AIRWAY MANAGEMENT SYSTEM

DONE BY:
HARINI.J
GRADE:XII
YEAR: 2023-2024

INDEX

1. INTRODUCTION

The project aims to design an Airline Reservation System application which enables the customers to search and book flight. Airline Reservations Systems contain fare tariffs, passenger reservations and ticket records.

1.1 OVERVIEW OF THE PROJECT

The Airline Reservation System project is an implementation of a general Airline Ticketing website like Orbits, which helps the customers to search the availability of flights, book and cancel the flight tickets. This project also covers adding, deleting or modifying the customer details and flights. In general, this website would be designed to perform like any other airline ticketing website available online.

1.2 PURPOSE OF THE PROJECT

The purpose of this project is to implement or to design a database for an airline reservation system to check the flight details, book and cancel flight tickets. It makes the process of booking and cancelling flight tickets simple and easy for the passengers.

1.3 PROBLEM DEFINITION

Normally a person wants to reserve his ticket and he has to contact at nearest Travels branch. The Airline Reservation System provides an interface to schedule flights and reservations for an airline through internet. Its responsibility is to keep track of system users, customers, Airbus information, flight information and cancellation.

1.4 PROJECT PLAN AND SCOPE OF THE PROJECT

Airline Reservation System is one the modifications that were carried out in the Passenger Service System so that the working and availability of Service area can be broadened. On one hand, it helps the customers and on the other, it also makes the life of the airline service companies easier by keeping all the records of the passengers and if there is any change in the flight due to some reason, the passengers are promptly informed. This system is also used by companies to keep track of user preferences of regular travelers so that they can provide better service and give offers to customers.

2. SYSTEM ANALYSIS

Systems analysis is the study of sets of interacting entities, including computer systems analysis. This field is closely related to requirement analysis or operations research

. It is also "an explicit formal inquiry carried out to help someone (referred to as the decision maker). It identifies a better course of action and make a better decision than he might otherwise have made."

The development of a computer-based information system includes a systems analysis phase which produces or enhances data model which itself is a precursor to creating or enhancing a database.

There are a number of different approaches to system analysis. When a computerbased information system is developed, systems analysis would constitute the following steps:

- The development of a feasibility study, involving determining whether a project

is economically, socially, technologically and organizationally feasible.

- Conducting fact-finding measures, designed to ascertain the requirements of the

system's end-users. These typically span interviews, questionnaires, or visual observations of work on the existing system.

2.1 GENERAL

Systems analysis researchers apply methodology to the analysis of systems involved to form an database for checking their results. System analysis is used in every field where there is a work of developing something. Analysis can also be defined as a series of components that perform organic function together.

2.2 EXISTING SYSTEM

The existing system is that the passenger must fill up the data manually and must submit it to the reservation counter. It may take a lot of time to process it and to book the flight. Therefore there is wastage of time. Since the data is entered manually, the probability of error or mistakes is high.

2.2.1 PROPOSED SYSTEM

Entering Record-

Entry of each record is done manually each time the record is done manual each time the record is maintained on paper and it maximizes the maintenance of additional files.

Searching the record-

Due to absence of unique identification of a flight, the searching of record takes much time and increases the time wastage.

Deleting the Record-

In the current system the concept of deleting record is tedious.

Modification of Records-

If any modification is required it is done directly on the documents being preserved in correspondence to account information.

2.3 FEATURES OF PROPOSED SYSTEM

To avoid the limitation of current system it's necessary to design and develop a new system which have the following benefit and the existing system.

1. Everything is automated which reduce the risk factor.

2. Flexibility in generating of information.

3. Quick retrieved and maintenance of data.

4. Highly accurate.

5. User satisfaction.

2.3.1ADVANTAGES OF PROPOSED SYSTEM

- The proposed system due to computerized is much faster in reservation process, cancellation process and transactions.
- Transfer of information from various branches would become easier and faster.
- Managing and maintaining data becomes easier and cost effective due to very high amount and reliability of storage space available in the proposed system.

 Customer services can not only be satisfied but also enha nced to the extent that one can obtain or cancel a reservation from any given time.

2.4 FEASIBILITY STUDY

Feasibility study is a report directed management. It evaluates the impact of the proposed changes in the area(s) in question. The report is a formal document for management, brief enough and sufficiently, nontechnical to be understandable, yet detailed enough to provide the basis for system design.

2.4.1 Technical feasibility

Technical feasibility centers around the existing system (hardware, software, etc.) into what it can sort the proposed addition.

SYSTEM REQUIREMENTS

3.1HARDWARE REQUIREMENT-

An Intel based central processing unit capable of running any sort of windows operating system such as Pentium based workstation.

1. Minimum 64 MB RAM (128 MB Desirable) at server.

2. Minimum 60 MB of free disk space for files.

3. A CD Rom drive

4. Minimum 48 MB of RAM at workstation.

5. VGA 15” colour monitor for workstation.

3.2 SOFTWARE REQUIREMENT-

The software requirements are as follows.

1. Windows 98 or Above

2. Microsoft word

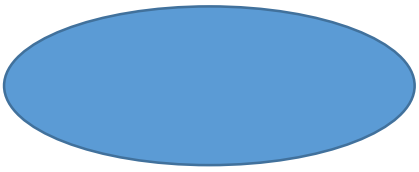
3.2.1 SOFTWARE FRONT END:

The front end for airline reservation system is PYTHON

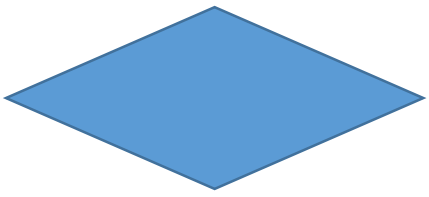
4.1 SYSTEM DESIGN

Entity relationship diagram is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems. An entity is a piece of data-an object or concept about which data is stored. A relationship is how the data is shared between entities.

