Best Journey For You

Name : Kukadiya Arjun	**Enroll:**92100133060
Tilala Yash	92100133088

Subject : Discrete Mathematics and Graph Theory

Guided By

Prof.Foram Rajdev

✞ About Concept :

In this project we use the concept of graph theory dijkstra’s algorithm for finding the shortest path .

✞ About Project :

In our project customer have the option that how it is plan for his/her journey. Is it go through direct flight or it will go through some station.

If that is select the direct flight then customer enter the two city’s name and according if the flight is available then the route will be displayed.

If customer select the more than one station than customer give the source name ,destination name and stop point. According to the path will be displayed.

Once the path was displayed the if customer want to book the ticket than it can.

✞ Software :

1.Python

2.HTML

✞ **Code**:

Main code:

```
from tkinter import *
import tkinter
from PIL import Image,ImageTk
from tkinter import font
import data
import tkinter.messagebox
from tkinter import ttk
import gmplot
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from webdriver_manager.chrome import ChromeDriverManager
from selenium.webdriver.chrome.options import Options
from selenium.webdriver.common.by import By
import time

root = Tk()
formap = Frame(root)
forlogin = Frame(root)
frame1=Frame(root)
book = Frame(root)
map1=Frame(root)
map2=Frame(root)
root.geometry("1200x600")
root.title("Best Joureny for You")
font1 = font.Font(family='Georgia', size='18', weight='bold')
api1=data.apikeys1
api2=data.apikeys2
# city_list=data.datalist
city_list=data.datalist1
# print(city_list)
myFont = font.Font(family='Helvetica')

def choise():
    label1 = Label(frame1,image=photo2)
    label1.pack()
    btn2 = Button(frame1, text="Direct Flight",font=myFont,
command=to_map,fg="red",relief=SUNKEN).place(x=500,y=275)
    btn3 = Button(frame1, text="More than onle station",font=myFont,
command=to_map1,fg="red",relief=SUNKEN).place(x=500,y=310)
    # btn2.pack(side=RIGHT,padx=100,pady=50)
    # btn4 = Button(frame1, text="Next",
command=to_map,font=myFont,fg="red",relief=SUNKEN).place(x=500,y=345)
    # btn5 = Button(frame1, text="Back",
command=to_map,fg="red",relief=SUNKEN).place(x=550,y=525)
    frame1.pack(fill='both', expand=1)
    forlogin.pack_forget()
    formap.pack_forget()
    map2.pack_forget()

def ticket():
    label1 = Label(book,image=photo3)
    label1.pack()
    my_list=data.datelist
    my_list1=data.monthlist
    my_list2=["Yes","No"]
    name1 = Label(book, text="From",font="Times 16 italic
bold").place(x=800,y=400)
    name2 = Label(book, text="TO",font="Times 16 italic
bold").place(x=800,y=430)
    name3 = Label(book, text="Return",font="Times 16 italic
bold").place(x=800,y=460)
    # name1.pack(side=RIGHT,padx=100,pady=50)
    nameentry3 = Entry(book,font="Times 16
italic",textvariable=namevalue3,width=20).place(x=855,y=400)
    nameentry4 = Entry(book,font="Times 16 italic",
textvariable=namevalue4,width=20).place(x=855,y=430)
    cmb1 = ttk.Combobox(book,value=my_list,width=10).place(x=775,y=490)
    cmb2 = ttk.Combobox(book,value=my_list1,width=15).place(x=875,y=490)
    cmb3 = ttk.Combobox(book,value=my_list2,width=15).place(x=875,y=460)
    btn5 = Button(book, text="Payment",font="Times 16 italic
bold",fg="blue",relief=SUNKEN,bg="gray").place(x=800,y=520)
    book.pack(fill='both', expand=1)
    map2.pack_forget()


def get_val():

    # print(f"{namevalue1.get(),namevalue2.get()}")
    a=data.datacheck(namevalue1.get(),namevalue2.get())
    if a==False:
        tkinter.messagebox.showinfo("Best Journey for you","Sorry we can't
Find the Location")

def to_map():
    # image1 = Image.open("C:\map1.jpg")
    # # photo = ImageTk.PhotoImage(image1)
    label1 = Label(formap,image=photo)
    label1.pack()
