GUI for VEHICLE PARKING MANAGEMENT SYSTEM

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By:

S.no.	*Name*	*Roll No.*	*Registration no.*
1.	*T.AJAY TEJA*	*34*	*11904002*
2.	*M.PAVAN KALYAN REDDY*	*27*	*11904409*

Courses Code: INT213

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab (India)

FINAL REPORT FOR THE VEHICLE PARKING MANAGEMENT SYSTEM

INTRODUCTION:

Parking management system for managing the records of the incoming and outgoingvehicles in an parking house.It's an easy for Admin to retrieve the data if the vehicle has been visited through number he can get that data.

Now days in many public places such as malls, multiplex system, hospitals, offices,market areas there is a crucial problem of vehicle parking. The vehicle parking area hasmany lanes/slots for car parking. So to park a vehicle one has to look for all the lanes.Moreover this involves a lot of manual labour and investment. Instead of vehicle caughtin towing the vehicle can park on safe and security with low cost.Parking control system has been generated in such a way that it is filled with manysecure devices such as, parking control gates, toll gates, time and attendance machine,car counting system etc. These features are hereby very necessary nowadays to secureyour car and also to evaluate the fee structure for every vehicles entry and exitThe objective of this project is to build a Vehicle Parking management system thatenables the time management and control of vehicles using number plate recognition.The system that will track the entry and exit of cars, maintain a listing of cars within theparking lot, and determine if the parking lot is full or not. It will determine the cost of pervehicle according to their time consumption.

OBJECTIVE:

We can park our vehicle in our own slot by paying.

-->Because of that there is no towing problems.

-->And our vehicle has been parked as a secure condition.

-->There is no risk for vehicle owner for parking the car.

-->In case of any damages and problem of vehicle that will claim by

parkingmanagement.

As the world is facing many threads daily, robberies are done easily with no trackto trace, bomb blasts occur with the use of vehicle, so if a proper system isadopted each and every record can be saved and anyone can be track easilytherefore mainly is to make a better and fast software, most important user-friendly.

-->Maintain records in short time of period.

-->Determines the parking area is full or not.