REPRESENTS TABLE NAME



REPRESENTS ATTRIBUTE OF TABLE



REPRESENTS RELETIONSHIP OF TWO TABLE

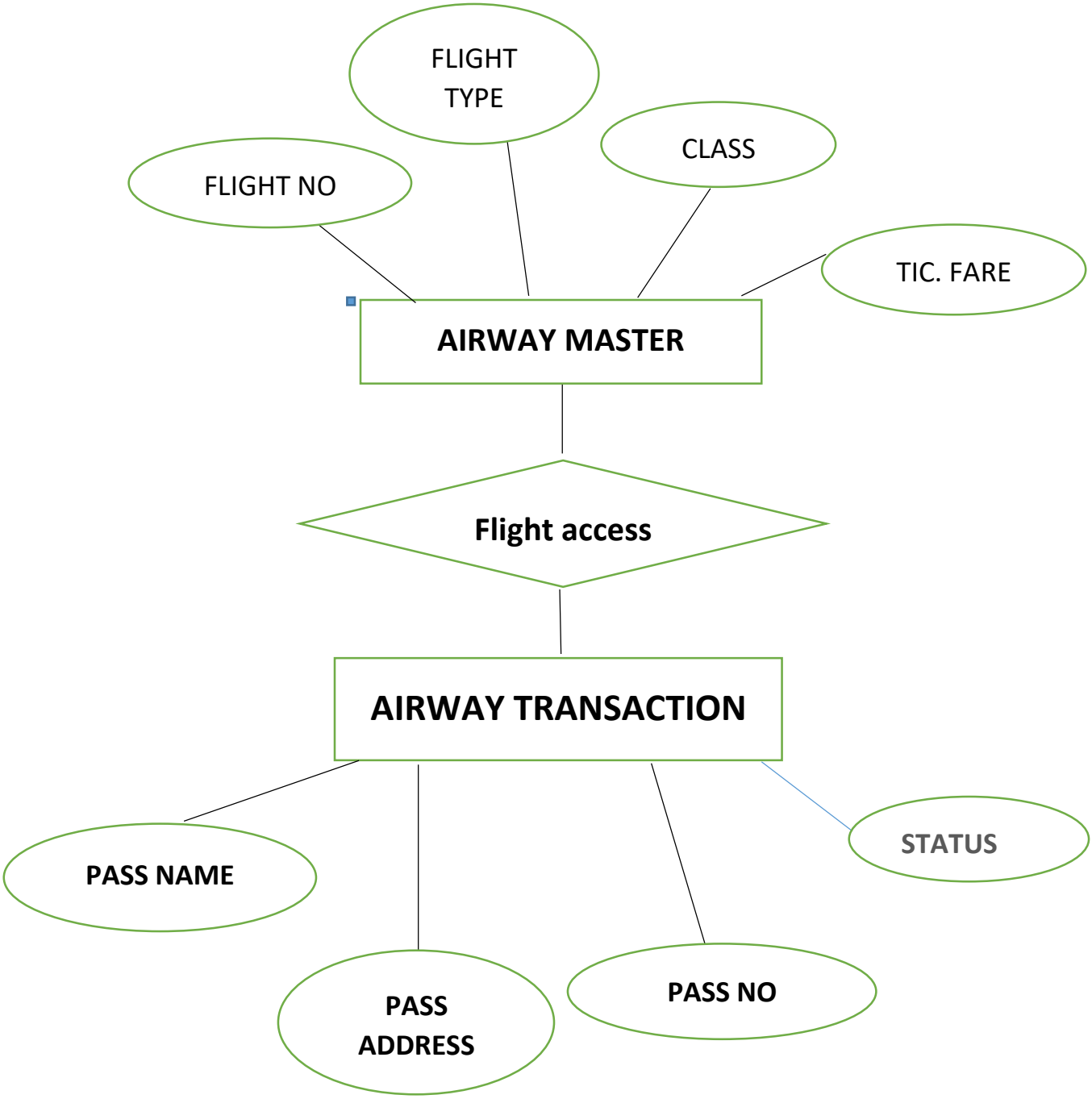
4.2 DESCRIPTION OF ENTITIES

FLIGHT DATA- this table contains all the information about the flight such as flight type, PNR number, source, destination, branch code, service code and class.

PASSENGER DATA

- this table contains all the details about the passenger like name, address, passport number, and status of the passenger.

4.3 ER-DIAGRAM



5.1 ABOUT PYTHON

Python is a dynamic, high-level, free open source, and interpreted programming language. It supports object-oriented programming as well as procedural-oriented programming. In Python, we don't need to declare the type of variable because it is a dynamically typed language. For example, x = 10 Here, x can be anything such as String, int, etc.

Features in Python

There are many features in Python, some of which are discussed below as follows:

1. Free and Open Source

Python language is freely available at the official website and you can download it from the given download link below click on the Download

Python keyword. Download Python Since it is open-source, this means that source code is also available to the public. So you can download it, use it as well as share it.

2. Easy to code

Python is a high-level programming language. Python is very easy to learn the language as compared to other languages like C, C#, Javascript, Java, etc. It is very

easy to code in the Python language and anybody can learn Python basics in a few hours or days. It is also a developer-friendly language.

3. Easy to Read

As you will see, learning Python is quite simple. As was already established, Python's syntax is really straightforward. The code block is defined by the indentations rather than by semicolons or brackets.