    can_widget = Canvas(formap,width=1200,height=600)
    can_widget.pack()
    can_widget.create_line(0,0,500,200)
    name1 = Label(formap, text="From",font="Times 16 italic
bold").place(x=600,y=465)
```

```
    name2 = Label(formap, text="TO",font="Times 16 italic
bold").place(x=600,y=505)
    nameentry1 = Entry(formap,font="Times 16 italic",
textvariable=namevalue1,width=40).place(x=655,y=465)
    nameentry2 = Entry(formap,font="Times 16 italic",
textvariable=namevalue2,width=40).place(x=655,y=505)
    # nameentry.pack()
    btn1 = Button(formap,text="See the Route",font="Times 16 italic
bold",fg="green",relief=SUNKEN,bg="gray",command=for_plot).place(x=750, y=540)
    btn5 = Button(formap, text="Back", command=choise,font="Times 16 italic
bold",fg="red",relief=SUNKEN,bg="gray").place(x=780,y=590)
    formap.pack(fill='both', expand=1)
    frame1.pack_forget()

def to_map1():
    # image1 = Image.open("C:\map1.jpg")
    # # photo = ImageTk.PhotoImage(image1)
    label1 = Label(map1,image=photo)
    label1.pack()
    can_widget = Canvas(map1,width=1200,height=600)
    can_widget.pack()
    can_widget.create_line(0,0,500,200)
    name1 = Label(map1, text="Stop",font="Times 16 italic
bold").place(x=600,y=465)
    name2 = Label(map1, text="To",font="Times 16 italic
bold").place(x=600,y=505)
    name3 = Label(map1, text="From",font="Times 16 italic
bold").place(x=600,y=435)
    nameentry1 = Entry(map1,font="Times 16 italic",
textvariable=namevalue1,width=40).place(x=655,y=465)
    nameentry2 = Entry(map1,font="Times 16 italic",
textvariable=namevalue2,width=40).place(x=655,y=505)
    nameentry2 = Entry(map1,font="Times 16 italic",
textvariable=namevalue5,width=40).place(x=655,y=435)
    # nameentry.pack()
    btn1 = Button(map1,text="See Route",font="Times 16 italic
bold",fg="green",relief=SUNKEN,bg="gray",command=for_plot1).place(x=750,
y=540)
    btn5 = Button(map1, text="Back", command=choise,font="Times 16 italic
bold",fg="red",relief=SUNKEN,bg="gray").place(x=780,y=590)
    map1.pack(fill='both', expand=1)
    frame1.pack_forget()


def to_login():
   btn2 = Button(forlogin, text="Start Your
Journey",font=myFont,command=choise,fg="red",relief=SUNKEN).place(x=550,y=525)
   forlogin.pack(fill='both', expand=1)
```

```
    formap.pack_forget()

def for_plot1():
    city2=namevalue1.get()
    city3=namevalue2.get()
    city1=namevalue5.get()
    if city1=="" or city2=="" or city3=="":
        tkinter.messagebox.showinfo("Best Journey for you","Please Enter the 
city")
    elif city1.lower() in city_list and city2.lower() in city_list:
        to_map2()
        lst1=[]
        lst2=[]
        for i in range(0,len(city_list)):
            if city1.lower() == city_list[i]:
                lst1.append(api1[i])
                lst2.append(api2[i])
        for i in range(0,len(city_list)):
            if city2.lower() == city_list[i]:
                lst1.append(api1[i])
                lst2.append(api2[i])
        for i in range(0,len(city_list)):
            if city3.lower() == city_list[i]:
                lst1.append(api1[i])
                lst2.append(api2[i])
        print(lst1)
        print(lst2)
        gm = gmplot.GoogleMapPlotter(lst1[0],lst2[2],8)
        gm.scatter(lst1,lst2,'#ff000',size=50,marker=True)
        gm.plot(lst1,lst2,'blue',edge_width=2.5)
        gm.draw("map1.html")
        options = Options()
        browser = webdriver.Chrome(executable_path ='C:\SEM-
3\PYTHON\Project\ONLINE FILLING THE 
FORM\chromedriver_win32\chromedriver.exe',chrome_options=options)
        browser.maximize_window()
        browser.get("file:///C:/SEM-3/DMGT/Practice/map1.html")
        time.sleep(40)


        # browser.implicitly_wait(30)
        # time.sleep(2)

    else:
        tkinter.messagebox.showinfo("Best Journey for you","Sorry we can't 
Find the Location")
```

