Airline Reservation System

Mane Mansi Ganeshrao

CHAPTER ONE

INTRODUCTION

1.0 Background of study

Many people are travelling with airplanes, either as means of daily transportation to and from work or when going on vacation, to mention a few. To make reservations for such travels, airline companies' websites holds the functionality for the user to book a travel himself. A functionality which these websites lack is the option for the user to set up specific requirements for a travel, such as; minimal travel time or travel distance(Jarvenpaa L. S, 1996). The purpose of this project is to develop an easy-to-use airline reservation system, which accommodates these functionalities.

In addition, the system should also be of use for travel agencies. These should have the same functionalities in the system as the private users, but with the difference of also having a minimum spanning tree at their disposal, thus enabling them a greater understanding of the fight network.

Evolving from manual records and logs in the early 1930s, Airlines Reservations System (Arsanjani) is the improved, computerized feature of airline reservations(Winston, 1995). ARS helps in systematic and effective organization of bookings, prices, schedules and customer data. Airlines reservations system has today evolved into Computer Reservations System (CRS). ARS, when integrated with Global Distribution System (GDS), can be used by multiple distribution channels such as travel agencies, which can then use it for hotel rentals, flight booking, car hires as well as activities and tours via single system. ARS consists of several areas such as the inventory management, availability display and reservation and fare quotes and tickets. American Airlines introduced the first automated ARS called the Electromechanical Reservisor in 1946. It was followed by a new machine

called MagnetronicReservisor. In 1959, to improve the existing Reservisor, an improved automated booking system called SABRE was introduced(Winston, 1995). Similarly, other airlines created their own systems for ticket booking and management. Today, many brands co-operate with the world airlines companies for user-friendly direct systems, increased productivity and efficiency. Some of the major ARS brands today are Abacus, Amadeus, Navitaire, Sabre and TravelSky.

1.1 Statement of the problem

The management of Airline in Nigeria for example has over the years attracted poor patronage as a result of errors inherent in the system. It is no longer a new thing that loss of customers' goods is now the order of the day; the reasons are not far-fetched. Due to this manual procedure involved in Airline management, clients have no other option than to be at the mercy of these error prone procedures. The method of information storage is poor. This limits the number of official documents accessible by the customers because the system is not capable of managing old items of reservations which could be of use to the customer at any time. There is little or no security control system where the customers" goods, document and classified information of the customer could be safe guarded from unauthorized access.

1.2 Purpose of the study

This project is aimed at exposing the relevance and importance of Airline reservation system (ARS). It is projected towards enhancing the relationship between customers and airline agencies through the use of ARSs, thereby easing the flight ticketing and selling process and all air travelling operations.

1.3 Aim and Objectives

The aim of the project is to design and implement the software which helps the Airline System employees to issue reservation tickets for various Air flights and maintain the records of various passengers and provide quick services to the passengers.

The objectives include:

1. To maintain consistency among different access modes, e.g. by phone, by web, at the information desk and across different physical locations.
2. To minimize repetitive work done by the system administrator and reservation clerks.
3. To maintain customer information in case of emergency..
4. To Increase awareness among frequent travelers about various special offers and discounts.
5. To minimize the number of vacant seats on a flight and maximize flight capacity utilization.
6. To maintain the capability to adopt a flexible pricing policy.

1.4 Significance of study

The significance of Airline reservation system is the computerization of the activities of the organization. It helps to facilitate the daily operation of the organization. The economy of the organization is affected positively because of the computerization of their operation.

The findings of this research will also help the management to increase the income generation and smooth running of the everyday activities. This presentation will be beneficial to all those who make use of Airline flight information system (ARSs), flight operators, air travelling operators, travel agents and airline agencies.

1.5 Methodology

- A survey on the mode of operation of aero land agency in ikeja in order to create bases for a new proposed system.
- A general review of the existing system so as to spot the bottle neck, correct them and the procedure in a new solution.
- A system design which cut across designing software and a new system operation. i.e. operation manual.

1.6 Scope of study

This study is restricted to the full operations of Airline reservation with respect to online airline ticketing portal for travelling services provider.

CHAPTER TWO

LITERATURE REVIEW

2.0 Introduction

An airline reservation system is a part of so-called Passenger Services Systems (PSS), which are applications supporting the direct contract with the passenger(Crosby, 2015).

The airline reservation system was one of the earliest changes to improve efficiency. Airline reservation system eventually evolved into the Computer Reservation System(CRS). Computer reservation systems is used for the reservation of a particular airline and interface with a Global Distribution System (GDS) which travel agency and other distribution channels in making reservations for most major airlines in a single system

2.1 Review

The history of airline reservations systems began in the late 1950s when Airlines required a system that would allow real-time access to flight details in all of its offices, and the integration and automation of its booking and ticketing processes. As a result, Sabre(Semi-Automated Business Research Environment) was developed and launched in 1964(C. Winston, 1995). Sabre's breakthrough was its ability to keep inventory correct in real time, accessible to agents around the world. Prior to this, manual systems required centralized reservation centres, groups of people in a room with the physical cards that represented inventory, in this case, seats on airplanes.

The deregulation of the airline industry, in the Airline Deregulation Act, meant that airlines, which had previously operated under government-set fares ensuring airlines at least broke even, now needed to improve efficiency to compete in a free market(Shon, 2003). In this deregulated environment the ARS

and its descendants became vital to the travel industry. In the early days of American commercial, passengers were relatively few, and each airline's routes and fares were tightly regulated by the Civil Aeronautics Board. These were published in a volume entitled The Official Airline Guide, from which travel agents or consumers could construct an itinerary, then call or telex airline staff, which would mark the reservation on a card and file it. As demand for air travel increased and schedules grew more complex, this process became impractical. This system was used in the hospitality industry.

In 1946, Airlines installed the first automated booking system, the experimental electromechanical Reservation. A newer machine with temporary storage based on a magnetic drum, the Magnetronic Reservisor, soon followed(Smith, 2002). This system proved successful, and was soon being used by several airlines, as well as Sheraton Hotel sand Goodyear for inventory control. It was seriously hampered by the need for local human operators to do the actual lookups; ticketing agents would have to call a booking office, whose operators would direct a small team operating the Reservisor and then read the results over the telephone. There was no way for agents to directly query the system.

In 1953, Trans-Canada Airlines(TCA) started investigating a computer-based system with remote terminals, testing one design on the University of Toronto's Manchester Mark 1machine that summer. Though successful, the researchers found that input and output was a major problem. Canada became involved in the project and suggested a new system using cards and a transistorized computer in place of the unreliable tube-based Mark I. The resulting system, Reser Vec, started operation in 1962, and took over all booking operations in January 1963(Shon, 2003). Terminals were placed in all of TCA's ticketing offices, allowing all queries and bookings to complete in about one second with no remote operators needed.

In 1953, American Airlines CEO Smith chanced to sit next to R. Blair Smith, a senior IBM sales representative, on a flight from Los Angeles to New York. C.R. invited Blair to visit their Reservisor system and look for ways that IBM could improve the system. Blair alerted Thomas Watson Jr.that American was interested in a major collaboration, and a series of low-level studies started. Their idea of an automated Airline Reservation System(Arsanjani) resulted in a 1959 venture known as the Semi-Automatic Business Research Environment(SABRE), launched the following year(Johanson, 2014).By the time the network was completed in December 1964, it was the largest civil data processing system in the world.

Other airlines soon established their own systems. Delta Air Lines launched the Delta Automated Travel Account System (DATAS) in 1968. Airline sand Airlines followed in 1971 with the Apollo(Winston, 1995).

Reservation System and Programmed Airline Reservation System(PARS), respectively. Soon, travel agents began pushing for a system that could automate their side of the process by accessing the various ARS es directly to make reservations. Fearful this would place too much power in the hands of agents, American Airlines executive Crandall proposed creating an industry-wide Computer Reservation System to be a central clearinghouse for U.S. travel; other airlines demurred, citing fear of antitrust prosecution(Rudstrom, 2004).

An Airline Reservation System is part of the so-called Service Systems(PSS), which are applications supporting the direct contact with the passenger.

The Airline Reservations System (Arsanjani) was one of the earliest changes to improve efficiency. ARS eventually evolved into the Computer Reservations System(CRS). A Computer Reservation System is used for the reservations of a particular airline and interfaces with a Global Distribution System(GDS) which supports travel agencies and other distribution channels in making reservations for most major airlines in a single system(Rudstrom, 2004).

Airline Reservations Systems contain airline schedules, fare tariffs, passenger reservations and ticket records. An airline's direct distribution works within their own reservation system, as well as pushing out information to the GDS. A second type of direct distribution channel is consumers who use the internet or mobile applications to make their own reservations. Travel agencies and other indirect distribution channels access the same GDS as those accessed by the airlines' reservation systems, and all messaging is transmitted by a standardized messaging system that functions primarily on TTY messaging called SITA(Erradi, 2006). Since airline flight information system are business critical applications, and their functionally quite complex, the operation of an in-house airline reservation system is relatively expensive.

Prior to deregulation, airlines owned their own reservation systems with travel agents subscribing to them. Today, the GDS are run by independent companies with airlines and travel agencies as major subscribers.

As of February 2009, there are only three major GDS providers in the market space: Amadeus, Travel port(the merged span and Galileo systems), Sabre and Shares. There is one major Regional GDS,

Abacus, serving the Asian marketplace and a number of regional players serving single countries, including Travel sky (China), Infini and Axes (both Japan) and Topaz (South Korea). There is a secondary GDS called Navitaire that hosts "ticket less" airlines such as AirTran, and previously JetBlue(Atkinson, 2015). Virgin America is hosted by iflyRes(AIRes), which is a new generation reservation system developed and operated by IBS Software Service Pvt. Ltd.

In additional to these "standardized" GDS, some airlines have proprietary versions which they use to run their flight operations. A few examples of this kind of system are Deltamatic (built off the Worldspan platform) and EDS SHARES. SITA Reservations remains the largest neutral multi-host airline reservations system, with over 100 airlines currently managing inventory.

An airline's inventory contains all flights with their available seats. The inventory of an airline is generally divided into service classes (e.g. First, Business or Economy class) and up to 26 booking classes, for which different prices and booking conditions apply(Bilotkack, 2011). Inventory data is imported and maintained through a Schedule Distribution System over standardized interfaces. One of the core functions of the inventory management is the inventory control. Inventory control steers how many seats are available in the different booking classes, by opening and closing individual booking classes for sale. In combination with the fares and booking conditions stored in the Fare Quote System the price for each sold seat is determined(Li X, 2004). In most cases inventory control has a real time interface to an airline's Yield management system to support a permanent optimization of the offered booking classes in response to changes in demand or pricing strategies of a competitor.