4. Object-Oriented Language

One of the key features of Python is Object-Oriented programming. Python supports object-oriented language and concepts of classes, object encapsulation, etc.

5. GUI Programming Support

Graphical User interfaces can be made using a module such as PyQt5, PyQt4, Python, or Tk in Python. PyQt5 is the most popular option for creating graphical apps with Python.

6. High-Level Language

Python is a high-level language. When we write programs in Python, we do not need to remember the system architecture, nor do we need to manage the memory.

7. Large Community Support

Python has gained popularity over the years. Our questions are constantly answered by the enormous StackOverflow community. These websites have already provided answers to many questions about Python, so Python users can consult them as needed.

8. Easy to Debug

Excellent information for mistake tracing. You will be able to quickly identify and correct the majority of your program's issues once you understand how to interpret Python's error traces. Simply by glancing at the code, you can determine what it is designed to perform.

9. Python is a Portable language

Python language is also a portable language. For example, if we have Python code for Windows and if we want to run this code on other platforms such as Linux, Unix, and Mac then we do not need to change it, we can run this code on any platform.

10. Python is an Integrated language

Python is also an Integrated language because we can easily integrate Python with other languages like C, C++, etc.

11. Interpreted Language:

Python is an Interpreted Language because Python code is executed line by line at a time. like other languages C, C++, Java, etc. there is no need to compile Python code this makes it easier to debug our code. The source code of Python is converted into an immediate form called byte code.

12. Large Standard Library

Python has a large standard library that provides a rich set of modules and functions so you do not have to write your own code for every single thing. There are many libraries present in Python such as regular expressions, unit-testing, web browsers, etc.

13. Dynamically Typed Language

Python is a dynamically-typed language. That means the type (for example- int, double, long, etc.) for a variable is decided at run time not in advance because of this feature we don't need to specify the type of variable.

14. Frontend and backend development

With a new project py script, you can run and write Python codes in HTML with the help of some simple tags <py-script & gt;, <py-env& gt;, etc. This will help you do frontend development work in Python like javascript. Backend is the strong forte of Python it's extensively used for this work cause of its frameworks like Django and Flask.

15. Allocating Memory Dynamically

In Python, the variable data type does not need to be specified. The memory is automatically allocated to a variable at runtime when it is given a value. Developers do not need to write int y = 18 if the integer value 15 is set to y. You may just type y=18.

Python is a high-level, interpreted, and general-purpose dynamic programming language that focuses on code readability. It generally has small programs when compared to Java and C. It was founded in 1991 by developer Guido Van Resume. Python ranks among the most popular and fastest-growing languages in the world. Python is a powerful, flexible, and easy- to-use language. In addition, the python community is very active. It is used in many organizations as it supports multiple programming paradigms. It also performs automatic memory management.

Advantages:

1. Presence of third-party modules
2. Extensive support libraries (NumPy for numerical calculations, Pandas for data analytics, etc.)
3. Open source and large active community base
4. Versatile, Easy to read, learn and write
5. User-friendly data structures

6. High-level language

7. Dynamically typed language (No need to mention data type based on the value assigned, it takes data type)

8. Object-Oriented and Procedural Programming language

9. Portable and Interactive

10. Ideal for prototypes – provide more functionality with less coding

11. Highly Efficient(Python's clean object-oriented design provides enhanced process control, and the language is equipped with excellent text processing and integration capabilities, as well as its own unit testing framework, which makes it more efficient.)

12. Internet of Things (IoT) Opportunities

13. Interpreted Language

14. Portable across Operating systems

Disadvantages:

1.Performance: Python is an interpreted language, which means that it can be slower than compiled languages like C or Java. This can be an issue for performance- intensive tasks.

2. Global Interpreter Lock: The Global Interpreter Lock (GIL) is a mechanism in Python that prevents multiple threads from executing Python code at once. This can limit the parallelism and concurrency of some applications.

3. Memory consumption: Python can consume a lot of memory, especially when working with large datasets or running complex algorithms.

4. Dynamically typed: Python is a dynamically typed language, which means that the types of variables can change at runtime. This can make it more difficult to catch errors and can lead to bugs.

5. Packaging and versioning: Python has a large number of packages and libraries, which can sometimes lead to versioning issues and package conflicts.

6. Lack of strictness: Python's flexibility can sometimes be a double-edged sword. While it can be great for rapid development and prototyping, it can also lead to code that is difficult to read and maintain.

7. Steep learning curve: While Python is generally considered to be a relatively easy language to learn, it can still have a steep learning curve for beginners, especially if they have no prior experience with programming.

Applications:

1. GUI-based desktop applications
2. Graphic design, image processing applications, Games, and Scientific/ computational Applications
3. Web frameworks and applications
4. Enterprise and Business applications
5. Operating Systems
6. Education
7. Database Access
8. Language Development
9. Prototyping
10. Software Development
11. Data Science and Machine Learning
12. Scripting

Organizations using Python:

1. Google (Components of Google spider and Search Engine)

2. Yahoo (Maps)

3. YouTube

4. Mozilla

5. Dropbox

6. Microsoft

5.2 BINARY FILE

What is a Binary File?

A binary file is one that does not contain text. It is used to store data in the form of bytes, which are typically interpreted as something other than textual characters. These files usually contain instructions in their headers to determine how to read the data stored in them. These can be used to store any type of data in a computer.

Even though all files store data similarly, your operating system does not treat them all in the same way. This means that even though a sound file and an image are stored in the file system as continuous strings of data, an image can not be played in a music player, and neither can a music file be opened in a photo viewing software. The format of the file moderates this behavior. Extensions, such as ".mp3" and ".jpg", attached to the files are used to determine the kind of data they're expected to contain.

