A black sign with white text

Description automatically generated

**INTELLIGENT SEAT ALLOCATION SYSTEM**

**GROUP PROJECT**

**BY**

|  |  |  |
| --- | --- | --- |
| Name | Roll no | Reg No |
| Shubham Kumar Singh | 5 | 11805253 |
| Abhishek Singh | 6 | 11804984 |
| Abhishek Kumar Gupta | 7 | 11804985 |
| Shashank Sharma | 8 | 11803923 |

**Section:** K18FG

**Submitted to:** AmandeepKaur

**Department of Intelligent Systems**

**School of Computer Science Engineering**

**Lovely Professional University, Jalandhar**

Introduction

The main aim of the project is to provide a comfortable seating to the people. As we know different aged people needs to be seated comfortably in order to have a good journey without facing any issues.

As we know the physically handicapped people or the elder people should be given the first priority of seating so that they don’t face any issues in their journey. And the next should be the young aged people who face less issues than the elder people. In present there are no such kind of systems which provide the seating based on age. Therefore it’s very important to provide a seating based on age so that people are seated comfortably and travel happily.

Objectives

The main goal of the project is to provide a perfect seating to the people based on their age. The physically handicapped people and the elder people are given the first and foremost seating which is then followed by middle aged people and then the young people. The intelligent seat allocation is done by a GUI application where customers enter their details and are given a perfect seating plan so that they don’t face any issue.

Outcome

We have successfully created an intelligent seat allocation system that allocates the seats to the user in a perfect way according to the information entered by them.

Pictorial Flow of Project

User has to enter the total no. of passengers to be seated which has a capacity of 18 seats.

Then the name of the passenger and their respective seat numbers are printed.

Then user has to enter the names of all the passengers

Then user must specify the no. of people aged greater than 60, 30-60 and then less than 30 years.

Work Division

Roll no 6 and 5 made the code for all the logic of the program.

Roll no 7 and 8 made the report and layout of the front page of gui.

Code

from tkinter import \* #Importing modules

from PIL import ImageTk,Image

from tkinter import messagebox

def seatallocation():

global seat

seat = Toplevel(info)

seat.title("Seat Allocation")

#seat.configure(bg = "yellow")

#win.iconbitmap("ap.ico")

global width

global height

width = seat.winfo\_screenwidth()

height = seat.winfo\_screenheight()

seat.geometry(f'{width}x{height}')

canvas = Canvas(seat,width = width,height = height)

image = ImageTk.PhotoImage(Image.open("pic.jpg"))

canvas.create\_image(0,0,anchor = "nw",image = image)

canvas.place(relx = 0,rely = 0)

#================================================Passenger Of Age above 60==============================================================================

for i in range(len(namelist1)):

namelist1[i] = namelist1[i].get()

print(namelist1[i])

Label(seat,text = "Name",bg = 'black',fg = 'white',width = 8,font =('comic sans ms','20'),anchor = "w").place(relx = 0.01,rely = 0.1)

Label(seat,text = "Allocated Seat",bg = 'black',fg = 'white',width = 12,font =('comic sans ms','20'),anchor = "w").place(relx = 0.18,rely = 0.1)

for i in range(len(namelist1)):

i += 2

Label(seat,text = namelist1[i-2],bg = 'green',fg = 'white',width = 15,font =('comic sans ms','20'),anchor = "w").place(relx = 0.01,rely = (i\*0.1))

Label(seat,text = "Seat No."+ str(i-1),bg = 'green',fg = 'white',width = 8,font =('comic sans ms','20'),anchor = "w").place(relx = 0.18,rely = (i\*0.1))

#=======================================================Passenger of age 30 to 60===================================================================================

for i in range(len(namelist2)):

namelist2[i] = namelist2[i].get()

print(namelist2[i])

Label(seat,text = "Name",bg = 'black',fg = 'white',width = 8,font =('comic sans ms','20'),anchor = "w").place(relx = 0.35,rely = 0.1)

Label(seat,text = "Allocated Seat",bg = 'black',fg = 'white',width = 12,font =('comic sans ms','20'),anchor = "w").place(relx = 0.52,rely = 0.1)

for i in range(len(namelist2)):

i += 2

Label(seat,text = namelist2[i-2],bg = 'green',fg = 'white',width = 15,font =('comic sans ms','20'),anchor = "w").place(relx = 0.35,rely = (i\*0.1))