Users access an airline's inventory through an availability display. It contains all offered flights for a particular city-pair with their available seats in the different booking classes. This display contains flights, which are operated by the airline itself as well as code share flights which are operated in co-operation with another airline(Shao, 2008). If the city pair is not one on which the airline offers service it may display a connection using its' own flights or display the flights of other airlines. The availability of seats of other airlines is updated through standard industry interfaces. Depending on the type of co-operation it supports access to the last seat (Last Seat Availability) in real-time. Reservations for individual passengers or groups are stored in a socalled Passenger Name Record (PNR). Among other data, the PNR contains personal information such as name, contact information or special services requests (SSRs) e.g. for a vegetarian meal, as well as the flights (segments) and issued tickets. Some reservation systems also allow to store customer data in profiles to avoid data re-entry each time a new reservation is made for a known passenger(Arsanjani, 2004). In addition, most systems have interfaces

to CRM systems or customer loyalty applications (aka Frequent Traveller Systems). Before a flight departs the so-called Passenger Name List (PNL) is handed over to the System that is used to check-in passengers and baggage. Reservation data such as the number of booked passengers and special service requests is also transferred to Flight Operations Systems, Crew Management and Catering Systems. Once a flight has departed the reservation system is updated with a list of the checked-in passengers (e.g. passengers who had a reservation but did not check in (No Shows) and passengers who checked in, but didn't have a reservation (Go Shows)). Finally, data needed for revenue accounting and reporting is handed over to the administrative systems.

A computer reservations system (CRS) is a computerized system used to store and retrieve information and conduct transactions related to air travel(Johanson, 2014). Originally designed and operated by airlines, CRSes were later extended for the use of travel agencies. Major CRS operations that book and sell tickets for multiple airlines are known as global distribution systems (GDS). Airlines have divested most of their direct holdings to dedicated GDS companies, who make their systems accessible to consumers through Internet gateways. Modern GDSes typically allow users to book hotel rooms and cars as well as airline tickets(GazetteLive, 2011). They also provide access to railway reservations in some markets although these are not always integrated with the main system.

2.2 Working mechanism of flight booking

The users can easily purchase an e-ticket by going to the ticket sale website, searching and selecting the destination, entering the details such as name, way of travel, luggage information and dates and finally making the payment via bank cards, bank transfer or through online payment companies. The e-ticket is then emailed or texted on the telephone of the customer. While previously travel agents and airlines assisted customers in making ticket purchases, today with the improved internet system, it is getting easier and easier to book the flights on your own. After the customer makes the purchase, the electronic record and the details of the ticket are saved into airline's database. The database is integrated with the passenger service system, which is then connected to the airports, airlines, travel agencies for sharing real time information.

2.3 Advantages of airline reservation system

The reason why more and more people are purchasing their flight tickets online is because of the many advantages and conveniences online flight booking system has to offer. Opposed to the traditional way of purchasing tickets through travel agents' office, online reservation today provides the access to hundreds of flight paths, their prices and other services at the click of a button. This can be very important for the customer in order to find the best possible deal and the most convenient connection. Online system of booking flight tickets gives more control to the customers in terms of their holiday planning and helps them make informed decisions. The system of searching the flights is so easy, that anybody with a computer and internet access can easily search connections, compare prices and purchase any connection they wish to. Not only this, but this method is much faster, easier, hassle-free and convenient. Many flight companies these days also provide other services such as airport transfer, accommodation services, car hire and airport parking. The customers can also take advantages of frequent discounts, promotions and other offers from the airlines through subscribing to their email or simply by viewing their website. The clients can also fully be aware of the various policies of the company such as cancellation, baggage rules and others. and can easily make use of those services. Online flight booking not only provides easy and fast services to the customer from the convenience of computer from anywhere in the world, but often also provides services at more reasonable prices. Majority of airlines companies also provide the services of online check-in after which the customer can easily print the boarding pass at home and save considerable time and hassles at the airport.

2.4 Disadvantages of Airline reservation system

Online airline reservation system has its own set of limitations as well. Prospective clients in Asia and other parts of the world where the connection speed and availability is poor may find it hard to access the website of flight companies and book tickets. In much of the less developed part of the world, besides internet connectivity, the limited knowledge and access to technology can also hinder the chances of using this service. As a result, many people in the developing world still use the traditional method of purchasing tickets from the offices of travel agents. Machine failures such as lost connection or unresponsive program can sometimes also cause the disappearance of all the flight itinerary.

Furthermore, the customer might not be able to get correct or enough information from the website alone as it is not done face to face with a person who knows everything about the connections, offers and company policies. Then there is also the issue of security: it is sometimes possible that somebody can steal your credit card information but developers are improving the security of the system more and more with the advancement of technology.

CHAPTER THREE

METHODOLOGY AND SYSTEM ANALYSIS OF THE EXISTING SYSTEM

3.0 The Research Methodology

The creation of database management system (DBMS) which ensure that computer records are kept up to date and made available on demand to those who need them for planning and operational purpose. The level of success achieved in carrying out this research work is owed to the methodology adopted. A research methodology is a systematically programming approach of a well-defined procedure that should be followed in carrying out a thorough research work .and adequately suitable methodology would ensure a very detailed research work and ensure a higher degree of accuracy and efficiency is adopted. In other to attain quite a reasonable acceptance of the research works we made use of the internationally accepted software engineering model, which is Structured System Analysis and Design Methodology.

3.1 Fact finding method used

The fact finding method used during this research are as follows

- **Review of related document**
  Important document that has one or two things to do with the study were critically studied and much needed details with close relevance to the study were taken care of and noted.

- **Interview Method**
  Some members of staff of Arik airline, Lagos were closely interviewed and their responses were found to be useful to this study.

3.2 Input, Process and Output Analysis

This deals with the process used to feed in data, work on the data and eventually gives out a resultant information.

3.2.1 Input Analysis

This deals with the process used to feed data to the system for processing. The following data are fed to the system for processing.

Tell us where you'd like to fly?

Nature of the trip, departing from, going to,

Outbound Cabin/Fare type, Returning, Inbound Cabin/Fare type

How many passengers will be travelling?

Adults (Age 12+)

Children (2-11)

Infants (Under 2)

Choose the dates you'd like to travel

Departing date Returning date

Currency

Passenger details

Title, first name, last name, mobile phone, contact no, eagle flier no, seat preference

Payment details

Email address

Confirm email address

Master card no

3.2.2 Process Analysis

Once the inputs are collected, the obtained data are processed properly for effective use. The data/information processed is stored in the computer for subsequent use.
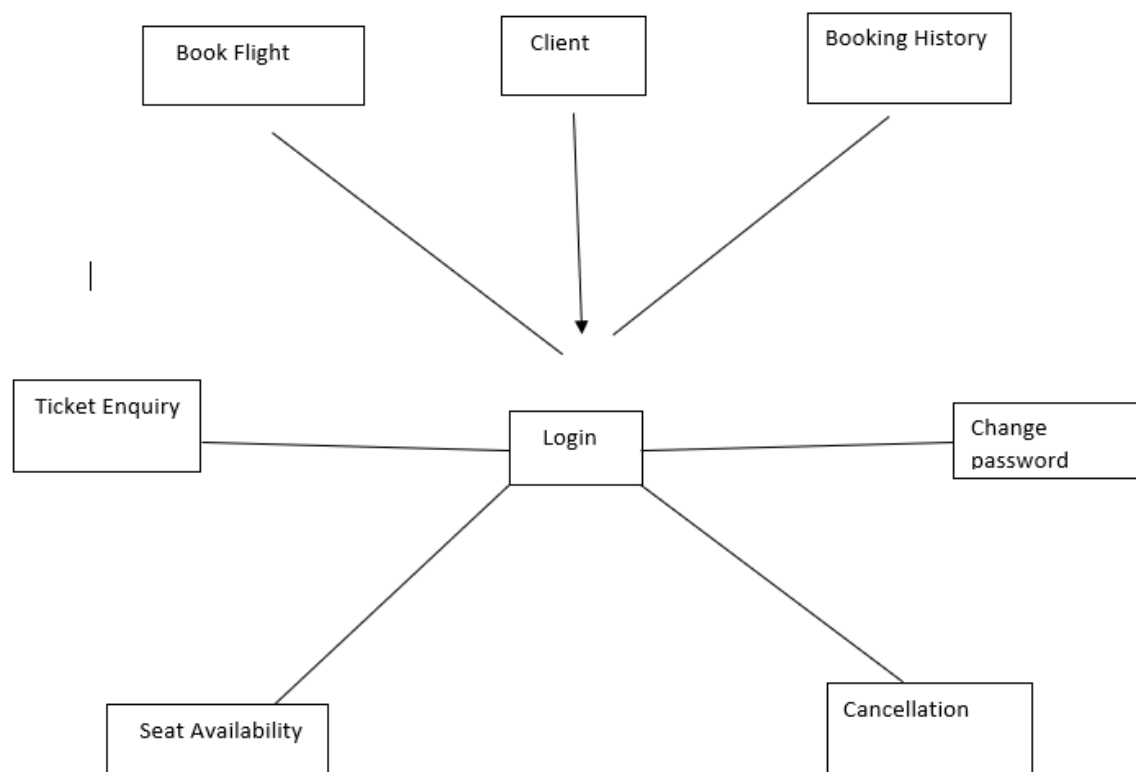

Figure 3.3: Modules available for Costumers

3.2.3 Output Analysis

This involves the resultant documentation generated after processing of data/information supplied to the system. The output here can be:

- Travel Detail
- Passenger Details
- Payment Details
- Information Flow Diagram

3.3 Problems of Existing System

- The airline information is recorded in more than one place in separate register to back it up, which increased the amount of filling procedures.
- The efforts wasted in duplication often causes lack of interest in the job.
- Making reference to records most times becomes a problem because some of the records are not easily found.

It creates avenue for doubts on the capability of the company by the customer or clients

3.4 Justification for the new System

The outcome of this study will provide a basis for developing the appropriate approach to the problems associated with air traveling operations in relation to Airline flight information system (AFIS). The following objectives are to be achieved.