Broadly speaking, all files can be classified into two major formats — text and binary. Binary files encompass all non-text files, while text files are highly restrictive, and can only store textual data.

Binary files can store any kind of data, as long as the header of the file defines the file type and other relevant information like the block and body sizes accurately.

Let us understand the differences and similarities between the two now.

Binary Files

All files that are not used to store textual data fall into this category. Any custom file type can be created using a binary file, as long as the necessary information on how to read the file is stored in the file. These files store multiple types of data like image, video, and audio in the same file. The only requirement that they present is to have a suitable program for reading such kind of data present in the system.

The PNG format is a great example of the above use-case. A PNG file can be read by most image viewers and shows graphical information. If you open a PNG file with a text editor, most of the file will be composed of unrecognizable characters. But you will also find pieces of readable text scattered all over the file. This is because the PNG file includes small sections for storing textual data along with the graphical information. Some other file formats support this too, and this is possible due to the dynamic nature of binary files. Binary files contain a header at the top. This header is the key to the file. It is used to store the information that identifies the file's content. Usually, headers contain the file type and other metadata like size and date last modified. If a binary file's header is damaged, it is equivalent to the key being lost, which means you can not access meaningful data from the file anymore.

Advantages of Using Binary Files

Binary files provide multiple benefits compared to plain text files. Let's take a look at some of them:

Efficiency via Compression

Data is stored in binary files according to custom rules for use-case specific optimizations. PNG is a great example of this because it can be used to create small and efficient image files.

Better Security

Once again, the customization that binary files offer allows businesses to create custom encoding standards, which can be difficult to reverse engineer. More often than not, the only way to read a custom-encoded binary file is to guess how data has been stored in it.

Unmatched Speed

As the data is stored in a raw format, and is not encoded using any character encoding standards, it is faster to read and store. This is the primary reason why data stores for applications are not built using plain text files.

Issues with Binary Files

While binary files offer many benefits over plain text files, they also present several issues. Following are the most common problems faced when using binary files:

Difficult to manipulate

Binary files can not be read by conventional text processors, so editing them is a difficult task. More often than not, applications choose to save their data using custom encoding schemes. This data can then be manipulated only inside of the applications with the same encoding scheme.

Efficiency gain is not uniform

While storing data in binary format might be fast and efficient in formats like PNG, other data types may not receive any noticeable performance benefits. Storing textual data adds another hassle of encoding and decoding data while viewing.

Can get confusing for machines

Computers can have different ways of storing and accessing data. When binary data is transmitted between two computers with different architectures, issues like

NUXI can arise. If a computer saves “UNIX” in a binary file, and the file is opened on another computer with a different architecture, it might be read as “NUXI”. Textual data stored in TXT or similar formats are immune to such issues, due to the presence of standards like ASCII.

Where are Binary Files Used?

Having seen the various aspects of binary files, it is now important to understand where to use them. Here are some of the top use-cases of binary files:

Software development. Most compilers like the JVM create optimized bytecode from source code for faster execution. While the source code is stored in the form of text files, it does not make much sense in storing the bytecode similarly. This is so because source code has to be viewed by a developer in the process of creating the software, while the bytecode has to be read by the machine. It will always be faster for machines to process bytes rather than encoded characters.

Image handling. As mentioned earlier, most image formats like PNG are stored as binary files to optimize performance, and also allow the storage of small image files. If the data associated with a movie file were in the form of encoded characters, a standard DVD would have never been able to store a complete movie.

Game development. Games require a great amount of integer and character-based calculations. It is convenient to store these numbers in the form of bytes and operate on them, as encoding and decoding a 32-bit integer adds a considerable amount of time to the process.

Storing large datasets. Datasets for tasks like machine learning model training are often required to be stored and made available to computers. While storing a dataset

in a text file makes sense, as it can be viewed easily, it can pose performance issues for the machine.

Pickling: The process of converting the structure (lists and dictionary etc.) into a byte stream just before writing to the file. This is also called as object serialization.

dump() function: We use dump() method to perform pickling operation on our **Binary Files.** It returns the object representation in byte mode. The dump() method belongs to pickle module.

Unpickling: The reverse conversion of byte stream back to the structure (lists, dictionary, tuples etc.) refers to unpickling. Basically it is the reverse operation of pickling. This is also called de-serialization. We use load() method for unpickling. This load() function/method also belongs to pickle module.

load() function: In pickle module, load() function is used to read data from a binary file or file object.