Label(seat,text = "Seat No."+ str(len(namelist1)+i-1),bg = 'green',fg = 'white',width = 8,font =('comic sans ms','20'),anchor = "w").place(relx = 0.52,rely = (i\*0.1))

#======================================================Passenger OF Age Below30==================================================================

Label(seat,text = "Name",bg = 'black',fg = 'white',width = 15,font =('comic sans ms','20'),anchor = "w").place(relx = 0.69,rely = 0.1)

Label(seat,text = "Allocated Seat",bg = 'black',fg = 'white',width = 12,font =('comic sans ms','20'),anchor = "w").place(relx = 0.86,rely = 0.1)

for i in range(len(namelist3)):

namelist3[i] = namelist3[i].get()

print(namelist3[i])

for i in range(len(namelist3)):

i += 2

Label(seat,text = namelist3[i-2],bg = 'green',fg = 'white',width = 15,font =('comic sans ms','20'),anchor = "w").place(relx = 0.69,rely = (i\*0.1))

Label(seat,text = "Seat No."+ str(len(namelist2)+len(namelist1)+i-1),bg = 'green',fg = 'white',width = 8,font =('comic sans ms','20'),anchor = "w").place(relx = 0.88,rely = (i\*0.1))

Button(seat,text = " EXIT ",command = lambda:destroy(),font =('comic sans ms','20')).place(relx = 0.45,rely = 0.8)

#=====================================================Page Destroy==============================================================

def destroy():

seat.destroy()

info.destroy()

win.destroy()

seat.mainloop()

def landingpage():

global win

win = Tk()

win.title("Seat Allocation System")

win.configure(bg = "yellow")

#win.iconbitmap("ap.ico")

global width

global height

width = win.winfo\_screenwidth()

height = win.winfo\_screenheight()

win.geometry(f'{width}x{height}')

canvas = Canvas(win,width = width,height = height)

image = ImageTk.PhotoImage(Image.open("pic.jpg"))

canvas.create\_image(0,0,anchor = "nw",image = image)

canvas.place(relx = 0,rely = 0)

label = Label(win,text = "Enter number of passengers:",font =('comic sans ms','20'),bg = "black",fg = "white",anchor = "w",width = 25)

label.place(relx = '0.2',rely = '0.2')

label = Label(win,text = "Worry Less, Travel More",font =('Times new roman','35'),fg = "black",anchor = "w",width = 19)

label.place(relx = '0.38',rely = '0.8')

global entry

passengers = IntVar()

passengers.set(0)

entry = Entry(win,text = passengers,font =('comic sans ms','20'),width = 5)

entry.place(relx = '0.47',rely = '0.2')

button = Button(win,text = "OK",bg = 'black',fg = 'white',font =('comic sans ms','25'),command = lambda:get\_count())

button.place(relx = '0.45',rely = '0.29')

def get\_count():

if(int(entry.get())<=0 or int(entry.get())>18):

messagebox.showinfo("Warning","Invalid number of passengers.")

else:

getlist = [int(0) for i in range(3)]

print(getlist)

for \_ in range(3):

getlist[\_] = IntVar()

getlist[\_].set(0)

l = ['greater than 60:','Between 30 and 60:','Below 30:']

label1 = Label(win,text = "Enter number of passengers Of age "+l[\_],font =('comic sans ms','20'),bg = "green",fg = "white",anchor = "w",width = 45)

label1.place(relx = '0.13',rely = (\_\*0.1)+0.4)

entry1 = Entry(win,text = getlist[\_],font =('comic sans ms','20'),width = 5)

entry1.place(relx = '0.61',rely = (\_\*0.1)+0.4)

Button(win,text = " OK ",command = lambda:get\_data(),font =('comic sans ms','20')).place(relx = 0.45,rely = 0.7)

def get\_data():

for \_ in range(3):

getlist[\_] = int(getlist[\_].get())

global namelist1,namelist2,namelist3

namelist1 = [0 for i in range(int(getlist[0]))]

namelist2 = [0 for i in range(int(getlist[1]))]

namelist3 = [0 for i in range(int(getlist[2]))]

if(sum(getlist) != int(passengers.get())):

messagebox.showinfo("Warning","Invalid number of passengers.")