- Improve forecasting accuracy,
- Freeing up labor bottlenecks,

- Reducing the amount of product handling,

- Installing improvement targets,

- Enhancing efficiency in their operation

3.5 System Features

The Airline Reservation System has the following features:

This project is mainly intended for two types of audiences. One is the customer or the end user and the other is the administrator of the website. Some of the major functions of the system can be categorized under two different categories that are for the administrator and the user.

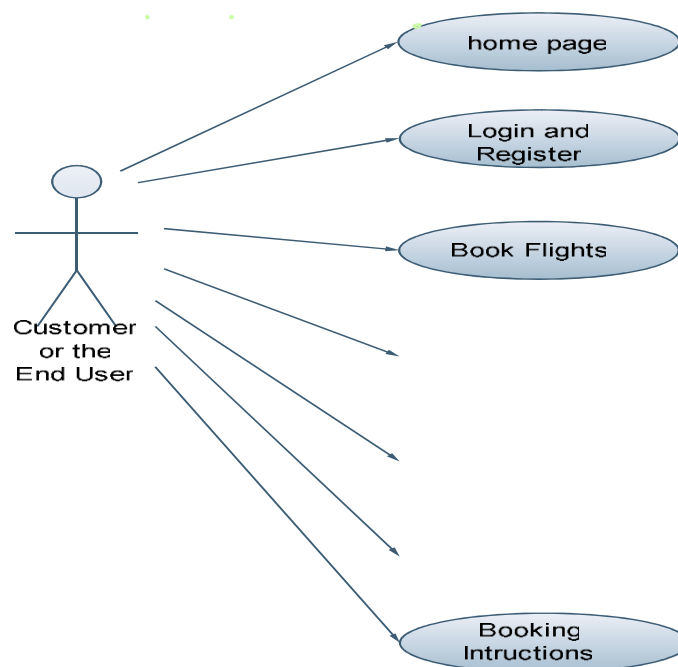### 3.5.1 Customer / End user activities

Figure 3.2 - Customer use case diagram

Figure 3.2 is a use case diagram that depicts all the functions or activities that a user or a customer can perform on the application. They can be discussed in detail as follows:

Home Page: Like all the other airline websites available online, the user can access the user home page of the Airline Reservation System website, after he logs into the system. Here, he can look up information regarding flights schedules.

Login and Register: The Airline Reservation System also comes with the customer registration details page, where the customer can enter his details and register. He can also create a username and password. Moreover, he will also be able to modify the registration information in case of a change in his e-mail address or any other information.

Payment: The customer can pay for his/her flight ticket using this system by supply inhis/her credit card details for transaction.

Cancel Flight: This is a functional module provided for user to cancel flight in case of any situation that may occur.

Update Flight: This functionality is helps the user to modify their flight reservation by changing their departing date date/time or/and arrival date/time.

Booking Instructions: The website also provides instructions to the customers on how to book airline tickets or motels along with the different packages.

Some of the functions of the Airline Reservation System, such as creating, maintaining and updating the database are available only to the administrator. The functions of the administrator, explained in detail are as follows:
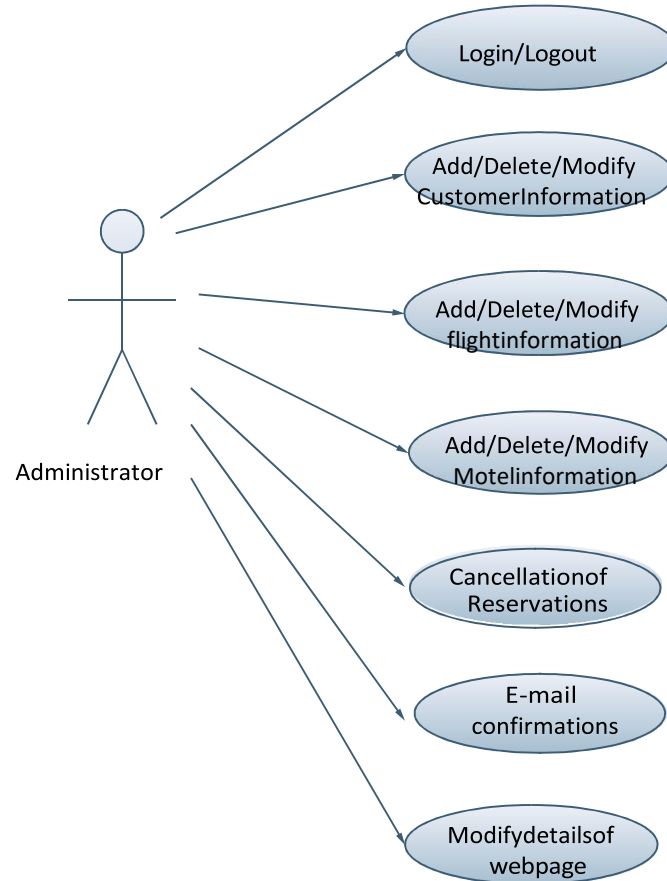
3.5.2   Administrator Activities

Figure 3.3 - Administrator use case diagram

Login/Logout: The administrator has to login first in order to be able to make changes to the Airline Reservation System, by adding, deleting or modifying the data in the Airline Reservation System database. After making the necessary changes, he then has to logout of the system, in order to prevent misuse of the data.

Add/Modify Customer Information: Daily, the Airline Reservation System will have many customers registering with the website and many of them unsubscribing. Only the administrator will have the sole rights to modify the database accordingly.

Add/Modify Flight Information: The Administrator also has the sole rights to add, delete or modify the flight information. Sometimes, flights get cancelled for some reason, so such flights would be removed from the list of flights available to the customer. Similarly, whenever any flight information has to be modified or if any new flights need to be added to the database, these operations are performed by the administrator.

Cancellation of Reservations: Sometimes, after making a reservation, a customer might cancel the reservation he has made. So, the administrator also handles such special situations and sends the customer an e-mail confirmation after deleting the specific transaction.

E-mail confirmations: Whenever a customer makes or cancels a reservation, the administrator is responsible for sending confirmation e-mails to the customer, confirming the transaction.

3.6 Functional Requirements

The functional requirements of the Airline Reservation System are divided among the customer and the administrator of the application.

These functional requirements can be explained in detail as follows:

3.6.1   Use Case name: User Registration

- **Description:** This use case describes the scenario where the user registers with the application by providing all the necessary details, in order to make reservations or bookings for flights, motels, special packages.

- **Actor:** User or the Customer

- **Input:** The user or the customer will have to provide all the necessary details present in the customer registration form of the application.

- **Output:** All the details entered in the customer registration page will be verified and accepted by the system into the database.

3.6.2 Use Case name: User Login

- **Description:** This use case describes the scenario where the user logs into the application, with the username and password he has provided while registering with the system.
- **Actor:** User or the Customer
- **Input:** The user or the customer creates a username and password at the time of registering with the system. He then uses them to logon to the system and make reservations or view any information.
- **Output:** The application then verifies the authenticity of the username and password that the customer has provided and allows the user to view the information available on the system, if the username and password are valid.

3.6.3 Use Case name: Contact the company

- **Description:** This use case describes the scenario where the user contacts the company for any information.
- **Actor:** User or the Customer
- **Input:** The customer can contact the airline company, requesting them for any information he needs.
- **Output:** The application verifies the authenticity of the username and password that the customer has provided and allows the user to view the contact information for the company

3.6.4 Use Case name: Booking Instructions

- **Description:** This use case describes the scenario where the user views the instructions for booking flights, packages, or motels.
- **Actor:** User or the Customer
- **Input:** After the customer logs onto the application with his username and password, he can look up the instructions posted on the website for booking flights, packages or motels.

- **Output:** The application verifies the authenticity of the username and password and displays the how to book instructions page.

3.6.5 Use Case name: Book Flights

- **Description:** This use case describes the scenario where the user books airline tickets.

- **Actor:** User or the Customer

- **Input:** After logging into the application, the customer looks up the information related to various airlines and checks the availability of seats on flights. If he finds that there are any available tickets, he then purchases them.

- **Output:** The application verifies the authenticity of the username and password and then displays information related to various flights to the customer.

3.6.6 Use Case name: Cancel Flight

- **Description:** This use case describes the scenario where the user cancels a reserved flight for one case or the other**.**

- **Actor:** Customer or the user

- **Input:** After logging onto the application, the customer looks up the website navigation, click flight history. The system requires the user to select the reserved flight to be cancelled.

- **Output:** The system deletes the selected flight reservation of a particular user in the user-reservation database**.**

The administrator activities use cases will be described here:

3.6.7 Use Case name: Login/Logout

- **Description:** This use case describes the scenario where the administrator of the application, logs into the system and logs out after the work is done.

- **Actor:** Administrator

- **Input:** The administrator of the website logs into the application with the username and password provided to him.

- **Output:** The application verifies the authenticity and displays the home page of the administrator.

3.6.8 Use Case name: Add/Delete or Modify Customer information

- **Description:** This use case describes the scenario where the administrator adds, deletes or modifies customer information in the system database

- **Actor:** Administrator

- **Input:** The administrator of the applications logs onto the system with his username and password.

- **Output:** The application authenticates the administrator, and then displays the page where the administrator can add new customers to the database, or delete existing customers or modify details of customers in the database.

3.6.9 Use Case name: Add/Delete or Modify flight information

- **Description:** This use case describes the scenario where the administrator adds, deletes or modifies flight information in the application database

- **Actor:** Administrator

- **Input:** The administrator logs onto the system with the username and password provided to him.

- **Output:** The application authenticates the administrator, by verifying the username and password. Then the application displays the page where the administrator can add new flights to the database, delete the flights that have been cancelled or modify information for the flights.

### 3.6.10 Use Case name: Cancellation of Reservations

- **Description:** This use case describes the scenario where the administrator handles the cancellation of reservations by the customers.

- **Actor:** Administrator

- **Input:** The administrator logs onto the system with the given username and password.

- **Output:** The application authenticates the administrator and then displays the page where the administrator looks up the id of the customer who has requested cancellation of reservation.

After canceling the reservation, the administrator then sends a confirmation e-mail to the customer.

3.6.11 Use Case name: E-mail confirmations

- **Description:** This use case describes the scenario where the administrator sends e-mail confirmations to the customers of the application.

- **Actor:** Administrator

- **Input:** The administrator logs onto the application with the username and password provided.