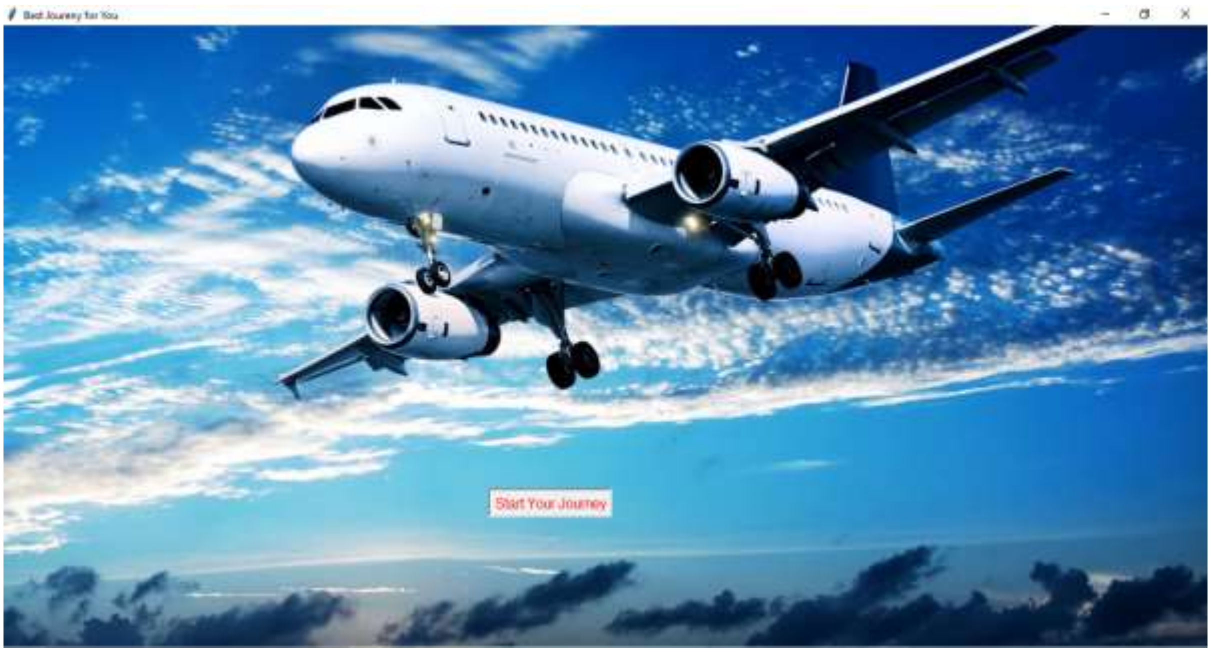
```
def for_plot():
    city1=namevalue1.get()
    city2=namevalue2.get()
    if city1=="" or city2=="" or city1==city2:
        tkinter.messagebox.showinfo("Best Journey for you","Please Enter the
city")
    elif city1.lower() in city_list and city2.lower() in city_list:
        to_map2()
        options = Options()
        browser = webdriver.Chrome(executable_path ='C:\SEM-
3\PYTHON\Project\ONLINE FILLING THE
FORM\chromedriver_win32\chromedriver.exe',chrome_options=options)
        browser.maximize_window()
        browser.get("https://maps.google.co.in/")
        # browser.implicitly_wait(30)
        # time.sleep(2)

        conti = browser.find_element(By.XPATH,'//*[@id="hArJGc"]')
        conti.click()
        time.sleep(5)

        conti1 = browser.find_element(By.XPATH,'//*[@id="omnibox-
directions"]/div/div[2]/div/div/div/div[6]/button/img')
        conti1.click()
        time.sleep(3)
        # //*[@id="sb_ifc50"]/input
        first_city = browser.find_element(By.XPATH,
"//html/body/div[3]/div[9]/div[3]/div[1]/div[2]/div/div[3]/div[1]/div[1]/div[2
]/div[1]/div/input")
        first_city.send_keys(city1)

        last_city = browser.find_element(By.XPATH,
"/html/body/div[3]/div[9]/div[3]/div[1]/div[2]/div/div[3]/div[1]/div[2]/div[2]
/div[1]/div/input")
        last_city.send_keys(city2)

        search = browser.find_element(By.XPATH,'//*[@id="directions-searchbox-
1"]/button[1]')
        search.click()
        time.sleep(20)



    else:
        tkinter.messagebox.showinfo("Best Journey for you","Sorry we can't
Find the Location")
```

