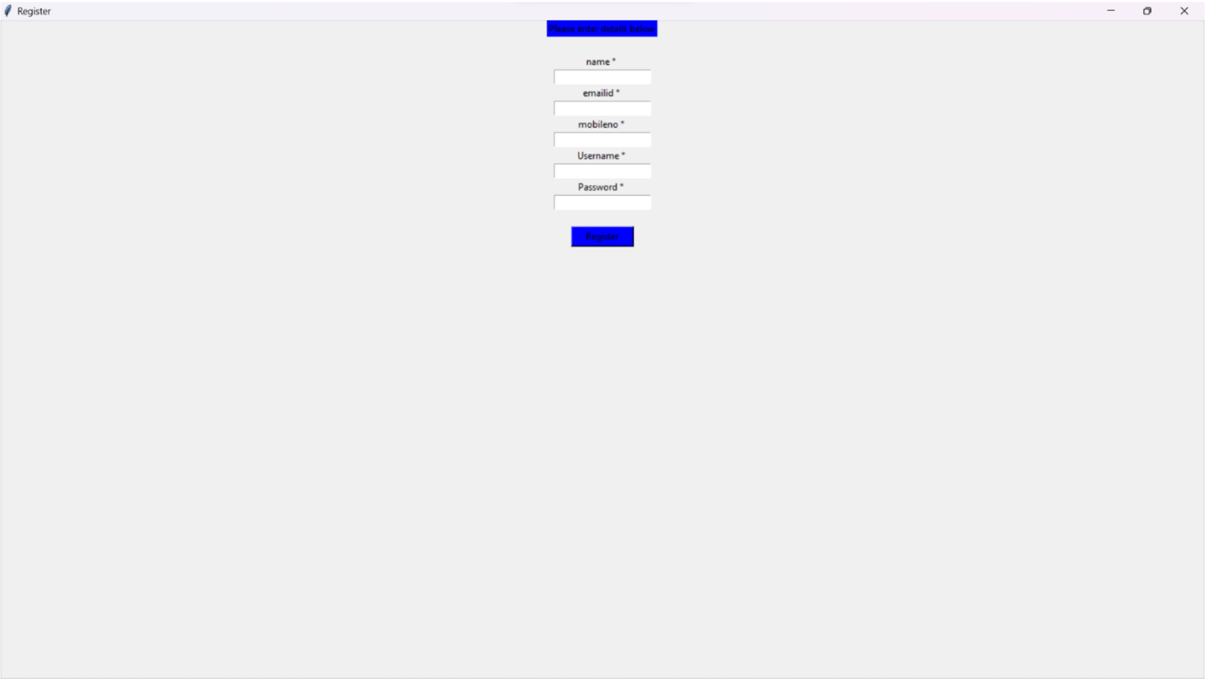
Now, this is the *login* window where the student is taking is the _login _button is pressed. The _login _form requires the _Username _and *Password* of the already registered student.



User Not Found

Fig 3

New Student’s Registration



Please enter details below

Fig 4

This is the window that opens when the student presses the *New User* button in the opening window. This form contains five fields, so when a student fills this and press the *Registered* button he or she gets registered and the data is stored in the database.