- **Output:** The application then authenticates the administrator and displays the page where the administrator can send e-mail confirmations to the customer. These e-mail confirmations may be sent in cases where the customer has cancelled a reservation or changed the personal information available on the website.

3.6.12 DFD (DATA FLOW DIGRAM)
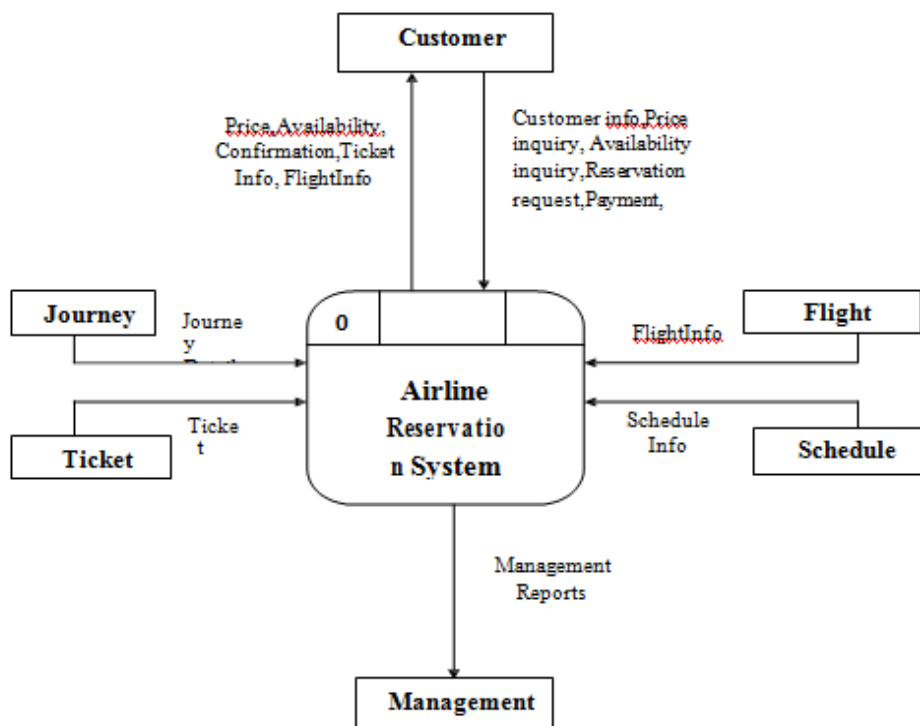
3.7 Performance Requirements

The Airline Reservation System application should be able to respond to the queries submitted by the customer without much delay. When a user searches for a flight leaving from a particular place to another place, the application should not take much time to return the results, similarly for the motel and package information. Considering that the application is of moderate size, it should be able to display 10 results at a time on each page, when the customer looks up for any particular data. Since the Airline Reservation websites have much traffic, the user should also be able to logon to the system using high speed internet. Most of the requests sent to the application should be answered in less than 5 seconds.

3.8 Non-Functional System Requirements

Performance Requirements

Some Performance requirements identified is listed below:

- The database shall be able to accommodate a minimum of 10,000 records of customer data.
- The software shall support use of multiple users at a time.
- There are no other specific performance requirements that will affect development.

3.9 Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

3.10 Security Requirements

Some of the factors that are identified to protect the software from accidental or malicious access, use, modification, destruction, or disclosure are described below.

- Keep specific log or history data sets.
- Assign certain functions to different modules.
- Restrict communications between some areas of the program.
- Check data integrity for critical variables.

Communication needs to be restricted when the application is validating the user or license. (i.e., using https).

3.11 Software Quality Attributes

The Quality of the System is maintained in such a way so that it can be very user friendly to all the users.

The software quality attributes are assumed as under:

- Accurate and hence reliable.
- Secured.
- Fast speed.
- Compatibility.

3.12 System Interfaces

This section describes how the software interfaces with other software products or users for input or output.

3.13 User Interface

Application will be accessed through a Browser Interface. The interface would be viewed best using 1024 x 768 and 800 x 600 pixels' resolution setting. The software would be fully compatible with Microsoft Internet Explorer for version 6 and above. No user would be able to access any part of the application without logging on to the system.

3.14 Hardware Interfaces

- Operating System: Windows 7/xp or above, MAC or UNIX.
- Processor: Pentium 3.0 GHz or higher
- RAM: 256 Mb or more
- Hard Drive: 10 GB or more

3.15   User Characteristics

There are two kinds of users for the Airline Reservation System. One is the customer and the other is the administrator. The customers do not need to have any prior training to use the application. However, instructions for making flight and motel reservations would be provided to them on the airline website. The administrators would however need to be trained in order to use the application.

3.16   Constraints

In case of changes made to the database, the application should be able to show the updated information on the website, without much delay. The database for the project is designed to be of moderate size.

3.17   Class Diagram
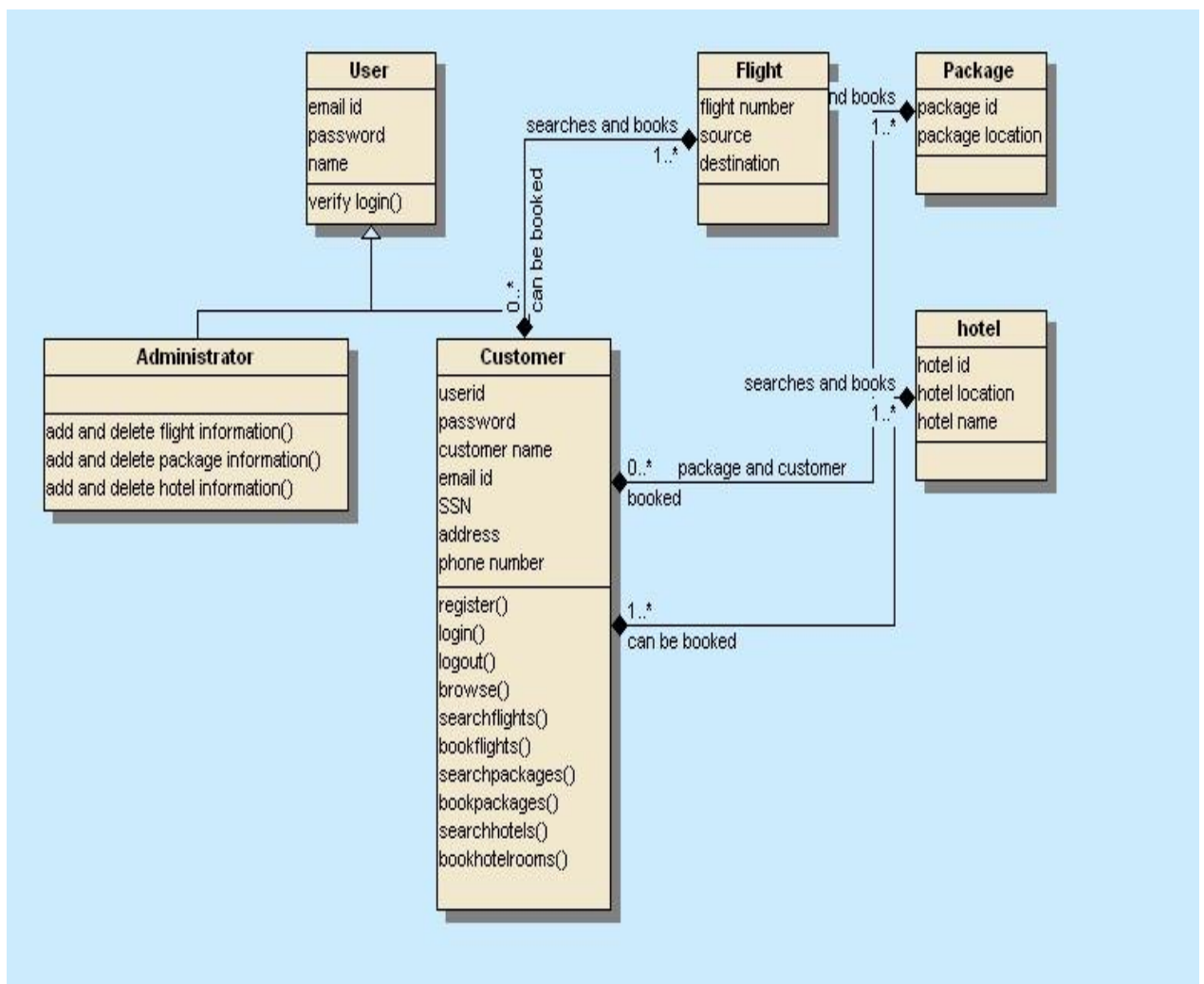
CHAPTER FOUR

SYSTEM DESIGN, CODING, TESTING, AND IMPLEMENTATION

4.0 Introduction

The system was designed using Dreamweaver platform. This platform is enriched with many programming languages such as php, c++, c#, JavaScript and much more for windows programming. At the developing stage, after undergoing research and experiment on choices of programs and databases, PHP, JavaScript and MySQL were the major programming language and database used for developing the entire system.

4.1 System Design

The system is designed to have multiple windows form. This form tends to create the unique interaction between the user and the system to support them into achieving their flight reservation task. The form has been simplified to have various object representing certain identity that enable customers accomplish the reservation and to produce the output in due time.