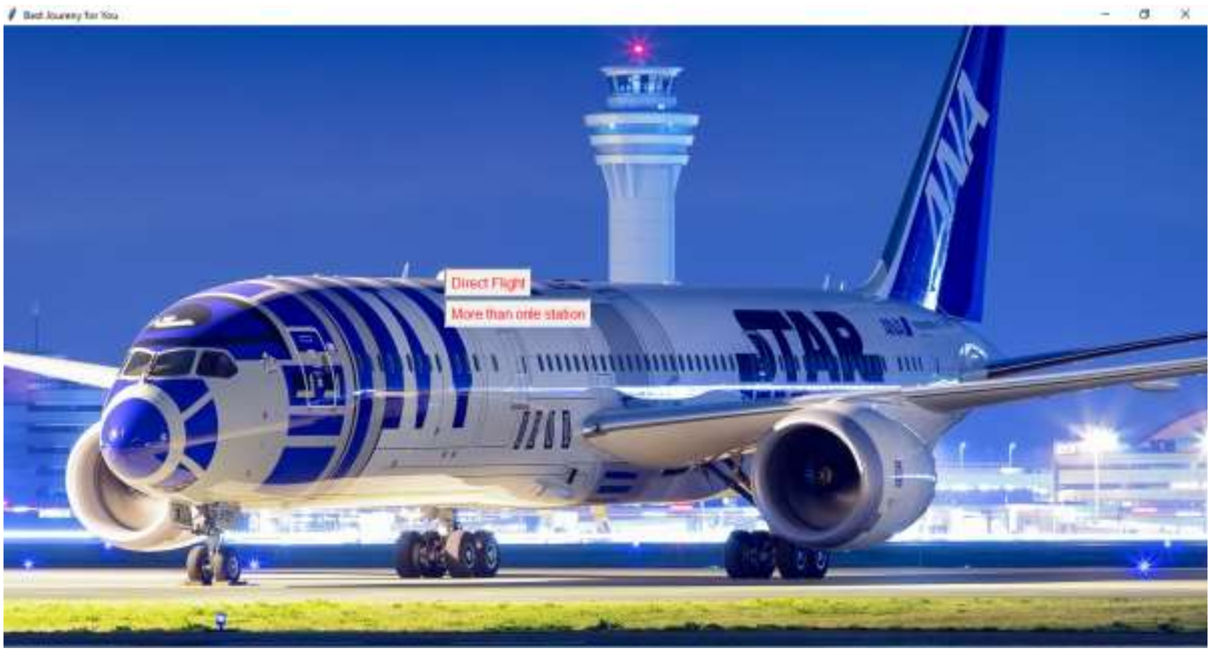
```
def to_map2():
    # image1 = Image.open("C:\map1.jpg")
    # # photo = ImageTk.PhotoImage(image1)
    label1 = Label(map2,image=photo)
    label1.pack()
    can_widget = Canvas(map2,width=1200,height=600)
    can_widget.pack()
    can_widget.create_line(0,0,500,200)
    # nameentry.pack()
    btn1 = Button(map2,text="Ticket Book",font="Times 16 italic
bold",fg="green",relief=SUNKEN,bg="gray",command=ticket).place(x=750, y=540)
    btn5 = Button(map2, text="Back", command=choise,font="Times 16 italic
bold",fg="red",relief=SUNKEN,bg="gray").place(x=780,y=590)
    map2.pack(fill='both', expand=1)
    formap.pack_forget()
    map1.pack_forget()

image1 = Image.open("C:\map1.jpg")
photo = ImageTk.PhotoImage(image1)
image2 = Image.open("C:\image1.jpg")
photo1 = ImageTk.PhotoImage(image2)
image3 = Image.open("C:\SEM-3\DMGT\Practice\image3.jpg")
photo2 = ImageTk.PhotoImage(image3)
image4 = Image.open("C:\SEM-3\DMGT\Practice\image4.jpg")
photo3 = ImageTk.PhotoImage(image4)
namevalue1 = StringVar()
namevalue2 = StringVar()
namevalue3 = StringVar()
namevalue4 = StringVar()
namevalue5 = StringVar()
namevalue6 = StringVar()
# label1 = Label(formap, text="Hey ", foreground="green3",image=photo)
# label1 = Label(formap,image=photo)
# label1.pack()
label2 = Label(forlogin,foreground="blue",image=photo1)
label2.pack()
to_login()
# to_map1()
# choise()
# ticket()
root.mainloop()
```