New Supervisor’s Login Window

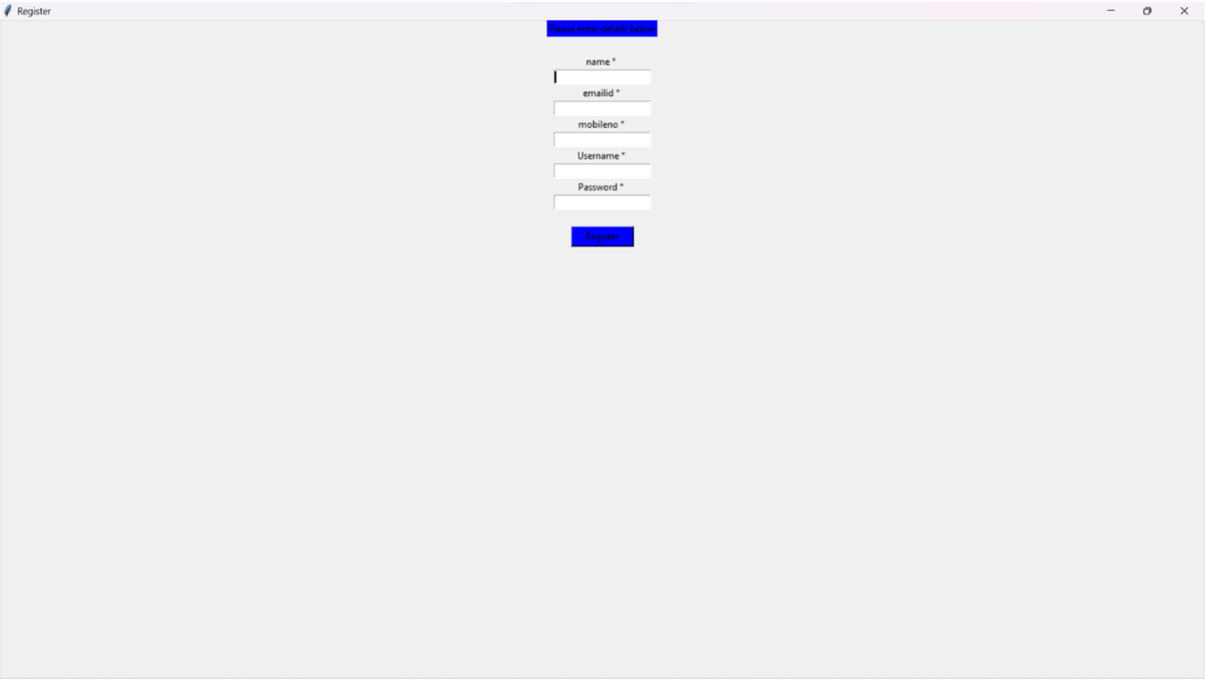


Fig 5

This is the _ Login_ window which opens when a supervisor or user clicks the login button. It is the same as the login window for the students. An already registered uses need to fill in the _Username _and the _Password _credentials to log in.