4.2 Customer Module Design

Main Page

The main page comprises an introduction into the view of the different type of users of the system. This page consists of different navigation button that relates the user and the system together. It includes:

- Customer (End user)
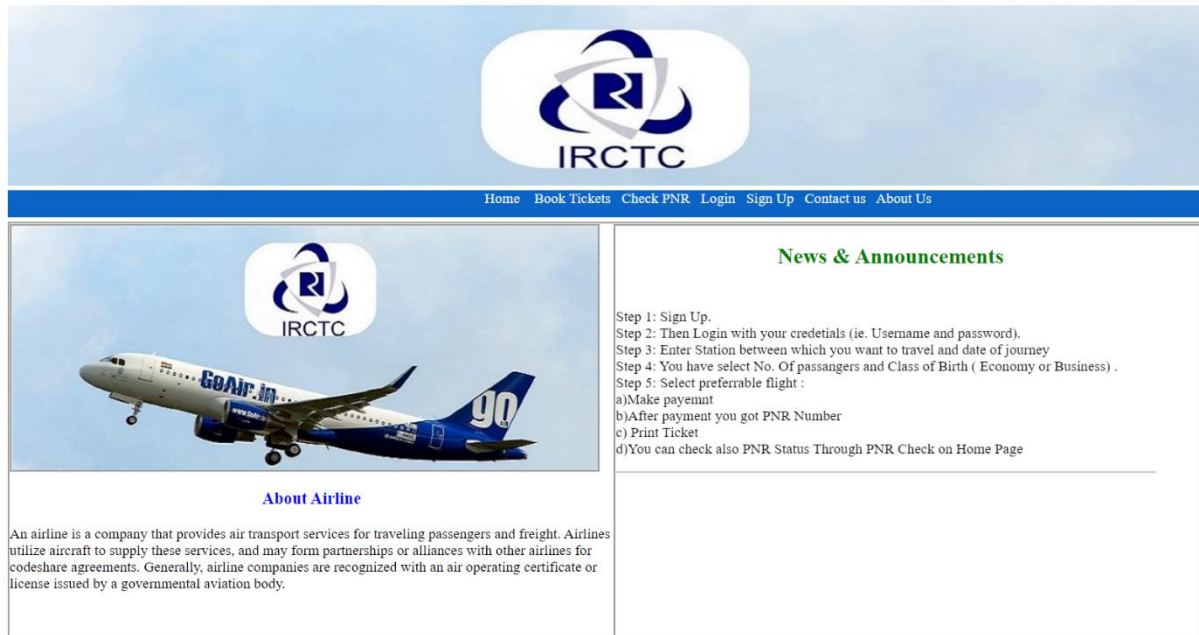- Administrator
- Feedback

Index.php

Figure 4.1: Home Page of Airline Reservation System

This is the homepage of the system which consists of a dropdown navigation button that allows both the user and the administrator to login/signup to the system

Registration and Login Feature.

After the user views the home page of the Airline Reservation System website, he can begin to browse the links available on the home page of the application. The main purpose of the Register feature is to help the customer of the website to create a login and password, which he will later use to log on to the system. The Register page will appear to the customer once he clicks the Passenger button on the home page of the application. The screen shot gives a description of the above explanation:

Sign up.php

On this page, if the user does not enter the correct details into the data fields of the registration page, then the system prompts an error message to the user asking him to enter the correct format of data into the fields.

Once all the details have been entered into the system by the user, the customer then clicks the submit button on the registration page. All the details are then accepted and verified by the system and then the user is redirected to the home page for the customer, where he can then search and book the various flights.

Login.php

Figure 4.3: Customer login form

Flight Search and Booking

The main feature of this part of the application is to enable the user to search and book for the flights available to him/her through the website. When the user clicks the flight booking link available to him on the Booking page, the user would be redirected to the Flight Search page where the user would be able to see the list of available flights. The screen shot of the Flight Search page would be as follows:

Search.php

Figure 4.4: Customer search flight page.

The user can then search for a specific flight which has a choice of his source and destination. From the radio button available to the user, the user can select a place of his choice from the source and a place of choice from the destination. If there is a flight in the database with the above choice of the customer, he can then view the schedule and the flight information of the selected choice:

Flight Booking History

This is a page that displays the user booking records of history, booked flight information. This page also consists of a function that enable user to cancel flight reservation due to one situation or the other.

Booking.php

Figure 4.5 customer flight reservation history page

4.2.2 Admin Module Design

Admin_index.php
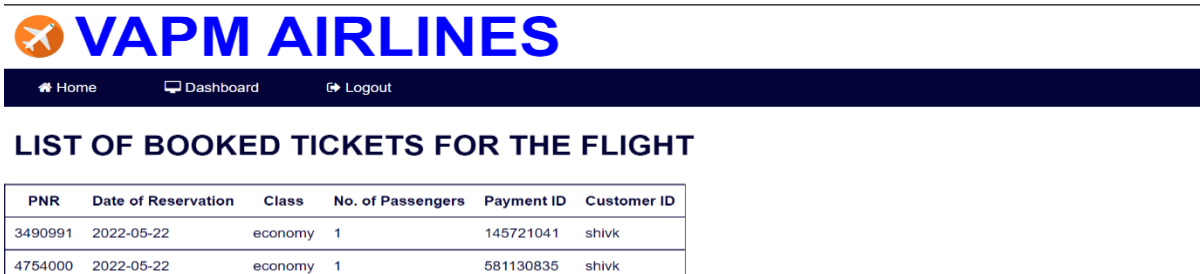
The admin is different from customer's login form; this is done to avoid deadlock during transaction. The system requires that the admin feed in his/her username and password to interact with the system to authenticate the account.

Figure 4.6 Admin login page

View_Booked_tickets.php

LIST OF BOOKED TICKETS FOR THE FLIGHT

PNR	Date of Reservation	Class	No. of Passengers	Payment ID	Customer ID
3490991	2022-05-22	economy	1	145721041	shivk
4754000	2022-05-22	economy	1	581130835	shivk

This page is designed for the administrator to view user information as they create account in order to monitor transactions.

4.3. CODING

Index.php

```
<html>

<head>

<link rel="stylesheet" type="text/css" href="styles.css"></link>

</head>

 <body>

<div class="page">

 <div class="headerpart"></div>

  <div class="linkspart">

   <center>

    <a href="index.html">Home</a>&nbsp&nbsp

<a href="login_page.php">Book Tickets</a>&nbsp&nbsp

<a href="pnrall.php">Check PNR</a>&nbsp&nbsp

<a href="login_page.php">Login</a>&nbsp&nbsp

<a href="new_user.php">Sign Up</a>&nbsp&nbsp

<a href="index.html">Contact us</a>&nbsp&nbsp

<a href="index.html">About Us</a>&nbsp&nbsp

   </center>

 </div>

<div class="bodypart">
```

<div class="imagepart"></div>

<div class="newspart">

<h2> News & Announcements </h2>

 <div class="news">

  <marquee direction="up" height="100%">

  <h4>How to Book Tickets </h4>

  <p> Step 1: Sign Up.

Step 2: Then Login with your credetials (ie. Username and password).

Step 3: Enter Station between which you want to travel and date of journey

Step 4: You have select No. Of passangers and Class of Birth (Economy or Business) .

Step 5: Select preferrable flight :

           a)Makepayemnt

           b)After payment you got PNR Number

           c) Print Ticket

           d)You can check also PNR Status Through PNR Check on Home Page

  <hr></marquee>

  more news...

</div></div>

<h3> About Airline </h3>

<p> An airline is a company that provides air transport services for traveling passengers and freight. Airlines utilize aircraft to supply these services, and may form partnerships or alliances with other airlines for codeshare agreements. Generally, airline companies are recognized with an air operating certificate or license issued by a governmental aviation body.</p>

</div><div class="footerpart">

<center> 2022 All Right Reserved. Airline</center>

</div></div>

</body></html>

home_page.php

```php
<?php

    session_start();

?>
```

```html
<html> <head>

 <title>

        Welcome to Airlines

    </title>

    <link rel="stylesheet" type="text/css" href="css/style.css"/>

    <link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css">

  </head><body>

    <img class="logo" src="images/irctc.jpg"/>

    <h1 id="title">

      VAPM Airways

    </h1><div>        <ul>

 <li><a href="home_page.php"><i class="fa fa-home" aria-hidden="true"></i> Home</a></li><li>
```

```php
 <?php

if(isset($_SESSION['login_user'])&&$_SESSION['user_type']=='Customer')

{
```

```php
echo "<a href=\"book_tickets.php\"><i class=\"fa fa-ticket\" aria-hidden=\"true\"></i> Book
Tickets</a>";}

else

if(isset($_SESSION['login_user'])&&$_SESSION['user_type']=='Administrator') {

echo "<a href=\"admin_ticket_message.php\"><i class=\"fa fa-ticket\" aria-hidden=\"true\"></i> Book
Tickets</a>";                }

              else                {

echo "<a href=\"login_page.php\"><i class=\"fa fa-ticket\" aria-hidden=\"true\"></i> Book
Tickets</a>";                }

 ?></li>

<li><a href="home_page.php"><i class="fa fa-plane" aria-hidden="true"></i> About Us</a></li>

<li><a href="home_page.php"><i class="fa fa-phone" aria-hidden="true"></i> Contact Us</a></li>

<li><a href="pnrall.php"><i class="fa fa-ticket" aria-hidden="true"></i> Check PNR </a></li><li>

<?php

if(isset($_SESSION['login_user'])&&$_SESSION['user_type']=='Customer') {

echo "<a href=\"customer_homepage.php\"><i class=\"fa fa-sign-in\" aria-hidden=\"true\"></i>
Login</a>"; }

else if(isset($_SESSION['login_user'])&&$_SESSION['user_type']=='Administrator')

{

 echo "<a href=\"admin_homepage.php\"><i class=\"fa fa-sign-in\" aria-hidden=\"true\"></i>
Login</a>";                }

else                {



echo "<a href=\"login_page.php\"><i class=\"fa fa-sign-in\" aria-hidden=\"true\"></i> Login</a>"}
```

?>   </div>

<div class="container">

 <div class="welcome_text">Welcome to VAPM !</div>

<imgsrc="images/1.png" width=100%>

 </div><!--check out addling local host in links and other places

 shift login/logout buttons to right side

-->   </body></html>

Login_page.php

<?php session_start();?>

<html>   <head>

    <title> Account Login   </title>

<style>

input {

        border: 1.5px solid #030337;

        border-radius: 4px;

        padding: 7px 30px;

    }

input[type=submit] {

        background-color: #030337;

        color: white;

        border-radius: 4px;

        padding: 7px 45px;

margin: 0px 60px

    } </style>

<link rel="stylesheet" type="text/css" href="css/style.css"/> <link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css">

</head> <body>

<h1 id="title">VAPM Airways </h1>

<div>

<i class="fa fa-home" aria-hidden="true"></i> Home

<i class="fa fa-ticket" aria-hidden="true"></i> Book Tickets

<i class="fa fa-plane" aria-hidden="true"></i> About Us

<i class="fa fa-phone" aria-hidden="true"></i> Contact Us

<i class="fa fa-sign-in" aria-hidden="true"></i> Login </div>

<form class="float_form" style="padding-left: 40px" action="login_handler.php" method="POST">

<fieldset>

<legend>Login Details:-</legend>Username:

<input type="text" name="username" placeholder="Enter your username" required>

    Password:
<input type="password" name="password" placeholder="Enter your password" required>

User Type:

Customer <input type='radio' name='user_type' value='Customer' checked/> Administrator <input type='radio' name='user_type' value='Administrator'/>

```php
<?php

   if(isset($_GET['msg']) && $_GET['msg']=='failed')          {

      echo "<br>

      <strong style='color:red'>Invalid Username/Password</strong>

      <br><br>";          }?>
```

<input type="submit" name="Login" value="Login">

</fieldset>

<i class="fa fa-user-plus" aria-hidden="true"></i> Create New User Account?   </form>   </body></html>

New_user.php

```html
<html>   <head>      <title> Create New User Account</title>

   <style>

   input {

      border: 1.5px solid #030337;

      border-radius: 4px;

      padding: 7px 30px;          }


   input[type=submit] {

      background-color: #030337;

      color: white;

      border-radius: 4px;

      padding: 7px 45px;
```

```
        margin: 0px 135px          }          </style>

    <link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-
awesome.min.css">   </head>   <body>        <img class="logo" src="images/shutterstock_22.jpg"/>

    <h1 id="title">          VAPMS Airways      </h1>        <div>          <ul>

        <li><a href="home_page.php"><i class="fa fa-home" aria-hidden="true"></i>
Home</a></li>

        <li><a href="login_page.php"><i class="fa fa-ticket" aria-hidden="true"></i> Book
Tickets</a></li>

        <li><a href="home_page.php"><i class="fa fa-plane" aria-hidden="true"></i> About
Us</a></li>

        <li><a href="home_page.php"><i class="fa fa-phone" aria-hidden="true"></i> Contact
Us</a></li>

        <li><a href="login_page.php"><i class="fa fa-sign-in" aria-hidden="true"></i>
Login</a></li></ul></div>        <br>

<form class="center_form" action="new_user_form_handler.php" method="POST"
id="new_user_from">

      <h2><i class="fa fa-user-plus" aria-hidden="true"></i> CREATE NEW USER
ACCOUNT</h2>          <br>

<table cellpadding='10'><strong>ENTER LOGIN DETAILS</strong><tr>

<td>Enter a valid username  </td>

<td><input type="text" name="username" required><br><br></td></tr><tr>

<td>Enter your desired password  </td>

<td><input type="password" name="password" required><br><br></td></tr><tr>

<td>Enter your email ID</td>

 <td><input type="text" name="email" required><br><br></td></tr> </table>          <br>
```

<table cellpadding='10'>

ENTER CUSTOMER'S PERSONAL DETAILS   <tr>

<td>Enter your name </td><td><input type="text" name="name" required>

</td>   </tr><tr><td>Enter your phone no.</td>

<td><input type="text" name="phone_no" required>

</td> </tr><tr>

 <td>Enter your address</td>

<td><input type="text" name="address" required>

</td></tr></table>

<input type="submit" value="Submit" name="Submit">
 </form> </body></html>

Admin_homepage.php

<?php session_start();?>

<html> <head>   <title>   Welcome Administrator   </title>

<link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css"></head> <body><h1 id="title">

VAPM Airways   </h1>   <div>

<i class="fa fa-home" aria-hidden="true"></i> Home

<i class="fa fa-desktop" aria-hidden="true"></i> Dashboard

<i class="fa fa-sign-out" aria-hidden="true"></i> Logout </div><h2>Welcome Administrator!</h2><table cellpadding="5">

<tr> <td class="admin_func"><i class="fa fa-plane" aria-hidden="true"></i> View List of Booked Tickets for a Flight</td></tr>   <tr>

<td class="admin_func"><i class="fa fa-plane" aria-hidden="true"></i> Add Flight Schedule Details</td></tr><!--<tr>

```
<td class="admin_func"><a href="modify_flight_details.php"><i class="fa fa-plane" aria-hidden="true"></i> Modify Flight Schedule Details</a>

</td></tr> --><tr>

<td class="admin_func"><a href="delete_flight_details.php"><i class="fa fa-plane" aria-hidden="true"></i> Delete Flight Schedule Details</a></td></tr><tr>

<td class="admin_func"><a href="add_jet_details.php"><i class="fa fa-plane" aria-hidden="true"></i> Add Aircrafts Details</a></td></tr><tr>

<td class="admin_func"><a href="activate_jet_details.php"><i class="fa fa-plane" aria-hidden="true"></i> Activate Aircraft</a></td></tr><tr>

<td class="admin_func"><a href="deactivate_jet_details.php"><i class="fa fa-plane" aria-hidden="true"></i> Deactivate Aircraft</a></td></tr></table></body></html>
```

Customer_homepage.php

```
<?php session_start();

// if($_SESSION['login_user']==null){

// header('location:home_page.php');    // } ?>

 <html><head><title> Welcome Customer</title>

 <style>

input {

        border: 1.5px solid #030337;

        border-radius: 4px;

        padding: 7px 30px; }

input[type=submit] {

        background-color: #030337;
```

```
        color: white;

        border-radius: 4px;

        padding: 7px 45px;

        margin: 0px 127px}

input[type=date] {

        border: 1.5px solid #030337;

        border-radius: 4px;

        padding: 5.5px 44.5px; }


select {

        border: 1.5px solid #030337;

        border-radius: 4px;

        padding: 6.5px 75.5px; }


</style><link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css"></head>

<body> <img class="logo" src="images/irctc.jpg"/><h1 id="title">VAPM Airways

</h1><div><ul>

<li><a href="customer_homepage.php"><i class="fa fa-home" aria-hidden="true"></i>
Home</a></li>

<li><a href="customer_homepage.php"><i class="fa fa-desktop" aria-hidden="true"></i>
Dashboard</a></li>

<li><a href="pnr.php"><i class="fa fa-desktop" aria-hidden="true"></i> Print Ticket</a></li>
```

```
<li><a href="customer_homepage.php"><i class="fa fa-plane" aria-hidden="true"></i> About Us</a></li>

<li><a href="customer_homepage.php"><i class="fa fa-phone" aria-hidden="true"></i> Contact Us</a></li>

<li><a href="logout_handler.php"><i class="fa fa-sign-out" aria-hidden="true"></i> Logout</a></li>     </ul>   </div>

<?php

echo "<h2>Welcome ".$_SESSION['login_user']."</h2>";

require_once('Database Connection file/mysqli_connect.php');

     $query="SELECT count(*),frequent_flier_no,mileage FROM Frequent_Flier_Details WHERE customer_id=?";

$stmt=mysqli_prepare($dbc,$query);

mysqli_stmt_bind_param($stmt,"s",$_SESSION['login_user']);

mysqli_stmt_execute($stmt);

mysqli_stmt_bind_result($stmt,$cnt,$frequent_flier_no,$mileage);

mysqli_stmt_fetch($stmt);

     if($cnt==1) {

        echo "<h4 style='padding-left: 20px;'>Frequent Flier No.: ".$frequent_flier_no."                       Mileage: ".$mileage." points</h4><br>"; }

mysqli_stmt_close($stmt);

mysqli_close($dbc); ?> <table cellpadding="5"><tr>

<td class="admin_func"><a href="book_tickets.php"><i class="fa fa-plane" aria-hidden="true"></i> Book Flight Tickets</a></td></tr><tr>
```

<td class="admin_func"><i class="fa fa-plane" aria-hidden="true"></i> View Booked Flight Tickets</td></tr><tr>

<td class="admin_func"><i class="fa fa-plane" aria-hidden="true"></i> Print Booked Ticket </td></tr><tr>

<td class="admin_func"><i class="fa fa-plane" aria-hidden="true"></i> Cancel Booked Flight Tickets </td> </tr> </table>

<!--Following data fields were empty!   ...

ADD VIEW FLIGHT DETAILS AND VIEW JETS/ASSETS DETAILS for ADMIN

PREDEFINED LOCATION WHEN BOOKING TICKETS--></body></html>

Add_flight_details.php

<?php session_start();?><html><head><title> Add Flight Schedule Details</title><style>

input {

border: 1.5px solid #030337;

border-radius: 4px;

padding: 7px 30px;   }

input[type=submit] {

background-color: #030337;

color: white;

border-radius: 4px;

padding: 7px 45px;

margin: 0px 200px   }   </style>

<link rel="stylesheet" type="text/css" href="css/style.css"/>

 <link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css">

</head> <body> <h1 id="title">

VAPM AIRLINES </h1> <div>

<i class="fa fa-home" aria-hidden="true"></i> Home

<i class="fa fa-desktop" aria-hidden="true"></i> Dashboard

<i class="fa fa-sign-out" aria-hidden="true"></i> Logout </div> <form action="add_flight_details_form_handler.php" method="post">

<h2>ENTER THE FLIGHT SCHEDULE DETAILS</h2>

 <?php   if(isset($_GET['msg']) && $_GET['msg']=='success') {

echo "<strong style='color: green'>The Flight Schedule has been successfully added.

"; }

else if(isset($_GET['msg']) && $_GET['msg']=='failed') {

echo "<strong style='color: red'>*Invalid Flight Schedule Details, please enter again.

"; } ?>

<table cellpadding="5"><tr><td class="fix_table">Flight Number</td> </tr><tr>   <td class="fix_table"><input type="text" name="flight_no" required></td> </tr>

</table>
<hr><table cellpadding="5"><tr><td class="fix_table">Origin</td>   <td class="fix_table">Destination</td > </tr> <tr>

<td class="fix_table"><input type="text" name="origin" required></td>

<td class="fix_table"><input type="text" name="destination" required></td> </tr>

</table>
<hr><table cellpadding="5"><tr><td class="fix_table">Departure Date</td>   <td class="fix_table">Arrival Date</td></tr><tr>

<td class="fix_table"><input type="date" name="dep_date" required></td>

<td class="fix_table"><input type="date" name="arr_date" required></td> </tr>

```
</table><br><hr><table cellpadding="5"><tr><td class="fix_table">Departure Time</td>

<td class="fix_table">Arrival Time</td></tr> <tr>

<td class="fix_table"><input type="time" name="dep_time" required></td>

<td class="fix_table"><input type="time" name="arr_time" required></td>

</tr> </table> <br> <hr><table cellpadding="5"> <tr>

<td class="fix_table">Number of Seats in Economy Class</td>

<td class="fix_table">Number of Seats in Business Class</td>

</tr> <tr>

<td class="fix_table"><input type="number" name="seats_eco" required></td>          <td
class="fix_table"><input type="number"" name="seats_bus" required></td>

</tr></table><br><hr><table cellpadding="5"><tr>

<td class="fix_table">Ticket Price(Economy Class)</td>

<td class="fix_table">Ticket Price(Business Class)</td> </tr> </table>

 <table cellpadding="5"><tr><td class="fix_table">

<input type="number" name="price_eco" required> </td>

<td class="fix_table">  <input type="number" name="price_bus" required>

</td></tr> </table><br><hr> <table cellpadding="5"> <tr>

  <td class="fix_table">Jet ID</td></tr><tr><td class="fix_table"><input type="text" name="jet_id"
required>

</td></tr></table><br><input type="submit" value="Submit" name="Submit">

</form><!--check out addling local host in links and other places--></body></html>
```