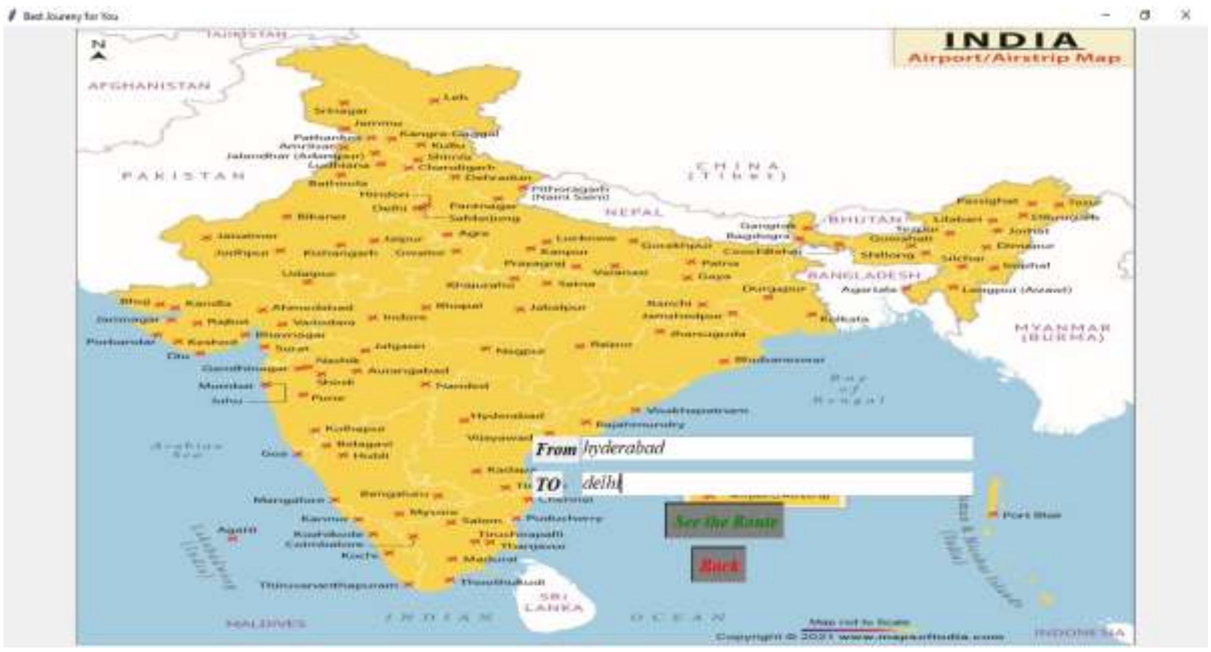
Data Code:

```
datalist=["Bhopal","Indore","Leh","Srinagar",
          "Jammu","Kangra-Gaggal","Kullu",
          "Shimla","Ludhiana","Chandigrah","Dehradun",
          "Bathinda","Hindon","Delhi","Patnagar","Safdarjung",
          "Bikaner","Agra","Jaipur","Jaisalmer","Jodhpur","Lucknow",
          "Kishangarh Gwalior","Kanpur","Gorakhpur","CoochBehar",
          "Udaipur","Prayagraj","Varanasi","Patna","Gaya",
          "Khajuraho","Satna","Durgapur","Bhuj","Kandla",
          "Jabalpur","Jamnagar","Rajkot","Vadodara","Porbandar","Keshod",
          "Bhavnagar","Surat","Jalgaon","Nashik","Gandhinagar",
          "Aurangabad","Mumbai","Shirdi","Nagpur","Nanded",
          "Juhu","Pune","Div","Kolhapur","Hyderabad","Belagavi",
          "Vijayawada","Belagavi","Hubli","Goa","KAdapa",
          "Bengaluru","Tirupati","Mysore","Chennai","Kannur",
          "Puducherry","Salem","Kozhikode","Coimbatore","Kochi",
          "Tiruchiralli","Thanjavur","Madurai",
          "Thiruvananthapura","Thoothukudi","Kadapa",
          "Raipur","Nagpur","Jabalpur","Satna","Patana",
          "Ranchi","Jamshedpur","Lucknow","Ranchi","Jharsuguda",
          "Visakhapatnam","Rajahmundry","Bhuvneswer","Kolkata",
          "Gangtok","Bagdogra","Guwahati","Shillong","Silchar",
          "Agartala","Agartala","Lengpui","Imphal",
          "Dimapur","Jorhat","Lilabari","Passighat",
          "Dibrugrah","Tezu","Ahmedabad"
          ]
datalist1=[]
for s in datalist:
    datalist1.append(s.lower())


datelist=[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31]

monthlist=["January","February","March","April","May","june","July","Augest","
September","Octomber","Nivember","December"]