SOURCE CODE:

```
from tkinter import*
import sqlite3
import tkinter.messagebox
root=Tk()
root.geometry("650x320")
root.title("LOG in")
regno = StringVar()
password = StringVar()
cur_user=""
con=sqlite3.connect("login.db")
root.configure(background="#519657")
def new():
  root.destroy()
  root13 = Tk()
  root13.geometry('800x800')
  canvas= Canvas(root13,width = 800, height = 800,bg="darksalmon" )
```

```
canvas.pack(expand = YES, fill = BOTH)
root13.title("Registration Form")

name=StringVar()
username=StringVar()
a=StringVar()
b=StringVar()
email=StringVar()
password=StringVar()
mobile=StringVar()
b=StringVar()
c=StringVar()
conn=sqlite3.connect("login.db")

def database():
    nm=name.get()
    usrnm=username.get()
    sts=a.get()
    gnr=b.get()
    mail=email.get()
    pswd=password.get()
    mbl=mobile.get()
    dpt=c.get()
    conn.execute("create table if not exists register(Name text,Username text,
Status text,Gender text,Emailid text,Password text,Mobile text,Department text);")
    conn.execute("insert into register
```

```
values(?,?,?,?,?,?,?,?)",(nm,usrnm,sts,gnr,mail,pswd,mbl,dpt))
      conn.commit()
      p=conn.execute("select * from register")
      for i in p:
         print("Username",i[1])
         print("Password",i[5])

      tkinter.messagebox.showinfo("Parking Management","Registered successfully")
      root13.destroy()
   label_0 = Label(root13, text="Registration form for Parking",width=25,font=("",
"24", "bold"), foreground="white",bg="brown3")
   label_0.place(x=80,y=40)
   label_1 = Label(root13, text="Name",width=20,font=("bold",
10),bg="brown2",fg="white")
   label_1.place(x=80,y=130)
   entry_1 = Entry(root13,textvariable=name,bg="brown2",fg="white")
   entry_1.place(x=300,y=130)
   label_2 = Label(root13, text="Registration no.",width=20,font=("bold",
10),bg="brown2",fg="white")
   label_2.place(x=80,y=180)
   entry_2 = Entry(root13,textvariable=username,bg="brown2",fg="white")
   entry_2.place(x=300,y=180)
   label_3 = Label(root13, text="Status",width=20,font=("bold",
10),bg="brown2",fg="white")
   label_3.place(x=80,y=230)
   list1=['FACULTY','STUDENT']
```

```
  a=StringVar()
  droplist=OptionMenu(root13,a, *list1)
  droplist.config(width=40,height=1,bg="brown2",fg="black")
  a.set('Select your Status')
  droplist.place(x=300,y=230)
 label_4 = Label(root13, text="Gender",width=20,font=("bold",
10),bg="brown2",fg="white")
  label_4.place(x=80,y=280)
  list2=['MALE','FEMALE']
  b=StringVar()
  droplist=OptionMenu(root13,b, *list2)
  droplist.config(width=40,height=1,bg="brown2",fg="black")
  b.set('Select your Gender')
  droplist.place(x=300,y=280)

  label_5 = Label(root13, text="Email-id",width=20,font=("bold",
10),bg="brown2",fg="white")
  label_5.place(x=80,y=330)

  entry_5 = Entry(root13,textvariable=email,bg="brown2",fg="white")
  entry_5.place(x=300,y=330)

  label_6 = Label(root13, text="Password",width=20,font=("bold",