Add_jet_details.php

```
<?php session_start(); ?>
```

```
<html><head><title> Add Aircrafts Details </title><style>

input {

        border: 1.5px solid #030337;

        border-radius: 4px;

        padding: 7px 30px; }

input[type=submit] {

        background-color: #030337;

        color: white;

        border-radius: 4px;

        padding: 7px 45px;

        margin: 0px 60px}

</style>

<link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css">

</head><body><img class="logo" src="images/shutterstock_22.jpg"/>

<h1 id="title">VAPM AIRLINES</h1><div><ul>

<li><a href="admin_homepage.php"><i class="fa fa-home" aria-hidden="true"></i> Home</a></li>

<li><a href="admin_homepage.php"><i class="fa fa-desktop" aria-hidden="true"></i>
Dashboard</a></li>

<li><a href="logout_handler.php"><i class="fa fa-sign-out" aria-hidden="true"></i>
Logout</a></li></ul></div>

<form action="add_jet_details_form_handler.php" method="post">

<h2>ENTER THE AIRCRAFTS DETAILS</h2><div>

<?php
```

if(isset($_GET['msg']) && $_GET['msg']=='success') {

echo "<strong style='color: green'>The Aircraft has been successfully added.

";}

else if(isset($_GET['msg']) && $_GET['msg']=='failed') {

echo "<strong style='color:red'>*Jet ID already exists, please enter a new Jet ID.

";}?><table cellpadding="5"><tr><td class="fix_table">Enter a valid Jet ID</td>

</tr><tr><td class="fix_table"><input type="text" name="jet_id" required></td>

</tr></table>
<table cellpadding="5"><tr><td class="fix_table">Enter the Jet Type/Model</td></tr><tr>

<td class="fix_table"><input type="text" name="jet_type" required></td>

</tr></table>
<table cellpadding="5"><tr><td class="fix_table">Enter the total capacity of the Jet</td></tr><tr><td class="fix_table"><input type="number" name="jet_capacity" required></td></tr>

</table>

<input type="submit" value="Submit" name="Submit">

</div></form></body></html>

Flightlist.php

<?phpsession_start();

$con=mysqli_connect("localhost","root","","airline_reservation");

if(!isset($con)) {

die("Database Not Found");}

if(isset($_REQUEST["u_sub"])){

$id=$_POST['pnr'];

 if($id!='') {

```php
$query=mysqli_query($con ,"select * from flight_details where departure_date='".$id."'");

$res=mysqli_fetch_row($query);

$query0=mysqli_query($con ,"select * from flight_details where departure_date='".$id."'");

$res0=mysqli_fetch_row($query0);

$query1=mysqli_query($con ,"select * from flight_details where departure_date='".$id."'");

$res1=mysqli_fetch_row($query1);

if($res{

 $_SESSION['user']=$id;

 header('location:flightcheck.php'); }

 else{

 echo '<script>';

 echo 'alert("No Flight")';

 echo '</script>';   }

if($res0)  {

 $_SESSION['user']=$id;

 header('location:flightcheck.php');  }

 else  {

 echo '<script>';

 echo 'alert("No Flight")';

 echo '</script>';   }

 if($res1)  {

 $_SESSION['user']=$id;

 header('location:flightcheck.php');  }
```

```
else  {

  echo '<script>';

  echo 'alert("No Flight")';

  echo '</script>';   }  }

 else {

echo '<script>';

echo 'alert("No Flight")';

echo '</script>';  }}?>

<html><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<link type="text/css" rel="stylesheet" href="css/login.css"></link>

<link rel="stylesheet" href="bootstrap/bootstrap.min.css">

<link rel="stylesheet" href="bootstrap/bootstrap-theme.min.css">

<script src="bootstrap/jquery.min.js"></script>

<script src="bootstrap/bootstrap.min.js"></script>        <title></title>

</head><body  style="background-image:url('./images/inbg.jpg');" >

<form id="index" action="flightlist.php" method="post">

<div class="container-fluid">    <div class="row"><div class="col-sm-12">

</div></div><div  id="divtop"><center><br><br><br><div id="dmain"  >

<center><imgsrc="./images/irctc.jpg" width="180px" height="150px" ></center>

<br><input type="text" id="u_id" name="pnr" class="form-control" style="width:300px; margin-left:
66px;" placeholder="YYYY-MM-DD"><br>

<input type="submit" id="u_sub" name="u_sub" value="Check Flights" class="toggle btnbtn-primary"
style="width:100px; margin-left: 70px;"><b

</div></div></div></div></div>  </div></form>  </body></html>
```

Ticket_sucess.php

```php
<?php  session_start(); ?>

<html><head><title>Ticket Booking Successful</title><style>

input {

border: 1.5px solid #030337;

border-radius: 4px;

padding: 7px 30px; }

input[type=submit] {

background-color: #030337;

color: white;

border-radius: 4px;

 padding: 7px 45px;

margin: 0px 127px}</style>

<link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css">

</head><body><img class="logo" src="images/shutterstock_22.jpg"/>

<h1 id="title">VAPM AIRLINES</h1><div><ul>

<li><a href="customer_homepage.php"><i class="fa fa-home" aria-hidden="true"></i> Home</a></li>

<li><a href="customer_homepage.php"><i class="fa fa-desktop" aria-hidden="true"></i>
Dashboard</a></li>

<li><a href="home_page.php"><i class="fa fa-plane" aria-hidden="true"></i> About Us</a></li>

<li><a href="home_page.php"><i class="fa fa-phone" aria-hidden="true"></i> Contact Us</a></li>
```

<i class="fa fa-sign-out" aria-hidden="true"></i> Logout
</div><h2>BOOKING SUCCESSFUL</h2>

<h3>Your payment of ₹ <?php echo $_SESSION['total_amount']; ?> has been received.

 Your PNR is <?php echo $_SESSION['pnr'];?>. Your tickets have been booked successfully.</h3>

<!--Following data fields were empty...

ADD VIEW FLIGHT DETAILS AND VIEW JETS/ASSETS DETAILS for ADMIN-->

</body></html>

View_booked_tickets.php

<?phpsession_start();?><html><head><title>View Booked Tickets</title>

<style>

input {

border: 1.5px solid #030337;

border-radius: 4px;

padding: 7px 30px; }

input[type=submit] {

background-color: #030337;

color: white;

border-radius: 4px;

padding: 7px 45px;

margin: 0px 390px}

table {

border-collapse: collapse;

```
        margin-left: 10%;

        margin-right: 10%;}

    tr/*:nth-child(3)*/ {

    border: solid thin; }

    .set_nice_size{

        font-size: 17pt; }
```

```html
</style><link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css">

</head><body><img class="logo" src="images/shutterstock_22.jpg"/>

<h1 id="title">VAPM AIRLINES</h1><div><ul>

<li><a href="customer_homepage.php"><i class="fa fa-home" aria-hidden="true"></i> Home</a></li>

<li><a href="customer_homepage.php"><i class="fa fa-desktop" aria-hidden="true"></i>
Dashboard</a></li>

<li><a href="home_page.php"><i class="fa fa-plane" aria-hidden="true"></i> About Us</a></li>

<li><a href="home_page.php"><i class="fa fa-phone" aria-hidden="true"></i> Contact Us</a></li>

<li><a href="logout_handler.php"><i class="fa fa-sign-out" aria-hidden="true"></i>
Logout</a></li></ul>

</div><h2>VIEW BOOKED FLIGHT TICKETS</h2><h3
class='set_nice_size'><center><u>Upcoming Trips</u></center></h3>
```

```php
<?php

    $todays_date=date('Y-m-d');

    $thirty_days_before_date=date_create(date('Y-m-d'));

    date_sub($thirty_days_before_date,date_interval_create_from_date_string("30 days"));

$thirty_days_before_date=date_format($thirty_days_before_date,"Y-m-d");
```

```php
$customer_id=$_SESSION['login_user'];

require_once('Database Connection file/mysqli_connect.php');

$query="SELECT
pnr,date_of_reservation,flight_no,journey_date,class,booking_status,no_of_passengers,payment_id
FROM Ticket_Details where customer_id=? AND journey_date>=? AND
booking_status='CONFIRMED' ORDER BY  journey_date";

$stmt=mysqli_prepare($dbc,$query);

mysqli_stmt_bind_param($stmt,"ss",$customer_id,$todays_date);

mysqli_stmt_execute($stmt);

mysqli_stmt_bind_result($stmt,$pnr,$date_of_reservation,$flight_no,$journey_date,$class,$booking_status,$no_of_passengers,$payment_id);

mysqli_stmt_store_result($stmt);

if(mysqli_stmt_num_rows($stmt)==0) {

    echo "<h3><center>No upcoming trips!</center></h3>";}

    else{

echo "<table cellpadding=\"10\"";echo "<tr><th>PNR</th><th>Date of Reservation</th>

<th>Flight No.</th><th>Journey Date</th><th>Class</th><th>Booking Status</th>

<th>No. of Passengers</th><th>Payment ID</th></tr>";while(mysqli_stmt_fetch($stmt)) {

echo "<tr><td>".$pnr."</td><td>".$date_of_reservation."</td><td>".$flight_no."</td>

<td>".$journey_date."</td><td>".$class."</td><td>".$booking_status."</td>

<td>".$no_of_passengers."</td<td>".$payment_id."</td></tr>";}

echo "</table><br>";}

echo "<br><h3 class=\"set_nice_size\"><center><u>Completed Trips</u></center></h3>";

$query="SELECT
pnr,date_of_reservation,flight_no,journey_date,class,booking_status,no_of_passengers,payment_id
```

```php
FROM Ticket_Details where customer_id=? and journey_date<? and journey_date>=? ORDER BY journey_date";

$stmt=mysqli_prepare($dbc,$query);          mysqli_stmt_bind_param($stmt,"sss",$customer_id,$todays_date,$thirty_days_before_date);

mysqli_stmt_execute($stmt);          mysqli_stmt_bind_result($stmt,$pnr,$date_of_reservation,$flight_no,$journey_date,$class,$booking_status,$no_of_passengers,$payment_id);

        mysqli_stmt_store_result($stmt);

        if(mysqli_stmt_num_rows($stmt)==0) {

echo "<h3><center>No trips completed in the past 30 days!</center></h3>";}

else{

        echo "<table cellpadding=\"10\"";

        echo "<tr><th>PNR</th>

<th>Date of Reservation</th><th>Flight No.</th><th>Journey Date</th><th>Class</th>

<th>Booking Status</th><th>No. of Passengers</th><th>Payment ID</th></tr>";

while(mysqli_stmt_fetch($stmt)) {

echo "<tr><td>".$pnr."</td><td>".$date_of_reservation."</td>

<td>".$flight_no."</td><td>".$journey_date."</td><td>".$class."</td>

<td>".$booking_status."</td>

<td>".$no_of_passengers."</td><td>".$payment_id."</td></tr>";}

echo "</table><br>";}mysqli_stmt_close($stmt); mysqli_close($dbc);?>

</body></html>
```