# print(datalist1)


#           "Juhu","Pune","Div","Kolhapur","Hyderabad","Belagavi",
#           "Vijayawada","Belagavi","Hubli","Goa","KAdapa",
#           "Bengaluru","Tirupati","Mysore","Chennai","Kannur",

apikeys1=[23.291124,22.728090,34.143233,34.002317,32.681156,32.165169,31.82540
7,31.085943,30.850658,30.668108,30.195274,30.271674,28.705237,29.033694,28.583
698,
          28.055768,27.162085,26.829112,26.870836,26.264585,26.761921,26.589638,26.409782,26.746580,26.329915,24.639582,25.430424,25.448320,25.599584,24.749051,
          24.810633,24.571242,23.618041,23.275660,23.109917,23.183197,22.460881,22.309073,22.333855,21.647892,21.318210,21.753941,21.121259,20.961632,20.113238,
          19.963796,19.867962,19.097353,19.692892,21.177035,19.183536,]
apikeys2=[77.335670,75.804204,77.554237,74.762488,74.842260,76.260181,77.472878,77.066105,75.956378,76.786035,78.192176,74.745860,77.342336,79.469030,77.211172,
          73.196316,77.970820,75.805664,70.855922,73.050593,80.885766,74.817085,80.409494,83.442908,89.469959,73.890818,81.471249,82.856953,85.085679,84.943779,
          79.912506,80.854105,87.240207,69.663998,70.104276,80.057008,70.015910,70.782321,73.226678,69.661041,70.267115,72.183477,72.742006,75.619148,73.893802,
          73.807493,75.395819,72.874745,74.394217,79.107196,77.334778,]
# print(datalist[len(apikeys1)])
def datacheck(a,b):
    if a in datalist and b in datalist:
        lst1=[]
        lst2=[]
        i=0
        j=0
        for s in datalist:
            if s==a:
                lst1.append(apikeys1[i])
                lst2.append(apikeys2[i])
                break
            else:
                i=i+1
        for s in datalist:
            if s==b:
                lst1.append(apikeys1[j])
                lst2.append(apikeys2[j])
                break
            else:
                j=j+1
        return lst1,lst2
    else:
        return False
```