else:

global info

info = Toplevel(win)

info.title("Name\_Page")

info.configure(bg = "yellow")

#win.iconbitmap("ap.ico")

global width

global height

width = info.winfo\_screenwidth()

height = info.winfo\_screenheight()

info.geometry(f'{width}x{height}')

canvas = Canvas(info,width = width,height = height)

image = ImageTk.PhotoImage(Image.open("pic.jpg"))

canvas.create\_image(0,0,anchor = "nw",image = image)

canvas.place(relx = 0,rely = 0)

#global f = 0

Label(info,text = "Enter names of passengers of age above 60",font =('comic sans ms','15'),bg = "black",fg = "white",anchor = "w",width = 35).place(relx = 0.01,rely = 0.01)

for i in range(getlist[0]):

Name\_label = Label(info,text = "Enter name of passenger "+str(i+1),font =('comic sans ms','12'),bg = "green",fg = "white",anchor = "w",width = 20)

Name\_label.place(relx = '0.008',rely = ((i\*0.05)+0.06))

namelist1[i] = StringVar()

namelist1[i].set("Enter name here")

nameentry = Entry(info,text = namelist1[i],font =('comic sans ms','12'),width = 15)

nameentry.place(relx = '0.15',rely = ((i\*0.05)+0.06))

f = ((i\*0.05)+0.06)

if(getlist[1] == 0 and getlist[2] != 0):

b = Button(info,text = "OK",font =('comic sans ms','12'),command = lambda:BelowThirty())

b.place(relx = 0.16,rely = f + 0.1)

elif(getlist[1] == 0 and getlist[2] == 0):

b = Button(info,text = "OK",font =('comic sans ms','12'),command = lambda:seatallocation())

b.place(relx = 0.16,rely = f + 0.1)

else:

b = Button(info,text = "OK",font =('comic sans ms','12'),command = lambda:ThirtyTOSixty())

b.place(relx = 0.16,rely = f + 0.1)

def ThirtyTOSixty():

l1 = Label(info,text = "Enter names of passengers of age 30 to 60:",font =('comic sans ms','15'),bg = "black",fg = "white",anchor = "w",width = 35)

l1.place(relx = 0.31,rely = 0.01)

for i in range(getlist[1]):

Name\_label = Label(info,text = "Enter name of passenger "+str(i+1),font =('comic sans ms','12'),bg = "green",fg = "white",anchor = "w",width = 20)

Name\_label.place(relx = '0.35',rely = ((i\*0.05)+0.06))

namelist2[i] = StringVar()

namelist2[i].set("Enter name here")

nameentry = Entry(info,text = namelist2[i],font =('comic sans ms','12'),width = 15)

nameentry.place(relx = '0.5',rely = ((i\*0.05)+0.06))

f = ((i\*0.05)+0.06)

print("getlist2 is:",getlist[2])

print(type(getlist[2]))

if(getlist[2] == 0):

b = Button(info,text = "OK",font =('comic sans ms','12'),command = lambda:seatallocation())

b.place(relx = 0.46,rely = f + 0.1)

else:

b = Button(info,text = "OK",font =('comic sans ms','12'),command = lambda:BelowThirty())

b.place(relx = 0.46,rely = f + 0.1)

def BelowThirty():

Label(info,text = "Enter names of passengers of age below 30",font =('comic sans ms','15'),bg = "black",fg = "white",anchor = "w",width = 34).place(relx = 0.6,rely = 0.01)

for i in range(getlist[2]):

Name\_label = Label(info,text = "Enter name of passenger "+str(i+1),font =('comic sans ms','12'),bg = "green",fg = "white",anchor = "w",width = 20)

Name\_label.place(relx = '0.7',rely = ((i\*0.05)+0.06))

namelist3[i] = StringVar()

namelist3[i].set("Enter name here")

nameentry = Entry(info,text = namelist3[i],font =('comic sans ms','12'),width = 15)

nameentry.place(relx = '0.84',rely = ((i\*0.05)+0.06))

Button(info,text = "OK",font =('comic sans ms','15'),command = lambda:seatallocation()).place(relx = 0.75,rely = f+0.1)

info.mainloop()

win.mainloop()

landingpage()

Implementation