Cancel_booked_tickets.php

```php
<?phpsession_start();?><html><head><title>Cancel Booked Tickets</title><style>
```

```
input {

        border: 1.5px solid #030337;

        border-radius: 4px;

        padding: 7px 30px; }

    input[type=submit] {

        background-color: #030337;

        color: white;

        border-radius: 4px;

        padding: 7px 45px;

        margin: 0px 68px}
```

</style><link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css">

</head><body>

<h1 id="title">VAPM AIRLINES</h1><div>

<i class="fa fa-home" aria-hidden="true"></i> Home

<i class="fa fa-desktop" aria-hidden="true"></i> Dashboard

<i class="fa fa-plane" aria-hidden="true"></i> About Us

<i class="fa fa-phone" aria-hidden="true"></i> Contact Us

<i class="fa fa-sign-out" aria-hidden="true"></i> Logout
</div>

<form action="cancel_booked_tickets_form_handler.php" method="post">

<h2>CANCEL BOOKED TICKETS</h2><?php

if(isset($_GET['msg']) && $_GET['msg']=='failed') {

echo "<strong style='color: red'>*Invalid PNR, please enter PNR again

";}

?> <table cellpadding="5" style="padding-left: 30px;"> <tr><td class="fix_table">Enter the PNR</td></tr><tr><td class="fix_table"><input type="text" name="pnr" required></td></tr></table>
<input type="submit" value="Cancel Ticket" name="Cancel_Ticket">

</form><!--Following data fields were empty!...

ADD VIEW FLIGHT DETAILS AND VIEW JETS/ASSETS DETAILS for ADMIN

PREDEFINED LOCATION WHEN BOOKING TICKETS--></body></html>

Cancel_booked_tickets_handler.php

<?phpsession_start();?><html><head><title>Cancel Booked Tickets</title><style>

input {

       border: 1.5px solid #030337;

       border-radius: 4px;

       padding: 7px 30px; }

    input[type=submit] {

       background-color: #030337;

       color: white;

       border-radius: 4px;

       padding: 7px 45px;

       margin: 0px 68px}

</style><link rel="stylesheet" type="text/css" href="css/style.css"/>

<link rel="stylesheet" href="font-awesome-4.7.0\css\font-awesome.min.css"></head><body>

<h1 id="title">VAPM AIRLINES</h1><div>

<i class="fa fa-home" aria-hidden="true"></i> Home

<i class="fa fa-desktop" aria-hidden="true"></i> Dashboard

<i class="fa fa-plane" aria-hidden="true"></i> About Us

<i class="fa fa-phone" aria-hidden="true"></i> Contact Us

<i class="fa fa-sign-out" aria-hidden="true"></i> Logout
</div><h2>CANCEL BOOKED TICKETS</h2>

<h3 style='padding-left: 40px;'>Your ticket has been cancelled successfully.

Your amount of ₹ <?php echo $_SESSION['refund_amount']?> will be refunded to your bank account (Cancellation charge on 15% of your ticket amount has been deducted).</td>

</h3>
<!--Following data fields were empty!...

ADD VIEW FLIGHT DETAILS AND VIEW JETS/ASSETS DETAILS for ADMIN

PREDEFINED LOCATION WHEN BOOKING TICKETS--></body></html>

4.4 Testing

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. Testing is the process of executing a program with the explicit intention of finding errors that is making the program fail. The tester maybe an analyst, programmer or a specialist trained for software testing.

4.4.1   Unit Testing

In unit testing we have to test the programs making up the system. For this reason, unit testing is sometimes called as the Program testing. The software units in a system are modules and routines that are assembled and integrated to perform a specific function. Unit testing focuses first on modules, independently of one another, to locate errors. This enables, to detect errors in coding and logic that are contained within the module alone. Unit testing can be performed from the bottom up, starting with the lowest level modules and proceeding one at a time. Unit testing is done for each module in Airline Reservation System. This ensures that the value we enter match with the data type and within the specified limits.

4.4.2   Integration Testing

Data can be lost across any interface, one module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integration testing is a systematic testing for conducting tests to uncover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here correction is difficult because the vast expenses of the entire program complicate the isolation of causes. Thus in the integration testing step, all the errors are corrected for the next testing steps. In Airline Reservation System each module is integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data.

4.4.3   Validation Testing

At the culmination of integration testing, software is completely assembled as a package; interfacing errors have been recovered and corrected and a final series of a software tests-validation tests begin. Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably expected by the customer.

4.5   Implementation

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation, of change over methods. The more complex system being implemented, the more involved will be the system analysis and the design effort required just for implementation.

An implementation co-ordination committee based on policies of individual organization has been appointed. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions are made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. This is the most critical stage in achieving a successful new system and in giving the users confidence that the new system will work is effective. The system can be implemented only after thorough testing. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

CHAPTER FIVE

SUMMARY, RECOMMENDATION AND CONCLUSION

5.0 Summary

A new computer system for an Airline reservation has been designed and implemented in this project work. The exercise was carried out based on the loopholes existing in the reservation of air Transport Service. Number of problems encountered in the manual service process thereby creating opportunity for:

- Reliability in the management of customer's document.
- Fast discharge of auditing responsibilities.
- Accuracy of computations.
- Provision of easy adjustment and update of customer records.

5.1 Recommendation

For this research work, I recommend a parallel change over, parallel change over means a situation where by the old and the new way of working is implemented. That is, the old (manual) and the newly designed software computerized for transaction of business between the company and her customers are being run.

This is to enable a sound understanding of the new system design for the effectiveness of this changeover; there are other tasks that must be accomplished. They include the following.

- Installation of cooling devices such as the air condition for cooling the computer system
- The company should be sending their staff for training so that they acquire more skills and experience in operating the new designed system
- Provision of security measures for the system e.g. Alarm gadgets and burglary proof should be in place.
- Installation of telephone services is needed so that there won't be any problem in communication.
- Provision of false floor and false ceiling.
- Provision of house cleaner to keep the computer room neat and dust free.

5.2 Conclusion

The software package "Airline Reservation System" provides convenient online uploading the report from executive and viewing that report by the managing director in an online fashion.

To input the data in a highly validated manner and generating the different reports, this involves complex process that was being done on based manner. This package is designed and developed in a compact manner, which is ready to meet the user's specification and to serve them in an effective as well as in an enhanced manner. The actual problem has been observed with keen interest and it has been defined and analyzed in such a way that it never causes choice to the user. More ever the limitation that has been prevailing in the existing system had been overcome to suit the need of the user.

High precision and care has been taken to design the data base, input forms an output reports since they should be given due importance which could otherwise to serious consequences thus affecting the whole system. The system thus developed has been implemented successfully which has been

performed to scrutinized the validation of each data and errors were spotted out and then finally cleared in a sophisticated manner.

The added feature of this system is that it has been provided with many provisions for future enhancement in order to maintain the system in such a way that the future requirement of the user could also be satisfied and upgraded.

REFERENCES

Arsanjani, A. (2004). how to identify, specify, and realize services for your SOA. *Service-oriented modeling and architecture*

Atkinson, B. (2015). How does online check in work?   , from
http://www.travelsupermarket.com/blog/how-does-online-check-in-work/

Bilotkack, V. and Rupp, N. . (2011). A guide to booking airline tickets online. *Mimeo,*

Uni-versity of California, Irvine.

C. Winston, S. Morrison. (1995). The Evolution of the Airline

Industry. *Brookings Institution Press*.

Crosby, T. (5th August, 2015). How Airline E-Tickets Work. from
http://adventure.howstuffworks.com/destinations/travel-guide/tips/how-airline-etickets-work.htm/

Erradi, A., Anand, S., Kulkani, N. (2006). An Architectural Framework for Service Definition and Realization. *On the proceedings of IEEE international conference on services computing, 6.*

GazetteLive. (2011). The benefits of booking flights online. from
http://www.gazettelive.co.uk/news/local-news/benefits-booking-flights-online3693877/

Jarvenpaa L. S, and Todd A. P. (1996). Consumer reactions to electronic shopping on

the World Wide Web. *International Journal of Electronic Commerce, Vol1(2)*, 59-88.

Johanson, M. (2014). How The Airline Industry Has Evolved In 100 Years of Commercial Air Travel.
from http://www.ibtimes.com/how-airline-industry-has-evolved-100-years-commercialair-travel-1524238/

Li X, Liu Z and He J. (2004). A Formal Semantics of UML Sequence Diagram. *Presented at and published in the proceedings of ASWEC2004.*

Rudstrom, A. and Fagerberg, P. (2004). Socially Enhanced Travel Booking. *a Case*

Study. Journal of Information Technology and Tourism, 6(3).

Shao, Q., Chen, Y., Tao, S., Yan, X. and Anerousis, N. (2008). a ticket

routing recommendation engine for enterprise problem resolution. *Proceedings of the*

VLDB Endowment, Vol 1(2).

Shon, Z., Chen, F. and Chang, Y. (2003). Airline e-commerce: the revolution in

ticketing channels. *Journal of Air Transport Management., Vol 9(5).*

Smith, M. J. (2002). The Airline Encyclopedia. *Scarecrow Press, New York. Microsoft Encarta*

   Premium

Winston, C., Morrison, S. (1995). The Evolution of the Airline Industry. *Brookings Institution Press.*