PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013) 100

Feet Ring Road, BSK III Stage, Bengaluru-560 085

Department of Computer Science & Engineering

Title: Student Database

Team member details:

Member 1: PES1UG21CS704 Vedant Jain

Member 2: PES1UG21CS719 Vishnu S Honnavalli

Member 3: PES1UG21CS725 Vivek YV

Abstract:

The goal of this project is to create a student database using python programming language. Both the teacher and student can login and use it in different ways. This project was created in such a way that the teacher has more options to edit the database whereas the student can only view the information the teacher has entered. In this project mainly the concepts of functions, GUI, lists, if-else, try-except were used.

Table of Contents:

Introduction:

The purpose of this project is to create a student database. It consists of 2 sections, the teacher part and student part. The teacher can login using a pre-decided password and create a new student, search for an existing student,delete a student, enter the marks of student, check average of student and also check if student passed or failed. The students can login using personal SRN only to check for an existing student, see their obtained marks, average and also to see if they passed or failed. .

Design/Implementation : Code

```
from tkinter import *
import pickle


root = Tk()
root.title("ABC Institutes")
root.geometry("640x340")



def Student():
    root.destroy()
    loot = Tk()
    loot.title("ABC Institutes")
    loot.geometry("640x340")
    login_page = PhotoImage(file="login page.png")
    login_bg = Label(loot, image=login_page)
    login_bg.place(x=0, y=0)
    tlabel = Label(loot, text='Welcome to Student login page')
```

```
tlabel.config(font=('Helvetica bold', 20))
tlabel.place(x=120, y=15)
te1 = Entry(loot, width=40, borderwidth=1)
te1.place(x=190, y=171)
l = Label(loot, text='Enter SRN:')
l.place(x=120, y=170)


def Sview():
    with open("STUDENT.DAT", "rb") as F:
      c = False
      try:
         while not c:
            R = pickle.load(F)
            for i in R:
               if te1.get() == i[0]:
                  loot.destroy()
                  t_1 = Tk()
                  t_1.title("ABC Institutes")
                  t_1.geometry("500x500")
                  student_background = PhotoImage(file = "background2.png")
                  student_background_label = Label(t_1, image=student_background)
```

```
student_background_label.place(x=0,y=0)
mylabel31 = Label(t_1, text=f"Welcome, {i[1]}")
mylabel31.config(font=('Helvetica bold', 20))
mylabel31.place(x=120, y=10)
mylabel32 = Label(t_1, text=f"Your SRN: {i[0]}")
mylabel32.place(x=50,y=80)
mylabel33 = Label(t_1, text=f"Your semester: {i[2]}st")
mylabel33.place(x=50,y=110)
phy_marks = Label(t_1, text=i[3])
phy_marks.place(x=350,y=200)
maths_marks = Label(t_1, text=i[4])
maths_marks.place(x=350,y=232)
cse_marks = Label(t_1, text=i[5])
cse_marks.place(x=350,y=264)
electrical_marks = Label(t_1, text=i[6])
electrical_marks.place(x=350,y=296)
mechanical_marks = Label(t_1, text=i[7])
mechanical_marks.place(x=350,y=329)
evs_marks = Label(t_1, text=i[8])
evs_marks.place(x=350,y=362)
```

```
                t = 3
                while t < 9:
                    if int(i[t]) < 40:
                        passmsg = Label(t_1, text='Sorry! U failed')
                        passmsg.place(x=200, y=450)
                        break
                    else:
                        t += 1
                else:
                    failmsg = Label(t_1, text='Congratulations! U passed')
                    failmsg.place(x=190, y=450)

                avgmarks = (int(i[3]) + int(i[4]) + int(i[5]) + int(i[6]) + int(i[7]) +
int(i[8])) / 6
                avgmsg = Label(t_1, text=f"Your average marks is {avgmarks}")
                avgmsg.place(x=172,y=415)

                t_1.mainloop()
                c = True
                break

            else:
```

```
                mylabel3 = Label(loot, text='Student not found')
                mylabel3.place(x=250,y=300)
        except:
            pass
    tb1 = Button(loot, text='Login', command=Sview)
    tb1.place(x=280, y=200)
    loot.mainloop()



def Teacher():
    root.destroy()
    noot = Tk()
    noot.title("ABC Institutes")
    noot.geometry("640x340")
    login_page = PhotoImage(file="login page.png")
    login_bg = Label(noot, image=login_page)
    login_bg.place(x=0, y=0)
    slabel = Label(noot, text='Welcome to Teachers login page')
    slabel.config(font=('Helvetica bold', 20))