SOURCE CODE:

```
print('**************************************************
****************************************')
print('                    AIRWAY MANAGEMENT SYSTEM' )
print('**************************************************
****************************************')


def airway_master():
   print('****************************** AIRWAY MANAGEMENT
**********************************')
   import pickle


   def Adddata(airway_code):#p
      print("ADDING INFO AIRWAY MASTER")
      fp = open('airwaysindia1.dat','wb')
      n=int(input("ENTER NO. OF PASSENGERS:"))
      print()
      ch = 'Y'
      st = {}
      for i in range(n):
```

```
while ch.upper() == 'Y':
    PNR_NO = int(input('enter PNR number'))
    Name= input("Enter the Name")
    a = True
    while a:
        if Name.isalpha()!= True:
            print("enter valid Name")
            Name = input("enter name")
        elif Name.isalpha() == True:
                a = False
    DOB = input('Enter date of birth(dd/mm/yyyy)')
    Aadhar = input("Enter the aadhar no")
    a = True
    while a:
        if Aadhar.isdigit() != True:
            print("enter valid Aadhar no")
            Aadhar = input("enter Aadhar no")
        elif Aadhar.isdigit()== True:
                a = False
    Phone_no = input('Enter ph no')
```

```
a = True
while a:
    if Phone_no.isdigit()!= True:
        print("enter valid phone no")
        Phone_no = input("enter phone no")
    elif Phone_no.isdigit()== True:
            a = False
Gmail_id = input('Enter gmail')
Amount = input('enter amount')
a = True
while a:
    if Amount.isdigit()!= True:
        print("enter valid amount")
        Amount = input("enter amount")
    elif Amount.isdigit()== True:
            a = False
AIRWAY_NAME = input('ENTER THE AIRWAY')
st['PNR_NO'] = PNR_NO
st['A_code'] = airway_code
st['Name'] = Name
```

```
            st['DOB'] = DOB
            st['Aadhar'] = Aadhar
            st['Phone_no'] = Phone_no
            st['Gmail_id'] = Gmail_id
            st['Amount'] = Amount
            st['AIRWAY_NAME'] = AIRWAY_NAME
            pickle.dump(st,fp)
            print(st)
            ch = input('do u want to add more records into AIRWAY
MASTER (Y/N)')
        fp.close()

    def Update():
        print("UPDATING INFO AIRWAY MASTER")
        fp = open('airwaysindia1.dat','rb+')
        rn = input('Enter name in which data is to be updated')
        found = False
        data = {}
        try:
            while True:
```

```
            cur_pos=fp.tell()
            data=pickle.load(fp)
            if data['Name'] == rn:
                data['PNR_NO'] =  int(input('enter PNR no'))
                data['A_code'] = int(input('enter branch code'))
                data['Name'] = input('enter name')
                data['DOB'] = input('enter dob')
                data['Aadhar'] = int(input('enter aadhar no'))
                data['Phone_no'] = int(input('enter phone no'))
                data['Gmail_id'] = input('enter gmail')
                data['Amount'] = int(input('enter amount'))
                data['AIRWAY_NAME'] = input('enter airway name')
                fp.seek(cur_pos)
                pickle.dump(data,fp)
                found = True
                break
        except EOFError:
            if found == False:
                print('record not found')
            else:
```

```
            print('record updated')
        fp.close()


    def Display():
        import pickle
        print("DISPLAYING INFO AIRWAY MASTER")
        fp = open('airwaysindia1.dat','rb')
        while True:
            try:
                rec = pickle.load(fp)
                print(rec)
            except EOFError:
                break
        fp.close()


    def Search():
        print("SEARCHING INFO AIRWAY MASTER")
        fp = open('airwaysindia1.dat','rb')
        rn = int(input('Enter PNR no in which data is to be searched'))
        flag = True
```

```
    while True:
        try:
            rec = pickle.load(fp)
            if rec['PNR_NO'] == rn:
                print('Yes,the above data is present')
                print(rec)
                flag = False
        except EOFError:
            break
    if flag == True:
        print('record not found')
    fp.close()

def Delete():
    print("DELETING INFO AIRWAY MASTER")
    import pickle
    import os
    fp=open("airwaysindia1.dat","rb")
    fp1=open("airwayssss1.dat","wb")
    a=int(input("enter the PNR_NO to delete:"))
```

```
    while True:
        try:
            r=pickle.load(fp)
            if r["PNR_NO"]!=a:
                pickle.dump(r,fp1)
        except:
            break
    fp.close()
    fp1.close()
    os.remove("airwaysindia1.dat")
    os.rename("airwayssss1.dat","airwaysindia1.dat")

print("MAIN MENU-AIRWAY MASTER")
ch = 'Y'
while ch.upper() == 'Y':
    print('1.Add Passenger')
    print('2.Update Passenger info')
    print('3.Display Passenger info')
    print('4.Search for a given passenger')
    print('5.Delete a given Passenger info')
```

```
        print('6.Exit')
        c = int(input('enter from 1/2/3/4/5'))
        if c==1:

           Adddata(18)#p
        elif c == 2:
           Update()
        elif c == 3:
           Display()
        elif c == 4:
           Search()
        elif c == 5:
           Delete()
        else:
           exit
        ch = input('do you want to continue or exit main menu of
AIRWAY MASTER (Y/N)')

def airway_transaction():
   import pickle
```

```
    print('****************************** AIRWAY TRANSACTION
******************************')
    def Adddata1():
        fp=open('airways1.dat','wb')
        st1={}
        ch='Y'
        while(ch.upper()=='Y'):
            PNR_NO=int(input('enter pnr no.'))
            nop_=input("enter passenger name:")
            agp_=int(input('enter passenger age:'))
            flnm=input("enter flight name:")

            fcl=input('enter class type:')
            if fcl=='economy':
                print('$999')

            if fcl=='business':
                print('$1200')

            if fcl=='firstclass':
```

```
        print('$1500')

    start=input("enter starting place:")

    end=input("enter ending place:")
    st1['PNRNO']=PNR_NO
    st1['na_OF_pass']=nop_
    st1['AGE_OF_pass']=agp_


    st1['FLIGHT_NAME']=flnm
    st1['CLASS']=fcl
    st1['STARTING_PLACE']=start
    st1['ENDING_PLACE']=end

    pickle.dump(st1,fp)
    print(st1)
    ch=input('Do you want to add more(Y/N)')
fp.close()
```

```
def Update():
    print("UPDATING INFO AIRWAY TRANSACTION")
    fp=open("airways1.dat","rb+")
    pc=int(input("enter the pnr number"))
    #f.seek(0)
    while True:
        try:
            cur_pos=fp.tell()
            data=pickle.load(fp)
            if data['PNRNO']==pc:
                data['PNRNO']=int(input('enter pnr no.'))

                data['na_OF_pass']=input("enter passenger name:")

                data['AGE_OF_pass']=int(input('enter passenger age:'))

                data['FLIGHTNAME']=input('enter flight name:')
```

```
            data['CLASS']=input('enter class type:')

            data['STARTINGPLACE']=input("enter starting place:")

            data['ENDINGPLACE']=input("enter ending place:")

            fp.seek(cur_pos)
            pickle.dump(data,fp)
            break
        except EOFError:
            if found == False:
                print('record not found')
            else:
                print('record updated')
    fp.close()

def Display():
    print("DISPLAY INFO AIRWAY TRANSACTION")
    fp = open('airways1.dat','rb')
    while True:
```

```
        try:
            rec = pickle.load(fp)
            print(rec)
        except EOFError:
            break
    fp.close()

def Search():
    print("SEARCH INFO AIRWAY TRANSACTION")
    fp = open('airways1.dat','rb')
    rn = int(input('Enter PNR no to be searched'))
    while True:
        try:
            rec = pickle.load(fp)
            if rec['PNRNO'] == rn:
                print('Yes , the above data is present')
                print(rec)
        except EOFError:
            break
    fp.close()
```

```
def Delete1():
    import pickle
    import os
    fp = open('airwayS1.dat','rb')
    fp1 = open('airways2.dat','wb')
    a = int(input('enter PNR no to delete:'))
    while True:
        try:
            r = pickle.load(fp)
            if r['PNRNO']!=a:
                pickle.dump(r,fp1)
        except EOFError:
            break
    fp.close()
    fp1.close()
    os.remove('airways1.dat')
    os.rename('airways2.dat','airways2.dat')

print("INTO MAIN MENU-AIRWAY TRANSACTION")
```

```
ch = 'Y'
while ch.upper() == 'Y':
    print('1.Add transaction of Passenger')
    print('2.Update Passanger details')
    print('3.Display Passenger details')
    print('4.Search for a given Passenger')
    print('5.Delete a given Passenger details')
    print('6.Exit')
    c = int(input('Enter from 1/2/3/4/5'))
    if c==1:
        Adddata1()
    elif c == 2:
        Update()
    elif c == 3:
        Display()
    elif c == 4:
        Search()
    elif c == 5:
        Delete1()
    else:
```

```
            exit
        ch = input('do you want to exit the main menu of transaction
(Y/N)')


ch = 'Y'
while ch.upper() == 'Y':
    print("MASTER MENU")
    print('1.AIRWAY MASTER')
    print('2.AIRWAY TRANSACTION')
    c = int(input('ENTER NO: 1/2'))
    if c == 1:
        airway_master()
    elif c == 2:
        airway_transaction()
    else:
        exit
    ch = input('DO YOU WANT TO CONTINUE(Y/N)')
```