10),bg="brown2",fg="white")
  label_6.place(x=80,y=380)
  bullet="\u2022"
```

```
  entry_6 =
Entry(root13,textvariable=password,show=bullet,bg="brown2",fg="white")

  entry_6.place(x=300,y=380)

  label_7 = Label(root13, text="Confirm Password",width=20,font=("bold",
10),bg="brown2",fg="white")

  label_7.place(x=80,y=430)

  entry_7 = Entry(root13,show=bullet,bg="brown2",fg="white")

  entry_7.place(x=300,y=430)

  label_8 = Label(root13, text="Mobile Number",width=20,font=("bold",
10),bg="brown2",fg="white")

  label_8.place(x=80,y=480)

  entry_8 = Entry(root13,textvariable=mobile,bg="brown2",fg="white")

  entry_8.place(x=300,y=480)

  label_9 = Label(root13, text="Department",width=20,font=("bold",
10),bg="brown2",fg="white")

  label_9.place(x=80,y=530)

  list3 = ['AGRICULTURE','BIOENGINEERING AND
BIOSCIENCES','BUSINESS','ENGINEERING','COMPUTER APPLICATIONS','FASHION
DESIGN','HOTEL MANAGEMENT','FINE ARTS','LAW','PHARMACY'];

  c=StringVar()

  droplist=OptionMenu(root13,c, *list3)

  droplist.config(width=40,height=1,bg="brown2",fg="black")

  c.set('Select your Department')

  droplist.place(x=300,y=530)

  Button(root13, text='Submit',width=
20,bg='brown',fg='black',command=database).place(x=240,y=600)

  root13.mainloop()
```

```
def request(curr_user):
  root12 = Tk()
  root12.title("Parking Availability")
  root12.geometry("640x400")
  db = sqlite3.connect("login.db")
  root12.configure(background='#90CAF9')
  reg_no = StringVar()
  block=IntVar()
  fare=StringVar()
  currr=StringVar()
  currr.set(curr_user)
  fare.set("-")
  drop=StringVar(root12)
  print("current user::",curr_user)
  drop.set("Choose Time Period(in Hrs.)")
  db.execute("create table if not exists parking(regno varchar(20),block int,fare
int);")
  def reserve():
     print("in Reserve")
     if(fare.get() is '-'):
        tkinter.messagebox.showinfo("Warning","Calculate the fare first by Selecting
time Period")
     elif(reg_no.get() == ""):
        tkinter.messagebox.showinfo("Warning","Please Provide a Registration
Number")
     else:
```

```
        _Reg_no = reg_no.get()
        _Block = block.get()
        _Fare = fare.get()
        print(_Reg_no)
        db.execute("insert into parking values(?,?,?)",(_Reg_no,_Block,_Fare))
        db.commit()
        if(_Block == 0):
          confirm = "You Can Park Vehice at Block 29 with Fare: "+ _Fare
          tkinter.messagebox.showinfo("Confirmation",confirm)
        else:
          confirm = "You Can Park Vehice at Block 30 with Fare: "+ _Fare
          tkinter.messagebox.showinfo("Confirmation",confirm)
    db.close()
  def calculate_fare():
    _drop = drop.get()
    if(_drop == "Choose Time Period(in Hrs.)"):
      tkinter.messagebox.showinfo("Warning","Calculate the fare first by Selecting
time Period")
      fare.set("-")
    elif(_drop == "3"):
      fare.set(3*20)
    elif(_drop == "6"):
      fare.set(6*20)
    elif(_drop == "8"):
      fare.set(8*20)
```