    slabel.place(x=120, y=15)
    se1 = Entry(noot, width=40, borderwidth=1)
```

```
    se1.place(x=220, y=171)
    se2 = Label(noot, text='Enter password: ')
    se2.place(x=120, y=170)


    def login():
        if se1.get() == 'admin123':
            noot.destroy()
            addorsee = Tk()
            addorsee.title("ABC Institutes")
            addorsee.geometry("800x600")
            background_image = PhotoImage(file = "background1.png")
            background_label = Label(addorsee, image=background_image)
            background_label.place(x=0,y=0)
            mylabel12 = Label(addorsee, text='Would you like to add students or manage
existing ones')
            mylabel12.config(font=('Helvetica bold', 20))
            mylabel12.place(x=55,y=15)


            def Add():
                add_1 = Tk()
                add_1.title("ABC Institutes")
                add_1.geometry("500x600")
```

```
mylabel2 = Label(add_1, text='Create new Student')
mylabel2.config(font=('Helvetica bold', 20))
mylabel2.place(x=120,y=10)
ae1 = Entry(add_1, width=40, borderwidth=5)
ae1.place(x=200,y=100)
le1 = Label(add_1, text='Enter SRN: ')
le1.place(x=20,y=100)
ae2 = Entry(add_1, width=40, borderwidth=5)
ae2.place(x=200,y=150)
le2 = Label(add_1, text='Enter Name:')
le2.place(x=20,y=150)
ae3 = Entry(add_1, width=40, borderwidth=5)
ae3.place(x=200,y=200)
le3 = Label(add_1, text='Enter Semester:')
le3.place(x=20,y=200)
ae4 = Entry(add_1, width=40, borderwidth=5)
ae4.place(x=200,y=250)
le4 = Label(add_1, text='Enter marks for Physics:')
le4.place(x=20,y=250)
ae5 = Entry(add_1, width=40, borderwidth=5)
ae5.place(x=200,y=300)
```

```
le5 = Label(add_1, text='Enter marks for Maths:')
le5.place(x=20,y=300)
ae6 = Entry(add_1, width=40, borderwidth=5)
ae6.place(x=200,y=350)
le6 = Label(add_1, text='Enter marks for Computers:')
le6.place(x=20,y=350)
ae7 = Entry(add_1, width=40, borderwidth=5)
ae7.place(x=200,y=400)
le7 = Label(add_1, text='Enter marks for Electrical:')
le7.place(x=20,y=400)
ae8 = Entry(add_1, width=40, borderwidth=5)
ae8.place(x=200,y=450)
le8 = Label(add_1, text='Enter marks for Mechanical:')
le8.place(x=20,y=450)
ae9 = Entry(add_1, width=40, borderwidth=5)
ae9.place(x=200,y=500)
le9 = Label(add_1, text='Enter marks for EVS:')
le9.place(x=20,y=500)



def Save():
```

```
            srn = ae1.get()
            name = ae2.get()
            sem = ae3.get()
            marksp = ae4.get()
            marksm = ae5.get()
            marksc = ae6.get()
            markse = ae7.get()
            marksme = ae8.get()
            marksev = ae9.get()
            if srn.isalnum() and (
                name.isalpha() or '  ') and sem.isdigit() and marksp.isdigit() and
marksm.isdigit() and marksme.isdigit() and marksc.isdigit() and markse.isdigit() and
marksev.isdigit():
                with open("STUDENT.DAT", "ab") as F:
                    R = []
                    R.append([srn, name, sem, marksp, marksm, marksc, markse, marksme,
marksev])
                    pickle.dump(R, F)
                add_1.destroy()


            else:
```

```
            mylabel9 = Label(add_1, text='Wrong entries')
            mylabel9.place(x=220,y=550)

        ab1 = Button(add_1, text='Save', command=Save)
        ab1.place(x=240,y=575)
        add_1.mainloop()

    addb1 = Button(addorsee, text='Add new', command=Add)
    addb1.place(x=200,y=500)



    def Average():
        with open("STUDENT.DAT", "rb") as F:
            c = False
            try:
                while not c:
                    R = pickle.load(F)
                    for i in R:
                        if avg.get() == i[0]:
                            avg1 = (int(i[3]) + int(i[4]) + int(i[5]) + int(i[6]) + int(i[7]) +
int(i[8])) / 6
```

```
                    mymsg3 = Label(addorsee, text=f"Average of the student is
{avg1}")
                    mymsg3.place(x=400,y=150)
                    c = True
                    break
                else :
                    mymsg4 = Label(addorsee, text='Student not found')
                    mymsg4.place(x=400,y=150)
            except:
                pass

        mymsg = Label(addorsee, text="Enter SRN to check average of student: ")
        mymsg.place(x=120,y=120)
        avg = Entry(addorsee, width=40)
        avg.place(x=370,y=121)
        mybutton = Button(addorsee, text="Check", command=Average)
        mybutton.place(x=630,y=120)

        def Pass():
            with open("STUDENT.DAT", "rb") as F:
                c = False
                try:
```

```
while not c:
    R = pickle.load(F)
    for i in R:
        if pas.get() == i[0]:
            t = 3
            while t < 9:
                if int(i[t]) < 40:
                    break
                else:
                    t += 1
            else:
                mymsg5 = Label(addorsee, text='Student has passed!')
                mymsg5.place(x=400,y=220)
                break
            mymsg3 = Label(addorsee, text='Student has failed!')
            mymsg3.place(x=400,y=220)
            c = True
            break
    else:
        mymsg4 = Label(addorsee, text='Student not found')
        mymsg4.place(x=400,y=220)
```

```
            except:
                pass
        mymsg2 = Label(addorsee, text='Enter SRN to check  if student has passed or
failed: ')
        mymsg2.place(x=90,y=190)
        pas = Entry(addorsee, width=40)
        pas.place(x=370,y=191)
        mybutton2 = Button(addorsee, text="Check", command=Pass)
        mybutton2.place(x=630,y=190)


        def Delete():
            with open("STUDENT.DAT", "rb") as F:
                student_found = False
                try:
                    while True:
                        R = pickle.load(F)
                        for i in R:
                            if de1.get() == i[0]:
                                student_found = True
                except:
                    pass
```

```
with open("STUDENT.DAT", "rb") as F:
    l = []
    try:
        while True:
            R = pickle.load(F)
            for i in R:
                if de1.get() != i[0]:
                    l.append(i)
    except:
        pass


if student_found == True:
    with open("STUDENT.DAT", "wb") as T:
        pickle.dump(l, T)
        mymsg4 = Label(addorsee, text='Student is deleted')
        mymsg4.place(x=400, y=290)
if student_found == False:
    mymsg5 = Label(addorsee, text='Student not found')
    mymsg5.place(x=400, y=290)
```

```
        mylabel10 = Label(addorsee, text='Enter SRN of student to be deleted: ')
        mylabel10.place(x=130,y=260)
        de1 = Entry(addorsee, width=40)
        de1.place(x=370, y=261)
        addb2 = Button(addorsee, text='Delete', command=Delete)
        addb2.place(x=630,y=260)



        def AboveAvg():
            x=0
            with open("STUDENT.DAT", "rb") as F:
                try:
                    while True:
                        R = pickle.load(F)
                        avgcheckint = int(avgcheck.get())
                        for i in R:
                            avg2 = (int(i[3]) + int(i[4]) + int(i[5]) + int(i[6]) + int(i[7]) + int(i[8]))
/ 6
                            if avgcheckint <= avg2:
                                x += 1
                        mymsg3 = Label(addorsee, text=f"There are {x} students having
average more than your target.")
```

```
            mymsg3.place(x=350, y=370)
        except:
            pass



mymsg3 = Label(addorsee, text='What is your target average? ')
mymsg3.place(x=150,y=330)
avgcheck = Entry(addorsee, width=40)
avgcheck.place(x=370,y=331)
mybutton2 = Button(addorsee, text="Submit", command=AboveAvg)
mybutton2.place(x=630,y=330)



def SearchT():
    t_12 = Tk()
    t_12.title("ABC Institutes")
    t_12.geometry("1470x400")
    ycord = 0
    with open("STUDENT.DAT", "rb") as F:
        try:
```

```
            while True:
                R = pickle.load(F)
                for i in R:
                    mylabel31 = Label(t_12, text=f"Name of the student: {i[1]}, Semester: {i[2]}, Physics marks: {i[3]},"
                                                f" Maths marks: {i[4]}, CSE marks: {i[5]}, Electrical marks: {i[6]}, Mechanical marks: {i[7]},"
                                                f" EVS marks: {i[8]} \n SRN: {i[0]} ", font=("Arial", 15),)
                    mylabel31.place(x=10, y=ycord)
                    ycord += 70
            except:
                pass
        t_12.mainloop()


    addb3 = Button(addorsee, text='View All', command=SearchT)
    addb3.place(x=500,y=500)
    addorsee.mainloop()


else:
```

```
        mylabel8 = Label(noot, text='Wrong password')
        mylabel8.place(x=280,y=300)

    sb1 = Button(noot, text='Login', command=login)
    sb1.place(x=320, y=200)
    noot.mainloop()



main_page_image = PhotoImage(file="main page.png")
mylabel1 = Label(root, image=main_page_image)
mylabel1.place(x=0, y=0)
button1 = Button(root, text='Teacher', command=Teacher)
button1.place(x=450, y=170)
button2 = Button(root, text='Student', command=Student)
button2.place(x=450, y=210)
root.mainloop()
```

