**Appendix-2**

**Linked with success criteria 1**

def \_\_init\_\_(self,root):

self.root = root

self.root.title('login window')

self.root.geometry('1350x700+0+0')

self.root.config(bg='grey')

frame=Frame(self.root,bg='crimson')

frame.place(x=480,y=100,width=700,height=500)

self.Username = StringVar()

self.Password = StringVar()

self.var = StringVar()

r1 = Radiobutton(frame,variable=self.var, font=('times',25,'bold'),bg='lightyellow', text='Super Admin',value=1)

r2 = Radiobutton(frame,variable=self.var, font=('times',25,'bold'),bg='lightyellow', text='Junior Admin', value=2)

self.var.set(1)

r1.place(x=100,y=30)

r2.place(x=400,y=30)

title=Label(frame,text='Login here',font=('times',30,'bold'),justify=CENTER,bg='blue').place(x=50,y=100)

Username=Label(frame,text='Username',font=('times',20,'bold'),bg='red').place(x=60,y=190)

Password=Label(frame,text='Password',font=('times',20,'bold'),bg='lightgrey').place(x=60,y=260)

Username=Entry(frame,textvariable=self.Username,font=('times',25,'bold'),bg='lightyellow').place(x=200,y=190)

Password=Entry(frame,textvariable=self.Password,font=('times',25,'bold'),bg='lightyellow').place(x=200,y=260)

btn\_login=Button(frame,font=('times',25,'bold'),text='Submit',bg='blue',command=self.check,relief=RIDGE).place(x=290,y=330)

**Linked with success criteria 2**

def check(self):

con=mysql.connector.connect(host='localhost',username='root',password='shrey',database='rms')

cur=con.cursor()

try:

if self.Username.get()=="" or self.Password.get()=="":

messagebox.showerror("Error","Please Fill All Fields",parent=self.root)

else:

if self.var.get()=='1':

#cur.execute("select \* from login where username=? and password=? and admin='1'",(self.Username.get(),self.Password.get(),))

cur.execute("select \* from login where username=%s and password=%s and admin='1'",(self.Username.get(),self.Password.get()))

else:

cur.execute("select \* from login where username=%s and password=%s and admin='2'",(self.Username.get(),self.Password.get()))

#cur.execute("select \* from login where username=? and password=? and admin='2'",(self.Username.get(),self.Password.get(),))

row=cur.fetchone()

if row==None:

messagebox.showerror("Error","Invalid username or password",parent=self.root)

else:

#cur.execute("insert into session (time,user,admin) values(?,?,?)",(int(time.time()),self.Username.get(),self.var.get(),))

cur.execute("insert into session (time,user,admin)values(%s,%s,%s)",(int(time.time()),self.Username.get(),self.var.get()))

con.commit()

os.system("python dashboard.py")

os.\_exit(0)

except Exception as ex:

messagebox.showerror("Error",f"Error due to{str(ex)}")

**Linked with success criteria 3**

def add\_course(self):

self.new\_win=Toplevel(self.root)

self.new\_obj=CourseClass(self.new\_win)

btn\_courses=Button(M\_frame,text='Course',font=('times',15,'bold'),bg='blue',fg='white',command=self.add\_course).place(x=20,y=5,width=200,height=40)

**Linked with success criteria 4**

def add\_student(self):

self.new\_win=Toplevel(self.root)

self.new\_obj=studentClass(self.new\_win)

btn\_student=Button(M\_frame,text='Student',font=('times',15,'bold'),bg='blue',fg='white',command=self.add\_student).place(x=240,y=5,width=200,height=40)

**Linked with success criteria 5**

def add\_result(self):

self.new\_win=Toplevel(self.root)

self.new\_obj=resultClass(self.new\_win)

btn\_result=Button(M\_frame,text='Marks',font=('times',15,'bold'),bg='blue',fg='white',command=self.add\_result).place(x=460,y=5,width=200,height=40)