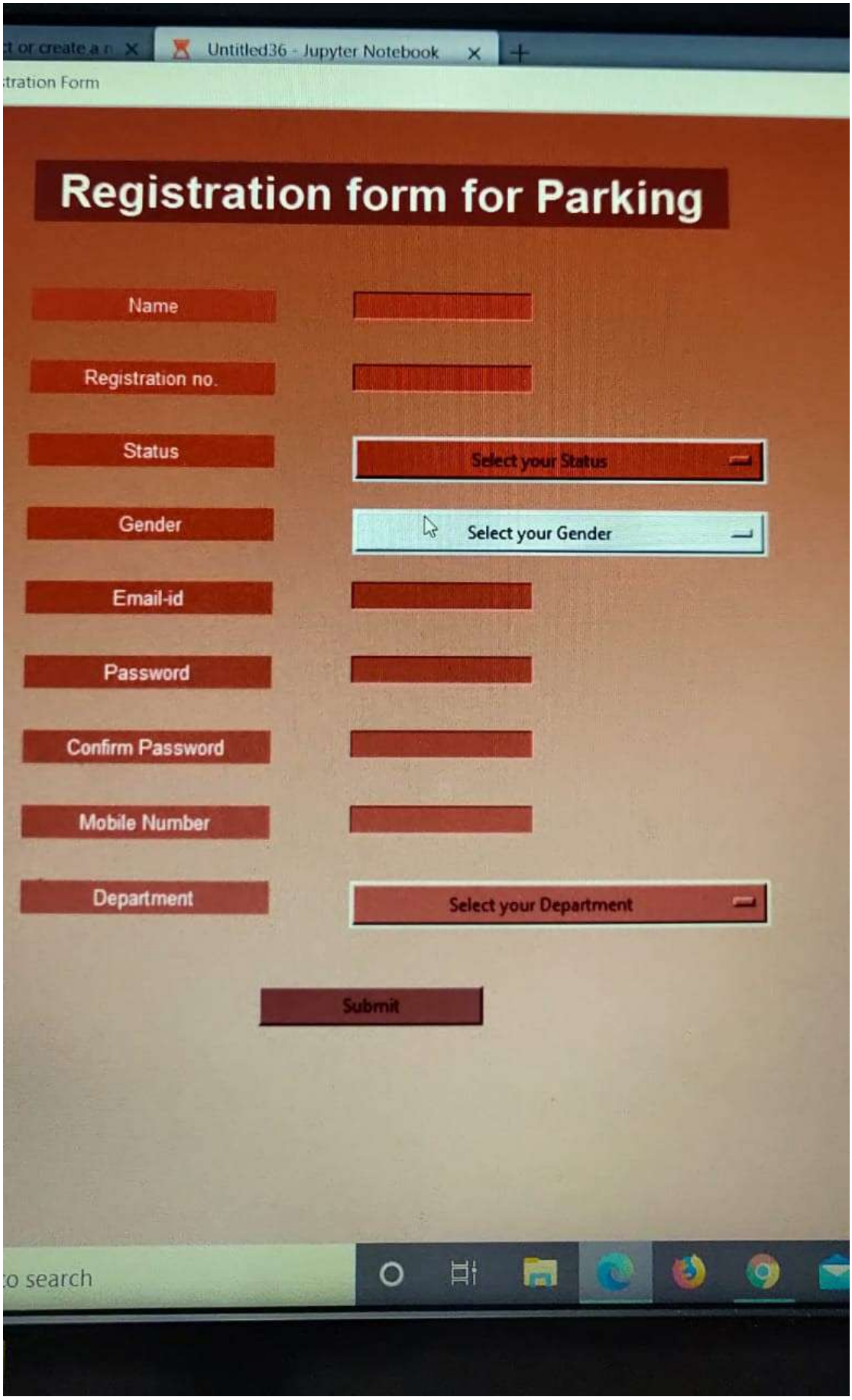
```
  # GUI DESIGN FOR PARKING REQUEST

  title=Label(root12,text="Slot Booking").place(x=100,y=130)

  canvas=Canvas(root12,height=300,width=600,relief="raised",bg="#
42A5F5").place(x=15,y=30)

  Label(root12,text="Parking Request",bg="#42A5F5",fg="white",font=("Courier",
24)).place(x=200,y=30)

  l1=Label(root12,text="Reg.No:",fg="white",bg="#42A5F5").place(x=70,y=80)

  reg_entry=Entry(root12,bd=2,textvariable=reg_no).place(x=130,y=75)

  Label(root12,text="Block:",bg="#42A5F5",fg="white").place(x=70,y=110)

  blk29=Radiobutton(root12,text="Block 29",variable=block,value=0,bg="#
42A5F5",fg="white").place(x=130,y=110)

  blk30=Radiobutton(root12,text="Block 30",variable=block,value=1,bg="#
42A5F5",fg="white").place(x=220,y=110)

  Label(root12,text="Time Period:",bg="#42A5F5",fg="white").place(x=70,y=150)

  Timeprd=OptionMenu(root12,drop,"3","6","8").place(x=160,y=150)

  Fare=Label(root12,bg="#42A5F5",fg="white",textvariable=fare).place(x=160,y=
190)

  f=Label(root12,text="Fare(in Rs):",bg="#42A5F5",fg="white").place(x=70,y=190)

  Label(root12,text="Current User::",bg="#42A5F5",fg="white").place(x=425,y=40)

  currentuser=Label(root12,textvariable=currr,bg="#
42A5F5",fg="white",font=("Courier", 15,"bold")).place(x=520,y=40)

  # calculate fare button

  FareCal=Button(root12,text="Calculate
Fare",bg="red",command=calculate_fare).place(x=210,y=190)

  # booking slot button for parking

  park=Button(root12,text="Reserve Parking
Slot",bg="red",command=reserve).place(x=80,y=230)
```

```
  root12.mainloop()
def punch():
  name=regno.get()
  pword=password.get()
  if name == "" and pword == "":
    tkinter.messagebox.showinfo( "Parking Management", "Login UNSuccessful \n
Enter your Name and Password")
  elif name=="":
      tkinter.messagebox.showinfo( "Parking Management", "Login UNSuccessful \n
Enter your Reg No.")
  elif pword =="":
      tkinter.messagebox.showinfo( "Parking Management", "Login UNSuccessful \n
Enter your Password")

  else:
    con.execute("create table if not exists Login1(regno varchar(20),password
varchar(20))")
    con.execute("insert into Login1 values(?,?)",(name,pword))
    currentuser=name
    con.commit()
    d = con.execute("select * from register")
    count=0
    for i in d:
      if name==i[1] and pword==i[5]:
        count=count+1
        cur_user=i[0]
```