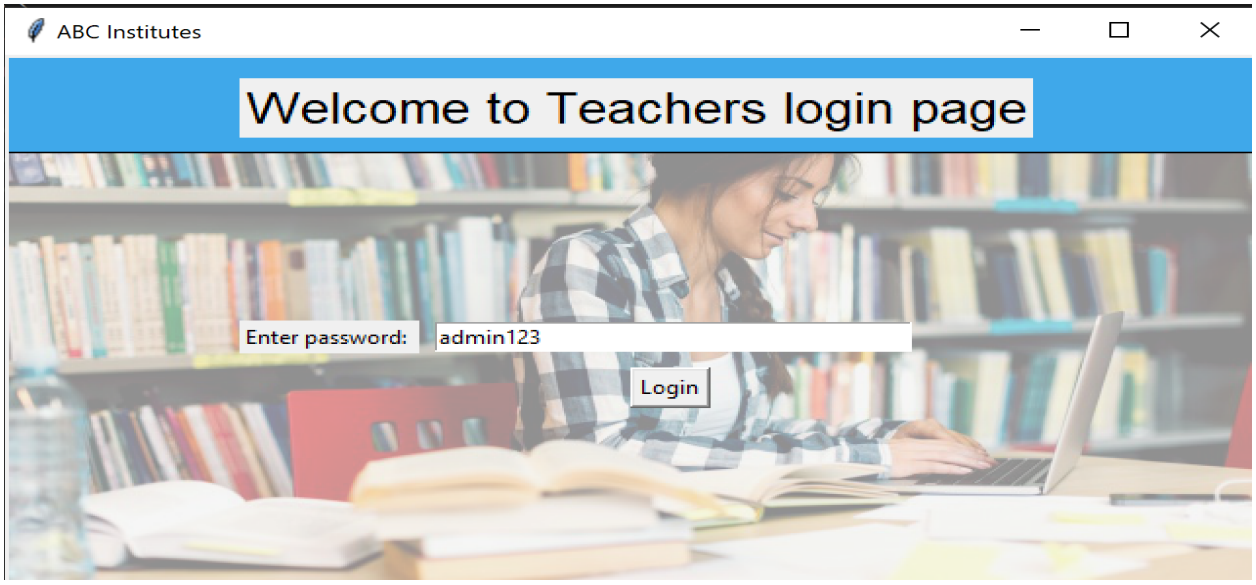
Testing:

The project begins by asking the user whether user is a teacher or a student.If the user is a teacher, he is required to enter a password which is already hardcoded.If the password is correct the teacher adding and editing page is opened where a student can be added,deleted,checked whether the student has passed or not and checking the average of student and how many students did well by comparing it to a value. Here all the lists of students can be viewed. Every value is checked and if the value does not satisfy the criteria the output is not shown instead a message is shown stating the values entered are wrong. Whereas if the user is a student then the user is required to enter his SRN and after logging in , the user can see his deails.

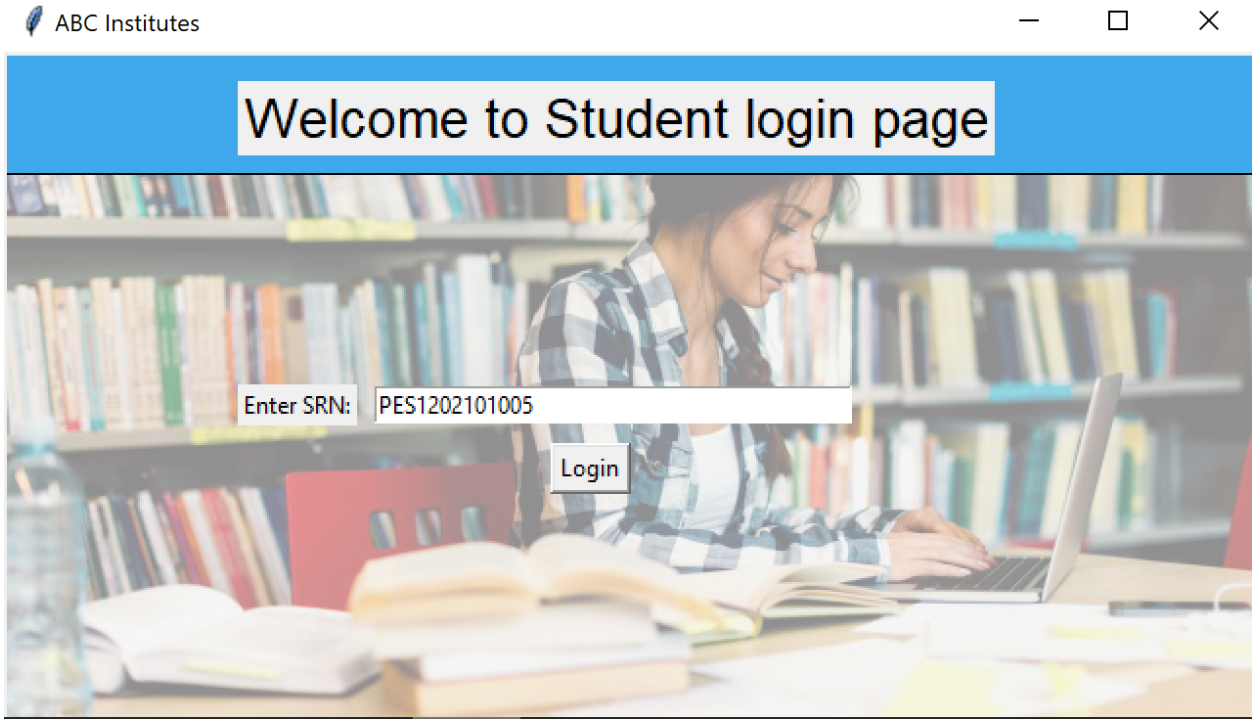
INPUT	OUTPUT
Enter name: Zakhir	Name gets displayed at the top
Check average of student: (SRN)	95.0
Check if you passed/failed:(SRN)	You Passed!!!
Enter student to be deleted:(SRN)	Student is deleted