Screenshots:

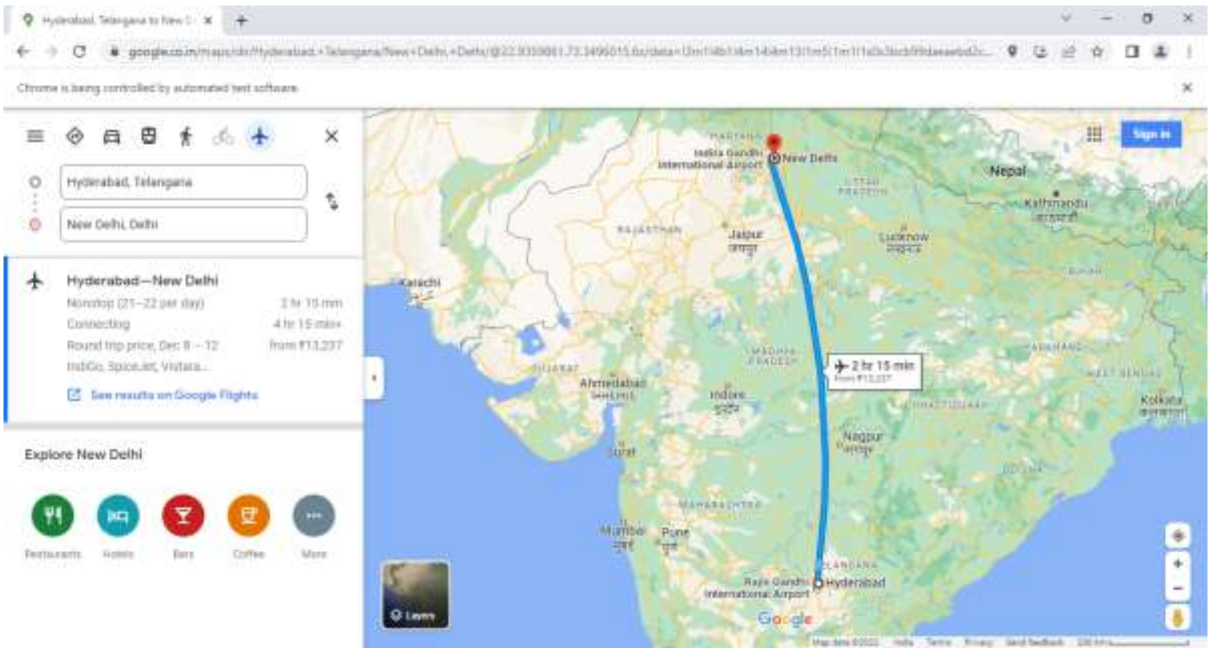






From hyderabad

TO delhi



Hyderabad, Telangana

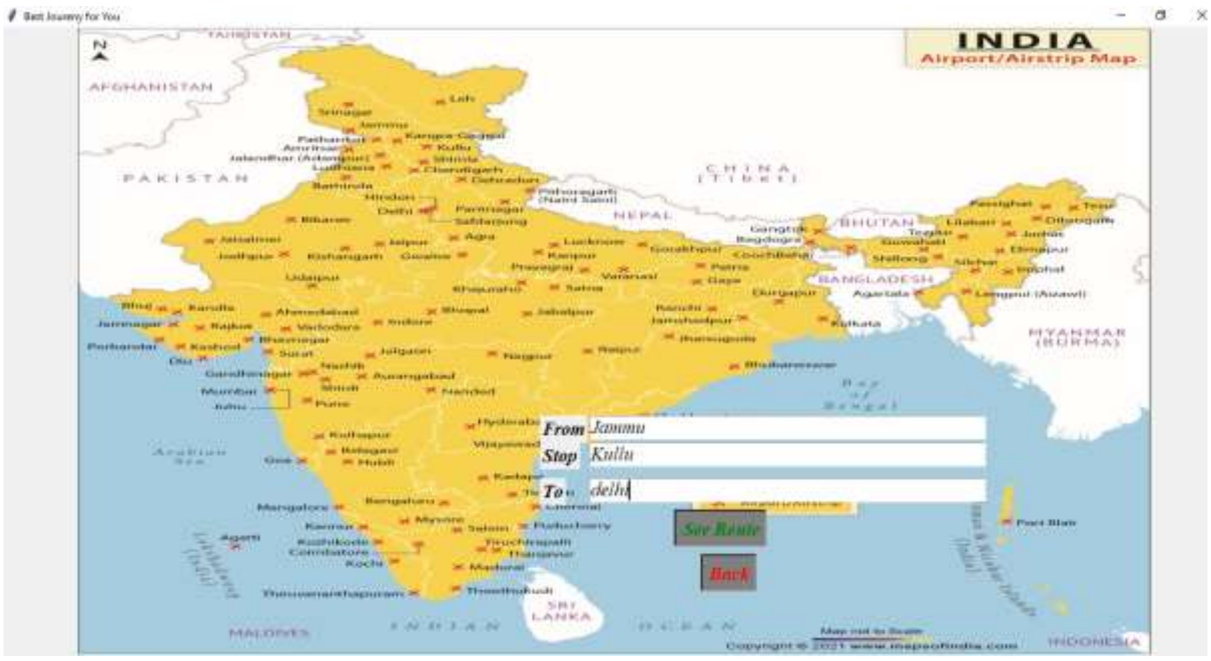
New Delhi, Delhi

Hyderabad—New Delhi

Nonstop (21–22 per day)	2 hr 15 min
Connecting	4 hr 15 min+