SCREEN SHORTS:

1. IN AIRWAY MASTER

1.1SELECTING MENU

```
6.Exit
enter from 1/2/3/4/51
ADDING INFO AIRWAY MASTER
ENTER NO. OF PASSENGERS:2

enter PNR number12
Enter the NameANIL
Enter date of birth(dd/mm/yyyy)1202001
Enter the aadhar no987698765432
Enter ph no9841616432
Enter gmailHELLO@gmail.com
enter amount6000
ENTER THE AIRWAYINDIAN AIRWAY
{'PNR_NO': 12, 'A_code': 18, 'Name': 'ANIL', 'DOB': '1202001', 'Aadhar': '987698765432', 'Phone_no':
'9841616432', 'Gmail_id': 'HELLO@gmail.com', 'Amount': '6000', 'AIRWAY_NAME': 'INDIAN AIRWAY'}
do u want to add more records into AIRWAY MASTER (Y/N)Y
enter PNR number23
Enter the NameTANVI
Enter date of birth(dd/mm/yyyy)31/01/2002
Enter the aadhar no997655534789
Enter ph no8976543478
Enter gmailHI@gmail.com
enter amount5000
ENTER THE AIRWAYINDIGO
{'PNR_NO': 23, 'A_code': 18, 'Name': 'TANVI', 'DOB': '31/01/2002', 'Aadhar': '997655534789', 'Phone_n
o': '8976543478', 'Gmail_id': 'HI@gmail.com', 'Amount': '5000', 'AIRWAY_NAME': 'INDIGO'}
do u want to add more records into AIRWAY MASTER (Y/N)
```

```
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: D:/Users/Harini/Downloads/NO.py ==========
*****************************************************************************************
                    FLIGHT MANAGEMENT SYSTEM
*****************************************************************************************
MASTER MENU
1.AIRWAY MASTER
2.AIRWAY TRANSACTION
ENTER NO: 1/21
***************************** AIRWAY MANAGEMENT ***********************************
MAIN MENU-AIRWAY MASTER
1.Add Passenger
2.Update Passenger info
3.Display Passenger info
4.Search for a given passenger
5.Delete a given Passenger info
6.Exit
enter from 1/2/3/4/51
ADDING INFO AIRWAY MASTER
ENTER NO. OF PASSENGERS:2

enter PNR number12
Enter the NameANIL
Enter date of birth(dd/mm/yyyy)1202001
Enter the aadhar no987698765432
```