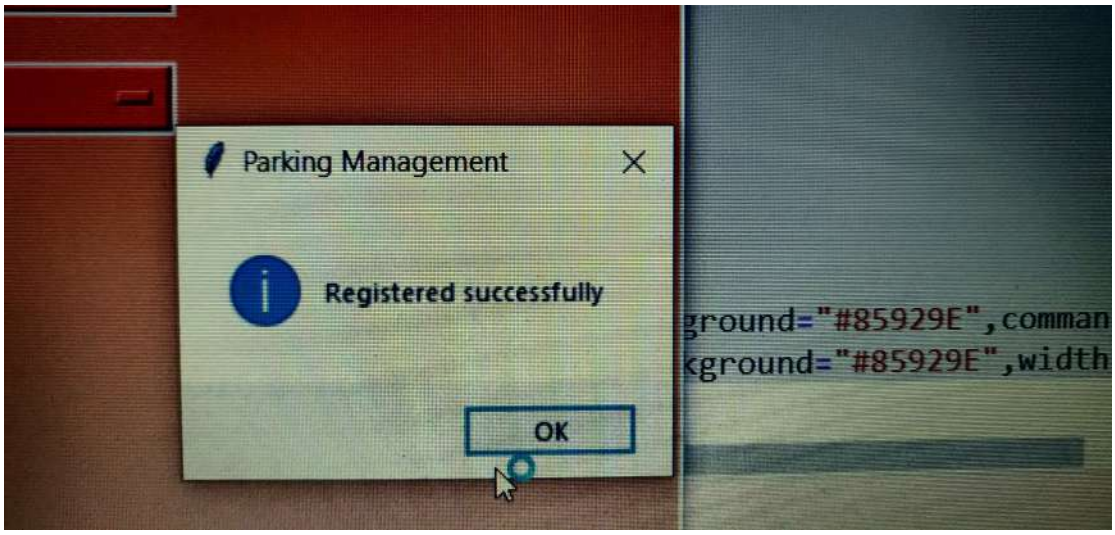
```
    if count!=0:
      tkinter.messagebox.showinfo( "Parking Management", "Login Successful")
      root.destroy()
      request(cur_user)
    else:
      tkinter.messagebox.showinfo( "Parking Management", "Login UnSuccessful")
      regno.set("")
      password.set("")


labelfont = ('Broadway', 20, 'bold italic')
e4=Label(root,text="Parking Management
System",bg="#b2fab4",font=labelfont).place(x=170,y=50)
c1=Canvas(root,relief="raised",border=2,width=400,height=
200,bg="#b2fab4").place(x=120,y=100)
l1=Label(c1,text="Registration No.",bg="#b2fab4",font = "Verdana 8 bold
italic").place(x=190,y=130)
e1=Entry(c1,bd=2,textvariable=regno).place(x=300,y=130)
l2=Label(c1,text="Password",bg="#b2fab4",font = "Verdana 10 bold italic").place(x=
190,y=160)
bullet="\u2022"
e2=Entry(c1,bd=2,textvariable=password,show=bullet).place(x=300,y=160)
l3=Label(c1,text="Or",bg="#b2fab4",font="Elephant 8").place(x=350,y=230)
b1=Button(c1,text="LOGIN",bg="#7DCEA0" ,font="Verdana 8
bold",activebackground="#85929E",command=punch,width=15).place(x=300,y=200)
b2=Button(c1,text="Sign Up",bg="#7DCEA0",font="Verdana 8
bold",activebackground="#85929E",width=15,command=new).place(x=300,y=250)
root.mainloop()
```