Result and Analysis: Output snapshots with proper captions

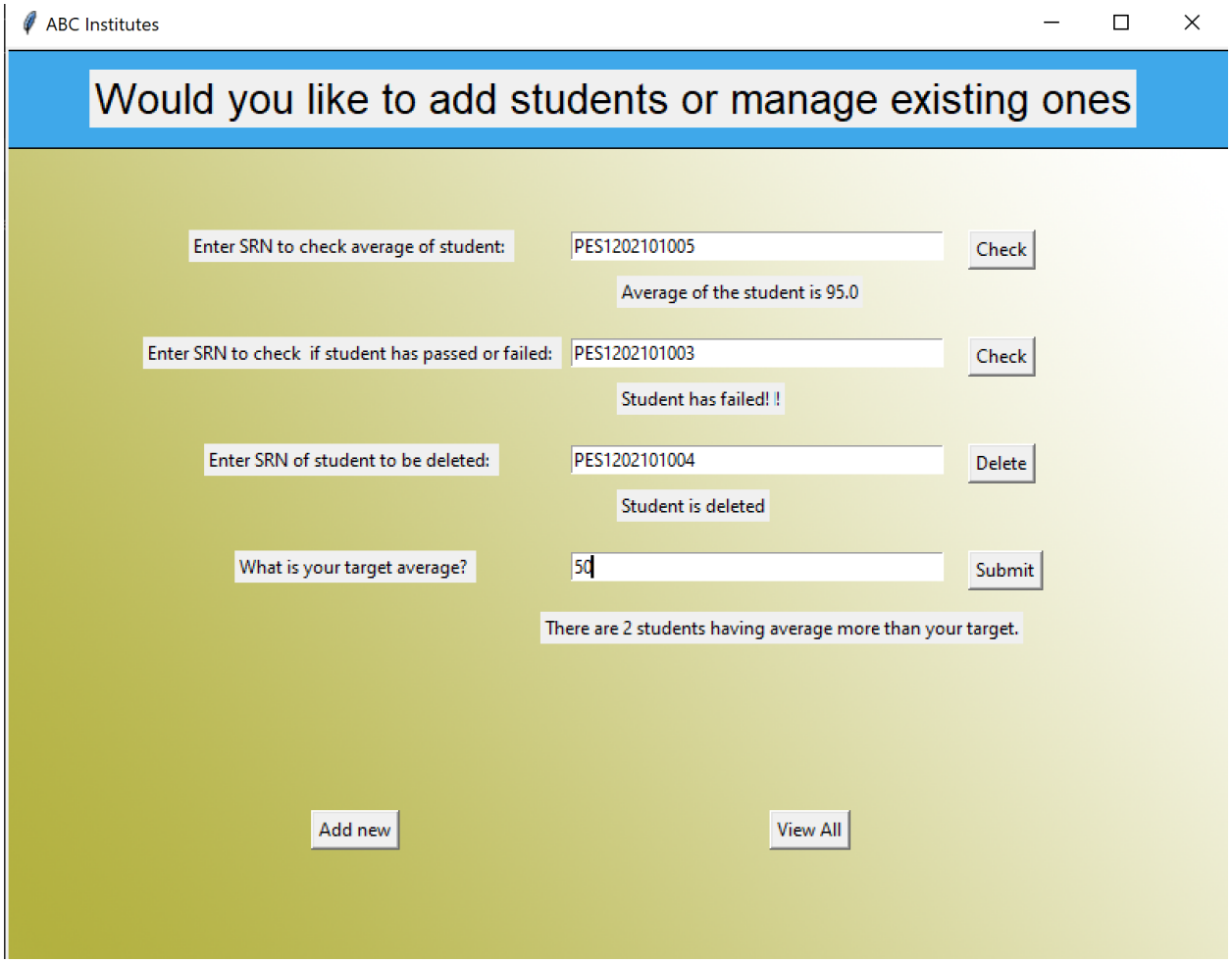
THIS IS THE MAIN LOGIN PAGE



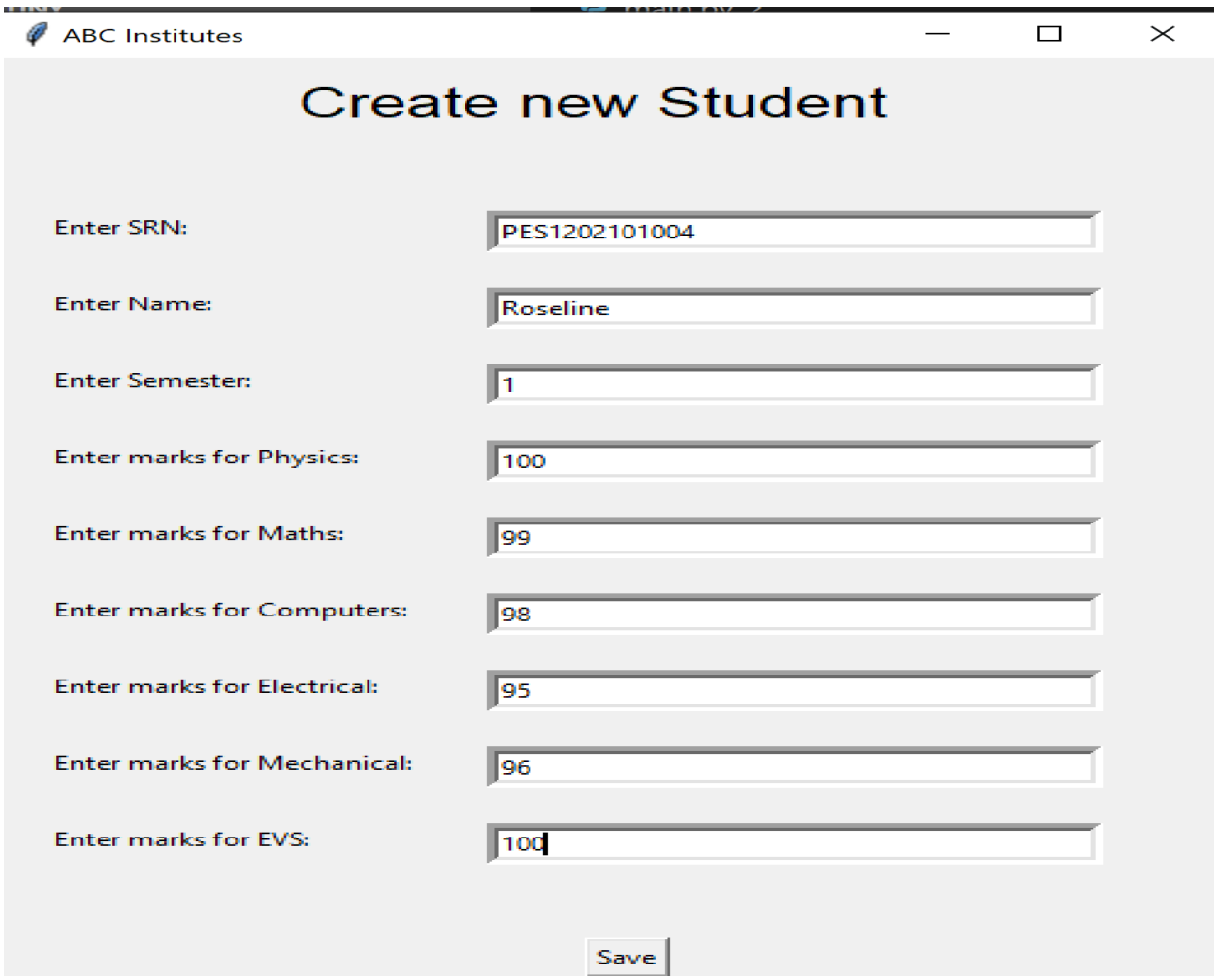
TEACHER'S LOGIN PAGE



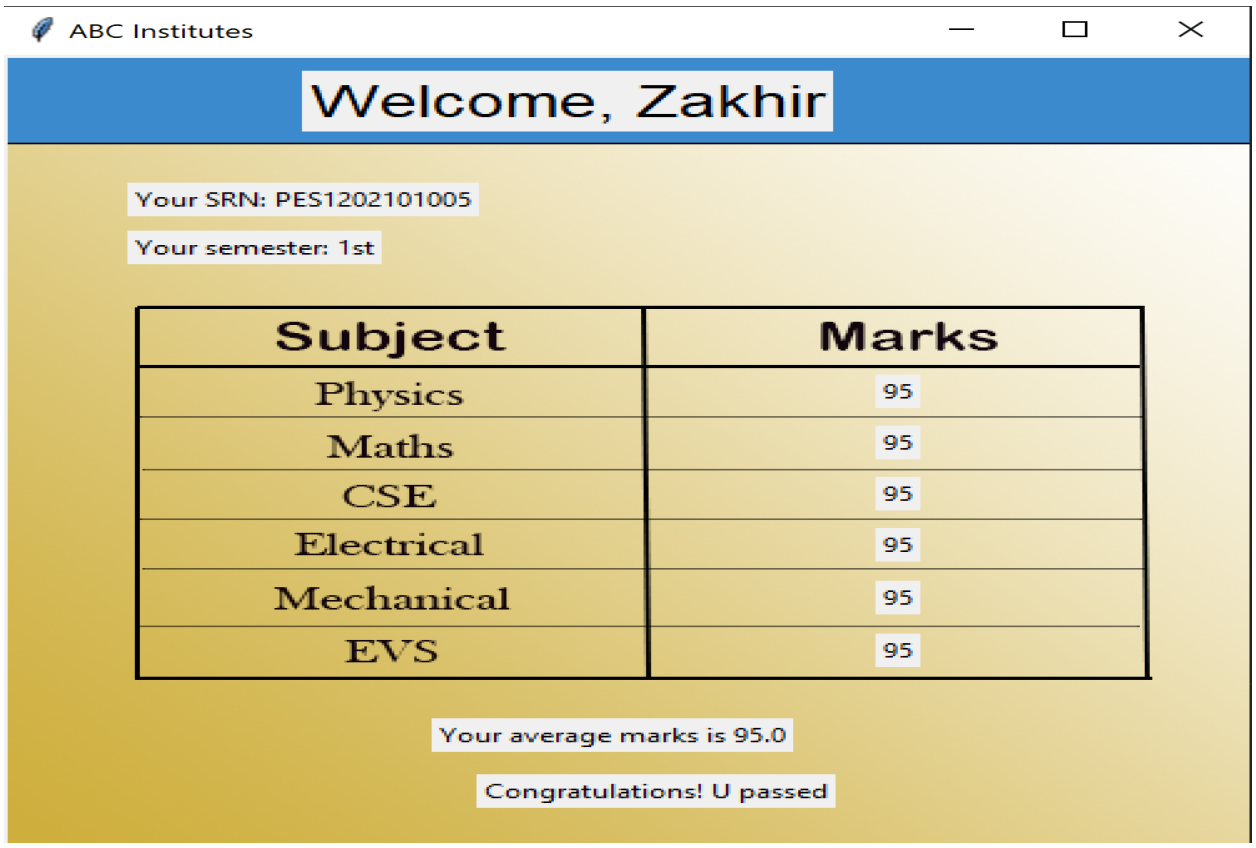
STUDENT'S LOGIN PAGE