Reference

1. GeeksforGeeks.org
2. StackOverflow
3. TutorialsPoint

ANNEXURE-A:

```
from tkinter import *

import os


# Designing window for registration


def register():

    global register_screen

    register_screen = Toplevel(main_screen)

    register_screen.title("Register")

    register_screen.geometry("300x250")


    global name

    global emailid

    global mobileno

    global username

    global password

    global name_entry

    global emailid_entry

    global mobileno_entry

    global username_entry
    global password_entry
    name = StringVar()
    emailid = StringVar()
    mobileno = StringVar()
    username = StringVar()
    password = StringVar()

    Label(register_screen, text="Please enter details below", bg="blue").pack()
    Label(register_screen, text="").pack()
    name_lable = Label(register_screen, text="name * ")
    name_lable.pack()
    name_entry = Entry(register_screen, textvariable=name)
    name_entry.pack()
    emailid_lable = Label(register_screen, text="emailid * ")
    emailid_lable.pack()
    emailid_entry = Entry(register_screen, textvariable=emailid)
    emailid_entry.pack()
    mobileno_lable = Label(register_screen, text="mobileno * ")
    mobileno_lable.pack()
    mobileno_entry = Entry(register_screen, textvariable=mobileno)
    mobileno_entry.pack()
    username_lable = Label(register_screen, text="Username * ")

    username_lable.pack()

    username_entry = Entry(register_screen, textvariable=username)

    username_entry.pack()

    password_lable = Label(register_screen, text="Password * ")

    password_lable.pack()

    password_entry = Entry(register_screen, textvariable=password,
show='*')

    password_entry.pack()

    Label(register_screen, text="").pack()

    Button(register_screen, text="Register", width=10, height=1,
bg="blue", command = register_user).pack()

#new supervisor

def newsupervisor():

    global newsupervisor_screen

    newsupervisor_screen = Toplevel(main_screen)

    newsupervisor_screen.title("Register")

    newsupervisor_screen.geometry("300x250")



    global name

    global emailid

    global mobileno
    global username
    global password
    global name_entry
    global emailid_entry
    global mobileno_entry
    global username_entry
    global password_entry
    name = StringVar()
    emailid = StringVar()
    mobileno = StringVar()
    username = StringVar()
    password = StringVar()

    Label(newsupervisor_screen, text="Please enter details below", bg="blue").pack()
    Label(newsupervisor_screen, text="").pack()
    name_lable = Label(newsupervisor_screen, text="name * ")
    name_lable.pack()
    name_entry = Entry(newsupervisor_screen, textvariable=name)
    name_entry.pack()
    emailid_lable = Label(newsupervisor_screen, text="emailid * ")
    emailid_lable.pack()
    emailid_entry = Entry(newsupervisor_screen, textvariable=emailid)

    emailid_entry.pack()

    mobileno_lable = Label(newsupervisor_screen, text="mobileno *
")

    mobileno_lable.pack()

    mobileno_entry = Entry(newsupervisor_screen,
textvariable=mobileno)

    mobileno_entry.pack()

    username_lable = Label(newsupervisor_screen, text="Username *
")

    username_lable.pack()

    username_entry = Entry(newsupervisor_screen,
textvariable=username)

    username_entry.pack()

    password_lable = Label(newsupervisor_screen, text="Password *
")

    password_lable.pack()

    password_entry = Entry(newsupervisor_screen,
textvariable=password, show='*')

    password_entry.pack()

    Label(newsupervisor_screen, text="").pack()

    Button(newsupervisor_screen, text="Register", width=10,
height=1, bg="blue", command = newsupervisor_user).pack()
```

```
# Designing window for login


def login():
    global login_screen
    login_screen = Toplevel(main_screen)
    login_screen.title("Login")
    login_screen.geometry("300x250")
    Label(login_screen, text="Please enter details below to
login").pack()
    Label(login_screen, text="").pack()

    global username_verify
    global password_verify

    username_verify = StringVar()
    password_verify = StringVar()

    global username_login_entry
    global password_login_entry

    Label(login_screen, text="Username * ").pack()
    username_login_entry = Entry(login_screen,
textvariable=username_verify)

    username_login_entry.pack()

    Label(login_screen, text="").pack()

    Label(login_screen, text="Password * ").pack()

    password_login_entry = Entry(login_screen,
textvariable=password_verify, show= '*')

    password_login_entry.pack()

    Label(login_screen, text="").pack()

    Button(login_screen, text="Login", width=10, height=1, command
= login_verify).pack()


# Implementing event on register button



def register_user():

   # name_info = username.get()

    #register_user_screen.geometry("650x450")

    username_info = username.get()

    password_info = password.get()



    file = open(username_info, "w")

    file.write(username_info + "\n")

    file.write(password_info)
    file.close()



    username_entry.delete(0, END)
    password_entry.delete(0, END)

    Label(register_screen, text="Registration Success", fg="green",
font=("calibri", 11)).pack()
# new supervisor2
def newsupervisor_user():
   # name_info = username.get()
    #register_user_screen.geometry("650x450")
    username_info = username.get()
    password_info = password.get()

    file = open(username_info, "w")
    file.write(username_info + "\n")
    file.write(password_info)
    file.close()



    username_entry.delete(0, END)
    password_entry.delete(0, END)


    Label(newsupervisor_screen, text="Registration Success", fg="green", font=("calibri", 11)).pack()
# Implementing event on login button


def login_verify():
    username1 = username_verify.get()
    password1 = password_verify.get()
    username_login_entry.delete(0, END)
    password_login_entry.delete(0, END)


    list_of_files = os.listdir()
    if username1 in list_of_files:
        file1 = open(username1, "r")
        verify = file1.read().splitlines()
        if password1 in verify:
            login_sucess()


        else:
            password_not_recognised()
    else:
        user_not_found()


# Designing popup for login success


def login_sucess():
    global login_success_screen
    login_success_screen = Toplevel(login_screen)
    login_success_screen.title("Success")
    login_success_screen.geometry("250x150")
    Label(login_success_screen, text="Login Success").pack()
    Button(login_success_screen, text="OK",
command=delete_login_success).pack()


# Designing popup for login invalid password


def password_not_recognised():
    global password_not_recog_screen
    password_not_recog_screen = Toplevel(login_screen)
    password_not_recog_screen.title("Success")
    password_not_recog_screen.geometry("150x100")
    Label(password_not_recog_screen, text="Invalid Password ").pack()
    Button(password_not_recog_screen, text="OK", command=delete_password_not_recognised).pack()


# Designing popup for user not found


def user_not_found():
    global user_not_found_screen
    user_not_found_screen = Toplevel(login_screen)
    user_not_found_screen.title("Success")
    user_not_found_screen.geometry("150x100")
    Label(user_not_found_screen, text="User Not Found").pack()
    Button(user_not_found_screen, text="OK", command=delete_user_not_found_screen).pack()


# Deleting popups


def delete_login_success():
    login_success_screen.destroy()




def delete_password_not_recognised():
    password_not_recog_screen.destroy()



def delete_user_not_found_screen():
    user_not_found_screen.destroy()



# Designing Main(first) window


def main_account_screen():
    global main_screen
    main_screen = Tk()
    main_screen.geometry("650x400")
    main_screen.title("Account Login")
    Label(text="Select Your Choice", bg="blue", width="300", height="2", font=("Calibri", 13)).pack()
    Label(text="").pack()
    Button(text="Login", height="2", width="30", command = login).pack()
    Label(text="").pack()
    Button(text="Register", height="2", width="30", command=register).pack()
    Label(text="").pack()
    Button(text="New supervisor", height="2", width="30", command=newsupervisor).pack()

    main_screen.mainloop()




main_account_screen(
```