GUI SCREENSHOTS:

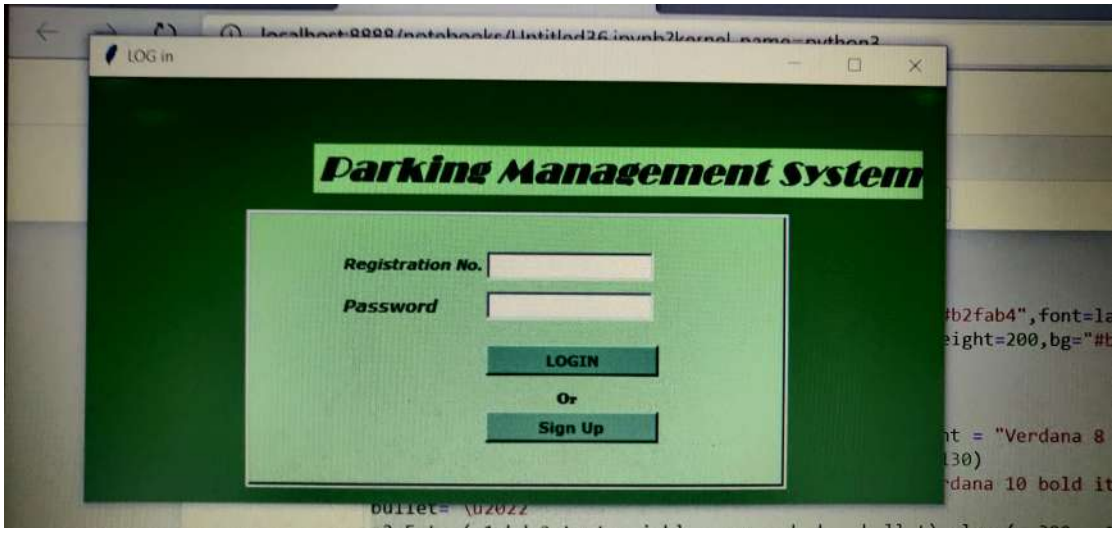
REGISTRATION FORM: Here we can register for the parking by providing necessary details.


Untitled36 - Jupyter Notebook
Registration Form
Registration form for Parking
Name
Registration no.
Status
Select your Status
Gender
Select your Gender
Email-id
Password
Confirm Password
Mobile Number
Department
Select your Department
Submit
search

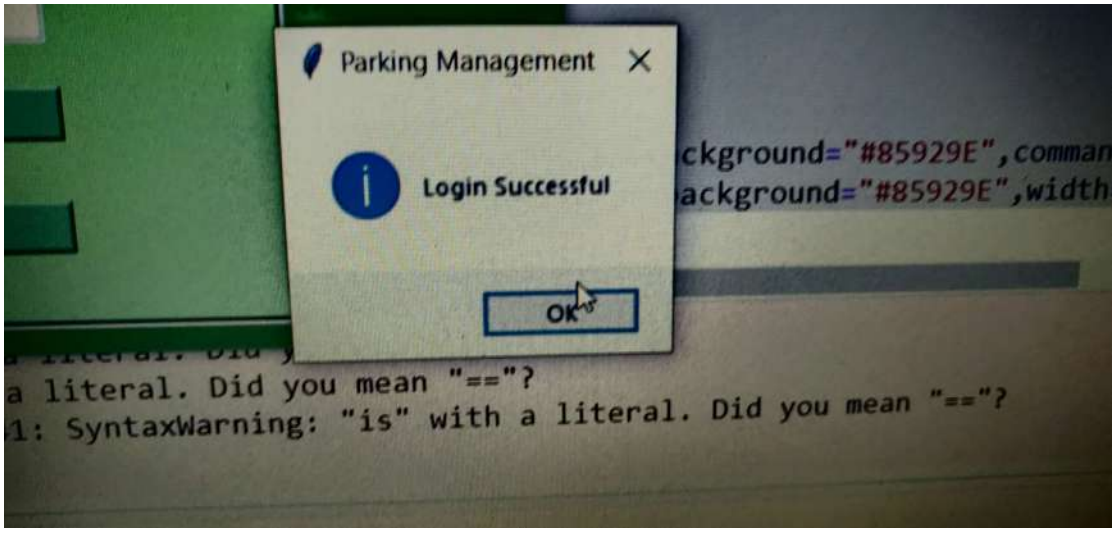
Now it shows ur registration succed or not:



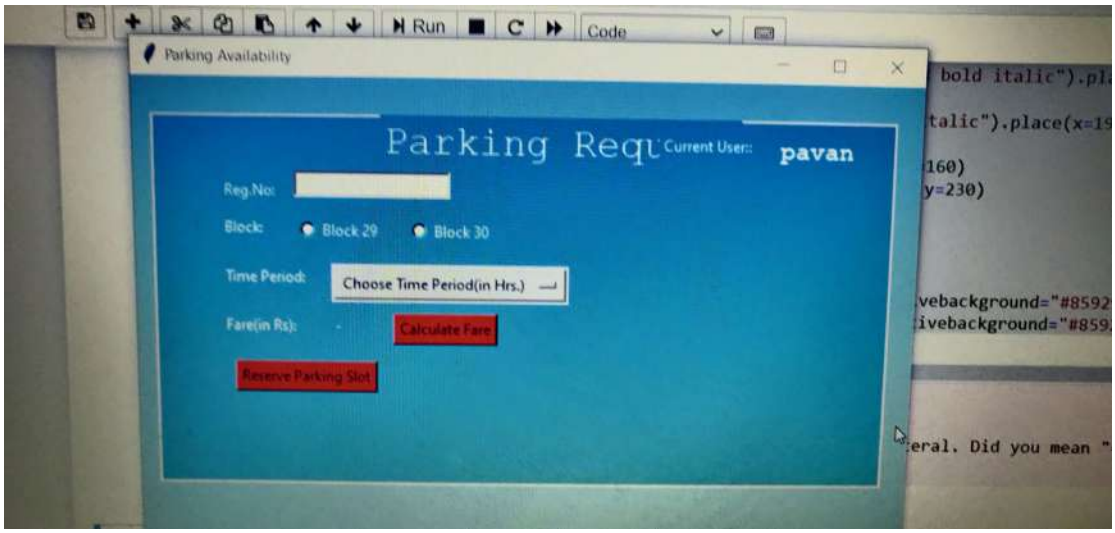
LOGIN PAGE:



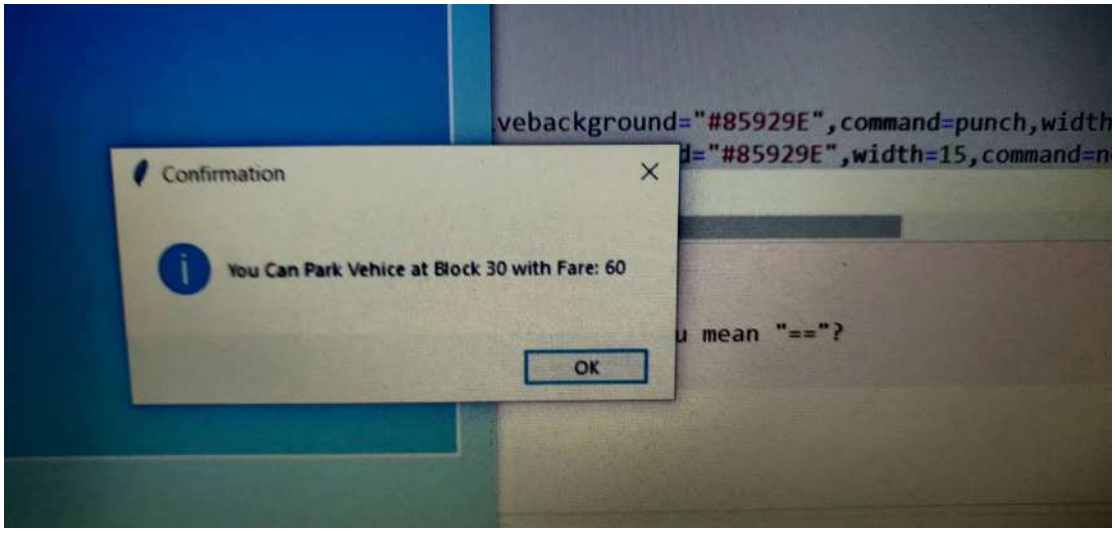
NOW IT SHOWS THE LOGGING IN SUCCEDED OR NOT:



PARKING REQUEST:



IT SHOWS IN WHICH PLACE THE VEHICLE TO BE PARKED:



CONCLUSION:

vehicle parking management system improves the existing system since we are in computerized world. With this new system is mandatory, it enables the user of the system (client, employee, System administrator) to reserve a parking lot online and this reduces the wasting of time of the clients looking for where to park, increase the safety of the property since the parking lot is numbering.

REFERENCES:

http://www.asuresoftware.com/asure-space/workspace-manager/parking-spacemanagement-system/car-parking-management-system-overview

http://www.delmatic.com/systems/parkingmanagement-systems