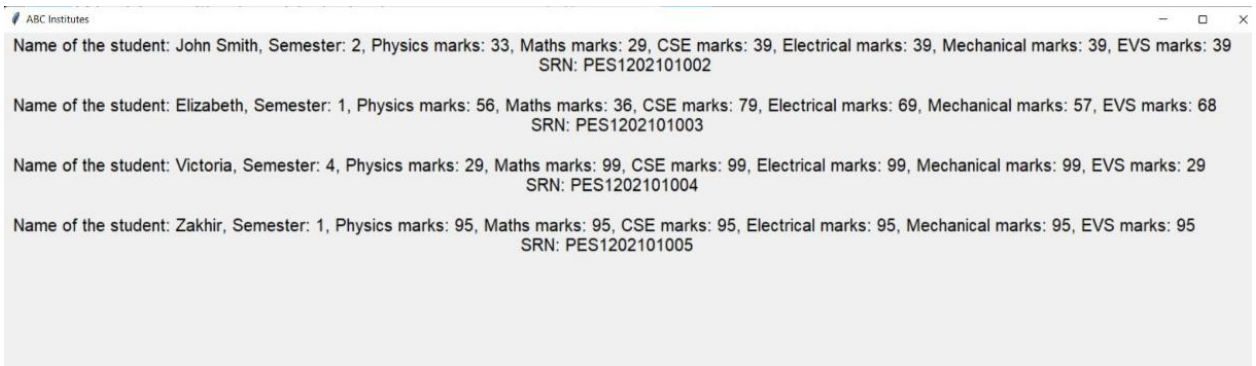
THIS IS TEACHER'S MAIN PAGE WHERE THEY CAN ADD NEW OR MANAGE EXISTING STUDENTS



CREATING A NEW STUDENT



THIS IS STUDENT'S MAIN PAGE WHERE THEY CAN ONLY SEE THE DATA ENTERED BY TEACHER



VIEWING ALL THE STUDENTS

Conclusions & future enhancements:

In the future we hope to improve this project by:

- ➔ Including back buttons to quickly go back to the previous page
- ➔ Making this into a webpage

References:

https://www.youtube.com/watch?v=YXPyB4XeYLA&t=1650s

https://github.com/yvvivek9/Python-Project

geeksforgeeks.org