Round trip price, Dec 8 – 12	from ₹13,237

IndiGo, SpiceJet, Vistara...

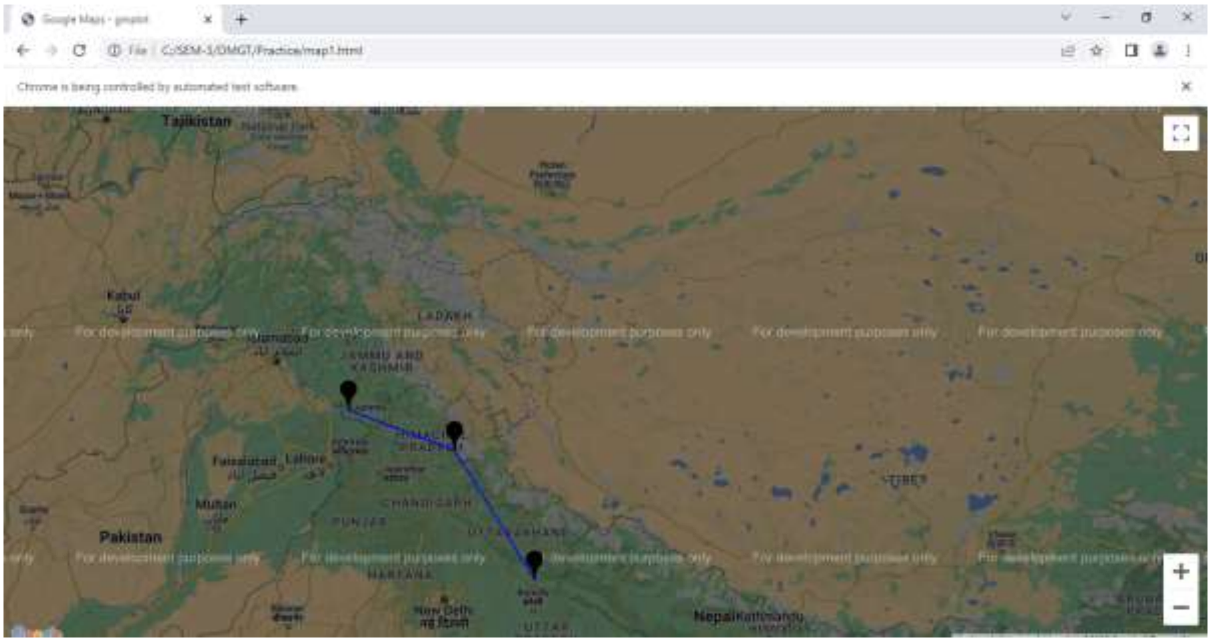
See results on Google Flights



From Jammu

Stop Kullu

To delhi



Chrome is being controlled by automated test software.