**Linked with success criteria 6**

def add\_report(self):

self.new\_win=Toplevel(self.root)

self.new\_obj=reportClass(self.new\_win)

btn\_view=Button(M\_frame,text='View Student Results',font=('times',15,'bold'),bg='blue',fg='white',command=self.add\_report).place(x=680,y=5,width=200,height=40)

**Linked with success criteria 7**

def add(self):

try:

if self.var\_course.get() and self.var\_charges.get()=="":

messagebox.showerror("Error","all fields are required",parent=self.root)

else:

cur.execute("select \* from course where name=%s",(self.var\_course.get(),))

row=cur.fetchone()

if row!=None:

messagebox.showerror("Error","Course name already present",parent=self.root)

else:

cur.execute("insert into course (name,duration,charges,description) values(%s,%s,%s,%s)",(self.var\_course.get(),self.var\_duration.get(),self.var\_charges.get(),

self.txt\_description.get("1.0",END)

))

con.commit()

messagebox.showinfo("Success","Record added successfully",parent=self.root)

self.show()

except Exception as ex:

messagebox.showerror("Error",f"Error due to{str(ex)}")

**Linked with success criteria 8**

def delete(self):

con=mysql.connector.connect(host='localhost',username='root',password='shrey',database='rms')

cur=con.cursor()

try:

if self.var\_course.get() and self.var\_charges.get()=="":

messagebox.showerror("Error","All fields are required",parent=self.root)

else:

cur.execute("select \* from course where cid=%s",(self.var\_course.get(),))

row=cur.fetchone()

if row==None:

messagebox.showerror("Error","Please select the course from the list first",parent=self.root)

else:

op=messagebox.askyesno("Confirm","Do you really want to delete this course?",parent=self.root)

if op==True:

cur.execute("delete from course where cid=%s",(self.var\_course.get(),))

con.commit()

messagebox.showinfo("Delete","Course deleted successfully",parent=self.root)

self.clear

except Exception as ex:

messagebox.showerror("Error",f"Error due to {str(ex)}")

**Linked with success criteria 9**

if authen[3] == "1":

self.btn\_delete=Button(self.root,text='Delete',font=('times',15,'bold'),bg='red',fg='white',command=self.delete)

self.btn\_delete.place(x=390,y=400,width=110,height=40)

**Linked with success criteria 10**

def update(self):

try:

if self.var\_course.get() and self.var\_charges.get()=="":

messagebox.showerror("Error","All fields are required",parent=self.root)

else:

print(self.var\_course.get())

cur.execute("select \* from course where cid = %s",(self.var\_course.get(),))

row=cur.fetchone()

#print(row)

if row==None:

messagebox.showerror("Error","Select course from list",parent=self.root)

else:

cur.execute("update course set duration=%s, charges=%s, description=%s where cid=%s",(

self.var\_duration.get(),

self.var\_charges.get(),

self.txt\_description.get("1.0",END),

self.var\_course.get()

))

con.commit()

messagebox.showinfo("Success","Record Updated successfully",parent=self.root)

self.show()

except Exception as ex:

messagebox.showerror("Error",f"Error due to{str(ex)}")

**Linked with success criteria 11**

def clear(self):

self.show()

self.var\_course.set("")

self.var\_duration.set("")

self.var\_charges.set("")

self.var\_search.set("")

self.txt\_description.delete('1.0',END)

self.txt\_courseName.config(state=NORMAL)

**Linked with success criteria 12**

def search(self):

try:

cur.execute(f"select \* from course where name LIKE '%{self.var\_search.get()}%'")

rows=cur.fetchall()

self.CourseTable.delete(\*self.CourseTable.get\_children())

for row in rows:

self.CourseTable.insert('',END,values=row)

except Exception as ex:

messagebox.showerror("Error",f"Error due to {str(ex)}")

**Linked with success criteria 13**

def logout(self):

op=messagebox.askyesno("Confirm","Do you really want to logout?",parent=self.root)

if op==True:

cur.execute("DELETE FROM session")

con.commit()

self.root.destroy()

os.system("python register.py")

**Linked with success criteria 14**

def exit(self):

op=messagebox.askyesno("Confirm","Do you really want to exit?",parent=self.root)

if op==True:

self.root.destroy()