1.2ADDING OF DATA

1.3UPDATING OF DICTIONARY

```
do u want to add more records into AIRWAY MASTER (Y/N)N
do you want to continue or exit main menu of AIRWAY MASTER (Y/N)Y
1.Add Passenger
2.Update Passenger info
3.Display Passenger info
4.Search for a given passenger
5.Delete a given Passenger info
6.Exit
enter from 1/2/3/4/52
UPDATING INFO AIRWAY MASTER
Enter name in which data is to be updatedTANVI
enter PNR no23
enter branch code56
enter nameTANVI
enter dob05/06/1999
enter aadhar no9876987699876
enter phone no9687987667
enter gmailTAN@gmail.com
enter amount7000
enter airway nameAir India Express
do you want to continue or exit main menu of AIRWAY MASTER (Y/N)n
```

1.4DISPLAYING OF DATA

```
MASTER MENU
1.AIRWAY MASTER
2.AIRWAY TRANSACTION
ENTER NO: 1/21
****************************** AIRWAY MANAGEMENT ************************************
MAIN MENU-AIRWAY MASTER
1.Add Passenger
2.Update Passenger info
3.Display Passenger info
4.Search for a given passenger
5.Delete a given Passenger info
6.Exit
enter from 1/2/3/4/53
DISPLAYING INFO AIRWAY MASTER
{'PNR_NO': 12, 'A_code': 18, 'Name': 'ANIL', 'DOB': '1202001', 'Aadhar': '987698765432', 'Phone_no':
'9841616432', 'Gmail_id': 'HELLO@gmail.com', 'Amount': '6000', 'AIRWAY_NAME': 'INDIAN AIRWAY'}
{'PNR_NO': 23, 'A_code': 56, 'Name': 'TANVI', 'DOB': '05/06/1999', 'Aadhar': 9876987699876, 'Phone_no
': 9687987667, 'Gmail_id': 'TAN@gmail.com', 'Amount': 7000, 'AIRWAY_NAME': 'Air India Express'}
```

1.5SEARCHING OF DATA

```
1.Add Passenger
2.Update Passenger info
3.Display Passenger info
4.Search for a given passenger
5.Delete a given Passenger info
6.Exit
enter from 1/2/3/4/53
DISPLAYING INFO AIRWAY MASTER
{'PNR_NO': 23, 'A_code': 18, 'Name': 'JAN', 'DOB': '06/09/2000', 'Aadhar': '987659876598', 'Phone_no'
: '9876598765', 'Gmail_id': 'KAN@gmail.com', 'Amount': '7000', 'AIRWAY_NAME': 'indigo'}
do you want to continue or exit main menu of AIRWAY MASTER (Y/N)y
1.Add Passenger
2.Update Passenger info
3.Display Passenger info
4.Search for a given passenger
5.Delete a given Passenger info
6.Exit
enter from 1/2/3/4/54
SEARCHING INFO AIRWAY MASTER
Enter PNR no in which data is to be searched23
Yes,the above data is present
{'PNR_NO': 23, 'A_code': 18, 'Name': 'JAN', 'DOB': '06/09/2000', 'Aadhar': '987659876598', 'Phone_no'
: '9876598765', 'Gmail_id': 'KAN@gmail.com', 'Amount': '7000', 'AIRWAY_NAME': 'indigo'}
do you want to continue or exit main menu of AIRWAY MASTER (Y/N)
```

1.5 DELETING AND DISPLAYING

```
enter amount5000
ENTER THE AIRWAYINDIGO
{'PNR_NO': 56, 'A_code': 18, 'Name': 'GANDU', 'DOB': '07/09/1999', 'Aadhar': '9876540987', 'Phone_no'
: '9876556789', 'Gmail_id': 'GANDU@gmail.com', 'Amount': '5000', 'AIRWAY_NAME': 'INDIGO'}
do u want to add more records into AIRWAY MASTER (Y/N)N
do you want to continue or exit main menu of AIRWAY MASTER (Y/N)Y
1.Add Passenger
2.Update Passenger info
3.Display Passenger info
4.Search for a given passenger
5.Delete a given Passenger info
6.Exit
enter from 1/2/3/4/55
DELETING INFO AIRWAY MASTER
enter the PNR_NO to delete:45
do you want to continue or exit main menu of AIRWAY MASTER (Y/N)Y
1.Add Passenger
2.Update Passenger info
3.Display Passenger info
4.Search for a given passenger
5.Delete a given Passenger info
6.Exit
enter from 1/2/3/4/53
DISPLAYING INFO AIRWAY MASTER
{'PNR_NO': 56, 'A_code': 18, 'Name': 'GANDU', 'DOB': '07/09/1999', 'Aadhar': '9876540987', 'Phone_no'
: '9876556789', 'Gmail_id': 'GANDU@gmail.com', 'Amount': '5000', 'AIRWAY_NAME': 'INDIGO'}
do you want to continue or exit main menu of AIRWAY MASTER (Y/N)
```

2. IN AIRWAY TRANSACTION

2.1 SELECTING OF MENU

```
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: D:/Users/Harini/Downloads/NO.py ==========
*************************************************************************************************
                        FLIGHT MANAGEMENT SYSTEM
*************************************************************************************************
MASTER MENU
1.AIRWAY MASTER
2.AIRWAY TRANSACTION
ENTER NO: 1/22
**************************** AIRWAY TRANSACTION ******************************
INTO MAIN MENU-AIRWAY TRANSACTION
1.Add transaction of Passenger
2.Update Passanger details
3.Display Passenger details
4.Search for a given Passenger
5.Delete a given Passenger details
6.Exit
Enter from 1/2/3/4/51
enter pnr no.34
enter passenger name:HANI
enter passenger age:15
enter flight name:INDIGO
enter class type:economy
$999
```

2.2 ADDING OF DATA

```
**************************** AIRWAY TRANSACTION ******************************
INTO MAIN MENU-AIRWAY TRANSACTION
1.Add transaction of Passenger
2.Update Passanger details
3.Display Passenger details
4.Search for a given Passenger
5.Delete a given Passenger details
6.Exit
Enter from 1/2/3/4/51
enter pnr no.34
enter passenger name:HANI
enter passenger age:15
enter flight name:INDIGO
enter class type:economy
$999
enter starting place:mumbai
enter ending place:chennai
{'PNRNO': 34, 'na_OF_pass': 'HANI', 'AGE_OF_pass': 15, 'FLIGHT_NAME': 'INDIGO', 'CLASS': 'economy', '
STARTING_PLACE': 'mumbai', 'ENDING_PLACE': 'chennai'}
Do you want to add more(Y/N)y
enter pnr no.56
enter passenger name:fanu
enter passenger age:45
enter flight name:indigo
enter class type:economy
$999
```

2.3UPDATING OF DATA

```
Enter from 1/2/3/4/52
UPDATING INFO AIRWAY TRANSACTION
enter the pnr number56
enter pnr no.34
enter passenger name:hanunam
enter passenger age:32
enter flight name:indigo
enter class type:economy
enter starting place:chennai
enter ending place:mumbai
do you want to exit the main menu of transaction (Y/N)y
1.Add transaction of Passenger
2.Update Passanger details
3.Display Passenger details
4.Search for a given Passenger
5.Delete a given Passenger details
6.Exit
```

1.4 DISPLAYING OF DATA

```
Enter from 1/2/3/4/52
UPDATING INFO AIRWAY TRANSACTION
enter the pnr number56
enter pnr no.34
enter passenger name:hanunam
enter passenger age:32
enter flight name:indigo
enter class type:economy
enter starting place:chennai
enter ending place:mumbai
do you want to exit the main menu of transaction (Y/N)y
1.Add transaction of Passenger
2.Update Passanger details
3.Display Passenger details
4.Search for a given Passenger
5.Delete a given Passenger details
6.Exit
```

1.5 SEARCHING OF DATA

```
========== RESTART: D:/Users/Harini/Downloads/NO.py ==========
**********************************************************************************************
                    FLIGHT MANAGEMENT SYSTEM
**********************************************************************************************
MASTER MENU
1.AIRWAY MASTER
2.AIRWAY TRANSACTION
ENTER NO: 1/22
**************************** AIRWAY TRANSACTION ******************************
INTO MAIN MENU-AIRWAY TRANSACTION
1.Add transaction of Passenger
2.Update Passanger details
3.Display Passenger details
4.Search for a given Passenger
5.Delete a given Passenger details
6.Exit
Enter from 1/2/3/4/54
SEARCH INFO AIRWAY TRANSACTION
Enter PNR no to be searched34
Yes , the above data is present
{'PNRNO': 34, 'na_OF_pass': 'HANI', 'AGE_OF_pass': 15, 'FLIGHT_NAME': 'INDIGO', 'CLASS': 'economy', '
STARTING_PLACE': 'mumbai', 'ENDING_PLACE': 'chennai'}
Yes , the above data is present
{'PNRNO': 34, 'na_OF_pass': 'hanunam', 'AGE_OF_pass': 32, 'FLIGHT_NAME': 'indigo', 'CLASS': 'economy'
, 'STARTING_PLACE': 'mumbai', 'ENDING_PLACE': 'chennai', 'FLIGHTNAME': 'indigo', 'STARTINGPLACE': 'ch
ennai', 'ENDINGPLACE': 'mumbai'}
do you want to exit the main menu of transaction (Y/N)
```

1.6 DELETING OF DATA

```
**********************************************************************************************
                    FLIGHT MANAGEMENT SYSTEM
**********************************************************************************************
MASTER MENU
1.AIRWAY MASTER
2.AIRWAY TRANSACTION
ENTER NO: 1/22
**************************** AIRWAY TRANSACTION ******************************
INTO MAIN MENU-AIRWAY TRANSACTION
1.Add transaction of Passenger
2.Update Passanger details
3.Display Passenger details
4.Search for a given Passenger
5.Delete a given Passenger details
6.Exit
Enter from 1/2/3/4/54
SEARCH INFO AIRWAY TRANSACTION
Enter PNR no to be searched34
Yes , the above data is present
{'PNRNO': 34, 'na_OF_pass': 'HANI', 'AGE_OF_pass': 15, 'FLIGHT_NAME': 'INDIGO', 'CLASS': 'economy', '
STARTING_PLACE': 'mumbai', 'ENDING_PLACE': 'chennai'}
Yes , the above data is present
{'PNRNO': 34, 'na_OF_pass': 'hanunam', 'AGE_OF_pass': 32, 'FLIGHT_NAME': 'indigo', 'CLASS': 'economy'
, 'STARTING_PLACE': 'mumbai', 'ENDING_PLACE': 'chennai', 'FLIGHTNAME': 'indigo', 'STARTINGPLACE': 'ch
ennai', 'ENDINGPLACE': 'mumbai'}
do you want to exit the main menu of transaction (Y/N)
```

CONCLUSION

In the conclusion we are glad to inform that the proposed system implementation and testing has been completed successfully. The proposed system successfully avoids the unfair seat allotment practice of the waiting list passengers after the chart has been prepared. It is also proven that the system is user friendly and very easy to use. The proposed system is designed in such a way that it can book multiple tickets by multi users at the same time. With the help of the proposed system we hope to reduce the corruption and the unfair seat allotment to the waiting list passengers.

BIBILOGRAPHY

- **Grade**
- **Google**
- **Chrome -12 CSC textbook-SUMITA ARORA**